RÉPUBLIQUE FRANÇAISE
Liberté Égalité Fraternité

Université de Toulouse

Doctorat de l'Université de Toulouse

préparé à l'ISAE-SUPAERO

Estimation d'état sur groupes de Lie pour les systèmes robotiques sous incertitudes

Thèse présentée et soutenue, le 30 mars 2026 par

Vivien PRAVONG

École doctorale
SYSTEMES

Spécialité
Robotique et Automatique

Unité de recherche
ISAE-ONERA ACDC - Analyse, Commande Dynamique et Conception des systèmes

Thèse dirigée par
Jean-Philippe CONDOMINES et Gustav ÖMAN LUNDIN

Composition du jury
Mme Audrey GIREMUS, Présidente, Rapporteure, Université de Bordeaux
M. Gildas BESANÇON, Rapporteur, Grenoble INP
M. Karim DAHIA, Examinateur, ONERA Palaiseau
M. Silvère BONNABEL, Examinateur, Mines Paris - PSL
M. Jean-Philippe CONDOMINES, Directeur de thèse, ENAC
M. Gustav ÖMAN LUNDIN, Co-directeur de thèse, ONERA Toulouse

Membres invités
M. Philippe MOUYON, ONERA Toulouse
M. Stéphane PUECHMOREL, ENAC

Liberté • Égalité • Fraternité
RÉPUBLIQUE FRANÇAISE

Université de Toulouse

THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par : *l'Institut Supérieur de l'Aéronautique et de l'Espace (ISAE)*

Présentée et soutenue le *30/03/2026* **par :**

Vivien PRAVONG

State estimation on Lie groups for robotic systems under uncertainties

Estimation d'état sur groupes de Lie pour les systèmes robotiques sous incertitudes

---

JURY

Audrey GIREMUS	Université de Bordeaux	Rapporteure
Gildas BESANÇON	Grenoble INP	Rapporteur
Silvère BONNABEL	Mines Paris – PSL	Examinateur
Karim DAHIA	ONERA	Examinateur
Jean-Philippe CONDOMINES	ENAC	Directeur de thèse
Gustav ÖMAN LUNDIN	ONERA	Co-directeur de thèse

École doctorale et spécialité :
EDSYS : Robotique et Automatique

Unité de Recherche :
ISAE-ONERA ACDC - Analyse, Commande Dynamique et Conception des systèmes

Directeur(s) de Thèse :
Jean-Philippe CONDOMINES et *Gustav ÖMAN LUNDIN*

Rapporteur(e)s :
Audrey GIREMUS et *Gildas BESANÇON*

"All models are wrong, but some are useful."
"Tous les modèles sont faux, mais certains sont utiles."

— George Box, 1976

Abstract

This thesis focuses on the development of pose estimation techniques (including attitude and position) for mobile robotic systems operating with noisy and imperfect measurements. Classical filtering approaches, in particular Bayesian filtering methods and Lie group–based Kalman filters, form the methodological foundation of the thesis. Matrix Lie groups, notably SO(n) and SE(n), provide a mathematically sound framework for modeling rigid-body motion while naturally accounting for the intrinsic geometry of navigation problems. In this setting, uncertainty characterization and propagation are key elements in the design of effective Kalman filters. However, these approaches commonly rely on so-called "concentrated" Gaussian distribution assumptions, whose validity can be questionable in many engineering applications. The objective of this thesis is therefore to develop uncertainty modeling and propagation methods that are better suited to these spaces, with the aim of improving the performance and consistency of an Unscented Kalman Filter on Lie groups (UKF-LG).

A first research direction addresses the notion of covariance on the Lie group SE(n). Although covariance is widely used as a measure of dispersion for random variables, it may lose its relevance for SE(n) when its random variables are highly spread, especially regarding its angular components. This limitation becomes critical when uncertainties and noise are modeled using Gaussian probability densities, since their modeling typically rely on the notion of covariance. To overcome this issue, a redefinition of the probability density on the group is proposed, based on a hybrid distribution that combines a von Mises distribution for angular variables with a Gaussian distribution for translational components. This hybrid distribution is then incorporated into the UKF-LG framework, which is reformulated to enable its effective use.

A second line of research focuses on optimizing the placement of sigma-points in regimes where the notion of covariance remains valid and the Gaussian assumption holds, regardless of the nature of the random variables involved. This approach relies on a stochastic adjustment of the sigma-points through optimal quantization. In particular, a stochastic gradient descent algorithm is employed to minimize a cost function and enhance the performance of the standard UKF-LG.

These two methodological contributions led to the development of two variants of the UKF-LG. Both were experimentally validated through simulations on simulated data, as well as on real data drawn from an open source dataset.

Résumé 🇫🇷

Ce travail de thèse s'inscrit dans le développement de techniques d'estimation de pose (incluant l'attitude et la position) appliquées aux systèmes robotiques mobiles à partir de mesures bruitées et imparfaites. Les méthodes classiques de filtrage, en particulier le filtrage bayésien et plus spécifiquement les filtres de Kalman sur groupes de Lie, constituent le socle méthodologique de cette étude. Les groupes de Lie matriciels, notamment les groupes SO(n) et SE(n), offrent un cadre mathématique adapté à la modélisation du mouvement des corps rigides et permettent de prendre en compte la géométrie intrinsèque des problèmes de navigation. Dans ce contexte, la caractérisation et la propagation de l'incertitude jouent un rôle central dans la conception de filtres de Kalman performants. Toutefois, ces approches reposent le plus souvent sur des hypothèses de distributions gaussiennes dites "concentrées", dont la validité peut être remise en question dans de nombreux problèmes d'ingénierie. L'objectif de cette thèse est ainsi de proposer des méthodes de modélisation et de propagation de l'incertitude qui soient mieux adaptées à ces espaces, afin d'améliorer les performances et la consistance d'un filtre de Kalman sans-parfum sur groupe de Lie (UKF-LG).

Un premier axe de recherche porte sur la notion de covariance sur le groupe de Lie SE(n). Bien que largement utilisée comme mesure de dispersion des variables aléatoires, la covariance peut perdre sa pertinence pour un groupe de Lie lorsque ses variables aléatoires sont fortement dispersées, en particulier pour les composantes angulaires de SE(n). Cette limitation devient problématique lorsque les incertitudes et bruits sont modélisés par des densités de probabilité gaussiennes, car celles-ci reposent justement sur la notion de covariance. Pour répondre à cette problématique, une redéfinition de la densité de probabilité sur le groupe est proposée, reposant sur une distribution hybride combinant une loi de von Mises pour les variables angulaires et une distribution gaussienne pour les variables translationnelles. Cette distribution est ensuite intégrée au cadre de l'UKF-LG, lequel est reformulé afin de permettre son utilisation effective.

Un second axe de recherche concerne l'optimisation du placement des sigma-points dans un régime où la notion de covariance reste valide et où l'hypothèse gaussienne demeure pertinente, indépendamment de la nature des variables aléatoires considérées. Cette approche repose sur un réajustement aléatoire des sigma-points par quantification optimale. En particulier, un algorithme de descente de gradient stochastique visant à minimiser un critère de coût est employé afin d'améliorer les performances du UKF-LG standard.

Ces deux axes méthodologiques ont conduit au développement de deux variantes du UKF-LG. Elles ont été validées expérimentalement à travers des simulations sur données synthétiques, ainsi que sur des données réelles issues d'un jeu de données open source.

Remerciements 🇫🇷

Ce travail de thèse est le fruit de trois années de recherche au sein du complexe scientifique ONERA-ENAC-ISAE, basé à Toulouse. Il n'aurait pu voir le jour sans l'intervention de nombreuses personnes m'ayant accompagné, conseillé et aidé dans cette tâche qu'est la recherche du savoir, tout en m'ayant fait grandir sur le plan professionnel et individuel.

Aux directeurs de thèse...

Je tiens à remercier en premier lieu mes directeurs de thèse, Jean-Philippe CONDOMINES et Gustav ÖMAN LUNDIN, pour m'avoir proposé ce sujet riche en thématiques, mais aussi pour leur accompagnement, leur savoir-faire et leur savoir-être.

À Jean-Philippe, merci surtout pour la rigueur et l'exigence que tu m'as imposées lors de nos échanges, de l'écriture des publications jusqu'à l'écriture de ce manuscrit. Je pense sincèrement que sans cela, je n'aurais guère obtenu suffisamment de matière pour valider cette thèse. Face à ces exigences, j'admets avoir souvent eu quelques difficultés à m'y plier, mais tes encouragements m'ont aidé à ce que je me les impose à moi-même.

À Gustav, merci pour la patience et le temps que tu m'as accordé, et pour m'avoir permis de te solliciter très souvent à l'ONERA, que ce soit pour des questions techniques ou de simples demandes. Je te remercie aussi pour les nombreuses discussions enrichissantes que l'on a eu, que ce soit à propos de la recherche (le "savoir-faire" et "le faire savoir"), ou concernant la vie de tous les jours. Cela à toujours été très agréable de partager mes réflexions avec toi.

Aux contributeurs...

Je tiens ensuite à remercier Stéphane PUECHMOREL, co-auteur des trois papiers ayant été publiés dans le cadre de ces travaux de recherche. À toi Stéphane, j'admets avoir souvent eu des difficultés à comprendre ton langage de géomètre, mais ton expertise sur le sujet a été grandement bénéfique.

Je remercie ensuite Philippe MOUYON, dont l'expertise m'a permis de lever pas mal de doutes. Bien que tu avais sûrement d'autres tâches à gérer, tu m'as quand même accordé de ton précieux temps, et pour cela je t'en remercie.

Au jury et au comité de suivi...

Je remercie enfin les membres du jury, Audrey GIREMUS, Gildas BESANÇON, Karim DAHIA et Silvère BONNABEL, mais aussi les membres de mon comité de suivi, Patrick DANÈS et Yoko WATANABE, pour leur intérêt sincère à mes travaux de recherche. Tous ayant contribués, de près ou de loin, à la qualité finale de cette thèse et de ce manuscrit.

Aux collègues et aux amis...

Merci également au reste des collègues et amis, que ce soit de l'ONERA, de l'ENAC, ou de l'ISAE. Votre bonne humeur et votre présence, même silencieuse, m'ont permis de ne

pas me sentir accablé par le poids du doctorat. Finalement, j'ai eu l'occasion de grandir personnellement à vos côtés, et pour cela je vous en suis extrêmement reconnaissant.

Je pense notamment à mes contemporains de l'ONERA (Idriss, Tommaso, Tudor, Matisse, Elena, Dimitri, Cédric, Victor, Pierre, Noémie, Anthony, Franca, William, David, Guido, Jorge et j'en passe), de l'ENAC (Mohamad, Maël, Amélie, Leonardo et Axel) et de l'ISAE (Jason et Baptiste).

Je pense aussi aux membres des unités IDCO et AEI, que je laisse représentés par leurs chefs respectifs, Thierry LEMOING (désormais Mario CASSARO) et Cristelle CUMER. Merci pour votre présence et vos encouragements, en particulier de la part de Sofiane KRAIEM (co-directeur non-officiel par intérim) qui m'a encouragé à m'améliorer tant sur le plan professionnel que personnel.

À l'ENAC, je pense à l'équipe SYSDYN. Je remercie enfin Florence NICOL pour avoir organiser les séances de groupe de (Lie)cture qui m'ont été très bénéfiques pour ma compréhension du sujet.

À la famille...

On dit qu'il ne faut pas oublier d'où on vient. Donc pour finir, je remercie ma mère pour son soutien inconditionnel, de même que le reste de ma famille et mes amis de Villepinte, dont la présence simple et discrète a été un encouragement constant.

Contents

List of Figures **xiii**

List of Tables **xv**

List of Acronyms **xvi**

List of Symbols **xviii**

List of Publications **xix**

1 Introduction **1**
1.1 Context of the thesis 1
1.2 What are the flaws of Kalman filtering? 2
1.3 Objectives of the thesis 3
1.4 Organization of the manuscript 3

2 State of the Art **7**
2.1 Bayesian framework 8
2.2 Historical overview 14
2.3 Approximation of Bayesian filtering by linearization 17
2.4 Approximation of Bayesian filtering by sigma-points 20
2.5 An alternative solution to the problem of sigma-points adjustement 27
2.6 Extension to symmetry-preserving observers 34
2.7 Invariant observers on Lie groups 39

3 The Unscented Kalman Filter on Lie groups and its assumptions **43**
3.1 Matrix Lie groups for state estimation 44
3.2 Uncertainty modeling and Bayesian estimation on Lie groups 50
3.3 The UKF-LG equations 54
3.4 Limitations of Lie-theoretic uncertainty models and alternatives 62

4 Development of an uncertainty model for the UKF-LG **69**
4.1 Practical limitations of the Lie-theoretic covariance 70
4.2 Mathematical preliminaries on the Gauss-von Mises distribution 72
4.3 Development of an equivariant von Mises-Gaussian distribution for $\boldsymbol{SE}(n)$ 77
4.4 Adaptation of the UKF-LG to the von Mises-Gaussian distribution 82

5 Development of an optimal UKF-LG under valid Gaussian assumptions **91**
5.1 Limitations of the UT on Lie groups for valid Gaussian: numerical example 92
5.2 Recalls of the main results on optimal quantization 95
5.3 Development of an optimal quantization-based UKF-LG 96

6 Experimental validations **109**
6.1 Experimental setup for simulated data . . . 110
6.2 Validations of the vMG-UKF-LG on simulated data . . . 115
6.3 Validations of the OQ-UKF-LG on simulated data . . . 130
6.4 Experimental setup for real data . . . 137
6.5 Validations of the vMG-UKF-LG on real data . . . 140

7 Conclusions and Perspectives **149**
7.1 Main contributions . . . 149
7.2 Identified limits and drawbacks . . . 150
7.3 Perspectives . . . 151

A Probability Materials **153**
A.1 Usual definitions . . . 153
A.2 Proof of the recursive Bayesian filter . . . 155

B Additional details about Lie groups **156**
B.1 Demonstration of the group jacobians . . . 156
B.2 Usual matrix Lie groups . . . 157
B.3 Additional details for the kinematical model on $\boldsymbol{SE}(2)$ . . . 162
B.4 Evaluation metrics . . . 164
B.5 Additional results on the OQ-UKF-LG . . . 168

C Résumé étendu en Français 🇫🇷 - French summary **181**

List of Figures

2.1 Hidden Markov Model Structure. 10

2.2 Transformation of a random variable into a nonlinear function $\mathbf{y} = \mathbf{g}(\mathbf{x})$. Image taken from [Wan and Van Der Merwe, 2000]. 21

2.3 Quantization of a random vector $\mathbf{x} \in \mathbb{R}^d$ governed by a probability distribution $\mathcal{P}_\mathbf{x}$. The sample obtained from the N-quantizer $\mathcal{Q}_N$ captures the same distribution with a finite set of N supporting points. 28

2.4 Example of a state living in the 2-sphere. $\mathbf{x} = [\mathrm{x}_1, \mathrm{x}_2, \mathrm{x}_3]^\top$. 34

2.5 Orbits of the group action and definition of the function $\boldsymbol{\gamma}$. 37

2.6 Chronological timeline of invariant filtering (from early 2000s to nowadays). 42

3.1 Visual representation of a manifold of dimension $\dim(\boldsymbol{\mathcal{M}}) = d$. At each point $P \in \boldsymbol{\mathcal{M}}$, one can associate (locally) a tangent space $T_P\boldsymbol{\mathcal{M}}$ that "looks like" a usual Euclidean space $\mathbb{R}^d$. 44

3.2 Mappings between the manifold $\boldsymbol{G}$, the Lie algebra $\mathfrak{g}$ and the vector space $\mathbb{R}^d$. $[\cdot]^\wedge$ and $[\cdot]^\vee$ are the linear invertible maps defined by (3.1) and (3.2), respectively. The mapping $\exp / \log$ maps the Lie algebra to/from the manifold. 46

3.3 Evolution of a quantity $\boldsymbol{\chi}$ from time instant t to $t + \tau$. 46

3.4 Left-equivariant Gaussian uncertainty for a random variable $\boldsymbol{\chi} \in \boldsymbol{G}$. 52

4.1 Comparison between the sample variances (4.2) and (4.3), for $N = 10^4$ particles, as a function of σ_θ. 71

4.2 Illustration of the dispersion of an angle ξ_θ when: (blue) ξ_θ is treated as an unbounded random variable on $\mathbb{R}$ with a non-periodic distribution $\mathcal{N}(\xi_\theta)$; (red) ξ_θ is treated as a bounded random variable in $]-\pi, \pi]$ with a circular distribution $\mathcal{VM}(\xi_\theta)$. 71

4.3 A random variable $\theta \sim \mathcal{VM}(\theta; \overline{\theta}, \kappa)$ defined on the unit circle $\mathbb{S}$. The resultant $\frac{I_1(\kappa)}{I_0(\kappa)}$ measures the dispersion via the concentration parameter κ. If $\kappa \to +\infty$, the distribution is said highly concentrated and the resultant $\frac{I_1(\kappa)}{I_0(\kappa)} \to 1$. Inversely, if $\kappa \to 0$, the distribution is highly spread and the resultant $\frac{I_1(\kappa)}{I_0(\kappa)} \to 0$. 73

4.4 Prediction/Correction scheme of the revised UKF-LG. 85

5.1 Error ε, as a function of a ($b = 3.1623$), of the propagated covariance $\mathbf{P}_k^{[\emptyset]}$ in comparison to an accurate propagated covariance $\mathbf{P}_k$ computed from Monte-Carlo transformations. 94

5.2 Error ε, as a function of a ($b = 3.1623$), of the propagated variance of θ (left), ρ_1 (middle) and ρ_2 (right). The order of magnitude of the errors indicates that the angular variable contributes the most to the global error of each unscented method. 94

5.3 Standard deviation of the Monte-Carlo transformations along the θ-axis. This standard deviation indicates that the propagated covariance $\mathbf{P}_k^{MC}$ can be considered inside the validity domain of the Lie-theoretic definition. . . 94

5.4 Error $\varepsilon(\mathbf{P}_k^{UT1}, \mathbf{P}_k)$, as a function of a ($b = 3.1623$). UT1 is compared to its counterpart augmented with optimal quantization, whose parameters for these results are $\nu = 10$, $n_{max} = 10000$ (CLVQ is applied on the axes of θ, ρ_1 and ρ_2 only). 103

5.5 Error $\varepsilon(\mathbf{P}_k^{UT1}, \mathbf{P}_k)$, as a function of a ($b = 3.1623$), of the propagated variance of θ (left), ρ_1 (middle) and ρ_2 (right). UT1 is compared to its counterpart augmented with optimal quantization, whose parameters for these results are $\nu = 10$, $n_{max} = 10000$ (CLVQ is applied on the axes of θ, ρ_1 and ρ_2 only). 103

5.6 Distortion (computed using (5.10)) of the set of sigma-points (UT1) as a function of a, before and after applying the CLVQ. 104

5.7 Error $\varepsilon(\mathbf{P}_k^{CT}, \mathbf{P}_k)$, as a function of a ($b = 3.1623$). CT is compared to its counterpart augmented with optimal quantization, whose parameters for these results are $\nu = 10$, $n_{max} = 10000$ (CLVQ is applied on the axes of θ, ρ_1 and ρ_2 only). 104

5.8 Error $\varepsilon(\mathbf{P}_k^{CT}, \mathbf{P}_k)$, as a function of a ($b = 3.1623$), of the propagated variance of θ (left), ρ_1 (middle) and ρ_2 (right). CT is compared to its counterpart augmented with optimal quantization, whose parameters for these results are $\nu = 10$, $n_{max} = 10000$ (CLVQ is applied on the axes of θ, ρ_1 and ρ_2 only). 105

5.9 Distortion (computed using (5.10)) of the set of sigma-points (CT) as a function of a, before and after applying the CLVQ. 105

5.10 Error $\varepsilon(\mathbf{P}_k^{UT2}, \mathbf{P}_k)$, as a function of a ($b = 3.1623$). UT2 is compared to its counterpart augmented with optimal quantization, whose parameters for these results are $\nu = 100$, $n_{max} = 10000$ (CLVQ is applied on the axes of ρ_1 and ρ_2 only). 106

5.11 Error $\varepsilon(\mathbf{P}_k^{UT2}, \mathbf{P}_k)$, as a function of a ($b = 3.1623$), of the propagated variance of θ (left), ρ_1 (middle) and ρ_2 (right). UT2 is compared to its counterpart augmented with optimal quantization, whose parameters for these results are $\nu = 100$, $n_{max} = 10000$ (CLVQ is applied on the axes of ρ_1 and ρ_2 only). 106

5.12 Distortion (computed using (5.10)) of the set of sigma-points (UT2) as a function of a, before and after applying the CLVQ. 107

6.1 Example of a pose estimation problem with landmarks detection. The position of landmarks with respect to the fixed inertial frame $\mathcal{F}_{\mathcal{I}}$ is known, but at time instant k, the system measures only the position of landmarks with respect to its own reference frame $\mathcal{F}_{\mathcal{B}}$. 111
6.2 Sequence of nominal inputs $\mathbf{u}$ used to generate the nominal trajectory. The data u_1, u_4 and u_5 have been obtained from gyrometer and odometry measurements of a holonomic ground robot moving in a flat surface. 112
6.3 State evolution and estimation by Kalman filtering on $\boldsymbol{SE}(3)$. The sequence $\{\mathbf{u}_k\}_{k\geq 1}$ drives the shape of the trajectory. For simulated data, the noise vectors $\mathbf{w}_k$ and $\mathbf{v}_k$ are generated from a pseudo-random number generator with chosen characteristics $\mathbf{Q}_k$ and $\mathbf{R}_k$. The Kalman prediction and correction are parameterized by $\hat{\mathbf{Q}}_k$ and $\hat{\mathbf{R}}_k$, respectively. 113
6.4 Monte-Carlo simulations provide different possible realizations of the same nominal trajectory, where the different realizations are spanned by $\mathcal{P}(\mathbf{w}_k; \mathbf{0}, \mathbf{Q}_k)$ and $\mathcal{P}(\mathbf{v}_k; \mathbf{0}, \mathbf{R}_k)$. 114
6.5 Approximation of the function $\vartheta(\sigma^2)$ relating a variance σ^2 with a concentration parameter κ, using an exponential model of the form (6.7). 116
6.6 Evolution of the log-Euclidean error as a function of time. 117
6.7 Evolution of the log-Euclidean error as a function of time. 118
6.8 RMSE in rotation (left plot) and in position (right plot), as a function of the noise level a. For each noise level a, the median of the RMSE is plotted along with its confidence interval. 119
6.9 Estimation error as a function of time k. The green curves are the actual 3σ standard deviations of the estimation error. The blue curves are the variances given by the filter (averaged over the N_s simulations) in terms of 3σ standard deviation. The black curve is the mean of the estimation error. The reported results are obtained from Monte-Carlo runs corresponding to the noise level $a = 31.7$. 120
6.10 AvMS of the filters on the variable ω_1 as a function of time instant k. . . . 121
6.11 RMSE in rotation (left plot) and in position (right plot) as a function of the noise level a (with observation model $\mathbf{h}_{\text{gps}}$). 122
6.12 RMSE in rotation (left plot) and in position (right plot) as a function of the noise level a. 123
6.13 RMSE in rotation (left plot) and in position (right plot) as a function of the noise level a. 124
6.14 RMSE in rotation (left plot) and in position (right plot) as a function of the noise level a. 125
6.15 RMSE in rotation (left plot) and in position (right plot) as a function of the noise level a. 126
6.16 Cramér-Rao lower bounds and efficiency performance of the filters. 128

6.17 RMSE in rotation (left) and in position (right) as a function of the noise level a (b fixed). 131

6.18 RMSE in rotation (left) and in position (right) as a function of the noise level b (a fixed). 131

6.19 L^2-norm of the log-Euclidean errors (in rotation and in position) as a function of time, corresponding to a single simulation ($a = 50$, $b = 50$). 132

6.20 Evolution of the variance for ($a = 50$, $b = 50$). The green curves are the 3σ standard deviations of the estimation errors of the Monte-Carlo runs. The blue and pink curves are the variances returned by the UKF-LG and the OQ-UKF-LG, respectively, averaged over the N_s simulations, in terms of 3σ standard deviation. The black curve is the mean of the estimation errors. 132

6.21 RMSE in rotation (left) and in position (right) as a function of the noise level a (b fixed). 133

6.22 RMSE in rotation (left) and in position (right) as a function of the noise level a (b fixed). 134

6.23 RMSE in rotation (left) and in position (right) as a function of the noise level b (a fixed). 135

6.24 Performance gain as a function of computation time. The first point corresponds to the performance given by the UKF-LG. The remaining points correspond to the performance given by the OQ-UKF-LG with different parameters n_{max}. 136

6.25 Presentation of the robotic system. Images taken from [Burri et al., 2016]. 137

6.26 Presentation of the environment and scenarios. Images taken from [Burri et al., 2016]. 138

6.27 Motion characteristics of the difficult scenario over a window of 20 seconds of flight. 141

6.28 [Difficult] Attitude error along 20 seconds of flight. 142

6.29 [Difficult] Log-rotation error of the variable ω_i, $\forall i = \{1, 2, 3\}$, as a function of time. These errors are those obtained from using the UKF-LG and the vMG-UKF-LG. The dotted lines are the variances given by the filters in terms of 3σ standard deviations. 143

6.30 [Difficult] Position error along 20 seconds of flight. 143

6.31 Position error of the variable x_i, $\forall i = \{1, 2, 3\}$, as a function of time. These errors are those obtained from using the UKF-LG and the vMG-UKF-LG. The dotted lines are the variances given by the filters in terms of 3σ standard deviations. 144

B.1 The NEES is computed along the trajectory (for all k), and quantifies the consistency of the estimated covariance $\mathbf{P}_k$ with respect to the squared log-Euclidean estimation error (indicated by the covariance of the estimation error $\tilde{\mathbf{P}}_k$). If $b_1 < \text{NEES}(\mathbf{P}_k, \tilde{\mathbf{P}}_k) < b_2$, then the filter is consistent: its estimation of the error correctly matches the true error. If $\text{NEES}(\mathbf{P}_k, \tilde{\mathbf{P}}_k) < b_1$, the filter is conservative: the error is estimated larger than it truely is. If $\text{NEES}(\mathbf{P}_k, \tilde{\mathbf{P}}_k) > b_2$, the filter is inconsistent: the filter estimates the error to be lower than it is in reality. 166
B.2 Nominal inputs and trajectory for the OQ-UKF-LG on $\boldsymbol{SE}(2)$. 169
B.3 L^2-norm of the log-Euclidean errors (in rotation and in position) as a function of time 170
B.4 L^2-norm of the log-Euclidean errors (in rotation and in position) as a function of time. $f_{obs} = 1$ Hz, $\hat{\mathbf{Q}} = 50^2 \times \mathbf{Q}$, $\hat{\mathbf{R}} = \mathbf{R}$. CLVQ on angular dimension with $\gamma_{1,n} = \frac{\sqrt{P_\theta}}{n}$. 172
B.5 Log-Euclidean errors on all three state variables θ, x_1 and x_2, as a function of k. 173
B.6 Comparison of the ANEES between (a) the UKF-LG and (b) the OQ-UKF-LG. $f_{obs} = 1$ Hz, $\hat{\mathbf{Q}} = 50^2 \times \mathbf{Q}$, $\hat{\mathbf{R}} = \mathbf{R}$. 174
B.7 RMSE of all Monte-Carlo simulations in rotation and in position, as a function of the noise level a, with b fixed. For each noise level a, the average Monte-Carlo of the RMSE is plotted along with its confidence interval (upper-half) in the colored area. 174
B.8 RMSE of all Monte-Carlo simulations in rotation and in position, as a function of the noise level b, with a fixed. For each noise level b, the average Monte-Carlo of the RMSE is plotted along with its confidence interval (upper-half) in the colored area. 175
B.9 RMSE of the OQ-UKF-LG as a function of the noise level a, and for different values of n_{max}. 176
B.10 RMSE of the OQ-UKF-LG as a function of the noise level a, and for different values of m. 177
B.11 RMSE of all Monte-Carlo simulations in rotation and in position, as a function of the noise level a, with b fixed, and for the observation model $\mathbf{h}_{\text{lidar}}$. 178
B.12 RMSE of all Monte-Carlo simulations in rotation and in position, as a function of the noise level b, with a fixed, and for the observation model $\mathbf{h}_{\text{lidar}}$. 178
B.13 RMSE of all Monte-Carlo simulations in rotation and in position, as a function of the noise level a, with b fixed, and for the observation model $\mathbf{h}_{\text{lidar}}$. 179
B.14 RMSE of all Monte-Carlo simulations in rotation and in position, as a function of the noise level b, with a fixed, and for the observation model $\mathbf{h}_{\text{lidar}}$. 179

B.15 Evolution of the error as a function of time. 180

List of Tables

2.1 Three different sets of parameters for the sigma-point transformation . . . 26

4.1 Set of $2d+1$ (unspecified) nodes and weights for the state space $\mathbb{S} \times \mathbb{R}^\ell$. These are all symmetric nodes centered around zero. The spread in ϕ is given by $\pm\eta$, and the spread in z_j is given by $\pm\alpha$, $\forall j = 1, \ldots, \ell$. 75

4.2 Constraints on the nodes and weights [Horwood and Poore, 2014]. 76

6.1 Performance gain VS computation cost of the vMG-UKF against the UKF-LG. 129

6.2 [Difficult] RMSE in rotation (°) for three independent simulations. For each simulation, the best performance is highlighted in green. 141

6.3 [Difficult] RMSE in position (m) for three independent simulations. For each simulation, the best performance is highlighted in green. 144

6.4 [Medium] RMSE in rotation (°) for three independent simulations. For each simulation, the best performance is highlighted in green. 145

6.5 [Medium] RMSE in position (m) for three independent simulations. For each simulation, the best performance is highlighted in green. 145

6.6 Execution times of the different filters for the prediction and update. The indicated values corresponded to the average duration for 20 seconds of flight of the medium scenario. 146

6.7 [Easy] RMSE in rotation (°) for three independent simulations. For each simulation, the best performance is highlighted in green. 147

6.8 [Easy] RMSE in position (m) for three independent simulations. For each simulation, the best performance is highlighted in green. 147

B.1 Some examples of Lie groups and their characteristics. 156

B.2 Root Mean (over the entire trajectory) Square Error for the results of Figure B.3. 169

B.3 RMSE (over the entire trajectory), for different overestimations of the true covariance $\mathbf{Q}$. The value in percentage corresponds to the relative difference between the RMSE of the OQ-UKF-LG and the RMSE of the UKF-LG, taken as the reference. 170

B.4 RMSE (over the entire trajectory) for an underestimation of the true covariance $\mathbf{Q}$. The value in percentage corresponds to the relative difference between the RMSE of the OQ-UKF-LG and the RMSE of the UKF-LG, taken as the reference. 171

List of Acronyms

BCH Baker-Campbell-Hausdorff. 49, 157

CLVQ Competitive Learning Vector Quantization. 32, 100

CRLB Cramér-Rao Lower Bounds. 167

IMU Inertial Measurement Unit. 1

NEES Normalized Estimation Error Squared. 165

RMSE Root Mean Square Error. 164

UAV Unmanned Aerial Vehicle. 1

UGV Unmanned Ground Vehicle. 1

UKF-LG Unscented Kalman Filter on Lie Groups. 4, 53

UT Unscented Transformation. 20

vMG von Mises-Gaussian. 79

List of Symbols

General

$\boldsymbol{G}$	Lie group	–
$\boldsymbol{\chi}$	State matrix (element of $\boldsymbol{G}$)	–
$\mathbf{x}$	State vector $\in \mathbb{R}^d$	–
$\hat{\boldsymbol{\chi}}/\hat{\mathbf{x}}$	Estimated state (matrix/vector form)	–
$\mathbf{P}$	Covariance matrix of the estimated state	–
$\mathfrak{g}$	Lie algebra of $\boldsymbol{G}$	–
$\boldsymbol{\xi}$	Exponential coordinates vector $\in \mathbb{R}^{\dim(\boldsymbol{G})}$	–
$[\cdot]^\wedge$	*wedge* operator $[\cdot]^\wedge : \mathbb{R}^{\dim(\boldsymbol{G})} \mapsto \mathfrak{g}$	–
$[\cdot]^\vee$	*vee* operator $[\cdot]^\vee : \mathfrak{g} \mapsto \mathbb{R}^{\dim(\boldsymbol{G})}$	–
$\mathbf{u}$	Input vector $\in \mathbb{R}^q$	–
$\mathbf{y}$	Measurement/observation/output vector $\in \mathbb{R}^p$	–
$\mathbf{P}_\mathbf{y}$	Covariance matrix of vector $\mathbf{y}$	–
$\mathcal{F}_\mathcal{I}$	Inertial frame	–
$\mathcal{F}_\mathcal{B}$	Body-fixed frame	–
Ad	Adjoint operator of $\boldsymbol{G}$	–
ad	adjoint operator of $\mathfrak{g}$	–
$\boldsymbol{\mathcal{J}}_{\mathsf{L/R}}$	Left/Right Jacobian of $\boldsymbol{G}$	–
$\mathbf{J}_{\mathsf{L/R}}$	Left/Right Jacobian of $\boldsymbol{SO}(n)$	–

2D case

$\boldsymbol{SO}(2)$	2D rotation group	–
$\boldsymbol{SE}(2)$	2D motion group	–
θ	Body orientation	rad
$\boldsymbol{x} = [x_1, x_2]^\top$	Position vector w.r.t. $\mathcal{F}_\mathcal{I} = (O, \boldsymbol{e}_{\mathcal{I}_1}, \boldsymbol{e}_{\mathcal{I}_2})$	m

$\boldsymbol{v} = [v_1, v_2]^\top$ Velocity vector w.r.t. $\mathcal{F}_\mathcal{I} = (O, \boldsymbol{e}_{\mathcal{I}_1}, \boldsymbol{e}_{\mathcal{I}_2})$ $\mathrm{m.s}^{-1}$

$\boldsymbol{\mathcal{R}}(\theta)$ Rotation matrix $\in \boldsymbol{SO}(2)$ associated to θ –

$\mathfrak{so}(2)$ Lie algebra of $\boldsymbol{SO}(2)$ –

$\mathfrak{se}(2)$ Lie algebra of $\boldsymbol{SE}(2)$ –

$\boldsymbol{\xi} = [\theta, \rho_1, \rho_2]^\top$ Exponential coordinates vector –

3D case

$\boldsymbol{SO}(3)$ 3D rotation group –

$\boldsymbol{SE}(3)$ 3D motion group –

$\boldsymbol{x} = [x_1, x_2, x_3]^\top$ Position vector expressed in $\mathcal{F}_\mathcal{I} = (O, \boldsymbol{e}_{\mathcal{I}_1}, \boldsymbol{e}_{\mathcal{I}_2}, \boldsymbol{e}_{\mathcal{I}_3})$ m

$\boldsymbol{v} = [v_1, v_2, v_3]^\top$ Velocity vector expressed in $\mathcal{F}_\mathcal{I} = (O, \boldsymbol{e}_{\mathcal{I}_1}, \boldsymbol{e}_{\mathcal{I}_2}, \boldsymbol{e}_{\mathcal{I}_3})$ $\mathrm{m.s}^{-1}$

$\boldsymbol{\mathcal{R}}$ Rotation matrix $\in \boldsymbol{SO}(3)$ –

$\mathfrak{so}(3)$ Lie algebra of $\boldsymbol{SO}(3)$ –

$\mathfrak{se}(3)$ Lie algebra of $\boldsymbol{SE}(3)$ –

$\boldsymbol{\xi} = [\boldsymbol{\omega}^\top, \boldsymbol{\rho}^\top]^\top$ Exponential coordinates vector –

$\boldsymbol{\rho}$ Linear/Translational part –

$\boldsymbol{\omega}$ Angular part –

List of Publications

- Pravong et al., 2024. V. Pravong, J. P. Condomines, G. Öman Lundin and S. Puechmorel; Unscented Kalman Filter using Optimal Quantization; 27th International Conference on Information Fusion (FUSION), Venice, Italy, 2024
- Pravong et al., 2025a. V. Pravong, J. P. Condomines, G. Öman Lundin and S. Puechmorel; An Optimal Quantization-based Unscented Kalman Filter on SE(2): Application to Trajectory Tracking; 23rd European Control Conference (ECC), Greece, Thessaloniki, 2025
- Pravong et al., 2025b. V. Pravong, J. P. Condomines, G. Öman Lundin and S. Puechmorel; An Equivariant von Mises-Gaussian distribution on SE(n) for Unscented Kalman Filtering; 64th Conference on Decision and Control (CDC), Brazil, Rio de Janeiro, 2025

Chapter 1

Introduction

Contents

1.1 Context of the thesis 1
1.2 What are the flaws of Kalman filtering? 2
1.3 Objectives of the thesis 3
1.4 Organization of the manuscript 3

1.1 Context of the thesis

This thesis, co-funded by the ONERA–ISAE–ENAC research federation and conducted in Toulouse (France), addresses problems arising in the field of navigation, including terrestrial navigation with Unmanned Ground Vehicles (UGV) and aerial navigation with Unmanned Aerial Vehicles (UAV). The rapid growth of increasingly demanding applications, whether civilian (e.g., autonomous drone delivery, inspection of hazardous areas, search-and-rescue operations) or military (e.g., coordinated drone fleets, reconnaissance missions, electronic warfare environments), requires robotic systems to achieve a high level of autonomy in mission execution.

A central requirement for such systems is the ability to reliably estimate their own state over time. For mobile robots, this state generally includes position and orientation (collectively referred to as the pose), as well as additional quantities such as velocity or sensor biases. Accurate pose estimation is a prerequisite for higher-level tasks such as planning, control, and decision-making, since state-feedback control strategies critically depend on the quality of the estimated state.

In practice, state estimation must rely on noisy and imperfect measurements provided by onboard sensors. Constraints on mass, volume, and power consumption limit the range and quality of instruments that can be embedded, particularly in mobile and aerial platforms, often embedded with Inertial Measurement Units (IMU), lidar, sonar, barometers, etc., all of which provide partial measurements of the state. This motivates the use of

data fusion techniques capable of combining multiple information sources while explicitly accounting for their respective uncertainties.

Among the various approaches to state estimation, Kalman filtering, originally introduced in 1960, and its nonlinear extensions, continuously developed since then, have become standard tools in engineering applications. More recently, the formulation of Kalman filters directly on matrix Lie groups has gained significant attention over the past two decades. By exploiting the underlying geometric structure of rigid-body motion, Lie group–based filters offer a systematic way to represent pose variables and to preserve the intrinsic symmetries of the estimation problem. In particular, filters defined on groups such as $\boldsymbol{SO}(n)$ and $\boldsymbol{SE}(n)$ have demonstrated improved consistency and robustness compared to classical Euclidean formulations in many navigation scenarios.

1.2 What are the flaws of Kalman filtering?

Despite its widespread success and long-standing use in engineering applications, Kalman filtering relies on assumptions that can become restrictive in modern navigation problems. These limitations are particularly visible in nonlinear settings and in applications involving geometric state spaces, such as pose estimation for mobile robots.

In general, Kalman filtering fundamentally relies on mathematical models that describe both the system dynamics and the relationship between the system state and the sensor measurements. These models are typically derived from physical principles and are therefore idealized representations of reality. In practice, they only approximate the true system behavior, as unmodeled dynamics, external disturbances, and sensor imperfections inevitably affect real-world systems. In addition, Kalman filters require the specification and tuning of several parameters, most notably those characterizing process and measurement noise. These parameters are intended to encode the magnitude of uncertainty affecting the system, yet this uncertainty is itself largely unknown.

This uncertainty is most often characterized by probabilistic assumptions that are only approximately satisfied in practice. Most formulations rely on Gaussian probability models and on second-order statistics, notably the covariance matrix, to characterize uncertainty. While these assumptions are well justified for linear systems with small uncertainties, their validity becomes questionable when uncertainties grow, nonlinearities are strong, or the underlying state space is a non-Euclidean manifold.

The development of Kalman filters on matrix Lie groups was motivated precisely by the need to account for the intrinsic geometry of pose variables. By formulating the estimation problem directly on groups such as $\boldsymbol{SO}(n)$ or $\boldsymbol{SE}(n)$, these filters preserve important invariance properties and avoid many inconsistencies associated with Euclidean representations. However, although Lie group–based Kalman filters provide a sound ge-

ometric framework, they largely retain the probabilistic machinery of classical Kalman filtering. In particular, uncertainty is still commonly represented through a Lie-theoretic covariance defined in a tangent space, together with a concentrated Gaussian assumption.

From an application standpoint, these theoretical limitations translate into practical difficulties. Real-world robotic systems often operate in conditions where uncertainties are large and models are only approximate. In such situations, standard Kalman filtering techniques (whether formulated in Euclidean spaces or on Lie groups), may exhibit degraded performance, loss of consistency, or sensitivity to tuning parameters.

These observations highlight the need for improved uncertainty modeling and propagation mechanisms within the Kalman filtering framework. Addressing these challenges, while preserving the favorable structure and computational efficiency of Kalman-based methods, constitutes the main motivation in this thesis.

1.3 Objectives of the thesis

This thesis focuses on the development of probabilistic tools aimed at enhancing the accuracy and the consistency of such Lie group-based Kalman filters.

Hence, the objectives to be achieved by the end of this thesis are:

1. The development of solutions to address the problem of estimating the pose of a rigid body when large uncertainties are involved. In particular, our interest will focus on the stochastic approach under Lie group-based Kalman filtering methods, and focuses on the accurate representation and propagation of uncertainty for such mathematical objects.
2. The validation of the developed estimation schemes on simulations. This step will mainly rely on simulated data that will be purposely polluted by noise using an arbitrary choice of probability density functions. This will allow for the assessment of the filter's performances under the right noise assumption.
3. The validation of the developed estimation schemes on real experiments. This step will rely on experimental data coming from pretreated open source datasets.

1.4 Organization of the manuscript

The present manuscript is divided into five chapters:

■ Chapter 2 provides a state of the art of the theoretical foundations and existing methodologies relevant to Bayesian state estimation, with a progressive focus on Kalman filtering techniques. It begins by introducing the general Bayesian filtering framework, thereby establishing the probabilistic principles underlying modern estimation algorithms.

The chapter then reviews classical suboptimal approximations of Bayesian filtering in Euclidean spaces, first through linearization-based methods and subsequently through sigma-point–based approaches. Particular attention is given to the Unscented Kalman Filter and to the practical limitations associated with the placement and adjustment of sigma-points, which motivate the introduction of optimal quantization as a possible alternative strategy. Building on these Euclidean developments, the chapter then shifts perspective toward symmetry-preserving estimation. It introduces invariant observers theory and Kalman filtering formulations on Lie groups, highlighting how geometric structure and system symmetries can be explicitly exploited in nonlinear estimation problems. This progression sets the conceptual groundwork for the framework adopted in the subsequent chapters.

■ Chapter 3 builds directly on the conclusions of the state of the art by adopting the Unscented Kalman Filter on Lie Groups (UKF-LG) as the core estimation framework of the thesis, with a particular emphasis on uncertainty representation and propagation aspects. The chapter first recalls essential notions related to matrix Lie groups and motivates their use for state estimation problems. It then details how random variables and uncertainty can be modeled on such manifolds, before presenting the structure and equations of the UKF-LG. The chapter concludes with a critical discussion of the limitations inherent to Lie-theoretic uncertainty modeling, especially those associated with covariance-based and concentrated Gaussian assumptions, and outlines potential alternative formulations that motivate the developments introduced in the subsequent chapters.

■ Chapter 4 presents the first contribution of the thesis and directly builds on the limitations identified at the end of Chapter 3. It focuses on addressing the shortcomings of uncertainty representation on Lie groups, in particular on $\boldsymbol{SE}(n)$ (which involves angular periodic variables through $\boldsymbol{SO}(n)$), by adopting a modeling-oriented aspect. The chapter investigates alternatives to the classical Lie-theoretic covariance and proposes a revised probabilistic description better suited to angular variables. It begins by illustrating, on a simple example, the practical limitations of covariance-based uncertainty descriptions and introduces the notion of concentration parameters together with von Mises distributions as an alternative. Building on these ideas, the chapter then reviews the Gauss–von Mises distribution from the literature and uses it as a foundation to develop an equivariant von Mises–Gaussian probability model tailored to $\boldsymbol{SE}(n)$ and its extensions. Finally, the Unscented Kalman Filter on Lie groups is reformulated so as to incorporate this new uncertainty model within its prediction and correction steps, resulting in a von Mises-Gaussian UKF on Lie groups (vMG-UKF-LG), thereby enabling consistent filtering with the proposed probability density.

■ Chapter 5 develops the second contribution of the thesis, but rather focuses on numerical aspects of uncertainty propagation in the UKF-LG. Building on the issues highlighted in Chapter 2 regarding sigma-point placement, and assuming a valid Lie-

theoretic covariance together with a valid concentrated Gaussian model, the chapter first illustrates, through a numerical example, deficiencies in the standard Unscented Transform on Lie groups and motivates the need for improved sigma-point placement. It then recalls the known results and theorems of optimal quantization, already detailed in Chapter 2, and adapts them to the problem of sigma-point generation for uncertainty propagation on Lie groups. As a by-product, it demonstrates numerically how it can mitigate the issues identified earlier in the chapter. Based on this adaptation, we propose an optimal quantization–based UKF-LG (OQ-UKF-LG), which integrates the resulting sigma-point construction into the standard UKF-LG framework without reformulating its overall structure. The chapter concludes by highlighting that, although developed for Gaussian vectors, the approach could in principle benefit to filtering methods based on alternative probability models, including the vMG-UKF-LG from Chapter 4, offering promising avenues for future work.

■ Chapter 6 presents a comprehensive set of experimental results aimed at validating the two filtering algorithms developed in Chapters 4 and 5, namely the vMG-UKF-LG and the OQ-UKF-LG. Since these two approaches rely on different probabilistic assumptions, their validation is carried out through distinct experimental protocols, but based on the same experimental setup. The chapter first focuses on simulated data, which allows full control over the noise characteristics and ensures that the underlying assumptions are satisfied for each developed filter. To this end, a complete simulation framework is built, enabling the generation of trajectories and measurements consistent with the assumed noise models. Within this controlled setting, the vMG-UKF-LG is evaluated under von Mises and Gaussian noise conditions, while the OQ-UKF-LG is assessed under purely Gaussian noise. Both filters are compared with the standard UKF-LG as a benchmark algorithm. The chapter then extends the validation to real-world data, using the publicly available EuRoC dataset. An experimental setup tailored to this dataset is introduced, and the performance of the vMG-UKF-LG is evaluated in realistic conditions. The OQ-UKF-LG is also tested on the EuRoC dataset; however, its results are reported separately in the appendix. Overall, this chapter provides a thorough empirical assessment of the proposed methods, highlighting their respective domains of relevance and practical performance in both simulated and real navigation scenarios.

Chapter 2

State of the Art

Contents

2.1 Bayesian framework . . . 8
2.1.1 Formalization of the stochastic dynamical system . . . 8
2.1.2 General solution to the problem of stochastic filtering . . . 9
2.2 Historical overview . . . 14
2.2.1 Pioneering works . . . 14
2.2.2 Overview on Bayesian suboptimal filtering . . . 15
2.3 Approximation of Bayesian filtering by linearization . . . 17
2.3.1 The Bayesian filter under Gaussian assumptions . . . 17
2.3.2 The linear Kalman filter . . . 18
2.3.3 The Extended Kalman filter . . . 19
2.4 Approximation of Bayesian filtering by sigma-points . . . 20
2.4.1 Approximation of moment integrals by quadrature . . . 21
2.4.2 The Unscented Kalman filter . . . 23
2.4.3 The problem of sigma-points adjustment . . . 24
2.5 An alternative solution to the problem of sigma-points adjustement . . . 27
2.5.1 Optimal quantization of a random vector . . . 28
2.5.2 Numerical integration by optimal quantization . . . 29
2.5.3 Stochastic gradient descent . . . 30
2.5.4 Optimal quantization of a normally-distributed random vector . . . 31
2.6 Extension to symmetry-preserving observers . . . 34
2.6.1 Introductive example . . . 34
2.6.2 Invariant systems . . . 35
2.6.3 Symmetry-preserving observers . . . 36
2.7 Invariant observers on Lie groups . . . 39
2.7.1 Kalman filters on Lie groups . . . 39
2.7.2 Other Lie group-based observers . . . 40

2.1 Bayesian framework

In many estimation and navigation problems, the system of interest evolves under uncertain dynamics and observed through noisy measurements. The Bayesian framework provides a rigorous and general way to formalize such problems by modeling both the state evolution and the observation as random processes. The objective is to infer, at each time step, the posterior distribution of the state given all past observations.

2.1.1 Formalization of the stochastic dynamical system

Estimating the state of a dynamical system requires a model of its evolution. A purely deterministic description is often insufficient, as unknown phenomena (e.g., sensor noise, actuator faults, or external disturbances) affect the state evolution. Stochastic processes offer a framework to represent these uncertainties without explicitly modeling their physical sources. Moreover, deterministic models rarely capture all dynamics: some input-output relationships are difficult to characterize, and other influences may simply be unknown. Thus, even an accurate deterministic model remains an idealized approximation of the dominant behavior, whereas a stochastic formulation explicitly accounts for neglected and uncertain effects, yielding a more realistic representation of the system.

Dynamical system

We then consider a nonlinear system in discrete-time (the continuous-time case may be handled via stochastic differential equations and suitable discretization). In the classical Euclidean setting, the stochastic dynamical model is written as

$$\begin{aligned} \mathbf{x}_k &= \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1}, \tau) \\ \mathbf{y}_k &= \mathbf{h}(\mathbf{x}_k, \mathbf{u}_k, \mathbf{v}_k) \end{aligned} \tag{2.1}$$

where $\mathbf{x}_k \in \mathbb{R}^d$ is the vector state at instant time indexed by $k \in \mathbb{N}$, and $\mathbf{u}_k \in \mathbb{R}^q$ denotes known inputs or control commands applied during a small delay τ between time instants $k-1$ and k. The timestep τ is often implicit in such discrete formulations, and therefore can be removed for notational convenience. The vector $\mathbf{y}_k \in \mathbb{R}^p$ is a measurement (or observation) available at time k, while $\mathbf{w}_k$ and $\mathbf{v}_k$ are the model and measurement noises, respectively. Hence, the model $\mathbf{f}$ describes the time evolution of the system's state, and the observation model $\mathbf{h}$ describes a set of sensors that provides (partial, complete or indirect) information on the current state of the said system.

State estimation under uncertainty aims to accurately approximate the true state of a system from a sequence of noisy measurements, given a model that is itself subject to unknown perturbations and imperfect knowledge of the underlying dynamics. Hence, it is a process that involves the filtering of the noise coming from sensors.

Nature of uncertainty

In practice, all sources of uncertainty in a stochastic dynamical model are represented by random variables with known or assumed probability distributions. These random variables capture various forms of uncertainty affecting the system, including:

- unknown external disturbances (such as wind, slippage, or turbulence),
- imperfect sensor measurements,
- model mismatch or discretization errors arising from approximate numerical representations.

This stochastic formulation thus provides a unified way to account for both epistemic uncertainty (i.e., stemming from incomplete knowledge of the system dynamics or measurement models) and aleatory uncertainty, associated with the variability and randomness caused by the environment.

Problem definition

From the stochastic perspective adopted above, the system state and noise processes are modeled as random variables, whose uncertainties are described by probability density functions (pdfs), denoted by $\mathcal{P}$. Within this framework, the estimation problem can be formulated in probabilistic terms as the computation of the conditional probability density of the state given all past and current observations. Let $\mathbf{y}_{0:k} = \{\mathbf{y}_0, \ldots, \mathbf{y}_k\}$ denote the sequence of measurements available up to time k. The objective is then to determine the conditional pdf $\mathcal{P}(\mathbf{x}_k \mid \mathbf{y}_{0:k})$. This conditional density summarizes all available information about the system state, combining prior knowledge provided by the dynamical model with information brought by the observations.

Accordingly, the estimation problem can be viewed as a statistical inference task: given a sequence of noisy measurements generated by imperfect sensors and corrupted by stochastic disturbances, the goal is to infer the underlying (unknown) system state $\mathbf{x}_k$ at each time step.

2.1.2 General solution to the problem of stochastic filtering

It turns out that the computation of $\mathcal{P}(\mathbf{x}_k|\mathbf{y}_{0:k})$ has an explicit solution, which is given by the Bayesian filter. First, it requires a generic probabilistic modeling of the system dynamics.

Hidden markov Process

From a more general point of view, instead of functions $\mathbf{f}$ and $\mathbf{h}$, we can consider a model solely based on the pdf of $\mathbf{x}$ and $\mathbf{y}$[1]. The stochastic process $\mathbf{x}$ is assumed to possess the Markov property (2.2):

$$\mathcal{P}(\mathbf{x}_k|\mathbf{x}_{k-1}, ..., \mathbf{x}_0) = \mathcal{P}(\mathbf{x}_k|\mathbf{x}_{k-1}) \tag{2.2}$$

[1]For notational convenience, the noise processes $\mathbf{w}$ and $\mathbf{v}$ (affecting $\mathbf{x}$ and $\mathbf{y}$ respectively) are implicit.

This means that the realization $\mathbf{x}_k$ at time instant k contains all information about the past. Therefore, it is sufficient to use the last available information about the state.

Given a stochastic process verifying the Markov property, a Hidden Markov Model (HMM) is defined by:

$$\begin{aligned} \mathbf{x}_k &\sim \mathcal{P}(\mathbf{x}_k|\mathbf{x}_{k-1}) \ \text{(state transition density)} \\ \mathbf{y}_k &\sim \mathcal{P}(\mathbf{y}_k|\mathbf{x}_k) \ \text{(likelihood)} \end{aligned} \tag{2.3}$$

The pdf $\mathcal{P}(\mathbf{x}_k|\mathbf{x}_{k-1})$ describes the evolution of the state over time. It can be of any form and can include nonlinearities. The stochastic process $\mathbf{x}$ is an unobserved (hence the term "hidden") Markov process. Information about the unknown state is obtained indirectly from an observation $\mathbf{y}_k$ and its likelihood $\mathcal{P}(\mathbf{y}_k|\mathbf{x}_k)$. The observation process $\mathbf{y}$ is assumed to be conditionally independent[2] of the state process $\mathbf{x}$, which means that an observation $\mathbf{y}_k$ only depends on the state $\mathbf{x}_k$ at the same time instant, as expressed by (2.4). The observation process $\mathbf{y}$ is also assumed mutually independent conditionally to the state $\mathbf{x}$, as expressed by (2.5).

$$\mathcal{P}(\mathbf{y}_k|\mathbf{x}_n, \mathbf{x}_{n-1}, \ldots, \mathbf{x}_0) = \mathcal{P}(\mathbf{y}_k|\mathbf{x}_k) \quad \forall n \geq k \geq 0 \tag{2.4}$$

$$\mathcal{P}(\mathbf{y}_n, \ldots, \mathbf{y}_0|\mathbf{x}_n, \ldots, \mathbf{x}_0) = \prod_{k=1}^{n} \mathcal{P}(\mathbf{y}_k|\mathbf{x}_n, \ldots, \mathbf{x}_0) = \prod_{k=1}^{n} \mathcal{P}(\mathbf{y}_k|\mathbf{x}_k) \tag{2.5}$$

A Hidden Markov Model can be schematically summarized as presented in Figure 2.1.

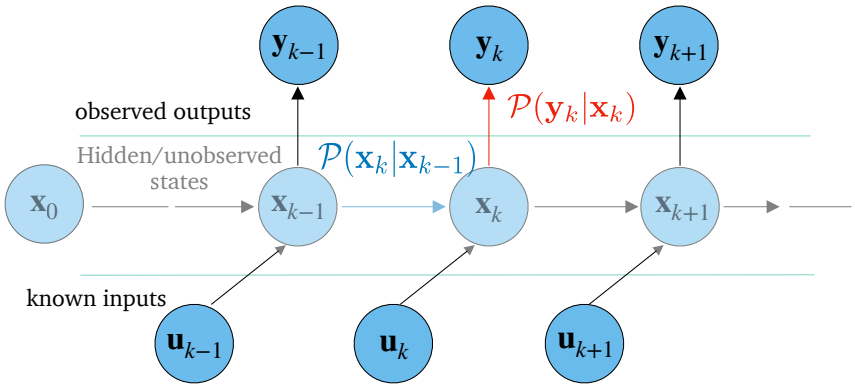


Figure 2.1: Hidden Markov Model Structure.

Recursive Bayesian filtering equations

The recursive Bayesian filter uses prior and causal knowledge to infer the conditional probability $\mathcal{P}(\mathbf{x}_k|\mathbf{y}_{0:k})$. It takes the form of a "prediction/correction" scheme, which enables us to recursively compute, at each time instant k, an *a priori* probability density during prediction, provided that one knows the state transition density $\mathcal{P}(\mathbf{x}_k|\mathbf{x}_{k-1})$. The prior density is then updated during correction, by weighting it with the likelihood $\mathcal{P}(\mathbf{y}_k|\mathbf{x}_k)$ of an available observation $\mathbf{y}_k$. The correction gives us the *a posteriori* probability density, providing us with the complete solution of the inference problem.

[2]Definition of independency A.4.

Hence, assuming that one knows the initial probability $\mathcal{P}(\mathbf{x}_0|\mathbf{y}_0)$, the procedure is given as follows ($\forall k = 1, \dots$).

Prediction:

$$\mathcal{P}(\mathbf{x}_k|\mathbf{y}_{0:k-1}) = \int_{\mathbb{R}^d} \mathcal{P}(\mathbf{x}_k|\mathbf{x}_{k-1})\mathcal{P}(\mathbf{x}_{k-1}|\mathbf{y}_{0:k-1})d\mathbf{x}_{k-1} \tag{2.6}$$

Correction:

$$\mathcal{P}(\mathbf{x}_k|\mathbf{y}_{0:k}) = \frac{\mathcal{P}(\mathbf{y}_k|\mathbf{x}_k)\mathcal{P}(\mathbf{x}_k|\mathbf{y}_{0:k-1})}{\mathcal{P}(\mathbf{y}_k|\mathbf{y}_{0:k-1})} \tag{2.7}$$

The expression (2.6) gives the prior pdf of $\mathbf{x}_k$ from the knowledge of all past observations $\mathbf{y}_{0:k-1}$, and is referred to as the Chapman-Kolmogorov equation [Pavliotis, 2014]. Given a current observation $\mathbf{y}_k$, the prior density is then refined to yield the posterior density of $\mathbf{x}_k$ from the knowledge of past and current observations. The denominator $\mathcal{P}(\mathbf{y}_k|\mathbf{y}_{0:k-1})$, whose formal expression is given by

$$\mathcal{P}(\mathbf{y}_k|\mathbf{y}_{0:k-1}) = \int_{\mathbb{R}^d} \mathcal{P}(\mathbf{y}_k|\mathbf{x}_k)\mathcal{P}(\mathbf{x}_k|\mathbf{y}_{0:k-1})d\mathbf{x}_k \tag{2.8}$$

is a normalization term, ensuring that the integral of probabilities is equal to one.

It turns out that solving a problem of state estimation by using this full Bayesian inference requires the need to compute multidimensional probability density integrals, which are analytically non-tractable. Practical methods then seek approximations of the prior density and the posterior density, either by constraining the form of the pdf (e.g., assuming Gaussianity) or adapting it to the nature of the models $\mathbf{f}$ and $\mathbf{h}$. Besides, the estimation of $\mathcal{P}(\mathbf{x})$ is often replaced by the estimation of low-order statistical *moments* (see Appendix A.5 for the definition of moments), such as the mean and covariance.

Approximated approaches and optimality criteria

Instead of looking for the full probability density $\mathcal{P}(\mathbf{x}_k|\mathbf{y}_{0:k})$, it may be more convenient to look directly for an estimated realization[3] denoted $\hat{\mathbf{x}}_{k|k}$ (coinciding with the mean of a chosen maximizor). Moreover, most Bayesian estimator approximations do not directly use a state transition density $\mathcal{P}(\mathbf{x}_k|\mathbf{x}_{k-1})$ or a likelihood $\mathcal{P}(\mathbf{y}_k|\mathbf{x}_k)$, since they are difficult to manipulate in practice. Instead, it is more convenient to apply the prediction/correction scheme by exploiting the system of physical equations $\mathbf{f}$ and $\mathbf{h}$ in (2.1). In particular, the state transition function $\mathbf{f}$ is an equivalent substitute for the transition density, while the observation function $\mathbf{h}$ is an equivalent substitute for the likelihood.

In that case, assuming an initial estimate $\hat{\mathbf{x}}_{0|0}$, the problem of synthesizing an optimal recursive state estimator for the system (2.1) comes down to finding a procedure of the form

$$\mathbf{f}_{\text{est}} : \begin{cases} \text{Prediction:} & \hat{\mathbf{x}}_{k|k-1} = \mathbf{f}(\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{u}_{k-1}, \mathbf{0}) \\ \text{Correction:} & \hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \Delta(\mathbf{y}_k, \hat{\mathbf{x}}_{k|k-1}, \mathbf{u}_k) \end{cases} \quad (\forall k = 1, \dots) \tag{2.9}$$

[3] The term "realization" must be taken in a probabilistic sense, i.e., a certain value of the random variable $\mathbf{x}_k$ drawn with respect to its probability density.

consisting of a "copy" of the state transition $\mathbf{f}$ including a correction term $\Delta(\mathbf{y}_k, \hat{\mathbf{x}}_{k|k-1}, \mathbf{u}_k)$. The latter is a function of the available measurement $\mathbf{y}_k$, the input $\mathbf{u}_k$ (eventually), and a prediction $\hat{\mathbf{x}}_{k|k-1}$ estimated from $\mathbf{f}$. This correction term incorporates and weighs all the uncertainties involved in the system of equations $\mathbf{f}$ and $\mathbf{h}$.

The idea behind finding a "good" correction term is to find a direction for an arbitrary cost function J that measures how far $\hat{\mathbf{x}}_{k|k}$ is from its real value $\mathbf{x}_k$. In terms of mathematics, the problem of finding a procedure $\mathbf{f}_{\text{est}}$ allowing us to estimate the posterior value $\hat{\mathbf{x}}_{k|k}$ that matches the most probable realization of $\mathbf{x}_k$, can be performed by defining a cost function $J(\mathbf{x}_k|\mathbf{y}_{0:k})$ that we would like to optimize (i.e. find the minimum or maximum):

$$\hat{\mathbf{x}}_{k|k} = \underset{\mathbf{x}_k}{\operatorname{argopt}}\big(J(\mathbf{x}_k|\mathbf{y}_{0:k})\big) \tag{2.10}$$

This approach transforms the task of determining the full probability density $\mathcal{P}(\mathbf{x}_k|\mathbf{y}_{0:k})$ into the more tractable task of determining an optimal value of $\mathbf{x}_k$ (in the sense of the criterion J) that is able to explain the observable evolutions of the system.

Three common optimization problems are usually addressed to synthesize different estimators of $\mathbf{x}_k$: the Minimum Variance (MV), the Maximum Likelihood (ML) or the Maximum A Posteriori (MAP).

- The Minimum Variance criterion looks for the estimate $\hat{\mathbf{x}}_k$ that minimizes the expected value of the quadratic distance from its true value $\mathbf{x}_k$ (which is the variance of the estimation error), given past observations $\mathbf{y}_{0:k}$.

$$\hat{\mathbf{x}}_{k|k} = \underset{\hat{\mathbf{x}}_k}{\operatorname{argmin}} \Big(\mathbb{E}\big((\hat{\mathbf{x}}_k - \mathbf{x}_k)^\top(\hat{\mathbf{x}}_k - \mathbf{x}_k)|\mathbf{y}_{0:k}\big)\Big) \tag{2.11}$$

- The Maximum Likelihood criterion looks for the realization of $\mathbf{x}_k$ that maximizes the likelihood of the observations $\mathbf{y}_{0:k}$

$$\hat{\mathbf{x}}_{k|k} = \underset{\mathbf{x}_k}{\operatorname{argmax}} \Big(\mathcal{P}\big(\mathbf{y}_{0:k}|\mathbf{x}_k\big)\Big) \tag{2.12}$$

- The Maximum A Posteriori criterion looks for the realization of $\mathbf{x}_k$ that maximizes the posterior probability.

$$\hat{\mathbf{x}}_{k|k} = \underset{\mathbf{x}_k}{\operatorname{argmax}} \Big(\mathcal{P}\big(\mathbf{x}_k|\mathbf{y}_{0:k}\big)\Big) \tag{2.13}$$

In the case of linear models and Gaussian noise, the solutions $\hat{\mathbf{x}}_{k|k}$ given by the MAP and MV optimization problems are equivalent, and both correspond to the mean $\mathbb{E}(\mathbf{x}_k|\mathbf{y}_{0:k})$. In practice, various estimators can be derived to approximate the Bayesian filter, and their construction may depend on several things (e.g., a chosen optimality criterion, noise assumptions, modeling issues, the amount of computations that one allows us to perform, etc.).

Discussion on the choice of the probability distribution

For the moment, let us return to the general point of view given in the form of probability densities. A fundamental component of the Bayesian framework is the prior distribution $\mathcal{P}(\mathbf{x}_k|\mathbf{y}_{0:k-1})$ that encodes initial knowledge about the propagation of the system's state. In the absence of precise information, one often seeks a distribution that reflects the highest amount of uncertainty while respecting known statistics (e.g., mean, covariance, or higher-order moments). From the perspective of information theory, such a distribution corresponds to the maximum entropy distribution.

The term *entropy* has been introduced by Rudolf Clausius in 1865, and often characterizes the "degree of disorder"[4] of a system. Commonly used in physics and thermodynamics, this concept has been extended for many domains of application, such as information theory, where the notion of Shannon's entropy [Shannon, 1948] plays a prominent role. In this context, the entropy measures the information content provided by a source (for example, a source delivering a message). If that source contains a large amount of variability of information, then the entropy is said to be large (and the kind of message that the source provides would be highly uncertain). Inversely, if the source contains a low amount of variability of information, then the entropy is said to be low (and the message delivered by the source will be easily predictable since the uncertainty of delivering a given message would be small).

For Bayesian inference, the principle of maximum entropy is often used to obtain prior probability distributions, an approach strongly advocated by E.T. Jaynes, claiming that the maximum entropy distribution represents the distribution that spreads out as uniformly as possible, without contradicting the given information (i.e., the prescribed moments) [Jaynes, 1968]. In particular, he stressed that despite being conceptually very different, Bayes' rule and the maximum entropy principle are, in fact, complementary to each other [Jaynes, 1988].

This principle naturally leads to the Gaussian distribution when the mean and covariance are prescribed, making it the preferable distribution in many estimation problems. Another reason why a Gaussian pdf is often used is the fact that it appears as the resulting distribution of the Central Limit Theorem (CLT), which states that the contribution of many sources of noise results in a Gaussian density. Under this assumption on the noise and on the state's uncertainty, suboptimal approximations of the Bayesian recursive filter have been developed throughout the years.

[4] Actually, the term "degree of disorder" is an abuse of language in physics. To be more precise, the entropy of an object rather describes the "range of possible configurations" in which this object can take its form. For example, a gas is an object of higher entropy than a solid, since its constituting atoms can take many configurations while still recognizing the same gas. Inversely, the constituting atoms of a solid are heavily constrained, organized in a fixed structure, which means less possible configurations and less entropy.

2.2 Historical overview

The recursive Bayesian filter provides a general and conceptually elegant solution to the state estimation problem. However, its exact implementation is limited to specific cases where the integrals admit closed-form expressions. Moreover, maintaining a full probabilistic representation of $\mathcal{P}(\mathbf{x}|\mathbf{y})$ is often computationally prohibitive in high-dimensional state spaces. That is why various approximation families have been proposed over the decades, with the aim of looking for statistical properties of the unknown state vector $\mathbf{x}_k$ in the Euclidean setting.

2.2.1 Pioneering works

Bayesian theory was introduced by Thomas Bayes and posthumously published by Richard Price in 1763. Bayes' rule, from which the recursive Bayesian filter is derived (see Appendix A.2), formalizes the fundamental principles of probabilistic inference. Bayesian inference has since become a major framework in statistics, with successful applications in decision theory, detection, estimation, pattern recognition, and machine learning. However, it did not gain its deserved attention during the early days until its modern form was rediscovered by the French mathematician Pierre-Simon Laplace in its *Théorie analytique des probabilités* in 1812. In this book, Laplace provided fundamental tools for the theory of probability, which were later applied to stochastic filtering.

Earlier, Carl Friedrich Gauss introduced the Least Squares (LS) estimator in 1795 to study the motion of planets and comets. Initially formulated in a deterministic framework, the LS estimator provides an optimal solution to estimation problems with imperfect measurements, without explicit probabilistic assumptions on the variables or noise. Under suitable conditions, its optimality is formalized by the Gauss–Markov theorem, which explains its success in accurately estimating the trajectory of the Ceres asteroid discovered by Giuseppe Piazzi in 1801.

Years later, stochastic filtering theory emerged during the first half of the 20^{th} century through the pioneering contributions of Norbert Wiener, Andrey N. Kolmogorov, and Ronald A. Fisher. In 1942, Wiener introduced a linear stochastic filter for stationary processes that fully exploited statistical properties and minimized the estimation variance, with a solution initially formulated in the spectral domain. Stochastic filtering later culminated in the publication of the linear Kalman filter [Kalman, 1960] by Rudolf E. Kalman and Richard S. Bucy, paving the way for modern filtering theory. Kalman filtering relies on a state-space representation and a recursive time-domain formulation for nonstationary processes, and can be interpreted as a time-varying Wiener filter [Anderson and Moore, 2007]. Under linearity, white-noise, and Gaussian assumptions, the Kalman filter provides the minimum-variance state estimate, making it a special case of the recursive Bayesian filter and the optimal linear filter among all linear estimators.

The introduction of the Kalman filter initiated extensive theoretical and applied research, as its principles enabled numerically efficient algorithms suitable for real-world implementation. This development led to the field of data fusion for navigation, particularly within the aerospace industry, where the abundance of estimation problems provided an ideal ground for its application. While the Kalman filter provided effective solutions within the linear framework, increasing attention from the 1960s onward focused on extending these methods to nonlinear systems. As system complexity grew, applying the linear Kalman filter to nonlinear models often resulted in poor performance or divergence. This motivated a major methodological shift, characterized by the development of suboptimal filtering techniques as approximations to the Bayesian filter, such as [Wonham, 1964], [Zakai, 1969], or [Bucy, 1970].

2.2.2 Overview on Bayesian suboptimal filtering

Unlike the linear case, where the Kalman filter yields an exact minimum-variance solution, nonlinear estimation methods generally lack formal guarantees of optimality, performance, or stability. Nonetheless, numerous suboptimal approaches have been developed to address nonlinear state estimation, differing mainly in how they approximate the intractable Bayesian integrals. These methods can be broadly classified into functional approximations, Gaussian-sum approximations, and sampling-based (numerical) approaches.

Functional approximations

Functional approximation techniques address nonlinear filtering by locally approximating the deterministic nonlinear functions involved, rather than the Bayesian integrals themselves. The most representative approach in this class relies on Taylor series expansions (TSE), leading to the Extended Kalman Filter (EKF), initially introduced by Stanley Schmidt[5] to extend Kalman filtering to nonlinear spacecraft navigation problems [Bellantoni and Dodge, 1967]. The EKF has been extensively studied [Jazwinski, 2007; Simon, 2006] and remains effective for certain classes of nonlinear systems, although its Gaussian assumption may lead to poor performance when the true state distribution is significantly non-Gaussian. To mitigate limitations induced by first-order TSE truncation, several extensions have been proposed, including higher-order EKFs [Athans et al., 1968] and the Iterated EKF [Meng et al., 2010]. Additional variants include adaptive and hybrid Kalman filters, which account for unmodeled dynamics or combine different filtering paradigms [Simon, 2006; Zeng et al., 2011]. Despite their widespread use in INS/GNSS data fusion [Mohamed and Schwarz, 1999; Hide et al., 2003] and robotics applications such as SLAM and visual odometry, EKF-based methods suffer from known limitations in consistency, particularly in visual-SLAM settings [Bailey et al., 2006; Liang et al., 2023].

[5]It was during a visit by Kalman to the NASA Ames Research Center that Schmidt saw the potential of Kalman's ideas to the nonlinear problem of trajectory estimation for the Apollo program, resulting in its incorporation in the Apollo navigation computer.

Gaussian sum approximations

Unlike the linearized EKF, which focuses solely on a single mean and covariance, Gaussian sum filters (GSF), first introduced by [Alspach and Sorenson, 1972], approximate the underlying probability distribution using a weighted mixture of Gaussian components. This representation is better suited to capturing non-Gaussian behavior, which cannot be adequately modeled by a single Gaussian. GSFs rely on the observation that a large class of non-Gaussian distributions can be approximated by a sum of Gaussians. Accordingly, these filters run a bank of EKFs in parallel, updating the mean, covariance, and weight of each Gaussian in the mixture. Like EKFs, GSFs have been widely applied to engineering problems, with particular attention to space surveillance of debris [Horwood et al., 2011; DeMars et al., 2009; Giza et al., 2010]. Their popularity in the context of space situational awareness (SSA)[6] stems from their ability to address *uncertainty realism*, which is the accurate and truthful representation of the space debris' density. In this sense, Gaussian sum filters offer a practical way to approximate the Bayesian estimator by overcoming the limitations of a single Gaussian and capturing uncertainty more faithfully. However, a question that can arise with this approach is then how to fuse these different Gaussians to yield a single estimate and covariance for decision-making.

Sampling approximations

Sampling-based approaches represent the system state distribution with a set of particles, avoiding the need for partial derivatives. The idea is to generate a sample with a sufficient amount of particles, which are then propagated into the nonlinear functions to reconstruct useful statistics. They are divided into deterministic sampling and stochastic sampling, and their typical representation is the Unscented Kalman filter (UKF) and the Particle Filter (PF), respectively.

The UKF, first proposed by [Julier and Uhlmann, 1997], has a structure similar to the EKF but propagates the state through the nonlinear system model directly using a cloud of points. This method often outperforms the EKF while maintaining comparable computational complexity [Wan and Van Der Merwe, 2000]. UKFs rely on a limited number of points, the so-called *sigma-points*, to capture some known statistics of the underlying distribution. Prior to the development of UKFs, the notions of orthogonal polynomials, developed in the late 19^{th} century by P. L. Chebyshev, and pursued by A. A. Markov and T. J. Stieltjes, enabled many developments in applied mathematics, especially in numerical analysis and probability theory. The UKF, and in particular the Unscented transformation (the process from which sigma-points are propagated and used to reconstruct statistical moments), typically rely on such tools. Unlike the EKF, the UKF does not assume Gaussianity: varying the sigma-point dispersion allows representation of different distributions depending on the chosen statistics. A review of UKF extensions is presented later.

[6] SSA encompasses the tracking of all space objects, prediction of space events, potential collisions, threats, and activities.

In the early 1950s, Hammersley introduced Monte-Carlo methods based on Sequential Importance Sampling (SIS) to address statistical problems, laying the foundation for many Monte-Carlo–based techniques in subsequent decades. However, due to limited computing power and the "weights degeneration" problem, these algorithms were not highlighted until the early 1990s, when [Gordon et al., 1993] introduced the resampling step, partially resolving weight degeneration, and proposed the Sampling Importance Resampling (SIR) Particle Filter (PF). Particle filters employ many stochastic samples to perform recursive Bayesian filtering via Monte-Carlo simulations [Doucet et al., 2000]. They represent the posterior distribution with a set of weighted particles, updating each weight as new measurements arrive and resampling to focus on high-probability regions. The posterior is then approximated by the empirical distribution of these particles, enabling flexible non-parametric estimation for highly nonlinear and non-Gaussian systems. Subsequent research focused on improved resampling techniques to enhance PF performance and accuracy [Li et al., 2015]. Fundamentally, PFs differ from UKFs in that they generally use more points, drawn randomly, whereas UKFs rely on deterministic procedures. PFs offer higher estimation performance than KFs but at increased computational cost.

From this historical perspective, the families of Kalman and Particle filters stand out as the fundamental classes of Bayesian estimators for nonlinear stochastic state estimation. Originally developed for linear Gaussian systems, Kalman filters provide a recursive, analytically suboptimal solution for estimating the state of a dynamic system under uncertainty, while Particle filters leverage modern computational power to achieve improved estimation performance. Overall, PFs often stands out as the best approach if accuracy is prioritized, while EKFs are mostly chosen for their low computational cost. In the middle, UKFs often provide a compromise between computational cost and accuracy.

2.3 Approximation of Bayesian filtering by linearization

Kalman filters are part of the more general group of Gaussian filters. Gaussianity often comes as a first pre-requisite to approximate the Bayesian filter. Combined with linearization, it allows the derivation of tractable analytical solutions. In this section, we propose to review the algorithmic scheme enabled by Gaussian filters and its linearized approximations.

2.3.1 The Bayesian filter under Gaussian assumptions

A first assumption to simplify the equations of the Bayesian filter is to suppose that any random variable involved in the prediction/correction scheme follows a Gaussian distribution. Under this assumption, the recursive Bayesian scheme boils down to the following procedure: Assuming that $\hat{\mathbf{x}}_{k-1|k-1}$ and $\mathbf{P}_{k-1|k-1}$ are, respectively, the estimated

mean and covariance of the state $\mathbf{x}_{k-1}$ at time $k-1$, the general form of the Gaussian filter is given by

$$\begin{aligned}
\text{Prediction:} \quad & \begin{cases}
\hat{\mathbf{x}}_{k|k-1} = \displaystyle\int_{\mathbb{R}^d} \mathbf{f}(\mathbf{x}_{k-1}) \mathcal{N}(\mathbf{x}_{k-1};\ \hat{\mathbf{x}}_{k-1|k-1}, \mathbf{P}_{k-1|k-1}) d\mathbf{x}_{k-1} \\
\mathbf{P}_{k|k-1} = \displaystyle\int_{\mathbb{R}^d} \left(\mathbf{f}(\mathbf{x}_{k-1}) - \hat{\mathbf{x}}_{k|k-1}\right) \left(\mathbf{f}(\mathbf{x}_{k-1}) - \hat{\mathbf{x}}_{k|k-1}\right)^\top \\
\qquad\qquad \times \mathcal{N}(\mathbf{x}_{k-1};\ \hat{\mathbf{x}}_{k-1|k-1}, \mathbf{P}_{k-1|k-1}) d\mathbf{x}_{k-1}
\end{cases} \\
\text{Correction:} \quad & \begin{cases}
\hat{\mathbf{y}}_k = \displaystyle\int_{\mathbb{R}^p} \mathbf{h}(\mathbf{x}_k) \mathcal{N}(\mathbf{x}_k;\ \hat{\mathbf{x}}_{k|k-1}, \mathbf{P}_{k|k-1}) d\mathbf{x}_k \\
\mathbf{P}_{\mathbf{y}_k} = \displaystyle\int_{\mathbb{R}^p} \left(\mathbf{h}(\mathbf{x}_k) - \hat{\mathbf{y}}_k\right) \left(\mathbf{h}(\mathbf{x}_k) - \hat{\mathbf{y}}_k\right)^\top \\
\qquad\qquad \times \mathcal{N}(\mathbf{x}_k;\ \hat{\mathbf{x}}_{k|k-1}, \mathbf{P}_{k|k-1}) d\mathbf{x}_k \\
\mathbf{P}_{\mathbf{x}_k\mathbf{y}_k} = \displaystyle\int_{\mathbb{R}^p} \left(\mathbf{f}(\mathbf{x}_{k-1}) - \hat{\mathbf{x}}_{k|k-1}\right) \left(\mathbf{h}(\mathbf{x}_k) - \hat{\mathbf{y}}_k\right)^\top \\
\qquad\qquad \times \mathcal{N}(\mathbf{x}_k;\ \hat{\mathbf{x}}_{k|k-1}, \mathbf{P}_{k|k-1}) d\mathbf{x}_k \\
\mathbf{K}_k = \mathbf{P}_{\mathbf{x}_k\mathbf{y}_k} \mathbf{P}_{\mathbf{y}_k}^{-1} \\
\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k(\mathbf{y}_k - \hat{\mathbf{y}}_k) \\
\mathbf{P}_{k|k} = \mathbf{P}_{k|k-1} - \mathbf{K}_k \mathbf{P}_{\mathbf{y}_k} \mathbf{K}_k^\top
\end{cases}
\end{aligned} \tag{2.14}$$

Hence, contrary to the general Bayesian filter, Gaussian filters only rely on the propagation of the mean and covariance of the state, since a normal distribution can be entirely described by these low-order moments. As seen from the above equations, the moments are defined as integrals whose integrands are all of the form *propagation function* $\times$ *Gaussian pdf*. For any function $\mathbf{g}$, the propagated mean and covariance are expressed as

$$\begin{aligned}
\mathbb{E}[\mathbf{g}(\mathbf{x})] &= \int \mathbf{g}(\mathbf{x})\, \mathcal{N}(\mathbf{x};\, \overline{\mathbf{x}}, \mathbf{P}) d\mathbf{x} \\
\mathrm{Cov}[\mathbf{g}(\mathbf{x})] &= \int \left(\mathbf{g}(\mathbf{x}) - \mathbb{E}[\mathbf{g}(\mathbf{x})]\right) \left(\mathbf{g}(\mathbf{x}) - \mathbb{E}[\mathbf{g}(\mathbf{x})]\right)^\top \mathcal{N}(\mathbf{x};\, \overline{\mathbf{x}}, \mathbf{P}) d\mathbf{x}
\end{aligned} \tag{2.15}$$

When the function $\mathbf{g}$ is linear, these integrals admit closed-form solutions. In contrast, for nonlinear $\mathbf{g}$, analytical evaluation becomes difficult to handle, and numerical approximations may be required. In both cases, the mean is used as the state estimate, as it coincides with the most probable value under a Gaussian assumption, while the covariance quantifies the uncertainty of the estimate by indicating the confidence level associated with the inferred state.

Different approaches to solving the moment integrals in (2.14) give rise to different Gaussian filters. In particular, if the models $\mathbf{f}$ and $\mathbf{h}$ are linear, these integrals simplify to closed-form and tractable expressions, and boils down to the linear Kalman filter equations.

2.3.2 The linear Kalman filter

The Kalman filter is optimal (in the sense of the minimum variance criterion) when the system dynamics $\mathbf{f}$ and the observation model $\mathbf{h}$ are linear and both the state uncertainty

and noise processes are Gaussian[7]. The discrete dynamical system is described as follows:

$$\begin{aligned} \mathbf{x}_k &= \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1}) = \mathbf{A}\mathbf{x}_{k-1} + \mathbf{B}\mathbf{u}_{k-1} + \mathbf{M}\mathbf{w}_{k-1} \\ \mathbf{y}_k &= \mathbf{h}(\mathbf{x}_k, \mathbf{v}_k) = \mathbf{C}\mathbf{x}_k + \mathbf{N}\mathbf{v}_k \end{aligned} \tag{2.16}$$

with the following assumptions on the noise:

$$\mathbf{x} \sim \mathcal{N}(\hat{\mathbf{x}}, \mathbf{P}), \quad \mathbf{w} \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}), \quad \mathbf{v} \sim \mathcal{N}(\mathbf{0}, \mathbf{R}) \tag{2.17}$$

The Kalman filter initialization is performed by means of a known (or assumed) state and a covariance matrix:

$$\hat{\mathbf{x}}_{0|0} = \mathbb{E}(\mathbf{x}_0), \quad \mathbf{P}_{0|0} = \mathbb{E}[(\mathbf{x}_0 - \hat{\mathbf{x}}_{0|0})(\mathbf{x}_0 - \hat{\mathbf{x}}_{0|0})^\top] \tag{2.18}$$

After that, the prediction/correction scheme is given by the following equations, computed for each time index $k = 1, 2, ...$ (provided that we have access to a measurement $\mathbf{y}_k$):

$$\begin{aligned} \text{Prediction:} & \begin{cases} \hat{\mathbf{x}}_{k|k-1} = \mathbf{A}\hat{\mathbf{x}}_{k-1|k-1} + \mathbf{B}\mathbf{u}_{k-1} \\ \mathbf{P}_{k|k-1} = \mathbf{A}\mathbf{P}_{k-1|k-1}\mathbf{A}^\top + \mathbf{M}\mathbf{Q}_{k-1}\mathbf{M}^\top \end{cases} \\ \text{Correction:} & \begin{cases} \mathbf{K}_k = \mathbf{P}_{k|k-1}\mathbf{C}^\top(\mathbf{C}\mathbf{P}_{k|k-1}\mathbf{C}^\top + \mathbf{N}\mathbf{R}_k\mathbf{N}^\top)^{-1} \\ \hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k(\mathbf{y}_k - \mathbf{C}\hat{\mathbf{x}}_{k|k-1}) \\ \mathbf{P}_{k|k} = (\mathbf{I} - \mathbf{K}_k\mathbf{C})\mathbf{P}_{k|k-1} \end{cases} \end{aligned} \tag{2.19}$$

$\mathbf{Q}_k$ and $\mathbf{R}_k$ appear as user-defined matrices whose goal is to capture any assumed uncertainty about the models $\mathbf{f}$ and $\mathbf{h}$. In practice, tuning these matrices correctly is not straightforward, as they parameterize the amount of correction we would like to apply on the predicted state: if $\mathbf{Q}_k$ is set to a level that is relatively superior to $\mathbf{R}_k$, then low confidence is placed on the dynamical model (while high confidence is placed on the measurement), and the predicted state will be strongly corrected. Inversely, if $\mathbf{R}_k$ is set to a level that is relatively superior to $\mathbf{Q}_k$, then low confidence is placed on the measurement (while high confidence is placed on the dynamical model), and the predicted state will be weakly corrected. Thus, tuning these matrices may strongly depend on the confidence one has in the quality of the observation (coming from imperfect sensors) and/or the accuracy of the model (whose precision is difficult to evaluate).

2.3.3 The Extended Kalman filter

Most physical systems are nonlinear and the Kalman filter can no longer provide an optimal solution to the minimum variance optimization problem since requirements about noise are no longer satisfied. Despite from that, many systems are close enough to linear to the point that linear estimation approaches give satisfactory results. In that case, the EKF provides a suboptimal solution by linearizing functions $\mathbf{f}$ and $\mathbf{h}$ around the estimate $\hat{\mathbf{x}}$. The EKF methodology keeps the same prediction/correction scheme as the linear case,

[7] More precisely, under these conditions, it is the optimal estimator within the class of estimators that are linear combinations of the measurements (see [Simon, 2006], p. 130), although nonlinear estimators may, in principle, achieve superior performance.

but it computes, at each time instant, a linearized approximation of the models (based on a first-order Taylor series expansion), and evaluated at the current known estimate:

$$\text{For prediction} \begin{cases} \mathbf{A}_{k-1} = \dfrac{\partial \mathbf{f}}{\partial \mathbf{x}}\Big|_{\mathbf{x}=\hat{\mathbf{x}}_{k-1|k-1},\mathbf{u}=\mathbf{u}_{k-1},\mathbf{w}=\mathbf{0}} \\ \mathbf{M}_{k-1} = \dfrac{\partial \mathbf{f}}{\partial \mathbf{w}}\Big|_{\mathbf{x}=\hat{\mathbf{x}}_{k-1|k-1},\mathbf{u}=\mathbf{u}_{k-1},\mathbf{w}=\mathbf{0}} \end{cases} \tag{2.20}$$

$$\text{For correction} \begin{cases} \mathbf{C}_{k} = \dfrac{\partial \mathbf{h}}{\partial \mathbf{x}}\Big|_{\mathbf{x}=\hat{\mathbf{x}}_{k|k-1},\mathbf{v}=\mathbf{0}} \\ \mathbf{N}_{k} = \dfrac{\partial \mathbf{h}}{\partial \mathbf{v}}\Big|_{\mathbf{x}=\hat{\mathbf{x}}_{k|k-1},\mathbf{v}=\mathbf{0}} \end{cases} \tag{2.21}$$

The expressions above are the jacobians that replace the matrices $\mathbf{A}$, $\mathbf{C}$, $\mathbf{M}$ and $\mathbf{N}$ in the linear Kalman filter. This gives the extended algorithm:

$$\begin{aligned} \text{Prediction:} & \begin{cases} \hat{\mathbf{x}}_{k|k-1} = \mathbf{f}(\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{u}_{k-1}, \mathbf{0}) \\ \mathbf{P}_{k|k-1} = \mathbf{A}_{k-1}\mathbf{P}_{k-1|k-1}\mathbf{A}_{k-1}^{\top} + \mathbf{M}\mathbf{Q}_{k-1}\mathbf{M}^{\top} \end{cases} \\ \text{Correction:} & \begin{cases} \mathbf{K}_k = \mathbf{P}_{k|k-1}\mathbf{C}_k^{\top}(\mathbf{C}_k\mathbf{P}_{k|k-1}\mathbf{C}_k^{\top} + \mathbf{N}\mathbf{R}_k\mathbf{N}^{\top})^{-1} \\ \hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k\Big(\mathbf{y}_k - \mathbf{h}(\hat{\mathbf{x}}_{k|k-1}, \mathbf{0})\Big) \\ \mathbf{P}_{k|k} = (\mathbf{I} - \mathbf{K}_k\mathbf{C}_k)\mathbf{P}_{k|k-1} \end{cases} \end{aligned} \tag{2.22}$$

Beyond the requirement that the functions $\mathbf{f}$ and $\mathbf{h}$ must be differentiable, a fundamental limitation of the EKF is that system nonlinearities must remain small relative to the uncertainty level for the linearization to be valid. Additionally, in many practical applications, Jacobian computations (often performed numerically) can introduce further errors, thereby enhancing those from modeling approximations and simplifying assumptions. To mitigate these issues, one may resort to second- or higher-order approximations to optimize the quality of the estimate.

2.4 Approximation of Bayesian filtering by sigma-points

Another formulation of the Kalman filter exploits numerical approximations of moment integrals. The work initiated by [Julier et al., 1995] led to a method for state estimation in nonlinear dynamical systems under random disturbances. This approach generates a set of points capturing the statistical properties of the estimated state, based on the intuition that *approximating a probability distribution is easier than approximating an arbitrary nonlinear function*. Points are generated so that their sample mean and covariance match the current estimate $\hat{\mathbf{x}}k|k$ and covariance $\mathbf{P}k|k$. The nonlinear function is applied to each point to yield a transformed sample, from which the predicted mean and covariance are reconstructed. This procedure is known as the Unscented Transformation (UT), or sigma-point transformation, and is illustrated in Figure 2.2.

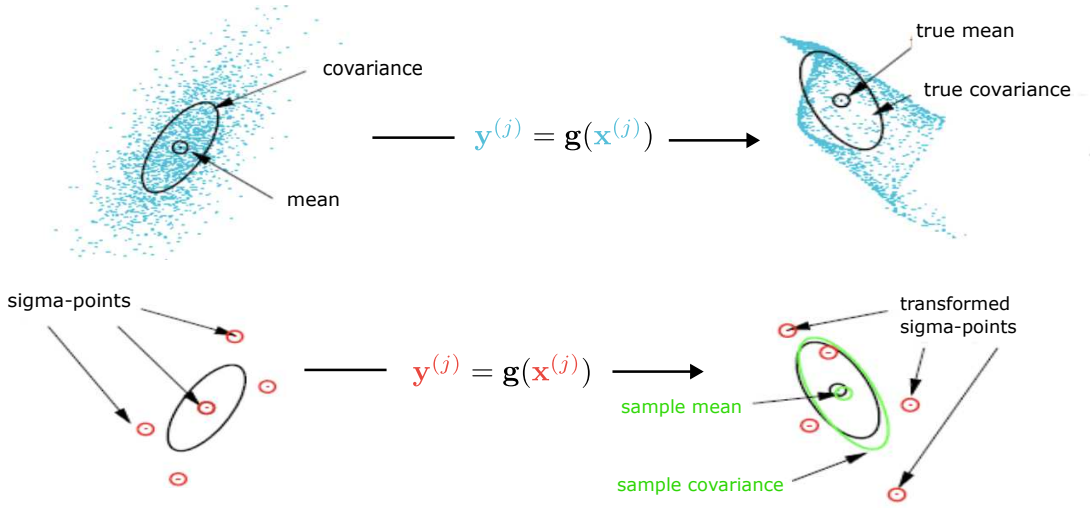


Figure 2.2: Transformation of a random variable into a nonlinear function $\mathbf{y} = \mathbf{g}(\mathbf{x})$. Image taken from [Wan and Van Der Merwe, 2000].

Instead of generating a large number of particles (blue dots) and propagating them into $\mathbf{g}$ to yield a transformed sample from which the true mean and covariance can be computed, it is (numerically) less expensive to generate a small number of (adequately placed) sigma-points and propagate them into $\mathbf{g}$ to yield a transformed sample from which the approximated mean and covariance can be computed.

Although this approach may appear similar to a Monte-Carlo method, the samples are not randomly drawn. Instead, they are deterministically selected to encode specific information about the underlying distribution. A common strategy relies on moment-matching, where the sampling is designed to reproduce the first and second moments of the distribution, assumed to be known before propagation. The determination of such samples is derived from a class of methods, called N-point *quadrature rules* [Smyth, 1998], which are well-suited to numerically approximate moment integrals.

2.4.1 Approximation of moment integrals by quadrature

An N-point quadrature method (in the mono-dimensional case) explicitly aims to approximate the following integral

$$\int_a^b \mathrm{g}(x)w(x)dx \tag{2.23}$$

defined within a given interval $[a, b]$, where g is any nonlinear propagation function of the variable x, with a weight function $w(x)$ (not necessarily a pdf at this moment). The approximation is made thanks to a small number of N points $\{x^{(j)}\}_{j=0,\dots,N-1}$ with associated weights $\{W^{(j)}\}_{j=0,\dots,N-1}$, such that

$$\int_a^b \mathrm{g}(x)w(x)dx \approx \sum_{j=0}^{N-1} W^{(j)}\mathrm{g}(x^{(j)}). \tag{2.24}$$

The weighted sum given in the expression (2.24) is often referred to as a *quadrature formula*, which is expected to be as close as possible to the continuous integral using well-chosen sets of points and weights. Hence, finding these points (also called *nodes*) and

weights is the precise objective of quadrature methods.

For that, one often has to establish a *quadrature rule*, which consists of making the quadrature formula "exact for g being part of a class of orthogonal polynamials, of degree up to a chosen order". In particular, N-point *Gauss quadrature rules* are specifically designed to make the quadrature formula exact for orthogonal polynomials of degree up to $2N-1$. Selecting these polynomials usually depends on the form of the weight function $w(x)$, and the interval $[a,b]$ in which they are orthogonal with respect to $w(x)$. There exist several Gauss quadrature rules (e.g., Gauss-Legendre, Gauss-Radeau, Gauss-Lobatto, etc.), but the one of interest (if one seeks to approximate moment integrals of Gaussian distributions) is the Gauss-Hermite rule [Arasaratnam et al., 2007].

The Gauss-Hermite rule uses Hermite polynomials $H_n(x)$ [Weisstein, 20XX] (where n is the degree of the polynomial), orthogonals in the interval $]-\infty,+\infty[$ with respect to a weight function of the form $w(x)=\exp(-x^2)$. Hence, provided that we have the correct set of nodes and weights $\{x^{(j)},W^{(j)}\}_{j=0,\dots,N-1}$, the following expression

$$\int_{-\infty}^{+\infty} \mathrm{g}(x)\exp(-x^2)dx = \sum_{j=0}^{N-1} W^{(j)}\mathrm{g}(x^{(j)}) \tag{2.25}$$

is designed to be exact for $\mathrm{g}(x)$ being a Hermite polynomial of degree inferior or equal to $2N-1$. It turns out that the nodes $\{x^{(j)}\}_{j=0,\dots,N-1}$ are the N zeros of the polynomial $H_N(x)$, and the weights $\{W^{(j)}\}_{j=0,\dots,N-1}$ are obtained using the following expression:

$$W^{(j)} = \frac{2^{N-1}N!\sqrt{\pi}}{N^2[H_{N-1}(x^{(j)})]^2}, \quad \forall j=0,\dots,N-1 \tag{2.26}$$

For a given number of N points, tables of nodes and weights have been well-established in the literature [Štecha and Havlena, 2012]. In particular, if the weight function is of the form $w(x)=\exp(-\frac{1}{2}x^2)$ (which is typically the form of a univariate zero-mean Gaussian distribution with variance 1), then the Gauss-Hermite method provides us with the nodes and weights of this centered-normalized Gaussian $\mathcal{N}(0,1)$. Hence, the resulting set of points and weights would exactly capture the moments of this distribution, providing us with the following moment constraints

$$\begin{aligned}\int_{-\infty}^{+\infty} x\,\mathcal{N}(0,1)dx &= \sum_{j=0}^{N-1} W^{(j)}x^{(j)} = 0\\ \int_{-\infty}^{+\infty} x^2\,\mathcal{N}(0,1)dx &= \sum_{j=0}^{N-1} W^{(j)}(x^{(j)})^2 = 1\end{aligned} \tag{2.27}$$

Besides, an important condition to make sure that the sample respect the definition of a probability density is that the sum of the weights should be equal 1, which is exactly the case for the weights resulting from a Gauss-Hermite method, i.e.

$$\int_{-\infty}^{+\infty} \mathcal{N}(0,1)dx = \sum_{j} W^{(j)} = 1. \tag{2.28}$$

Extending this method to the multi-dimensional case ($\mathbf{x} \in \mathbb{R}^d$) is straightforward. Consider an N-point Gauss quadrature rule in the d-dimensional case ($d > 1$). It turns out that a 3-point Gauss quadrature rule can be applied for the random variable x_i, $i = \{1, \dots, d\}$. Hence, $N = 2d + 1$ points would be necessary to capture the distribution of $\mathbf{x} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ (where $\mathbf{I}$ is a $d \times d$ identity matrix), and the moment constraints (2.27) would hold for every x_i, i.e.

$$\begin{aligned} \int_{-\infty}^{+\infty} x_i \mathcal{N}(0,1) dx_i &= \sum_{j=0}^{2d} W^{(j)} x_i^{(j)} = 0, \quad \forall i = 1, \dots, d \\ \int_{-\infty}^{+\infty} x_i^2 \mathcal{N}(0,1) dx_i &= \sum_{j=0}^{2d} W^{(j)} (x_i^{(j)})^2 = 1, \quad \forall i = 1, \dots, d \end{aligned} \tag{2.29}$$

Endowed with nice properties, a centered-normalized Gaussian random vector $\mathbf{x}_{01} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ remains a Gaussian random vector after a linear transformation, i.e. $\overline{\mathbf{x}} + \sqrt{\mathbf{P}}\, \mathbf{x}_{01} \sim \mathcal{N}(\overline{\mathbf{x}}, \mathbf{P})$. Applying this type of transformation on each point $\mathbf{x}^{(j)}$ enables us to get the sigma-points of a random vector $\mathbf{x} \sim \mathcal{N}(\overline{\mathbf{x}}, \mathbf{P})$. Finally, these sigma-points and weights can be used to approximate integrals of the form (2.15) for any nonlinear function $\mathbf{g}$, by resorting to a quadrature formula.

The Gauss-Hermite rule is one possible way to generate sigma-points and weights, and is specifically tailored for a Gaussian pdf. However, assuming that a different weight function $w(x)$ was involved, a different quadrature rule endowed with different orthogonal polynomials would be necessary. Whatever the case, they all provide ways to deterministically generate sigma-points and weights for an assumed pdf, all for the sake of using quadrature formulas instead of looking for analytical derivations of integrals. This principle is at the core of UKFs, where prediction and correction steps typically take the form of weighted sums.

2.4.2 The Unscented Kalman filter

On the basis of the numerical methods introduced previously, the prediction/correction scheme in nonlinear settings can be modified to produce approximations of the mean estimate and the covariance with a higher accuracy than those given by an EKF. Hence, the Kalman filter equations can be reformulated as follows:

$$\text{Prediction:} \begin{cases} \mathbf{x}_{k|k-1}^{(j)} = \mathbf{f}(\mathbf{x}_{k-1|k-1}^{(j)}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1}^{(j)}) \\ \hat{\mathbf{x}}_{k|k-1} = \sum_{j=0}^{N-1} W^{(j)} \mathbf{x}_{k|k-1}^{(j)} \\ \mathbf{P}_{k|k-1} = \sum_{j=0}^{N-1} W^{(j)} \left(\mathbf{x}_{k|k-1}^{(j)} - \hat{\mathbf{x}}_{k|k-1} \right) \left(\mathbf{x}_{k|k-1}^{(j)} - \hat{\mathbf{x}}_{k|k-1} \right)^\top \end{cases} \tag{2.30}$$

$$\text{Correction:} \begin{cases} \mathbf{y}_k^{(j)} = \mathbf{h}(\mathbf{x}_{k|k-1}^{(j)}, \mathbf{v}_k^{(j)}) \\ \hat{\mathbf{y}}_k = \sum_{j=0}^{N-1} W^{(j)} \mathbf{y}_k^{(j)} \\ \mathbf{P}_{\mathbf{y}_k} = \sum_{j=0}^{N-1} W^{(j)} \left(\mathbf{y}_k^{(j)} - \hat{\mathbf{y}}_k\right)\left(\mathbf{y}_k^{(j)} - \hat{\mathbf{y}}_k\right)^\top \\ \mathbf{P}_{\mathbf{x}_k\mathbf{y}_k} = \sum_{j=0}^{N-1} W^{(j)} \left(\mathbf{x}_{k|k-1}^{(j)} - \hat{\mathbf{x}}_{k|k-1}\right)\left(\mathbf{y}_k^{(j)} - \hat{\mathbf{y}}_k\right)^\top \\ \mathbf{K}_k = \mathbf{P}_{\mathbf{x}_k\mathbf{y}_k}\mathbf{P}_{\mathbf{y}_k}^{-1} \\ \hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k(\mathbf{y}_k - \hat{\mathbf{y}}_k) \\ \mathbf{P}_{k|k} = \mathbf{P}_{k|k-1} - \mathbf{K}_k\mathbf{P}_{\mathbf{y}_k}\mathbf{K}_k^\top \end{cases} \tag{2.31}$$

In practice, the UKF is implemented with an efficient and numerically stable algorithm, which exploits the Cholesky factor decomposition, and known as the Square-Root UKF (SR-UKF) [Van der Merwe and Wan, 2001].

2.4.3 The problem of sigma-points adjustment

Accurate placement of the sigma-points (and their associated weights) is crucial to provide good performance for UKFs. As seen previously, sigma-points and weights can be found by relying on N-point quadrature methods. In addition, taking a minimal number of $N = 2d + 1$ points may appear as a constraint if one wants to maintain good computational efficiency to make the UKF useable in real-time.

For normal distributions, a third-order[8] Gauss-Hermite rule gives exactly the $2d + 1$ nodes and weights such that they coincide with the current known estimate $\hat{\mathbf{x}}_{k-1|k-1}$ and its covariance $\mathbf{P}_{k-1|k-1}$. Yet, there is no guarantee that a normal distribution can be assumed every time. Moreover, depending on the nonlinearities of the propagation function, the mean and covariance reconstructed from the propagated sample may be more or less exact. In that case, we often would like to add extra degrees of freedom allowing us to tune the sigma-point placement. In the early publications of the UKF [Julier et al., 2000], the authors proposed to generate these sets as follows:

$$\begin{aligned} \mathbf{x}_{k-1|k-1}^{(0)} &= \hat{\mathbf{x}}_{k-1|k-1} & W^{(0)} &= \frac{\kappa}{d+\kappa} \\ \mathbf{x}_{k-1|k-1}^{(j)} &= \hat{\mathbf{x}}_{k-1|k-1} - \left(\sqrt{(d+\kappa)\mathbf{P}_{k-1|k-1}}\right)_j & W^{(j)} &= \frac{1}{2(d+\kappa)} \quad \forall j = 1, \ldots, d \\ \mathbf{x}_{k-1|k-1}^{(d+j)} &= \hat{\mathbf{x}}_{k-1|k-1} + \left(\sqrt{(d+\kappa)\mathbf{P}_{k-1|k-1}}\right)_j & W^{(j)} &= \frac{1}{2(d+\kappa)} \quad \forall j = 1, \ldots, d \end{aligned} \tag{2.32}$$

where $\left(\sqrt{(d+\kappa)\mathbf{P}}\right)_j$ denotes the j^{th} row or column of the matrix square-root of $(d+\kappa)\mathbf{P}$. The coefficient $\kappa \in \mathbb{R}$ provides an extra degree of freedom to "fine tune" the higher-order errors in the mean and covariance approximation. Any κ can be used, as long as $d+\kappa \neq 0$.

[8] In the sense where three points per principal axis is used (two symmetric points and a central point), making the method exact for Hermite polynomials up to the third degree.

Remark 2.1. In the special case where $\kappa = 3-d$ is chosen, it turns out that the heuristic (2.32) coincides with the third-order Gauss-Hermite rule. Indeed, for any sigma-point other than the central one, choosing $\kappa = 3-d$ would give the following scaling factor and weight

$$\sqrt{d+\kappa} = \sqrt{3} \quad \text{and} \quad W^{(j)} = \frac{1}{2(d+\kappa)} = \frac{1}{6} \quad \forall j \neq 0.$$

These values are exactly the same values given by a 3-point Gauss-Hermite rule for a random variable $x \sim \mathcal{N}(0,1)$. The only difference lies in the weight associated to the central point. Indeed, we have:

3-point Gauss-Hermite rule	Heuristic (2.32)
$W^{(0)} = \frac{2}{3}$	$W^{(0)} = \frac{\kappa}{d+\kappa}, \quad (d \geq 1)$

These two quantities match if $d = 1$, but for $d > 1$, the sum of the weights must maintain the condition $\sum_{j=0}^{2d} W^{(j)} = 1$, which implies

$$W^{(0)} + \frac{d}{d+\kappa} = 1 \quad \Rightarrow \quad W^{(0)} = \frac{\kappa}{d+\kappa}. \tag{2.33}$$

If another distribution is assumed for $\mathbf{x}_{k-1}$ instead of a Gaussian, then tuning κ differently might be more appropriate. If κ alone is not sufficient, or worse, provides unwanted effects, additional parameters can be used to correct these side effects. Indeed, in some cases, it may be necessary to take $\kappa < 0$ to spread sigma-points closer to the mean. However, this may result in a non-positive semi-definite posterior covariance, which is defective for the proper conduct of Kalman filter equations.

As an alternative, the Scaled Unscented Transformation (SUT) [Julier, 2002] introduces extra degrees of freedom to maintain the scaling of the sigma-points and avoiding non-positive covariance through the tuning of a new parameter α. Another parameter β is also introduced, and affects the weighting of the zero$^{\text{th}}$ sigma-point. By making the distinction between the weight $W_m^{(0)}$ used for the mean, and the weight $W_c^{(0)}$ used for the covariance, the SUT unifies the previous heuristic (2.32), and generates sigma-points and weights as follows:

$$\begin{aligned}
&\lambda = \alpha^2(d+\kappa) - d && W_m^{(0)} = \frac{\lambda}{d+\lambda} \\
&\mathbf{x}_{k-1|k-1}^{(0)} = \hat{\mathbf{x}}_{k-1|k-1} && W_c^{(0)} = W_m^{(0)} + (1-\alpha^2+\beta) \\
&\mathbf{x}_{k-1|k-1}^{(j)} = \hat{\mathbf{x}}_{k-1|k-1} - \left(\sqrt{(d+\lambda)\mathbf{P}_{k-1|k-1}}\right)_j && W^{(j)} = \frac{1}{2(d+\lambda)} \quad \forall j = 1,\dots,d \\
&\mathbf{x}_{k-1|k-1}^{(d+j)} = \hat{\mathbf{x}}_{k-1|k-1} + \left(\sqrt{(d+\lambda)\mathbf{P}_{k-1|k-1}}\right)_j && W^{(j)} = \frac{1}{2(d+\lambda)} \quad \forall j = 1,\dots,d
\end{aligned} \tag{2.34}$$

The parameters α, β and κ are usually manually tuned by the user, and affect the UT in the following way:

- α is a scaling factor that spreads the sigma-points around $\hat{\mathbf{x}}_{k-1|k-1}$. It is a real positive number, often chosen between 10^{-1} and 1.

- κ guarantees the semidefinite positiveness of the covariance. It is suggested to take it superior or equal to zero.
- $\beta \in \mathbb{R}$ is used to partially minimize the higher-order errors in the covariance estimate. At the time it was introduced [Julier, 2002], it was argued that choosing $\beta = 2$ is optimal if the distribution is assumed to be Gaussian. However, the latter statement is still discussed in recent literature [Nielsen et al., 2021].

It becomes clear that the key difficulty with unscented methods in a deterministic setting appears to be the adjustment of these three scaling parameters. Beyond the SUT, Julier et al. ntroduced alternative formulations, including the simplex UT, which reduces computational complexity by relying on $d+1$ asymmetric points [Julier and Uhlmann, 2002], and the spherical simplex UT, aimed at mitigating numerical instabilities of the simplex UT [Julier, 2003]. The literature is also filled with some works dealing with asymmetric sigma-points [Masoumnezhad et al., 2015], but the majority of them still adopt the symmetric placement. Numerous studies have sought to systematically determine optimal parameters [Straka et al., 2014; Scardua and da Cruz, 2017; Sakai and Kuroda, 2010; Nielsen et al., 2021], including data-driven approaches such as discriminative training [Sakai and Kuroda, 2010] and genetic algorithms [Masoumnezhad et al., 2015]. However, these methods require substantial training data and remain specific to particular applications. Alternative approaches cast parameter tuning as an optimization problem, for instance by maximizing the measurement log-likelihood with respect to $\{\alpha, \beta, \kappa\}$ [Scardua and da Cruz, 2017]. In [Nielsen et al., 2021], guidelines were proposed to accurately set these parameters, but only in the scalar case, while the multivariate case was briefly addressed. As shown in [Rhudy, 2023], optimal tuning is problem-dependent, as different types of nonlinearity influence the choice of scaling parameters.

Despite all the previous ideas, the values of $\{\alpha, \beta, \kappa\}$ are often left unchanged from the early publications of Julier and Uhlmann. Another set can be derived from the Cubature rule provided in [Arasaratnam and Haykin, 2009], and the resulting sigma-point transformation can be referred to as Cubature Transformation (CT). Table 2.1 summarizes three common sets of parameters. Although these sets may differ by their respective values, the commonality among all of them is that they derive from quadrature rules.

	α	β	κ
UT1 [Julier et al., 1995]	1	0	$3-d$
UT2 [Wan and Van Der Merwe, 2000]	10^{-3}	2	0
CT [Arasaratnam and Haykin, 2009]	1	0	0

Table 2.1: Three different sets of parameters for the sigma-point transformation

Finally, in the general case of a function $\mathbf{f}$ possessing internal noise, the different sigma-points $\mathbf{x}_{k-1|k-1}^{(j)}$ and $\mathbf{w}_{k-1}^{(j)}$ can be generated by performing a state augmentation $\mathbf{x}_{aug} = [\mathbf{x}_{k-1}^\top, \mathbf{w}_{k-1}^\top]^\top$, and applying one of the deterministic procedures detailed above with the

augmented vector $\mathbf{x}_{aug}$. The prediction step of the Kalman filter described in (2.30) can then be executed to produce the propagated mean and covariance. As for the correction step (2.31), it follows exactly the same procedure, except that sigma-points are generated after a state augmentation with the observation noise $\mathbf{x}_{aug} = [\mathbf{x}_k^\top, \mathbf{v}_k^\top]^\top$.

2.5 An alternative solution to the problem of sigma-points adjustement

Another strategy is possible for improving sigma-point placement, while avoiding excessive efforts on tuning the scaling parameters. Let $\mathcal{P}(x)$ be a distribution for the random variable x. Instead of looking for a quadrature rule such that

$$\int_I \mathrm{g}(x)\mathcal{P}(x)dx \approx \sum_{j=1}^{N} W^{(i)}\mathrm{g}(x^{(i)}) \tag{2.35}$$

is exact for g being a polynomial of some degree, an alternative approach would consist of dividing the domain I into N subdomains $\{A_1, \ldots, A_N\}$, and approximating the continuous probability measure $\mu(x)$ as a sum of Dirac measures

$$\mu(x) := \int_I \mathcal{P}(x)dx \approx \sum_{j=1}^{N} W^{(i)}\mathbf{1}_{A_i}(x), \quad \text{with} \quad \mathbf{1}_{A_i}(x) = \begin{cases} 1 & \text{if } x \in A_i \\ 0 & \text{if } x \notin A_i \end{cases} \tag{2.36}$$

such that this approximation satisfies some closeness criterion between the continuous probability measure and the discrete probability measure. This approach can be dealt with by turning to *optimal quantization*.

Originating in signal processing in the early 1950s, "quantization" was initially introduced to discretize continuous signals for transmission purposes. In a probabilistic setting, quantization refers to a class of methods aimed at approximating a continuous d-dimensional probability density by a discrete density supported on N points [Graf and Luschgy, 2000]. "Optimal" quantization seeks to minimize the approximation error by determining an optimal placement of these points through the minimization of a distortion criterion $D(\Gamma)$ where $\Gamma = \{\mathbf{x}^{(i)}\}_{i=1,\ldots,N}$. This framework has been applied to nonlinear filtering with discrete-time observations [Pagès and Pham, 2005], including the Gaussian case [Pagès and Printems, 2003]. While some applications exist in finance [Pagès et al., 2004; Sellami, 2005], optimal quantization (explicitly based on this distortion criterion) has seen limited use in Kalman filtering. Related approaches can be found in the literature, albeit based on an alternative closeness criterion such as the Cramér–von Mises distance [Hanebeck et al., 2009; Steinbring et al., 2015].

In what follows, we present the mathematical foundations of optimal quantization, whose principal results will be instrumental for the developments carried out later in this manuscript. We begin by introducing a formal definition of quantization and formulating

the associated problem of approximating probability distributions. Most of the following definitions and theorems have been taken from [Pagès and Printems, 2003].

2.5.1 Optimal quantization of a random vector

The basic idea of quantization is to replace a random vector $\mathbf{x}$ of a probability space (Definition A.1), denoted $\{\Omega, \mathcal{F}, \mathcal{P}_\mathbf{x}\}$[9] taking its values in $\mathbb{R}^d$, by a random vector $\mathcal{Q}(\mathbf{x})$ taking finitely many values in $\mathbb{R}^d$. The mapping $\mathcal{Q} : \mathbb{R}^d \mapsto \Gamma \subset \mathbb{R}^d$ is called a quantizer. For $N \in \mathbb{N}$, an N-quantizer is a mapping $\mathcal{Q}_N$ such that $|\Gamma| \leq N$, i.e., the target space Γ has N finite elements $\Gamma = (\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(N)})$. There does not exist a single N-quantizer. Indeed, let B_N be the set of all Borel functions[10] taking their values in $(\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(N)})$. The elements of B_N are all N-quantizers in the sense that $\mathcal{Q}_N \in B_N$ gives a quantized version of $\mathbf{x}$. Figure 2.3 illustrates the idea of quantization. Regarding optimal quantization, the

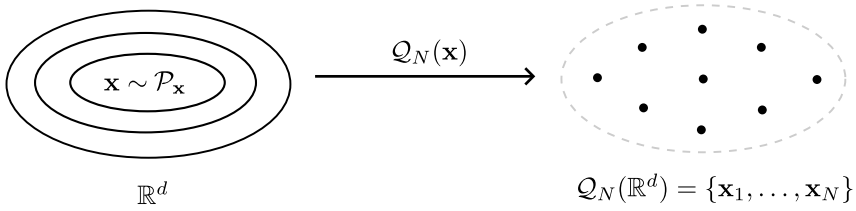


Figure 2.3: Quantization of a random vector $\mathbf{x} \in \mathbb{R}^d$ governed by a probability distribution $\mathcal{P}_\mathbf{x}$. The sample obtained from the N-quantizer $\mathcal{Q}_N$ captures the same distribution with a finite set of N supporting points.

goal is to select the N-quantizer that minimizes the p-quantization error:

$$||\mathbf{x} - \mathcal{Q}_N(\mathbf{x})||_\mathrm{p}^\mathrm{p} \tag{2.37}$$

where $||\cdot||_\mathrm{p}^\mathrm{p}$ denotes the L^p-norm (to the power p). Let $\Gamma = \mathcal{Q}_N(\mathbb{R}^d)$ and consider a closest neighbor rule projection $\mathrm{Proj}_\Gamma : \mathbb{R}^d \mapsto V$ such that

$$||\mathbf{x} - \mathrm{Proj}_\Gamma(\mathbf{x})||_\mathrm{p}^\mathrm{p} \leq ||\mathbf{x} - \mathcal{Q}_N(\mathbf{x})||_\mathrm{p}^\mathrm{p}$$

Closest neighbor rule projections Proj_Γ are in one-to-one correspondence with *Voronoï tesselations* of Γ, namely, with partitions $C_1(\Gamma), \ldots, C_N(\Gamma)$ of $\mathbb{R}^d$ satisfying:

$$C_i(\Gamma) \subset \{\boldsymbol{\zeta} \in \mathbb{R}^d : |\boldsymbol{\zeta} - \mathbf{x}^{(i)}| \leq |\boldsymbol{\zeta} - \mathbf{x}^{(j)}|,\ j = 1, \ldots, N\}$$

The partition $C_i(\Gamma)$ is the i^{th} *Voronoï cell* and consists of all points $\boldsymbol{\zeta}$ for which $\mathbf{x}^{(i)}$ is the closest element among all $\Gamma = \{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(N)}\}$. Then, one may define the projection (which we refer to as a Voronoï quantizer)

$$\mathrm{Proj}_\Gamma(\mathbf{x}) = \sum_{i=1}^{N} \mathbf{x}^{(i)} \mathbf{1}_{C_i(\Gamma)}(\mathbf{x}) = \mathcal{Q}_\mathrm{vor}(\mathbf{x}) = \mathbf{x}_\mathrm{q} \tag{2.38}$$

The notation $\mathbf{x}_\mathrm{q}$ denotes the quantized version of $\mathbf{x}$, and is still a random vector but takes (at most) N possible realizations. $\mathbf{1}_{C_i(\Gamma)}(\mathbf{x})$ is the Dirac delta function, defined as

$$\mathbf{1}_{C_i(\Gamma)}(\mathbf{x}) = \begin{cases} 1 & \text{if } \mathbf{x} \in C_i(\Gamma) \\ 0 & \text{if } \mathbf{x} \notin C_i(\Gamma) \end{cases} \tag{2.39}$$

[9] Note that to avoid any ambiguity in this section, we explicitly indicate for which quantity the probability density is associated, by denoting $\mathcal{P}_\mathbf{x}$ instead of just $\mathcal{P}$.

[10] A Borel function is a mapping between two measurable spaces. A measurable space is a structure allowing to define measures (e.g. the Lebesgues measure).

The error induced by this projection, called p-quantization error, is given by $||\mathbf{x}-\text{Proj}_\Gamma(\mathbf{x})||_\text{p}$, and depends on the grid Γ. It can be shown from (2.38) that we have[11]

$$||\mathbf{x} - \text{Proj}_\Gamma(\mathbf{x})||_\text{p}^\text{p} = ||\mathbf{x} - \mathbf{x}_\text{q}||_\text{p}^\text{p} = \mathbb{E}\Big[\min_{1\leq i\leq N} |\mathbf{x} - \mathbf{x}^{(i)}|_\text{p}^\text{p}\Big] \tag{2.40}$$

for $\Gamma = \{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(N)}\}$. So, if one identifies a grid Γ of size N with the N-tuple $(\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(N)}) \in (\mathbb{R}^d)^N$, the p^{th} power of the p-quantization error is a symmetric (continuous) function D_N^p defined on a N-tuple with pairwise distinct components by

$$D_N^\text{p}(\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(N)}) = \int_{\mathbb{R}^d} \min_{1\leq i\leq N} ||\boldsymbol{\zeta} - \mathbf{x}^{(i)}||_\text{p}^\text{p}\, \mathcal{P}_\mathbf{x}(\boldsymbol{\zeta}) d\boldsymbol{\zeta} \tag{2.41}$$

In Information theory and signal processing, the quantity D_N^p is also called *distortion*. The distortion depends only on the set $\Gamma = (\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(N)})$ and on the probability distribution of $\mathbf{x}$. In summary, among all possible N-quantizers $\mathcal{Q}_N : \mathbb{R}^d \mapsto \Gamma$, all those defined by (2.38) minimize the local quantization error between $\mathbf{x}$ and $\mathcal{Q}_N(\mathbf{x})$. Therefore, a Voronoï quantizer appears as an appropriate candidate to quantize an $\mathbb{R}^d$-valued random vector $\mathbf{x}$. In addition, the probability distribution of $\mathbf{x}$ also has its importance in the quantization, since the expected value of $d_N^\text{p} = \min_{1\leq i\leq N} ||\boldsymbol{\zeta} - \mathbf{x}^{(i)}||_\text{p}^\text{p}$ is related to some distribution $\mathcal{P}_\mathbf{x}$. Overall, the distortion can be interpreted as the local quantization error d_N^p averaged over the entire domain of $\mathbf{x}$, with respect to a given probability distribution $\mathcal{P}_\mathbf{x}$. In other words, it is the expected value of the quantization error.

Main results have established that D_N^p always reaches (at least) one minimum, at some Γ_* having pairwise distinct components ($\mathbf{x}^{(i)} \neq \mathbf{x}^{(j)}$). Furthermore, this minimum decreases to zero as the size N of the optimal quantizer increases to infinity. Hence, it is of high interest to find an N-tuple Γ_* with minimal possible distortion, since it provides the best possible approximation of a random vector $\mathbf{x}$ by a random vector taking N values.

2.5.2 Numerical integration by optimal quantization

A useful feature of optimal quantization is that it allows us to perform numerical integration. Recall that $\mathbf{x}$ denotes the continuous random vector and $\mathbf{x}_\text{q}$ its quantized version. Assume that we have a nonlinear function $\mathbf{g}$, integrable in the domain $\mathbb{R}^d$ with respect to the probability density $\mathcal{P}_\mathbf{x}$, and that we want to compute the following moment integral:

$$\mathbb{E}[\mathbf{g}(\mathbf{x})] := \int_{\mathbb{R}^d} \mathbf{g}(\boldsymbol{\zeta}) \mathcal{P}_\mathbf{x}(\boldsymbol{\zeta}) d\boldsymbol{\zeta} \tag{2.42}$$

It turns out that the above expression can be approximated by the moment integral associated with $\mathbf{x}_\text{q}$, defined as

$$\mathbb{E}[\mathbf{g}(\mathbf{x}_\text{q})] := \int_{\mathbb{R}^d} \mathbf{g}(\boldsymbol{\zeta}) \mathcal{P}_{\mathbf{x}_\text{q}}(\boldsymbol{\zeta}) d\boldsymbol{\zeta} = \sum_{i=1}^N \mathbf{g}(\mathbf{x}^{(i)}) \underbrace{\mathcal{P}_\mathbf{x}(C_i(\Gamma))}_{\equiv W^{(i)}} \tag{2.43}$$

[11] Here, we define a norm $||\cdot||_\text{p}^\text{p}$ on random variables using the mean $\mathbb{E}$.

The numerical computation of this quantity needs to have access not only to the (hopefully optimal) quantizer of $\mathbf{x}$, but also to the $\mathcal{P}_{\mathbf{x}}$-mass of the Voronoï cell $\mathcal{P}_{\mathbf{x}}(C_i(\Gamma))$ (considered as the weight $W^{(i)}$ associated to the i^{th} point).

Finding a set of points $\mathbf{\Gamma} = \{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(N)}\}$, such that it minimizes the distortion D_N^{p} would allow one to find a p-optimal approximation of (2.42) by (2.43). Indeed, the following proposition is accepted [Pagès and Printems, 2003]:

Proposition 2.1. if $\mathbf{g}$ is Lipschitz[12] continuous, then

$$\begin{aligned}\left\| \int_{\mathbb{R}^d} \mathbf{g}(\boldsymbol{\zeta}) \mathcal{P}_{\mathbf{x}}(\boldsymbol{\zeta}) d\boldsymbol{\zeta} - \int_{\Gamma} \mathbf{g}(\boldsymbol{\zeta}) \mathcal{P}_{\mathbf{x}_{\text{q}}}(\boldsymbol{\zeta}) d\boldsymbol{\zeta} \right\|_{\text{p}} &= \left\| \mathbb{E}[\mathbf{g}(\mathbf{x})] - \mathbb{E}[\mathbf{g}(\mathbf{x}_{\text{q}})] \right\|_{\text{p}} \\ &\leq \mathbb{E}\left[\left\| \mathbf{g}(\mathbf{x}) - \mathbf{g}(\mathbf{x}_{\text{q}}) \right\|_{\text{p}} \right] \\ &\leq [\mathbf{g}]_{\text{Lip}} \, \mathbb{E}\left[\left\| \mathbf{x} - \mathbf{x}_{\text{q}} \right\|_{\text{p}} \right] \\ &\leq [\mathbf{g}]_{\text{Lip}} \, \sqrt[\text{p}]{D_N^{\text{p}}(\Gamma)}\end{aligned} \tag{2.44}$$

This means that if we are able to minimize the distortion, then it is possible to make the integral $\int_{\Gamma} \mathbf{g}(\boldsymbol{\zeta}) \mathcal{P}_{\mathbf{x}_{\text{q}}}(\boldsymbol{\zeta}) d\boldsymbol{\zeta}$ (weakly) converges toward $\int_{\mathbb{R}^d} \mathbf{g}(\boldsymbol{\zeta}) \mathcal{P}_{\mathbf{x}}(\boldsymbol{\zeta}) d\boldsymbol{\zeta}$. Consequently, $\sum_i \mathbf{g}(\mathbf{x}^{(i)}) \mathcal{N}(C_i(\Gamma))$ converges as well. This proposition states that the set of points Γ (resulting from solving the minimization of $D_N^{\text{p}}(\Gamma)$) would be an optimal choice to approximate the probability density of a random vector $\mathbf{x}$ transformed into a nonlinear propagation $\mathbf{g}$.

2.5.3 Stochastic gradient descent

The problem is to numerically obtain an optimal N-tuple Γ_* with minimal distortion D_N^{p}. Among the few methods provided in [Pagès and Printems, 2003] or [Pagès et al., 2004], the approach based on stochastic gradient descent is of particular interest. Before diving into it, let us first provide some admitted definitions and theorems.

Let E be a finite-dimensional $\mathbb{R}$-vector space, U a nonempty open subset of E and let $\mu(\cdot)$ be a probability measure on $\mathbb{R}^d$. Suppose that we are given a continuously differentiable function $f : U \mapsto \mathbb{R}$ with differential $df : U \mapsto E$.

Definition 2.1 (Integral representation with respect to μ)**.** We say that df has an *integral representation on U with respect to μ* if there exists a function $dF : U \times \mathbb{R}^d \mapsto E$ such that $dF(x, \cdot)$ is integrable with respect to μ for every $x \in U$, and

$$df(x) = \int_{\mathbb{R}^d} dF(x, \boldsymbol{\zeta}) d\mu(\boldsymbol{\zeta}) \tag{2.45}$$

Usually, the objective of a classical gradient method is to find an optimum of a cost function f by finding a zero of its gradient df. In a stochastic gradient, the computation of df (which supposedly has an integral representation in the sense of Definition 2.1) is

[12] A Liptschitz function is a function whose differential is bounded by some constant K called Lipschitz constant. In simple terms, it is a function whose variations are not too abrupt.

out of reach because it requires the computation of integrals with respect to μ in higher dimensions. However, we can recursively approximate a zero of df, using the local gradient dF and some independent random vectors numerically drawn from μ. Hence, we have the following definition.

Definition 2.2 (Stochastic gradient method)**.** Let f be a twice differentiable function from E to $\mathbb{R}$ such that df has an integral representation on E with respect to μ. We call *stochastic gradient method* in E for f, a triplet of sequences

$$\Big(\{x_n\}_{n\geq 0},\, \{\boldsymbol{\zeta}_n\}_{n\geq 1},\, \{\gamma_n\}_{n\geq 1}\Big)$$

with values respectively in E, $\mathbb{R}^d$ and $[0,+\infty[$ satisfying, for every $n \geq 1$,

$$x_{n+1} = x_n - \gamma_{n+1}\, dF(x_n, \boldsymbol{\zeta}_{n+1}) \tag{2.46}$$

$$\{\boldsymbol{\zeta}_n\}_{n\geq 1} \quad \text{i.i.d.} \quad \text{with the law } \mu \tag{2.47}$$

$$\gamma_n > 0 \quad \text{for every } n \geq 1 \quad \text{and} \quad \sum_{n\geq 1} \gamma_n = +\infty \tag{2.48}$$

The sequence $\{\gamma_n\}_{n\geq 1}$ is called the *step* or *gain parameter* sequence.

The stochastic gradient as defined above allows one to obtain a term dF by drawing $\boldsymbol{\zeta}_{n+1}$ with respect to μ in order to be consistent with the integral representation (2.45). The stochastic gradient method converges, as stated by the following theorem.

Theorem 2.1 (Asymptotic convergence)**.** Let $g : E \mapsto \mathbb{R}_+$ be a continuously differentiable function whose differential dg admits an integral representation on E with respect to μ. Assume that df and dF satisfy

$$\lim_{|x|\to+\infty} f(x) = +\infty \quad \text{and} \quad df \text{ is Liptschitz continuous} \tag{2.49}$$

$$\int_{\mathbb{R}^d} |dF(x,\boldsymbol{\zeta})|^2 d\mu(\boldsymbol{\zeta}) = O(f(x)) \quad \text{as} \quad |x| \to +\infty \tag{2.50}$$

Let $\Big(\{x_n\}_{n\geq 0},\, \{\boldsymbol{\zeta}_n\}_{n\geq 1},\, \{\gamma_n\}_{n\geq 1}\Big)$ be a stochastic gradient method with a positive gain parameter sequence satisfying,

$$\gamma_n \in]0,1[\,, \quad \sum_{n\geq 1} \gamma_n = +\infty \quad \text{and} \quad \sum_{n\geq 1} \gamma_n^2 < +\infty \tag{2.51}$$

Then $f(x_n)$ asymptotically converges to some nonnegative optimum value $f_* \in \mathbb{R}_+$ and X_n asymptotically converges toward some random connected component x_* of $\{df = 0\} \cap \{f = g_*\}$. In particular,

$$x_n \to x_* \quad \text{asymptotically,} \quad \text{as } n \to +\infty. \tag{2.52}$$

2.5.4 Optimal quantization of a normally-distributed random vector

Given the previous definitions and theorems, we may apply the stochastic gradient method by considering $f = D_N^{\mathrm{p}}(\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(N)})$ to be the function to minimize in order to

find the p-optimal quantization of a Gaussian-distributed random vector $\mathbf{x} \in \mathbb{R}^d$ ($\mathbf{x} \sim \mathcal{N}_\mathbf{x}$). We deal with the following optimization problem (OP):

$$(\text{OP}) : \begin{cases} \text{Find a } N\text{-tuple } \Gamma_* = (\mathbf{x}_*^{(1)}, \dots, \mathbf{x}_*^{(N)}) \text{ s.t.} \\ D_N^{\text{p}}(\mathbf{x}_*^{(1)}, \dots, \mathbf{x}_*^{(N)}) \leq D_N^{\text{p}}(\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(N)}), \quad \forall \Gamma = (\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(N)}) \in (\mathbb{R}^d)^N, \end{cases}$$

where in this case, $D_N^{\text{p}} : (\mathbb{R}^d)^N \mapsto \mathbb{R}^+$ and writes

$$D_N^{\text{p}}(\Gamma) = \sum_{i=1}^N \int_{C(\mathbf{x}^{(i)})} \|\mathbf{x}^{(i)} - \boldsymbol{\zeta}\|_{\text{p}}^{\text{p}} \, \mathcal{N}_\mathbf{x}(\boldsymbol{\zeta}) d\boldsymbol{\zeta} \tag{2.53}$$

where $C(\mathbf{x}^{(i)}) \equiv C_i(\Gamma)$ for notational convenience. It can be shown that D_N^{p} is continuously differentiable and that its integral representation in the Gaussian case, with $U = \mathbb{R}^d$ (cf. Definition 2.45), is

$$df = d\left(D_N^{\text{p}}(\Gamma)\right) = \mathbb{E}\left[\mathbf{1}_{C(\mathbf{x}^{(i)})}(\boldsymbol{\zeta})(\mathbf{x}^{(i)} - \boldsymbol{\zeta})\right] = \int_{\mathbb{R}^d} \underbrace{\left[\mathbf{1}_{C(\mathbf{x}^{(i)})}(\boldsymbol{\zeta})(\mathbf{x}^{(i)} - \boldsymbol{\zeta})\right]}_{dF(\mathbf{x}^{(i)}, \boldsymbol{\zeta})} \mathcal{N}_\mathbf{x}(\boldsymbol{\zeta}) d\boldsymbol{\zeta} \tag{2.54}$$

Subsequently, the stochastic gradient procedure minimizing D_N^{p} is given by

$$\mathbf{x}_{n+1}^{(i)} = \mathbf{x}_n^{(i)} - \gamma_{n+1} \mathbf{1}_{C(\mathbf{x}_n^{(i)})}(\boldsymbol{\zeta}_{n+1})(\mathbf{x}_n^{(i)} - \boldsymbol{\zeta}_{n+1}), \quad \forall i = 1, \dots, N \quad \text{and} \quad n \geq 0 \tag{2.55}$$

Or equivalently, if we define i_{win} as the integer such that $\boldsymbol{\zeta}_{n+1} \in C(\mathbf{x}_n^{(i_{\text{win}})})$,

$$\begin{cases} \mathbf{x}_{n+1}^{(i)} = \mathbf{x}_n^{(i)} - \gamma_{n+1}(\mathbf{x}_n^{(i)} - \boldsymbol{\zeta}_{n+1}) & \text{if } \; i = i_{\text{win}} \\ \mathbf{x}_{n+1}^{(i)} = \mathbf{x}_n^{(i)} & \text{if } \; i \neq i_{\text{win}} \end{cases} \tag{2.56}$$

This procedure is known as the *Competitive Learning Vector Quantization* (CLVQ) algorithm. It is decomposed into two phases:

- Competitive phase: Selection of the "winning index" i_{win} using a closest neighbor search
- Learning phase: Update of the winning component by a homothety centered at $\boldsymbol{\zeta}_{n+1}$ with ratio $(1 - \gamma_{n+1})$. Hence, γ can also be considered as a learning rate.

From a numerical point of view the most time consuming task is to compute the winner component $\mathbf{x}_n^{(i)}$ that is the closest to $\boldsymbol{\zeta}_{n+1}$.

During this procedure, it is possible to compute some "companions" as by-products. On the first hand, one can compute the $\mathcal{N}_\mathbf{x}$-masses, $\mathcal{N}_\mathbf{x}\left(C(\mathbf{x}^{(i)})\right)$ $\forall i = 1, \dots, N$, of the Voronoï cells, which can be used as weights $W^{(i)}$. To do that, one simply increments a counter as follows:

$$c_{n+1}^{(i)} = c_n^{(i)} + \mathbf{1}_{\{i = i_{\text{win}}\}}, \quad \forall i = 1, \dots, N \tag{2.57}$$

so that $c_n^{(i)}/n \to \mathcal{N}_\mathbf{x}\left(C(\mathbf{x}_*^{(i)})\right)$ as $n \to +\infty$. On the second hand, it is possible to compute the resulting distortion by proceeding in a similar way: Initialize $D_{N,0}^p = 0$ and perform

$$D_{N,n+1}^{\text{p}} = D_{N,n}^{\text{p}} + ||\mathbf{x}_n^{i_{\text{win}}} - \boldsymbol{\zeta}_{n+1}||_{\text{p}}^{\text{p}} \tag{2.58}$$

so that $D_{N,n}^{\text{p}}/n \to D_N^{\text{p}}(\Gamma_*)$ as $n \to +\infty$.

These results will be useful for the subsequent developments of Chapter 5. For the time being, let us continue with the literature review of filtering methods.

• • •

All the previous Kalman filtering techniques rely on a Euclidean framework, where addition and metrics are well-defined in vector spaces. However, the literature on navigation and robotics is filled with state spaces that often involve rotations coupled with translations, all embedded into non Euclidean manifolds, for which such operations lose their geometric meaning. This required an intrinsic and geometrically-aware formulation for the design of state estimators.

2.6 Extension to symmetry-preserving observers

When designing a state estimator, it is often desirable to preserve the geometric constraints between the variables contained in the state vector. In many estimation problems, the state variables are not independent, but coupled through the underlying geometry of the system's state space. Applying additive corrections within the Euclidean framework does not necessarily preserve this structure, and the resulting posterior estimate may no longer be meaningful with respect to the interdependence of the state components. The classical Euclidean formulation of Kalman filters does not really care about this relationship between state variables. In other words, it does not necessarily consider the underlying manifold in which the state variables live.

2.6.1 Introductive example

To illustrate the non-meaningless of the additive correction term used by standard Kalman filters, we consider the example of a variable $\boldsymbol{\chi}$, living in the 2-sphere manifold as shown in Figure 2.4 (the same example is provided in [Brossard, 2020]). Consider an initial estimate $\hat{\mathbf{x}}_{k|k-1}$ satisfying the following constraint:

$$\boldsymbol{\chi}_k \in \mathbb{S}^2 := \{\mathbf{x}_k \in \mathbb{R}^3 \mid ||\mathbf{x}_k|| = 1\} \tag{2.59}$$

Assume that we get an observation $\mathbf{y}_k$, and apply a Kalman correction term as

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}\big(\mathbf{y}_k - \mathbf{h}(\hat{\mathbf{x}}_{k|k-1})\big). \tag{2.60}$$

It follows that the posterior estimate $\hat{\mathbf{x}}_{k|k}$ has no reason to preserve the geometrical constraint. As a consequence, the variable $\hat{\boldsymbol{\chi}}_{k|k}$ (extracted from $\hat{\mathbf{x}}_{k|k}$) also has no reason to remain in the manifold $\mathbb{S}^2$. The Kalman filter indeed assumes that $\mathbf{x}$ lives in a vector space $\mathbb{R}^d$. However, most applications require the need to directly work on non Euclidean manifolds instead of Euclidean vector spaces. This example shows the type of problem

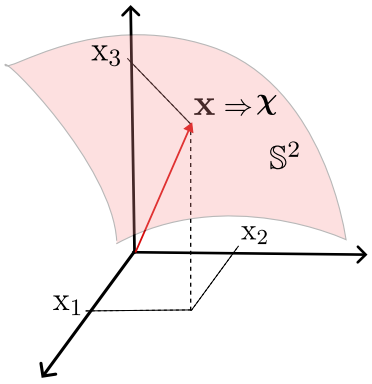


Figure 2.4: Example of a state living in the 2-sphere. $\mathbf{x} = [\mathrm{x}_1, \mathrm{x}_2, \mathrm{x}_3]^\top$.

that classical estimation methods have to deal with.

Preserving geometrical constraints on a state space is fundamentally tied to preserving the symmetries of the underlying manifold. In mathematics, a *symmetry* refers to a

transformation that leaves the intrinsic properties of an object unchanged. In navigation, physical systems naturally exhibit symmetries through their dynamical models. Indeed, the fundamental laws governing the motion of a rigid body are invariant by translations and rotations. Identifying these transformations for a given system model enables the design of observers that respect and preserve the associated structural invariances. This perspective led to the theory of symmetry-preserving (or invariant) observers in the early 2000s, with developments continuing up to recent years [Barrau and Bonnabel, 2022].

The rise of symmetry-preserving observers dates back to the early 2000s, with the pioneering work of [Aghannan and Rouchon, 2002], then further detailed in [Bonnabel, 2007a][Bonnabel et al., 2008]. The systematic construction of invariant observers requires to look for transformations (or changes of variables) that preserve the structure of dynamical systems, through different definitions related to Lie groups. Lie groups are differential manifolds that satisfy group axioms: closure under a group operation $\circ$, identity element $\boldsymbol{e}$, inverse element, and associativity. Elements of a Lie group come with the power to transform elements of other sets, producing, e.g. rotated vectors, translated vectors, scaled vectors, and combinations of them.

2.6.2 Invariant systems

Consider the system described by the following (noise-free for the time being) nonlinear model (formalized in continuous-time)

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}, \mathbf{u}) \\ \mathbf{y} &= \mathbf{h}(\mathbf{x}, \mathbf{u}) \end{aligned} \tag{2.61}$$

The first pre-requisite to build symmetry-preserving observers is to identify the group of transformations for which the system (2.61) writes the same after transformations. This requires some generic definitions.

Definition 2.3 (Group action)**.** Let $\boldsymbol{G}$ be a Lie group and $\mathcal{V}$ an open set of dimension $q = \dim(\mathcal{V})$. Let $\boldsymbol{g} \in \boldsymbol{G}$ and $\mathbf{v} \in \mathcal{V}$. We denote by $\varphi_{\boldsymbol{g}}(\mathbf{v})$ the action of $\boldsymbol{g}$ on $\mathbf{v}$,

$$\begin{aligned} \varphi_{\boldsymbol{g}} : \mathcal{V} &\mapsto \mathcal{V} \\ \mathbf{v} &\mapsto \varphi_{\boldsymbol{g}}(\mathbf{v}) \end{aligned} \tag{2.62}$$

For $\{\varphi_{\boldsymbol{g}}\}_{\boldsymbol{g}\in\boldsymbol{G}}$ to be a group action, it must satisfy the following properties:

$$\text{Identity: } \exists \boldsymbol{e} \in \boldsymbol{G}, \quad \varphi_{\boldsymbol{e}}(\mathbf{v}) = \mathbf{v} \tag{2.63}$$

$$\text{Compatibility: } \varphi_{(\boldsymbol{g}_2 \circ \boldsymbol{g}_1)}(\mathbf{v}) = \varphi_{\boldsymbol{g}_2}\big(\varphi_{\boldsymbol{g}_1}(\mathbf{v})\big) \quad \forall \boldsymbol{g}_1, \boldsymbol{g}_2 \in \boldsymbol{G} \tag{2.64}$$

Hence, considering the system (2.61) where $\mathbf{x} \in \mathcal{X} \subset \mathbb{R}^{\dim(\mathbf{x})}$, $\mathbf{u} \in \mathcal{U} \subset \mathbb{R}^{\dim(\mathbf{u})}$, and $\mathbf{y} \in \mathcal{Y} \subset \mathbb{R}^{\dim(\mathbf{y})}$, we define the pair of transformations $(\varphi_{\boldsymbol{g}}, \psi_{\boldsymbol{g}})$ acting on $(\mathcal{X} \times \mathcal{U})$, such that

$$\Big(\mathbf{X} = \varphi_{\boldsymbol{g}}(\mathbf{x}),\ \mathbf{U} = \psi_{\boldsymbol{g}}(\mathbf{u})\Big) \tag{2.65}$$

and for which the system (2.61) is said to be G-invariant under the group of transformations hereby defined. A G-invariant system is defined as follows:

Definition 2.4 (G-INVARIANT SYSTEM)**.** A system $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u})$ is said G-invariant if $\forall(\boldsymbol{g}, \mathbf{x}, \mathbf{u}) \in \boldsymbol{G} \times \mathcal{X} \times \mathcal{U}$, $\exists(\varphi_{\boldsymbol{g}}, \psi_{\boldsymbol{g}})$ such that:

$$\mathbf{f}\left(\varphi_{\boldsymbol{g}}(\mathbf{x}), \psi_{\boldsymbol{g}}(\mathbf{u})\right) = D\varphi_{\boldsymbol{g}}(\mathbf{x}) \cdot \mathbf{f}(\mathbf{x}, \mathbf{u}) \tag{2.66}$$

where $D\varphi_{\boldsymbol{g}}(\mathbf{x})$ denotes the differential mapping evaluated at $\varphi_{\boldsymbol{g}}(\mathbf{x})$. Equivalently, the property also writes $\dot{\mathbf{X}} = \mathbf{f}(\mathbf{X}, \mathbf{U})$, i.e., the system is left unchanged under the change of coordinates (2.65).

To build invariant observers, a condition is also required on the output, which states that $\mathbf{y} = \mathbf{h}(\mathbf{x}, \mathbf{u})$ must be G-equivariant:

Definition 2.5 (G-EQUIVARIANT OUTPUT)**.** The output $\mathbf{y} = \mathbf{h}(\mathbf{x}, \mathbf{u})$ is said G-equivariant if, $\forall(\boldsymbol{g}, \mathbf{x}, \mathbf{u}) \in \boldsymbol{G} \times \mathcal{X} \times \mathcal{U}$, there exists an transformation $\varrho_{\boldsymbol{g}}$ acting on $\mathcal{Y}$ such that

$$\mathbf{h}\left(\varphi_{\boldsymbol{g}}(\mathbf{x}), \psi_{\boldsymbol{g}}(\mathbf{u})\right) = \varrho_{\boldsymbol{g}}\left(\mathbf{h}(\mathbf{x}, \mathbf{u})\right) \tag{2.67}$$

In the following, the system (2.61) is always considered G-invariant with G-equivariant output. For this type of system, it is possible to build candidate observers (pre-observers) that preserve this symmetry, while providing additional advantages over classical observers.

2.6.3 Symmetry-preserving observers

Definition 2.6 (PRE-OBSERVER)**.** The system $\dot{\hat{\mathbf{x}}} = \mathbf{f}_{\text{est}}(\hat{\mathbf{x}}, \mathbf{u}, \mathbf{y})$ is a pre-observer of (2.61) if $\forall\hat{\mathbf{x}}, \mathbf{u}$, we have $\mathbf{f}_{\text{est}}(\hat{\mathbf{x}}, \mathbf{u}, \mathbf{y}) = \mathbf{f}(\mathbf{x}, \mathbf{u})$.

If $\hat{\mathbf{x}}(t) \to \mathbf{x}(t)$ as $t \to +\infty$ for any initial condition sufficiently close, the pre-observer is an asymptotic observer.

Definition 2.7 (INVARIANT PRE-OBSERVER)**.** The pre-observer $\dot{\hat{\mathbf{x}}} = \mathbf{f}_{\text{est}}(\hat{\mathbf{x}}, \mathbf{u}, \mathbf{y})$ is G-invariant if $\forall\boldsymbol{g}, \hat{\mathbf{x}}, \mathbf{u}, \mathbf{y}$,

$$\mathbf{f}_{\text{est}}(\varphi_{\boldsymbol{g}}(\hat{\mathbf{x}}), \psi_{\boldsymbol{g}}(\mathbf{u}), \varrho_{\boldsymbol{g}}(\mathbf{y})) = D\varphi_{\boldsymbol{g}}(\hat{\mathbf{x}}) \cdot \mathbf{f}_{\text{est}}(\hat{\mathbf{x}}, \mathbf{u}, \mathbf{y}) \tag{2.68}$$

In general, the usual state error $\tilde{\mathbf{x}} = \hat{\mathbf{x}} - \mathbf{x}$ and output error $\tilde{\mathbf{y}} = \hat{\mathbf{y}} - \mathbf{y}$ do not preserve the geometry of the system. We already saw that an observer consists of a copy of the dynamics $\mathbf{f}(\mathbf{x}, \mathbf{u})$ endowed with a correction term $\Delta(\hat{\mathbf{x}}, \mathbf{u}, \mathbf{y})$. If a traditional error is used (whether for the state or for the output), then the correction term no longer allows the observer to be invariant under the action of a group element $\boldsymbol{g}$. The key idea, as proposed in [Aghannan and Rouchon, 2002], is to use an invariant output error instead of a linear output error.

The construction of a symmetry-preserving observer for the system (2.61) is conditioned by the existence of a mobile frame $\boldsymbol{\gamma}(\mathbf{x}) \in \boldsymbol{G}$ at any point $\mathbf{x} \in \mathcal{X}$, allowing us to find invariant frames from which it is possible to derive a correction term $\Delta(\hat{\mathbf{x}}, \mathbf{u}, \mathbf{y})$ that

preserves the symmetry properties of the original system. The method relies on Cartan's moving frame [Olver, 1999].

Before presenting this method, we define base and fiber coordinates. We assume that $\boldsymbol{G}$ is the group of dimension $r \leq \dim(\mathcal{X})$ that acts on $(\mathcal{X},\mathcal{U})$ (see (2.65)). The group action restricted to $\mathcal{X}$ generates orbits. The moving frame method yields a set of fundamental local invariants $\boldsymbol{z}_b \in \mathbb{R}^{\dim(\mathcal{X})-r}$ of the group action restricted to $\mathcal{X}$ alone, namely $\varphi_{\boldsymbol{g}}(\mathbf{x})$. It is completed with $\boldsymbol{z}_a \in \mathbb{R}^r$ so that $\boldsymbol{z} = (\boldsymbol{z}_a, \boldsymbol{z}_b)$ form coordinates of $\mathcal{X}$. These coordinates are called fiber $(\boldsymbol{z}_a)$ and base $(\boldsymbol{z}_b)$ coordinates[13], such that $\varphi_{\boldsymbol{g}}$ can be decomposed into $\varphi^a_{\boldsymbol{g}}$ and $\varphi^b_{\boldsymbol{g}}$. Besides, it is always possible to choose $\boldsymbol{z}_a$ such that, $\forall \boldsymbol{g} \in \boldsymbol{G}$, the group action writes

$$\varphi_{\boldsymbol{g}}(\boldsymbol{z}_a, \boldsymbol{z}_b) = (\varpi_{\boldsymbol{g}}(\boldsymbol{z}_a), \boldsymbol{z}_b)$$

with $\varpi_{\boldsymbol{g}}(\mathbf{x})$ invertible $\forall \mathbf{x} \in \mathbb{R}^r$ (meaning that $\boldsymbol{z}_a$ is a parameterizaion of the elements of the group). Let $\boldsymbol{\gamma}$ be the moving frame which maps $\boldsymbol{z}$ to the coordinate cross-section $\{\boldsymbol{z}_a = \mathbf{c}\}$. The invariant dynamics $\mathbf{f}$ writes locally in the new coordinates:

$$\begin{aligned} \dot{\boldsymbol{z}}_a &= D\varphi^a_{\boldsymbol{\gamma}(z)^{-1}} \mathbf{f}_a \left(c, \boldsymbol{z}_b, \psi_{\boldsymbol{\gamma}(z)}(\mathbf{u})\right) \\ \dot{\boldsymbol{z}}_b &= \mathbf{f}_b \left(c, \boldsymbol{z}_b, \psi_{\boldsymbol{\gamma}(z)}(\mathbf{u})\right) \end{aligned} \tag{2.69}$$

since the system is invariant.

In particular, Cartan's moving frame method requires the solution of the normalization equation $\varphi_{\boldsymbol{g}}(\hat{\mathbf{x}}) = \mathbf{c}$, consisting of finding the element $\boldsymbol{g}$ for which we have $\boldsymbol{g} = \boldsymbol{\gamma}(\hat{\mathbf{x}})$. One can say that the quantity $\varphi_{\boldsymbol{g}}(\mathbf{x}) = \mathbf{c}$ defines a coordinate cross-section to the orbits, and $\boldsymbol{g} = \boldsymbol{\gamma}(\mathbf{x})$ is the unique group element that maps $\mathbf{x}$ to the cross-section (Figure 2.5).

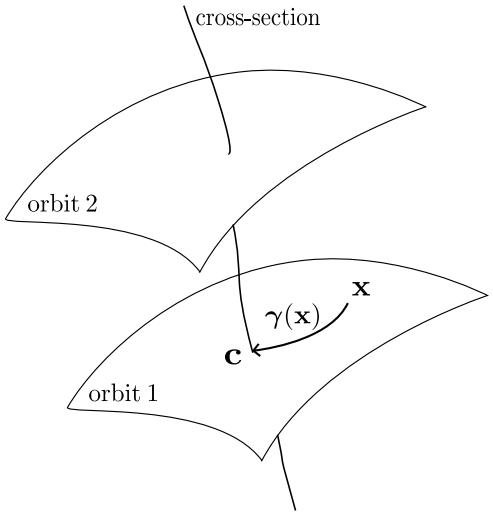


Figure 2.5: Orbits of the group action and definition of the function $\boldsymbol{\gamma}$.

Besides, an invariant output error $E(\hat{\mathbf{x}}, \mathbf{u}, \mathbf{y})$ must be defined, which is given by

$$E(\hat{\mathbf{x}}, \mathbf{u}, \mathbf{y}) := \varrho_{\boldsymbol{\gamma}(\hat{\mathbf{x}})}(\mathbf{h}(\hat{\mathbf{x}}, \mathbf{u})) - \varrho_{\boldsymbol{\gamma}(\hat{\mathbf{x}})}(\mathbf{y}) \tag{2.70}$$

[13] Base coordinates $(\boldsymbol{z}_b)$ numerize the orbits, and fiber coordinates $(\boldsymbol{z}_a)$ parameterize these orbits.

thus replacing the linear output error $\tilde{\mathbf{y}} = \mathbf{h}(\hat{\mathbf{x}}, \mathbf{u}) - \mathbf{y}$. It follows that a set of invariants $I(\hat{\mathbf{x}}, \mathbf{u})$ can be found and defined as

$$I(\hat{\mathbf{x}}, \mathbf{u}, \mathbf{y}) := \big(\psi_{\boldsymbol{\gamma}(\hat{\mathbf{x}})}(\mathbf{u}),\ \varrho_{\boldsymbol{\gamma}(\hat{\mathbf{x}})}(\mathbf{y})\big) \tag{2.71}$$

The last step of the constructive method requires finding the invariant frame of vector fields $(\boldsymbol{w}_1(\mathbf{x}), \dots, \boldsymbol{w}_n(\mathbf{x}))$, representing a canonical basis of the tangent space of $\mathcal{X}$ at $\mathbf{x}$. A vector field $\boldsymbol{w}_i(\mathbf{x})$ is defined as

$$\boldsymbol{w}_i(\mathbf{x}) := \big(D\varphi_{\boldsymbol{\gamma}(\mathbf{x})}(\mathbf{x})\big)^{-1} \frac{\partial}{\partial x_i}, \quad \forall i = 1, \dots, n \tag{2.72}$$

Once all of these quantities have been found, an invariant (pre)-observer is given by the following expression

$$\begin{aligned} &\mathbf{f}_{\text{est}}(\hat{\mathbf{x}}, \mathbf{u}, \mathbf{y}) := \mathbf{f}(\hat{\mathbf{x}}, \mathbf{u}) + \Delta(\hat{\mathbf{x}}, \mathbf{u}, \mathbf{y}) \\ &\text{with,} \quad \Delta(\hat{\mathbf{x}}, \mathbf{u}, \mathbf{y}) = \begin{bmatrix} \boldsymbol{w}_1 & \dots & \boldsymbol{w}_n \end{bmatrix} \begin{pmatrix} K_1\left(I(\hat{\mathbf{x}}, \mathbf{u}), E(\hat{\mathbf{x}}, \mathbf{u}, \mathbf{y})\right) \\ \vdots \\ K_n\left(I(\hat{\mathbf{x}}, \mathbf{u}), E(\hat{\mathbf{x}}, \mathbf{u}, \mathbf{y})\right) \end{pmatrix} E(\hat{\mathbf{x}}, \mathbf{u}, \mathbf{y}) \end{aligned} \tag{2.73}$$

where the correction term Δ ensures that the updated estimate satisfies the geometric constraints.

This constructive method is systematic for the design of pre-observers: in [Bonnabel, 2007b], it has been applied to derive an Invariant EKF (IEKF), and to solve an attitude estimation problem on quaternions, taking noise modeling into account. However, it relies on necessary conditions regarding the smoothness and properties of the group action on the noise, which remain an obstacle to overcome if one wishes to make it systematic for the design of a filter. Despite from that, the pre-observer comes up with a particularly interesting property: the time evolution of the state error $\boldsymbol{\eta}(\hat{\mathbf{x}}, \mathbf{x}) = \varphi_{\gamma(\mathbf{x})}(\hat{\mathbf{x}}) - \varphi_{\gamma(\mathbf{x})}(\mathbf{x})$ depends only on itself and the known invariants $I(\hat{\mathbf{x}}, \mathbf{u})$, i.e., we have

$$\dot{\boldsymbol{\eta}} = \Upsilon\big(\boldsymbol{\eta}, I(\hat{\mathbf{x}}, \mathbf{u})\big) \tag{2.74}$$

for some smooth function Υ. Given the fact that $I(\hat{\mathbf{x}}, \mathbf{u})$ is invariant, whatever the value of $\hat{\mathbf{x}}$, the error is completely autonomous, a property akin to linear systems.

Finally, if the transformation $\varphi_{\boldsymbol{g}}$ is full rank, then additional properties allow us to simplify the procedure. A group action is said to be full rank if its dimension r matches the dimension the group $\boldsymbol{G}$. Besides, if $r = \dim(\boldsymbol{G}) = \dim(\mathcal{X})$, then $\boldsymbol{G}$ and $\mathcal{X}$ can be identified with each other, and the action $\varphi_{\boldsymbol{g}}$ simply boils down to a left (or right) group operation

$$\varphi_{\boldsymbol{g}}(\mathbf{x}) = \begin{cases} \mathsf{L}_{\boldsymbol{g}}(\mathbf{x}) = \boldsymbol{g} \circ \mathbf{x} & \text{(left group operation)} \\ \mathsf{R}_{\boldsymbol{g}}(\mathbf{x}) = \mathbf{x} \circ \boldsymbol{g} & \text{(right group operation)} \end{cases} \tag{2.75}$$

In this case, there is only one orbit, and the cross-section to this orbit is $\mathbf{c} = \boldsymbol{e}$ (identity element). Solving the normalization equation by taking $\mathbf{c} = \boldsymbol{e}$ allows us to get $\boldsymbol{\gamma}(\hat{\mathbf{x}}) = \hat{\mathbf{x}}^{-1}$, since

$$\varphi_{\boldsymbol{g}}(\hat{\mathbf{x}}) = \mathbf{c} = \boldsymbol{g} \circ \hat{\mathbf{x}} = \boldsymbol{e} \quad \Rightarrow \quad \boldsymbol{\gamma}(\hat{\mathbf{x}}) = \boldsymbol{g} = \hat{\mathbf{x}}^{-1}$$

In this case, the constructive method simplifies, and the invariant frame can always be obtained by resorting to the identity element $\boldsymbol{e}$. In fact, the tangent space at the identity plays a key role, and if a state space $\mathcal{X}$ can be identified with a group $\boldsymbol{G}$, then the theory of Lie groups [Chirikjian, 2011] provides us with a large panel of tools, particularly useful for robotic applications [Solà et al., 2021], where the geometry of the problem (involving rotations and translations) needs to be specifically formalized by such mathematical objects.

2.7 Invariant observers on Lie groups

Instead of working with a state vector $\mathbf{x} \in \mathcal{X} \subset \mathbb{R}^{\dim(\mathbf{x})}$ and a G-invariant system as in (2.61), it can be more convenient to work directly on the underlying manifold $\boldsymbol{G}$. If $\boldsymbol{G}$ can be identified with $\mathcal{X}$, the invariant frame can be obtained via the identity element. This frame is closely related to the Lie algebra, defined as the tangent space at the identity. Then, introducing the notion of group error, it becomes possible to formulate invariant observers in a way that naturally incorporates the system's symmetries while simplifying the underlying mathematical operations. This principle underlies Kalman filtering on Lie groups, which has been extensively developed over the past decade.

2.7.1 Kalman filters on Lie groups

Given the fact that errors can be described by exploiting the Lie algebra, prior works on uncertainty characterization for Lie groups have been introduced [Wang and Chirikjian, 2006],[Wolfe et al., 2011], [Chirikjian, 2011], and extend metrics, probability measures and Bayesian filtering to these manifolds. Kalman filtering on matrix Lie groups fully emerged after these developments to address stochastic pose estimation problems. It was first [Bourmaud et al., 2013], who began to take advantage of recent developments on uncertainty characterization/propagation from Chirikjian, et al., in order to produce a discrete EKF (D-LG-EKF) that applies directly on matrix Lie groups. Later, a continuous-discrete EKF (CD-LG-EKF) has been formalized in [Bourmaud et al., 2015] for continuous-time propagation and discrete-time update. This LG-EKF made extensive use of the Lie group theory, the concentrated Gaussian distribution introduced in [Wang and Chirikjian, 2006], and the Lie algebra of the group, where differential quantities such as group jacobians are well-defined. On matrix Lie groups, a stochastic propagation model takes the form of a group multiplication, e.g., $\dot{\boldsymbol{\chi}} = \boldsymbol{f}(\boldsymbol{\chi}, \mathbf{u}, \mathbf{w}) = \boldsymbol{\chi}(\mathbf{u} + \mathbf{w})^{\wedge}$, where $[\cdot]^{\wedge}$ denotes an element of the Lie algebra. In this context, the Baker-Campbell-Hausdorff (BCH) [Campbell, 1897] formula plays a prominent role (similar to partial derivatives in the TSE), as it provides a way to propagate uncertainty in the Lie algebra when group multiplication is performed. Hence, the LG-EKF prediction step can be considered as a 1^{st}-order approximation of the BCH formula, analogous to the traditional 1^{st}-order approximation of the TSE when using a classical EKF prediction. The update step also takes the form of a group multiplication, and ensures that the posterior estimate still lives in the group.

After that, and similarly to their Euclidean counterparts, a plethora of Lie group-based Kalman filters have been developed and tested on many experiments. Regarding the "Extended" approach, [Barrau and Bonnabel, 2016] first changed their original IEKF equations to cast them into a matrix Lie group framework, more handy to use than the formulation (2.73). They showed that the IEKF possesses guaranteed stability properties that the EKF lacks (although the study of stability properties was possible only in a deterministic context). Indeed, it has been shown in [Barrau and Bonnabel, 2015] that the IEKF solved inconsistency issues encountered with the usual EKF in the context of SLAM. Other versions of the IEKF (sometimes just applications within specific cases) can be found in the literature to solve numerous engineering problems: [Hartley et al., 2018], [Phogat and Chang, 2020], [Brossard et al., 2020a], [Lin et al., 2021] [Labsir et al., 2022].

Unscented approach

It was also during this period that the Unscented approach, in the continuity of the prior works of Julier and Uhlmann, was adopted to come up with Invariant UKFs. Early attempts to merge invariant theory with the UKF methodology can be found in [Condomines et al., 2013] and [Condomines et al., 2014]. The latter, named π-IUKF, consisted of a reformulation of the observation equation $\mathbf{h}$ via a group transformation, denoted π, such that the output is G-equivariant with the considered dynamical equations. Later, [Loianno et al., 2016] used the structure of $\boldsymbol{SE}(3)$ with concentrated Gaussian assumptions to come up with a SE(3)-UKF designed for quadrotors equipped with Visual Inertial Odometry (VIO) sensors. Following the same philosophy, [Brossard et al., 2017] designed a generic formulation of the UKF-LG that is not restricted to a single Lie group, but can be applied to possibly all matrix Lie groups. It was later applied to a problem of Visual Inertial SLAM [Brossard et al., 2018] for the case of the extended Special Euclidean group $\boldsymbol{SE}_{2+N}(3)$ to track the body attitude, velocity, and position of a UAV, as well as the position of N landmarks. The difference from the SE(3)-UKF of Loianno et al. concerns the final covariance update, where the SE(3)-UKF uses parallel transport while the UKF-LG uses the adjoint operator and the group jacobian. Another formulation of the UKF on Lie groups is proposed in [Sjøberg and Egeland, 2022], where the time propagation is directly expressed in the Lie algebra thanks to a reformulation of the group jacobian. The resulting Kalman filter, called the Lie-Algebraic UKF (LA-UKF), is given in terms of logarithms in vector form, which makes its implementation similar to a standard UKF in $\mathbb{R}^d$. It retains the global performance of the UKF-LG [Brossard et al., 2017] while reducing computational cost. Finally, [Brossard et al., 2020b] proposed the UKF-M, a generalization of the UKF-LG to parallelizable manifolds (a class of manifolds that includes matrix Lie groups).

2.7.2 Other Lie group-based observers

In parallel to the aforementioned filters, the complementary filter on $\boldsymbol{SO}(3)$ was introduced in [Mahony et al., 2005] and later extended in [Mahony et al., 2008]. These works

made extensive use of the invariant errors on $\boldsymbol{SO}(3)$ and its properties to address attitude estimation and gyroscope bias correction for a low-cost IMU in UAV applications. This filter resembles the traditional complementary filter in that it fuses high-frequency gyroscope measurements with low-frequency accelerometer measurements so that they complete each other. Note that the complementary filter on $\boldsymbol{SO}(3)$ proposed by Mahony, Hamel et al. (although able to account for sensor noise and reduce them) still remains a "deterministic filter" in the sense that it does not treat $\boldsymbol{\chi}$ as a random variable. Nonetheless, the proposed observer design is based on a Lyapunov analysis, ensuring (almost) global stability of the error. This stability domain being only "almost global" is due to the structure of $\boldsymbol{SO}(3)$ (the related issues will be discussed later in the context of this thesis). The authors of [Wu et al., 2015] addressed this issue and proposed a global asymptotic attitude observer on $\boldsymbol{SO}(3)$. Many other observers defined on Lie groups have been derived, building upon the works of Mahony et al., [Hua et al., 2011; Scandaroli et al., 2011; Khosravian et al., 2015; Bouazza et al., 2023; van Goor et al., 2023]. However, the previous deterministic approaches must be distinguished from the Bayesian estimation problem addressed in this manuscript, whose objective is to determine a probability density, or at least the relevant statistical moments that allow us to fully characterize the uncertainty associated with the estimated state.

In summary, the chronological timeline of invariant observers, from the early 2000s to nowadays, can be drawn in Figure 2.6. Overall, we can highlight three main developments (the chronology is approximative): the first one consisted in exploiting symmetry, an idea initiated by Rouchon, Aghannan, Bonnabel, Salaün, Mahony, Hamel et al. during the first decade. The second development was more in line with extending Lie groups theory so that it can apply to filtering problems (taking noise into account), especially by reconnecting with the general Bayesian framework. This development was carried out by the works of Chirikjian, Wolfe, Barfoot et al. at the beginning of the second decade. Building upon these works and with extensive use of the equivariant concentrated Gaussian uncertainty characterization, EKFs and UKFs directly expressed on Lie groups have thrived. Last but not least, further sample-based methods were developed, some along the lines of UKFs, while others preferred to take the direction of Particle filtering [Chahbazian, 2023], [Lopez et al., 2024]. More recently, there are many directions being pursued by invariant observers based on Lie groups:

- Fusion with deep learning: [Brossard, 2020], [Tan et al., 2023], [Falorsi et al., 2019]
- Uncertainty characterization and propagation: [Mangelson et al., 2020], [Ye et al., 2025], [Brossard et al., 2022]
- Point-clouds matching (*registration*): [Brossard et al., 2020c] [Tao et al., 2022]
- Coupling with directional statistics: [Kurz et al., 2014], [Marković et al., 2016], [Lee, 2018], [Wang and Lee, 2022]
- Cluster of space debris: [Labsir, 2020]

Among all of these directions, the present thesis places strong interest in UKFs on Lie groups. Although they differ from classical UKFs in the sense that a state matrix replaces a state vector, similar limitations persist, especially in the generation of sigma-points for the UT. Indeed, existing Lie group–based UKFs still rely on classical sigma-point constructions on a vector space. The problematic that consists to generate sigma-points on a vector space was already outlined in subsection §2.4.3. Using the UKF-LG of [Brossard et al., 2017] as a reference, a part of the contributions aims to address these issues. Another part of the contributions addresses uncertainty modeling for random variables evolving on the motion group $\boldsymbol{SE}(n)$, with an emphasis on questions related to the concentrated equivariant Gaussian framework. To clarify our positionning within the current state of the art, we underline the works that directly relate to these challenges in Figure 2.6. In the next chapter, we further develop these concerns.

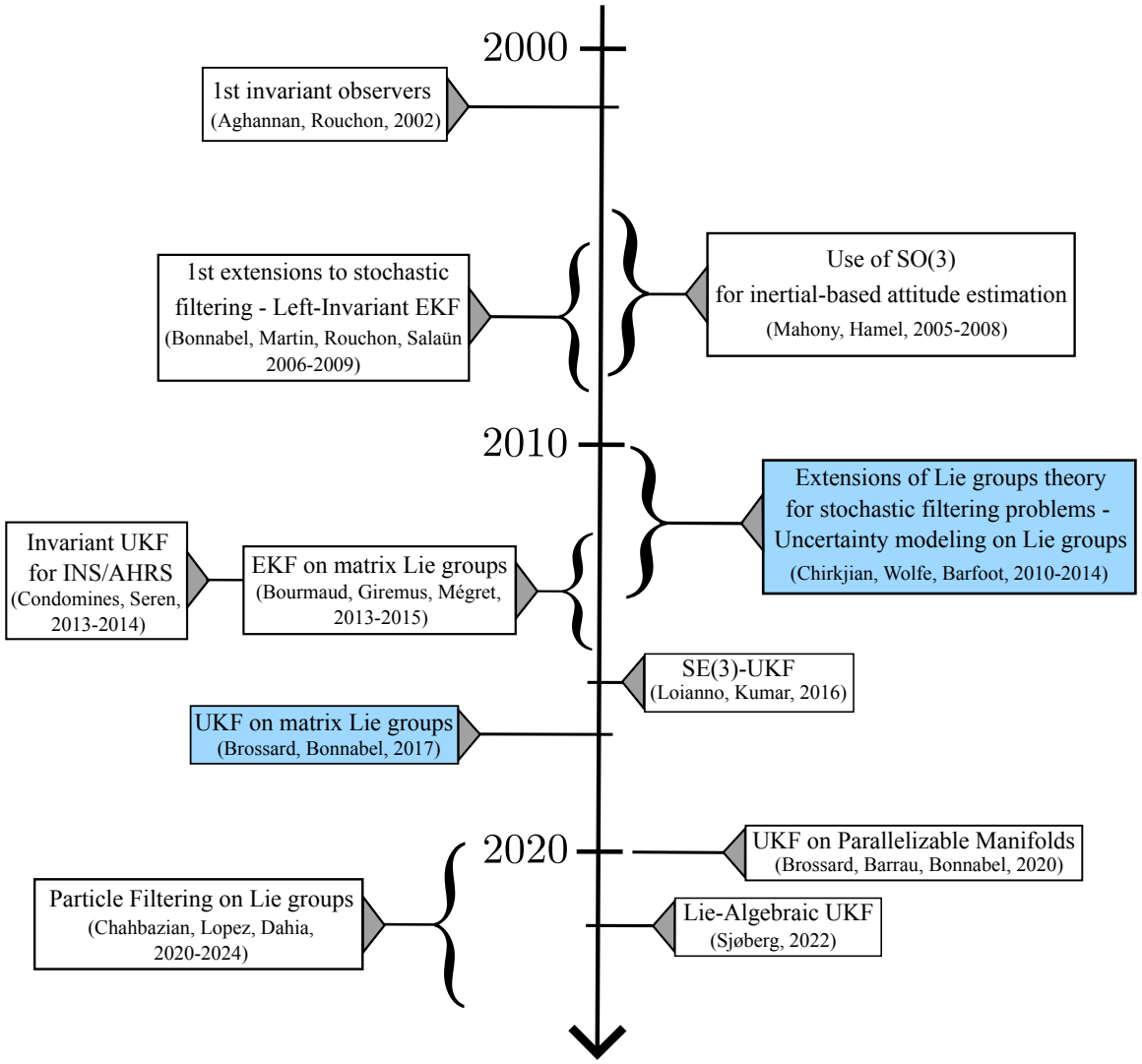


Figure 2.6: Chronological timeline of invariant filtering (from early 2000s to nowadays).

Chapter 3

The Unscented Kalman Filter on Lie groups and its assumptions

Contents

3.1 Matrix Lie groups for state estimation 44
3.1.1 Definitions and basic concepts 44
3.1.2 Kinematical modeling on a matrix Lie group 46
3.1.3 Properties of systems on Lie groups 47
3.1.4 Additional definitions 48
3.2 Uncertainty modeling and Bayesian estimation on Lie groups 50
3.2.1 Equivariant Gaussian uncertainty models 50
3.2.2 Bayesian filtering on Lie groups 52
3.3 The UKF-LG equations 54
3.3.1 System modeling for prediction and observation 54
3.3.2 State and covariance prediction 55
3.3.3 State and covariance correction 56
3.3.4 Algorithmic summary of the UKF-LG 58
3.4 Limitations of Lie-theoretic uncertainty models and alternatives 62
3.4.1 Validity of the notion of Lie-theoretic covariance 62
3.4.2 Probability distributions for attitude-only estimation 63
3.4.3 Hybrid formulations for rotation/translation coupling 65

Extensive modeling of the uncertainty on matrix Lie groups (and non-Euclidean manifolds in general) has been performed throughout the years to solve state estimation problems. In particular, the UKF-LG relies on a concentrated equivariant Gaussian assumption on a matrix Lie group $\boldsymbol{G}$ to perform mean and covariance estimation of the unknown state. This chapter provides details on this uncertainty modeling, and on the UKF-LG equations. Additionally, it presents some limitations of the equivariant Gaussian formulation, and alternative probability modeling on other manifolds.

3.1 Matrix Lie groups for state estimation

In the previous chapter, Lie groups were introduced primarily through their axiomatic definition, emphasizing the fact that an element $\boldsymbol{g} \in \boldsymbol{G}$ induces transformations under which a dynamical system remains invariant. In this section, we focus more specifically on matrix Lie groups, which naturally arise in navigation problems (leveraging the structure of $\boldsymbol{SO}(n)$ or $\boldsymbol{SE}(n)$). We therefore introduce the fundamental definitions, properties, and mathematical tools required to understand how these structures are exploited within the UKF-LG framework.

3.1.1 Definitions and basic concepts

Recall that a *manifold* is a curved, smooth (hyper)-surface, embedded in a space of higher dimension, and locally homeomorphic to the Euclidean space (Figure 3.1). More precisely, a d-dimensional manifold $\boldsymbol{\mathcal{M}}$ is a topological space with the property that each point has a neighborhood that is homeomorphic to an open subset of $\mathbb{R}^d$, where d is the number of degrees of freedom of the manifold.

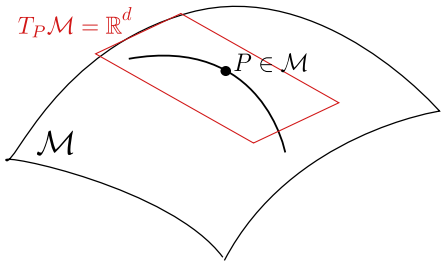


Figure 3.1: Visual representation of a manifold of dimension $\dim(\boldsymbol{\mathcal{M}}) = d$. At each point $P \in \boldsymbol{\mathcal{M}}$, one can associate (locally) a tangent space $T_P\boldsymbol{\mathcal{M}}$ that "looks like" a usual Euclidean space $\mathbb{R}^d$.

Definition 3.1 (Lie group)**.** A Lie group $\boldsymbol{G}$ is a differential (smooth) manifold, with a composition operation $\circ$, satisfying the group axioms:

- Closure under $\circ$: $\forall \boldsymbol{\chi}_a, \boldsymbol{\chi}_b \in \boldsymbol{G}, \quad \boldsymbol{\chi}_a \circ \boldsymbol{\chi}_b \in \boldsymbol{G}$
- Identity element: $\exists \boldsymbol{e} \in \boldsymbol{G}$ s.t. $\forall \boldsymbol{\chi} \in \boldsymbol{G}, \quad \boldsymbol{\chi} \circ \boldsymbol{e} = \boldsymbol{e} \circ \boldsymbol{\chi} = \boldsymbol{\chi}$
- Inverse: $\forall \boldsymbol{\chi} \in \boldsymbol{G}, \ \exists \boldsymbol{\chi}^{-1} \in \boldsymbol{G}$ s.t. $\boldsymbol{\chi} \circ \boldsymbol{\chi}^{-1} = \boldsymbol{\chi}^{-1} \circ \boldsymbol{\chi} = \boldsymbol{e}$
- Associativity: $\forall \boldsymbol{\chi}_a, \boldsymbol{\chi}_b, \boldsymbol{\chi}_c \in \boldsymbol{G}, \quad (\boldsymbol{\chi}_a \circ \boldsymbol{\chi}_b) \circ \boldsymbol{\chi}_c = \boldsymbol{\chi}_a \circ (\boldsymbol{\chi}_b \circ \boldsymbol{\chi}_c)$

The group structure imposes that the compositions of elements of the manifold remains on the manifold, which is the closure property. A group also possesses an identity element $\boldsymbol{e}$ that allows us to let unchanged an element $\boldsymbol{\chi}$ when composed with the identity (either by the left or right composition). Any element $\boldsymbol{\chi}$ of the manifold also has an inverse $\boldsymbol{\chi}^{-1}$ that brings $\boldsymbol{\chi}$ back to the identity by composition. Remark that the group composition $\circ$ can be interpreted as a group action on itself, $\circ : \boldsymbol{G} \times \boldsymbol{G} \mapsto \boldsymbol{G}$. It should also be noted that *commutativity* is not part of the group axioms, as $\boldsymbol{\chi}_a \circ \boldsymbol{\chi}_b \neq \boldsymbol{\chi}_b \circ \boldsymbol{\chi}_a$ in general. A Lie group is of dimension $\dim(\boldsymbol{G}) = d$ in the sense that it has d degrees of freedom.

Definition 3.2 (Matrix Lie groups)**.** A matrix Lie group is a Lie group whose elements are square invertible matrices $\boldsymbol{\chi} \in \mathbb{R}^{n\times n}$. In this case, the group operation $\circ$ is the usual matrix product and the identity element is the identity matrix $\boldsymbol{e} = \mathbf{I}_{n\times n}$.

Being a smooth manifold, a unique tangent space exists at every point. In particular, the tangent space at the identity holds special significance, and is known as the *Lie algebra*.

Definition 3.3 (Lie algebra)**.** The Lie algebra of a group $\boldsymbol{G}$ is defined as the tangent space $T_{\mathbf{I}}\boldsymbol{G} = \mathfrak{g}$, and it is isomorphic to $\mathbb{R}^d$. The Lie algebra is a vector space of dimension d endowed with a binary operation, called the *Lie bracket*, $[\cdot,\cdot] : \mathfrak{g} \times \mathfrak{g} \mapsto \mathfrak{g}$.

The Lie algebra can be expressed as linear combinations of some base elements E_i, where $\{E_i\}_{i=1,\dots,d}$ are called the *generators* of $\mathfrak{g}$. One can associate any vector $\boldsymbol{\xi} \in \mathbb{R}^d$ to an element of the Lie algebra by an isomorphism denoted by the "wedge" operator $[\cdot]^\wedge : \mathbb{R}^d \mapsto \mathfrak{g}$. An element $\boldsymbol{\xi}^\wedge \in \mathfrak{g}$ is then expressed as

$$\boldsymbol{\xi}^\wedge = \sum_{i=1}^{d} \xi_i E_i \tag{3.1}$$

Being an isomorphism, the inverse transformation, denoted by the "vee" operator $[\cdot]^\vee : \mathfrak{g} \mapsto \mathbb{R}^d$ writes

$$\boldsymbol{\xi}^{\wedge\vee} = \boldsymbol{\xi} = \sum_{i=1}^{d} \xi_i \mathbf{e}_i \tag{3.2}$$

where $\{\mathbf{e}_i\}_{i=1,\dots,d}$ are the base vectors in $\mathbb{R}^d$ (s.t. $\mathbf{e}_i^\wedge = E_i$). The coordinates of $\boldsymbol{\xi}$ are called *exponential coordinates*, as they are the pre-images of the mapping that produces an element $\boldsymbol{\chi} \in \boldsymbol{G}$. This mapping is the exponential map, defined below.

Definition 3.4 (Exponential and logarithmic maps)**.** The curved geometry of $\boldsymbol{G}$ can be encoded and decoded thanks to two mappings known as the group exponential $\exp : \mathfrak{g} \mapsto \boldsymbol{G}$ and group logarithm $\log : \boldsymbol{G} \mapsto \mathfrak{g}$. Their expressions, for matrix Lie groups, are obtained through power series (see [Chirikjian, 2011], section §10.2.3, p.18-19 for further details):

$$\forall \boldsymbol{\xi}^\wedge \in \mathfrak{g}, \qquad \exp(\boldsymbol{\xi}^\wedge) = \sum_{k=0}^{+\infty} \frac{(\boldsymbol{\xi}^\wedge)^k}{k!} \tag{3.3}$$

$$\forall \boldsymbol{\chi} \in \boldsymbol{G}, \qquad \log(\boldsymbol{\chi}) = \sum_{k=1}^{+\infty} (-1)^{k+1} \frac{(\boldsymbol{\chi} - \mathbf{I}_{n\times n})^k}{k} \tag{3.4}$$

It has to be noted that the exponential map is bijective only in a small region near the identity element, meaning that there is a one-to-one correspondance between an element of the group and an element of the algebra only locally[1]. Figure 3.2 illustrates the different mappings between the vector space, the algebra, and the group.

State estimation requires the need to define an error between an estimate $\hat{\boldsymbol{\chi}}$ and a reference value $\boldsymbol{\chi}$. This can be formalized in two different ways, since the group composition generally does not commute.

[1] Globally, the exponential map is surjective, meaning that the logarithm may possess more than one result.

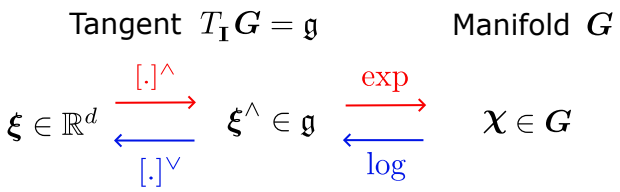


Figure 3.2: Mappings between the manifold $\boldsymbol{G}$, the Lie algebra $\mathfrak{g}$ and the vector space $\mathbb{R}^d$. $[\cdot]^\wedge$ and $[\cdot]^\vee$ are the linear invertible maps defined by (3.1) and (3.2), respectively. The mapping exp / log maps the Lie algebra to/from the manifold.

Definition 3.5 (Group errors)**.** Let a quantity $\hat{\boldsymbol{\chi}} \in \boldsymbol{G}$ fand a quantity of reference $\boldsymbol{\chi} \in \boldsymbol{G}$. The left group error $\tilde{\boldsymbol{\chi}}_{\mathsf{L}}$ and the right group error $\tilde{\boldsymbol{\chi}}_{\mathsf{R}}$, are respectively defined as

$$\tilde{\boldsymbol{\chi}}_{\mathsf{L}} = \boldsymbol{\chi}^{-1}\hat{\boldsymbol{\chi}} \tag{3.5}$$

$$\tilde{\boldsymbol{\chi}}_{\mathsf{R}} = \hat{\boldsymbol{\chi}}\boldsymbol{\chi}^{-1} \tag{3.6}$$

These two definitions ensure that the error remains in $\boldsymbol{G}$. It can be extended to the *log-Euclidean* error as follows.

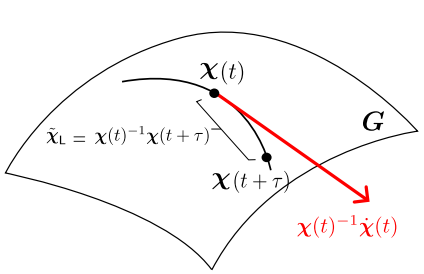
Definition 3.6 (Log-Euclidean errors)**.**

$$\tilde{\boldsymbol{\xi}}_{\mathsf{L/R}} = \log(\tilde{\boldsymbol{\chi}}_{\mathsf{L/R}})^\vee \tag{3.7}$$

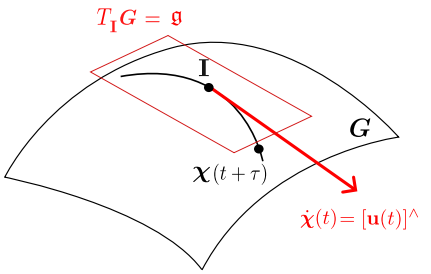
where $\tilde{\boldsymbol{\chi}}_{\mathsf{L/R}}$ is either the left of right group error defined above.

3.1.2 Kinematical modeling on a matrix Lie group

The group error allows us to derive a propagation model on the manifold $\boldsymbol{G}$. Consider an element $\boldsymbol{\chi}$ moving from a given configuration at time t to another at time $t+\tau$ (see Figure 3.3). Taking a left[2] group error $\tilde{\boldsymbol{\chi}}_{\mathsf{L}} = \boldsymbol{\chi}(t)^{-1}\boldsymbol{\chi}(t+\tau)$ allows us to write the



(a) On $\boldsymbol{G}$, we define a difference by the left group error $\tilde{\boldsymbol{\chi}}_{\mathsf{L}} = \boldsymbol{\chi}(t)^{-1}\boldsymbol{\chi}(t+\tau)$, and a tangent "vector" at $\boldsymbol{\chi}(t)$ is then given by $\boldsymbol{\chi}(t)^{-1}\dot{\boldsymbol{\chi}}(t)$.



(b) If $\boldsymbol{\chi}(t) = \mathbf{I}$, the tangent space is the Lie algebra $\mathfrak{g}$, and we can very well call it the space of instantaneous velocities, since $\dot{\boldsymbol{\chi}}(t) = [\mathbf{u}(t)]^\wedge$.

Figure 3.3: Evolution of a quantity $\boldsymbol{\chi}$ from time instant t to $t+\tau$.

time-derivative of $\boldsymbol{\chi}(t)$ as

$$\lim_{\tau\to 0} \frac{\boldsymbol{\chi}(t)^{-1}\boldsymbol{\chi}(t+\tau)}{\tau} = \boldsymbol{\chi}(t)^{-1}\frac{d}{d\tau}\boldsymbol{\chi}(t+\tau)\Big|_{\tau=0} = \boldsymbol{\chi}(t)^{-1}\dot{\boldsymbol{\chi}}(t) \tag{3.8}$$

[2]Taking the right group error instead is also possible and does not change the idea behind the construction of the propagation model.

Now, suppose that $\boldsymbol{\chi}(t) = \mathbf{I}$ (see Figure 3.3b). In this case, we have

$$\lim_{\tau \to 0} \frac{\boldsymbol{\chi}(t+\tau)}{\tau} = \frac{d}{d\tau}\boldsymbol{\chi}(t+\tau)\Big|_{\tau=0} = \dot{\boldsymbol{\chi}}(t) \in \mathfrak{g} \tag{3.9}$$

By denoting $\dot{\boldsymbol{\chi}}(t) = [\mathbf{u}(t)]^\wedge$, it is possible to interpret $\mathbf{u}$ as a velocity vector. Therefore, we can very well call the Lie algebra as the space of instantaneous velocities. Hence, from a given element $\boldsymbol{\chi}(t)$, it is possible to write a continuous propagation model taking velocity inputs $\mathbf{u}(t)$ as

$$\dot{\boldsymbol{\chi}}(t) = \boldsymbol{\chi}(t)[\mathbf{u}(t)]^\wedge \in \mathfrak{g}. \tag{3.10}$$

Discretizing this model (assuming that $\mathbf{u}$ evolves by steps and remains constant during a time interval $\tau = t_k - t_{k-1}$), we get

$$\boldsymbol{\chi}_k = \boldsymbol{\chi}_{k-1} \exp(\mathbf{u}_{k-1}^\wedge \tau) \in \boldsymbol{G} \tag{3.11}$$

For $\boldsymbol{G} = \boldsymbol{SE}(n)$, the expressions above describe rigid body motions. For $\boldsymbol{G} = \boldsymbol{SO}(n)$, these expressions describe the evolution of the attitude.

3.1.3 Properties of systems on Lie groups

It turns out that the expression (3.10) is part of a class of dynamical systems of the form

$$\dot{\boldsymbol{\chi}}(t) = \frac{d}{dt}\boldsymbol{\chi}_t = \boldsymbol{f}(\boldsymbol{\chi}_t, \mathbf{u}_t) \in \mathfrak{g}. \tag{3.12}$$

This class of systems has a striking property with respect to the left or right group error $\tilde{\boldsymbol{\chi}}_t^{\mathsf{L/R}}$. Indeed, the time-evolution of the group errors is state-trajectory independent, as dictates the following theorem (found in [Barrau and Bonnabel, 2016]):

Theorem 3.1 (Group affine property)**.** The three following conditions are equivalent for the dynamics (3.12):

i. The left group error $\tilde{\boldsymbol{\chi}}_t^{\mathsf{L}}$ is state-trajectory independent.

ii. The right group error $\tilde{\boldsymbol{\chi}}_t^{\mathsf{R}}$ is state-trajectory independent.

iii. $\forall t > 0$ and $\boldsymbol{a}, \boldsymbol{b} \in \boldsymbol{G}$, we have (in the tangent space at $\boldsymbol{ab}$):

$$\boldsymbol{f}(\boldsymbol{ab}, \mathbf{u}_t) = \boldsymbol{f}(\boldsymbol{a}, \mathbf{u}_t)\boldsymbol{b} + \boldsymbol{a}\boldsymbol{f}(\boldsymbol{b}, \mathbf{u}_t) - \boldsymbol{a}\boldsymbol{f}(\mathbf{I}_{n\times n}, \mathbf{u}_t)\boldsymbol{b}. \tag{3.13}$$

If one of these conditions is satisfied, we have

$$\begin{aligned} \frac{d}{dt}\tilde{\boldsymbol{\chi}}_t^{\mathsf{L}} &= \mathbf{g}_{\mathsf{L}}(\tilde{\boldsymbol{\chi}}_t^{\mathsf{L}}, \mathbf{u}_t) \quad \text{where} \quad \mathbf{g}_{\mathsf{L}}(\tilde{\boldsymbol{\chi}}, \mathbf{u}_t) = \boldsymbol{f}(\tilde{\boldsymbol{\chi}}, \mathbf{u}_t) - \boldsymbol{f}(\mathbf{I}_{n\times n}, \mathbf{u}_t)\tilde{\boldsymbol{\chi}} \\ \frac{d}{dt}\tilde{\boldsymbol{\chi}}_t^{\mathsf{R}} &= \mathbf{g}_{\mathsf{R}}(\tilde{\boldsymbol{\chi}}_t^{\mathsf{R}}, \mathbf{u}_t) \quad \text{where} \quad \mathbf{g}_{\mathsf{R}}(\tilde{\boldsymbol{\chi}}, \mathbf{u}_t) = \boldsymbol{f}(\tilde{\boldsymbol{\chi}}, \mathbf{u}_t) - \tilde{\boldsymbol{\chi}}\boldsymbol{f}(\mathbf{I}_{n\times n}, \mathbf{u}_t) \end{aligned} \tag{3.14}$$

Moreover, the system (3.12) with property (3.1) implies the next property, generalizing that of linear systems.

Theorem 3.2 (Log-linear property of the error propagation)**.** Consider the left/right group error $\tilde{\boldsymbol{\chi}}_t^{\mathsf{i}}$ between two arbitrarily trajectories of the system (3.12), and satisfying Theorem (3.1). The superscript i denotes indifferently L or R. Let $[\cdot]^\wedge$ and

$\exp(\cdot)$ be defined as in (3.1) and (3.3), respectively. Let $\tilde{\boldsymbol{\xi}}^{\mathsf{i}}_t \in \mathbb{R}^d$ be the log-Euclidean error such that we initially have $\exp\left([\tilde{\boldsymbol{\xi}}^{\mathsf{i}}_0]^\wedge\right) = \tilde{\boldsymbol{\chi}}^{\mathsf{i}}_0$. Let $\mathbf{A}^{\mathsf{i}}_{\mathbf{u}_t}$ be defined by $\mathbf{g}\left(\exp([\tilde{\boldsymbol{\xi}}]^\wedge), \mathbf{u}_t\right) = [\mathbf{A}_{\mathbf{u}_t}\tilde{\boldsymbol{\xi}}]^\wedge + O(\|\boldsymbol{\xi}\|^2)$. If $\tilde{\boldsymbol{\xi}}^{\mathsf{i}}_t$ is defined, for $t > 0$, by the linear differential equation in $\mathbb{R}^d$

$$\frac{d}{dt}\tilde{\boldsymbol{\xi}}^{\mathsf{i}}_t = \mathbf{A}^{\mathsf{i}}_{\mathbf{u}_t}\tilde{\boldsymbol{\xi}}^{\mathsf{i}}_t, \tag{3.15}$$

then, we have for the true non-linear error $\tilde{\boldsymbol{\chi}}^{\mathsf{i}}_t$, the correspondance at all times

$$\forall t \geq 0, \quad \tilde{\boldsymbol{\chi}}^{\mathsf{i}}_t = \exp\left([\tilde{\boldsymbol{\xi}}^{\mathsf{i}}_t]^\wedge\right) \tag{3.16}$$

The above theorem is fundamental in invariant filtering on Lie groups, as it allows us to completely encode the dynamics of the group error $\frac{d}{dt}\tilde{\boldsymbol{\chi}}^{\mathsf{i}}_t = \mathbf{g}_{\mathsf{i}}(\tilde{\boldsymbol{\chi}}^{\mathsf{i}}_t, \mathbf{u}_t)$ in the dynamics of the log-Euclidean error $\frac{d}{dt}\tilde{\boldsymbol{\xi}}^{\mathsf{i}}_t = \mathbf{A}^{\mathsf{i}}_{\mathbf{u}_t}\tilde{\boldsymbol{\xi}}^{\mathsf{i}}_t$, through the mere knowledge of the matrix $\mathbf{A}^{\mathsf{i}}_{\mathbf{u}_t}$ which only depends on the input $\mathbf{u}_t$. Note that these properties have been formalized in continuous-time, but discrete-time formulations exists [Barrau and Bonnabel, 2018].

3.1.4 Additional definitions

Although there exists a unique Lie algebra, its definition as the tangent space at the identity and the non-commutativity of the group operation imply that it can be built by sending back an element of the group to the identity using either the left or right action. This remark introduces the notion of Adjoint operator. Actually, two formulations of (3.10) are possible depending on the side of multiplication we have chosen for the group error:

Body velocity	Spatial velocity
$\dot{\boldsymbol{\chi}} = \boldsymbol{\chi}\mathbf{u}^\wedge_{\mathsf{L}} \quad \Leftrightarrow \quad \boldsymbol{\chi}^{-1}\dot{\boldsymbol{\chi}} = \mathbf{u}^\wedge_{\mathsf{L}}$	$\dot{\boldsymbol{\chi}} = \mathbf{u}^\wedge_{\mathsf{R}}\boldsymbol{\chi} \quad \Leftrightarrow \quad \dot{\boldsymbol{\chi}}\boldsymbol{\chi}^{-1} = \mathbf{u}^\wedge_{\mathsf{R}}$

(3.17)

These operations are two different ways to go back to the Lie algebra. They are not incompatible, they are just two different point-of-views of the same space. To be more precise, the quantity $\mathbf{u}^\wedge_{\mathsf{L}}$ is an element of the algebra as seen from the body-fixed frame (as if we had place ourselves from the point-of-view of the moving-body, and watch the world move). It is also called *Body velocity*. In parallel, the quantity $\mathbf{u}^\wedge_{\mathsf{R}}$ is an element of the algebra as seen from the inertial frame (as if we had place ourselves from the point-of-view of a fixed-reference frame, and watch the body move). It is also called *Spatial velocity*. The Adjoint property allows to "link" the algebras, and ensures that the tangent space has the same structure at all points on the manifold.

Definition 3.7 (Group Adjoint)**.** The Adjoint operator $\mathrm{Ad} : \boldsymbol{G} \mapsto \boldsymbol{GL}(n)$ (where $\boldsymbol{GL}(n)$ is the group of all invertible real matrices) is defined through the following property

$$\left(\mathrm{Ad}(\boldsymbol{\chi})\mathbf{a}\right)^\wedge = \boldsymbol{\chi}\mathbf{a}^\wedge\boldsymbol{\chi}^{-1}, \quad \forall \mathbf{a} \in \mathbb{R}^{\dim(\boldsymbol{G})} \tag{3.18}$$

Additionally, it provides a pseudo-commutator, thus allowing to convert Body velocity into Spatial velocity and vice-versa:

$$\dot{\boldsymbol{\chi}} = \boldsymbol{\chi}\mathbf{u}^\wedge_{\mathsf{L}} = \boldsymbol{\chi}\mathbf{u}^\wedge_{\mathsf{L}}\boldsymbol{\chi}^{-1}\boldsymbol{\chi} = \underbrace{\left(\mathrm{Ad}(\boldsymbol{\chi})\mathbf{u}_{\mathsf{L}}\right)^\wedge}_{\mathbf{u}^\wedge_{\mathsf{R}}}\boldsymbol{\chi} \tag{3.19}$$

An equivalent formulation can be defined in the Lie algebra, providing the same feature, and known as the *small adjoint*, denoted $\text{ad} : \mathfrak{g} \mapsto \mathbb{R}^{n\times n}$.

Definition 3.8 ((SMALL) ADJOINT)**.** Let $\mathfrak{a}, \mathfrak{b} \in \mathfrak{g}$. The small adjoint is defined to satisfy the property

$$\text{ad}(\mathfrak{a})\mathfrak{b} = \frac{d}{dt}\Big(\text{Ad}(e^{t\mathfrak{a}})\mathfrak{b}\Big)\Big|_{t=0} = \Big[\mathfrak{a}, \mathfrak{b}\Big] = \mathfrak{a}\mathfrak{b} - \mathfrak{b}\mathfrak{a} \tag{3.20}$$

One of the main tools used to manipulate Lie group elements is the Baker-Campbell-Hausdorff formula. In its simplest form, it shows how quantities behave in the Lie algebra when performing a matrix composition of the form $\boldsymbol{\chi}_a\boldsymbol{\chi}_b$.

Definition 3.9 (BAKER-CAMPBELL-HAUSDORFF FORMULA)**.** Let $\boldsymbol{\chi}_a = \exp(\boldsymbol{\xi}_a^\wedge)$ and $\boldsymbol{\chi}_b = \exp(\boldsymbol{\xi}_b^\wedge)$. The Baker-Campbell-Hausdorff (BCH) formula is expressed as

$$\begin{aligned}\log(\boldsymbol{\chi}_a\boldsymbol{\chi}_b) = \log\big(\exp(\boldsymbol{\xi}_a^\wedge)\exp(\boldsymbol{\xi}_b^\wedge)\big) = \boldsymbol{\xi}_a^\wedge + \boldsymbol{\xi}_b^\wedge + \frac{1}{2}\Big[\boldsymbol{\xi}_a^\wedge, \boldsymbol{\xi}_b^\wedge\Big] \\ + \frac{1}{12}\Big[\boldsymbol{\xi}_a^\wedge, \big[\boldsymbol{\xi}_a^\wedge, \boldsymbol{\xi}_b^\wedge\big]\Big] + \frac{1}{12}\Big[\boldsymbol{\xi}_b^\wedge, \big[\boldsymbol{\xi}_b^\wedge, \boldsymbol{\xi}_a^\wedge\big]\Big] - \frac{1}{24}\Big[\boldsymbol{\xi}_b^\wedge, \big[\boldsymbol{\xi}_a^\wedge, \big[\boldsymbol{\xi}_a^\wedge, \boldsymbol{\xi}_b^\wedge\big]\big]\Big] + \dots\end{aligned} \tag{3.21}$$

$\forall \boldsymbol{\xi}_a^\wedge, \boldsymbol{\xi}_b^\wedge \in \mathfrak{g}$ and where $\big[\cdot,\cdot\big]$ denotes the Lie brackets, defined such that $\big[\boldsymbol{\xi}_a^\wedge, \boldsymbol{\xi}_b^\wedge\big] = \boldsymbol{\xi}_a^\wedge\boldsymbol{\xi}_b^\wedge - \boldsymbol{\xi}_b^\wedge\boldsymbol{\xi}_a^\wedge$.

Another formulation exists by exploiting the $\vee$ operator and the equality $\Big[\boldsymbol{\xi}_a^\wedge, \boldsymbol{\xi}_b^\wedge\Big]^\vee = \text{ad}(\boldsymbol{\xi}_a^\wedge)\boldsymbol{\xi}_b$:

$$\begin{aligned}\log(\boldsymbol{\chi}_a\boldsymbol{\chi}_b)^\vee = \log\big(\exp(\boldsymbol{\xi}_a^\wedge)\exp(\boldsymbol{\xi}_b^\wedge)\big)^\vee = \boldsymbol{\xi}_a + \boldsymbol{\xi}_b + \frac{1}{2}\text{ad}(\boldsymbol{\xi}_a^\wedge)\boldsymbol{\xi}_b \\ + \frac{1}{12}\text{ad}(\boldsymbol{x}^\wedge)\text{ad}(\boldsymbol{\xi}_a^\wedge)\boldsymbol{\xi}_b + \frac{1}{12}\text{ad}(\boldsymbol{\xi}_b^\wedge)\text{ad}(\boldsymbol{\xi}_b^\wedge)\boldsymbol{\xi}_a - \frac{1}{24}\text{ad}(\boldsymbol{\xi}_b^\wedge)\text{ad}(\boldsymbol{\xi}_a^\wedge)\text{ad}(\boldsymbol{\xi}_a^\wedge)\boldsymbol{\xi}_b + \dots\end{aligned} \tag{3.22}$$

The BCH formula can be used in numerous ways, such as formalizing invariant propagation models directly expressed in the Lie algebra, or providing useful expressions in closed-form depending on approximations. For different assumptions on the quantities $||\boldsymbol{\xi}_a||$ and $||\boldsymbol{\xi}_b||$, different levels of truncation are chosen. In particular, assuming that both $\boldsymbol{\xi}_a$ and $\boldsymbol{\xi}_b$ are small, the following first-order approximation is sufficient:

$$\log(\boldsymbol{\chi}_a\boldsymbol{\chi}_b)^\vee = \log\big(\exp(\boldsymbol{\xi}_a^\wedge)\exp(\boldsymbol{\xi}_b^\wedge)\big)^\vee \approx \boldsymbol{\xi}_a + \boldsymbol{\xi}_b \tag{3.23}$$

If one of them is small with respect to the other, the approximation brings out the notion of group jacobian. Hence, we can define two jacobians (cf. Appendix B.1 for further details about their demonstration).

Definition 3.10 (GROUP JACOBIAN)**.** $\forall \boldsymbol{\xi} \in \mathfrak{g}$, the left jacobian is defined as

$$\boldsymbol{\mathcal{J}}_\mathsf{L}\big(\text{ad}(\boldsymbol{\xi}^\wedge)\big) = \sum_{k=0}^{+\infty}\frac{1}{(k+1)!}\big(\text{ad}(\boldsymbol{\xi}^\wedge)\big)^k \tag{3.24}$$

and the right jacobian can be computed from the left jacobian as

$$\boldsymbol{\mathcal{J}}_\mathsf{R}\big(\text{ad}(\boldsymbol{\xi}^\wedge)\big) = \boldsymbol{\mathcal{J}}_\mathsf{L}\big(-\text{ad}(\boldsymbol{\xi}^\wedge)\big) \tag{3.25}$$

All of the previous definitions have specific expressions depending on the considered Lie group. Some of these expressions are provided in Appendix B.2 for the group of interests of the thesis ($\boldsymbol{SO}(n)$, $\boldsymbol{SE}(n)$ and $\boldsymbol{SE}_k(n)$).

3.2 Uncertainty modeling and Bayesian estimation on Lie groups

Solving filtering problems require the need to characterize a random variable in terms of probability. When the considered state lives in a Lie group $\boldsymbol{G}$, it can generally be defined by a function $\mathcal{P}(\boldsymbol{\chi})$ satisfying the following requirements:

$$\begin{cases} \mathcal{P}(\boldsymbol{\chi}) > 0 \quad \forall \boldsymbol{\chi} \in \boldsymbol{G} \\ \displaystyle\int_{\boldsymbol{G}} \mathcal{P}(\boldsymbol{\chi}) d\boldsymbol{\chi} = 1 \end{cases} \tag{3.26}$$

One of the main difficulties in defining a probability distribution on a manifold lies in giving meaningful interpretations to its statistical characteristics, particularly its moments. The standard operations used in Euclidean statistics cannot be directly generalized to manifolds. They must therefore be reformulated in accordance with the composition laws and intrinsic distances of the manifold under consideration.

3.2.1 Equivariant Gaussian uncertainty models

When the manifold is a matrix Lie group $\boldsymbol{G}$, one would ideally prefer a distribution that can be applied to any $\boldsymbol{G}$, regardless of their own structure. From this idea, the (concentrated) equivariant Gaussian distribution [Barfoot and Furgale, 2014] has emerged as a powerful formulation that enables us to use the standard Gaussian while respecting the underlying group operations. This density is defined such that it satisfies

$$\int_{\boldsymbol{G}} \mathcal{N}_{\mathsf{L/R}}(\boldsymbol{\chi}; \hat{\boldsymbol{\chi}}, \mathbf{P}) d\boldsymbol{\chi} = \int_{\mathbb{R}^d} \mathcal{N}(\boldsymbol{\xi}_{\mathsf{L/R}}; \mathbf{0}, \mathbf{P}) |\det(\boldsymbol{\mathcal{J}})| d\boldsymbol{\xi} = 1 \tag{3.27}$$

where $\boldsymbol{\mathcal{J}}$ is either the left or right group jacobian[3] $\boldsymbol{\mathcal{J}}_{\mathsf{L/R}}$. The quantity $d\boldsymbol{\chi} = |\det(\boldsymbol{\mathcal{J}})| d\boldsymbol{\xi}$ is called the *Haar* measure. It preserves invariance properties by right or left multiplication, and generalizes the usual *Lebesgue* measure, which is invariant by translation. The element $\hat{\boldsymbol{\chi}}$ is the mean of $\boldsymbol{\chi}$, and $\mathbf{P}$ its covariance. Given the left error $\tilde{\boldsymbol{\chi}}_{\mathsf{L}} = \hat{\boldsymbol{\chi}}^{-1}\boldsymbol{\chi}$ (or right error $\tilde{\boldsymbol{\chi}}_{\mathsf{R}} = \boldsymbol{\chi}\hat{\boldsymbol{\chi}}^{-1}$), the pdf $\mathcal{N}_{\mathsf{L/R}}(\boldsymbol{\chi}; \hat{\boldsymbol{\chi}}, \mathbf{P})$ then writes

$$\mathcal{N}_{\mathsf{L/R}}(\boldsymbol{\chi}; \hat{\boldsymbol{\chi}}, \mathbf{P}) = \frac{C(\mathbf{P})}{|\det(\boldsymbol{\mathcal{J}})|} \exp\left(-\frac{1}{2} \log(\tilde{\boldsymbol{\chi}}_{\mathsf{L/R}})^{\vee\top} \mathbf{P}^{-1} \log(\tilde{\boldsymbol{\chi}}_{\mathsf{L/R}})^{\vee} \right) \tag{3.28}$$

where $C(\mathbf{P}) = \frac{1}{\sqrt{(2\pi)^d \det(\mathbf{P})}}$ is the usual normalization constant of the Gaussian pdf. The equivariant Gaussian pdf only makes sense when the covariance $\mathbf{P}$ is small enough to consider the distribution as "concentrated".

The concept of "concentrated distributions" (not necessarily Gaussians to begin with) on Lie groups was introduced in [Wang and Chirikjian, 2008], and further detailed in [Chirikjian, 2011] (Section §20.6). They defined the notions of *Lie-theoretic* mean and

[3] In fact, it does not matter since $|\det(\boldsymbol{\mathcal{J}}_{\mathsf{L}})| = |\det(\boldsymbol{\mathcal{J}}_{\mathsf{R}})|$.

covariance (valid under some conditions): Let $\mathcal{P}(\boldsymbol{\chi})$ be any pdf for the random variable $\boldsymbol{\chi}$. If a unique value $\hat{\boldsymbol{\chi}} \in \boldsymbol{G}$ exists for which we have

$$\int_{\boldsymbol{G}} \log(\hat{\boldsymbol{\chi}}^{-1}\boldsymbol{\chi})^{\vee}\mathcal{P}(\boldsymbol{\chi})d\boldsymbol{\chi} = \int_{\mathbb{R}^d} \boldsymbol{\xi}\,\mathcal{P}(\boldsymbol{\xi})|\det(\boldsymbol{\mathcal{J}})|d\boldsymbol{\xi} = \mathbf{0} \tag{3.29}$$

then $\hat{\boldsymbol{\chi}}$ will be called the mean of the distribution $\mathcal{P}(\boldsymbol{\chi})$. It can be seen as a left[4] definition of the mean. To be more precise, the mean is found by minimizing a cost function of the form

$$J(\boldsymbol{g}) = \int_{\boldsymbol{G}} ||\log(\boldsymbol{g}^{-1}\boldsymbol{\chi})||^2\mathcal{P}(\boldsymbol{\chi})d\boldsymbol{\chi} \tag{3.30}$$

and it does not necessarily return a value $\boldsymbol{g}_{min}$ that is equal to $\hat{\boldsymbol{\chi}}$ (for a general manifold), because it may depend on the topology of the manifold, or the form of the distribution. However, if $\mathcal{P}(\boldsymbol{\chi})$ is unimodal and very concentrated, then we will get $\boldsymbol{g}_{min} \approx \hat{\boldsymbol{\chi}}$. It can be shown that the mean of $\boldsymbol{\chi}$ in (3.28) satisfies the condition of the Lie-theoretic mean (3.29).

Furthermore, if a distribution is highly concentrated about its mean $\hat{\boldsymbol{\chi}}$, the covariance is well-defined and expressed as

$$\mathbf{P} = \int_{\boldsymbol{G}} \log(\hat{\boldsymbol{\chi}}^{-1}\boldsymbol{\chi})^{\vee}[\log(\hat{\boldsymbol{\chi}}^{-1}\boldsymbol{\chi})^{\vee}]^{\top}\mathcal{P}(\boldsymbol{\chi})d\boldsymbol{\chi} = \int_{\mathbb{R}^d} \boldsymbol{\xi}\boldsymbol{\xi}^{\top}\mathcal{P}(\boldsymbol{\xi})\,|\det(\boldsymbol{\mathcal{J}})|d\boldsymbol{\xi} \tag{3.31}$$

Similarly, the covariance $\mathbf{P}$ defined in (3.28) matches the definition of $\mathbf{P}$ in (3.31). Relying on a normal distribution makes easier the derivation of closed-form (and analytically tractable) formulas for quantifying the uncertainty propagation of a moving rigid body, as can be seen in [Long et al., 2013], or [Barfoot and Furgale, 2014].

Finally, the concentrated equivariant Gaussian can be compactly denoted as follows: Let $\boldsymbol{\chi}$ be a random variable in a unimodular matrix Lie group $\boldsymbol{G}$. Assume that $\hat{\boldsymbol{\chi}}$ denotes the mean of the random variable $\boldsymbol{\chi}$. The dispersion of $\boldsymbol{\chi}$ can be defined as follows

$$\begin{aligned} \boldsymbol{\chi} &= \begin{cases} \mathsf{L}_{\hat{\boldsymbol{\chi}}}(\exp(\boldsymbol{\xi}^{\wedge})) = \hat{\boldsymbol{\chi}}\exp(\boldsymbol{\xi}^{\wedge}) \sim \mathcal{N}_{\mathsf{L}}(\hat{\boldsymbol{\chi}}, \mathbf{P}) & \text{(Left equivariant)} \\ \mathsf{R}_{\hat{\boldsymbol{\chi}}}(\exp(\boldsymbol{\xi}^{\wedge})) = \exp(\boldsymbol{\xi}^{\wedge})\hat{\boldsymbol{\chi}} \sim \mathcal{N}_{\mathsf{R}}(\hat{\boldsymbol{\chi}}, \mathbf{P}) & \text{(Right equivariant)} \end{cases} \\ \boldsymbol{\xi} &\sim \mathcal{N}(\mathbf{0}, \mathbf{P}) \end{aligned} \tag{3.32}$$

where $\boldsymbol{\xi} \in \mathbb{R}^d$. The matrix $\mathbf{P}$ is a covariance matrix of dimension $d \times d$. This formulation of uncertainty by means of a Gaussian distribution in exponential coordinates signifies that the vector $\boldsymbol{\xi}$ contains all the uncertainty of the random variable $\boldsymbol{\chi}$. It is then translated into the group via the exponential map to yield a distribution with mean located at the identity.

Remark 3.1. Defining the uncertainty on the Lie algebra by means of a random vector $\boldsymbol{\xi}$ and mapping it into the group as $\exp(\boldsymbol{\xi}^{\wedge})$ is permitted by the log-linear Theorem 3.2, and the fact that we always have the correspondance between a group error $\tilde{\boldsymbol{\chi}}$ and a log-Euclidean error $\tilde{\boldsymbol{\xi}}$, as dictated by (3.16).

[4] Again, a right definition is also valid by choosing $\log(\boldsymbol{\chi}\hat{\boldsymbol{\chi}}^{-1})$ instead, which was the case in [Barfoot and Furgale, 2014].

Finally, this density is, in turn, translated to the mean value $\hat{\boldsymbol{\chi}}$ by left action $\mathsf{L}_{\hat{\boldsymbol{\chi}}}(\cdot)$ or right action $\mathsf{R}_{\hat{\boldsymbol{\chi}}}(\cdot)$, leading to two different probability densities, $\mathcal{N}_{\mathsf{L}}(\hat{\boldsymbol{\chi}}, \mathbf{P})$ and $\mathcal{N}_{\mathsf{R}}(\hat{\boldsymbol{\chi}}, \mathbf{P})$. Figure 3.4 illustrates the operation henceforth performed.

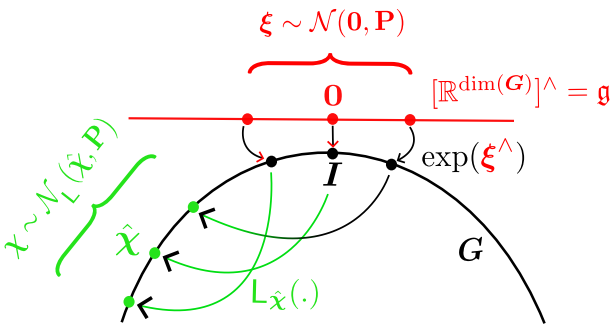


Figure 3.4: Left-equivariant Gaussian uncertainty for a random variable $\boldsymbol{\chi} \in \boldsymbol{G}$.

The key understanding behind this uncertainty characterization is that the dispersion of variables is expressed in the vector space $\mathbb{R}^d$, thanks to the logarithmic map into $\mathfrak{g}$ and the isomorphism $[\cdot]^\vee$. This parameterization using exponential coordinates enables us to capture the distribution of $\boldsymbol{\chi}$ classically in the vector space $\mathbb{R}^d$, thus relying solely on the mean and covariance defined by (3.29) and (3.31). As a result, we can work directly within the vector space at the identity and apply standard notions and definitions, while preserving the geometrical advantages of Lie groups via the left or right action.

3.2.2 Bayesian filtering on Lie groups

When the state $\boldsymbol{\chi}$ lies in $\boldsymbol{G}$, and provided that we have vector outputs $\mathbf{y}$, the recursive Bayesian estimator still holds: Given a known initial conditional probability

$$\mathcal{P}(\boldsymbol{\chi}_{k-1}|\mathbf{y}_{0:k-1}), \tag{3.33}$$

Prediction: Propagate the state density

$$\mathcal{P}(\boldsymbol{\chi}_k|\mathbf{y}_{0:k-1}) = \int_{\boldsymbol{G}} \mathcal{P}(\boldsymbol{\chi}_k|\boldsymbol{\chi}_{k-1})\mathcal{P}(\boldsymbol{\chi}_{k-1}|\mathbf{y}_{0:k-1})d\boldsymbol{\chi}_{k-1}. \tag{3.34}$$

Correction: When an observation $\mathbf{y}_k$ is available, compute the update

$$\mathcal{P}(\boldsymbol{\chi}_k|\mathbf{y}_{0:k}) \propto \mathcal{P}(\mathbf{y}_k|\boldsymbol{\chi}_k)\mathcal{P}(\boldsymbol{\chi}_k|\mathbf{y}_{0:k-1}). \tag{3.35}$$

Similarly to its Euclidean counterpart, finding a suboptimal approximation of the Bayesian filter remains a challenge, especially for the case of a state variable living in non Euclidean manifolds such as Lie groups. However, the solution can still be approximated by relying on a concentrated equivariant Gaussian distribution. At this stage, the definitions of mean and covariance provided in (3.29) and (3.31) become relevant, as the problem of identifying the probability density $\mathcal{P}(\boldsymbol{\chi}_k|\mathbf{y}_{0:k})$ reduces to the more tractable task of determining its associated mean $\hat{\boldsymbol{\chi}}_{k|k}$ and covariance $\mathbf{P}_{k|k}$.

From this approach, the probability densities involved in the recursive scheme are all assumed to be equivariant Gaussians (or simply Gaussian for the observation $\mathbf{y}_k$). In particular, the initial conditional density (3.33) is replaced by

$$\mathcal{P}(\boldsymbol{\chi}_{k-1}|\mathbf{y}_{0:k-1}) \to \mathcal{N}_{\mathsf{L}}(\hat{\boldsymbol{\chi}}_{k-1|k-1}, \mathbf{P}_{k-1|k-1}).$$

The prior density (3.34), obtained from the prediction, is replaced by

$$\mathcal{P}(\boldsymbol{\chi}_{k}|\mathbf{y}_{0:k-1}) \to \mathcal{N}_{\mathsf{L}}(\hat{\boldsymbol{\chi}}_{k|k-1}, \mathbf{P}_{k|k-1}).$$

The posterior density (3.35), obtained from the correction, is replaced by

$$\mathcal{P}(\boldsymbol{\chi}_{k}|\mathbf{y}_{0:k}) \to \mathcal{N}_{\mathsf{L}}(\hat{\boldsymbol{\chi}}_{k|k}, \mathbf{P}_{k|k}).$$

Hence, given known initial moments $\hat{\boldsymbol{\chi}}_{k-1|k-1}$ and $\mathbf{P}_{k-1|k-1}$, the propagation of the density (prediction step) boils down to computing the prior mean $\hat{\boldsymbol{\chi}}_{k|k-1}$, defined as

$$\hat{\boldsymbol{\chi}}_{k|k-1} := \int_{\boldsymbol{G}} \boldsymbol{f}(\boldsymbol{\chi}_{k-1}) \mathcal{N}_{\mathsf{L}}(\hat{\boldsymbol{\chi}}_{k-1|k-1}, \mathbf{P}_{k-1|k-1}) d\boldsymbol{\chi}_{k-1}, \tag{3.36}$$

and computing the prior covariance $\mathbf{P}_{k|k-1}$, defined as

$$\mathbf{P}_{k|k-1} := \int_{\boldsymbol{G}} \log(\hat{\boldsymbol{\chi}}_{k|k-1}^{-1}\boldsymbol{\chi}_k)^{\vee} \log(\hat{\boldsymbol{\chi}}_{k|k-1}^{-1}\boldsymbol{\chi}_k)^{\vee\top} \mathcal{N}_{\mathsf{L}}(\hat{\boldsymbol{\chi}}_{k-1|k-1}, \mathbf{P}_{k-1|k-1}) d\boldsymbol{\chi}_{k-1}. \tag{3.37}$$

The mean $\hat{\boldsymbol{\chi}}_{k|k-1}$ must be computed such that it also satisfies an equivariant Gaussian, namely $\boldsymbol{\chi}_k = \hat{\boldsymbol{\chi}}_{k|k-1} \exp(\boldsymbol{\xi}_{k|k-1}^{\wedge})$, with $\boldsymbol{\xi}_{k|k-1} \sim \mathcal{N}(\mathbf{0}, \mathbf{P}_{k|k-1})$. Back to the Lie algebra, this implies that the mean of the propagated random vector $\boldsymbol{\xi}_{k|k-1}$ must be zero, in order to consider $\hat{\boldsymbol{\chi}}_{k|k-1}$ as the mean of $\boldsymbol{\chi}_k$. In mathematical terms, the following condition must hold:

$$\mathbb{E}(\boldsymbol{\xi}_{k|k-1}) := \int_{\boldsymbol{G}} \log(\hat{\boldsymbol{\chi}}_{k|k-1}^{-1}\boldsymbol{\chi}_k)^{\vee} \mathcal{N}_{\mathsf{L}}(\hat{\boldsymbol{\chi}}_{k|k-1}, \mathbf{P}_{k|k-1}) d\boldsymbol{\chi}_k = \mathbf{0}. \tag{3.38}$$

If a measurement $\mathbf{y}_k$ is available, the prior mean $\hat{\boldsymbol{\chi}}_{k|k-1}$ will be updated to yield an posterior mean $\hat{\boldsymbol{\chi}}_{k|k}$, given by an update of the form

$$\hat{\boldsymbol{\chi}}_{k|k} = \hat{\boldsymbol{\chi}}_{k|k-1} \exp\left(\Delta(\mathbf{y}_k, \hat{\boldsymbol{\chi}}_{k|k-1})^{\wedge}\right). \tag{3.39}$$

Within this framework, the correction term is thus computed in the Lie algebra, and mapped to the group via the exponential map. This operation preserves the group symmetries, and the challenge lies in finding the said correction term. The prior covariance $\mathbf{P}_{k|k-1}$ will also be subject to an update to yield a posterior covariance $\mathbf{P}_{k|k}$.

The Unscented Kalman Filter on Lie Groups (UKF-LG) of [Brossard et al., 2017] adopts this equivariant Gaussian modeling and this estimation scheme to recursively infer the value of $\boldsymbol{\chi}_k$ from an observation $\mathbf{y}_k$. The next section explicitly details its equations.

3.3 The UKF-LG equations

In the following, we adopt the left-equivariant Gaussian ($\mathcal{N}_\mathsf{L}$) in (3.32) to make explanations more accessible, but it should be noted that all subsequent derivations and reasonings remain valid for the right definition ($\mathcal{N}_\mathsf{R}$). Hence, the dispersion of the random variable $\boldsymbol{\chi}_k$ is modeled as an equivariant Gaussian distribution $\boldsymbol{\chi}_k \sim \mathcal{N}_\mathsf{L}(\hat{\boldsymbol{\chi}}_{k|l}, \mathbf{P}_{k|l})$:

$$\boldsymbol{\chi}_k = \hat{\boldsymbol{\chi}}_{k|l} \exp(\boldsymbol{\xi}_{k|l}^\wedge) \quad \text{with} \quad \boldsymbol{\xi}_{k|l} \sim \mathcal{N}(\mathbf{0}, \mathbf{P}_{k|l}) \tag{3.40}$$

and l taking the index k or $k-1$.

3.3.1 System modeling for prediction and observation

For the prediction step of the UKF-LG, a propagation model $\boldsymbol{f}$ must be defined on the group. Recall from subsection §3.1.2 that a discrete kinematical model is given by

$$\boldsymbol{\chi}_k = \boldsymbol{\chi}_{k-1} \exp(\mathbf{u}_{k-1}^\wedge \tau) \qquad \forall k = 1, \dots$$

where τ is a timestep, and $\mathbf{u}_{k-1}$ is a velocity input (where components are expressed in the Lie algebra). If $\mathbf{u}_{k-1}$ is exactly known without any error, then this kinematical model exactly describes rigid body motions (provided that τ is small enough). However, since $\mathbf{u}_{k-1}$ is not perfectly known (because it is obtained from noisy inertial sensors), a stochastic kinematical model emerges: $\exp(\mathbf{u}^\wedge \tau)$ belongs to $\boldsymbol{G}$ the same way $\boldsymbol{\chi}$ belongs to $\boldsymbol{G}$, then it inherits the same uncertainty characterization as the state. Therefore, the stochastic kinematical model writes

$$\begin{aligned} &\boldsymbol{\chi}_k = \boldsymbol{f}(\boldsymbol{\chi}_{k-1}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1}) = \boldsymbol{\chi}_{k-1} \boldsymbol{U}_{k-1} \\ &\text{with,} \;\; \boldsymbol{U}_{k-1} = \exp(\mathbf{u}_{k-1}^\wedge \tau) \exp(\mathbf{w}_{k-1}^\wedge \tau) \end{aligned} \tag{3.41}$$

where $\boldsymbol{U}_{k-1} \sim \mathcal{N}_\mathsf{L}(\exp(\mathbf{u}_{k-1}^\wedge \tau), \mathbf{Q}_{k-1})$ and $\mathbf{w}_{k-1} \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_{k-1})$.

Regarding the observation model, two choices are possible:

(a) Similar to the propagation model $\boldsymbol{f}$, the observation model $\boldsymbol{h}$ is a $\boldsymbol{G}'$-valued function $\boldsymbol{h}(\boldsymbol{\chi}_k, \mathbf{v}_k) : \boldsymbol{G} \times \mathbb{R}^v \mapsto \boldsymbol{G}'$, where $\boldsymbol{G}' \subset \boldsymbol{G}$. This model assumes that measurements are given in the form of a matrix $\boldsymbol{Y}$ such that

$$\boldsymbol{Y}_k = \boldsymbol{h}(\boldsymbol{\chi}_k, \mathbf{v}_k) = \boldsymbol{\chi}_k \exp(\mathbf{v}_k^\wedge) \tag{3.42}$$

(b) Similar to the classical Euclidean case, the observation model is an $\mathbb{R}^p$-valued function $\mathbf{h}(\boldsymbol{\chi}_k, \mathbf{v}_k) : \boldsymbol{G} \times \mathbb{R}^v \mapsto \mathbb{R}^p$. This model assumes that measurements are given in the form of a vector $\mathbf{y}_k$ such that

$$\mathbf{y}_k = \mathbf{h}(\boldsymbol{\chi}_k, \mathbf{v}_k) \tag{3.43}$$

For both models, we have the Gaussian noise $\mathbf{v}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k)$. Notice that if a measurement of type (a) is used in a filtering process, we talk about Bayesian *fusion* on Lie groups [Wolfe et al., 2011], since both $\boldsymbol{Y}_k$ and $\boldsymbol{\chi}_k$ represent the same quantity, but obtained from

different sources, and fused to refine the estimation of $\boldsymbol{\chi}_k$. In the UKF-LG proposed in [Brossard et al., 2017], the model of type (a) can still be boiled down to a model of type (b). Indeed, for measurements represented by expression (3.42), it is possible to consider

$$\begin{aligned}\mathbf{y}_k &= \log(\hat{\boldsymbol{\chi}}_k^{-1}\boldsymbol{Y}_k)^\vee = \log\left(\hat{\boldsymbol{\chi}}_k^{-1}\boldsymbol{\chi}_k \exp(\mathbf{v}_k^\wedge)\right)^\vee = \log\left(\hat{\boldsymbol{\chi}}_k^{-1}\hat{\boldsymbol{\chi}}_k \exp(\boldsymbol{\xi}_k^\wedge)\exp(\mathbf{v}_k^\wedge)\right)^\vee \\ &= \log\left(\exp(\boldsymbol{\xi}_k^\wedge)\exp(\mathbf{v}_k^\wedge)\right)^\vee = \mathbf{h}(\boldsymbol{\xi}_k, \mathbf{v}_k)\end{aligned} \tag{3.44}$$

In this case, we have $p = d$. This model depends on $\boldsymbol{\chi}_k$ through $\boldsymbol{\xi}_k$. In the following, we only consider an observation model $\mathbf{h}$ of generic form (3.43).

3.3.2 State and covariance prediction

Finding an approximation of $\hat{\boldsymbol{\chi}}_{k|k-1}$ (3.36) while satisfying (3.38) is straightforward. Indeed, it is sufficient to propagate the mean by simply using the mean values of $\boldsymbol{\chi}_{k-1}$ and $\boldsymbol{U}_{k-1}$ in the kinematical model (3.41). Thus, we have

$$\hat{\boldsymbol{\chi}}_{k|k-1} = \boldsymbol{f}(\hat{\boldsymbol{\chi}}_{k-1|k-1}, \mathbf{u}_{k-1}, \mathbf{0}) = \hat{\boldsymbol{\chi}}_{k-1|k-1}\exp(\mathbf{u}_{k-1}^\wedge \tau). \tag{3.45}$$

In [Barfoot and Furgale, 2014], it has been shown that computing the prior mean in this way makes condition (3.38) true up to the third-order in the sense of the BCH formula expansion, and possibly further if $\mathbf{Q}_{k-1}$ is diagonal (i.e. no cross-correlations between the components of $\mathbf{w}_{k-1}$).

Finding an approximation of the prior covariance $\mathbf{P}_{k|k-1}$ (3.37) can be performed by relying on the classical Unscented Transform (UT), since all the uncertainty is contained in the vectors $\boldsymbol{\xi}_{k-1|k-1} \sim \mathcal{N}(\mathbf{0}, \mathbf{P}_{k-1|k-1})$ and $\mathbf{w}_{k-1} \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_{k-1})$. Let us assume that we have generated a set of N points and weights

$$\{\boldsymbol{\xi}_{k-1|k-1}^{(j)}, \mathbf{w}_{k-1}^{(j)}\}_{j=0,\dots,N-1} \quad \{W^{(j)}\}_{j=0,\dots,N-1}$$

Then, the prior covariance is computed thanks to the following procedure

$$\begin{aligned}\boldsymbol{\xi}_{k|k-1}^{(j)} &= \log\left(\hat{\boldsymbol{\chi}}_{k|k-1}^{-1}\hat{\boldsymbol{\chi}}_{k-1|k-1}\exp(\boldsymbol{\xi}_{k-1|k-1}^{(j)\wedge})\exp(\mathbf{u}_{k-1}^\wedge\tau)\exp(\mathbf{w}_{k-1}^{(j)\wedge}\tau)\right)^\vee \\ \mathbf{P}_{k|k-1} &= \sum_{i=0}^{N-1} W^{(j)}(\boldsymbol{\xi}_{k|k-1}^{(j)})(\boldsymbol{\xi}_{k|k-1}^{(j)})^\top\end{aligned} \tag{3.46}$$

Generating accurate sets of points and weights, capturing the distribution of $\boldsymbol{\xi}_{k-1|k-1} \sim \mathcal{N}(\mathbf{0}, \mathbf{P}_{k-1|k-1})$ and $\mathbf{w}_{k-1} \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_{k-1})$, can be performed classically, relying on the Scaled-UT (cf. expressions (2.34)) with its scaling parameters α, β, and κ. For notational convenience with the rest of this manuscript, these parameters are now re-denoted as

$$\alpha \to s_1, \quad \beta \to s_2, \quad \kappa \to s_3$$

■ Sigma-points and weights generation

Let the augmented state vector $\boldsymbol{\xi}_{aug} \in \mathbb{R}^{d'}$ with $d' = 2d$:

$$\boldsymbol{\xi}_{aug} = \begin{bmatrix}\boldsymbol{\xi}_{k-1|k-1} \\ \mathbf{w}_{k-1}\end{bmatrix} \sim \mathcal{N}\Big(\mathbf{0},\, \mathbf{P}_{aug} = \mathrm{diag}(\mathbf{P}_{k-1|k-1}, \mathbf{Q}_{k-1})\Big)$$

Following the Scaled Unscented method, $N = 2d' + 1$ sigma-points are generated via the usual deterministic procedure

$$\begin{aligned}
&\lambda = s_1^2(d' + s_3) - d' \\
&\boldsymbol{\xi}_{aug}^{(0)} = \mathbf{0} \\
&\boldsymbol{\xi}_{aug}^{(j)} = -\mathbf{S}_j^\top \sqrt{d' + \lambda}\,, \quad \forall j = j = 1, \dots, d' \\
&\boldsymbol{\xi}_{aug}^{(d'+j)} = \mathbf{S}_j^\top \sqrt{d' + \lambda}\,, \quad \forall j = 1, \dots, d'
\end{aligned} \tag{3.47}$$

where $\mathbf{S}_j$ is the j^{th} column of the upper-Cholesky factorization $\mathbf{S}$ of the covariance matrix $\mathbf{P}_{k-1|k-1} = \mathbf{S}^\top \mathbf{S}$. The weights are computed as follows:

$$\begin{aligned}
W^{(0)} &= \frac{\lambda}{d' + \lambda} + (1 - s_1^2 + s_2) \\
W^{(j)} &= \frac{1}{2(d' + \lambda)}\,, \quad \forall j = 1, \dots, 2d'
\end{aligned} \tag{3.48}$$

All of the above procedures are summarized in Algorithm 1.

3.3.3 State and covariance correction

From an available measurement $\mathbf{y}_k$, we can proceed to the correction of $\hat{\boldsymbol{\chi}}_{k|k-1}$ and $\mathbf{P}_{k|k-1}$. The UKF-LG provides a solution for the Bayesian update by using the measurement model $\mathbf{h}(\boldsymbol{\chi}_k, \mathbf{v}_k)$. This part does not differ much from what is usually done in the standard UKF. Given the noisy observation model

$$\mathbf{y}_k = \mathbf{h}\Big(\hat{\boldsymbol{\chi}}_{k|k-1} \exp(\boldsymbol{\xi}_{k|k-1}^{\wedge}), \mathbf{v}_k\Big),$$

we now augment the state vector $\boldsymbol{\xi}$ with the observation noise $\mathbf{v}$:

$$\boldsymbol{\xi}_{aug} = \begin{bmatrix} \boldsymbol{\xi}_{k|k-1} \\ \mathbf{v}_k \end{bmatrix} \sim \mathcal{N}\Big(\mathbf{0}, \mathbf{P}_{aug} = \mathrm{diag}(\mathbf{P}_{k|k-1}, \mathbf{R}_k)\Big)$$

The remaining procedure is thus similar to the UT as performed during the prediction, except that the sigma-points are now propagated into the function $\mathbf{h}$. This UT allows us to obtain an estimated measurement $\hat{\mathbf{y}}_k$ (which will be compared to the actual measurement $\mathbf{y}_k$), a covariance $\mathbf{P}_{\mathbf{y}_k}$, and a cross-covariance $\mathbf{P}_{\boldsymbol{\xi}_k \mathbf{y}_k}$, which will be used to compute the Kalman gain. The details are provided in Algorithm 2.

To perform the update and obtain the posterior mean and covariance, we start by computing the Kalman gain as

$$\mathbf{K} = \mathbf{P}_{\boldsymbol{\xi}_k \mathbf{y}_k} \mathbf{P}_{\mathbf{y}_k}^{-1} \tag{3.49}$$

Then, the prior covariance is updated in the same way as in the classical UKF to produce the posterior covariance:

$$\mathbf{P}_{k|k}^{-} = \mathbf{P}_{k|k-1} - \mathbf{K} \mathbf{P}_{\mathbf{y}_k} \mathbf{K}^\top \tag{3.50}$$

For the mean update, the correction term

$$\mathbf{K}\tilde{\mathbf{y}}_k = \mathbf{K}(\mathbf{y}_k - \hat{\mathbf{y}}_k) \tag{3.51}$$

is thus expressed in the Lie algebra. Hence, the posterior mean is given by

$$\hat{\boldsymbol{\chi}}_{k|k} = \hat{\boldsymbol{\chi}}_{k|k-1} \exp\left(\left[\mathbf{K}\tilde{\mathbf{y}}_k\right]^\wedge\right) \tag{3.52}$$

However, the posterior mean and covariance given above are merely approximations, valid up to the first-order in the sense of the BCH expansion. Hence, the covariance must proceed to a final update to account for this misestimation.

■ Final update

Before proceeding to the final update, it can be interesting to understand how the expression (3.52) has been found in the original paper [Brossard et al., 2017]. The Bayesian update implicitly starts by computing the uncertainty vector $\boldsymbol{\xi}^-_{k|k}$, which is a nonzero mean Gaussian such that $\boldsymbol{\xi}^-_{k|k} \sim \mathcal{N}(\mathbf{K}\tilde{\mathbf{y}}_k, \mathbf{P}^-_{k|k})$, namely,

$$\boldsymbol{\xi}^-_{k|k} = \mathbf{K}\tilde{\mathbf{y}}_k + \delta\boldsymbol{\xi} \quad \text{with} \quad \delta\boldsymbol{\xi} \sim \mathcal{N}(\mathbf{0}, \mathbf{P}^-_{k|k}) \tag{3.53}$$

Returning to the group, this means that we get

$$\boldsymbol{\chi}_{k|k} = \hat{\boldsymbol{\chi}}_{k|k-1} \exp\left(\boldsymbol{\xi}^{-\wedge}_{k|k}\right) = \hat{\boldsymbol{\chi}}_{k|k-1} \exp\left(\left[\mathbf{K}\tilde{\mathbf{y}}_k + \delta\boldsymbol{\xi}\right]^\wedge\right) \sim \mathcal{N}_{\mathsf{L}}(\hat{\boldsymbol{\chi}}_{k|k}, \mathbf{P}^-_{k|k}) \tag{3.54}$$

The equivariant Gaussian uncertainty framework imposes the condition that the posterior mean $\hat{\boldsymbol{\chi}}_{k|k}$ satisfies $\mathbb{E}(\boldsymbol{\xi}_{k|k}) = \mathbf{0}$. Thus, the expression (3.54) has to be developed in such a way that $\boldsymbol{\chi}_{k|k} = \hat{\boldsymbol{\chi}}_{k|k} \exp(\delta\boldsymbol{\xi}^\wedge)$. The approximation of the BCH formula to the first-order (cf. (3.23)) enables us to get

$$\begin{aligned} \boldsymbol{\chi}_{k|k} &= \hat{\boldsymbol{\chi}}_{k|k-1} \exp\left(\left[\mathbf{K}\tilde{\mathbf{y}}_k + \delta\boldsymbol{\xi}\right]^\wedge\right) \\ &\approx \underbrace{\hat{\boldsymbol{\chi}}_{k|k-1} \exp\left(\left[\mathbf{K}\tilde{\mathbf{y}}_k\right]^\wedge\right)}_{\hat{\boldsymbol{\chi}}_{k|k}} \exp(\delta\boldsymbol{\xi}^\wedge) \end{aligned} \tag{3.55}$$

We then label $\delta\boldsymbol{\xi}$ as $\boldsymbol{\xi}_{k|k}$ and get an updated state $\hat{\boldsymbol{\chi}}_{k|k}$ that satisfies the condition $\mathbb{E}(\boldsymbol{\xi}_{k|k}) = \mathbf{0}$.

However, as pointed out in [Brossard et al., 2018], this approximation is valid only if both the correction term $\mathbf{K}\tilde{\mathbf{y}}_k$ and the uncertainty $\delta\boldsymbol{\xi}$ are small (thus, justifying the approximation of the BCH to the first-order). In this case, the covariance $\mathbf{P}^-_{k|k}$ may be sufficient. But if $\delta\boldsymbol{\xi}$ is small compared to $\mathbf{K}\tilde{\mathbf{y}}_k$, a more accurate BCH approximation is necessary and takes advantage of the computation of the group jacobian[5]. It is known (see Appendix B.1) that for any $\mathbf{a}, \mathbf{b}, \mathbf{c} \in \mathbb{R}^d$ for which we have the exact relationship $\exp(\mathbf{c}^\wedge) = \exp(\mathbf{a}^\wedge)\exp(\mathbf{b}^\wedge)$, and assuming that $\mathbf{b}$ is small compared to $\mathbf{a}$, then a more accurate approximation of the BCH is given by

$$\mathbf{c} \approx \mathbf{a} + \boldsymbol{\mathcal{J}}_{\mathsf{R}}^{-1}\big(\mathrm{ad}(\mathbf{a}^\wedge)\big)\mathbf{b} \tag{3.56}$$

where $\mathrm{ad}(\cdot)$ is the small adjoint operator. Back to the group, we get

$$\begin{aligned} \exp(\mathbf{c}^\wedge) &\approx \exp\left(\left[\mathbf{a} + \boldsymbol{\mathcal{J}}_{\mathsf{R}}^{-1}\big(\mathrm{ad}(\mathbf{a}^\wedge)\big)\mathbf{b}\right]^\wedge\right) \\ \Rightarrow \exp(\mathbf{a}^\wedge)\exp(\mathbf{b}^\wedge) &\approx \exp\left(\left[\mathbf{a} + \boldsymbol{\mathcal{J}}_{\mathsf{R}}^{-1}\big(\mathrm{ad}(\mathbf{a}^\wedge)\big)\mathbf{b}\right]^\wedge\right) \end{aligned} \tag{3.57}$$

[5] If a left equivariant Gaussian is used, then the right jacobian is required. If a right equivariant Gaussian is used, then the left jacobian is required.

Now, if one relabels

$$\begin{aligned} \mathbf{a} &= \mathbf{K}\tilde{\mathbf{y}}_k \\ \mathcal{J}_{\mathsf{R}}^{-1}\big(\mathrm{ad}(\mathbf{a}^\wedge)\big)\mathbf{b} &= \delta\boldsymbol{\xi} \\ \Rightarrow \mathbf{b} &= \mathcal{J}_{\mathsf{R}}\big(\mathrm{ad}(\mathbf{a}^\wedge)\big)\delta\boldsymbol{\xi} = \mathcal{J}_{\mathsf{R}}\big(\mathrm{ad}([\mathbf{K}\tilde{\mathbf{y}}_k]^\wedge)\big)\delta\boldsymbol{\xi} \end{aligned} \tag{3.58}$$

then, this implies that

$$\exp\left([\mathbf{K}\tilde{\mathbf{y}}_k + \delta\boldsymbol{\xi}]^\wedge\right) \approx \exp\left([\mathbf{K}\tilde{\mathbf{y}}_k]^\wedge\right) \exp\left(\left[\mathcal{J}_{\mathsf{R}}\big(\mathrm{ad}([\mathbf{K}\tilde{\mathbf{y}}_k]^\wedge)\big)\delta\boldsymbol{\xi}\right]^\wedge\right) \tag{3.59}$$

In this case, the covariance is refined as

$$\mathbf{P}_{k|k} = \mathcal{J}_{\mathsf{R}}(\mathrm{ad}\left([\mathbf{K}\tilde{\mathbf{y}}_k]^\wedge\right))\,\mathbf{P}^-_{k|k}\mathcal{J}_{\mathsf{R}}(\mathrm{ad}\left([\mathbf{K}\tilde{\mathbf{y}}_k]^\wedge\right))^\top \tag{3.60}$$

This provides further (but necessary) overestimation of the error, since the corrected state (3.52) is misestimated due to a first-order approximation of the BCH formula.

It has to be noted that this final covariance update was absent from the original paper that introduced the UKF-LG [Brossard et al., 2017], but appeared in an extended work [Brossard et al., 2018]. As a sidenote, it also appeared in the Lie-Algebraic UKF of [Sjøberg and Egeland, 2022] and the CD-LG-EKF of [Bourmaud et al., 2015].

3.3.4 Algorithmic summary of the UKF-LG

The UKF-LG provides a suboptimal approximation of the Bayesian filter by resorting to the Unscented Transform (UT) in order to propagate the mean of the state, along with its covariance. The complete procedure is summarized in Algorithm 3 and will primarily serve as the baseline algorithm upon which the methodological contributions of this thesis are developed.

Although the UKF-LG provides a principled framework for performing Bayesian estimation on Lie groups, its performance ultimately depends on how uncertainty is represented and propagated through the kinematics of the system. The equivariant Gaussian framework, which is one of the foundations of the UKF-LG, offers a consistent approximation of uncertainty on the group manifold. However, this formulation also introduces intrinsic limitations. Overcoming these limitations is precisely the objective pursued in this thesis.

Algorithm 1: Unscented Transformation for the propagation model $\boldsymbol{f}$; $\boldsymbol{\chi} \sim \mathcal{N}_{\mathsf{L}}(\hat{\boldsymbol{\chi}}, \mathbf{P})$

Inputs: $\hat{\boldsymbol{\chi}}_{k-1|k-1}$, $\mathbf{P}_{k-1|k-1}$, $\mathbf{u}_{k-1}$, $\mathbf{Q}_{k-1}$, τ

1: Propagate the prior mean:

$$\hat{\boldsymbol{\chi}}_{k|k-1} = \hat{\boldsymbol{\chi}}_{k-1|k-1} \exp(\mathbf{u}_{k-1}^{\wedge}\tau)$$

2: State augmentation:

$$\mathbf{P}_{aug} = \operatorname{diag}(\mathbf{P}_{k-1|k-1}, \mathbf{Q}_{k-1}), \quad d' = 2d$$

3: Set $\{s_1, s_2, s_3\}$ and generate sigma-points:

$$\begin{aligned}
\lambda &= s_1^2(d' + s_3) - d' \\
\boldsymbol{\xi}_{aug}^{(0)} &= \mathbf{0} \\
\boldsymbol{\xi}_{aug}^{(j)} &= -\mathrm{Chol}(\mathbf{P}_{aug})_j^{\top} \sqrt{d' + \lambda}\,, \quad \forall j = 1, \ldots, d' \\
\boldsymbol{\xi}_{aug}^{(d'+j)} &= +\mathrm{Chol}(\mathbf{P}_{aug})_j^{\top} \sqrt{d' + \lambda}\,, \quad \forall j = 1, \ldots, d'
\end{aligned}$$

4: Generate weights:

$$\begin{aligned}
W^{(0)} &= \frac{\lambda}{d' + \lambda} + (1 - s_1^2 + s_2) \\
W^{(j)} &= \frac{1}{2(d' + \lambda)}\,, \quad \forall j = 1, \ldots, 2d'
\end{aligned}$$

5: Propagate sigma-points $\forall j = 0, \ldots, 2d'$:

$$\text{Let:} \quad \boldsymbol{\xi}_{aug}^{(j)} := \begin{bmatrix} \boldsymbol{\xi}_{k-1|k-1}^{(j)} \\ \mathbf{w}_{k-1}^{(j)} \end{bmatrix}$$

$$\boldsymbol{\xi}_{k|k-1}^{(j)} = \log\left(\hat{\boldsymbol{\chi}}_{k|k-1}^{-1} \hat{\boldsymbol{\chi}}_{k-1|k-1} \exp(\boldsymbol{\xi}_{k-1|k-1}^{(j)\wedge}) \exp(\mathbf{u}_{k-1}^{\wedge}\tau) \exp(\mathbf{w}_{k-1}^{(j)\wedge}\tau)\right)^{\vee}$$

6: Compute the covariance:

$$\mathbf{P}_{k|k-1} = \sum_{j=0}^{2d'} W^{(j)} \left(\boldsymbol{\xi}_{k|k-1}^{(j)}\right) \left(\boldsymbol{\xi}_{k|k-1}^{(j)}\right)^{\top}$$

Outputs: $\hat{\boldsymbol{\chi}}_{k|k-1}$, $\mathbf{P}_{k|k-1}$

Algorithm 2: Unscented Transformation for the observation model $\mathbf{h}$; $\boldsymbol{\chi} \sim \mathcal{N}_{\mathsf{L}}(\hat{\boldsymbol{\chi}}, \mathbf{P})$

Inputs: $\hat{\boldsymbol{\chi}}_{k|k-1}$, $\mathbf{P}_{k|k-1}$, $\mathbf{R}_k$

1: State augmentation:

$$\mathbf{P}_{aug} = \mathrm{diag}(\mathbf{P}_{k|k-1}, \mathbf{R}_k)\,, \quad d' = d + v$$

2: Set $\{s_1, s_2, s_3\}$ and generate sigma-points:

$$\begin{aligned}
\lambda &= s_1^2(d' + s_3) - d' \\
\boldsymbol{\xi}_{aug}^{(0)} &= \mathbf{0} \\
\boldsymbol{\xi}_{aug}^{(j)} &= -\mathrm{Chol}(\mathbf{P}_{aug})_j^\top \sqrt{d' + \lambda}\,, \quad \forall j = 1, \ldots, d' \\
\boldsymbol{\xi}_{aug}^{(d'+j)} &= +\mathrm{Chol}(\mathbf{P}_{aug})_j^\top \sqrt{d' + \lambda}\,, \quad \forall j = 1, \ldots, d'
\end{aligned}$$

3: Generate weights:

$$\begin{aligned}
W^{(0)} &= \frac{\lambda}{d' + \lambda} + (1 - s_1^2 + s_2) \\
W^{(j)} &= \frac{1}{2(d' + \lambda)}\,, \quad \forall j = 1, \ldots, 2d'
\end{aligned}$$

4: Propagate sigma-points $\forall j = 0, \ldots, 2d'$:

$$\begin{aligned}
\text{Let:} \quad \boldsymbol{\xi}_{aug}^{(j)} &:= \begin{bmatrix} \boldsymbol{\xi}_{k|k-1}^{(j)} \\ \mathbf{v}_k^{(j)} \end{bmatrix} \\
\mathbf{y}_k^{(j)} &= \mathbf{h}\Big(\hat{\boldsymbol{\chi}}_{k|k-1} \exp(\boldsymbol{\xi}_{k|k-1}^{(j)\wedge}), \mathbf{v}_k^{(j)}\Big)
\end{aligned}$$

5: Compute the mean and covariance of the measurement:

$$\begin{aligned}
\hat{\mathbf{y}}_k &= \sum_{j=0}^{2d'} W^{(j)} \mathbf{y}_k^{(j)} \\
\mathbf{P}_{\mathbf{y}_k} &= \sum_{j=0}^{2d'} W^{(j)} \big(\mathbf{y}_k^{(j)} - \hat{\mathbf{y}}_k\big)\big(\mathbf{y}_k^{(j)} - \hat{\mathbf{y}}_k\big)^\top
\end{aligned}$$

6: Compute the cross-covariance:

$$\mathbf{P}_{\boldsymbol{\xi}_k \mathbf{y}_k} = \sum_{j=0}^{2d'} W^{(j)} \big(\boldsymbol{\xi}_{k|k-1}^{(j)}\big)\big(\mathbf{y}_k^{(j)} - \hat{\mathbf{y}}_k\big)^\top$$

Outputs: $\hat{\mathbf{y}}_k$, $\mathbf{P}_{\mathbf{y}_k}$, $\mathbf{P}_{\boldsymbol{\xi}_k \mathbf{y}_k}$

Algorithm 3: UKF-LG algorithm ; $\boldsymbol{\mathcal{X}} \sim \mathcal{N}_{\mathsf{L}}(\hat{\boldsymbol{\mathcal{X}}}, \mathbf{P})$

Inputs: $\hat{\boldsymbol{\mathcal{X}}}_{k-1|k-1}$, $\mathbf{P}_{k-1|k-1}$, $\mathbf{u}_{k-1}$, $\mathbf{Q}_{k-1}$, $\mathbf{y}_k$, $\mathbf{R}_k$, τ

Time propagation from $k-1$ to k

1: Unscented Transformation through the model $\boldsymbol{f}$ (Algorithm 1):

$$\left[\hat{\boldsymbol{\mathcal{X}}}_{k|k-1}, \mathbf{P}_{k|k-1}\right] = UT_{\boldsymbol{f}}(\hat{\boldsymbol{\mathcal{X}}}_{k-1|k-1}, \mathbf{P}_{k-1|k-1}, \mathbf{u}_{k-1}, \mathbf{Q}_{k-1}, \tau)$$

Update using a measurement $\mathbf{y}_k$

2: Unscented Transformation through the model $\mathbf{h}$ (Algorithm 2):

$$\left[\hat{\mathbf{y}}_k, \mathbf{P}_{\mathbf{y}_k}, \mathbf{P}_{\boldsymbol{\xi}_k \mathbf{y}_k}\right] = UT_{\mathbf{h}}(\hat{\boldsymbol{\mathcal{X}}}_{k|k-1}, \mathbf{P}_{k|k-1}, \mathbf{R}_k)$$

3: Kalman gain and innovation:

$$\mathbf{K} = \mathbf{P}_{\boldsymbol{\xi}_k \mathbf{y}_k} \mathbf{P}_{\mathbf{y}_k}^{-1}$$
$$\tilde{\mathbf{y}}_k = \mathbf{y}_k - \hat{\mathbf{y}}_k$$

4: State update:

$$\hat{\boldsymbol{\mathcal{X}}}_{k|k} = \hat{\boldsymbol{\mathcal{X}}}_{k|k-1} \exp\left(\left[\mathbf{K}\tilde{\mathbf{y}}_k\right]^{\wedge}\right)$$
$$\mathbf{P}_{k|k}^{-} = \mathbf{P}_{k|k-1} - \mathbf{K}\mathbf{P}_{\mathbf{y}_k}\mathbf{K}^{\top}$$
$$\mathbf{P}_{k|k} = \boldsymbol{\mathcal{J}}_{\mathsf{R}}(\mathrm{ad}\big(\left[\mathbf{K}\tilde{\mathbf{y}}_k\right]^{\wedge}\big))\mathbf{P}_{k|k}^{-}\boldsymbol{\mathcal{J}}_{\mathsf{R}}(\mathrm{ad}\big(\left[\mathbf{K}\tilde{\mathbf{y}}_k\right]^{\wedge}\big))^{\top}$$

Outputs: $\hat{\boldsymbol{\mathcal{X}}}_{k|k}$, $\mathbf{P}_{k|k}$

3.4 Limitations of Lie-theoretic uncertainty models and alternatives

As discussed previously, the UKF-LG relies on an equivariant Gaussian representation of uncertainty. This formulation is convenient, as sigma points are generated in a vector space where a Gaussian probability density is well defined, and are then mapped onto the curved geometry of the group through the exponential map. However, the UKF-LG performs well if small uncertainty assumptions are met. In this case, a concentrated Gaussian assumption remains valid since the notion of Lie-theoretic covariance is also valid. However, this statement starts to become questionable for large uncertainties.

3.4.1 Validity of the notion of Lie-theoretic covariance

In subsection §3.2.1, we said that the notion of Lie-theoretic covariance is well-defined for highly concentrated pdfs. Thus, it justifies its use as a measure of dispersion in this case. However, if the uncertainty becomes large (i.e. if the covariance grows large), the notion becomes questionable. This drawback can lead to inconsistencies in a Bayesian filtering scheme.

In particular, recall that for any tightly focused distribution $\mathcal{P}$ on the random variable $\boldsymbol{\chi} \in \boldsymbol{G}$, the Lie-theoretic covariance is well-defined and expressed as

$$\mathbf{P} = \int_{\Gamma \subset \boldsymbol{G}} \log(\hat{\boldsymbol{\chi}}^{-1}\boldsymbol{\chi})^{\vee}[\log(\hat{\boldsymbol{\chi}}^{-1}\boldsymbol{\chi})^{\vee}]^{\top}\mathcal{P}(\boldsymbol{\chi})d\boldsymbol{\chi}$$

The reason why this definition of covariance "starts to break down when the uncertainty increases" [Chirikjian, 2011] (p. 383), is due to the contribution of elements that should be part of a set that is said to be of "measures zero". Ideally, we consider the elements of this set to have null probability measures to avoid the addition of redundant probabilities in the integral. However, a Lie group is not flat, and periodicity may appear for some elements of $\boldsymbol{G}$. This periodicity implies ambiguity in the integral domain. In particular, the exponential map is globally surjective, and several elements in $\mathfrak{g}$ can correspond to a single element in $\boldsymbol{G}$. When the distribution is concentrated, there is no problem since the exponential is bijective in the subdomain Γ, corresponding to "all of $\boldsymbol{G}$ minus the set of measures zero" ([Chirikjian, 2011] p.19). However, above this domain, problem appears due to the surjective nature of the exponential map. This is particularly seen for rotation groups $\boldsymbol{SO}(n)$ (and thus for the motion group $\boldsymbol{SE}(n)$ since it contains $\boldsymbol{SO}(n)$). Hence, for rotational variables involved within these Lie groups, a probability modeling that uses only the covariance as a notion of dispersion measure might not be the most appropriate, and the validity of an equivariant Gaussian framework is only local.

In that case, another fundamental limitation appears in the context of Bayesian filtering. For $\mathbb{R}^d$, uncertainties are most often assumed to be Gaussian-distributed in filtering applications because it has a number of remarkable properties, including the following:

1) The Gaussian distribution is indefinitely divisible and appears as a limiting distribution in the Central Limit Theorem.

2) It is the heat kernel (it solves a diffusion equation with constant coefficients).

3) It is the maximum entropy distribution subject to constraints on the mean and covariance.

For Lie groups, having all the previous properties satisfied by a parameterized family of distributions is generally not possible, and a compromise has to be made. At the end of subsection §2.1.2, we mentioned the importance of the maximum entropy distribution in a Bayesian filtering scheme, and the idea that it constitutes the most appropriate choice to model the prior distribution. In [Chirikjian, 2011] (p. 383), it is stated that "a maximum-entropy distribution on a Lie group cannot be defined by simply maximizing the entropy subject to a constraint on covariance, since the covariance concept itself becomes questionable". Hence, a Gaussian distribution, which is typically only described by its mean and covariance, cannot be the maximum entropy distribution on such manifolds.

For spherical data in general, other probability modelings (aside from an equivariant Gaussian) are possible, and have the advantage of providing global and geometrically-consistent uncertainty characterizations.

3.4.2 Probability distributions for attitude-only estimation

von Mises-Fisher distribution (vMF)

When talking about spherical data, the first distribution that comes to mind is the von Mises-Fisher (vMF) distribution [Mardia and Jupp, 2009], specifically tailored for a random variable living in the multi-dimensional sphere. In $\mathbb{R}^m$, we have the hyper-surface $\mathbb{S}^{m-1}$, and let $\mathbf{x}$ be a unit vector: $\mathbf{x}^\top\mathbf{x} = 1$. Let $\boldsymbol{\mu}$ be its mean direction (also unit-length), $\kappa > 0$ being a concentration parameter. The von Mises-Fisher pdf is given by

$$\mathcal{P}(\mathbf{x}) = \mathcal{VMF}(\mathbf{x}; \boldsymbol{\mu}, \kappa) = \frac{\kappa^{\frac{m}{2}-1}}{(2\pi)^{\frac{m}{2}} I_{\frac{m}{2}-1}(\kappa)} \exp\left(\kappa \boldsymbol{\mu}^\top \mathbf{x}\right) \tag{3.61}$$

where $I_n(\cdot)$ is the order n^{th} Bessel function of first kind. It is worth noticing that the vMF distribution is the one that maximizes the entropy on $\mathbb{S}^{m-1}$ under prescribed circular moments. Thus providing a suitable candidate for such manifolds.

For $m = 2$, it boils down to the univariate von Mises (vM) pdf $\mathcal{VM}(\theta; \overline{\theta}, \kappa)$ on $\mathbb{S}$, with:

$$\mathbf{x} = \begin{bmatrix} \cos(\theta) \\ \sin(\theta) \end{bmatrix} \quad \text{and} \quad \boldsymbol{\mu} = \begin{bmatrix} \cos(\overline{\theta}) \\ \sin(\overline{\theta}) \end{bmatrix}$$

So that $\boldsymbol{\mu}^\top\mathbf{x} = \begin{bmatrix} \cos(\overline{\theta}) & \sin(\overline{\theta}) \end{bmatrix} \begin{bmatrix} \cos(\theta) \\ \sin(\theta) \end{bmatrix} = \cos(\overline{\theta})\cos(\theta) + \sin(\overline{\theta})\sin(\theta) = \cos(\theta - \overline{\theta})$, and the normalization constant is $\frac{1}{2\pi I_0(\kappa)}$.

Matrix Fisher distribution (MF)

The matrix Fisher (MF) distribution models the attitude uncertainty when the random variable is a rotation matrix $\boldsymbol{\mathcal{R}} \in \boldsymbol{SO}(3)$ [Lee, 2016]. The pdf has the following form

$$\mathcal{P}(\boldsymbol{\mathcal{R}}) = \mathcal{MF}(\boldsymbol{\mathcal{R}}; \mathbf{F}) = \frac{1}{C(\mathbf{F})} \exp\Big(\text{tr}(\mathbf{F}^\top \boldsymbol{\mathcal{R}})\Big) \tag{3.62}$$

where $\mathbf{F}$ is a 3×3 matrix parameter, and $C(\mathbf{F})$ is a normalizing constant such that $\int_{\boldsymbol{SO}(3)} \mathcal{P}(\boldsymbol{\mathcal{R}}) d\boldsymbol{\mathcal{R}} = 1$. The measure $d\boldsymbol{\mathcal{R}}$ is still the Haar measure, introduced earlier, and bi-invariant under matrix multiplication.

The properties of the matrix Fisher distribution are mostly determined by the *proper* singular value decomposition (see [Lee, 2018] Definition 2.2) of the parameter $\mathbf{F}$ as

$$\mathbf{F} = \mathbf{U}\boldsymbol{S}\boldsymbol{V}^\top, \quad \boldsymbol{U}, \boldsymbol{V} \in \boldsymbol{SO}(3) \tag{3.63}$$

where $\boldsymbol{S} = \text{diag}(S_1, S_2, S_3)$ is the diagonal matrix of singular values.

Hence, the distribution of the random variable $\boldsymbol{\mathcal{R}}$ can be completely defined by these three parameter matrices. Specifically, the normalizing constant of depends only on the proper singular values, i.e. $C(\mathbf{F}) = C(\boldsymbol{S})$. The first moment of $\boldsymbol{\mathcal{R}}$ is given by $\mathbb{E}[\boldsymbol{\mathcal{R}}] = \boldsymbol{U}\mathbf{D}\boldsymbol{V}^\top$, where $\mathbf{D} = \text{diag}(D_1, D_2, D_3)$ with

$$D_i = \frac{1}{C(\boldsymbol{S})} \frac{\partial C(\boldsymbol{S})}{\partial S_i} \tag{3.64}$$

for $i = \{1, 2, 3\}$.

Multivariate von Mises distribution (mvM)

For m angular variables contained in the vector $\boldsymbol{\theta} = [\theta_1, ..., \theta_m]^\top$, the multivariate von Mises (mvM) pdf [Mardia et al., 2008] is similar to the von-Mises Fisher, and is defined as

$$\mathcal{P}(\boldsymbol{\theta}) = m\mathcal{VM}(\boldsymbol{\theta}; \overline{\boldsymbol{\theta}}, \boldsymbol{\kappa}, \boldsymbol{\Lambda}) = \frac{1}{T(\boldsymbol{\kappa}, \boldsymbol{\Lambda})} \exp\Big(\boldsymbol{\kappa}^\top \cos(\boldsymbol{\theta} - \overline{\boldsymbol{\theta}}) + \frac{1}{2} \sin(\boldsymbol{\theta} - \overline{\boldsymbol{\theta}})^\top \boldsymbol{\Lambda} \sin(\boldsymbol{\theta} - \overline{\boldsymbol{\theta}})\Big) \tag{3.65}$$

where

$$\begin{aligned} \boldsymbol{\theta} &= [\theta_1, \ldots, \theta_m]^\top, \quad \theta_i \in]-\pi, \pi] \; \forall i \\ \overline{\boldsymbol{\theta}} &= [\overline{\theta}_1, \ldots, \overline{\theta}_m]^\top, \quad \overline{\theta}_i \in]-\pi, \pi] \; \forall i \\ \boldsymbol{\kappa} &= [\kappa_1, ..., \kappa_m]^\top, \quad \kappa_i \geq 0 \end{aligned}$$

$T(\boldsymbol{\kappa}, \boldsymbol{\Lambda})$ is the normalization constant, ensuring that we have a probability. The cosine and sine functions are defined such that $\cos(\mathbf{a}) = \Big[\cos(\text{a}_1), \quad \ldots \quad, \cos(\text{a}_m)\Big]^\top$ and $\sin(\mathbf{a}) = \Big[\sin(\text{a}_1), \quad \ldots \quad, \sin(\text{a}_m)\Big]^\top$ for a given vector $\mathbf{a}$. The matrix $\boldsymbol{\Lambda}$ is a correlation parameter defined such that

$$\boldsymbol{\Lambda}_{ij} = \lambda_{ij} = \lambda_{ji}, \quad \text{for } i \neq j, \quad \lambda_{ii} = 0 \tag{3.66}$$

There are several differences with the von Mises-Fisher (3.61) pdf. First, it does not model the uncertainty of a unit vector, so the condition of unit-length is relaxed. Then, each

angular variable θ_i has its own concentration parameter. Also, an additional parameter $\mathbf{\Lambda}$ models correlation between angular components.

Under suitable limits, in particular when $\kappa_i \to +\infty$, the multivariate von Mises tends to a multivariate Gaussian, and we have

$$\boldsymbol{\theta} \sim \mathcal{N}(\overline{\boldsymbol{\theta}}, \mathbf{P}), \quad \text{where} \quad \mathbf{P}_{ii}^{-1} = \kappa_i,\ \mathbf{P}_{ij}^{-1} = -\lambda_{ij},\ i \neq j \tag{3.67}$$

Wrapped normal distribution (WN)

The wrapped normal (WN) pdf is simply obtained by periodizing the random variable. For a univariate case, it is given by

$$\mathcal{P}(\theta) = \mathcal{N}(\theta; \overline{\theta}, \sigma^2) = \frac{1}{\sigma\sqrt{2\pi}} \sum_{k=-\infty}^{+\infty} \exp\left[\frac{-(\theta - \overline{\theta} + 2k\pi)^2}{2\sigma^2}\right] \tag{3.68}$$

Bingham distribution

A random vector $\boldsymbol{q} \in \mathbb{S}^m$ follows a Bingham distribution [Bingham, 1974] with a $(m+1) \times (m+1)$ matrix parameter $\boldsymbol{A} = \boldsymbol{A}^\top$ if it has the following density function:

$$\mathcal{P}(\boldsymbol{q}) = \mathcal{B}(\boldsymbol{q}; \boldsymbol{A}) = \frac{1}{C(\boldsymbol{A})} \exp(\boldsymbol{q}^\top \boldsymbol{A} \boldsymbol{q}) \tag{3.69}$$

The Bingham distribution is antipodally symmetric, i.e., $\mathcal{P}(\boldsymbol{q}) = \mathcal{P}(-\boldsymbol{q})$ for any $\boldsymbol{q}$. The properties of a Bingham distribution are determined by the eigenvalue decomposition of the parameter $\boldsymbol{A} = \boldsymbol{M}\boldsymbol{Z}\boldsymbol{M}^\top$, where $\boldsymbol{M} = [M_0, \ldots, M_m]^\top$ are the eigenvectors of $\boldsymbol{A}$, and $\boldsymbol{Z} = \text{diag}(Z_0, \ldots, Z_m)$ are the corresponding eigenvalues.

Many other distributions exist for spherical data (e.g. Fisher-Bingham, Brownian, Kent, etc.), all of which can be found in [Mardia and Jupp, 2009].

3.4.3 Hybrid formulations for rotation/translation coupling

If the manifold is not only a sphere or a rotation group, but involves additional translational variables, hybrid formulations can still be developed, able to incorporate cross-correlations between rotations and translations.

Matrix Fisher-Gaussian distribution (MFG)

Defined for the product manifold $\boldsymbol{SO}(3) \times \mathbb{R}^n$, the matrix Fisher-Gaussian (MFG) pdf is given by

$$\begin{aligned} &\mathcal{P}(\boldsymbol{\mathcal{R}}, \boldsymbol{x}) = \mathcal{MFG}(\boldsymbol{\mathcal{R}}, \boldsymbol{x};\ \overline{\boldsymbol{x}}, \mathbf{P}, \boldsymbol{V}, \boldsymbol{S}, \boldsymbol{U}, \mathbf{M}) \\ &= \frac{1}{C(\boldsymbol{S})\sqrt{(2\pi)^n \det(\mathbf{P}_c)}} \exp\left(-\frac{1}{2}(\boldsymbol{x} - \overline{\boldsymbol{x}}_c)^\top \mathbf{P}_c^{-1} (\boldsymbol{x} - \overline{\boldsymbol{x}}_c)\right) \exp\left(\text{tr}(\mathbf{F}^\top \boldsymbol{\mathcal{R}})\right) \end{aligned} \tag{3.70}$$

where, $\overline{\boldsymbol{x}} \in \mathbb{R}^n$, $\mathbf{P} \in \mathbb{R}^{n \times n}$, $\boldsymbol{U}, \boldsymbol{V} \in \boldsymbol{SO}(3)$, $\boldsymbol{S} = \text{diag}(S_1, S_2, S_3)$ and $\mathbf{M} \in \mathbb{R}^{n \times 3}$. The matrix $\mathbf{M}$ is a coupling parameter such that we have $\overline{\boldsymbol{x}}_c \in \mathbb{R}^n$ given by

$$\overline{\boldsymbol{x}}_c = \overline{\boldsymbol{x}} + \mathbf{M}\boldsymbol{\nu}_{\boldsymbol{\mathcal{R}}} \quad \text{with,} \quad \boldsymbol{\nu}_{\boldsymbol{\mathcal{R}}} = (\boldsymbol{Q}\boldsymbol{S} - \boldsymbol{S}\boldsymbol{Q}^\top)^\vee \tag{3.71}$$

for $\boldsymbol{Q} = \boldsymbol{U}^\top \boldsymbol{\mathcal{R}} \boldsymbol{V}$, and $\mathbf{P}_c \in \mathbb{R}^{n\times n}$ is defined as

$$\mathbf{P}_c = \mathbf{P} - \mathbf{M}\left(\operatorname{tr}(\boldsymbol{S})\mathbf{I}_{3\times 3} - \boldsymbol{S}\right)\mathbf{M}^\top \tag{3.72}$$

Also, $\mathbf{F} = \boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^\top \in \mathbb{R}^{3\times 3}$, and $C(\boldsymbol{S})$ is the normalizing constant of the matrix Fisher distribution.

The MFG distribution has been used to solve Bayesian attitude estimation problem with gyro bias [Wang and Lee, 2022].

Bingham-Gaussian distribution (BG)

The Bingham-Gaussian (BG) [Wang and Lee, 2021] on $\mathbb{S}^3 \times \mathbb{R}^n$ is equivalent to the matrix Fisher-Gaussian on $\boldsymbol{SO}(3) \times \mathbb{R}^n$. Random elements $(\boldsymbol{q}, \boldsymbol{x}) \in \mathbb{S}^3 \times \mathbb{R}^n$ follow a Bingham-Gaussian distribution with parameters $\mathbf{M} \in \boldsymbol{SO}(4)$, $\boldsymbol{Z} = \operatorname{diag}(0, Z_1, Z_2, Z_3)$ where $0 \geq Z_1 \geq Z_2 \geq Z_3$, $\boldsymbol{\mu} \in \mathbb{R}^n$, $\mathbf{P} = \mathbf{P}^\top \in \mathbb{R}^{n\times n}$, and $\boldsymbol{N} \in \mathbb{R}^{n\times 3}$, if they have the following density function:

$$\begin{aligned}\mathcal{P}(\boldsymbol{q}, \boldsymbol{x}) &= \mathcal{BG}(\boldsymbol{q}, \boldsymbol{x};\, \boldsymbol{N}, \boldsymbol{Z}, \overline{\boldsymbol{x}}, \mathbf{P}, \mathbf{M}) \\ &= \frac{1}{C(\boldsymbol{Z})\sqrt{(2\pi)^n \det(\mathbf{P}_c)}} \exp\left(-\frac{1}{2}(\boldsymbol{x} - \overline{\boldsymbol{x}}_c)^\top \mathbf{P}_c^{-1}(\boldsymbol{x} - \overline{\boldsymbol{x}}_c)\right) \exp\left(\boldsymbol{q}^\top \boldsymbol{N}\boldsymbol{Z}\boldsymbol{N}^\top \boldsymbol{q}\right)\end{aligned} \tag{3.73}$$

where $C(\boldsymbol{Z})$ is the normalizing constant for the Bingham distribution. $\mathbf{P}_c \in \mathbb{R}^{n\times n}$ is given by

$$\mathbf{P}_c = \mathbf{P} + \frac{1}{2}\mathbf{M}\operatorname{diag}(Z_1, Z_2, Z_3)\,\mathbf{M}^\top \tag{3.74}$$

and $\overline{\boldsymbol{x}}_c \in \mathbb{R}^n$ is given by

$$\overline{\boldsymbol{x}}_c = \overline{\boldsymbol{x}} + \mathbf{M}\boldsymbol{\nu}_{\boldsymbol{q}} = \overline{\boldsymbol{x}} + \mathbf{M}\begin{bmatrix}(Z_2 - Z_3)\mathrm{a}_2\mathrm{a}_3 - Z_1\mathrm{a}_0\mathrm{a}_1 \\ (Z_3 - Z_1)\mathrm{a}_1\mathrm{a}_3 - Z_2\mathrm{a}_0\mathrm{a}_2 \\ (Z_1 - Z_2)\mathrm{a}_1\mathrm{a}_2 - Z_3\mathrm{a}_0\mathrm{a}_3\end{bmatrix} \tag{3.75}$$

with $[\mathrm{a}_0, \mathrm{a}_1, \mathrm{a}_2, \mathrm{a}_3]^\top = \boldsymbol{N}^\top \boldsymbol{q}$.

Partially wrapped normal distribution (PWN)

For a random vector $\boldsymbol{x} \in \mathbb{S} \times \mathbb{R}^2$, the pdf of the Partially Wrapped Normal (PWN) distribution is given by

$$\mathcal{P}(\boldsymbol{x}) = \mathcal{PWN}(\boldsymbol{x};\, \overline{\boldsymbol{x}}, \mathbf{P}) = \sum_{k=-\infty}^{+\infty} \mathcal{N}\left(\boldsymbol{x} + \begin{bmatrix}2k\pi \\ 0 \\ 0\end{bmatrix};\, \overline{\boldsymbol{x}}, \mathbf{P}\right)$$

with $\overline{\boldsymbol{x}} \in \mathbb{S} \times \mathbb{R}^2$ and $\mathbf{P} \in \mathbb{R}^{n\times n}$ is symmetric positive definite. These parameters do not retain their traditional meaning as mean and covariance. Although defined for the manifold $\mathbb{S} \times \mathbb{R}^2$, it was originally developed especially to solve a pose estimation problem on $\boldsymbol{SE}(2)$ [Kurz et al., 2014].

Gauss-von Mises distribution (GvM)

The pdf of the Gauss-von Mises (GvM) for an element $(\boldsymbol{x}, \theta) \in \mathbb{R}^n \times \mathbb{S}$ is described below

$$\mathcal{P}(\boldsymbol{x}, \theta) = \mathcal{GVM}(\boldsymbol{x}, \theta;\ \overline{\boldsymbol{x}}, \mathbf{P}_{\boldsymbol{xx}}, \overline{\theta}, \boldsymbol{\beta}, \boldsymbol{\Gamma}, \kappa) = \mathcal{N}(\boldsymbol{x}; \overline{\boldsymbol{x}}, \mathbf{P}_{\boldsymbol{xx}})\, \mathcal{VM}(\theta; \Theta(\boldsymbol{x}), \kappa) \tag{3.76}$$

$$\mathcal{N}(\boldsymbol{x}; \overline{\boldsymbol{x}}, \mathbf{P}_{\boldsymbol{xx}}) = \frac{1}{\sqrt{(2\pi)^n \det(\mathbf{P}_{\boldsymbol{xx}})}} \exp\left(-\frac{1}{2}(\boldsymbol{x} - \overline{\boldsymbol{x}})^\top \mathbf{P}_{\boldsymbol{xx}}^{-1} (\boldsymbol{x} - \overline{\boldsymbol{x}})\right) \tag{3.77}$$

$$\mathcal{VM}(\theta; \Theta(\boldsymbol{x}), \kappa) = \frac{1}{2\pi I_0(\kappa)} \exp\left(\kappa \cos(\theta - \Theta(\boldsymbol{x}))\right) \tag{3.78}$$

$$\text{where,} \quad \Theta(\boldsymbol{x}) = \overline{\theta} + \boldsymbol{\beta}^\top \boldsymbol{z} + \frac{1}{2}\boldsymbol{z}^\top \boldsymbol{\Gamma} \boldsymbol{z} \quad \text{with,} \quad \boldsymbol{z} = \mathbf{A}^{-1}(\boldsymbol{x} - \overline{\boldsymbol{x}}) \quad \text{and} \quad \mathbf{P}_{\boldsymbol{xx}} = \mathbf{A}\mathbf{A}^\top \tag{3.79}$$

The GvM is parameterized by six statistical parameters: a mean $\overline{\boldsymbol{x}} \in \mathbb{R}^n$ and a covariance matrix $\mathbf{P}_{\boldsymbol{xx}} \in \mathbb{R}^{n \times n}$, for the linear part $\boldsymbol{x}$. A univariate von Mises distribution is used to represent the uncertainty of the angular part θ, where $\kappa > 0$, and $\Theta(\boldsymbol{x})$ is a polynomial combination of $\boldsymbol{x}$ via its "centered-normalized" counterpart $\boldsymbol{z}$. The statistical parameters $\boldsymbol{\beta} \in \mathbb{R}^n$ and $\boldsymbol{\Gamma} \in \mathbb{R}^{n \times n}$ model correlations between $\boldsymbol{x}$ and θ. Thus, the scalar parameter $\overline{\theta}$ can be seen as the mean angle when there is no correlations between the translational and rotational components. The aim of this distribution, introduced in [Horwood and Poore, 2014], was to capture the banana-shaped density that can be observed in a cluster of spatial debris.

All of the distributions listed above are potential alternatives to model uncertainty on manifolds that exhibit coupled linear and angular variables. We see that depending on how angular elements are treated, correlations are handled differently. Furthermore, they all have different statistical parameters in their definition, and some of these distributions do not necessarily generalize to any matrix Lie group $\boldsymbol{G}$, which may lead to different filters design, each specific to a given problem. However, they all have a common characteristic: they do not rely on the covariance to measure the dispersion of the angular part.

Greatly inspired by these ideas, we propose in the rest of this manuscript (Chapter 4) to develop a candidate uncertainty model that could apply for any motion group $\boldsymbol{SE}(n)$, or extended formulations ($\boldsymbol{SE}_k(n)$, $\boldsymbol{SE}_k(n) \times \mathbb{R}^N$), with the additional objective to tailor it for a UKF-LG. This will be considered as the main contribution of the thesis. Moreover, a second part of the contributions (Chapter 5) concerns the objective to improve the UKF-LG's performance in the case where an equivariant Gaussian remains reasonably valid. This secondary contribution primarily relies on optimal quantization, whose concept has been reviewed in Chapter 2 Section §2.5.

Chapter 4

Development of an uncertainty model for the UKF-LG

Contents

4.1 Practical limitations of the Lie-theoretic covariance 70
4.2 Mathematical preliminaries on the Gauss-von Mises distribution 72
4.2.1 The von Mises distribution and its statistical parameters 72
4.2.2 Construction of the Gauss-von Mises distribution 73
4.2.3 Numerical quadrature of a Gauss-von Mises distribution 75
4.3 Development of an equivariant von Mises-Gaussian distribution for $\boldsymbol{SE}(n)$ 77
4.3.1 Proposed method 77
4.3.2 von Mises-Gaussian on $\mathbb{S}^m \times \mathbb{R}^\ell$ 78
4.3.3 Equivariant von Mises-Gaussian on $\boldsymbol{SE}(n)$ 80
4.4 Adaptation of the UKF-LG to the von Mises-Gaussian distribution 82
4.4.1 Sigma-points and weights generation 82
4.4.2 Application to the Unscented transform on Lie groups 84
4.4.3 Reformulation of the UKF-LG 85

This chapter describes a first approach to deal with the accurate representation and propagation of uncertainty in the specific case of a UKF on Lie groups. It aims to address the limitations presented previously about the validity of the Lie-theoretic covariance (cf. subsection §3.4.1) by focusing our efforts on designing a new probability model, and adapting the UKF-LG algorithm to account for the proposed probability distribution.

The remainder of this chapter is organized as follows: Section §4.1 illustrates the problems related to the notion of covariance by providing an example on $\boldsymbol{SO}(2)$. Additionally, we present how a concentration parameter (of the von Mises distribution) can be an alternative to model the dispersion of an angular variable. From this observation, we review in Section §4.2, the key mathematical tools underlying the GvM distribution (3.76) on the state space $\mathbb{S} \times \mathbb{R}^\ell$, how it is constructed, and how one can generate the sigma-points and weights of such a distribution. Building on these foundations, Section §4.3 presents

an alternative distribution model for the state space $\mathbb{S}^m \times \mathbb{R}^\ell$, followed by the equivariant distribution tailored to $\boldsymbol{SE}(n)$. Finally, Section §4.4 revisits the UKF-LG to incorporate this new equivariant uncertainty model into its recursive estimation scheme. The developments presented in this chapter have been the subject of a publication in the Control & Decision Conference (CDC) [Pravong et al., 2025a].

4.1 Practical limitations of the Lie-theoretic covariance

The purpose of this section is to illustrate a fundamental limitation of covariance-based uncertainty representations on Lie groups, which arises from the surjective nature of the exponential map. This limitation is best understood through a simple example on the rotation group $\boldsymbol{SO}(2)$. Each rotation matrix is uniquely defined, but can be represented by infinitely many angles, since angular coordinates are periodic modulo a 2π shift. As a consequence, the exponential map from the Lie algebra to the group is surjective: several elements of the Lie algebra correspond to the same element on the group.

This non-uniqueness becomes problematic when uncertainty is represented in exponential coordinates. Consider a random variable $\boldsymbol{\mathcal{R}} \in \boldsymbol{SO}(2)$ whose covariance is defined as

$$P = \int_{\Gamma \subset \boldsymbol{SO}(2)} \log(\hat{\boldsymbol{\mathcal{R}}}^{-1}\boldsymbol{\mathcal{R}})^\vee [\log(\hat{\boldsymbol{\mathcal{R}}}^{-1}\boldsymbol{\mathcal{R}})^\vee]^\top \mathcal{N}_{\mathsf{L}}(\boldsymbol{\mathcal{R}}) d\boldsymbol{\mathcal{R}} = \int_{]-\pi,\pi] \subset \mathbb{R}} \theta\theta^\top \mathcal{N}(\theta) |\det(\mathbf{J})| d\theta \tag{4.1}$$

which corresponds to the usual Lie-theoretic covariance under the Gaussian assumption (cf. expression (3.31)). When the dispersion is small, the logarithmic map is locally well-defined, and the covariance provides a meaningful measure of uncertainty. However, for large dispersions, this representation becomes ambiguous. Indeed, when a covariance is sufficiently large, samples drawn from this covariance may lie outside the principal domain $]-\pi, \pi]$. In this regime, different samples may correspond to identical rotations once mapped onto $\boldsymbol{SO}(2)$, due to angular periodicity. As a result, it becomes unclear whether a given angular value has contributed once or multiple times to the probability density. That is why we have to consider a "set of measures zero" (cf. Chapter 3 subsection §3.4.1) that must be excluded from the principle domain Γ so that they do not contribute to the probability density. In theory, it permits a locally valid definition of the covariance by the integral (4.1). But in practice, this structural limitation (local validity) brings out ambiguity when the covariance is used as the sole measure of dispersion. To illustrate this point numerically, we generate N particles in the Lie algebra $\{\xi_{\theta_i}\}_{i=1,\dots,N}$, drawn from $\xi_\theta \sim \mathcal{N}(0, \sigma_\theta^2)$, and such that we have $\boldsymbol{\mathcal{R}} = \hat{\boldsymbol{\mathcal{R}}} \exp(\xi_\theta^\wedge) \sim \mathcal{N}_{\mathsf{L}}(\hat{\boldsymbol{\mathcal{R}}}, \sigma_\theta^2)$. We then compute two empirical variances: first $P_s(\xi_\theta)$ as

$$P_s(\xi_\theta) := \sum_{i=1}^{N} \frac{1}{N} (\xi_{\theta_i})(\xi_{\theta_i})^\top, \tag{4.2}$$

which is directly computed from the generated sample, and $P_s(\xi'_\theta)$ as

$$P_s(\xi'_\theta) := \sum_{i=1}^{N} \frac{1}{N} (\xi'_{\theta_i})(\xi'_{\theta_i})^\top \tag{4.3}$$
$$\text{where,} \quad \xi'_{\theta_i} = \log(\hat{\boldsymbol{\mathcal{R}}}^{-1}\boldsymbol{\mathcal{R}}_i)^\vee = \log(\hat{\boldsymbol{\mathcal{R}}}^{-1}\hat{\boldsymbol{\mathcal{R}}}\exp(\xi^\wedge_{\theta_i}))^\vee = \log(\exp(\xi^\wedge_{\theta_i}))^\vee,$$

ensuring that all particles ξ'_{θ_i} lie within the principal domain $]-\pi, \pi]$. As shown in Figure 4.1, when the dispersion becomes large, the variance $P_s(\xi_\theta)$ (blue curve) grows unboundedly, while $P_s(\xi'_\theta)$ (orange curve) converges to a finite asymptotic value.

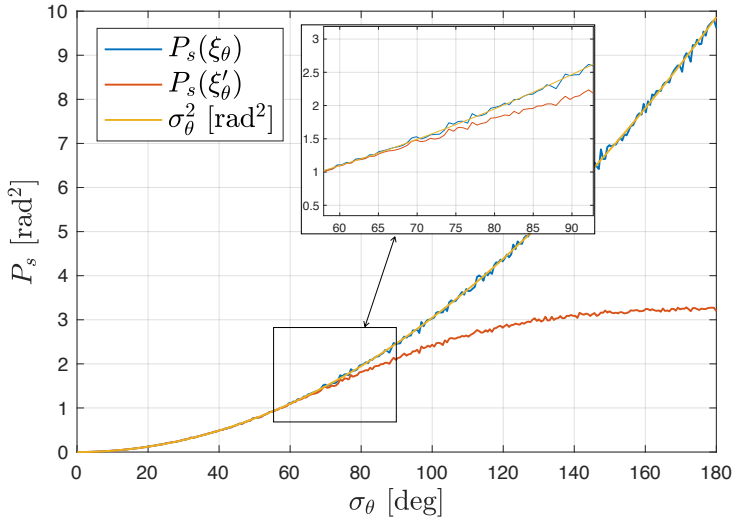


Figure 4.1: Comparison between the sample variances (4.2) and (4.3), for $N = 10^4$ particles, as a function of σ_θ.

This divergence between $P_s(\xi'_\theta)$ and $P_s(\xi_\theta)$ begins at $\sigma_\theta = 70°$. Beyond this threshold, a significant portion of the particles ξ_{θ_i} may lie outside the principal interval $]-\pi, \pi]$, leading to duplicated rotations due to periodicity (see the blue distribution of Figure 4.2). Hence, duplicates may have been generated among the N points ξ_{θ_i}. Although applying the logarithmic map after exponentiation removes these duplicates and yields a bounded variance, this observation clearly shows that generating samples directly from a Gaussian distribution in the Lie algebra is no longer meaningful when the dispersion is large. In particular, the covariance ceases to be a reliable measure of uncertainty.

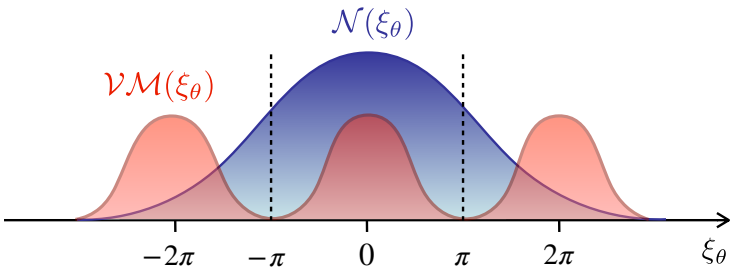


Figure 4.2: Illustration of the dispersion of an angle ξ_θ when: (blue) ξ_θ is treated as an unbounded random variable on $\mathbb{R}$ with a non-periodic distribution $\mathcal{N}(\xi_\theta)$; (red) ξ_θ is treated as a bounded random variable in $]-\pi, \pi]$ with a circular distribution $\mathcal{VM}(\xi_\theta)$.

This issue has direct implications for the UKF-LG since sigma-points are generated

from the state covariance $\mathbf{P}_{k-1}$ prior to applying the Unscented transform. When this covariance is large, the sigma-point generation itself becomes ambiguous. While the subsequent propagation through the system dynamics, via exponential and logarithmic maps, may partially mitigate the issue, there is no guarantee that the reconstructed covariance $\mathbf{P}_k$ remains meaningful if the uncertainty stays large due to the fact that external noise may contribute to a growing uncertainty during the dynamical transformations. Therefore, repeated sigma-point generation from such a covariance may be ill-founded. One way to overcome this ambiguity is to rely on alternative measures of dispersion that are intrinsically compatible with periodic variables. A natural candidate is the concentration parameter, whose empirical estimate κ_s can be computed from the sample. Interestingly, computing $\kappa_s(\xi_\theta)$ and $\kappa_s(\xi'_\theta)$ yields identical results, regardless of the magnitude of σ_θ. This demonstrates that the concentration parameter is invariant to angular periodicity, unlike the covariance. These observations strongly suggest that, when the state space includes periodic variables, it is preferable to adopt probability densities specifically designed for such variables, such as those given in subsection §3.4.3. The von Mises distribution is a particularly appropriate choice, since it naturally accounts for periodicity, as illustrated by the red distribution in Figure 4.2.

More generally, these results indicate that for large uncertainties on motion groups, such as $\boldsymbol{SE}(n)$, a more accurate probabilistic modeling must be introduced at the tangent space level. This motivates the development of hybrid distributions combining von Mises and Gaussian components. In this thesis, we follow this direction by introducing a "von Mises–Gaussian" (vMG) distribution, mainly inspired by the construction of the Gauss-von Mises (GvM) distribution (3.76).

4.2 Mathematical preliminaries on the Gauss-von Mises distribution

In this section, we review in detail the construction of the Gauss-von Mises distribution of [Horwood and Poore, 2014], its properties and how to generate sigma-points based on this uncertainty model. This will serve as the foundation for our subsequent development of the von Mises-Gaussian distribution.

4.2.1 The von Mises distribution and its statistical parameters

Recall that the von Mises distribution of a random variable θ, with mean direction $\overline{\theta}$ and concentration parameter κ, is given by:

$$\mathcal{VM}(\theta; \overline{\theta}, \kappa) = \frac{1}{2\pi I_0(\kappa)} \exp\left(\kappa \cos(\theta - \overline{\theta})\right). \tag{4.4}$$

The mean direction $\overline{\theta} \in]-\pi, \pi]$ corresponds to the mean angle formed by a unit vector with the real axis (cf. Figure 4.3).

The p^{th} circular moment of the random variable $\theta \sim \mathcal{VM}(\theta;\, \overline{\theta}, \kappa)$ is given by

$$\mathbb{E}[e^{ip\theta}] = \int_{-\pi}^{\pi} e^{ip\theta}\, \mathcal{VM}(\theta;\, \overline{\theta}, \kappa) d\theta = \frac{I_{|p|}(\kappa)}{I_0(\kappa)} e^{ip\overline{\theta}} = \frac{I_{|p|}(\kappa)}{I_0(\kappa)} \left(\cos(p\overline{\theta}) + i \sin(p\overline{\theta})\right) \tag{4.5}$$

In particular, the first circular moment ($p = 1$) characterizes both the mean direction and the dispersion of θ, as seen in Figure 4.3.

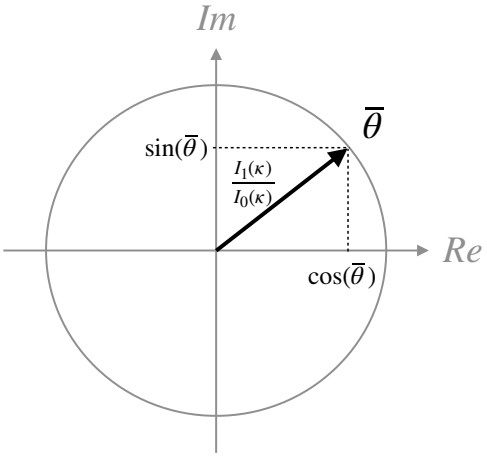


Figure 4.3: A random variable $\theta \sim \mathcal{VM}(\theta;\, \overline{\theta}, \kappa)$ defined on the unit circle $\mathbb{S}$. The resultant $\frac{I_1(\kappa)}{I_0(\kappa)}$ measures the dispersion via the concentration parameter κ. If $\kappa \to +\infty$, the distribution is said highly concentrated and the resultant $\frac{I_1(\kappa)}{I_0(\kappa)} \to 1$. Inversely, if $\kappa \to 0$, the distribution is highly spread and the resultant $\frac{I_1(\kappa)}{I_0(\kappa)} \to 0$.

4.2.2 Construction of the Gauss-von Mises distribution

The Gauss-von Mises (GvM) distribution is a joint distribution designed for systems that involve one angle and several translational variables:

$$\mathbf{x} = \begin{bmatrix} \theta \\ \boldsymbol{x} \end{bmatrix} \in \mathbb{S} \times \mathbb{R}^{\ell}, \quad \text{with} \;\; \ell \geq 1$$

In that sense, the angular variable θ is treated as a bounded variable living in the sphere $\mathbb{S}$ (instead of an unbounded cartesian variable in $\mathbb{R}$). The GvM pdf is constructed by combining a von Mises distribution for the angular variable and a Gaussian distribution for the translational components. Its construction starts with two random vectors $\boldsymbol{a} \in \mathbb{R}^{n_a}$ and $\boldsymbol{b} \in \mathbb{R}^{n_b}$, which have a joint Gaussian distribution:

$$\mathcal{P}(\boldsymbol{a}, \boldsymbol{b}) = \mathcal{N}\left(\begin{bmatrix} \overline{\boldsymbol{a}} \\ \overline{\boldsymbol{b}} \end{bmatrix}, \begin{bmatrix} \mathbf{P}_{\boldsymbol{aa}} & \mathbf{P}_{\boldsymbol{ba}}^{\top} \\ \mathbf{P}_{\boldsymbol{ba}} & \mathbf{P}_{\boldsymbol{bb}} \end{bmatrix} \right) \tag{4.6}$$

Applying the law of conditional probability (Def. A.2) and the Schur complement decomposition [Van Loan and Golub, 1996], the joint pdf can be expressed as:

$$\begin{aligned} \mathcal{P}(\boldsymbol{a}, \boldsymbol{b}) &= \underbrace{\mathcal{N}(\boldsymbol{a}; \overline{\boldsymbol{a}}, \mathbf{P}_{\boldsymbol{aa}})}_{\mathcal{P}(\boldsymbol{a})} \underbrace{\mathcal{N}(\boldsymbol{b}|\boldsymbol{a}; \boldsymbol{g}(\boldsymbol{a}), \tilde{\mathbf{P}}_{\boldsymbol{bb}})}_{\mathcal{P}(\boldsymbol{b}|\boldsymbol{a})} \\ \boldsymbol{g}(\boldsymbol{a}) &= \overline{\boldsymbol{b}} + \mathbf{P}_{\boldsymbol{ba}} \mathbf{P}_{\boldsymbol{aa}}^{-1} (\boldsymbol{a} - \overline{\boldsymbol{a}}) \\ \tilde{\mathbf{P}}_{\boldsymbol{bb}} &= \mathbf{P}_{\boldsymbol{bb}} - \mathbf{P}_{\boldsymbol{ba}} \mathbf{P}_{\boldsymbol{aa}}^{-1} \mathbf{P}_{\boldsymbol{ba}}^{\top} \end{aligned} \tag{4.7}$$

The expression above defines a pdf in $[\boldsymbol{a}^\top, \boldsymbol{b}^\top]^\top$ for any function $\boldsymbol{g}(\boldsymbol{a})$ and any symmetric positive-definite matrix $\tilde{\mathbf{P}}_{\boldsymbol{bb}}$. For the random vector $[\theta, \boldsymbol{x}]$, the authors proposed to relabel $\boldsymbol{a}$ as $\boldsymbol{x}$ ($n_a = \ell$), and $\boldsymbol{b}$ as θ ($n_b = 1$) such that (4.7) defines a pdf on $\mathbb{S} \times \mathbb{R}^\ell$. The pdf is then defined as the product of a multivariate Gaussian pdf with a univariate von Mises pdf:

$$\mathcal{P}(\boldsymbol{x}, \theta) = \mathcal{GVM}(\boldsymbol{x}, \theta;\ \overline{\boldsymbol{x}}, \mathbf{P}_{\boldsymbol{xx}}, \overline{\theta}, \boldsymbol{\beta}, \boldsymbol{\Gamma}, \kappa) = \underbrace{\mathcal{N}(\boldsymbol{x}; \overline{\boldsymbol{x}}, \mathbf{P}_{\boldsymbol{xx}})}_{\mathcal{P}(\boldsymbol{x})}\ \underbrace{\mathcal{VM}(\theta|\boldsymbol{x}; \Theta(\boldsymbol{x}), \kappa)}_{\mathcal{P}(\theta|\boldsymbol{x})} \tag{4.8}$$

with,

$$\mathcal{N}(\boldsymbol{x}; \overline{\boldsymbol{x}}, \mathbf{P}_{\boldsymbol{xx}}) = \frac{1}{\sqrt{(2\pi)^\ell \det(\mathbf{P}_{\boldsymbol{xx}})}} \exp\left(-\frac{1}{2}(\boldsymbol{x} - \overline{\boldsymbol{x}})^\top \mathbf{P}_{\boldsymbol{xx}}^{-1}(\boldsymbol{x} - \overline{\boldsymbol{x}})\right) \tag{4.9}$$

and

$$\mathcal{VM}(\theta; \Theta(\boldsymbol{x}), \kappa) = \frac{1}{2\pi I_0(\kappa)} \exp\left(\kappa \cos(\theta - \Theta(\boldsymbol{x}))\right). \tag{4.10}$$

Notice that the parameter $\tilde{\mathbf{P}}_{\theta\theta}$ has been replaced by a concentration parameter κ as measure of dispersion, and the function $\boldsymbol{g}(\boldsymbol{x})$ has been replaced by a function $\Theta(\boldsymbol{x})$ of the form

$$\Theta(\boldsymbol{x}) = \overline{\theta} + \boldsymbol{\beta}^\top \boldsymbol{z} + \frac{1}{2}\boldsymbol{z}^\top \boldsymbol{\Gamma} \boldsymbol{z} \quad \text{with} \quad \boldsymbol{z} = \mathbf{A}^{-1}(\boldsymbol{x} - \overline{\boldsymbol{x}}) \quad \text{and} \quad \mathbf{P}_{\boldsymbol{xx}} = \mathbf{A}\mathbf{A}^\top \tag{4.11}$$

to model correlations between translational and rotational components. The function $\boldsymbol{g}(\boldsymbol{x})$ initially involved a linear dependency between θ and $\boldsymbol{x}$, but $\Theta(\boldsymbol{x})$ involves a quadratic polynomial dependency. This was a choice of the authors of [Horwood and Poore, 2014] to model the observed banana-shaped distribution specific to the problem they considered.

■ Canonical form

The conversion into a canonical form simplifies the use of the Gauss–von Mises distribution by separating the angular and translational components. This form is obtained through the following change of variables:

$$\boldsymbol{z} = \mathbf{A}^{-1}(\boldsymbol{x} - \overline{\boldsymbol{x}}), \quad \phi = \theta - \overline{\theta} - \boldsymbol{\beta}^\top \boldsymbol{z} - \frac{1}{2}\boldsymbol{z}^\top \boldsymbol{\Gamma} \boldsymbol{z} \tag{4.12}$$

This transformation reduces $(\boldsymbol{x}, \theta) \sim \mathcal{GVM}(\boldsymbol{x}, \theta;\ \overline{\boldsymbol{x}}, \mathbf{P}_{\boldsymbol{xx}}, \overline{\theta}, \boldsymbol{\beta}, \boldsymbol{\Gamma}, \kappa)$ to its *canonical* (or *centered-normalized*) form, with pdf

$$\mathcal{GVM}(\boldsymbol{z}, \phi;\ \mathbf{0}, \mathbf{I}, 0, \mathbf{0}, \mathbf{0}, \kappa) = \mathcal{N}(\boldsymbol{z};\ \mathbf{0}, \mathbf{I}) \times \mathcal{VM}(\phi;\ 0, \kappa). \tag{4.13}$$

■ Low-order moments of the canonical form

The first and second (standard) moments of the translational part $\boldsymbol{z}$ are given by

$$\mathbb{E}[\boldsymbol{z}] = \mathbf{0}, \quad \mathbb{E}[\boldsymbol{z}\boldsymbol{z}^\top] = \mathbf{I}. \tag{4.14}$$

Additionally, for a canonical Gaussian, the third moment (skew) is zero, and the fourth moment (kurtosis) is 3 for all random variables z_j ($j = 1, \ldots, \ell$). The first and second (circular) moments of the rotational part ϕ are given by

$$\mathbb{E}[e^{i\phi}] = \frac{I_1(\kappa)}{I_0(\kappa)}, \quad \mathbb{E}[e^{i2\phi}] = \frac{I_2(\kappa)}{I_0(\kappa)} \tag{4.15}$$

These statistical moments are essential for the subsequent steps, in which a quadrature rule must be established to approximate moment integrals. This quadrature procedure enables the determination of the sigma-points required for uncertainty propagation.

4.2.3 Numerical quadrature of a Gauss-von Mises distribution

A numerical quadrature method is essential for propagating uncertainty in nonlinear systems, in particular within the framework of the UKF. Similarly to the full Gaussian quadrature (cf. subsection §2.4.1), the GvM quadrature seeks an approximation of a moment integral by a sum of $N = 2d + 1$ (with $d = 1 + \ell$ in this case) weighted points:

$$\mathbb{E}[\mathrm{g}(\theta, \boldsymbol{x})] := \int_{\mathbb{R}^\ell} \int_{-\pi}^{\pi} \mathrm{g}(\theta, \boldsymbol{x})\, \mathcal{GVM}(\theta, \boldsymbol{x}) d\theta d\boldsymbol{x} \approx \sum_{0=1}^{2d} W^{(i)} \mathrm{g}(\theta^{(i)}, \boldsymbol{x}^{(i)}) \tag{4.16}$$

The goal of the GvM quadrature is to find nodes and weights such that (4.16) is exact for a well-chosen function $\mathrm{g}(\theta, \boldsymbol{x})$. Without loss of generality, finding these nodes and weights can be solved by turning to the canonical GvM (4.13) as the weighting function:

$$\int_{\mathbb{R}^\ell} \int_{-\pi}^{\pi} \mathrm{g}(\phi, \boldsymbol{z})\, \mathcal{N}(\boldsymbol{z};\, \mathbf{0}, \mathbf{I})\, \mathcal{VM}(\phi;\, 0, \kappa)\, d\phi\, d\boldsymbol{z} \approx \sum_{i=0}^{2d} W^{(i)} \mathrm{g}(\phi^{(i)}, \boldsymbol{z}^{(i)}) \tag{4.17}$$

Indeed, after solving this quadrature problem, the nodes $(\theta^{(i)}, \boldsymbol{x}^{(i)})$ can be recovered from $(\phi^{(i)}, \boldsymbol{z}^{(i)})$ thanks to the inverse transformations (4.12).

In what follows, the set of nodes given in Table 4.1 will be required.

i	0	1	2	$\cdots$	d	$d+1$	$d+2$	$\cdots$	$2d$
$\phi^{(i)}$	0	$-\eta$	0	$\cdots$	0	$+\eta$	0	$\cdots$	0
$\boldsymbol{z}^{(i)}$	0	0	$-\alpha$	$\cdots$	0	0	$+\alpha$	0	0
	0	0	0	$\ddots$	$\vdots$	0	0	$\ddots$	$\vdots$
	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$-\alpha$	$\vdots$	$\vdots$	$\ddots$	$+\alpha$
$W^{(i)}$	$W^{(0)}$	$W^{(\eta)}$	$W^{(\alpha)}$	$\cdots$	$W^{(\alpha)}$	$W^{(\eta)}$	$W^{(\alpha)}$	$\cdots$	$W^{(\alpha)}$

Table 4.1: Set of $2d + 1$ (unspecified) nodes and weights for the state space $\mathbb{S} \times \mathbb{R}^\ell$. These are all symmetric nodes centered around zero. The spread in ϕ is given by $\pm\eta$, and the spread in z_j is given by $\pm\alpha$, $\forall j = 1, \ldots, \ell$.

We denote $W^{(0)}$ the weight associated with the zero$^{\text{th}}$ node, $W^{(\eta)}$ the weight associated with the nodes indexed by $i = 1$ and $i = d + 1$, and we denote $W^{(i)} = W^{(\alpha)}$ the weight associated with the remaining nodes ($\forall i \neq \{0, 1, d + 1\}$). In addition, it is required that the approximation (4.17) is exact for any function $\mathrm{g}(\phi, \boldsymbol{z})$ of the form

$$\mathrm{g}(\phi, \boldsymbol{z}) = a_0 + \frac{1}{2}\boldsymbol{z}^\top \mathbf{B}_0 \boldsymbol{z} + a_1 \cos(\phi) + a_2 \cos(2\phi) + \mathrm{g}_0(\phi, \boldsymbol{z}) \tag{4.18}$$

where $a_0, a_1, a_2 \in \mathbb{R}$, $\mathbf{B}_0$ is any $\ell \times \ell$ symmetric matrix, and g_0 is any function such that $\mathrm{g}_0(\phi, \boldsymbol{z}) = -\mathrm{g}_0(-\phi, \boldsymbol{z})$ or $\mathrm{g}_0(\phi, \boldsymbol{z}) = -\mathrm{g}_0(\phi, -\boldsymbol{z})$. The above function takes the role of an orthogonal polynomial, allowing us to solve the quadrature problem.

The values of the nodes and weights of Table 4.1 are found by establishing moment constraints from the quadrature formula (4.17) combined with a specific choice of the function $\mathrm{g}(\phi, \boldsymbol{z})$ taking value at the point $(\phi^{(i)}, \boldsymbol{z}^{(i)})$. For that, the low-order moments of the canonical GvM, given by (4.14) and (4.15), are useful since the quadrature problem simply boils down to a moment-matching problem. This takes the form of five moment constraints:

- **Constraint on the weights**: This ensures that the point distribution is that of a probability density. The constraint is given by

$$\int_{\mathbb{R}^\ell} \int_{-\pi}^{\pi} \mathcal{N}(\boldsymbol{z};\, \mathbf{0}, \mathbf{I})\, \mathcal{VM}(\phi;\, 0, \kappa)\, d\phi\, d\boldsymbol{z} = \sum_{i=0}^{2d} W^{(i)} = 1. \tag{4.19}$$

- **Constraint on the covariance**: This ensures that the set of points $\{\boldsymbol{z}^{(i)}\}_{\forall i}$ matches the covariance $\mathbb{E}[\boldsymbol{z}\boldsymbol{z}^\top] = \mathbf{I}$. It is given by

$$\int_{\mathbb{R}} z_j^2\, \mathcal{N}(z_j;\, 0, 1) dz_j = \sum_{i=0}^{2d} W^{(i)} (z_j^{(i)})^2 = 1\,, \quad \forall j = 1, \ldots, \ell. \tag{4.20}$$

- **Constraint on the kurtosis**: This ensures that the set of points $\{z_j^{(i)}\}_{i=0,\ldots,2d}$ matches the kurtosis of a canonical Gaussian ($\forall j = 1, \ldots, \ell$).

$$\int_{\mathbb{R}} z_j^4\, \mathcal{N}(z_j;\, 0, 1) dz_j = \sum_{i=0}^{2d} W^{(i)} (z_j^{(i)})^4 = 3\,, \quad \forall j = 1, \ldots, \ell. \tag{4.21}$$

- **Constraint on the first circular moment**: This ensures that the set $\{\phi^{(i)}\}_{\forall i}$ matches the first circular moment $\mathbb{E}[e^{i\phi}]$.

$$\int_{-\pi}^{\pi} \cos(\phi)\, \mathcal{VM}(\phi;\, 0, \kappa)\, d\phi = \sum_{i=0}^{2d} W^{(i)} \cos(\phi^{(i)}) = \frac{I_1(\kappa)}{I_0(\kappa)}. \tag{4.22}$$

- **Constraint on the second circular moment**: This ensures that the set $\{\phi^{(i)}\}_{\forall i}$ matches the second circular moment $\mathbb{E}[e^{i2\phi}]$.

$$\int_{-\pi}^{\pi} \cos(2\phi)\, \mathcal{VM}(\phi;\, 0, \kappa)\, d\phi = \sum_{i=0}^{2d} W^{(i)} \cos(2\phi^{(i)}) = \frac{I_2(\kappa)}{I_0(\kappa)}. \tag{4.23}$$

These five moment constraints are compactly summarized in Table 4.2.

$\mathrm{g}(\phi, \boldsymbol{z})$	Constraint
1	(4.19) $\Rightarrow W^{(0)} + 2W^{(\eta)} + 2\ell W^{(\alpha)} = 1$
$z_j^2,\ \forall j = 1, \ldots, \ell$	(4.20) $\Rightarrow 2\alpha^2 W^{(\alpha)} = 1$
$z_j^4,\ \forall j = 1, \ldots, \ell$	(4.21) $\Rightarrow 2\alpha^4 W^{(\alpha)} = 3$
$\cos(\phi)$	(4.22) $\Rightarrow W^{(0)} + 2\cos(\eta) W^{(\eta)} + 2\ell W^{(\alpha)} = \frac{I_1(\kappa)}{I_0(\kappa)}$
$\cos(2\phi)$	(4.23) $\Rightarrow W^{(0)} + 2\cos(2\eta) W^{(\eta)} + 2\ell W^{(\alpha)} = \frac{I_2(\kappa)}{I_0(\kappa)}$

Table 4.2: Constraints on the nodes and weights [Horwood and Poore, 2014].

Solving the system of equations in Table 4.2 allows us to obtain the nodes and weights of the canonical GvM distribution, whose solutions are given by:

$$\begin{aligned} &\alpha = \sqrt{3}, && W^{(\alpha)} = \frac{1}{6} \\ &\eta = \arccos\left(\frac{B_2(\kappa)}{2B_1(\kappa)} - 1\right), && W^{(\eta)} = \frac{B_1(\kappa)^2}{4B_1(\kappa) - B_2(\kappa)} \\ & && W^{(0)} = 1 - 2W^{(\eta)} - 2\ell W^{(\alpha)} \\ &\text{where,} \quad B_\nu(\kappa) = 1 - \frac{I_\nu(\kappa)}{I_0(\kappa)} \end{aligned} \tag{4.24}$$

Notice that the spreading parameter α and its associated weight $W^{(\alpha)}$ are exactly those obtained from a Gauss-Hermite quadrature rule. Indeed, the moment-matching corresponding to the Gaussian part of the GvM satisfies the conditions obtained from the Gauss-Hermite rule for a centered-normalized Gaussian (as already seen in Chapter 2 subsection §2.4.1). In summary, the set of nodes and weights listed in Table 4.1, and specified by the expressions in (4.24) are those of the canonical GvM. Then, to obtain the nodes $(\theta^{(i)}, \boldsymbol{x}^{(i)})$ of the GvM, the canonical points $(\phi^{(i)}, \boldsymbol{z}^{(i)})$ are transformed by the inverse of the functions (4.12). Finally, these sigma-points can be propagated into any propagation function so that we can proceed to the computation of propagated statistical properties.

4.3 Development of an equivariant von Mises-Gaussian distribution for $\boldsymbol{SE}(n)$

Building upon the previous foundations, this section is devoted to the development of a probability density model on $\boldsymbol{SE}(n)$ that preserves the practical advantages of exponential coordinates while remaining fully compatible with a UKF-LG framework.

4.3.1 Proposed method

The proposed methodology is to extend the equivariant Gaussian on $\boldsymbol{SE}(n)$ toward an equivariant distribution that incorporates a von Mises distribution in its design. Recall that the equivariant distribution framework is given by

$$\boldsymbol{\chi} = \hat{\boldsymbol{\chi}} \exp(\boldsymbol{\xi}^\wedge) \ \text{(Left)} \quad \text{or} \quad \boldsymbol{\chi} = \exp(\boldsymbol{\xi}^\wedge)\hat{\boldsymbol{\chi}} \ \text{(Right)} \tag{4.25}$$

where $\hat{\boldsymbol{\chi}}$ denotes the mean value, and all of the dispersion is carried by the random vector $\boldsymbol{\xi} = [\boldsymbol{\omega}^\top, \boldsymbol{\rho}^\top]^\top$. We propose to keep this framework, but instead of considering $\boldsymbol{\xi}$ to be fully Gaussian-distributed, we propose to model its uncertainty with a hybrid formulation, using a von Mises on the one hand and a Gaussian on the other hand. This formulation is a well-suited candidate for the Lie group $\boldsymbol{SE}(n)$, as it involves coupled rotational and translational variables. Moreover, the use of the von Mises distribution avoids relying exclusively on covariance by introducing concentration parameters as complementary measures of dispersion.

To do that, instead of considering the angular component ω_i $(\forall i)$ as an unbounded variable in $\mathbb{R}$, it could be better represented by considering it as a bounded variable in $\mathbb{S}$. Hence, instead of $\boldsymbol{\xi} \in \mathbb{R}^d$, we rather consider

$$\boldsymbol{\xi} = \begin{bmatrix} \boldsymbol{\omega} \\ \boldsymbol{\rho} \end{bmatrix} \in \mathbb{S}^m \times \mathbb{R}^\ell \subset \mathbb{R}^d \tag{4.26}$$

such that $m+\ell = d$, with m being the number of angular variables and ℓ is the number of translational variables. It follows that developing an equivariant distribution for $\boldsymbol{SE}(n)$ starts by defining a distribution on $\mathbb{S}^m \times \mathbb{R}^\ell$. Thus, we propose a distribution of the form,

$$\mathcal{P}(\boldsymbol{\xi}) = \mathcal{P}(\boldsymbol{\omega}, \boldsymbol{\rho}) = \text{von Mises}(\boldsymbol{\omega}) \times \text{Gaussian}(\boldsymbol{\rho}|\boldsymbol{\omega}). \tag{4.27}$$

This distribution, formalized in the vector space that is isomorphic to the Lie algebra, must satisfy two requirements:

(a) The mean of $\boldsymbol{\xi}$ must be zero ($\mathbb{E}[\boldsymbol{\xi}] = \mathbf{0}$) so that the distribution embedded in the group via the exponential map has its mean at the identity element.

(b) The correlation between $\boldsymbol{\omega}$ and $\boldsymbol{\rho}$ must be taken into account, since they are not decoupled but intrinsically related through the underlying structure of the group.

These requirements are essential to ensure that the equivariant framework (4.25) remains applicable.

4.3.2 von Mises-Gaussian on $\mathbb{S}^m \times \mathbb{R}^\ell$

We would like to construct a von Mises-Gaussian distribution for this state space in the same way it has been done for the GvM of Horwood. The common background begins with the joint pdf (4.6), which can be decomposed into (4.7). Unlike [Horwood and Poore, 2014], we chose to take $\boldsymbol{a} = \boldsymbol{\omega}$ $(n_a = m)$ and $\boldsymbol{b} = \boldsymbol{\rho}$ $(n_b = \ell)$ to obtain

$$\begin{aligned} \mathcal{P}(\boldsymbol{\omega}, \boldsymbol{\rho}) &= \underbrace{\mathcal{N}(\boldsymbol{\omega}; \overline{\boldsymbol{\omega}}, \mathbf{P}_{\omega\omega})}_{\mathcal{P}(\boldsymbol{\omega})} \underbrace{\mathcal{N}(\boldsymbol{\rho}|\boldsymbol{\omega}; \tilde{\boldsymbol{\rho}}(\boldsymbol{\omega}), \tilde{\mathbf{P}}_{\rho\rho})}_{\mathcal{P}(\boldsymbol{\rho}|\boldsymbol{\omega})} \\ \tilde{\boldsymbol{\rho}}(\boldsymbol{\omega}) &= \overline{\boldsymbol{\rho}} + \mathbf{P}_{\rho\omega}\mathbf{P}_{\omega\omega}^{-1}(\boldsymbol{\omega} - \overline{\boldsymbol{\omega}}), \quad \tilde{\mathbf{P}}_{\rho\rho} = \mathbf{P}_{\rho\rho} - \mathbf{P}_{\rho\omega}\mathbf{P}_{\omega\omega}^{-1}\mathbf{P}_{\rho\omega}^\top \end{aligned} \tag{4.28}$$

The reason for this choice comes from the direction of influence in the case of $\boldsymbol{SE}(n)$. Indeed, let us recall that if one wants to map the vector $\boldsymbol{\xi}$ into the group, we have

$$\exp(\boldsymbol{\xi}^\wedge) = \begin{pmatrix} \boldsymbol{\mathcal{R}}(\boldsymbol{\omega}) & \mathbf{J}_\mathsf{L}(\boldsymbol{\omega}) \times \boldsymbol{\rho} \\ \mathbf{0}_{1\times n} & 1 \end{pmatrix} \in \boldsymbol{SE}(n) \tag{4.29}$$

where $\mathbf{J}_\mathsf{L}(\boldsymbol{\omega})$ is the group jacobian of $\boldsymbol{SO}(n)$. Hence, the part corresponding to the position (top right of the matrix $\exp(\boldsymbol{\xi}^\wedge)$) has been directly influenced by the rotational component $\boldsymbol{\omega}$ via the group jacobian. This influence must be taken into account when building a probability density. In our case, we propose that this influence is represented through $\mathcal{P}(\boldsymbol{\rho}|\boldsymbol{\omega})$ with its conditional mean $\tilde{\boldsymbol{\rho}}(\boldsymbol{\omega})$ and conditional covariance $\tilde{\mathbf{P}}_{\rho\rho}$.

We then propose to consider $\overline{\boldsymbol{\omega}} = \mathbf{0}$ and $\overline{\boldsymbol{\rho}} = \mathbf{0}$ in (4.28), since we want our final distribution to be centered at zero. In addition, and similarly to [Horwood and Poore, 2014], we model the uncertainty of each angular variable ω_i $(i = 1, \dots, m)$ with a zero-mean univariate von Mises: $\omega_i \sim \mathcal{VM}(\omega_i; 0, \kappa_i)$. To extend it to the m-dimensional vector $\boldsymbol{\omega}$, we chose to model it by a product of uncorrelated von Mises:

$$\mathcal{P}(\boldsymbol{\omega}) = \prod_{i=1}^{m} \mathcal{VM}(\omega_i; 0, \kappa_i) \tag{4.30}$$

Finally, we assume that there exists a one-to-one correspondence[1] between κ_i and the variance $P_{\omega_i\omega_i}$, denoted by

$$\kappa_i = \vartheta(P_{\omega_i\omega_i}), \quad \forall i = 1, \dots, m \tag{4.31}$$

Since we assumed uncorrelated angular variables, we have to formalize a diagonal matrix $\mathbf{P}_{\boldsymbol{\omega}\boldsymbol{\omega}}$ as

$$\mathbf{P}_{\boldsymbol{\omega}\boldsymbol{\omega}} = \boldsymbol{\vartheta}^{-1}(\boldsymbol{\kappa}) = \begin{pmatrix} \vartheta^{-1}(\kappa_1) & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \vartheta^{-1}(\kappa_m) \end{pmatrix} \tag{4.32}$$

von Mises-Gaussian (vMG): In summary, we have the following pdf defined on $\mathbb{S}^m \times \mathbb{R}^\ell$:

$$\begin{aligned} &\mathcal{VMG}(\boldsymbol{\omega}, \boldsymbol{\rho}; \boldsymbol{\kappa}, \mathbf{M}, \tilde{\mathbf{P}}_{\boldsymbol{\rho}\boldsymbol{\rho}}) = \prod_{j=1}^{m} \mathcal{VM}(\omega_j; 0, \kappa_j) \times \mathcal{N}(\boldsymbol{\rho}|\boldsymbol{\omega}; \mathbf{M}\boldsymbol{\omega}, \tilde{\mathbf{P}}_{\boldsymbol{\rho}\boldsymbol{\rho}}) \\ &\text{with,} \quad \boldsymbol{\kappa} = [\kappa_1, ..., \kappa_m]^\top, \quad \mathbf{M} = \mathbf{P}_{\boldsymbol{\rho}\boldsymbol{\omega}}\mathbf{P}_{\boldsymbol{\omega}\boldsymbol{\omega}}^{-1}, \quad \tilde{\mathbf{P}}_{\boldsymbol{\rho}\boldsymbol{\rho}} = \mathbf{P}_{\boldsymbol{\rho}\boldsymbol{\rho}} - \mathbf{M}\mathbf{P}_{\boldsymbol{\omega}\boldsymbol{\omega}}\mathbf{M}^\top \\ &\text{given that we have} \quad \mathbf{P} = \begin{pmatrix} \mathbf{P}_{\boldsymbol{\omega}\boldsymbol{\omega}} & \mathbf{P}_{\boldsymbol{\rho}\boldsymbol{\omega}}^\top \\ \mathbf{P}_{\boldsymbol{\rho}\boldsymbol{\omega}} & \mathbf{P}_{\boldsymbol{\rho}\boldsymbol{\rho}} \end{pmatrix} \end{aligned} \tag{4.33}$$

This uncertainty model is characterized by:

- An rotational dispersion parameter $\boldsymbol{\kappa} = [\kappa_1, \dots, \kappa_m]^\top$.
- A coupling parameter $\mathbf{M} = \mathbf{P}_{\boldsymbol{\rho}\boldsymbol{\omega}}\mathbf{P}_{\boldsymbol{\omega}\boldsymbol{\omega}}^{-1}$.
- A translational dispersion parameter $\tilde{\mathbf{P}}_{\boldsymbol{\rho}\boldsymbol{\rho}} = \mathbf{P}_{\boldsymbol{\rho}\boldsymbol{\rho}} - \mathbf{M}\mathbf{P}_{\boldsymbol{\omega}\boldsymbol{\omega}}\mathbf{M}^\top$.

Remark 4.1. Notice that we still need the full covariance $\mathbf{P}$ in the proposed model although we argued that the notion of covariance may not be appropriate for angular variables. In fact, we still need to retain the angular part $\mathbf{P}_{\boldsymbol{\omega}\boldsymbol{\omega}}$ for practical reasons. Firstly, because we modeled the uncertainty of $\boldsymbol{\rho}$ with a Gaussian of covariance $\tilde{\mathbf{P}}_{\boldsymbol{\rho}\boldsymbol{\rho}}$ (which depends on cross-covariance $\mathbf{P}_{\boldsymbol{\rho}\boldsymbol{\omega}}$ and angular covariance $\mathbf{P}_{\boldsymbol{\omega}\boldsymbol{\omega}}$), but also to preserve the role of $\mathbf{P}$ within the UKF-LG framework.

Remark 4.2. Regarding the dependency of $\boldsymbol{\omega}$ on $\boldsymbol{\rho}$, the chosen model with mean $\tilde{\boldsymbol{\rho}}(\boldsymbol{\omega}) = \mathbf{M}\boldsymbol{\omega}$ and covariance $\tilde{\mathbf{P}}_{\boldsymbol{\rho}\boldsymbol{\rho}} = \mathbf{P}_{\boldsymbol{\rho}\boldsymbol{\rho}} - \mathbf{M}\mathbf{P}_{\boldsymbol{\omega}\boldsymbol{\omega}}\mathbf{M}^\top$ is, in reality, only valid and exact for a joint pdf

[1] This correspondance does not have a known analytical formula, but it is possible to obtain a numerical approximation by interpolation.

that is initially Gaussian (i.e., $\mathcal{P}(\boldsymbol{\omega}, \boldsymbol{\rho}) = \mathcal{N}(\boldsymbol{\omega}, \boldsymbol{\rho})$). That is why the decomposition (4.28) is allowed. In general, this decomposition is not allowed for any joint pdf $\mathcal{P}(\boldsymbol{\omega}, \boldsymbol{\rho})$, and another dependency model must be found between $\boldsymbol{\rho}$ and $\boldsymbol{\omega}$. However, we did not work on it and chose to retain this linear dependency as a first approximation. By intuition, this approximation should be reasonably valid at the tangent space level, but further analysis must be done to prove it, which lies beyond the scope of this thesis.

Differences with the GvM distribution of [Horwood and Poore, 2014]:

In terms of our notations, the GvM distribution writes

$$\mathcal{GVM}(\boldsymbol{\rho}, \omega;\ \overline{\boldsymbol{\rho}}, \mathbf{P}_{\rho\rho}, \overline{\omega}, \boldsymbol{\beta}, \boldsymbol{\Gamma}, \kappa) = \mathcal{VM}(\omega; \Theta(\boldsymbol{\rho}), \kappa) \times \mathcal{N}(\boldsymbol{\rho}; \overline{\boldsymbol{\rho}}, \mathbf{P}_{\rho\rho})$$

where $\Theta(\boldsymbol{\rho}) = \overline{\omega} + \boldsymbol{\beta}^\top \boldsymbol{z} + \frac{1}{2}\boldsymbol{z}^\top \boldsymbol{\Gamma} \boldsymbol{z}$, $\boldsymbol{z} = \mathbf{A}^{-1}(\boldsymbol{\rho} - \overline{\boldsymbol{\rho}})$, and $\mathbf{A}$ is such that $\mathbf{P}_{\rho\rho} = \mathbf{A}\mathbf{A}^\top$. We can see several differences with our proposed vMG:

(1) The vMG models the uncertainty on the manifold $\mathbb{S}^m \times \mathbb{R}^\ell$ rather than on $\mathbb{S} \times \mathbb{R}^\ell$. This allows for an extended uncertainty representation when the state space includes multiple angular variables.

(2) In the GvM, the mean of ω is conditioned by the translational vector $\boldsymbol{\rho}$ through the function $\Theta(\boldsymbol{\rho})$, which involved the coupling parameters $\boldsymbol{\beta}$ and $\boldsymbol{\Gamma}$. In our proposed vMG, it is the mean of $\boldsymbol{\rho}$ that is conditioned by the rotational vector $\boldsymbol{\omega}$ through the linear dependency $\mathbf{M}\boldsymbol{\omega}$, with $\mathbf{M}$ being the unique coupling parameter.

(3) The parameter $\boldsymbol{\Gamma}$ in the GvM allows to capture nonzero higher-order moments of the distribution. Our vMG does not possess such feature since we did not express the need to capture any higher-order moments beyond the mean, the covariance, the 1st and the 2nd circular moments.

4.3.3 Equivariant von Mises-Gaussian on $\boldsymbol{SE}(n)$

At this stage, we have proposed an uncertainty model for the zero-mean random vector $\boldsymbol{\xi} = [\boldsymbol{\omega}^\top, \boldsymbol{\rho}^\top]^\top \in \mathbb{S}^m \times \mathbb{R}^\ell$, defined in the tangent space at the identity of the group $\boldsymbol{SE}(n)$, such that:

$$\boldsymbol{\xi} \sim \mathcal{VMG}(\boldsymbol{\omega}, \boldsymbol{\rho}\,;\ \boldsymbol{\kappa}, \mathbf{M}, \tilde{\mathbf{P}}_{\rho\rho}) \tag{4.34}$$

where $\mathcal{VMG}(\boldsymbol{\omega}, \boldsymbol{\rho}\,;\ \boldsymbol{\kappa}, \mathbf{M}, \tilde{\mathbf{P}}_{\rho\rho})$ is given by (4.33). Since the full covariance $\mathbf{P}$ contains the necessary terms to compute both $\mathbf{M}$ and $\tilde{\mathbf{P}}_{\rho\rho}$, we can simply denote

$$\boldsymbol{\xi} \sim \mathcal{VMG}(\boldsymbol{\xi};\ \mathbf{0}, \boldsymbol{\kappa}, \mathbf{P}), \tag{4.35}$$

where $\boldsymbol{\kappa}$ is the additional dispersion parameter, then formalize the equivariant distribution as given in the next page.

■ Final retained uncertainty model

Finally, considering that $\boldsymbol{\xi}^\wedge$ lives in the Lie algebra, we can apply the mapping into the group via exponentiation, followed by a (left/right) action to produce a (left/right) "equivariant vMG" for $\boldsymbol{\chi} \in \boldsymbol{SE}(n)$, which we denote by $\boldsymbol{\chi} \sim \mathcal{VMG}_{\mathsf{L/R}}(\boldsymbol{\chi}; \hat{\boldsymbol{\chi}}, \boldsymbol{\kappa}, \mathbf{P})$:

Equivariant von Mises-Gaussian on $\boldsymbol{SE}(n)$:

$$\begin{cases} \text{(Left) } \boldsymbol{\chi} = \mathsf{L}_{\hat{\boldsymbol{\chi}}}(\exp(\boldsymbol{\xi}^\wedge)) = \hat{\boldsymbol{\chi}} \exp(\boldsymbol{\xi}^\wedge) \sim \mathcal{VMG}_{\mathsf{L}}(\boldsymbol{\chi}; \hat{\boldsymbol{\chi}}, \boldsymbol{\kappa}, \mathbf{P}) \\ \text{(Right) } \boldsymbol{\chi} = \mathsf{R}_{\hat{\boldsymbol{\chi}}}(\exp(\boldsymbol{\xi}^\wedge)) = \exp(\boldsymbol{\xi}^\wedge) \hat{\boldsymbol{\chi}} \sim \mathcal{VMG}_{\mathsf{R}}(\boldsymbol{\chi}; \hat{\boldsymbol{\chi}}, \boldsymbol{\kappa}, \mathbf{P}) \end{cases} \tag{4.36}$$
$$\text{with,} \quad \boldsymbol{\xi} = [\boldsymbol{\omega}^\top, \boldsymbol{\rho}^\top]^\top \sim \mathcal{VMG}(\boldsymbol{\xi}; \mathbf{0}, \boldsymbol{\kappa}, \mathbf{P})$$

Remark 4.3. We have defined a vMG distribution on $\boldsymbol{\xi} \in \mathbb{S}^m \times \mathbb{R}^\ell$ specifically for the group $\boldsymbol{SE}(n)$, with $m + \ell = d = \dim(\boldsymbol{SE}(n))$. However, for most applications, it is common to include additional translational vectors $(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k)$ in the state space. In that case, the extended group $\boldsymbol{SE}_k(n)$ can be used, and the uncertainty can still be modeled using the left or right equivariant framework with $\boldsymbol{\xi} = [\boldsymbol{\omega}, \boldsymbol{\rho}_1, \ldots, \boldsymbol{\rho}_k]^\top$, where each $\boldsymbol{\rho}_i \in \mathbb{R}^n$. Incorporating a vMG distribution into $\boldsymbol{SE}_k(n)$ essentially amounts to increasing the number of translational variables, and define the vMG for $\boldsymbol{\xi} \in \mathbb{S}^m \times \mathbb{R}^{nk}$ (without any further modifications), with $m + nk = \dim(\boldsymbol{SE}_k(n))$. Moreover, the mixed case $\mathcal{M} = \boldsymbol{SE}_k(n) \times \mathbb{R}^N$, where additional state variables are added outside the group structure (as in [Brossard et al., 2020b]), is also compatible. In this case, the variables in $\mathbb{R}^N$ are modeled as Gaussians, and the vMG can be defined for $\boldsymbol{\xi} \in \mathbb{S}^m \times \mathbb{R}^{nk+N}$ (also without any further modifications), with $m + nk + N = \dim(\boldsymbol{SE}_k(n)) + N$.

The only difference with an equivariant Gaussian is that the equivariant distribution (4.36) is redefined in the tangent space level by using the von Mises-Gaussian pdf (4.33). This reformulation introduces additional dispersion characteristics through the concentration parameters associated with the angular variables.

Consequently, if one wants to apply this uncertainty model on a Lie group-based UKF, sigma-points must be generated from moment-matching, using the statistical parameters $\boldsymbol{\kappa}$ and $\mathbf{P}$. Hence, the sigma-points $\{\boldsymbol{\omega}^{(j)}\}_j$ must be built so as to match the first circular moment $\mathbb{E}[e^{i\boldsymbol{\omega}}]$ and the second circular moment $\mathbb{E}[e^{i2\boldsymbol{\omega}}]$ given by

$$\mathbb{E}[e^{i\boldsymbol{\omega}}] = \left[\frac{I_1(\kappa_1)}{I_0(\kappa_1)}, \quad \ldots \quad, \frac{I_1(\kappa_m)}{I_0(\kappa_m)}\right]^\top, \qquad \mathbb{E}[e^{i2\boldsymbol{\omega}}] = \left[\frac{I_2(\kappa_1)}{I_0(\kappa_1)}, \quad \ldots \quad, \frac{I_2(\kappa_m)}{I_0(\kappa_m)}\right]^\top. \tag{4.37}$$

Clearly, knowing the full definition of the p^{th} circular moment (4.5), this signifies a sample mean that complies with the zero-mean direction $\mathbb{E}[\omega_i] = \overline{\omega}_i = 0$ $(\forall i = 1, \ldots, m)$, and a sample concentration parameter that complies with the concentration parameter at disposal κ_i. Similarly, the sigma-points $\{\boldsymbol{\rho}^{(j)}\}_j$ are required to satisfy $\mathbb{E}[\boldsymbol{\rho}] = \mathbf{0}$ and $\tilde{\mathbf{P}}_{\boldsymbol{\rho}\boldsymbol{\rho}}$. This construction leads to a reformulation of the Unscented Transform and, by extension, to an adaptation of the UKF-LG that incorporates this new uncertainty model within its prediction and correction scheme, which we detail in the next section.

4.4 Adaptation of the UKF-LG to the von Mises-Gaussian distribution

The objective is to include the proposed probability density into a UKF-LG framework. The first step is to design a procedure for generating weighted sigma-points that are consistent with the statistical parameters of the vMG distribution of $\boldsymbol{\xi} \in \mathbb{S}^m \times \mathbb{R}^\ell$, namely the full covariance $\mathbf{P}$ and the vector of concentration parameters $\boldsymbol{\kappa}$.

4.4.1 Sigma-points and weights generation

The generation of these weighted sigma-points follows a vMG quadrature problem, which is defined as finding $N+1$ nodes and weights such that the following quadrature formula is exact for any well-chosen function $\mathrm{g}(\boldsymbol{\omega}, \boldsymbol{\rho})$:

$$\begin{aligned}&\int_{\mathbb{R}^\ell}\int_{\mathbb{S}^m} \mathrm{g}(\boldsymbol{\omega},\boldsymbol{\rho})\,\mathcal{VMG}(\boldsymbol{\omega},\boldsymbol{\rho};\,\boldsymbol{\kappa},\mathbf{M},\mathbf{P})d\boldsymbol{\omega}d\boldsymbol{\rho}\\ &= \int_{\mathbb{R}^\ell}\int_{\mathbb{S}^m} \mathrm{g}(\boldsymbol{\omega},\boldsymbol{\rho})\prod_{i=1}^{m}\mathcal{VM}(\omega_i;0,\kappa_i)\,\mathcal{N}(\boldsymbol{\rho}|\boldsymbol{\omega};\mathbf{M}\boldsymbol{\omega},\tilde{\mathbf{P}}_{\boldsymbol{\rho}\boldsymbol{\rho}})d\boldsymbol{\omega}d\boldsymbol{\rho} \approx \sum_{j=0}^{N} W^{(j)}\,\mathrm{g}(\boldsymbol{\omega}^{(j)},\boldsymbol{\rho}^{(j)})\end{aligned} \tag{4.38}$$

Without loss of generality, finding the quadrature nodes and weights can be performed exclusively by working on the canonical form of the vMG, given by:

$$\begin{aligned}&\mathcal{VMG}(\boldsymbol{\omega},\boldsymbol{z};\boldsymbol{\kappa},\mathbf{I}_{\ell\times\ell}) = \prod_{i=1}^{m}\mathcal{VM}(\omega_i;0,\kappa_i)\,\mathcal{N}(\boldsymbol{z};\mathbf{0},\mathbf{I}_{\ell\times\ell})\\ &\text{with}\quad \boldsymbol{z} = \mathbf{S}_{\boldsymbol{\rho}\boldsymbol{\rho}}^{-1}(\boldsymbol{\rho} - \mathbf{S}_{\boldsymbol{\rho}\boldsymbol{\omega}}\mathbf{S}_{\boldsymbol{\omega}\boldsymbol{\omega}}^{-1}\boldsymbol{\omega})\end{aligned} \tag{4.39}$$

where we have $\mathbf{S} = [\mathbf{S}_{\boldsymbol{\omega}\boldsymbol{\omega}},\ \mathbf{0}_{m\times m}\ ;\ \mathbf{S}_{\boldsymbol{\rho}\boldsymbol{\omega}},\ \mathbf{S}_{\boldsymbol{\rho}\boldsymbol{\rho}}]$ such that $\mathbf{P} = \mathbf{S}\mathbf{S}^\top$. Moreover, the problem can be further simplified by separating the m angular variables, since they are assumed uncorrelated. Hence, we can solve the problem by focusing on a single angle ω, independently from the other angular variables, and find the sigma-points and weights for the pair $(\omega, \boldsymbol{z})$. Therefore, we have to find the $N+1$ weighted points such that the quadrature rule

$$\int_{\mathbb{R}^\ell}\int_{-\pi}^{\pi} \mathrm{g}(\omega,\boldsymbol{z})\mathcal{VM}(\omega;0,\kappa)\,\mathcal{N}(\boldsymbol{z};\mathbf{0},\mathbf{I}_{\ell\times\ell})d\omega d\boldsymbol{z} \approx \sum_{j=0}^{N} W^{(j)}\,\mathrm{g}(\omega^{(j)},\boldsymbol{z}^{(j)}) \tag{4.40}$$

is exact for any well-chosen function $\mathrm{g}(\omega, \boldsymbol{z})$. It turns out that this problem is exactly the same as the GvM quadrature presented in subsection §4.2.3, with $\mathrm{g}(\omega, \boldsymbol{z})$ being the polynomial given by (4.18). Indeed, the canonical form of the vMG distribution (with a single angle) is exactly identical to the canonical form of the GvM distribution (4.13). Consequently, the nodes and weights of the canonical GvM (Table 4.1) specified by the formulas (4.24), can be used exactly as the set of nodes and weights of the canonical vMG for the pair $(\omega_i, \boldsymbol{z})$, with the concentration parameter κ_i $(\forall i = 1, \ldots, m)$.

In summary, let us denote by $\boldsymbol{\zeta}^{(j)} = [\boldsymbol{\omega}^{(j)}, \boldsymbol{z}^{(j)}]^\top$ the j^{th} sigma-point associated with the canonical vMG. Then, the set of points $\{\boldsymbol{\zeta}^{(j)}\}_{j=0,\dots,2d}$ that captures the distribution of $[\boldsymbol{\omega}^\top, \boldsymbol{z}^\top]^\top \sim \mathcal{VMG}(\boldsymbol{\omega}, \boldsymbol{z}; \boldsymbol{\kappa}, \mathbf{I}_{\ell\times\ell})$, is computed as follows:

$$\begin{aligned} \boldsymbol{\zeta}^{(0)} &= \mathbf{0} \\ \forall j = 1,\dots,d: \quad \boldsymbol{\zeta}^{(j)} &= -\epsilon_j \times [\mathbf{I}_{d\times d}]_j \\ \forall j = 1,\dots,d: \quad \boldsymbol{\zeta}^{(d+j)} &= +\epsilon_{d+j} \times [\mathbf{I}_{d\times d}]_j \end{aligned} \tag{4.41}$$

where $[\cdot]_j$ denotes the j^{th} row of the matrix under consideration, and $d = m + \ell$. The scaling parameters $\epsilon_j > 0$ are given by

$$\begin{aligned} \forall j = 1,\dots,m: \quad \epsilon_j = \epsilon_{d+j} &= \arccos\left(\frac{B_2(\kappa_j)}{2B_1(\kappa_j)} - 1\right) \\ \forall j = m+1,\dots,d: \quad \epsilon_j = \epsilon_{d+j} &= \sqrt{3} \end{aligned} \tag{4.42}$$

with $B_\nu(\kappa) = 1 - \frac{I_\nu(\kappa)}{I_0(\kappa)}$. Finally, each sigma-point $\boldsymbol{\zeta}^{(j)}$ is associated to a weight $W^{(j)}$ computed as follows

$$\begin{aligned} \forall j = 1,\dots,m: \quad W^{(j)} = W^{(d+j)} &= \frac{B_1(\kappa_j)^2}{4B_1(\kappa_j) - B_2(\kappa_j)} \\ \forall j = m+1,\dots,d: \quad W^{(j)} = W^{(d+j)} &= \frac{1}{6} \\ W^{(0)} &= 1 - \sum_{j=1}^{2d} W^{(j)} \end{aligned} \tag{4.43}$$

NB: To avoid numerical problems, the ratios of Bessel functions $B_\nu(\kappa)$ can be numerically approximated following the method provided in [Kurz et al., 2013].

After that, we recover the sigma-points associated with $[\boldsymbol{\omega}^\top, \boldsymbol{\rho}^\top]^\top \sim \mathcal{VMG}(\boldsymbol{\omega}, \boldsymbol{\rho}; \boldsymbol{\kappa}, \mathbf{P})$ by applying the transformation from $\boldsymbol{z}$ to $\boldsymbol{\rho}$ thanks to the following heuristic:

$$\boldsymbol{\rho}^{(j)} = \text{Chol}(\mathbf{P}_{\rho\rho} + \underbrace{\mathbf{P}_{\rho\omega}\mathbf{P}_{\omega\omega}^{-1}\mathbf{P}_{\rho\omega}^\top}_{\mathbf{M}\mathbf{P}_{\omega\omega}\mathbf{M}^\top})_j \, \boldsymbol{z}^{(j)}, \quad \forall j = 0,\dots,2d \tag{4.44}$$

where $\text{Chol}(\cdot)$ is the Cholesky factorization[2] of the argument.

Remark 4.4. Normally, the transformation that constructs a random vector $\boldsymbol{\rho}$ correlated with $\boldsymbol{\omega}$ is:

$$\boldsymbol{\rho} = \mathbf{S}_{\rho\rho}\,\boldsymbol{z} + \mathbf{S}_{\rho\omega}\mathbf{S}_{\omega\omega}^{-1}\,\boldsymbol{\omega} \tag{4.45}$$

The heuristic (4.44) does not exactly correspond to (4.45). Firstly, because we do not manipulate random vectors but a set of deterministically chosen points. Secondly, because any method other than the proposed heuristic made the UKF diverge. Somehow, applying (4.44) was the only stable way to take correlations into account while preserving the numerical stability of the whole procedure. We can argue that (4.44) is the correct numerical procedure when we try to correlate a set of deterministically chosen sigma-points with another.

[2]For implementation purposes, it might be preferable to use the rank 1 update to Cholesky factorization "cholupdate" when using the SR-UKF framework whose general methodology was first introduced in [Van der Merwe and Wan, 2001].

4.4.2 Application to the Unscented transform on Lie groups

The constructed sigma-points can then be applied to the Unscented transformation into the system dynamics to reconstruct the corresponding propagated statistical parameters. Consider again the model $\boldsymbol{f} = \boldsymbol{\chi}_{k-1}\boldsymbol{U}_{k-1}$ but now assume that both the state and input are distributed with respect to a (left/right) equivariant-vMG:

$$\boldsymbol{\chi}_k \sim \mathcal{VMG}_{\mathsf{L/R}}(\hat{\boldsymbol{\chi}}_{k|l}, \boldsymbol{\kappa}_{k|l}, \mathbf{P}_{k|l}), \quad \boldsymbol{U}_k \sim \mathcal{VMG}_{\mathsf{L/R}}(\exp(\mathbf{u}_k^{\wedge}), \boldsymbol{\kappa}_{\mathbf{u}_k}, \mathbf{Q}_k).$$

The UT of the prediction step must be revised to account for these modifications. As a side note, we also assume that we know the correct noise parameters $\mathbf{Q}_k$ and $\boldsymbol{\kappa}_{\mathbf{u}_k}$ at all times[3].

■ **State augmentation for the prediction step**

Since the prediction function $\boldsymbol{f}$ contains internal noise $\mathbf{w}$, we incorporate it into the state space before generating points and weights. Let

$$\boldsymbol{\xi} = \begin{bmatrix} \boldsymbol{\omega} \\ \boldsymbol{\rho} \end{bmatrix} \quad \text{and} \quad \mathbf{w} = \begin{bmatrix} \mathbf{w}_{\boldsymbol{\omega}} \\ \mathbf{w}_{\boldsymbol{\rho}} \end{bmatrix} \tag{4.46}$$

At $k-1$ (before prediction), we know the concentration parameters of the angular states $\boldsymbol{\kappa}_{k-1|k-1}$, the full estimate covariance $\mathbf{P}_{k-1|k-1}$, the noise covariance $\mathbf{Q}_{k-1}$, and the concentration parameters of the angular input noise $\boldsymbol{\kappa}_{\mathbf{u}_{k-1}}$:

$$\boldsymbol{\kappa}_{k-1|k-1} = \begin{bmatrix} \kappa_1 \\ \vdots \\ \kappa_m \end{bmatrix}, \ \mathbf{P}_{k-1|k-1} = \begin{pmatrix} \mathbf{P}_{\boldsymbol{\omega}\boldsymbol{\omega}} & \mathbf{P}_{\boldsymbol{\rho}\boldsymbol{\omega}}^{\top} \\ \mathbf{P}_{\boldsymbol{\rho}\boldsymbol{\omega}} & \mathbf{P}_{\boldsymbol{\rho}\boldsymbol{\rho}} \end{pmatrix}, \ \mathbf{Q}_{k-1} = \begin{pmatrix} \mathbf{Q}_{\boldsymbol{\omega}\boldsymbol{\omega}} & \mathbf{Q}_{\boldsymbol{\rho}\boldsymbol{\omega}}^{\top} \\ \mathbf{Q}_{\boldsymbol{\rho}\boldsymbol{\omega}} & \mathbf{Q}_{\boldsymbol{\rho}\boldsymbol{\rho}} \end{pmatrix}, \ \boldsymbol{\kappa}_{\mathbf{u}_{k-1}} = \begin{bmatrix} \kappa_{u_1} \\ \vdots \\ \kappa_{u_m} \end{bmatrix} \tag{4.47}$$

Thus, the augmented concentration vector $\boldsymbol{\kappa}_{aug}$ is given by

$$\boldsymbol{\kappa}_{aug} = [\boldsymbol{\kappa}_{k-1|k-1}, \boldsymbol{\kappa}_{\mathbf{u}_{k-1}}]^{\top}, \tag{4.48}$$

and the augmented (canonical) random vector is

$$\boldsymbol{\zeta} = \begin{bmatrix} \boldsymbol{\omega} \\ \mathbf{w}_{\boldsymbol{\omega}} \\ \boldsymbol{z} \\ \mathbf{w}_{\boldsymbol{z}} \end{bmatrix} \sim \mathcal{VMG}([\boldsymbol{\omega}, \mathbf{w}_{\boldsymbol{\omega}}], [\boldsymbol{z}, \mathbf{w}_{\boldsymbol{z}}];\ \boldsymbol{\kappa}_{aug},\ \mathbf{I}_{2\ell\times 2\ell}) \quad \Rightarrow \quad \boldsymbol{\zeta}_{k-1|k-1}^{(j)} = \begin{bmatrix} \boldsymbol{\omega}_{k-1|k-1}^{(j)} \\ \boldsymbol{\omega}_{\mathbf{w}_{k-1}}^{(j)} \\ \boldsymbol{z}_{k-1|k-1}^{(j)} \\ \boldsymbol{z}_{\mathbf{w}_{k-1}}^{(j)} \end{bmatrix} \tag{4.49}$$

The sigma-point $\boldsymbol{\zeta}_{k-1|k-1}^{(j)}$ and its weight $W^{(j)}$ can be generated $\forall j$ using the expressions (4.41) to (4.43) with substitutions $m \leftarrow 2m$ and $\ell \leftarrow 2\ell$. To obtain the sigma-point $[\boldsymbol{\rho}_{k-1|k-1}^{(j)}, \boldsymbol{\rho}_{\mathbf{w}_{k-1}}^{(j)}]^{\top}$, we then apply the heuristic transformation:

$$\begin{bmatrix} \boldsymbol{\rho}_{k-1|k-1}^{(j)} \\ \boldsymbol{\rho}_{\mathbf{w}_{k-1}}^{(j)} \end{bmatrix} = \mathrm{Chol}\begin{pmatrix} \mathbf{P}_{\boldsymbol{\rho}\boldsymbol{\rho}} + \mathbf{P}_{\boldsymbol{\rho}\boldsymbol{\omega}}\mathbf{P}_{\boldsymbol{\omega}\boldsymbol{\omega}}^{-1}\mathbf{P}_{\boldsymbol{\rho}\boldsymbol{\omega}}^{\top} & \mathbf{0}_{\ell\times\ell} \\ \mathbf{0}_{\ell\times\ell} & \mathbf{Q}_{\boldsymbol{\rho}\boldsymbol{\rho}} + \mathbf{Q}_{\boldsymbol{\rho}\boldsymbol{\omega}}\mathbf{Q}_{\boldsymbol{\omega}\boldsymbol{\omega}}^{-1}\mathbf{Q}_{\boldsymbol{\rho}\boldsymbol{\omega}}^{\top} \end{pmatrix}_j \begin{bmatrix} \boldsymbol{z}_{k-1|k-1}^{(j)} \\ \boldsymbol{z}_{\mathbf{w}_{k-1}}^{(j)} \end{bmatrix} \quad \forall j \tag{4.50}$$

[3] In real-world scenarios, these parameters are guessed off-line, or computed on-line using approximation methods.

■ **Prediction step**

Let:

$$\boldsymbol{\xi}^{(j)}_{k-1|k-1} = \begin{bmatrix} \boldsymbol{\omega}^{(j)}_{k-1|k-1} \\ \boldsymbol{\rho}^{(j)}_{k-1|k-1} \end{bmatrix} \quad \text{and} \quad \mathbf{w}^{(j)}_{k-1} = \begin{bmatrix} \boldsymbol{\omega}^{(j)}_{\mathbf{w}_{k-1}} \\ \boldsymbol{\rho}^{(j)}_{\mathbf{w}_{k-1}} \end{bmatrix} \tag{4.51}$$

We can compute $\hat{\boldsymbol{\chi}}_{k|k-1}$ and $\mathbf{P}_{k|k-1}$ using the Unscented transform on Lie groups. In addition, we also have to compute the concentration parameters $\boldsymbol{\kappa}_{k|k-1}$. For that, we can use the resultant of the 1st circular mean, whose numerical approximation, $\forall i = 1, \ldots, m$, is

$$\overline{R}_i := \frac{I_1(\kappa_{i,k|k-1})}{I_0(\kappa_{i,k|k-1})} \approx \sqrt{\Big(\sum_{j=0} W^{(j)} \cos\left(\omega^{(j)}_{i,k|k-1}\right)\Big)^2 + \Big(\sum_{j=0} W^{(j)} \sin\left(\omega^{(j)}_{i,k|k-1}\right)\Big)^2} \tag{4.52}$$

Then, all that remains is to apply the reciprocal function $A^{-1}(\overline{R}_i)$ to extract the associated concentration parameter. However, it cannot be easily evaluated, and must rely on approximated methods [Hassan et al., 2012]. A reasonable approximation is proposed in [Dhillon and Sra, 2003]:

$$\kappa_{i,k|k-1} := A^{-1}(\overline{R}_i) \approx \frac{2\overline{R}_i - \overline{R}_i^3}{1 - \overline{R}_i^2}, \quad \forall i = 1, \ldots, m \tag{4.53}$$

In summary, the whole prediction step is summarized in Algorithm 4.

4.4.3 Reformulation of the UKF-LG

The goal of the revised UKF-LG is then to estimate the posterior mean pose and covariance of the estimate, together with the posterior concentration parameters as illustrated in Figure 4.4.

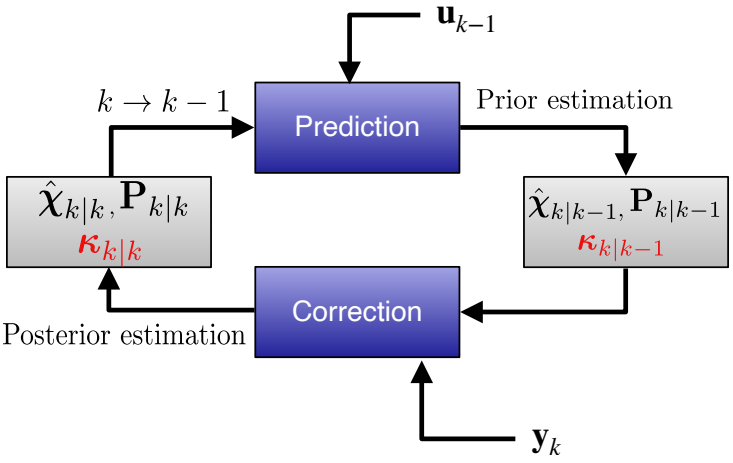


Figure 4.4: Prediction/Correction scheme of the revised UKF-LG.

The correction step of the revised UKF-LG does not differ much from what has been done for the prediction step, except that now we consider the observation model $\mathbf{h}$.

■ Correction step

The model $\mathbf{h}(\boldsymbol{\chi}, \mathbf{v})$ also contains internal noise $\mathbf{v} \in \mathbb{R}^v$. This time, we have

$$\mathbf{P}_{k|k-1} = \begin{pmatrix} \mathbf{P}_{\boldsymbol{\omega}\boldsymbol{\omega}} & \mathbf{P}_{\boldsymbol{\rho}\boldsymbol{\omega}}^\top \\ \mathbf{P}_{\boldsymbol{\rho}\boldsymbol{\omega}} & \mathbf{P}_{\boldsymbol{\rho}\boldsymbol{\rho}} \end{pmatrix} \quad \text{and} \quad \mathbf{v}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k) \tag{4.54}$$

and the augmented (canonical) random vector becomes

$$\boldsymbol{\zeta} = \begin{bmatrix} \boldsymbol{\omega} \\ \boldsymbol{z} \\ \mathbf{v}_{\boldsymbol{z}} \end{bmatrix} \sim \mathcal{VMG}(\boldsymbol{\omega}, [\boldsymbol{z}, \mathbf{v}_{\boldsymbol{z}}];\ \boldsymbol{\kappa}_{k|k-1}, \mathbf{I}_{(\ell+v)\times(\ell+v)}) \quad \Rightarrow \quad \boldsymbol{\zeta}_{k|k-1}^{(j)} = \begin{bmatrix} \boldsymbol{\omega}_{k|k-1}^{(j)} \\ \boldsymbol{z}_{k|k-1}^{(j)} \\ \boldsymbol{z}_{\mathbf{v}_k}^{(j)} \end{bmatrix} \tag{4.55}$$

Sigma-points and weights $\{\boldsymbol{\zeta}_{k|k-1}^{(j)}, W^{(j)}\}_{\forall j}$ of the canonical vMG can be generated using the expressions (4.41) to (4.43) with substitution $\ell \leftarrow \ell + v$. Then, the sigma-points $[\boldsymbol{\rho}_{k|k-1}^{(j)}, \mathbf{v}_k^{(j)}]^\top$ are obtained from the transformation:

$$\begin{bmatrix} \boldsymbol{\rho}_{k|k-1}^{(j)} \\ \mathbf{v}_k^{(j)} \end{bmatrix} = \text{Chol} \begin{pmatrix} \mathbf{P}_{\boldsymbol{\rho}\boldsymbol{\rho}} + \mathbf{P}_{\boldsymbol{\rho}\boldsymbol{\omega}}\mathbf{P}_{\boldsymbol{\omega}\boldsymbol{\omega}}^{-1}\mathbf{P}_{\boldsymbol{\rho}\boldsymbol{\omega}}^\top & \mathbf{0}_{\ell\times v} \\ \mathbf{0}_{v\times\ell} & \mathbf{R}_k \end{pmatrix}_j \begin{bmatrix} \boldsymbol{z}_{k|k-1}^{(j)} \\ \boldsymbol{z}_{\mathbf{v}_k}^{(j)} \end{bmatrix}, \quad \forall j \tag{4.56}$$

Hence, we have

$$\boldsymbol{\xi}_{k|k-1}^{(j)} = \begin{bmatrix} \boldsymbol{\omega}_{k|k-1}^{(j)} \\ \boldsymbol{\rho}_{k|k-1}^{(j)} \end{bmatrix} \tag{4.57}$$

and the UT into the function $\mathbf{h}$ can be performed to obtain the estimated mean of the measurement $\hat{\mathbf{y}}_k$, its covariance $\mathbf{P}_{\mathbf{y}_k}$ and the cross-covariance $\mathbf{P}_{\boldsymbol{\xi}_k\mathbf{y}_k}$. The whole procedure is summarized in Algorithm 5. Finally, we compute the posterior estimate $\hat{\boldsymbol{\chi}}_{k|k}$ and covariance $\mathbf{P}_{k|k}$ thanks to the UKF-LG update equations, as well as the posterior concentration parameters $\boldsymbol{\kappa}_{k|k}$. This concludes the proposed method, which is summarized in Algorithm 6.

Remark 4.5. Regarding the correction of $\boldsymbol{\kappa}_{k|k-1}$ in order to produce an updated vector $\boldsymbol{\kappa}_{k|k}$, we did not yet provide an analytical correction method. An alternative solution would be to extract the set of posterior κ_i from the variance of each angular variables $P_{\omega_i\omega_i}$, thanks to the relationship $\kappa_i = \vartheta(P_{\omega_i\omega_i})$. The problem is that we do not explicitly know the expression of such a relationship, except when the distribution in ω_i is highly concentrated, in which case we have

$$\kappa_i = \vartheta(P_{\omega_i\omega_i}) = P_{\omega_i\omega_i}^{-1}. \tag{4.58}$$

However, a very small variance $P_{\omega_i\omega_i}$ might numerically make κ_i tend to a singular value, which must be avoided in implementation. Instead, a more stable numerical approximation of the unkown function ϑ should be used. This approximation should behave similarly to $\vartheta(P_{\omega_i\omega_i}) = P_{\omega_i\omega_i}^{-1}$ for small variances, while avoiding the singularity as much as possible. For example, this can be performed thanks to data interpolation with a model of the form

$$\vartheta(x) = c_1 \exp(c_2 x) + c_3 \exp(c_4 x), \tag{4.59}$$

where c_1, c_2, c_3 and c_4 are scalar parameters whose values are obtained from an arbitrary interpolation method.

As a conclusion, the algorithmic contribution is the development of a vMG-UKF-LG, building upon the works of [Brossard et al., 2017] and [Horwood and Poore, 2014]. It provides a way to capture and propagate the uncertainty on the Lie groups $\boldsymbol{SE}(n)$, $\boldsymbol{SE}_k(n)$ or $\boldsymbol{SE}_k(n) \times \mathbb{R}^N$ using a revised sigma-point method. The set of sigma-points and weights seeks to capture the density of the proposed vMG distribution instead of a full Gaussian. Then, the algorithm classically propagates the mean $\hat{\boldsymbol{\chi}}$ and the covariance $\mathbf{P}$, but with the addition of a concentration parameter $\boldsymbol{\kappa}$ in the filtering scheme as a complementary dispersion measure for the angular components.

Algorithm 4: Unscented Transformation for the propagation model $\boldsymbol{f}$; $\boldsymbol{\mathcal{X}} \sim \mathcal{VMG}_{\mathsf{L}}(\hat{\boldsymbol{\mathcal{X}}}, \boldsymbol{\kappa}, \mathbf{P})$

Inputs: $\hat{\boldsymbol{\mathcal{X}}}_{k-1|k-1}$, $\mathbf{P}_{k-1|k-1}$, $\boldsymbol{\kappa}_{k-1|k-1}$, $\mathbf{u}_{k-1}$, $\mathbf{Q}_{k-1}$, $\boldsymbol{\kappa}_{\mathbf{u}_{k-1}}$, τ

1: Propagate the mean pose:

$$\hat{\boldsymbol{\mathcal{X}}}_{k|k-1} = \hat{\boldsymbol{\mathcal{X}}}_{k-1|k-1} \exp(\mathbf{u}_{k-1}^{\wedge}\tau)$$

2: Let the state augmentation $d' = 2d$ and the canonical form
$\boldsymbol{\zeta}_{aug} \sim \mathcal{VMG}([\boldsymbol{\omega}, \boldsymbol{\omega}_{\mathbf{w}}], [\boldsymbol{z}, \boldsymbol{z}_{\mathbf{w}}];\ \boldsymbol{\kappa}^{aug}_{k-1|k-1}, \mathbf{I}_{2\ell\times 2\ell})$ with $\boldsymbol{\kappa}^{aug}_{k-1|k-1} = [\boldsymbol{\kappa}^{\top}_{k-1|k-1}, \boldsymbol{\kappa}^{\top}_{\mathbf{u}_{k-1}}]^{\top}$

3: Generate sigma-points of the canonical vMG:

$$\begin{aligned}
\boldsymbol{\zeta}^{(0)}_{aug} &= \mathbf{0} \\
\boldsymbol{\zeta}^{(j)}_{aug} &= -\epsilon_j \times [\mathbf{I}_{d'\times d'}]_j\ , \quad \forall j = 1, \dots, d' \\
\boldsymbol{\zeta}^{(d'+j)}_{aug} &= +\epsilon_j \times [\mathbf{I}_{d'\times d'}]_j\ , \quad \forall j = 1, \dots, d'
\end{aligned}$$

$$\text{with:}\quad \epsilon_j = \epsilon_{d'+j} = \arccos\left(\frac{B_2(\kappa^{aug}_j)}{2B_1(\kappa^{aug}_j)} - 1\right), \quad \forall j = 1, \dots, 2m$$

$$\epsilon_j = \epsilon_{d'+j} = \sqrt{3}, \qquad \forall j = 2m+1, \dots, d'$$

4: Generate weights:

$$\begin{aligned}
W^{(j)} = W^{(d'+j)} &= \frac{B_1(\kappa^{aug}_j)^2}{4B_1(\kappa^{aug}_j) - B_2(\kappa^{aug}_j)}, \quad \forall j = 1, \dots, 2m \\
W^{(j)} = W^{(d'+j)} &= \frac{1}{6}\ , \qquad \forall j = 2m+1, \dots, d' \\
W^{(0)} &= 1 - \sum_{j=1}^{2d'} W^{(j)}
\end{aligned}$$

5: Compute $\boldsymbol{\xi}_{aug}$ by applying the following transformation $\forall j$:

$$\boldsymbol{\xi}^{(j)}_{aug} = \left[\ \mathrm{diag}\left(\mathbf{I}_{2m\times 2m}\ ,\ \mathrm{Chol}\begin{pmatrix} \mathbf{P}_{\boldsymbol{\rho}\boldsymbol{\rho}} + \mathbf{P}_{\boldsymbol{\rho}\boldsymbol{\omega}}\mathbf{P}^{-1}_{\boldsymbol{\omega}\boldsymbol{\omega}}\mathbf{P}^{\top}_{\boldsymbol{\rho}\boldsymbol{\omega}} & \mathbf{0}_{\ell\times\ell} \\ \mathbf{0}_{\ell\times\ell} & \mathbf{Q}_{\boldsymbol{\rho}\boldsymbol{\rho}} + \mathbf{Q}_{\boldsymbol{\rho}\boldsymbol{\omega}}\mathbf{Q}^{-1}_{\boldsymbol{\omega}\boldsymbol{\omega}}\mathbf{Q}^{\top}_{\boldsymbol{\rho}\boldsymbol{\omega}} \end{pmatrix}\right)\right]_j \times \boldsymbol{\zeta}^{(j)}_{aug}$$

$$\text{s.t.}\quad \boldsymbol{\xi}^{(j)}_{aug} := \left[\boldsymbol{\omega}^{(j)\top}_{k-1|k-1},\quad \boldsymbol{\omega}^{(j)\top}_{\mathbf{w}_{k-1}},\quad \boldsymbol{\rho}^{(j)\top}_{k-1|k-1},\quad \boldsymbol{\rho}^{(j)\top}_{\mathbf{w}_{k-1}}\right]^{\top}$$

6: Let $\boldsymbol{\xi}^{(j)}_{k-1|k-1} := [\boldsymbol{\omega}^{(j)\top}_{k-1|k-1}, \boldsymbol{\rho}^{(j)\top}_{k-1|k-1}]^{\top}$ and $\mathbf{w}^{(j)}_{k-1} := [\boldsymbol{\omega}^{(j)\top}_{\mathbf{w}_{k-1}}, \boldsymbol{\rho}^{(j)\top}_{\mathbf{w}_{k-1}}]^{\top}$. Propagate sigma-points, $\forall j = 0, \dots, 2d'$:

$$\boldsymbol{\xi}^{(j)}_{k|k-1} = \log\left(\hat{\boldsymbol{\mathcal{X}}}^{-1}_{k|k-1}\hat{\boldsymbol{\mathcal{X}}}_{k-1|k-1}\exp(\boldsymbol{\xi}^{(j)\wedge}_{k-1|k-1})\exp(\mathbf{u}^{\wedge}_{k-1}\tau)\exp(\mathbf{w}^{(j)\wedge}_{k-1}\tau)\right)^{\vee}$$

7: Reconstruct the covariance and concentration vector $\boldsymbol{\kappa}_{k|k-1} = [\kappa_1, \dots, \kappa_m]^{\top}$:

$$\mathbf{P}_{k|k-1} = \sum_{j=0}^{2d'} W^{(j)}\left(\boldsymbol{\xi}^{(j)}_k\right)\left(\boldsymbol{\xi}^{(j)}_k\right)^{\top}$$

$$\forall i = 1, \dots, m: \quad \overline{R}_i = \sqrt{\left(\sum_{j=0} W^{(j)}\cos\left(\omega^{(j)}_{i,k|k-1}\right)\right)^2 + \left(\sum_{j=0} W^{(j)}\sin\left(\omega^{(j)}_{i,k|k-1}\right)\right)^2} \Rightarrow \kappa_i = \frac{2\overline{R}_i - \overline{R}_i^3}{1-\overline{R}_i^2}$$

Outputs: $\hat{\boldsymbol{\mathcal{X}}}_{k|k-1}$, $\mathbf{P}_{k|k-1}$, $\boldsymbol{\kappa}_{k|k-1}$

Algorithm 5: Unscented Transformation for the observation model $\mathbf{h}$; $\boldsymbol{\chi} \sim \mathcal{VMG}_{\mathsf{L}}(\hat{\boldsymbol{\chi}}, \boldsymbol{\kappa}, \mathbf{P})$

Inputs: $\hat{\boldsymbol{\chi}}_{k|k-1}$, $\mathbf{P}_{k|k-1}$, $\boldsymbol{\kappa}_{k|k-1}$, $\mathbf{R}_k$

1: Let the state augmentation $d' = d + v$ and the canonical form
$\boldsymbol{\zeta}_{aug} \sim \mathcal{VMG}(\boldsymbol{\omega}, [\boldsymbol{z}, \mathbf{v}_{\boldsymbol{z}}]; \boldsymbol{\kappa}_{k|k-1}, \mathbf{I}_{(\ell+v)\times(\ell+v)})$

2: Generate sigma-points of the canonical vMG:

$$\begin{aligned}
\boldsymbol{\zeta}_{aug}^{(0)} &= \mathbf{0} \\
\boldsymbol{\zeta}_{aug}^{(j)} &= -\epsilon_j \times [\mathbf{I}_{d'\times d'}]_j \,, \quad \forall j = 1, \dots, d' \\
\boldsymbol{\zeta}_{aug}^{(d'+j)} &= +\epsilon_j \times [\mathbf{I}_{d'\times d'}]_j \,, \quad \forall j = 1, \dots, d'
\end{aligned}$$

$$\begin{aligned}
\text{with:} \quad \epsilon_j = \epsilon_{d'+j} &= \arccos\left(\frac{B_2(\kappa_j)}{2B_1(\kappa_j)} - 1\right), \quad \forall j = 1, \dots, m \\
\epsilon_j = \epsilon_{d'+j} &= \sqrt{3}, \qquad\qquad\qquad\quad \forall j = m+1, \dots, d'
\end{aligned}$$

3: Generate weights:

$$\begin{aligned}
W^{(j)} = W^{(d'+j)} &= \frac{B_1(\kappa_j)^2}{4B_1(\kappa_j) - B_2(\kappa_j)}, \quad \forall j = 1, \dots, m \\
W^{(j)} = W^{(d'+j)} &= \frac{1}{6} \,, \qquad\qquad\qquad\quad \forall j = m+1, \dots, d' \\
W^{(0)} &= 1 - \sum_{j=1}^{2d'} W^{(j)}
\end{aligned}$$

4: Compute $\boldsymbol{\xi}_{aug}$ by applying the following transformation $\forall j$

$$\begin{aligned}
\boldsymbol{\xi}_{aug}^{(j)} &= \left[\operatorname{diag}\left(\mathbf{I}_{m\times m} \,,\, \operatorname{Chol}\begin{pmatrix} \mathbf{P}_{\boldsymbol{\rho}\boldsymbol{\rho}} + \mathbf{P}_{\boldsymbol{\rho}\boldsymbol{\omega}}\mathbf{P}_{\boldsymbol{\omega}\boldsymbol{\omega}}^{-1}\mathbf{P}_{\boldsymbol{\rho}\boldsymbol{\omega}}^{\top} & \mathbf{0}_{\ell\times v} \\ \mathbf{0}_{v\times\ell} & \mathbf{R}_k \end{pmatrix}\right)\right]_j \times \boldsymbol{\zeta}_{aug}^{(j)}, \\
\text{s.t.} \quad \boldsymbol{\xi}_{aug}^{(j)} &:= \left[\boldsymbol{\omega}_{k|k-1}^{(j)\top}, \quad \boldsymbol{\rho}_{k|k-1}^{(j)\top}, \quad \mathbf{v}_k^{(j)\top}\right]^{\top}
\end{aligned}$$

5: Let $\boldsymbol{\xi}_{k|k-1}^{(j)} := [\boldsymbol{\omega}_{k|k-1}^{(j)\top}, \boldsymbol{\rho}_{k|k-1}^{(j)\top}]^{\top}$. Propagate sigma-points, $\forall j = 0, \dots, 2d'$:

$$\mathbf{y}_k^{(j)} = \mathbf{h}\left(\hat{\boldsymbol{\chi}}_{k|k-1} \exp(\boldsymbol{\xi}_{k|k-1}^{(j)\wedge}), \mathbf{v}_k^{(j)}\right)$$

6: Compute the mean and covariance of the measurement, and the cross-covariance:

$$\hat{\mathbf{y}}_k = \sum_{j=0}^{2d'} W^{(j)} \mathbf{y}_k^{(j)}, \qquad \mathbf{P}_{\mathbf{y}_k} = \sum_{j=0}^{2d'} W^{(j)} \left(\mathbf{y}_k^{(j)} - \hat{\mathbf{y}}_k\right)\left(\mathbf{y}_k^{(j)} - \hat{\mathbf{y}}_k\right)^{\top}$$

$$\mathbf{P}_{\boldsymbol{\xi}_k\mathbf{y}_k} = \sum_{j=0}^{2d'} W^{(j)} \left(\boldsymbol{\xi}_k^{(j)}\right)\left(\mathbf{y}_k^{(j)} - \hat{\mathbf{y}}_k\right)^{\top}$$

Outputs: $\hat{\mathbf{y}}_k$, $\mathbf{P}_{\mathbf{y}_k}$, $\mathbf{P}_{\boldsymbol{\xi}_k\mathbf{y}_k}$

Algorithm 6: The von Mises-Gaussian UKF-LG ; $\boldsymbol{\mathcal{X}} \sim \mathcal{VMG}_{\mathsf{L}}(\hat{\boldsymbol{\mathcal{X}}}, \boldsymbol{\kappa}, \mathbf{P})$

Inputs: $\hat{\boldsymbol{\mathcal{X}}}_{k-1|k-1}$, $\mathbf{P}_{k-1|k-1}$, $\boldsymbol{\kappa}_{k-1|k-1}$, $\mathbf{u}_{k-1}$, $\mathbf{Q}_{k-1}$, $\boldsymbol{\kappa}_{\mathbf{u}_{k-1}}$, $\mathbf{y}_k$, $\mathbf{R}_k$, τ

Time propagation from $k-1$ to k

1: von Mises-Gaussian Unscented Transformation through the model $\boldsymbol{f}$ (Algorithm 4):

$$\left[\hat{\boldsymbol{\mathcal{X}}}_{k|k-1}, \mathbf{P}_{k|k-1}, \boldsymbol{\kappa}_{k|k-1}\right] = \text{vMG-UT}_{\boldsymbol{f}}\left(\hat{\boldsymbol{\mathcal{X}}}_{k-1|k-1}, \mathbf{P}_{k-1|k-1}, \boldsymbol{\kappa}_{k-1|k-1}, \mathbf{u}_{k-1}, \mathbf{Q}_{k-1}, \boldsymbol{\kappa}_{\mathbf{u}_{k-1}}, \tau\right)$$

Update thanks to the measurement $\mathbf{y}_k$

2: von Mises-Gaussian Unscented Transformation through the model $\mathbf{h}$ (Algorithm 5):

$$\left[\hat{\mathbf{y}}_k, \mathbf{P}_{\mathbf{y}_k}, \mathbf{P}_{\boldsymbol{\xi}_k \mathbf{y}_k}\right] = \text{vMG-UT}_{\mathbf{h}}\left(\hat{\boldsymbol{\mathcal{X}}}_{k|k-1}, \mathbf{P}_{k|k-1}, \boldsymbol{\kappa}_{k|k-1}, \mathbf{R}_k\right)$$

3: Kalman gain and innovation:

$$\mathbf{K} = \mathbf{P}_{\boldsymbol{\xi}_k \mathbf{y}_k} \mathbf{P}_{\mathbf{y}_k}^{-1}$$
$$\tilde{\mathbf{y}}_k = \mathbf{y}_k - \hat{\mathbf{y}}_k$$

4: State update:

$$\hat{\boldsymbol{\mathcal{X}}}_{k|k} = \hat{\boldsymbol{\mathcal{X}}}_{k|k-1} \exp\left(\left[\mathbf{K}\tilde{\mathbf{y}}_k\right]^{\wedge}\right)$$
$$\mathbf{P}_{k|k}^{-} = \mathbf{P}_{k|k-1} - \mathbf{K}\mathbf{P}_{\mathbf{y}_k}\mathbf{K}^{\top}$$
$$\mathbf{P}_{k|k} = \boldsymbol{\mathcal{J}}_{\mathsf{R}}(\text{ad}\left(\left[\mathbf{K}\tilde{\mathbf{y}}_k\right]^{\wedge}\right))\mathbf{P}_{k|k}^{-}\boldsymbol{\mathcal{J}}_{\mathsf{R}}(\text{ad}\left(\left[\mathbf{K}\tilde{\mathbf{y}}_k\right]^{\wedge}\right))^{\top}$$

5: Deduce the posterior concentration parameter $\kappa_{i,k|k}$ $(\forall i = 1, \ldots, m)$, from the posterior variance $P_{\omega_i\omega_i}$:

$$\kappa_{i,k|k} \approx \vartheta(P_{\omega_i\omega_i}) \quad \Rightarrow \quad \boldsymbol{\kappa}_{k|k} = [\kappa_{1,k|k}, \ldots, \kappa_{m,k|k}]^{\top}$$

Outputs: $\hat{\boldsymbol{\mathcal{X}}}_{k|k}$, $\mathbf{P}_{k|k}$, $\boldsymbol{\kappa}_{k|k}$

In this chapter, we developed a probability model for the manifold $\mathbb{S}^m \times \mathbb{R}^\ell$, that we referred to as von Mises-Gaussian distribution. It models the uncertainty of a vector in exponential coordinates $\boldsymbol{\xi} = [\boldsymbol{\omega}^\top, \boldsymbol{\rho}^\top]^\top$, which is mapped into the group and translated to a given mean pose via left or right action to yield an equivariant von Mises-Gaussian for the variable $\boldsymbol{\mathcal{X}}$. The vMG still models the dispersion by means of a covariance matrix for the angular part $\boldsymbol{\omega}$, but assisted by concentration parameters κ_i. Then, this distribution was applied to a UKF-LG structure, which was adapted to account for this uncertainty model. By doing so, the proposed solution should address the limitations of the Lie-theoretic covariance under suitable limits. The development of this chapter will be evaluated on different filtering scenarios in Chapter 6.

Chapter 5

Development of an optimal UKF-LG under valid Gaussian assumptions

Contents

5.1 Limitations of the UT on Lie groups for valid Gaussian: numerical example 92
5.2 Recalls of the main results on optimal quantization 95
5.2.1 Optimal quantization of a random vector 95
5.2.2 Numerical integration by optimal quantization 95
5.2.3 Optimal quantization of a normally-distributed random vector . . . 96
5.3 Development of an optimal quantization-based UKF-LG 96
5.3.1 Validity assumptions of the proposed approach 97
5.3.2 Adaptation of the optimization problem for sigma-point adjustment 98
5.3.3 Application to the UKF-LG 99
5.3.4 Preliminary results: application of the optimal quantization to the UT on Lie groups 102

Whereas Chapter 4 introduced a first approach for improving the representation of uncertainty in a UKF-LG by focusing on probabilistic modeling aspects, the present chapter shifts the emphasis to numerical aspects, addressing the accurate propagation of uncertainty under the assumption that a Lie-theoretic covariance and an equivariant Gaussian model remain valid. In this regime, there is no need to resort to spherical probability densities to model angular uncertainty. However, as with UKFs in general, limitations persist in the accurate placement and adjustment of sigma-points, as discussed in Chapter 2, subsection §2.4.3. In Section §2.5, we introduced the optimal quantization as a helping tool to enhance the accuracy of the uncertainty propagation, and the present chapter proposes to adapt it to the standard UKF-LG.

The developments of this chapter have been the subject of two publications: the first, presented in the international conference on Information Fusion (FUSION) [Pravong et al., 2024], consisted in applying and evaluating the interest of optimal quantization on a classical UKF for the problem of 2D pose estimation in $\mathbb{R}^3$. The second publication,

presented in the European Control Conference (ECC) [Pravong et al., 2025b], consisted in applying the optimal quantization for the UKF-LG, as exactly described in the present chapter. This paper focused on a 2D pose estimation problem on $\boldsymbol{SE}(2)$, while presenting its interest within an Invariant Linear Quadratic Regulator framework.

5.1 Limitations of the UT on Lie groups for valid Gaussian: numerical example

Let us provide a numerical example to further motivate the need to enhance sigma-point placement and propagation in the case of a propagation model on a Lie group. Indeed, even if the dispersion is small enough to consider that an equivariant Gaussian remains relevant, it is still possible that the current sigma-point procedure (generation + propagation) poorly captures the first and second moments of the state. This question does not challenge the relevancy of using a Gaussian distribution, but rather challenges the algorithmic method used to generate sigma-points under valid Gaussian assumption. Actually, this question is not specific to the UKF-LG, but is initially asked for a standard UKF in the Euclidean setting, as already discussed in subsection §2.4.3. In any case, numerical errors in the reconstruction of the covariance after UT (whether for the propagation or for the correction step) can lead to state estimation errors within the Kalman filtering scheme, even when the notion of Lie-theoretic covariance remains valid.

Illustrative example: propagation on $\boldsymbol{SE}(2)$

We illustrate the imperfections of existing sigma-point methods that seek to propagate the uncertainty on the motion group $\boldsymbol{SE}(2)$. In practice, it is not perfectly clear at which threshold the notion of covariance becomes questionable. In the previous chapter (Section §4.1), we identified this threshold as $\sigma = 60°$ in the case of $\boldsymbol{SO}(2)$. However, from $\sigma = 60°$ to $\sigma = 100°$, the difference between $P_s(\xi_\theta)$ and $P_s(\xi'_\theta)$ could be considered negligible for a practitioner. In this chapter, we can consider the covariance validity domain to be $0° < \sigma < 100°$.

We now evaluate the propagation made by several UTs compared to a Monte-Carlo propagation, i.e., to compare the propagated covariance reconstructed from a prediction step of a Kalman filter and the propagated covariance reconstructed from M Monte-Carlo transformations. We perform the classical approximations made by Algorithm 1 with the scaling parameters $\{s_1, s_2, s_3\}$:

- The covariance $\mathbf{P}_k^{UT1}$, obtained with the scaling parameters of the Gauss-Hermite rule $\{s_1 = 1,\ s_2 = 0,\ s_3 = 3 - d'\}$.
- The covariance $\mathbf{P}_k^{UT2}$, obtained with the alternative Gaussian scaling parameters $\{s_1 = 10^{-3},\ s_2 = 2,\ s_3 = 0\}$.
- The covariance $\mathbf{P}_k^{CT}$, obtained with the scaling parameters of the cubature rule $\{s_1 =$

$1,\ s_2 = 0,\ s_3 = 0\}$.

We then compare each of them with the covariance matrix $\mathbf{P}_k$ computed from Monte-Carlo transformations. The difference is assessed via the Frobenius norm.

$$\varepsilon(\mathbf{P}_k^{[\emptyset]}, \mathbf{P}_k) = \sqrt{\mathrm{tr}\Big[(\mathbf{P}_k^{[\emptyset]} - \mathbf{P}_k)^\top (\mathbf{P}_k^{[\emptyset]} - \mathbf{P}_k)\Big]} \tag{5.1}$$

To compute the Monte-Carlo covariance, we draw a large number M of random samples $\{\boldsymbol{\xi}_{k-1}^{(j)}\}_{j=1,\dots,M}$ and $\{\mathbf{w}_{k-1}^{(j)}\}_{j=1,\dots,M}$, normally-distributed according to $\mathbf{P}_{k-1}$ and $\mathbf{Q}$, respectively. The Monte-Carlo transformation is identical to the propagation of sigma-points, but with equal weights given by $W^{(j)} = \frac{1}{M}, \forall j$. This slow (but accurate) approach serves as a benchmark to which the other much faster methods were compared.

Let $\boldsymbol{\chi} \in \boldsymbol{SE}(2)$. The initial mean state is $\hat{\mathbf{x}}_{k-1} = [\hat{\theta}_{k-1}, \hat{x}_{1,k-1}, \hat{x}_{2,k-1}]^\top$ and its associated matrix is:

$$\hat{\boldsymbol{\chi}}_{k-1} = \begin{pmatrix} \cos(\hat{\theta}_{k-1}) & -\sin(\hat{\theta}_{k-1}) & \hat{x}_{1,k-1} \\ \sin(\hat{\theta}_{k-1}) & \cos(\hat{\theta}_{k-1}) & \hat{x}_{2,k-1} \\ 0 & 0 & 1 \end{pmatrix} \tag{5.2}$$

with initial uncertainty:

$$\mathbf{P}_{k-1} = a^2 \times \begin{bmatrix} (\sigma_{\theta_{k-1}})^2 & 0 & 0 \\ 0 & (\sigma_{\rho_{1,k-1}})^2 & 0 \\ 0 & 0 & (\sigma_{\rho_{2,k-1}})^2 \end{bmatrix} \tag{5.3}$$

The inputs and their associated noise are

$$\mathbf{u}_{k-1} = [u_\theta, u_{\rho_1}, u_{\rho_2}]^\top, \quad \mathbf{Q} = b^2 \times \begin{bmatrix} (q_\theta)^2 & 0 & 0 \\ 0 & (q_{\rho_1})^2 & 0 \\ 0 & 0 & (q_{\rho_2})^2 \end{bmatrix} \tag{5.4}$$

Numerical application:

$\hat{\theta}_{k-1} = \frac{\pi}{2}$	(rad)	$\sigma_{\theta_{k-1}} = 0.1$	(rad)	$u_\theta = \frac{\pi}{180}$	(rad/s)	$q_\theta = 0.3162$	(rad/s)
$\hat{x}_{1,k-1} = 1$	(m)	$\sigma_{\rho_{1,k-1}} = 0.01$	(m)	$u_{\rho_1} = 0.05$	(m/s)	$q_{\rho_1} = 0.3162$	(m/s)
$\hat{x}_{2,k-1} = 2$	(m)	$\sigma_{\rho_{2,k-1}} = 0.01$	(m)	$u_{\rho_2} = 0.05$	(m/s)	$q_{\rho_2} = 0.3162$	(m/s)

We provide a high input noise by setting $b = 3.1623$. The timestep is $\tau = 1$ sec. We perform a single propagation from $k-1$ to k for different scaling factors $a \in [1; 18]$. and compute the Frobenius norm of the difference between $\mathbf{P}_k^{[\emptyset]}$ and $\mathbf{P}_k$ ($M = 50000$). The results of the three methods are summarized in Figure 5.1, where the Frobenius norm of the error is plotted as a function of the initial covariance, parameterized by a. These results indicate a growing discrepancy between the approximated covariance $\mathbf{P}_k^{[\emptyset]}$ obtained from a given Unscented Transform and the true empirical covariance $\mathbf{P}_k^{MC}$, even for moderate initial uncertainties. The propagated covariance $\mathbf{P}_k^{MC}$ remains valid, as confirmed by Figure 5.3, ensuring that the comparison is meaningful. Analyzing the errors at the level of individual state variables (cf. Figure 5.2) indicates that the covariance error is primarily due to the angular component.

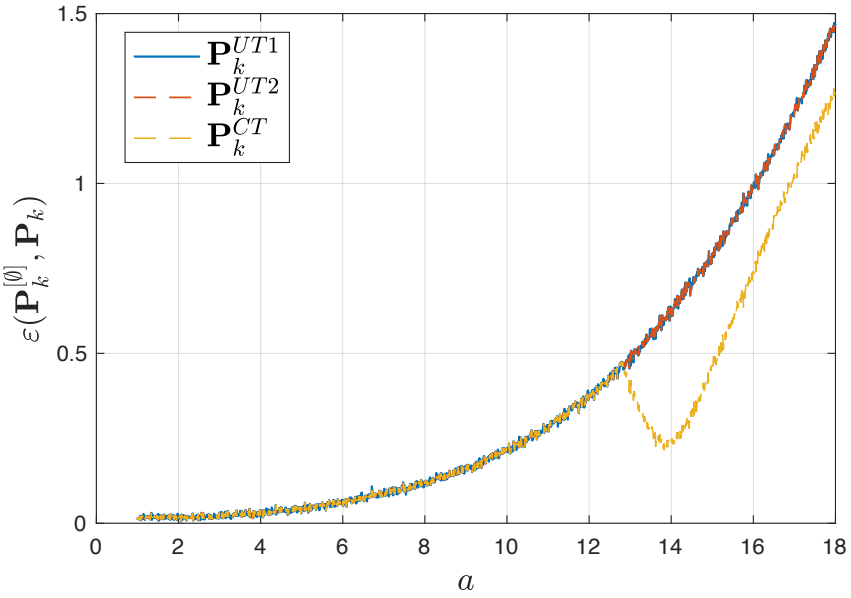


Figure 5.1: Error ε, as a function of a ($b = 3.1623$), of the propagated covariance $\mathbf{P}_k^{[\emptyset]}$ in comparison to an accurate propagated covariance $\mathbf{P}_k$ computed from Monte-Carlo transformations.

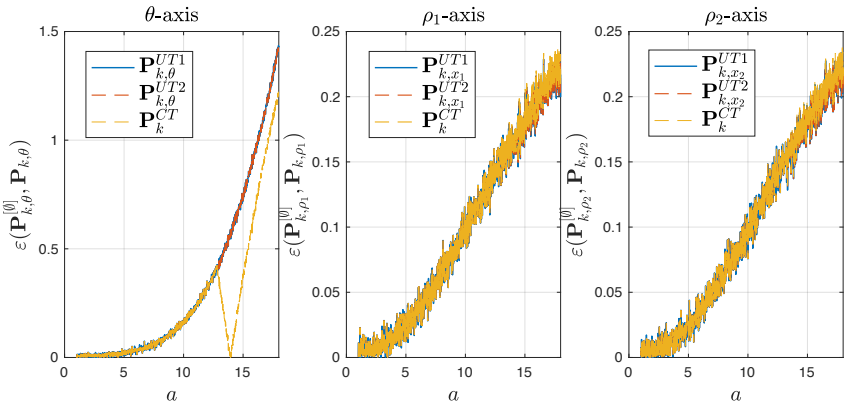


Figure 5.2: Error ε, as a function of a ($b = 3.1623$), of the propagated variance of θ (left), ρ_1 (middle) and ρ_2 (right). The order of magnitude of the errors indicates that the angular variable contributes the most to the global error of each unscented method.

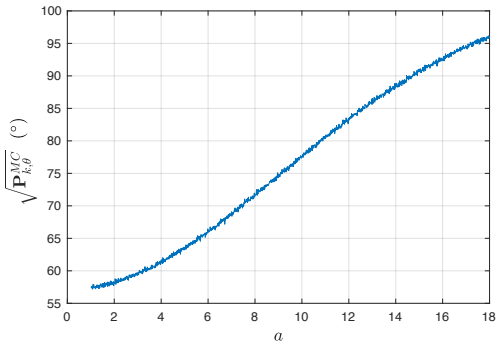


Figure 5.3: Standard deviation of the Monte-Carlo transformations along the θ-axis. This standard deviation indicates that the propagated covariance $\mathbf{P}_k^{MC}$ can be considered inside the validity domain of the Lie-theoretic definition.

The observed discrepancy between the covariance estimated through the Unscented Transform and the empirical covariance obtained via Monte-Carlo transformations highlights a numerical limitation. Even within the theoretical domain where the notion of covariance on Lie groups remains valid, the approximated covariance may still fail to accurately capture the true dispersion of the state. This observation suggests that refining the covariance approximation itself can lead to tangible improvements in estimation performance, independently of any change in the underlying probability model. Building on this observation, a second objective of this thesis is to enhance the Unscented Transform under Gaussian assumptions by introducing an optimal quantization approach, designed to refine the placement of sigma-points for a more accurate representation of the second moment of the propagated distribution.

5.2 Recalls of the main results on optimal quantization

Let us recall the main results on optimal quantization, previously detailed in Chapter 2, Section §2.5. These mathematical tools will serve for the subsequent developments presented in the following sections.

5.2.1 Optimal quantization of a random vector

For a continuous random vector $\boldsymbol{\xi} \in \mathbb{R}^d$ following a probability density function $\mathcal{P}(\boldsymbol{\xi})$, the goal of an p-optimal quantization problem is to find a N-tuple $\{\boldsymbol{\xi}^{(i)}\}_{i=0,\dots,N-1}$ such that the distortion

$$D_N^{\mathrm{p}}\left(\{\boldsymbol{\xi}^{(i)}\}_{i=0,\dots,N-1}\right) = \int_{\mathbb{R}^d} \min_{0\leq i\leq N-1} \|\boldsymbol{\xi} - \boldsymbol{\xi}^{(i)}\|_{\mathrm{p}}^{\mathrm{p}} \mathcal{P}(\boldsymbol{\xi}) d\boldsymbol{\xi} \tag{5.5}$$

is minimized. The distortion can be interpreted as a local error averaged over the domain $\mathbb{R}^d$ with respect to the given probability density $\mathcal{P}(\boldsymbol{\xi})$.

5.2.2 Numerical integration by optimal quantization

For any Liptschitz function $\mathbf{g}$, the expected value of the propagated random vector $\mathbf{g}(\boldsymbol{\xi})$ is given by

$$\mathbb{E}[\mathbf{g}(\boldsymbol{\xi})] := \int_{\mathbb{R}^d} \mathbf{g}(\boldsymbol{\zeta}) \mathcal{P}(\boldsymbol{\xi}) d\boldsymbol{\zeta} \tag{5.6}$$

and the expected value of the propagated set $\{\mathbf{g}(\boldsymbol{\xi}^{(i)})\}_{i=0,\dots,N-1}$ is given by

$$\mathbb{E}[\mathbf{g}(\boldsymbol{\xi}^{(i)})] := \sum_{i=0}^{N-1} W^{(i)} \mathbf{g}(\boldsymbol{\xi}^{(i)}) \tag{5.7}$$

It is possible to make $\mathbb{E}[\mathbf{g}(\boldsymbol{\xi}^{(i)})]$ converge to $\mathbb{E}[\mathbf{g}(\boldsymbol{\xi})]$ by looking for a minimization of D_N^{p}, as stressed by Proposition 2.1:

$$\|\mathbb{E}[\mathbf{g}(\boldsymbol{\xi})] - \mathbb{E}[\mathbf{g}(\boldsymbol{\xi}^{(i)})]\|_{\mathrm{p}} \leq [\mathbf{g}]_{\mathrm{Lip}} \sqrt[\mathrm{p}]{D_N^{\mathrm{p}}} \tag{5.8}$$

5.2.3 Optimal quantization of a normally-distributed random vector

In the case where the random vector follows a Gaussian pdf, $\mathcal{P}(\boldsymbol{\xi}) = \mathcal{N}(\boldsymbol{\xi})$, with mean $\overline{\boldsymbol{\xi}}$ and covariance $\mathbf{P}$, the distortion (5.5) simplifies to

$$D_N^{\mathrm{p}} = \sum_{i=0}^{N-1} \int_{C(\boldsymbol{\xi}^{(i)})} \|\boldsymbol{\zeta} - \boldsymbol{\xi}^{(i)}\|_{\mathrm{p}}^{\mathrm{p}} \, \mathcal{N}(\boldsymbol{\xi}; \overline{\boldsymbol{\xi}}, \mathbf{P}) d\boldsymbol{\zeta} \tag{5.9}$$

where $C(\boldsymbol{\xi}^{(i)})$ is the Voronoï cell of centroid $\boldsymbol{\xi}^{(i)}$. The expression (5.9) also writes

$$D_N^{\mathrm{p}} = \sum_{i=0}^{N-1} \left(\frac{1}{M} \sum_{j=1}^{M} \|\boldsymbol{\zeta}^{(j)} - \boldsymbol{\xi}^{(i)}\|_{\mathrm{p}}^{\mathrm{p}} \right) \tag{5.10}$$

for an arbitrarily large integer M, and where $\boldsymbol{\zeta}^{(j)}$ is a random realization drawn from $\mathcal{N}(\boldsymbol{\xi}; \overline{\boldsymbol{\xi}}, \mathbf{P})$.

For this distortion, it is possible to apply a stochastic gradient descent (cf. Definition 2.2) on an initial set $\{\boldsymbol{\xi}^{(i)}\}_{i=0,\dots,N-1}$ such that the distortion decreases. The stochastic gradient descent is given by the set of sequences $\left(\{\boldsymbol{\xi}_n\}_{n\geq 0}, \{\boldsymbol{\zeta}_n\}_{n\geq 1}, \{\gamma_n\}_{n\geq 1}\right)$, where the sequence of steps $\{\gamma_n\}_{n\geq 1}$ satisfies

$$\gamma_n \in]0,1[, \quad \sum_{n\geq 1} \gamma_n = +\infty \quad \text{and} \quad \sum_{n\geq 1} \gamma_n^2 < +\infty \tag{5.11}$$

and the procedure is solved by the CLVQ algorithm

$$\begin{cases} \boldsymbol{\xi}_n^{(i)} = \boldsymbol{\xi}_{n-1}^{(i)} - \gamma_n(\boldsymbol{\xi}_{n-1}^{(i)} - \boldsymbol{\zeta}_n) & \text{if } \; i = i_{\mathrm{win}} \\ \boldsymbol{\xi}_n^{(i)} = \boldsymbol{\xi}_{n-1}^{(i)} & \text{if } \; i \neq i_{\mathrm{win}} \end{cases} \tag{5.12}$$

where $\boldsymbol{\zeta}_n$ is randomly drawn from $\mathcal{N}(\boldsymbol{\xi}; \overline{\boldsymbol{\xi}}, \mathbf{P})$ at each step, and i_{win} is the winning index of the competitive phase.

5.3 Development of an optimal quantization-based UKF-LG

In this section, our goal is to develop an Unscented Transform solution that improves the accuracy of covariance propagation under Gaussian assumption. In particular, the objective is to reduce the observed discrepancy between the estimated propagated covariance and the true covariance obtained through Monte-Carlo transformations (cf. Section §5.1). This refinement is considered within a regime in which the notion of covariance remains meaningful and valid. Thus, focusing not on redefining the probability density in itself, but on enhancing the precision of its numerical propagation by refining the placement of the sigma-points. To do that, we propose to use the optimal quantization tools described previously.

5.3.1 Validity assumptions of the proposed approach

In the context of UKF-LG, recall that before generating and applying sigma-point transformations to nonlinear functions $\boldsymbol{f}$ or $\mathbf{h}$, we are given a random vector $\boldsymbol{\xi}_{aug} = [\boldsymbol{\xi}, (\cdot)^\top]^\top \in \mathbb{R}^{d'}$, where d' is the augmented dimension of the state (due to internal noise involved). Besides, classical methods (such as a Gauss-Hermite method) generates a set of $N = 2d' + 1$ points and weights. In general, this point number is sufficient to accurately capture the first and second moments of a normally-distributed random vector $\boldsymbol{\xi} \sim \mathcal{N}(\boldsymbol{\xi}; \overline{\boldsymbol{\xi}}, \mathbf{P})$. Hence, given a set $\{\boldsymbol{\xi}^{(i)}\}_{i=0,\dots,2d'}$, a classical quadrature method makes the following equalities exact

$$\mathbb{E}[\boldsymbol{\xi}] = \overline{\boldsymbol{\xi}} := \int_{\mathbb{R}^{d'}} \boldsymbol{\zeta}\, \mathcal{N}(\boldsymbol{\xi}; \overline{\boldsymbol{\xi}}, \mathbf{P}) d\boldsymbol{\zeta} = \sum_{i=0}^{2d'} W^{(i)} \boldsymbol{\xi}^{(i)} \tag{5.13}$$

$$\begin{aligned} \mathbb{E}\left[\left(\boldsymbol{\xi} - \overline{\boldsymbol{\xi}}\right) \left(\boldsymbol{\xi} - \overline{\boldsymbol{\xi}}\right)^\top \right] = \mathbf{P} &:= \int_{\mathbb{R}^{d'}} \left(\boldsymbol{\zeta} - \overline{\boldsymbol{\xi}}\right) \left(\boldsymbol{\zeta} - \overline{\boldsymbol{\xi}}\right)^\top \mathcal{N}(\boldsymbol{\xi}; \overline{\boldsymbol{\xi}}, \mathbf{P}) d\boldsymbol{\zeta} \\ &= \sum_{i=0}^{2d'} W^{(i)} \left(\boldsymbol{\xi}^{(i)} - \overline{\boldsymbol{\xi}}\right) \left(\boldsymbol{\xi}^{(i)} - \overline{\boldsymbol{\xi}}\right)^\top \end{aligned} \tag{5.14}$$

However, it does not necessarily imply that the approximations of the propagated moments (after propagation into a nonlinear function $\mathbf{g}$) will be exactly accurate as well. In the case of a Lie group propagation, we have empirically shown in Section §5.1 that errors still remain, at least for three common benchmark methods. Hence, moment-matching criteria may not be sufficient to generate sigma-points and weights. Instead, the distortion D_N^{p}, defined by (5.9), may be used as an alternative (or complementary) criterion to obtain "optimal" sigma-points for which the propagated moments will be accurate according to the inequality proposition (5.8).

However, this proposition holds if and only if the propagation function $\mathbf{g}$ is Lipschitz, which means that it does not evolve with too abrupt variations. In our case, we should verify if it applies for a propagation function on Lie groups. Indeed, we consider a $\boldsymbol{G}$-valued propagation model $\boldsymbol{f}$ of the form $\boldsymbol{f} = \boldsymbol{\chi}_{k-1} \exp(\mathbf{u}_{k-1}^\wedge \tau) \exp(\mathbf{w}_{k-1}^\wedge \tau)$, while using the Lie algebra to define the uncertainty. Hence, our function $\mathbf{g}$ writes (for a left-equivariant framework)

$$\mathbf{g}(\boldsymbol{\xi}) = \log\left(\hat{\boldsymbol{\chi}}_k^{-1} \hat{\boldsymbol{\chi}}_{k-1} \exp(\boldsymbol{\xi}^\wedge) \exp(\mathbf{u}_{k-1}^\wedge \tau) \exp(\mathbf{w}^\wedge \tau) \right)^\vee \tag{5.15}$$

This mapping is not globally Lipschitz, but it is locally Lipschitz under standard assumptions. Indeed, the exponential map, group multiplication, and linear identification $[\cdot]^\vee$ are smooth operations on Lie groups. The only potential source of non-regularity is the logarithm map, which is smooth only on a local neighborhood of the identity and becomes singular near rotations of angle π. As long as the argument $\boldsymbol{\xi}$ of the logarithm remains within a compact subset of this regular neighborhood, the differential of $\mathbf{g}(\boldsymbol{\xi})$ is continuous and bounded, which implies that $\mathbf{g}$ is Lipschitz on that domain. However, no global Lipschitz constant can exist on the whole space, since the logarithm map is not globally well-defined and its derivative becomes unbounded near its singularities. In the

present chapter, this local Lipschitz property is therefore assumed to hold[1], meaning that the analysis is restricted to trajectories that remain within the regular domain of the Lie logarithm. For the observation function $\mathbf{h}$, it is also assumed that it is differentiable with bounded evolutions; hence, Liptschitz as well.

5.3.2 Adaptation of the optimization problem for sigma-point adjustment

Generating sigma-points using a classical procedure already provides a $(2d'+1)$-quantizer of the random vector $\boldsymbol{\xi}$, but not optimal in the sense of the minimization of $D^{\mathrm{p}}_{2d'+1}(\boldsymbol{\xi}^{(0)},\dots,\boldsymbol{\xi}^{(2d')})$. Ideally, we would like to keep this point number (since a UKF does not necessarily mean more than that). Therefore, we can choose to start from the points already generated by these classical procedures and simply propose to update them by optimal quantization using the CLVQ procedure (5.12). Hence, the optimization problem we consider is redefined and writes

$$\text{(OP')}: \begin{cases} \text{Generate a } (2d'+1)\text{-tuple } \{\boldsymbol{\xi}_0^{(0)},\dots,\boldsymbol{\xi}_0^{(2d')}\} \text{ using any classical method.} \\ \text{Update this tuple by stochastic gradient to find } \{\boldsymbol{\xi}_*^{(0)},\dots,\boldsymbol{\xi}_*^{(2d')}\} \text{ s.t.} \\ D^{\mathrm{p}}_{2d'+1}(\boldsymbol{\xi}_*^{(1)},\dots,\boldsymbol{\xi}_*^{(2d')}) \leq D^{\mathrm{p}}_{2d'+1}(\boldsymbol{\xi}_0^{(1)},\dots,\boldsymbol{\xi}_0^{(N)}). \\ \text{Additionally, update the weights } W^{(i)} \text{ accordingly, to yield } W_*^{(i)}. \end{cases}$$

It is not expected that this new set of points and weights will still satisfy the moment-matching (5.13) and (5.14). However, it is expected that the propagated mean and covariance computed from this sample will be as close as possible to the moment integrals of the random vector $\mathbf{g}(\boldsymbol{\xi})$. In terms of mathematics, it is thus expected that

$$\mathbb{E}[\mathbf{g}(\boldsymbol{\xi})] := \overline{\boldsymbol{\xi}}_{\mathbf{g}} = \int_{\mathbb{R}^{d'}} \mathbf{g}(\boldsymbol{\zeta})\,\mathcal{N}(\boldsymbol{\xi};\overline{\boldsymbol{\xi}},\mathbf{P})d\boldsymbol{\zeta} \approx \sum_{i=0}^{2d'} W_*^{(i)}\mathbf{g}(\boldsymbol{\xi}_*^{(i)}) \tag{5.16}$$

is as exact as possible following the minimization criterion (5.8). As a corollary, it is also expected that the covariance of the propagated sample

$$\mathbb{E}\left[\left(\mathbf{g}(\boldsymbol{\xi})-\overline{\boldsymbol{\xi}}_{\mathbf{g}}\right)\left(\mathbf{g}(\boldsymbol{\xi})-\overline{\boldsymbol{\xi}}_{\mathbf{g}}\right)^{\top}\right] := \mathbf{P}_{\mathbf{g}} \approx \sum_{i=0}^{2d'} W_*^{(i)}\left(\mathbf{g}(\boldsymbol{\xi}_*^{(i)})-\overline{\boldsymbol{\xi}}'\right)\left(\mathbf{g}(\boldsymbol{\xi}_*^{(i)})-\overline{\boldsymbol{\xi}}'\right)^{\top} \tag{5.17}$$

is as exact as possible following the same criterion.

The objective of the optimization problem (OP') is to readjust the sigma-points to yield a new set that would minimize the distortion (without necessarily reaching a global minimum), and starting from the already generated points provided by the usual scaling parameters. Hence, from a suboptimal set of points $\{\boldsymbol{\xi}_0^{(i)}\}_{i=0,\dots,2d'}$, the goal is to converge to an optimal one $\{\boldsymbol{\xi}_*^{(i)}\}_{i=0,\dots,2d'}$ by applying the CLVQ algorithm (5.12). In addition, we will also modify this procedure to provide additional degrees of freedom:

[1] Since we assumed a valid Lie-theoretic covariance, which implicitly means that the logarithm is well-defined.

• First, the CLVQ updates a given vector by applying a unique learning rate γ for all scalar components of this vector. Ideally, we would like to add more degrees of freedom by being able to update these scalar components differently from each other. Thus, we propose a modification of the step sequence $\{\gamma_n\}_{n\geq 1}$: instead of updating a vector $\boldsymbol{\xi}_{n-1}^{(i)} \in \mathbb{R}^d$ by a scalar parameter γ_n, we prefer to update it by a d-by-d diagonal matrix,

$$\boldsymbol{\Gamma}_n = \mathrm{diag}(\gamma_{1,n}, \ldots, \gamma_{d,n}) \tag{5.18}$$

where $\forall \ell = 1, \ldots, d$, $\gamma_{\ell,n}$ should satisfy the conditions required by the sequence of steps (5.11) to ensure asymptotic convergence.

Remark 5.1. Notice that these updates only concern the "state" part of $\boldsymbol{\xi}_{aug} = [\boldsymbol{\xi}, (\cdot)^\top]^\top$ but not the "noise" part. Although it is entirely possible to update each component of the vector $\boldsymbol{\xi}_{aug}$ until d', we can choose to update only up to $d < d'$. Besides, among the different components of $\boldsymbol{\xi}$, we can also choose to update only some of them, and not the entire vector. In practice, these choices will depend on desired specifications, such as the performance/time ratio, the sensitivity of each state variable, etc. .

• A second modification concerns the choice of the learning rate $\gamma_{\ell,n}$ for each component of $\boldsymbol{\xi}$ at a given index n. In [Pagès and Printems, 2003], a heuristic method is provided, and gives an adequate choice for these parameters. In this thesis, the latter heuristic method is overlooked. Instead, we propose to set each step parameter $\gamma_{\ell,n}$ as follows:

$$\gamma_{\ell,n} = \frac{\frac{1}{\nu}\sigma_\ell}{n} \tag{5.19}$$

where σ_ℓ is the square-root of the ℓ^{th} diagonal component of $\mathbf{P}$, and $\nu > 1$ is a tuning parameter allowing us to handle how much we want the point $\boldsymbol{\xi}_{n-1}^{(i)}$ to be udpated in the direction of $\boldsymbol{\zeta}_n$: the more ν is large, the less $\boldsymbol{\xi}_{n-1}^{(i)}$ is corrected.

• Finally, a stoping criterion must be defined for the gradient descent. While we can defined a certain threshold that $D^{\mathrm{p}}_{2d'+1}$ (or its differential $d(D^{\mathrm{p}}_{2d'+1})$) must reach, we prefer to introduce a tuning parameter $n_{max} \in \mathbb{N}$ representing the end of the loop. It should be chosen sufficiently large to ensure convergence, while remaining small enough to limit unnecessary computational cost.

We finally come up with a revisited CLVQ procedure. Let a set of $2d' + 1$ sigma-points $\{\boldsymbol{\xi}_0^{(i)}\}_{i=0,\ldots,2d'}$, capturing a normally-distributed random vector $\boldsymbol{\xi}$ with mean $\overline{\boldsymbol{\xi}}$ and covariance $\mathbf{P}$. These sigma-points are updated by the revisited CLVQ procedure described in Algorithm 7.

5.3.3 Application to the UKF-LG

We can now develop an advanced Unscented Transform for the computation of the propagated covariance $\mathbf{P}_k$, which naturally leads to another version of the UKF-LG. In this case, the random vector of interest is the $\mathbb{R}^{2d}$-valued random vector

$$\boldsymbol{\xi}_{aug} = \begin{bmatrix} \boldsymbol{\xi}_{k-1} \\ \mathbf{w}_{k-1} \end{bmatrix} \sim \mathcal{N}\Big(\mathbf{0},\ \mathbf{P}_{aug} = \mathrm{diag}(\mathbf{P}_{k-1}, \mathbf{Q}_{k-1})\Big) \tag{5.20}$$

Algorithm 7: Revisited Competitive Learning Vector Quantization (CLVQ)

Inputs: $\{\boldsymbol{\xi}_0^{(i)}\}_{i=0,\dots,2d'}$, $\overline{\boldsymbol{\xi}} \in \mathbb{R}^d$, $\mathbf{P} \in \mathbb{R}^{d\times d}$, $n_{max} \in \mathbb{N}$, $\{\boldsymbol{\Gamma}_n\}_{n=1,\dots,n_{max}}$

1: **for** $n = 1$ to n_{max} **do**
2: Draw a random vector $\boldsymbol{\zeta}_n$ from probability distribution $\mathcal{N}(\boldsymbol{\xi}; \overline{\boldsymbol{\xi}}, \mathbf{P})$
3: Find the winning index i_{win} such that $i_{\text{win}} = \underset{i}{\text{argmin}}(||\boldsymbol{\xi}_{n-1}^{(i)} - \boldsymbol{\zeta}_n||_{\text{p}})$
4: $\boldsymbol{\xi}_n^{(i)} = \boldsymbol{\xi}_{n-1}^{(i)} \quad \forall i \neq i_{\text{win}}$
5: $\boldsymbol{\xi}_n^{(i_{\text{win}})} = \boldsymbol{\xi}_{n-1}^{(i_{\text{win}})} - \boldsymbol{\Gamma}_n(\boldsymbol{\xi}_{n-1}^{(i_{\text{win}})} - \boldsymbol{\zeta}_n)$
6: $n \leftarrow n + 1$
7: **end for**
8: $\boldsymbol{\xi}_*^{(i)} = \boldsymbol{\xi}_{n_{max}}^{(i)}, \quad \forall i = 0, \dots, 2d$

Outputs: $\{\boldsymbol{\xi}_*^{(i)}\}_{i=0,\dots,2d'}$

This vector is of length $d' = 2d$, therefore we have $N = 2d' + 1$ sigma-points to generate and update. But, as discussed in Remark 5.1, only rows related to the $\boldsymbol{\xi}$ part will be updated. Indeed, we only want to modify the computation of the covariance associated with the state, since $\mathbf{P}$ changes at each Kalman iteration. On the other hand, $\mathbf{Q}$ is assumed constant throughout the estimation (and hand-tuned off-line). In the case where the input noise is non stationary, then it may be necessary to update it using the CLVQ. However, we did not consider this case in this thesis.

■ Sigma-point adjustment for the prediction UT

The revised UT for the propagation function $\boldsymbol{f}$ on Lie groups begins by generating an initial set of sigma-points $\{\boldsymbol{\xi}_{k-1}^{(j)}\}_{j=0,\dots,2d'}$ from a distribution centered at zero with covariance $\mathbf{P}_{k-1}$, using the standard deterministic procedure with a chosen set of scaling parameters. In addition, weights $\{W^{(j)}\}_{j=0,\dots,2d'}$ are also generated according to this procedure. Before applying the CLVQ algorithm on these sigma-points, set the CLVQ parameters n_{max} and $\gamma_{\ell,n}$ (5.19) by tuning the coefficient ν, such that we have

$$\begin{aligned} \gamma_{\ell,n} &= \frac{\frac{1}{\nu}\sqrt{(\mathbf{P}_{k-1})_{\ell\ell}}}{n}, \quad \forall \ell = 1, \dots, d \quad \text{and} \quad \forall n = 1, \dots, n_{max} \\ \Rightarrow \boldsymbol{\Gamma}_n &= \text{diag}\left(\gamma_{1,n}, \dots, \gamma_{d,n}\right), \quad \forall n = 1, \dots, n_{max} \end{aligned} \tag{5.21}$$

Then, apply the CLVQ (Algorithm 7) on this initial set of sigma-points with the parameters n_{max}, $\{\boldsymbol{\Gamma}_n\}_{n=1,\dots,n_{max}}$, $\overline{\boldsymbol{\xi}} = \mathbf{0}$ and $\mathbf{P}_{k-1}$ as

$$\{\boldsymbol{\xi}_{k-1}^{(j)}\}_{j=0,\dots,2d'} \leftarrow \text{CLVQ}\Big(\{\boldsymbol{\xi}_{k-1}^{(j)}\}_{j=0,\dots,2d'},\ \mathbf{0},\ \mathbf{P}_{k-1},\ n_{max},\ \{\boldsymbol{\Gamma}_n\}_{n=1,\dots,n_{max}}\Big) \tag{5.22}$$

to yield a new set $\{\boldsymbol{\xi}_{k-1}^{(j)}\}_{j=0,\dots,2d'}$ that replaces the initial set. The updated sigma-points are then propagated into the function $\mathbf{g}(\boldsymbol{\xi})$ as follows:

$$\begin{aligned} &\text{Let:} \quad \boldsymbol{\xi}_{k-1}^{(j)} \quad \text{and} \quad \mathbf{w}_{k-1}^{(j)} \\ &\boldsymbol{\xi}_k^{(j)} = \mathbf{g}(\boldsymbol{\xi}_{k-1}^{(j)}) = \log\left(\hat{\boldsymbol{\chi}}_k^{-1}\hat{\boldsymbol{\chi}}_{k-1}\exp(\boldsymbol{\xi}_{k-1}^{(j)\wedge})\exp(\mathbf{u}_{k-1}^{\wedge}\tau)\exp(\mathbf{w}_{k-1}^{(j)\wedge}\tau)\right)^{\vee}, \quad \forall j = 0, \dots, 2d' \end{aligned} \tag{5.23}$$

knowing that we have computed $\hat{\boldsymbol{\chi}}_k$ from the non-noisy propagation

$$\hat{\boldsymbol{\chi}}_k = \boldsymbol{f}(\hat{\boldsymbol{\chi}}_{k-1}, \mathbf{u}_{k-1}, \mathbf{0}) = \hat{\boldsymbol{\chi}}_{k-1}\exp(\mathbf{u}_{k-1}^{\wedge}\tau) \tag{5.24}$$

Finally, the propagated covariance $\mathbf{P}_k$ is reconstructed from the propagated sample as

$$\mathbf{P}_k = \sum_{j=0}^{2d'} W^{(j)} \big(\boldsymbol{\xi}_k^{(j)}\big)\big(\boldsymbol{\xi}_k^{(j)}\big)^\top \tag{5.25}$$

Remark 5.2. Notice that the weights $W^{(j)}$ are left unchanged. Actually, in order for our method to remain consistent, the weights should also be readjusted with respect to the readjustment of the sigma-points. A potential idea is to set a counter $c_0^{(j)}$ and increment it during the CLVQ as $c_n^{(j)} = c_{n-1}^{(j)} + \mathbf{1}_{\{j=j_{\text{win}}\}}$, $\forall i = 0, \dots, 2d'$. Then, performing

$$W^{(j)} = c_{n_{max}}^{(j)}/n, \quad \forall j = 0, \dots, 2d'$$

should satisfy the condition $\sum_j W^{(j)} = 1$. During this thesis project, the intent was initially to apply this method along with the replacement of sigma-points, but we noticed that it resulted in worse performances than in the case where the $W^{(j)}$ were left unchanged. At the time of writing this manuscript, we did not find any solution to this problem. As a consequence, we decided to put aside the readjustment of the weights for the time being. However, this is still a problem that should eventually be addressed.

■ Sigma-point adjustment for the observation UT

To get a formulation of the UKF-LG enhanced with optimal quantization, the CLVQ algorithm should also be applied in the correction step, based on the current best guess of the covariance $\mathbf{P}_k$. It follows that the UT procedure for the observation model $\mathbf{y}_k = \mathbf{h}(\hat{\boldsymbol{\chi}}_{k|k-1} \exp(\boldsymbol{\xi}_k^\wedge), \mathbf{v}_k)$ does not differ much from what we have already done previously.

Hence, another set of sigma-points $\{\boldsymbol{\xi}_k^{(j)}\}_{j=0,\dots,2d'}$ (with $d' = d + v$ since $\mathbf{v}_k \in \mathbb{R}^v$) must be generated from $\mathbf{P}_k$, and updated with the CLVQ algorithm by tuning again the number of steps n_{max} and the learning rates matrix $\boldsymbol{\Gamma}$ (from tuning ν) such that we have

$$\{\boldsymbol{\xi}_k^{(j)}\}_{j=0,\dots,2d'} \leftarrow \text{CLVQ}\Big(\{\boldsymbol{\xi}_k^{(j)}\}_{j=0,\dots,2d'},\ \mathbf{0},\ \mathbf{P}_k,\ n_{max},\ \{\boldsymbol{\Gamma}_n\}_{n=1,\dots,n_{max}}\Big) \tag{5.26}$$

Then, we classicaly apply the Unscented transformation into $\mathbf{h}$ as follows:

$$\begin{aligned} &\text{Let:} \quad \boldsymbol{\xi}_k^{(j)} \quad \text{and} \quad \mathbf{v}_k^{(j)} \\ &\mathbf{y}_k^{(j)} = \mathbf{h}\Big(\hat{\boldsymbol{\chi}}_k \exp(\boldsymbol{\xi}_k^{(j)\wedge}), \mathbf{v}_k^{(j)}\Big), \quad \forall j = 0, \dots, 2d' \end{aligned} \tag{5.27}$$

where $\hat{\boldsymbol{\chi}}_k$ is known from the previous propagation into $\boldsymbol{f}$. After that, we can classically compute the estimated observation $\hat{\mathbf{y}}_k$, its covariance $\mathbf{P}_{\mathbf{y}_k}$ and the cross-covariance $\mathbf{P}_{\boldsymbol{\xi}_k \mathbf{y}_k}$ for the correction step.

■ Proposed algorithm

Finally, we come up with an optimal quantization-based UKF-LG, which we refer to as OQ-UKF-LG, summarized in Algorithm 8, and consisting of a standard UKF-LG, where each UT is enhanced with the CLVQ algorithm as described previously.

Algorithm 8: The optimal quantization-based UKF-LG ; $\boldsymbol{\mathcal{X}} \sim \mathcal{N}_{\mathsf{L}}(\hat{\boldsymbol{\mathcal{X}}}, \mathbf{P})$

Inputs: $\hat{\boldsymbol{\mathcal{X}}}_{k-1|k-1}$, $\mathbf{P}_{k-1|k-1}$, $\mathbf{u}_{k-1}$, $\mathbf{Q}_{k-1}$, $\mathbf{y}_k$, $\mathbf{R}_k$, τ

Time propagation from $k-1$ to k

1: Unscented Transformation through the model $\boldsymbol{f}$:

$$\left[\hat{\boldsymbol{\mathcal{X}}}_{k|k-1}, \mathbf{P}_{k|k-1}\right] = UT_{\boldsymbol{f}}^{+CLVQ}(\hat{\boldsymbol{\mathcal{X}}}_{k-1|k-1}, \mathbf{P}_{k-1|k-1}, \mathbf{u}_{k-1}, \mathbf{Q}_{k-1}, \tau)$$

Update thanks to the measurement $\mathbf{y}_k$

2: Unscented Transformation through the model $\mathbf{h}$:

$$\left[\hat{\mathbf{y}}_k, \mathbf{P}_{\mathbf{y}_k}, \mathbf{P}_{\boldsymbol{\xi}_k \mathbf{y}_k}\right] = UT_{\mathbf{h}}^{+CLVQ}(\hat{\boldsymbol{\mathcal{X}}}_{k|k-1}, \mathbf{P}_{k|k-1}, \mathbf{R}_k)$$

3: Kalman gain and innovation:

$$\mathbf{K} = \mathbf{P}_{\boldsymbol{\xi}_k \mathbf{y}_k} \mathbf{P}_{\mathbf{y}_k}^{-1}$$
$$\tilde{\mathbf{y}}_k = \mathbf{y}_k - \hat{\mathbf{y}}_k$$

4: State update:

$$\hat{\boldsymbol{\mathcal{X}}}_{k|k} = \hat{\boldsymbol{\mathcal{X}}}_{k|k-1} \exp\left(\left[\mathbf{K}\tilde{\mathbf{y}}_k\right]^{\wedge}\right)$$
$$\mathbf{P}_{k|k}^{-} = \mathbf{P}_{k|k-1} - \mathbf{K}\mathbf{P}_{\mathbf{y}_k}\mathbf{K}^{\top}$$
$$\mathbf{P}_{k|k} = \boldsymbol{\mathcal{J}}_{\mathsf{R}}(\mathrm{ad}\left(\left[\mathbf{K}\tilde{\mathbf{y}}_k\right]^{\wedge}\right))\mathbf{P}_{k|k}^{-}\boldsymbol{\mathcal{J}}_{\mathsf{R}}(\mathrm{ad}\left(\left[\mathbf{K}\tilde{\mathbf{y}}_k\right]^{\wedge}\right))^{\top}$$

Outputs: $\hat{\boldsymbol{\mathcal{X}}}_{k|k}$, $\mathbf{P}_{k|k}$

5.3.4 Preliminary results: application of the optimal quantization to the UT on Lie groups

The Unscented Transform developed thereby can bridge the gap between the propagated covariances $\mathbf{P}_k^{[\emptyset]}$ and $\mathbf{P}_k$ (observed in Section §5.1). Within the same numerical experiments, we demonstrate the performance of the three Unscented transformations (UT1, UT2, CT) given by the parameters of Table 2.1, where each one of them is enhanced by optimal quantization. For this set of experiments, we consider quadratic (p = 2) optimal quantization. The results are provided hereafter.

Focusing on the UT1 (Figures 5.4 and 5.5), the proposed UT1+OQ method achieves, on average, a lower approximation error than the standard UT1. For small initial covariance values $\mathbf{P}_{k-1}$, both methods yield comparable errors. However, as the initial covariance increases, the incorporation of optimal quantization becomes beneficial, making UT1+OQ particularly relevant in regimes characterized by large initial uncertainties. This improvement, however, holds only in an average sense. Indeed, UT1+OQ exhibits larger fluctuations than UT1, with variability increasing as the initial covariance grows. This behavior directly stems from the use of a stochastic gradient descent algorithm. The random sampling inherent to this approach introduces variability in the sigma-point placement, which in turn leads to fluctuations in the error of $\mathbf{P}_k^{UT1+OQ}$. Despite this limitation, the mean error ε is effectively reduced as a result of minimizing the quadratic distortion $D_{2d'+1}^2$ (cf. Figure 5.6), which implies a reduction in the discrepancy between

the true empirical distribution and its quantized approximation, as stated in Proposition 5.8. Recall that minimizing the distortion can be interpreted as minimizing the expected value of the quantization error. Consequently, even though individual realizations of the approximation $\mathbf{P}_k^{UT1+OQ}$ may be better than, equal to, or worse than $\mathbf{P}_k^{UT1}$, the method remains superior on average. An analysis of the variance errors for each state variable (Figure 5.5) further indicates that the angular component benefits the most from the use of optimal quantization.

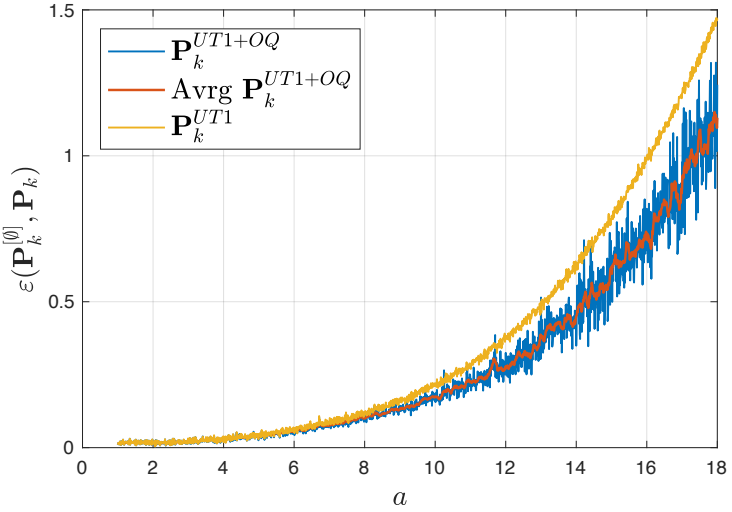


Figure 5.4: Error $\varepsilon(\mathbf{P}_k^{UT1}, \mathbf{P}_k)$, as a function of a ($b = 3.1623$). UT1 is compared to its counterpart augmented with optimal quantization, whose parameters for these results are $\nu = 10$, $n_{max} = 10000$ (CLVQ is applied on the axes of θ, ρ_1 and ρ_2 only).

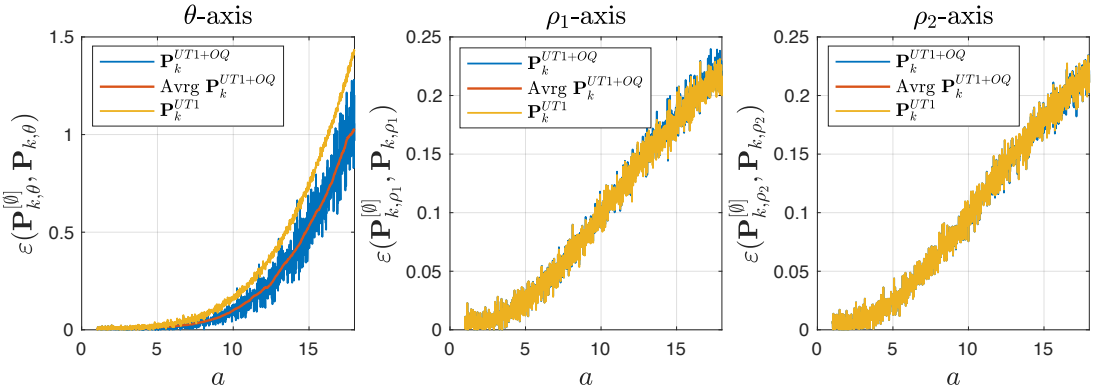
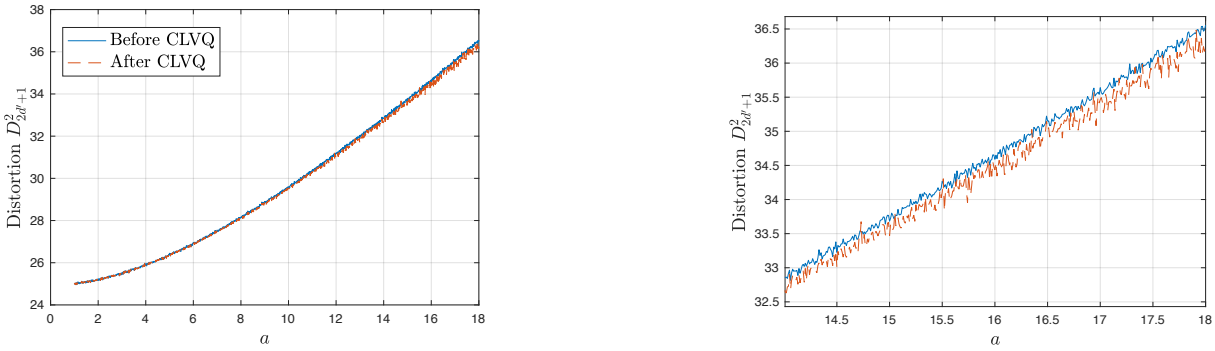


Figure 5.5: Error $\varepsilon(\mathbf{P}_k^{UT1}, \mathbf{P}_k)$, as a function of a ($b = 3.1623$), of the propagated variance of θ (left), ρ_1 (middle) and ρ_2 (right). UT1 is compared to its counterpart augmented with optimal quantization, whose parameters for these results are $\nu = 10$, $n_{max} = 10000$ (CLVQ is applied on the axes of θ, ρ_1 and ρ_2 only).



(a) Full picture

(b) Zoom

Figure 5.6: Distortion (computed using (5.10)) of the set of sigma-points (UT1) as a function of a, before and after applying the CLVQ.

Focusing on the CT (Figures 5.7 and 5.8), and using the same CLVQ parameters as for UT1, the CT+OQ approach exhibits a similar improvement pattern. An exception is observed in the range $a \in [13, 14]$, where the standard CT slightly outperforms its quantized counterpart. Beyond this interval, the CT+OQ method again yields lower approximation errors than the standard CT. Moreover, when optimal quantization is applied to CT, the resulting error displays reduced variability compared to UT1+OQ, despite both methods sharing identical CLVQ parameters. This behavior can be attributed to the initial sigma-point configuration prior to CLVQ refinement: with the scaling parameters of UT1, sigma-points are less dispersed than those generated by CT. As a result, variability induced by the stochastic optimization is less pronounced when sigma-points are already sufficiently spread. Finally, an examination of the variance error plots in Figure 5.8 indicates that the angular component again benefits the most from the incorporation of optimal quantization. In addition, the new set of of sigma-points also exhibits a reduced distortion compared to the former set (Figure 5.9), as expected.

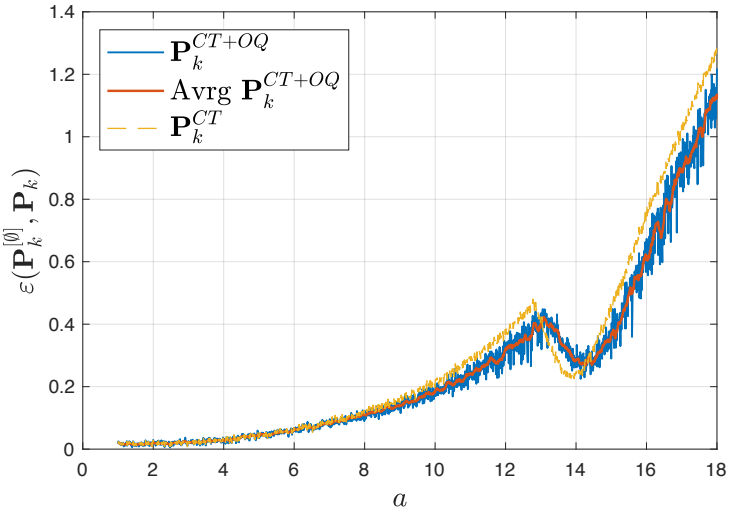


Figure 5.7: Error $\varepsilon(\mathbf{P}_k^{CT}, \mathbf{P}_k)$, as a function of a ($b = 3.1623$). CT is compared to its counterpart augmented with optimal quantization, whose parameters for these results are $\nu = 10$, $n_{max} = 10000$ (CLVQ is applied on the axes of θ, ρ_1 and ρ_2 only).

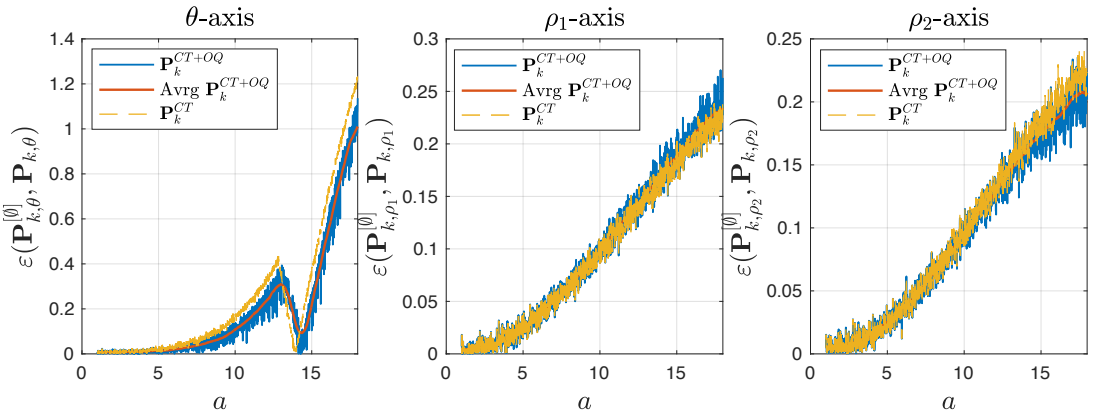
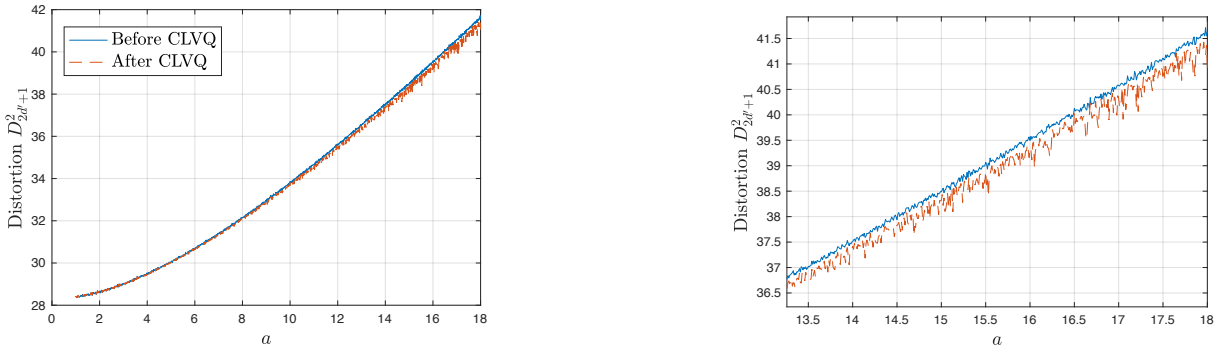


Figure 5.8: Error $\varepsilon(\mathbf{P}_k^{CT}, \mathbf{P}_k)$, as a function of a ($b = 3.1623$), of the propagated variance of θ (left), ρ_1 (middle) and ρ_2 (right). CT is compared to its counterpart augmented with optimal quantization, whose parameters for these results are $\nu = 10$, $n_{max} = 10000$ (CLVQ is applied on the axes of θ, ρ_1 and ρ_2 only).



(a) Full picture

(b) Zoom

Figure 5.9: Distortion (computed using (5.10)) of the set of sigma-points (CT) as a function of a, before and after applying the CLVQ.

Focusing on the UT2 (Figures 5.10 and 5.11), it was first necessary to adapt the CLVQ parameterization, as the initial sigma-point placement induced by the UT2 scaling parameters was excessively concentrated. Indeed, retaining the same CLVQ parameters as those used for UT1 and CT led to excessively large errors, reaching approximately four orders of magnitude. This observation highlights the strong dependence of the CLVQ parameterization on the initial sigma-point configuration. Consequently, the learning-rate parameter was set to $\nu = 100$ in order to ensure sufficiently small update steps during the sigma-point refinement process. In addition, CLVQ was applied only along the ρ_1 and ρ_2 axes, since previous results for UT1 and CT indicated that the angular component is significantly more sensitive to optimal quantization than the translational components. As a result, in this configuration, the reduction in the full covariance error is primarily driven by improvements in the translational variables, as illustrated in the middle and right plots of Figure 5.11. Although large fluctuations are observed in these plots (due to the tight concentration of sigma-points along each axis), the magnitude of ε remains insufficient to induce pronounced variations in the error of the full covariance. This suggests that, in this setting, the angular component continues to be the dominant

source of approximation error. Besides, it is seen from Figure 5.12, that the distortion has been poorly reduced, explaining the small gap between UT2 and UT2+QO in Figure 5.10.

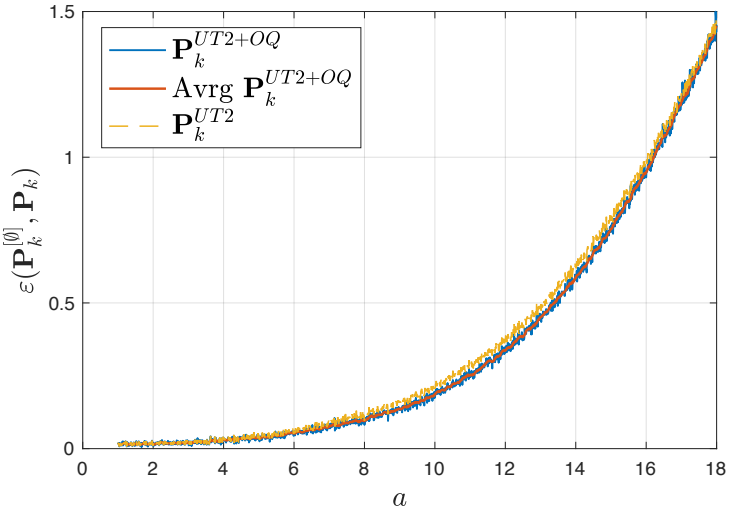


Figure 5.10: Error $\varepsilon(\mathbf{P}_k^{UT2}, \mathbf{P}_k)$, as a function of a ($b = 3.1623$). UT2 is compared to its counterpart augmented with optimal quantization, whose parameters for these results are $\nu = 100$, $n_{max} = 10000$ (CLVQ is applied on the axes of ρ_1 and ρ_2 only).

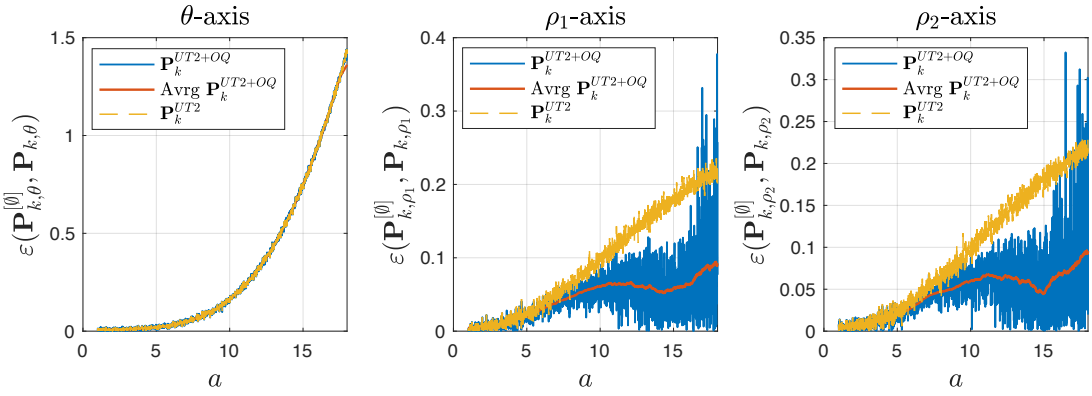
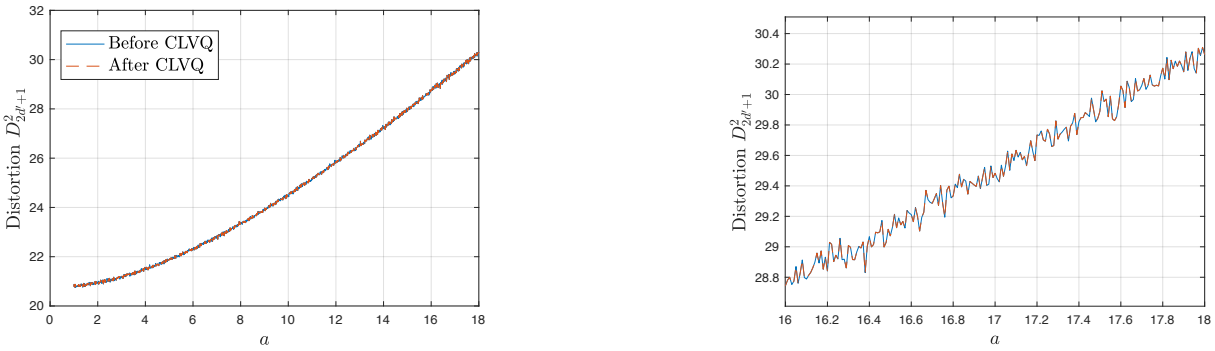


Figure 5.11: Error $\varepsilon(\mathbf{P}_k^{UT2}, \mathbf{P}_k)$, as a function of a ($b = 3.1623$), of the propagated variance of θ (left), ρ_1 (middle) and ρ_2 (right). UT2 is compared to its counterpart augmented with optimal quantization, whose parameters for these results are $\nu = 100$, $n_{max} = 10000$ (CLVQ is applied on the axes of ρ_1 and ρ_2 only).



(a) Full picture

(b) Zoom

Figure 5.12: Distortion (computed using (5.10)) of the set of sigma-points (UT2) as a function of a, before and after applying the CLVQ.

In this chapter, we focused on the development of a solution, based on optimal quantization, that enables the accurate propagation of a normally-distributed random vector. Although the method has been detailed in the case of Unscented Tranforms on Lie groups, it can be applied to any propagation into a given nonlinear function, as long as the random vector is assumed Gaussian. For propagation on Lie groups in particular, we presented this solution as a way to enhance the estimation performance of the UKF-LG, without questionning the validity of the Lie-theoretic covariance and the concentrated Gaussian assumption. Nevertheless, optimal quantization is not tied to a specific probability model, and Unscented Transforms based on alternative distributions could equally benefit from the ideas developed in this chapter. In particular, the vMG-UKF-LG introduced in Chapter 4 could potentially be enhanced through optimal quantization. But it would require further developments, especially an adaptation to circular metrics. This direction has not been explored in the present thesis but constitutes a promising avenue for future work. In the meantime, the next chapter proposes to experimentally validate the proposed OQ-UKF-LG algorithm.

Chapter 6

Experimental validations

Contents

6.1 Experimental setup for simulated data . . . 110
6.1.1 Observation models . . . 110
6.1.2 Description of a filtering scenario with simulated data . . . 111
6.1.3 Monte-Carlo simulations . . . 113
6.2 Validations of the vMG-UKF-LG on simulated data . . . 115
6.2.1 Convergence of the vMG-UKG-LG without noise . . . 116
6.2.2 Sensitivity to process and measurement noise . . . 118
6.2.3 Additional results . . . 126
6.3 Validations of the OQ-UKF-LG on simulated data . . . 130
6.3.1 Sensitvity to overestimation of **Q** and **R** . . . 130
6.3.2 Sensitivity to underestimation of **Q** and **R** . . . 134
6.3.3 Performance / Computation cost trade-off . . . 135
6.4 Experimental setup for real data . . . 137
6.4.1 Presentation of the datasets . . . 137
6.4.2 Presentation of the experimental setup . . . 138
6.5 Validations of the vMG-UKF-LG on real data . . . 140
6.5.1 Difficult scenario . . . 140
6.5.2 Medium scenario . . . 145
6.5.3 Easy scenario . . . 146

This chapter presents a comprehensive set of results obtained using the different algorithmic methods developed in Chapters 4 and 5, with evaluations conducted both on simulated data and on real-world datasets. The chapter first relies on controlled simulations, which allow the noise characteristics to be precisely specified and the underlying probabilistic assumptions of each filter to be rigorously tested. It then extends the analysis to real data, drawn from a publicly available dataset, in order to assess the practical relevance and robustness of the proposed methods in realistic navigation scenarios.

6.1 Experimental setup for simulated data

We present the experimental framework used to validate the proposed algorithms, namely the vMG-UKF-LG and the OQ-UKF-LG. In particular, the validation is carried out on simulated data in order to retain explicit control over the noise assumptions underlying each method. This necessitates the design of a dedicated simulation framework capable of generating trajectories and measurements consistent with the prescribed noise models.

We consider a rigid-body $\mathcal{B}$ in a fixed-reference frame $\mathcal{F}_{\mathcal{I}} = (O, \boldsymbol{e}_{\mathcal{I}_1}, \boldsymbol{e}_{\mathcal{I}_2}, \boldsymbol{e}_{\mathcal{I}_3})$ (cf. Figure 6.1). At time instant k, the pose $\boldsymbol{\chi}_k \in \boldsymbol{SE}(3)$ of the rigid-body is characterized by its attitude encoded in the rotation matrix $\boldsymbol{\mathcal{R}}_k \in \boldsymbol{SO}(3)$, and its position encoded in the vector $\boldsymbol{x}_k \in \mathbb{R}^3$ expressed in the inertial frame $\mathcal{F}_{\mathcal{I}}$. The propagation model from $k-1$ to k is given by:

$$\boldsymbol{\chi}_k = \boldsymbol{f}(\boldsymbol{\chi}_{k-1}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1}, \tau) = \boldsymbol{\chi}_{k-1} \exp(\mathbf{u}_{k-1}^{\wedge}\tau) \exp(\mathbf{w}_{k-1}^{\wedge}\tau)$$

with τ being the timestep, $\mathbf{u}_k$ the velocity inputs, and $\mathbf{w}_k$ the noise acting on the inputs.

6.1.1 Observation models

Regarding the observation model $\mathbf{y} = \mathbf{h}(\boldsymbol{\chi}, \mathbf{v})$, we evaluate the algorithms on two traditional types of observations: first, a simple observation model that provides cartesian positions, modeling for example noisy GPS measurements. Secondly, a processed range-and-bearing observation function, modeling for example measurements coming from noisy lidar data.

Cartesian position measurements

In terms of mathematics, the first observation model is expressed as

$$\begin{bmatrix} \mathbf{y}_k \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{h}_{\text{gps}}(\boldsymbol{\chi}_k, \mathbf{v}_k) \\ 1 \end{bmatrix} = \boldsymbol{\chi}_k \begin{bmatrix} \mathbf{0}_{3\times 1} \\ 1 \end{bmatrix} + \begin{bmatrix} \mathbf{v}_k \\ 0 \end{bmatrix} = \begin{bmatrix} \boldsymbol{x}_k + \mathbf{v}_k \\ 1 \end{bmatrix} \tag{6.1}$$

In this case, we have the $\mathbb{R}^3$-valued noise vector $\mathbf{v}_k \sim \mathcal{P}(\mathbf{v}_k; \mathbf{0}, \mathbf{R}_k)$ with $\mathbf{R}_k \in \mathbb{R}^{3\times 3}$, where $\mathcal{P}$ denotes any probability distribution.

Range-and-bearing measurements

Regarding the second observation model, a certain number of N_l distinct landmarks are introduced, whose absolute positions are known and denoted:

$$\boldsymbol{l}^{(i)} = \begin{bmatrix} l_1^{(i)} \\ l_2^{(i)} \\ l_3^{(i)} \end{bmatrix}, \quad \forall i = 1, \ldots, N_l \tag{6.2}$$

Assume that, at time instant k, a point-matching algorithm returns useful informations by processing range-and-bearing measurements, and provides us with the position of all

detected landmarks with respect to the moving body frame $\mathcal{F}_\mathcal{B}$. Hence, we consider the following observation model:

$$\mathbf{y}_k^{(i)} = \mathbf{h}_{\text{lidar}}^{(i)}(\boldsymbol{\chi}_k, \mathbf{v}_k^{(i)}) = \boldsymbol{\mathcal{R}}_k^\top(\boldsymbol{l}^{(i)} - \boldsymbol{x}_k) + \mathbf{v}_k^{(i)} \tag{6.3}$$

with the $\mathbb{R}^3$-valued noise vector $\mathbf{v}_k^{(i)} \sim \mathcal{P}(\mathbf{v}_k^{(i)}; \mathbf{0}, \mathbf{R}_k^{(i)})$, where $\mathbf{R}_k^{(i)} \in \mathbb{R}^{3\times3}$. All measurements $\mathbf{y}_k^{(i)}$ are then stacked into a single measurement vector as:

$$\mathbf{y}_k = \mathbf{h}_{\text{lidar}}(\boldsymbol{\chi}_k, \mathbf{v}_k) = \begin{bmatrix} \mathbf{y}_k^{(1)} \\ \vdots \\ \mathbf{y}_k^{(N_l)} \end{bmatrix} \in \mathbb{R}^{3N_l} \tag{6.4}$$

In this case, we have the $\mathbb{R}^{3N_l}$-valued noise vector $\mathbf{v}_k \sim \mathcal{P}(\mathbf{v}_k; \mathbf{0}, \mathbf{R}_k)$ where $\mathbf{R}_k \in \mathbb{R}^{3N_l\times3N_l}$. This observation function models experiments where absolute positions of the landmarks are not directly available, but their relative positions with respect to the body frame are available (Figure 6.1).

For the experimental validation on simulated data, we restricted ourselves to these two observation models, summarized as follows:

$$\mathbf{y}_k = \mathbf{h}(\boldsymbol{\chi}_k, \mathbf{v}_k) = \begin{cases} \mathbf{h}_{\text{gps}}(\boldsymbol{\chi}_k, \mathbf{v}_k) \in \mathbb{R}^3, & \mathbf{R}_k \in \mathbb{R}^{3\times3} \\ \mathbf{h}_{\text{lidar}}(\boldsymbol{\chi}_k, \mathbf{v}_k) \in \mathbb{R}^{3N_l}, & \mathbf{R}_k \in \mathbb{R}^{3N_l\times3N_l} \end{cases} \tag{6.5}$$

We denote by f_{obs} (Hz), the frequency at which a measurement $\mathbf{y}$ is available for the Kalman correction step.

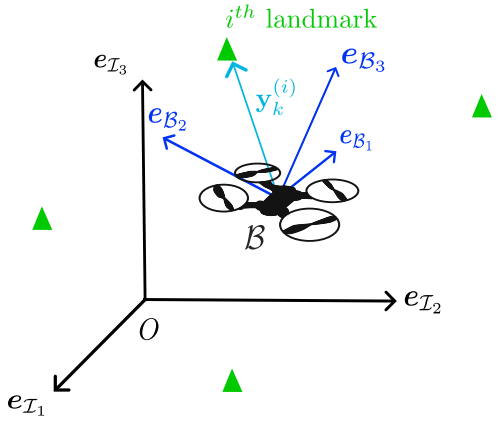


Figure 6.1: Example of a pose estimation problem with landmarks detection. The position of landmarks with respect to the fixed inertial frame $\mathcal{F}_\mathcal{I}$ is known, but at time instant k, the system measures only the position of landmarks with respect to its own reference frame $\mathcal{F}_\mathcal{B}$.

6.1.2 Description of a filtering scenario with simulated data

Generate a ground truth

When performing experiments with simulated data, we consider a constant timestep τ. Hence, the time update is performed at frequency $f_\tau = 1/\tau$ Hz. To generate a trajectory over a horizon $k \in [1; k_{end}]$, we use a sequence of nominal inputs $\{\mathbf{u}_k\}_{k=1,\dots,k_{end}}$ perturbed

by a sequence of noise $\{\mathbf{w}_k\}_{k=1,\dots,k_{end}}$, where $\mathbf{w}_k$ is generated from an arbitrarily chosen probability density $\mathcal{P}(\mathbf{w}_k; \mathbf{0}, \mathbf{Q}_k)$. At $k = 0$, we consider that the real state is initialized at some pose $\boldsymbol{\chi}_0$, and that the estimated state $\hat{\boldsymbol{\chi}}_{0|0}$ is initialized around the real pose with some error $\tilde{\boldsymbol{\chi}}_0$ such that

$$\tilde{\boldsymbol{\chi}}_0 = \boldsymbol{\chi}_0^{-1}\hat{\boldsymbol{\chi}}_{0|0} \sim \mathcal{P}_{\mathsf{L/R}}(\tilde{\boldsymbol{\chi}}_0; \mathbf{I}, \mathbf{P}_{0|0})$$

where $\mathcal{P}_{\mathsf{L/R}}$ is any arbitrary left or right equivariant distribution on the group, whose mean is located at the identity. The covariance of the estimation is also initialized at some arbitrary value $\mathbf{P}_{0|0}$. A simulated trajectory is generated online by applying the propagation model $\boldsymbol{f}$ at the frequency f_τ, starting from the true pose $\boldsymbol{\chi}_0$, and taking the sequence of nominal inputs $\{\mathbf{u}_k\}_{k=1,\dots,k_{end}}$ perturbed by the sequence of noise $\{\mathbf{w}_k\}_{k=1,\dots,k_{end}}$, where each $\mathbf{w}_k$ is randomly drawn from a pseudo-random variable generator that satisfies the pdf $\mathcal{P}(\mathbf{w}_k; \mathbf{0}, \mathbf{Q}_k)$. This generates the ground truth $\{\boldsymbol{\chi}_k\}_{k=1,\dots,k_{end}}$. For the sequence of nominal inputs $\{\mathbf{u}_k\}_{k=1,\dots,k_{end}}$, we consider the data illustrated in Figure 6.2, with frequency $f_\tau = 100$ Hz.

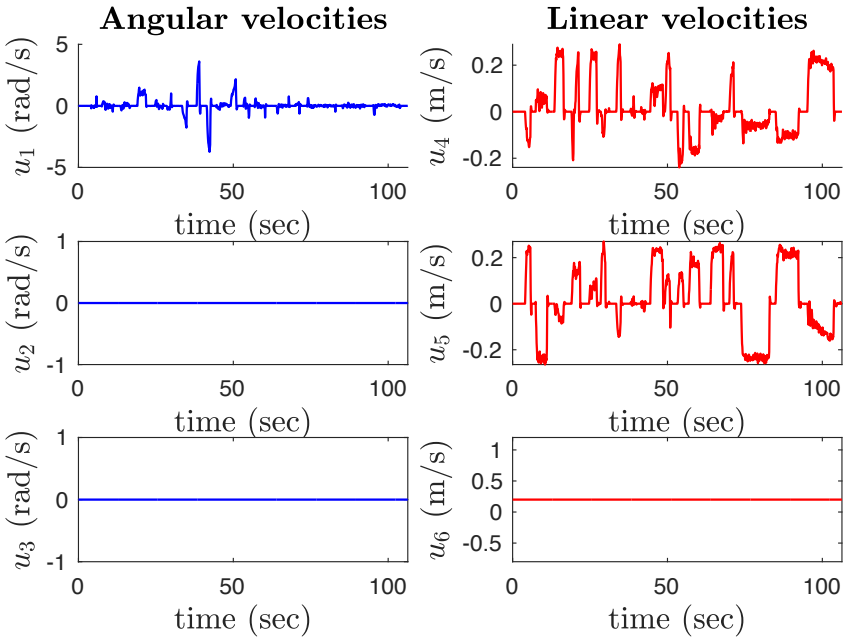


Figure 6.2: Sequence of nominal inputs $\mathbf{u}$ used to generate the nominal trajectory. The data u_1, u_4 and u_5 have been obtained from gyrometer and odometry measurements of a holonomic ground robot moving in a flat surface.

Generate an observation of the true state

Occasionally, a noisy measurement $\mathbf{y}_k$ of the true state $\boldsymbol{\chi}_k$ is observed at the frequency f_{obs}, with $f_{obs} \leq f_\tau$. Depending on the observation model, this partial information of the true state is extracted either from $\mathbf{h}_{\text{gps}}$ or from $\mathbf{h}_{\text{lidar}}$, with a pseudo-random realization $\mathbf{v}_k$ satisfying $\mathcal{P}(\mathbf{v}_k; \mathbf{0}, \mathbf{R}_k)$.

Estimation of the ground truth

In parallel, an a priori estimation of the pose $\boldsymbol{\chi}_k$ (given by $\hat{\boldsymbol{\chi}}_{k|k-1}$ and $\mathbf{P}_{k|k-1}$) runs at the same frequency f_τ, with a model noise parameter $\hat{\mathbf{Q}}_k$, which can be equal (or

not) to the real covariance $\mathbf{Q}_k$ that has been used to generate the pseudo-random vector $\mathbf{w}_k$. Likewise, the measurement noise parameter $\hat{\mathbf{R}}_k$ can be equal (or not) to the real covariance $\mathbf{R}_k$. In general, if we choose to match the noise parameters $(\hat{\mathbf{Q}}_k, \hat{\mathbf{R}}_k)$ with their respective true values $(\mathbf{Q}_k, \mathbf{R}_k)$, then this implies that the Kalman is well-parameterized. In this case, we say that the noise parameters are *consistent* with the true noise. Finally, a correction of the prior (given by $\hat{\boldsymbol{\chi}}_{k|k}$ and $\mathbf{P}_{k|k}$) is performed thanks to the measurement $\mathbf{y}_k$ acquired at the frequency f_{obs}. The whole procedure is illustrated in Figure 6.3.

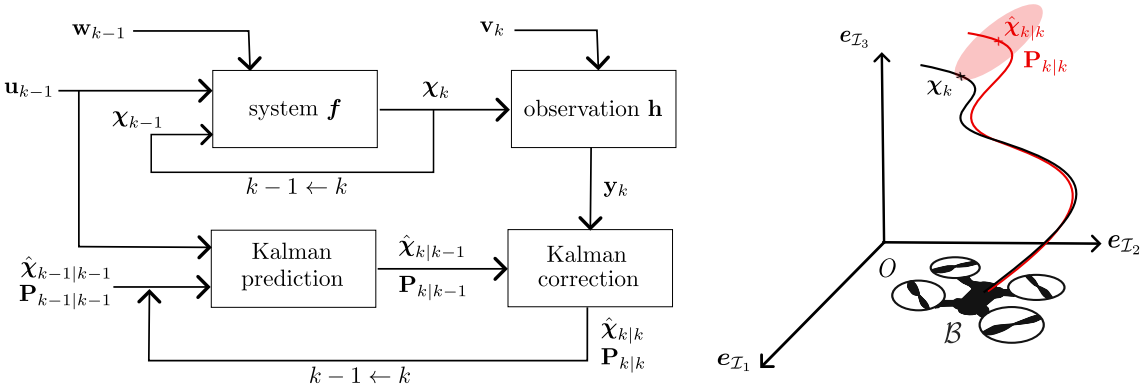


Figure 6.3: State evolution and estimation by Kalman filtering on $\boldsymbol{SE}(3)$. The sequence $\{\mathbf{u}_k\}_{k\geq 1}$ drives the shape of the trajectory. For simulated data, the noise vectors $\mathbf{w}_k$ and $\mathbf{v}_k$ are generated from a pseudo-random number generator with chosen characteristics $\mathbf{Q}_k$ and $\mathbf{R}_k$. The Kalman prediction and correction are parameterized by $\hat{\mathbf{Q}}_k$ and $\hat{\mathbf{R}}_k$, respectively.

This experimental setup essentially simulates a pose estimation scenario of a rigid-body moving in the three-dimensional space, where the true trajectory $\{\boldsymbol{\chi}_k\}_{k=1,\dots,k_{end}}$ is exactly related to the sequence of inputs $\{\exp(\mathbf{u}_k^\wedge \tau)\exp(\mathbf{w}_k^\wedge \tau)\}_{k=1,\dots,k_{end}}$. Hence, we consider that the pair of data $\{\boldsymbol{\chi}_k\,,\, \exp(\mathbf{u}_k^\wedge \tau)\exp(\mathbf{w}_k^\wedge \tau)\}_{k=1,\dots,k_{end}}$ is the ground truth, and that the Kalman filter only has access to the inputs $\mathbf{u}_k$ (without $\mathbf{w}_k$) for its prediction step. In a real experimental scenario, the velocity inputs $\mathbf{u}_k$ can be, for example, given by an IMU, where the components related to the rotation part would be given by processed gyrometer measurements, and the components related to the translation part would be given by processed accelerometer measurements (or simply by an odometer). In real applications, the true velocities of the rigid-body are unknown by the filter, and the IMU can only provide imperfect measurements of these quantities. In our simulator, we therefore consider that the combination of $\mathbf{u}_k$ and $\mathbf{w}_k$ serves as the true velocity of the moving body, and that $\mathbf{u}_k$, taken alone, is what an inertial sensor would give to the estimator.

6.1.3 Monte-Carlo simulations

To evaluate a stochastic observer, we often proceed to Monte-Carlo simulations to assess statistical significance under verified noise assumptions. Monte-Carlo simulations provide a set of trajectory estimates on which we can observe a global pattern about the filter's performances by analyzing statistics. Let us denote by $\square_k^{(i)}$, the i^{th} realization of the quantity $\square$ at time instant k. Monte-Carlo simulations generate N_s realizations of

the ground truth, where the sequence of inputs $\{\mathbf{u}_k\}_{k\geq 1}$ shapes the nominal behaviour, as well as N_s realizations of the estimate and its covariance. The procedure is detailed in Algorithm 9 and Figure 6.4 illustrates possible outputs.

Algorithm 9: N_s Monte-Carlo simulations

Inputs: $\{\mathbf{Q}_k\}_{k=0,\dots,k_{end}}$, $\{\mathbf{R}_k\}_{k=1,\dots,k_{end}}$, $\{\hat{\mathbf{Q}}_k\}_{k=0,\dots,k_{end}}$, $\{\hat{\mathbf{R}}_k\}_{k=1,\dots,k_{end}}$, $\{\mathbf{u}_k\}_{k=0,\dots,k_{end}}$

1: **for** $i = 1$ to N_s **do**
2: Initialize the true state $\boldsymbol{\chi}_0^{(i)}$ and the estimate $(\hat{\boldsymbol{\chi}}_{0|0}^{(i)}, \mathbf{P}_{0|0}^{(i)})$
3: **for** $k = 1$ to k_{end} **do**
4: Generate a realization of the true state: $\boldsymbol{\chi}_k^{(i)} = \boldsymbol{f}(\boldsymbol{\chi}_{k-1}^{(i)}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1}^{(i)})$, with $\mathbf{w}_{k-1}^{(i)} \sim \mathcal{P}(\mathbf{w}_{k-1}; \mathbf{0}, \mathbf{Q}_{k-1})$
5: Generate a realization of the measure: $\mathbf{y}_k^{(i)} = \mathbf{h}(\boldsymbol{\chi}_k^{(i)}, \mathbf{v}_k^{(i)})$, with $\mathbf{v}_k^{(i)} \sim \mathcal{P}(\mathbf{v}_k; \mathbf{0}, \mathbf{R}_k)$
6: Generate a Kalman estimate: $(\hat{\boldsymbol{\chi}}_{k|k}^{(i)}, \mathbf{P}_{k|k}^{(i)})$ with the noise parameters $\hat{\mathbf{Q}}_{k-1}$ and $\hat{\mathbf{R}}_k$ and the observed measure $\mathbf{y}_k^{(i)}$
7: **end for**
8: **end for**

Outputs: $\{\boldsymbol{\chi}_k^{(i)}\}_{k=1,\dots,k_{end}}^{i=1,\dots,N_s}$, $\{\hat{\boldsymbol{\chi}}_{k|k}^{(i)}\}_{k=1,\dots,k_{end}}^{i=1,\dots,N_s}$, $\{\mathbf{P}_{k|k}^{(i)}\}_{k=1,\dots,k_{end}}^{i=1,\dots,N_s}$

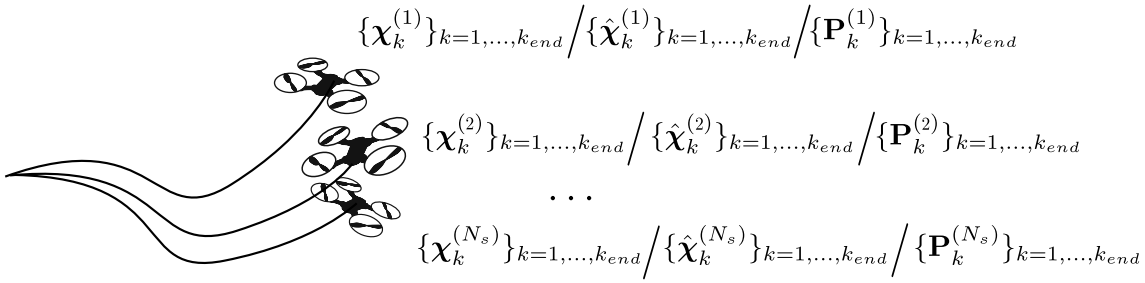


Figure 6.4: Monte-Carlo simulations provide different possible realizations of the same nominal trajectory, where the different realizations are spanned by $\mathcal{P}(\mathbf{w}_k; \mathbf{0}, \mathbf{Q}_k)$ and $\mathcal{P}(\mathbf{v}_k; \mathbf{0}, \mathbf{R}_k)$.

Algorithmic implementation

The simulator used to validate the algorithms on synthetic data was entirely developed from scratch in MATLAB using full script files. All closed-form expressions related to Lie groups (such as exponential and logarithmic maps, adjoint operators, jacobians, etc.) were manually implemented. To ensure numerical stability, all UKF-LG variants were implemented using the square-root formulation [Van der Merwe and Wan, 2001], which relies on a Cholesky factorization of the covariance matrix.

Evaluation metrics

From the Monte-Carlo outputs, we empirically assess statistical significance by resorting to usual metrics such as the Root Mean Square Error (RMSE), Normalized Estimation Error Squared (NEES), and other relevant metrics (see Appendix B.4 for further details).

6.2 Validations of the vMG-UKF-LG on simulated data

In this section, we provide a set of numerical results to validate the (Left)-vMG-UKF-LG. The benchmark algorithm (from which the proposed vMG-UKF-LG is compared) is the standard (Left)-UKF-LG. We perform Monte-Carlo simulations (Algorithm 9 with the simulator architecture described by Figure 6.3), while maintaining control of the noise acting on the system. To assess the performance of the vMG-UKF-LG under correct noise assumptions, the input noise $\mathbf{w}_k$ follows a von Mises-Gaussian distribution with uncorrelated angular/linear variables. Namely, we artificially add noise on the velocity input $\mathbf{u}_k$ with the following specifications:

$$\mathbf{w}_k \sim \mathcal{VMG}(\boldsymbol{\omega}_\mathbf{w}, \boldsymbol{\rho}_\mathbf{w}; \boldsymbol{\kappa}_\mathbf{u}, \mathbf{Q}) \quad \text{with} \quad \mathbf{Q} = \begin{bmatrix} \mathbf{Q}_{\boldsymbol{\omega}} & \mathbf{0} \\ \mathbf{0} & \mathbf{Q}_{\boldsymbol{\rho}} \end{bmatrix}, \quad \forall k \tag{6.6}$$

with stationary diagonal matrices $\mathbf{Q}_{\boldsymbol{\omega}}$ and $\mathbf{Q}_{\boldsymbol{\rho}}$. Regarding the noise acting on the measurement $\mathbf{y}_k$, we take $\mathbf{v}_k \sim \mathcal{N}(\mathbf{v}_k; \mathbf{0}, \mathbf{R})$ $\forall k$.

Hence, components of $\boldsymbol{\omega}_\mathbf{w}$ are generated from a pseudo-random generator satisfying a von Mises distribution with arbitrarily fixed concentration parameters $\boldsymbol{\kappa}_\mathbf{u}$. At each time k, we use the function "circ_vmpdf(0,κ_{u_i},1)" of the Matlab toolbox "CircStat" [Berens, 2009] to generate pseudo-random variables from the parameter κ_{u_i}, $i = \{1, 2, 3\}$. These parameters are set indirectly by first parameterizing $\mathbf{Q}_{\boldsymbol{\omega}}$, and extracting the corresponding κ_{u_i} from Q_{ω_i} using the relationship

$$\kappa_{u_i} = \vartheta(Q_{\omega_i}) = c_1 \exp(c_2 Q_{\omega_i}) + c_3 \exp(c_4 Q_{\omega_i}) \tag{6.7}$$

as explained in remark 4.5 of Chapter 4. The coefficients c_1, c_2, c_3 and c_4 are obtained from data interpolation, using Matlab's Curve Fitter tool. In particular, we have generated $N = 7000$ samples, where each sample $s = \{1, \dots, N\}$ contains 10^4 normally-distributed particles with dispersion σ_s^2. From each sample, we computed the corresponding concentration parameter κ_s using the function "circ_vmpar(sample)" of the CircStat toolbox. The interpolation produced the following coefficients (with their 95% confidence bounds):

$$\begin{aligned} c_1 &= 48.05 \quad (47.04,\ 49.07), \quad & c_2 &= -19.73 \quad (-20.17,\ -19.29) \\ c_3 &= 4.882 \quad (4.755,\ 5.009), \quad & c_4 &= -0.9679 \quad (-0.9966,\ -0.9392) \end{aligned} \tag{6.8}$$

As already stated in remark 4.5, for very small values of σ_s, it is undesirable for the concentration parameter κ_s to approach singular values. Therefore, the interpolation scheme was parameterized so as to control the curvature of ϑ, preventing excessively abrupt variations for small σ_s. The Figure 6.5 shows the shape of the approximation (6.7) with the coefficients (6.8). Additionally, we use the same function ϑ to extract the concentration parameter κ_i of the angular state from its associated variance $P_{\omega_i\omega_i}$. Note that the approximation ϑ can potentially overestimate the dispersion, but should avoid singular cases.

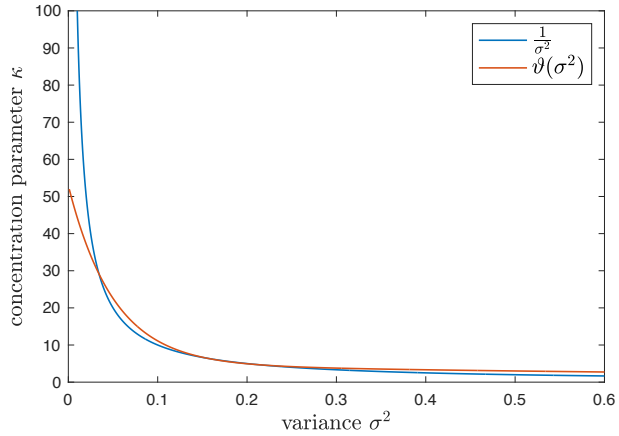


Figure 6.5: Approximation of the function $\vartheta(\sigma^2)$ relating a variance σ^2 with a concentration parameter κ, using an exponential model of the form (6.7).

6.2.1 Convergence of the vMG-UKG-LG without noise

We first evaluate if the proposed vMG-UKF-LG does converge from an initial state error in a nearly noise-free setting, that is, with the process and measurement noise covariances $\mathbf{Q}$ and $\mathbf{R}$ set to very small values. Meanwhile, we exploit this experiment to compare the two approximation schemes for the concentration parameter, namely $\kappa = \vartheta(\text{variance})$ and $\kappa = 1/\text{variance}$. To this end, we perform a filtering simulation with the configuration detailed below.

Hyper-parameters:

Monte-Carlo: $N_s = 1$
Measurement model: $\mathbf{h}_{\text{lidar}}$ with 3 landmarks $\boldsymbol{l}^{(1)} = [0, -0.5, 0]^\top$, $\boldsymbol{l}^{(2)} = [4, 1, 0]^\top$ and $\boldsymbol{l}^{(3)} = [0, 2, 0]^\top$
Measurement frequency: $f_{obs} = 1$ Hz
Initial covariance: $\mathbf{P}_{0|0} = \text{diag}\left(\left[(\frac{45\times\pi}{180})^2, (\frac{45\times\pi}{180})^2, (\frac{45\times\pi}{180})^2, 0.5^2, 0.5^2, 0.5^2\right]\right)$
The estimated pose $\hat{\boldsymbol{\chi}}_{0|0}$ is initialized with an error $\tilde{\boldsymbol{\chi}}_0 \sim \mathcal{VMG}_{\mathsf{L}}(\boldsymbol{\kappa}_{0|0}, \mathbf{P}_{0|0})$.
Input noise: $\mathbf{Q} = \text{diag}\left(\left[(\frac{0.01\times\pi}{180})^2, (\frac{0.01\times\pi}{180})^2, (\frac{0.01\times\pi}{180})^2, (0.01)^2, (0.01)^2, (0.01)^2\right]\right)$
Input uncertainty parameter: $\hat{\mathbf{Q}} = \text{diag}\left(\left[(\frac{\pi}{180})^2, (\frac{\pi}{180})^2, (\frac{\pi}{180})^2, (0.05)^2, (0.05)^2, (0.05)^2\right]\right)$
Output noise: $\mathbf{R}^{(i)} = \text{diag}\left([0.001^2, 0.001^2, 0.001^2]\right), \forall i = \{1, 2, 3\}$
Output uncertainty parameter: $\hat{\mathbf{R}}^{(i)} = \text{diag}\left([0.01^2, 0.01^2, 0.01^2]\right)$

Results:

A first simulation yields the results shown in Figure 6.6, which illustrates the convergence behavior of the vMG-UKF-LG when using either the exponential approximation ϑ (6.7) or the inverse-variance approximation. The plots report the norms of the log-rotation and log-position errors, defined as

$$\text{log-rotation error} = \log(\boldsymbol{\mathcal{R}}_k^{-1}\hat{\boldsymbol{\mathcal{R}}}_k)^\vee$$

$$\text{log-position error} = \log(\boldsymbol{\chi}_k^{-1}\hat{\boldsymbol{\chi}}_k)^\vee_{\boldsymbol{\rho}}$$

where $\log(\cdot)^{\vee}_{\boldsymbol{\rho}}$ denotes the restriction of the logarithmic map to the linear components. For this single realization, the inverse-variance approximation appears to provide superior convergence properties compared to the approximation based on ϑ. In particular, for the log-rotation error (left subfigure), the vMG-UKF-LG employing ϑ almost converges to zero, and exhibits noticeable fluctuations in the steady-state regime, whereas the inverse-variance formulation converges more smoothly, with no visible oscillations, and a reduced error. A similar behavior is observed for the log-position error (right subfigure): the vMG-UKF-LG based on ϑ fails to converge to zero, while the inverse-variance approximation achieves convergence without apparent difficulty. Based solely on this experiment, one might therefore be inclined to favor the inverse-variance approximation for extracting the concentration parameter. However, further independent simulations indicate that such a choice should be avoided in ulterior experiments.

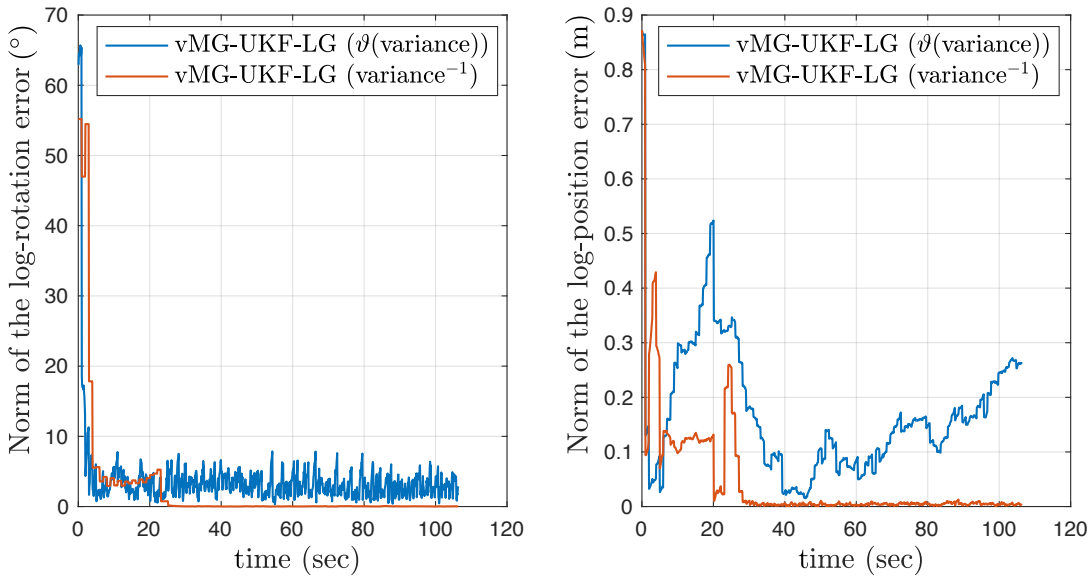


Figure 6.6: Evolution of the log-Euclidean error as a function of time.

Indeed, Figure 6.7 reports the results of another independent simulation conducted under the same noise configurations. In this case, although the vMG-UKF-LG based on the inverse-variance approximation achieves better steady-state convergence (approximately after 40 seconds), it exhibits pronounced difficulties during the transient phase, suggesting a strong sensitivity to initial estimation errors. By contrast, the vMG-UKF-LG based on ϑ shows reduced sensitivity to initial conditions and consistently greater robustness during the transient regime, albeit at the cost of a slightly reduced accuracy in the steady state regime.

To avoid such singularities, we therefore rely, at this stage, on the numerical approximation ϑ. Several alternative approximation models were investigated during the course of this thesis. In particular, a power-law model of the form $\kappa = \vartheta(x) = c_1 x^{c_2} + c_3$ can be employed, providing a closer approximation to $\kappa = x^{-1}$ than the exponential model (6.7). By parameterizing the interpolation method such that $c_1 \neq 1$, $c_2 \neq -1$, and $c_3 \neq 0$

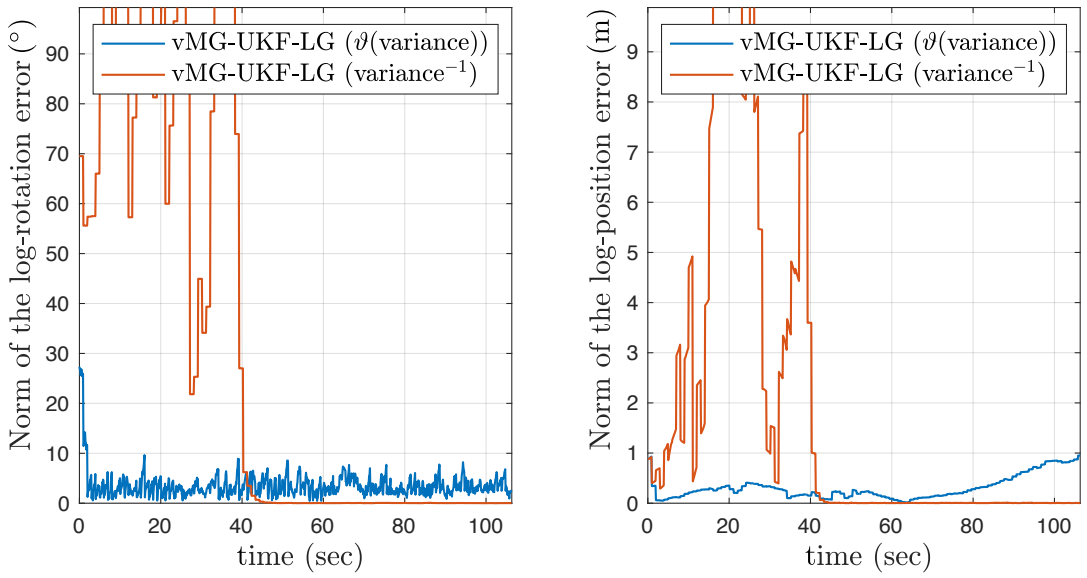


Figure 6.7: Evolution of the log-Euclidean error as a function of time.

(while remaining sufficiently close to these values) the occurrence of singularities can be mitigated. However, after considering the full set of experimental results conducted during the thesis, the exponential model (6.7) was ultimately retained to present the results of this manuscript.

In the next subsections, we assess the performance of the vMG-based UKF-LG against the Gaussian-based UKF-LG in the presence of high noise.

6.2.2 Sensitivity to process and measurement noise

We first investigate the sensitivity of the filters to an increase in the process noise $\mathbf{w}$, whose magnitude is parameterized by the covariance matrix $\mathbf{Q}$. Throughout these experiments, we set $\hat{\mathbf{Q}} = \mathbf{Q}$ (which implicitly implies $\hat{\boldsymbol{\kappa}}_{\mathbf{u}} = \boldsymbol{\kappa}_{\mathbf{u}}$ for the vMG filter) and $\hat{\mathbf{R}} = \mathbf{R}$, so that the vMG-UKF-LG remains correctly parameterized. The analysis begins by increasing the noise levels associated with the first angular component Q_{ω_1} and the three translational components Q_{ρ_1}, Q_{ρ_2}, Q_{ρ_3} by means of a scaling coefficient a. The remaining angular components, Q_{ω_2} and Q_{ω_3}, are kept unchanged so as to not influence the estimation task. The experimental configuration is therefore defined as follows.

Hyper-parameters:

Monte-Carlos: $N_s = 100$
Measurement model: $\mathbf{h}_{\text{lidar}}$
Measurement frequency: $f_{obs} = 1$ Hz
Initial covariance: $\mathbf{P}_{0|0} = \text{diag}\left(\left[(\frac{\pi}{180})^2, (\frac{\pi}{180})^2, (\frac{\pi}{180})^2, 0.01^2, 0.01^2, 0.01^2\right]\right)$
Input noise: $\mathbf{Q} = \text{diag}\left(\left[(a\frac{\pi}{180})^2, (\frac{0.1\pi}{180})^2, (\frac{0.1\pi}{180})^2, (a \times 0.1)^2, (a \times 0.1)^2, (a \times 0.05)^2\right]\right)$
Input uncertainty parameter: $\hat{\mathbf{Q}} = \mathbf{Q}$
Output noise: $\mathbf{R}^{(i)} = \text{diag}\left([0.05^2, 0.05^2, 0.05^2]\right)$, $\forall i = \{1, 2, 3\}$
Output uncertainty parameter: $\hat{\mathbf{R}}^{(i)} = \mathbf{R}^{(i)}$

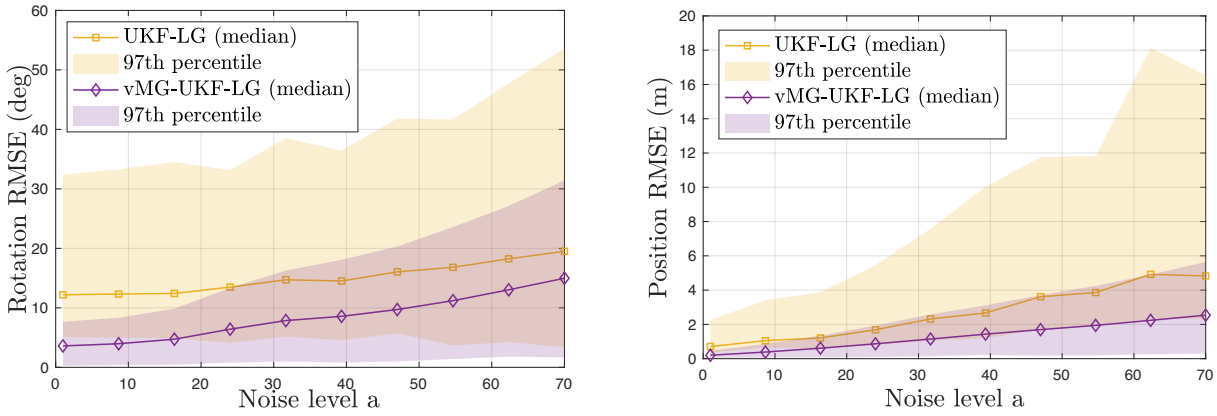
RMSE results:

For a given noise level a and each Monte-Carlo run, we compute the RMSE over the entire trajectory as follows

$$\text{Attitude RMSE} = \sqrt{\frac{1}{k_{end}} \sum_{k=1}^{k_{end}} || \log(\boldsymbol{\mathcal{R}}_k^{-1} \hat{\boldsymbol{\mathcal{R}}}_k)^\vee ||^2}$$

$$\text{Position RMSE} = \sqrt{\frac{1}{k_{end}} \sum_{k=1}^{k_{end}} || \log(\boldsymbol{\chi}_k^{-1} \hat{\boldsymbol{\chi}}_k)_{\boldsymbol{\rho}}^\vee ||^2}$$

For both filters, the RMSE median and its confidence bounds (97%), are plotted in Figure 6.8, as a function of the noise level a. We see from these results that the vMG-UKF-LG



(a) RMSE in rotation as a function of a.

(b) RMSE in position as a function of a.

Figure 6.8: RMSE in rotation (left plot) and in position (right plot), as a function of the noise level a. For each noise level a, the median of the RMSE is plotted along with its confidence interval.

exhibits reduced errors compared to its standard counterpart for both the attitude and position. Indeed, the RMSEs of the vMG-based filter are consistently smaller than those of the Gaussian-based filter. For example, for $a = 1$, the median rotation RMSE of the vMG-UKF is approximately 4°, while the median rotation RMSE of the UKF-LG is approximately 12.5°. For $a = 70$, both RMSEs have increased ($\approx 15°$ for the vMG-UKF and $\approx 19.5°$ for the UKF-LG). The same pattern appears for the position RMSE. Besides, the vMG-UKF seems to exhibit better robustness (whether for rotation or for position), as indicated by the magnitude of its 97^{th} quantile, which proves to be constantly inferior to those of the UKF-LG. This may indicates that the vMG-UKF is insensitive to high process noise in this case.

Consistency results:

These RMSE results can be explained by the fact that, when the input noise follows a von Mises–Gaussian distribution, the standard UKF-LG fails to represent the true uncertainty of the random variables, as it relies on a purely Gaussian probability model. In contrast, when appropriately parameterized, the vMG-UKF is able to capture these uncertainties more accurately. An empirical validation of this claim is provided through

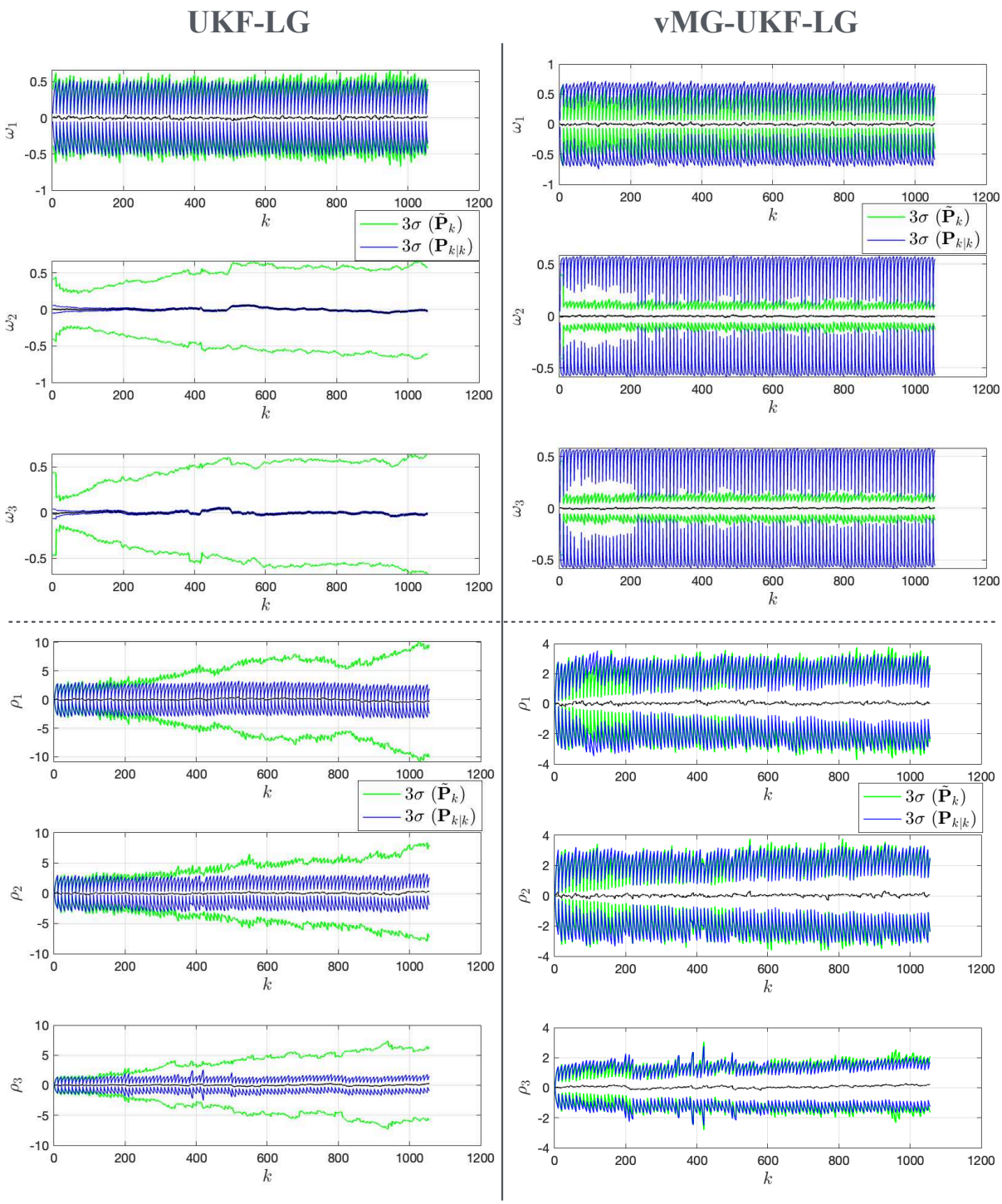


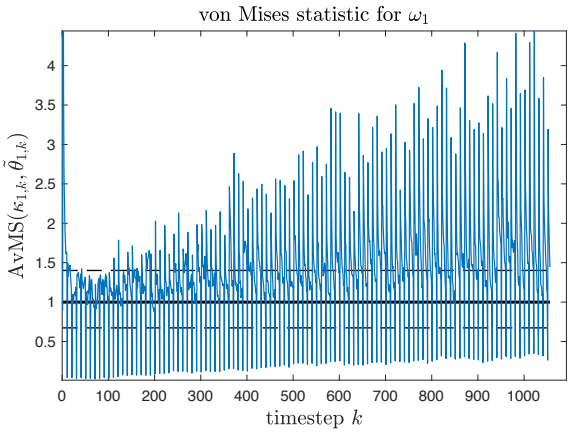
Figure 6.9: Estimation error as a function of time k. The green curves are the actual 3σ standard deviations of the estimation error. The blue curves are the variances given by the filter (averaged over the N_s simulations) in terms of 3σ standard deviation. The black curve is the mean of the estimation error. The reported results are obtained from Monte-Carlo runs corresponding to the noise level $a = 31.7$.

a consistency[1] analysis shown in Figure 6.9. This figure depicts the evolution of the empirical variance of the estimation errors $\tilde{\mathbf{P}}_k$ (green bounds) alongside the average variance $\mathbf{P}_k$ predicted by the estimator (blue bounds), both expressed as 3σ confidence intervals, for each component of $\boldsymbol{\omega}_k$ and $\boldsymbol{\rho}_k$.

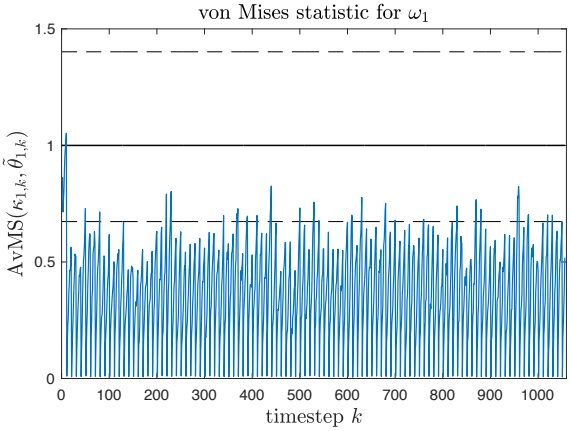
[1]See Appendix B.4 for details about this metric.

First, both filters appear to be unbiased, as indicated by the mean of the estimation errors (black curves), which remains close to zero for most of the time. However, the left subfigures show that the UKF-LG is generally inconsistent with respect to the true estimation accuracy, since its $\mathbf{P}_k < \tilde{\mathbf{P}}_k$ for all state variables, except for ω_1, where $P_{\omega_1,k} \approx \tilde{P}_{\omega_1,k}$. In contrast, the vMG-UKF seems to exhibit consistency in the estimation of $\boldsymbol{\rho}$ ($\mathbf{P}_{\boldsymbol{\rho}\boldsymbol{\rho},k} \approx \tilde{\mathbf{P}}_{\boldsymbol{\rho}\boldsymbol{\rho},k}$) and ω_1 ($P_{\omega_1,k} \approx \tilde{P}_{\omega_1,k}$), while appearing conservative for ω_2 and ω_3, with $P_{\omega_i,k} > \tilde{P}_{\omega_i,k}$ for $i = \{2,3\}$. Despite this conservatism, the vMG-UKF provides a closer approximation of the true estimation uncertainty than the UKF-LG, although it still underestimates its estimation performances. This overestimation of $P_{\omega_i,k}$ is likely caused by the approximation ϑ. Indeed, for low dispersions, such as those of ω_2 and ω_3, given the small inputs u_{ω_2} and u_{ω_3}, the function ϑ underestimates the concentration parameter (as illustrated in Figure 6.5) when compared to the approximation based on the inverse of the variance, which is more accurate in the small-dispersion regime (but tends to singularity). Consequently, sigma-points are generated using concentration parameters that are smaller than required, leading to an excessive spread and explaining the observed conservatism in the estimation of ω_2 and ω_3.

We can debate if comparing the variances for the angular part still makes sense (given the fact that we claimed in previously that the notion of covariance may break down). In the present case, since the 3σ standard deviation in ω_i does not reach a value above 0.7 rad ($\approx 40°$) $\forall i = \{1,2,3\}$, we can reasonably compare these quantities. In case of doubt, we can assess the Average von Mises Statistic (AvMS, cf. Appendix B.4), providing a quantitative measure of consistency for angular variables (cf. Figure 6.10). As shown in



(a) AvMS of the UKF-LG. Black dotted lines are the acceptance intervals with a 95% acceptance rate.



(b) AvMS of the vMG-UKF-LG. Black dotted lines are the acceptance intervals with 95% acceptance rate.

Figure 6.10: AvMS of the filters on the variable ω_1 as a function of time instant k.

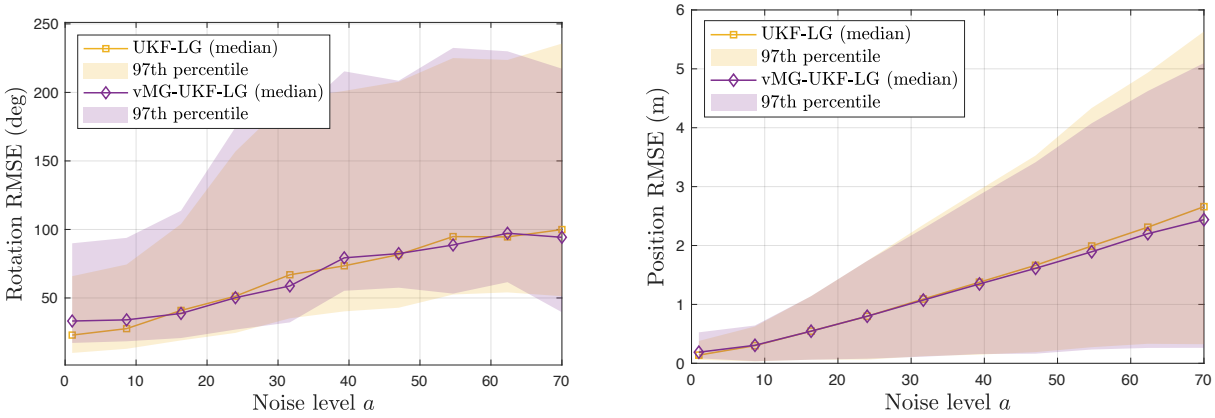
the left subfigure 6.10a, the UKF-LG is largely consistent for ω_1 during the first half of the trajectory ($k < 500$), but tends to become inconsistent afterward. This is reflected in the evolution of the AvMS, which mostly remains within the acceptance bounds (consistent domain) before $k = 500$, but increasingly exceeds the upper bound beyond this point

(inconsistent domain). In contrast, the vMG-UKF remains predominantly below the acceptance interval (indicating that it is actually in the conservative domain) throughout the trajectory, as shown in the right subfigure 6.10b, with only a few time instants where the AvMS enters the acceptance bounds of the consistent domain.

Despite these isolated observations, the overall consistency and conservatism of the vMG-UKF result in a superior RMSE performance compared to the UKF-LG, which is inconsistent across nearly all state variables. This inconsistency in the UKF-LG was expected, as the input noise on $\boldsymbol{\omega}_{\mathbf{w}}$ follows a von Mises distribution rather than a Gaussian, leading to errors when a Gaussian probability model is used. In contrast, a vMG-UKF-LG can mitigate these errors, assuming that this filter is properly parameterized with the real characteristics of the input noise.

RMSE results for cartesian position measurements:

It is also of interest to examine the filter's performance in the scenario where only cartesian position measurements are available, using a model $\mathbf{h}_{\text{gps}}$ with $\mathbf{R} = \text{diag}\left([0.05^2, 0.05^2, 0.05^2]\right)$. All hyper-parameters being equal, the RMSE results are given in Figure 6.11. In this con-



(a) RMSE in rotation as a function of a.

(b) RMSE in position as a function of a.

Figure 6.11: RMSE in rotation (left plot) and in position (right plot) as a function of the noise level a (with observation model $\mathbf{h}_{\text{gps}}$).

figuration, the vMG-UKF does not outperform the UKF-LG. Both filters exhibit large rotation errors (cf. subfigure 6.11a), primarily due to the absence of direct orientation observations. This contrasts with range-and-bearing measurements, where orientation information can be indirectly inferred from the measurement geometry. Regarding the rotation, the vMG-UKF seems slightly equal to the UKF-LG for all noise levels, with only marginal differences that are insufficient to support definitive conclusions. An analysis of the confidence bounds further indicates that the vMG filter is slightly less robust than its Gaussian counterpart. Regarding position (subfigure 6.11b), both filters seems equal for $a < 45$, and the vMG filter marginally outperforms the Gaussian filter at higher noise levels ($a > 45$), although the difference remains marginal. Overall, for cartesian position measurements, we can argue that both filters exhibit comparable performances, and the vMG-UKF does not compensate for the lack of attitude information.

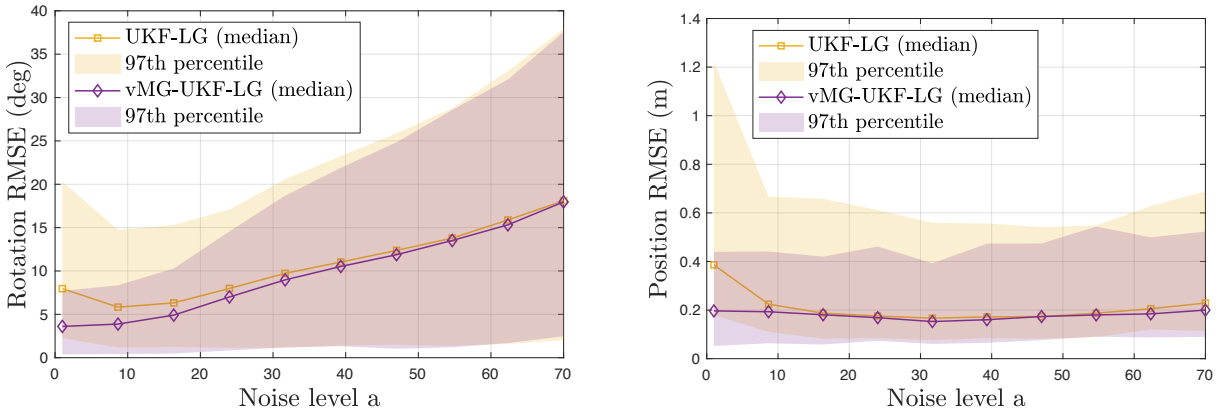
■ Sensitivity to noise on angular inputs only

Hyper-parameters:

Measurement model: $\mathbf{h}_{\text{lidar}}$
Input noise: $\mathbf{Q} = \text{diag}\left(\left[(a\frac{\pi}{180})^2, (a\frac{0.7\pi}{180})^2, (\frac{0.1\pi}{180})^2,\ (0.1)^2, (0.1)^2, (0.05)^2\right]\right)$
(All other things being equal)

RMSE results:

The RMSE results corresponding to the above set of hyper-parameters are presented in Figure 6.12. For the UKF-LG, the rotation RMSE decreases in the range $1 < a < 10$ before increasing when $a > 10$, while the position RMSE decreases and subsequently stabilizes. In contrast, the rotation RMSE of the vMG-UKF increases monotonically with a, whereas its position RMSE remains approximately constant for all values of a. Overall, the vMG filter achieves the best performance in this configuration. Nevertheless, the improvement over the UKF-LG is marginal for large values of a. This behavior can be attributed to the increasing correlation between ω_1 and ω_2 as the level a grows. Since the proposed vMG pdf is not designed to capture correlations between angular variables, the vMG-UKF may exhibit degraded performance when the noise simultaneously increases on multiple angular components.



(a) RMSE in rotation as a function of a.

(b) RMSE in position as a function of a.

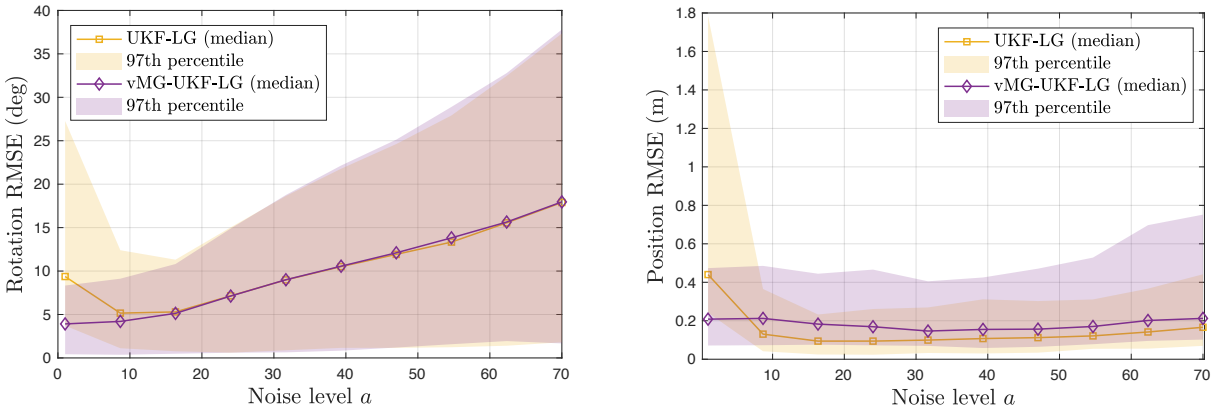
Figure 6.12: RMSE in rotation (left plot) and in position (right plot) as a function of the noise level a.

To better emphasize this aspect, let us perform the same experiment with the following setting:

$$\mathbf{Q} = \text{diag}\left(\left[\left(\frac{a \times \pi}{180}\right)^2, \left(\frac{a \times 0.7\pi}{180}\right)^2, \left(\frac{a \times 0.1\pi}{180}\right)^2, (0.1)^2, (0.1)^2, (0.05)^2\right]\right)$$

The results for this configuration are presented in Figure 6.13. The observed trends are consistent with those reported previously, albeit more pronounced. In particular, for $a > 15$, the rotation RMSE of the vMG filter becomes nearly indistinguishable from that of the Gaussian filter (subfigure 6.13a), while the UKF-LG achieves the best performance in terms of position RMSE (subfigure 6.13b). A closer examination of the UKF-LG position RMSE, in comparison with its behavior under lower noise levels (cf. subfigure 6.12b),

indicates that the UKF-LG benefits from the increased noise on ω_{w_3}. In contrast, the vMG filter does not exhibit such improvement, as the order of magnitude of its position RMSE remains unchanged. The rotation RMSE results shown in subfigure 6.13a, when compared



(a) RMSE in rotation as a function of a.

(b) RMSE in position as a function of a.

Figure 6.13: RMSE in rotation (left plot) and in position (right plot) as a function of the noise level a.

with those of subfigure 6.12a, suggest that the proposed vMG filter fails to account for angular correlations and to benefit from them. Indeed, increasing the uncertainty on ω_1, ω_2, and ω_3 through $\boldsymbol{\omega}_\mathbf{w}$ does not yield any noticeable improvement compared to augmenting the uncertainty on ω_1 and ω_2 only. This deficiency in correlation modeling propagates to the position estimation, as indicated by the unchanged position RMSE of the vMG filter when comparing subfigures 6.13b and 6.12b. In contrast, the Gaussian filter continues to properly capture these correlations, allowing it to exploit the increased noise levels and improve its performance. These observations underscore a structural limitation of the proposed vMG probability model.

■ Sensitivity to measurement noise

We assess the sensitivity of the vMG filter to the measurement noise $\mathbf{v}$, whose level is parameterized by $\mathbf{R}$.

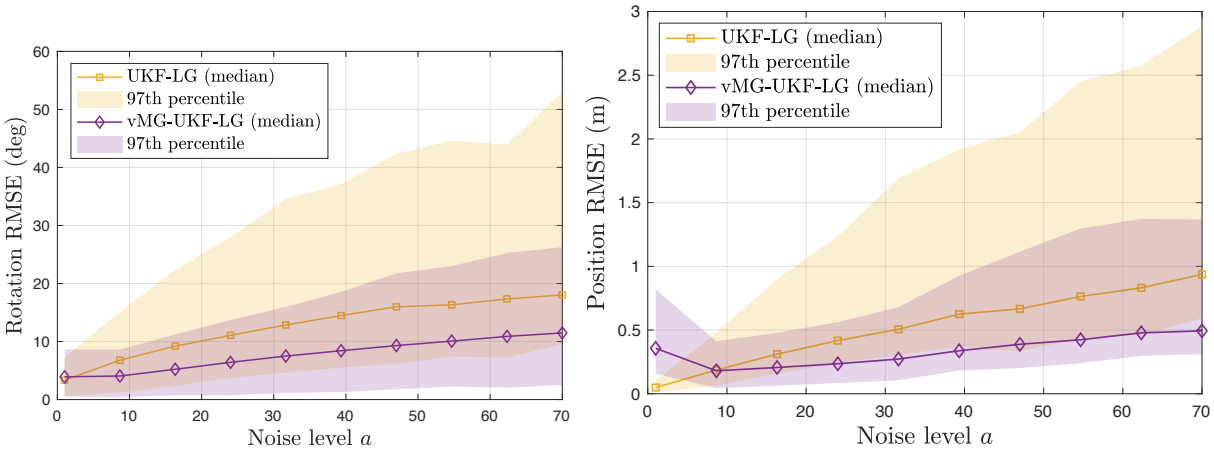
Hyper-parameters:

Measurement model: $\mathbf{h}_{\text{lidar}}$
Input noise: $\mathbf{Q} = \text{diag}\left(\left[(\frac{\pi}{180})^2, (\frac{\pi}{180})^2, (\frac{\pi}{180})^2, (0.05)^2, (0.05)^2, (0.05)^2\right]\right)$
Output noise: $\mathbf{R}^{(i)} = a^2 \times \text{diag}\left([0.01^2, 0.01^2, 0.01^2]\right), \forall i = \{1, 2, 3\}$
(All other things being equal)

Results:

The results for this configuration are shown in Figure 6.14. As illustrated in subfigure 6.14a, the vMG filter exhibits a lower sensitivity to measurement noise than the Gaussian filter in terms of rotational estimation. On average, the vMG-UKF achieves a smaller rotation RMSE than the UKF-LG, and its confidence bounds indicate increased robustness. A comparable behavior is observed for the position RMSE (subfigure 6.14b), except for $a = 1$ where the vMG-UKF position RMSE is slightly larger than the UKF-LG position

RMSE. Nevertheless, these results suggest that the vMG filter is primarily advantageous in regimes characterized by high measurement noise on $\mathbf{v}$ and low process noise on $\mathbf{w}$ in the case of range-and-bearing measurements.



(a) RMSE in rotation as a function of a.

(b) RMSE in position as a function of a.

Figure 6.14: RMSE in rotation (left plot) and in position (right plot) as a function of the noise level a.

Hyper-parameters:

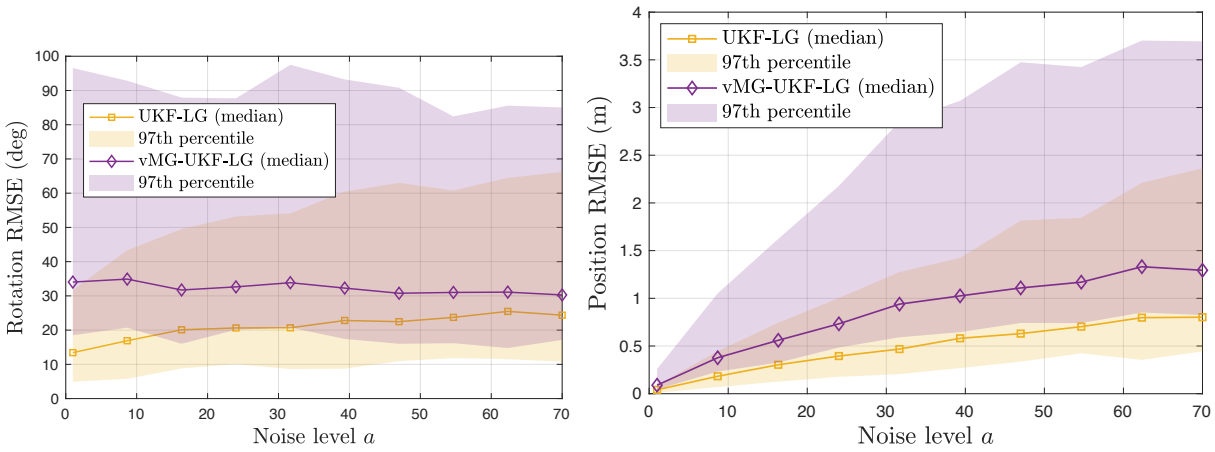
Measurement model: $\mathbf{h}_{\text{gps}}$
Output noise: $\mathbf{R} = a^2 \times \text{diag}\left([0.01^2, 0.01^2, 0.01^2]\right)$
(All other things being equal)

Results:

The results corresponding to the above setting are reported in Figure 6.15. It can be observed that the RMSE, both in rotation and position, is significantly higher for the vMG filter than for the Gaussian filter when cartesian position measurements are used. This behavior can be attributed to the absence of attitude information through measurements (recall that, in contrast, range-and-bearing measurements implicitly provide attitude-related information through the relative positions of landmarks, which helps mitigate rotational error propagation). This limitation is further amplified by the difficulty of the vMG filter in accurately handling rotational/translational correlations, due to the approximations underlying its probability model. As a result, the vMG filter exhibits a higher sensitivity to noise increase on $\mathbf{v}$ in the case of GPS-like measurements.

■ Concluding remarks

Based on the experiments conducted above, several conclusions can be drawn regarding the vMG Kalman filter. First, its performance is limited when only cartesian position measurements are available. In this configuration, Kalman filtering strongly relies on the accurate modeling of cross-correlations between the rotational and translational components of the state. However, the vMG probability model incorporates these cross-correlations only approximately, which hinders its ability to represent them correctly. Although the vMG-UKF can still achieve performance comparable to that of the stan-



(a) RMSE in rotation as a function of a.

(b) RMSE in position as a function of a.

Figure 6.15: RMSE in rotation (left plot) and in position (right plot) as a function of the noise level a.

dard UKF-LG (when the measurement noise on $\mathbf{v}$ is low and the process noise on $\mathbf{w}$ is high in the case of cartesian position measurements, cf. Figure 6.11), its performance deteriorates when the process noise on $\mathbf{w}$ is low and the measurement noise on $\mathbf{v}$ is high (cf. Figure 6.15). Nonetheless, in the case of range-and-bearing measurements, the vMG-UKF consistently provides the best performance across most configurations and exhibits significantly reduced sensitivity (more robustness) to noise amplification, whether affecting $\mathbf{v}$ (cf. Figure 6.14) or a single component of $\mathbf{w}$ (cf. Figure 6.8). These performances are explained by a certain consistency and conservatism in capturing the uncertainty of the state variables (cf. Figure 6.9). However, when the noise level increases on all components of $\boldsymbol{\omega}_{\mathbf{w}}$, the performance of the vMG filter degrades, primarily due to the absence of proper correlation modeling between angular variables (cf. Figures 6.12 and 6.13).

6.2.3 Additional results

■ Efficiency

We evaluate the efficiency of the proposed vMG filter compared to the UKF-LG in the sense of the Cramér-Rao Lower Bounds on Lie groups (CRLB-LG) (see Appendix B.4 for more details). The lower bounds are constructed based on the system model $(\boldsymbol{f}, \mathbf{h})$ and the noise matrices $\mathbf{Q}_k$ and $\mathbf{R}_k$. In particular, for the system of interest $\boldsymbol{f}$, we have the following jacobian:

$$\mathbf{F}_k = -\boldsymbol{\mathcal{J}}_{\mathsf{L}}\left(\log(\boldsymbol{\chi}_{k-1}^{-1}\boldsymbol{\chi}_k)\right) \tag{6.9}$$

For the observation model of type $\mathbf{h}_{\text{lidar}}$, we compute $\mathbf{H}_k^{(i)}$ (for the i^{th} landmark) as:

$$\begin{aligned}\mathbf{H}_k^{(i)} &= \left.\frac{\partial\left(\mathbf{y}_k - \mathbf{h}_{\text{lidar}}^{(i)}(\boldsymbol{\chi}_k \exp(\boldsymbol{\xi}^{\wedge}), \mathbf{0})\right)}{\partial \boldsymbol{\xi}}\right|_{\boldsymbol{\xi}=\mathbf{0}} = \\ &= (-1)\left[\boldsymbol{\mathcal{R}}^{\top}\mathbf{E}_1(\boldsymbol{l}^{(i)} - \boldsymbol{x}),\quad \boldsymbol{\mathcal{R}}^{\top}\mathbf{E}_2(\boldsymbol{l}^{(i)} - \boldsymbol{x}),\quad \boldsymbol{\mathcal{R}}^{\top}\mathbf{E}_3(\boldsymbol{l}^{(i)} - \boldsymbol{x}),\quad -\boldsymbol{\mathcal{R}}^{\top}\right] \in \mathbb{R}^{3\times 6}\end{aligned} \tag{6.10}$$

where E_1, E_2 and E_3 are the generators of $\mathfrak{so}(3)$:

$$\mathbf{E}_1 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{pmatrix}, \quad \mathbf{E}_2 = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{pmatrix}, \quad \mathbf{E}_3 = \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

Then, the full jacobian $\mathbf{H}_k$ is

$$\mathbf{H}_k = \begin{bmatrix} \mathbf{H}_k^{(1)} \\ \vdots \\ \mathbf{H}_k^{(N_l)} \end{bmatrix} \in \mathbb{R}^{3N_l \times 6} \tag{6.11}$$

Hence, the CRLB-LG are obtained from theses jacobians $(\mathbf{F}_k, \mathbf{H}_k)$, and the true noise characteristics $(\mathbf{Q}_k, \mathbf{R}_k)$, following the recursive algorithm of [Chahbazian et al., 2024]. The efficiency of the vMG-UKF is then compared to the efficiency of the UKF-LG, with the CRLB-LG as a comparison measure. In particular, we consider the configuration described below.

Hyper-parameters:

Measurement model: $\mathbf{h}_{\text{lidar}}$
Measurement frequency: $f_{obs} = 5$ Hz
Initial covariance: $\mathbf{P}_{0|0} = \text{diag}\left(\left[(\frac{60\pi}{180})^2, (\frac{60\pi}{180})^2, (\frac{60\pi}{180})^2, 0.5^2, 0.5^2, 0.5^2\right]\right)$
The estimated pose $\hat{\boldsymbol{\chi}}_{0|0}$ is initialized with an error $\tilde{\boldsymbol{\chi}}_0 \sim \mathcal{VMG}_{\mathsf{L}}(\boldsymbol{\kappa}_{0|0}, \mathbf{P}_{\boldsymbol{\rho},0|0})$.
Input noise: $\mathbf{Q} = \text{diag}\left(\left[(\frac{0.1\pi}{180})^2, (\frac{0.1\pi}{180})^2, (\frac{0.1\pi}{180})^2, (0.01)^2, (0.01)^2, 0.01)^2\right]\right)$
Input uncertainty parameter: $\hat{\mathbf{Q}} = \mathbf{Q}$
Output noise: $\mathbf{R}^{(i)} = \text{diag}\left([0.05^2, 0.05^2, 0.05^2]\right)$, $\forall i = \{1, 2, 3\}$
Output uncertainty parameter: $\hat{\mathbf{R}}^{(i)} = \mathbf{R}^{(i)}$

$N_s = 100$ Monte-Carlo runs are performed, and the mean (over the N_s simulations) squared error (MSE) along the trajectory is computed as

$$\mathbb{E}_{MC}[\tilde{\omega}_j \tilde{\omega}_j^\top] = \mathbb{E}_{MC}\left[\|\text{log-Euclidean error on } \omega_j\|^2\right] = \frac{1}{N_s}\sum_{i=1}^{N_s} \left\| \log\left((\boldsymbol{\mathcal{R}}_k^{(i)})^{-1} \hat{\boldsymbol{\mathcal{R}}}_k^{(i)} \right)_{\omega_j}^{\vee} \right\|^2$$

$$\mathbb{E}_{MC}[\tilde{\rho}_j \tilde{\rho}_j^\top] = \mathbb{E}_{MC}\left[\|\text{log-Euclidean error on } \rho_j\|^2\right] = \frac{1}{N_s}\sum_{i=1}^{N_s} \left\| \log\left((\boldsymbol{\chi}_k^{(i)})^{-1} \hat{\boldsymbol{\chi}}_k^{(i)} \right)_{\rho_j}^{\vee} \right\|^2$$

where $\log(\cdot)_{\omega_j}^{\vee}$ and $\log(\cdot)_{\rho_j}^{\vee}$ denotes the restriction of the logarithm to ω_j and ρ_j, respectively.

Results:

The results corresponding to the above configuration are shown in Figure 6.16. The pink curves represent the Cramér–Rao lower bounds for each state variable ω_j and ρ_j. The blue curves show the MSE obtained with the UKF-LG, while the green curves correspond to the MSE achieved by the vMG-UKF. For all rotational components ω_j (left subfigures), the vMG-UKF consistently yields MSE values closer to the lower bounds than the UKF-LG, although the bounds are not reached exactly. For the positional components ρ_j (right

subfigures), a similar trend is observed: the vMG-UKF generally remains closer to the lower bounds, and even attains the bounds at certain time instants. Recall that the Cramér–Rao lower bound characterizes the minimum achievable mean squared error for any unbiased estimator of a system $(\boldsymbol{f}, \mathbf{h})$ with noise statistics $(\mathbf{Q}, \mathbf{R})$. Overall, for the considered hyperparameters, these results indicate that the vMG-UKF is more efficient than the UKF-LG, in the sense that it more closely approaches the behavior of a minimum-variance estimator.

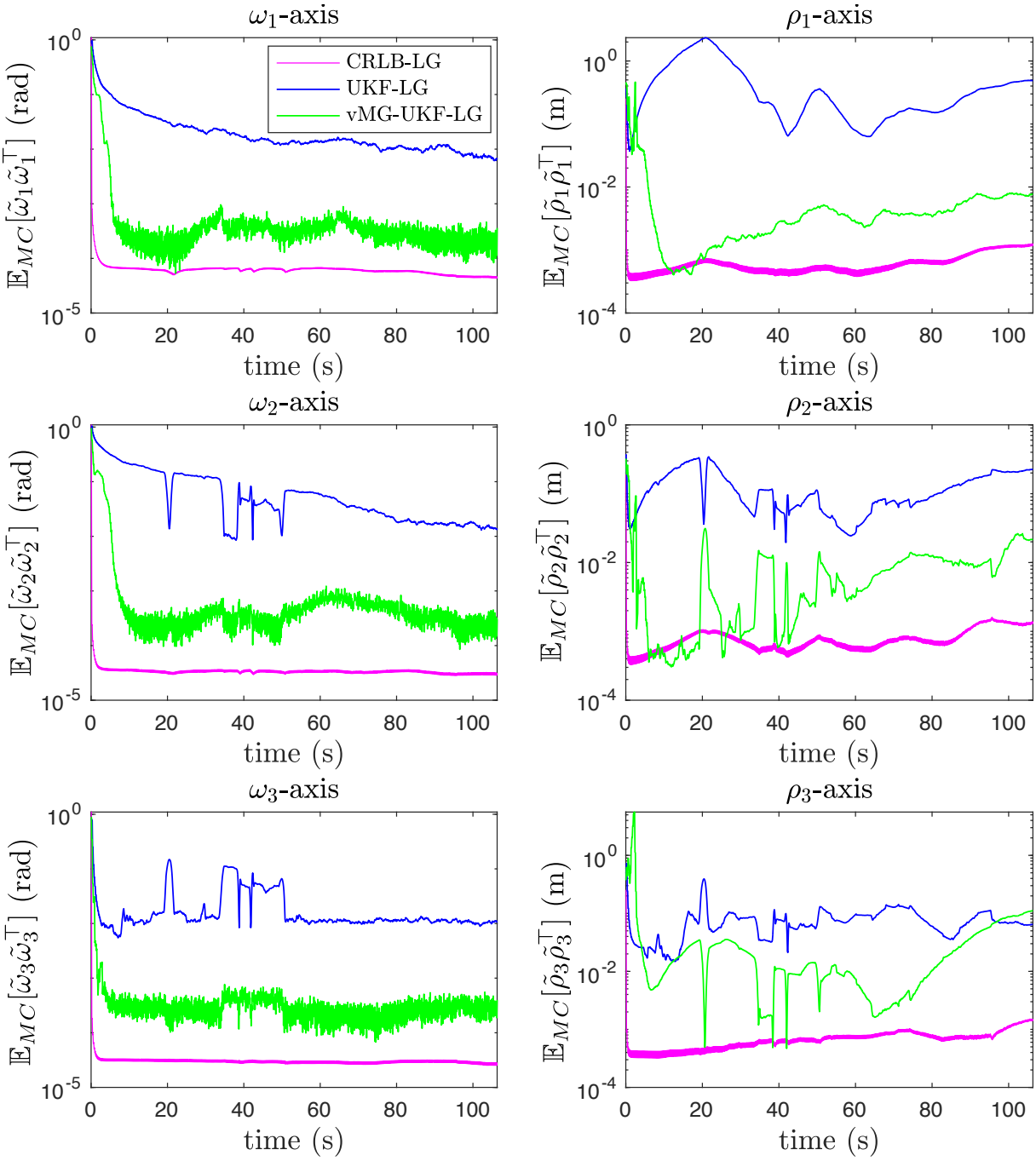


Figure 6.16: Cramér-Rao lower bounds and efficiency performance of the filters.

■ Concluding remarks

Regarding efficiency, the vMG-UKF appears more effective, as its mean squared error remains closer to the Cramér–Rao lower bounds than that of the UKF-LG. However, it should be noted that the lower bounds along the ω_i axes are likely underestimated in Figure 6.16. Indeed, these bounds were computed using the noise covariance $\mathbf{Q}$, whereas the actual noise injected into the simulated trajectory is generated from the concentration parameters $\boldsymbol{\kappa}_{\mathbf{u}}$, rather than directly from $\mathbf{Q}$. Owing to the approximation $\kappa_{u_i} \approx \vartheta(Q_{\omega_i})$, the effective noise variance is in fact larger than the specified argument Q_{ω_i} (cf. Figure 6.5). As a consequence, the true Cramér–Rao lower bounds for the rotational components should be higher than those depicted in Figure 6.16. This observation does not ruin the conclusions regarding the efficiency of the vMG-UKF; on the contrary, it may even suggest that it is actually closer to the bounds than what we see.

■ Performance / Computation cost trade-off

Based on the previous results, it is informative to evaluate the performance gains of the vMG-UKF relative to the additional computation time it requires. Table 6.1 reports, for each filter, the filtering runtime averaged over the N_s simulations, together with the MSE of each state variable ω_j and ρ_j, averaged over the full trajectory. Overall, the vMG-UKF incurs a 14.16% increase in computation time compared to the UKF-LG. This additional cost leads to substantial accuracy improvements, with MSE reductions exceeding 80% for most state variables, except for ρ_3, where the improvement is limited to 15.28%

	UKF-LG	vMG-UKF-LG
Average time (s)	1.51	1.72 (+14.16 %)
Average MSE ω_1 (rad)	0.032	0.006 (-81.28 %)
Average MSE ω_2 (rad)	0.089	0.0084 (-81.28 %)
Average MSE ω_3 (rad)	0.025	0.0028 (-88.68 %)
Average MSE ρ_1 (m)	0.4769	0.0092 (-98.05 %)
Average MSE ρ_2 (m)	0.125	0.0097 (-92.2 %)
Average MSE ρ_3 (m)	0.077	0.065 (-15.28 %)

Table 6.1: Performance gain VS computation cost of the vMG-UKF against the UKF-LG.

6.3 Validations of the OQ-UKF-LG on simulated data

In this section, we provide a set of results to validate the OQ-UKF-LG developed in Chapter 5. Contrary to the validation of the vMG-UKF-LG, we mainly assess the present method on simulated data satisfying the conditions of Gaussian noise. Hence, we have $\mathbf{w}_k \sim \mathcal{P}(\mathbf{w}_k) = \mathcal{N}(\mathbf{0}, \mathbf{Q})$ and $\mathbf{v}_k \sim \mathcal{P}(\mathbf{v}_k) = \mathcal{N}(\mathbf{0}, \mathbf{R})$. The nominal trajectory for these experiments is generated with the same nominal inputs as presented in Figure 6.2. We would like to evaluate how the optimal quantization approach can mitigate the errors when the UKF-LG's noise characteristics ($\hat{\mathbf{Q}}$ and $\hat{\mathbf{R}}$) are initially misparameterized.

6.3.1 Sensitvity to overestimation of Q and R

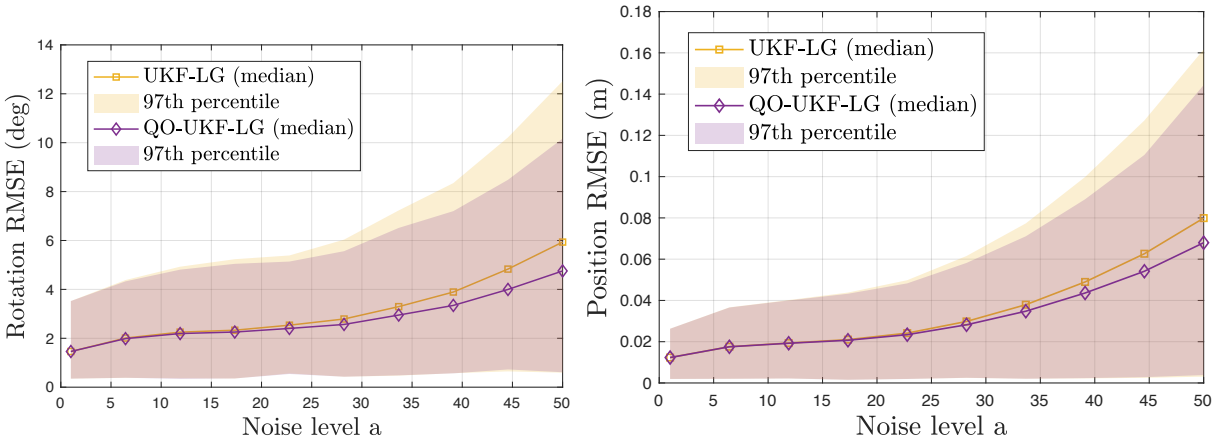
We perform N_s Monte-Carlo simulations and assess the sensitivity of the method to overestimations of $\mathbf{Q}$ and $\mathbf{R}$.

Hyper-parameters:

Monte-Carlos: $N_s = 100$
Measurement model: $\mathbf{h}_{\text{gps}}$
Initial covariance: $\mathbf{P}_{0|0} = \text{diag}\left((\frac{\pi}{180})^2, (\frac{\pi}{180})^2, (\frac{\pi}{180})^2, (0.01)^2, (0.01)^2, (0.01)^2\right)$
Input noise: $\mathbf{Q} = \text{diag}\left((\frac{\pi}{180})^2, (\frac{\pi}{180})^2, (\frac{\pi}{180})^2, (0.01)^2, (0.01)^2, (0.01)^2\right)$
Measurement frequency: $f_{obs} = 1$ Hz
Input uncertainty parameter: $\hat{\mathbf{Q}} = a^2 \times \mathbf{Q}$
Output noise: $\mathbf{R} = \text{diag}\left(0.01^2, 0.01^2, 0.01^2\right)$
Output uncertainty parameter: $\hat{\mathbf{R}} = b^2 \times \mathbf{R}$
CLVQ parameters (applied to prediction and correction): $n_{max} = 100$, $\gamma_{\ell,n} = \frac{\frac{1}{10}\sqrt{\mathbf{P}_{\ell\ell}}}{n}, \quad \forall \ell = \{1, 2, 3\}$

Results:

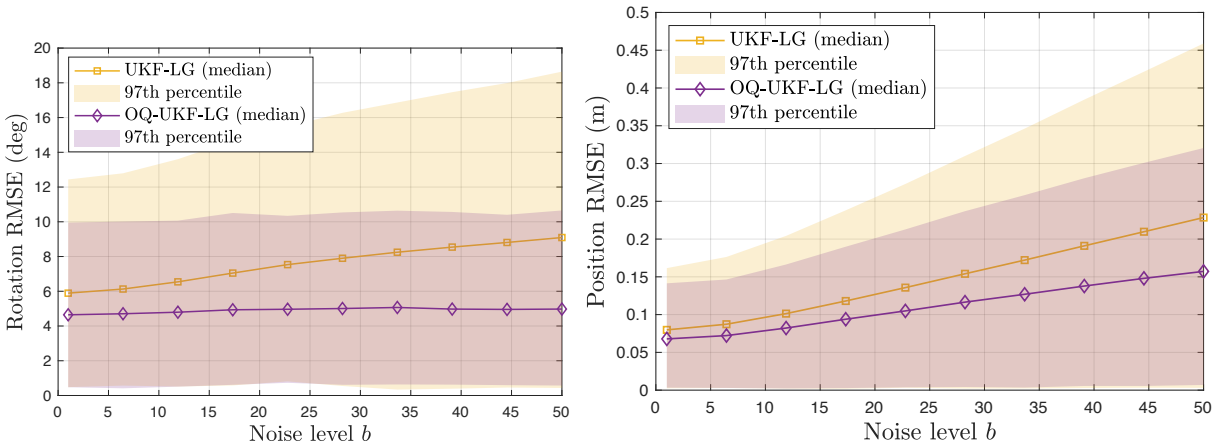
The results for this setting are given in Figure 6.17 for the sensitivity to overestimations of $\mathbf{Q}$, and in Figure 6.18 for the sensitivity to overestimations of $\mathbf{Q}$ and $\mathbf{R}$. With respect to both attitude and position RMSE, the OQ-UKF-LG exhibits the best overall performance in terms of accuracy, as reflected by the order of magnitude of the RMSE. In particular, when the uncertainty parameter $\hat{\mathbf{Q}}$ is overestimated relative to the true noise parameter $\mathbf{Q}$ (Figure 6.17), the incorporation of optimal quantization proves beneficial for large overestimations. Indeed, for $a < 20$, the rotation RMSE (subfigure 6.17a) or the position RMSE (subfigure 6.17b) of both filters are quite identical. Beyond this threshold, however, the performance gap between the OQ-UKF-LG and the standard UKF-LG becomes increasingly pronounced. Furthermore, by fixing the input noise level to $a = 50$ and progressively overestimating the uncertainty parameter $\hat{\mathbf{R}}$ relative to the true measurement noise parameter $\mathbf{R}$, this performance difference becomes even more significant. In particular, the rotation RMSE of the OQ-UKF-LG stabilizes, whereas that



(a) RMSE in rotation, as a function of a ($b = 1$).

(b) RMSE in position, as a function of a ($b = 1$).

Figure 6.17: RMSE in rotation (left) and in position (right) as a function of the noise level a (b fixed).



(a) RMSE in rotation as a function of b ($a = 50$).

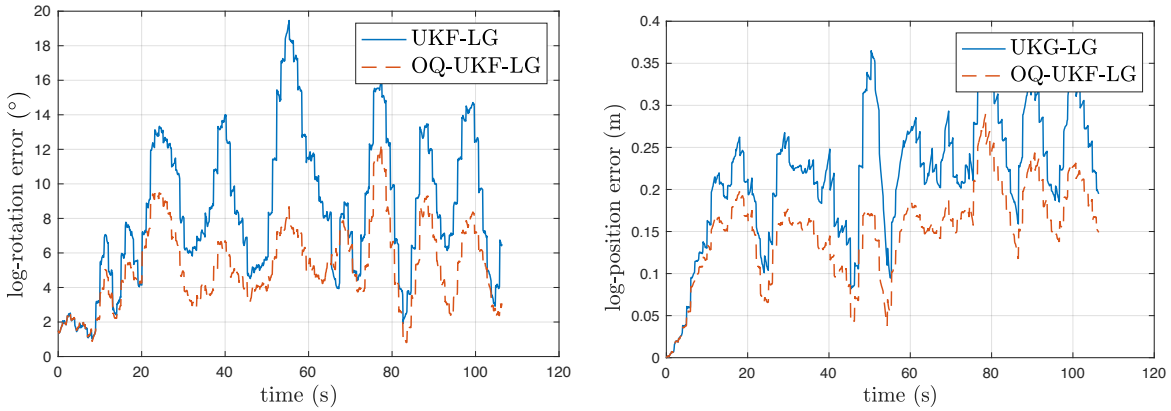
(b) RMSE in position as a function of b ($a = 50$).

Figure 6.18: RMSE in rotation (left) and in position (right) as a function of the noise level b (a fixed).

of the UKF-LG continues to increase (subfigure 6.18a). Regarding position (subfigure 6.18b), the OQ-UKF-LG again achieves the lowest RMSE, although with less stability than observed for the rotation. Finally, in terms of robustness, both filters seem to exhibit comparable behavior, as indicated by the magnitude of their respective confidence bounds.

A focus on the log-rotation and log-position errors of a single run, as a function of time, for the last case $a = 50$, $b = 50$ (Figure 6.19), allows us to show that the OQ-UKF-LG generally follows the same pattern as the UKF-LG but with a reduced error. For this last case, the time evolution of the variances returned by each filter is shown in Figure 6.20. The Monte Carlo average of the estimation errors (black curve) remains close to zero over time, indicating that both filters are unbiased for all state variables, which is a necessary property. Moreover, the empirical variance of the estimation errors (green curve) is tightly concentrated for both filters[2]. It can also be observed that the OQ-UKF-LG consistently produces smaller estimated variances than the UKF-LG, although both

[2]Only the variance associated with the OQ-UKF-LG is displayed, as the UKF-LG yields nearly identical results; the latter is omitted for clarity.



(a) L^2-norm of the log-rotation error.

(b) L^2-norm of the log-position error.

Figure 6.19: L^2-norm of the log-Euclidean errors (in rotation and in position) as a function of time, corresponding to a single simulation ($a = 50$, $b = 50$).

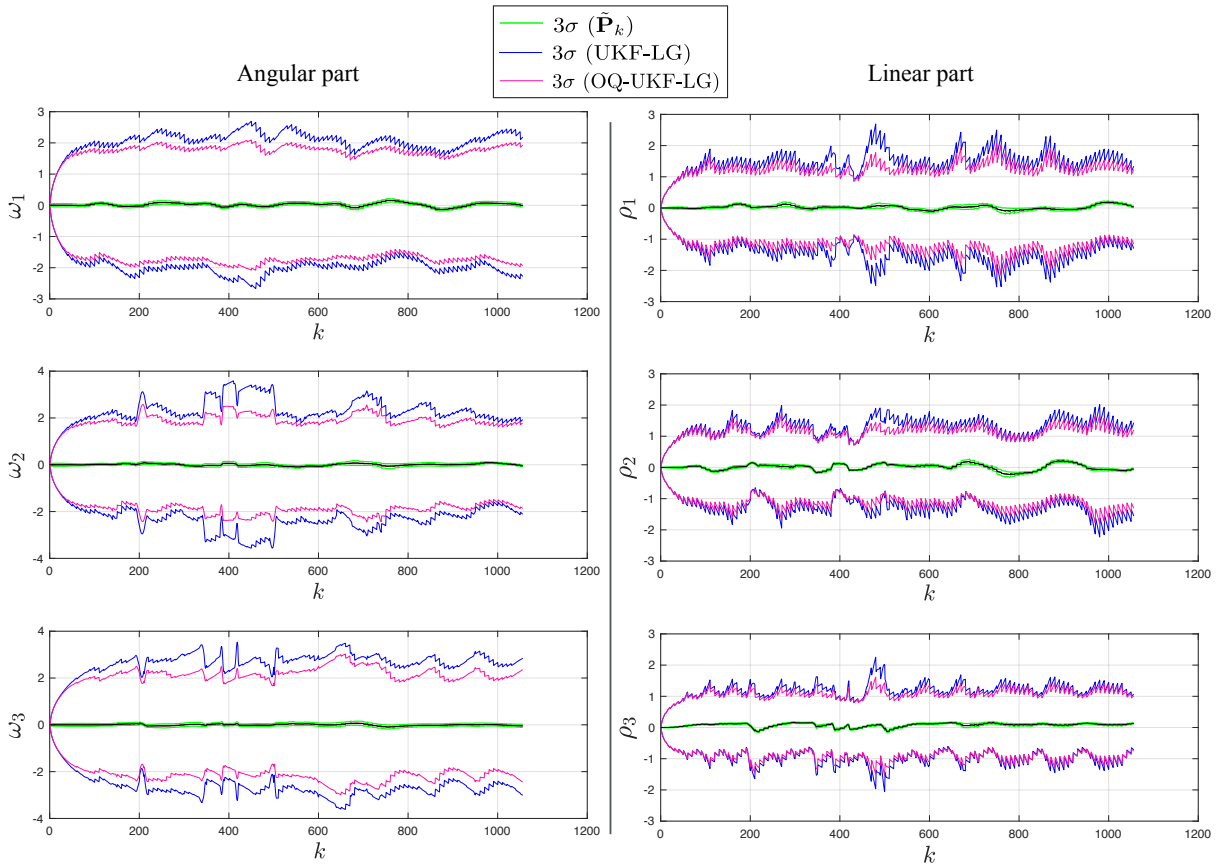


Figure 6.20: Evolution of the variance for ($a = 50$, $b = 50$). The green curves are the 3σ standard deviations of the estimation errors of the Monte-Carlo runs. The blue and pink curves are the variances returned by the UKF-LG and the OQ-UKF-LG, respectively, averaged over the N_s simulations, in terms of 3σ standard deviation. The black curve is the mean of the estimation errors.

filters remain conservative. This variance reduction is a desirable feature in the presence of noise, and highlights the ability of optimal quantization to mitigate the overestimation of uncertainty induced by inflated noise parameters $\hat{\mathbf{Q}}$ and $\hat{\mathbf{R}}$.

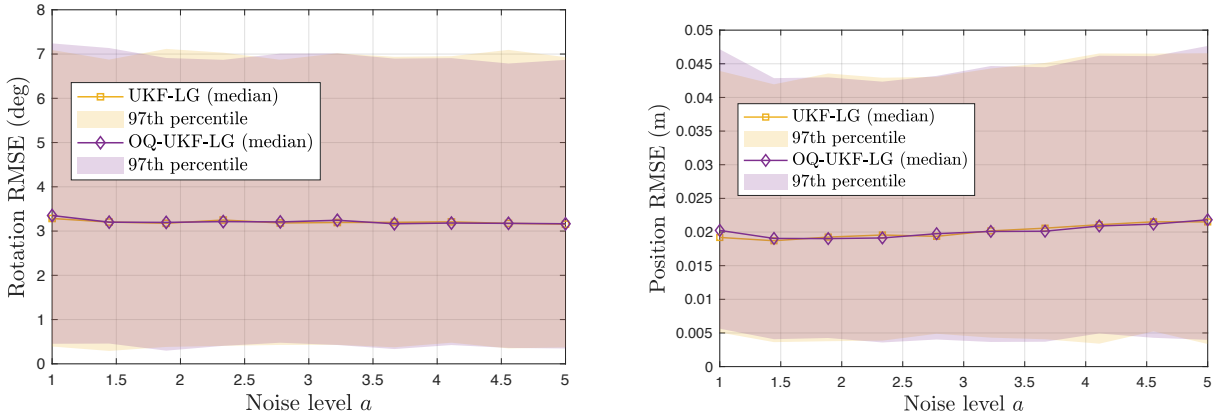
Results for range-and-bearing measurements

Hyper-parameters:

Monte-Carlos: $N_s = 100$
Measurement model: $\mathbf{h}_{\text{lidar}}$
Initial covariance: $\mathbf{P}_{0|0} = \text{diag}\left((\frac{\pi}{180})^2, (\frac{\pi}{180})^2, (\frac{\pi}{180})^2, (0.01)^2, (0.01)^2, (0.01)^2\right)$
Input noise: $\mathbf{Q} = \text{diag}\left((\frac{\pi}{180})^2, (\frac{\pi}{180})^2, (\frac{\pi}{180})^2, (0.01)^2, (0.01)^2, (0.01)^2\right)$
Input uncertainty parameter: $\hat{\mathbf{Q}} = a^2 \times \mathbf{Q}$
Output noise: $\mathbf{R} = \text{diag}\left((0.01)^2, (0.01)^2, (0.01)^2\right)$
Output uncertainty parameter: $\hat{\mathbf{R}} = \mathbf{R}$
CLVQ parameters (applied to prediction only): $n_{max} = 100$, $\gamma_{\ell,n} = \frac{\frac{1}{10}\sqrt{\mathbf{P}_{\ell\ell}}}{n}, \quad \forall \ell = \{1, 2, 3\}$

Results:

The OQ-UKF-LG did not produce significant improvements in the case of range-and-bearing measurements. Figure 6.21 presents the rotation and position RMSE for both filters. It can be observed that, for all noise, the two filters perform equivalently, although we see a slight improvement from the OQ-UKF-LG for high noise. Unlike the case of cartesian position measurements, the UKF-LG does not appear to require enhancement via optimal quantization of the angular components. Indeed, a filter relying solely on cartesian position measurements must compensate for the growth of angular covariance due to the absence of direct attitude observations, whereas a filter that observes relative landmark positions can inherently mitigate the growth of angular uncertainty. Furthermore, to obtain the results shown in Figure 6.21, the CLVQ algorithm was applied only to the UT of the prediction step, as applying it to the UT of the correction step led to the divergence of the OQ-UKF-LG. Hence, in this case, the optimal quantization approach was only necessary when no measurement was available (i.e, during the predictions steps, where in general, a Kalman filter typically exhibits an increase in covariance due to non correction).



(a) RMSE in rotation as a function of a ($b = 1$).

(b) RMSE in position as a function of a ($b = 1$).

Figure 6.21: RMSE in rotation (left) and in position (right) as a function of the noise level a (b fixed).

6.3.2 Sensitivity to underestimation of Q and R

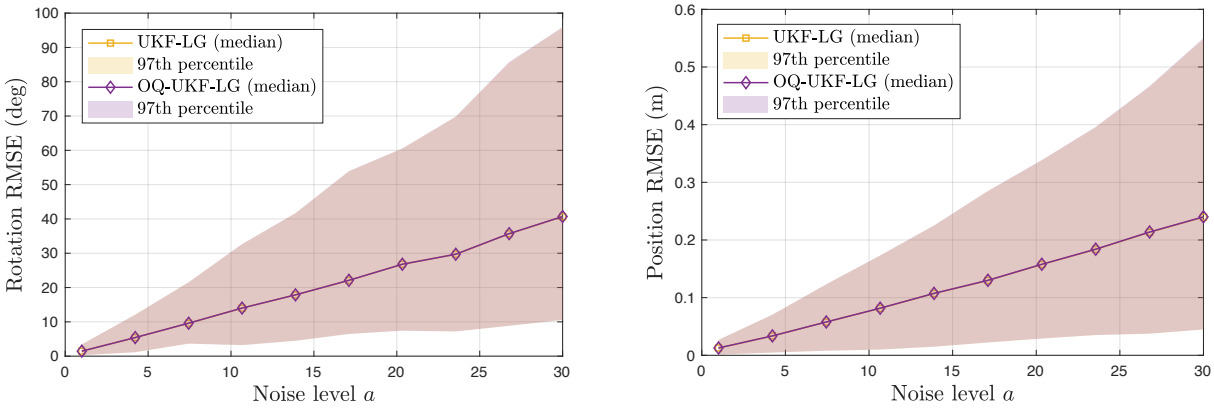
We perform N_s Monte-Carlo simulations and assess the sensitivity of the method to underestimations of $\mathbf{Q}$ and $\mathbf{R}$. This time, we set the uncertainty parameters $\hat{\mathbf{Q}}$ and $\hat{\mathbf{R}}$ to a small value while increasing $\mathbf{Q}$ and $\mathbf{R}$.

Hyper-parameters:

Monte-Carlos: $N_s = 100$
Measurement model: $\mathbf{h}_{\text{gps}}$
Initial covariance: $\mathbf{P}_{0|0} = \text{diag}\left((\frac{\pi}{180})^2, (\frac{\pi}{180})^2, (\frac{\pi}{180})^2, (0.01)^2, (0.01)^2, (0.01)^2\right)$
Input uncertainty parameter: $\hat{\mathbf{Q}} = \text{diag}\left((\frac{\pi}{180})^2, (\frac{\pi}{180})^2, (\frac{\pi}{180})^2, (0.01)^2, (0.01)^2, (0.01)^2\right)$
Input noise: $\mathbf{Q} = a^2 \times \hat{\mathbf{Q}}$
Output uncertainty parameter: $\hat{\mathbf{R}} = \text{diag}\left((0.01)^2, (0.01)^2, (0.01)^2\right)$
Output noise: $\mathbf{R} = b^2 \times \hat{\mathbf{R}}$
CLVQ parameters (for prediction and correction): $n_{max} = 100$, $\gamma_{\ell,n} = \frac{\frac{1}{10}\sqrt{\mathbf{P}_{\ell\ell}}}{n}$, $\forall \ell = \{1, 2, 3\}$

Results:

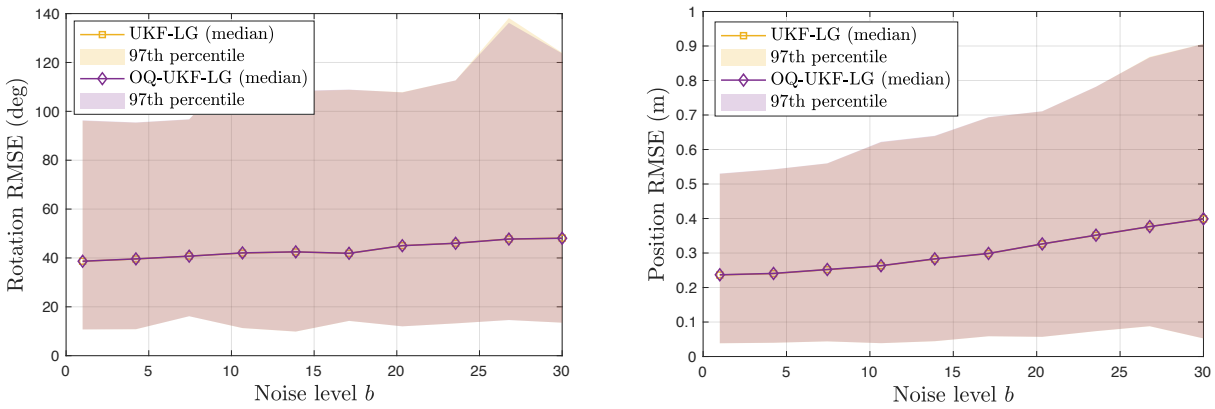
The results for the above setting are provided in Figure 6.22 and 6.23. We see that there is absolutely no difference between the OQ-UKF-LG and the UKF-LG in the case of underestimations of the noise. The same study conducted with the range-and-bearing measurements resulted in the same conclusions. We can argue that this is due to the non capacity of the CLVQ to largely spread the sigma-points so that the underestimation of the true noise can be compensated. In Appendix B.5, we provide further results that demonstrate the capacities of the OQ-UKF-LG in the case of a two-dimensional navigation on $\boldsymbol{SE}(2)$



(a) RMSE in rotation as a function of a ($b = 1$).

(b) RMSE in position as a function of a ($b = 1$).

Figure 6.22: RMSE in rotation (left) and in position (right) as a function of the noise level a (b fixed).



(a) RMSE in rotation as a function of b ($a = 30$).

(b) RMSE in position as a function of b ($a = 30$).

Figure 6.23: RMSE in rotation (left) and in position (right) as a function of the noise level b (a fixed).

6.3.3 Performance / Computation cost trade-off

We analyzed the gain in performance depending on the CLVQ parameter n_{max}, with the following configuration.

Hyper-parameters:

Monte-Carlos: $N_s = 10$
Measurement model: $\mathbf{h}_{\text{gps}}$
Initial covariance: $\mathbf{P}_{0|0} = \text{diag}\left((\frac{\pi}{180})^2, (\frac{\pi}{180})^2, (\frac{\pi}{180})^2, (0.01)^2, (0.01)^2, (0.01)^2\right)$
Input noise: $\mathbf{Q} = \text{diag}\left((\frac{\pi}{180})^2, (\frac{\pi}{180})^2, (\frac{\pi}{180})^2, (0.01)^2, (0.01)^2, (0.01)^2\right)$
Input uncertainty parameter: $\hat{\mathbf{Q}} = 50^2 \times \mathbf{Q}$
Output noise: $\mathbf{R} = \text{diag}\left((0.01)^2, (0.01)^2, (0.01)^2\right)$
Output uncertainty parameter: $\hat{\mathbf{R}} = \mathbf{R}$
CLVQ parameters (for prediction and correction): $\gamma_{\ell,n} = \frac{\frac{1}{10}\sqrt{\mathbf{P}_{\ell\ell}}}{n}, \quad \forall \ell = \{1, 2, 3\}$

Results:

Figure 6.24 reports the rotation and position RMSE as functions of the average Monte Carlo computation time required by the estimator for a single filtering run. The first point in each plot corresponds to the baseline UKF-LG without optimal quantization, which achieves an average computation time of 0.61 sec, an average rotation RMSE of approximately 6°, and an average position RMSE of approximately 0.08 m. The remaining points correspond to the OQ-UKF-LG for different values of the CLVQ parameter $n_{max} \in \{100, 300, 600, 800, 1000\}$. For the last configuration ($n_{max} = 1000$), the OQ-UKF-LG attains an average rotation RMSE of about 4.40°, and an average position RMSE of about 0.065 m, at the cost of an increased computation time of 2.90 sec. This corresponds to a 4.71× increase in computational cost, yielding relative performance gains of approximately 26.34% in rotation and 18.67% in position. An analysis of the 3σ confidence intervals further indicates that the OQ-UKF-LG exhibits more stable behavior in attitude estimation than in position estimation, as evidenced by the presence of pronounced spikes in the position RMSE (right subfigure). Nevertheless, for all considered parameterizations, the OQ-UKF-LG displays larger standard deviations than the baseline

UKF-LG. This behavior is consistent with the observations reported in subsection §5.3.4 of Chapter 5, and can be attributed to the stochastic nature of the CLVQ algorithm. Indeed, as a stochastic gradient descent method, CLVQ inherently introduces fluctuations in the sigma-point placement, which propagates to the overall performance of the filter.

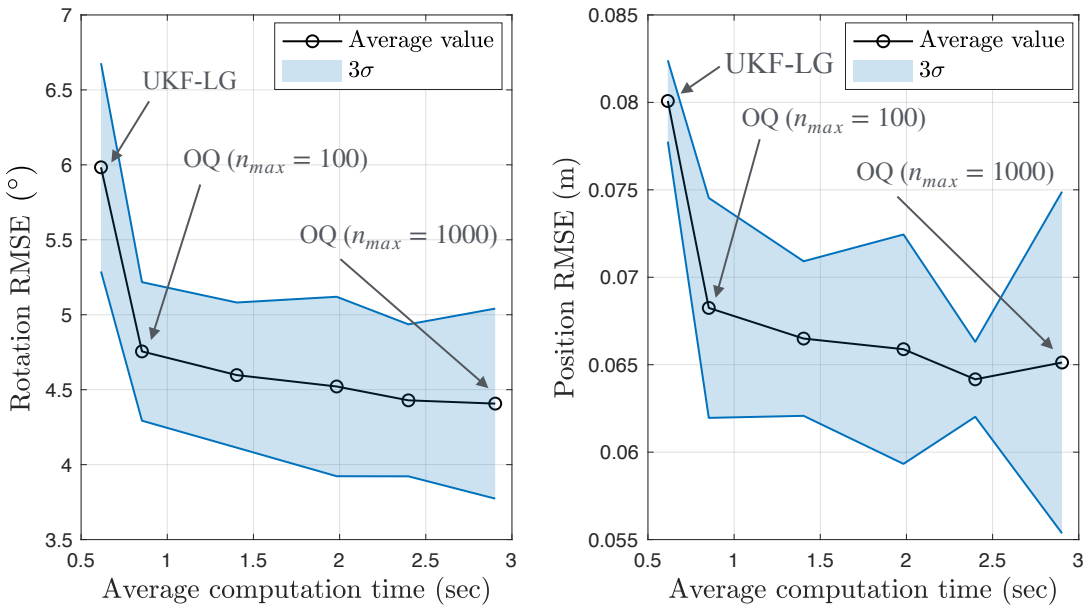


Figure 6.24: Performance gain as a function of computation time. The first point corresponds to the performance given by the UKF-LG. The remaining points correspond to the performance given by the OQ-UKF-LG with different parameters n_{max}.

In general, we can see that the performance gain of the OQ-UKF-LG ($n_{max} = 100$) against the UKF-LG was already significant. Indeed, for an increase of 38.61% in computational time, the performance gain in rotation and position was approximately 20.52% and 14.77%, respectively. In comparison, the performance gain of the OQ-UKF-LG ($n_{max} = 1000$) against the OQ-UKF-LG ($n_{max} = 100$) is less significant with respect to the time spent: for an increase of 240% in computational time, the performance gain in rotation and position was approximately 7.32% and 4.57%, respectively.

■ Concluding remarks

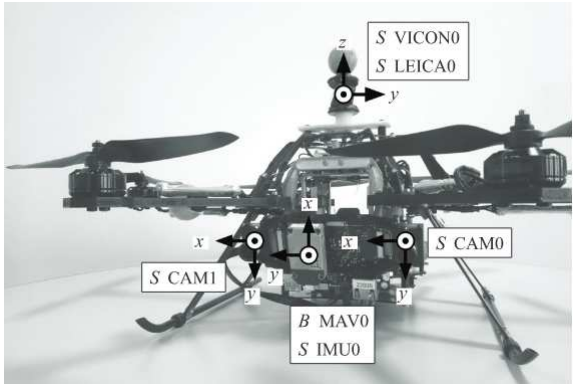
Overall, the optimal quantization–based UKF-LG exhibits improved performance compared to the standard UKF-LG in scenarios where uncertainty increases during filtering while the true estimation errors remain significantly smaller. In this sense, optimal quantization is particularly beneficial when the UKF-LG operates in a conservative regime (§6.3.1). The performance gain becomes more pronounced when some state variables are unobserved (as the case of positions measurements $\mathbf{h}_{\text{gps}}$), leading to increased uncertainty in those components. By contrast, when all state variables are observed, either directly or indirectly (as the case of range-and-bearing measurements $\mathbf{h}_{\text{lidar}}$), the benefit of optimal quantization remains marginal (cf. Figure 6.21). Finally, no improvement is observed in regimes where the filter is already consistent or operates in an inconsistent (underestimated uncertainty) regime (§6.3.2).

6.4 Experimental setup for real data

We now turn to an experimental validation based on real-world data obtained from an open-source dataset, the EuRoc MAV datasets [Burri et al., 2016]. Such datasets provide a particularly relevant testbed for the proposed methods, as the true nature of the noise affecting the system is unknown and cannot be explicitly controlled. In contrast to simulated data, where noise can be arbitrarily chosen, real sensor data are subject to complex, heterogeneous, and potentially non-Gaussian disturbances arising from hardware imperfections, environmental effects, and unmodeled dynamics.

6.4.1 Presentation of the datasets

The EuRoc datasets were collected from multiple flight scenarios involving a multirotor UAV equipped with a visual–inertial sensing system (Figure 6.25). A batch of the



(a) The AscTec "Firefly" hex-rotor helicopter used for dataset collection. A visual sensor provided stereo images and an IMU provided inertial measurements. Position ground truth was recorded with a Leica total station, and 6D pose ground truth with a Vicon system. The body frame is defined at the IMU sensor frame.

Table 1. Sensors and ground-truth instruments.

Sensor	Type	Rate	Characteristics
Cameras	MT9V034	2 × 20 Hz	WVGA, global shutter
IMU	ADIS16448	200 Hz	MEMS, intr. calibrated
Position	Leica MS50	20 Hz	Accuracy: ≈ 1mm
Pose	Vicon	100 Hz	6 D
Structure	Leica MS50	–	Accuracy: ≈ 1mm

(b) Sensors characteristics.

Figure 6.25: Presentation of the robotic system. Images taken from [Burri et al., 2016].

datasets was recorded in a Vicon-equipped room (Figure 6.26a), where 3D point clouds were acquired using the Leica Nova MS50 laser scanner from multiple viewpoints, providing millimeter-level ground truth of the environment structure. Three scenarios of increasing complexity were considered (Figure 6.26b).

IMU noise characteristics

The dataset files provide the noise model parameters for the gyroscope and accelerometer. The associated noise process is modeled as a combination of white noise and a standard random walk:

$$\begin{aligned} \mathbf{n}_{\text{gyro}}(t) &= \mathbf{b}_{\text{gyro}}(t) + \mathbf{w}_{\text{gyro}}(t) \\ \mathbf{n}_{\text{acc}}(t) &= \mathbf{b}_{\text{acc}}(t) + \mathbf{w}_{\text{acc}}(t) \end{aligned} \tag{6.12}$$



(a) Ground truth point cloud of the Vicon room. The size of the room is $8 \times 8.4 \times 4$ m^3.

Table 2. Dataset characteristics.

Name	Length / duration	Avg. vel. / angular vel.	Note
V1_01_easy	58.6m 144s	0.41ms^{-1} 0.28rad s^{-1}	Slow motion, bright scene
V1_02_medium	75.9m 83.5s	0.91ms^{-1} 0.56rad s^{-1}	Fast motion, bright scene
V1_03_difficult	79.0m 105s	0.75ms^{-1} 0.62rad s^{-1}	Fast motion, motion blur

(b) Scenarios of interest.

Figure 6.26: Presentation of the environment and scenarios. Images taken from [Burri et al., 2016].

where $\mathbf{b}_{[.]}(t)$ ($[\cdot]$ being either "gyro" or "acc") denotes the slowly varying bias process such that

$$\frac{d}{dt}\mathbf{b}_{[.]}(t) = \mathbf{w}_{[\cdot]_\text{bias}}(t) \sim \mathcal{P}(\mathbf{0}, \sigma^2_{[\cdot]_\text{bias}}\,\mathbf{I}_{3\times3}), \tag{6.13}$$

and $\mathbf{w}_{[.]}(t)$ denotes continuous-time, white noise such that

$$\mathbf{w}_{[.]}(t) \sim \mathcal{P}(\mathbf{0}, \sigma^2_{[.]}\,\mathbf{I}_{3\times3}). \tag{6.14}$$

The parameters for these noise are given in the yaml files of the dataset. In particular, we are provided with:

$$\begin{aligned}
\sigma_{\text{gyro}} &= 1.6968 \times 10^{-4} \times \sqrt{200} = 0.0024 \ \text{ rad.s}^{-1} = 0.1375 \ ^\circ/\text{s} \\
\sigma_{\text{gyro_bias}} &= 1.9393 \times 10^{-5} \times \sqrt{200} = 2.7426 \times 10^{-4} \ \text{ rad.s}^{-2} = 0.0157 \ ^\circ/\text{s}^2 \\
\sigma_{\text{acc}} &= 2 \times 10^{-3} \times \sqrt{200} = 0.0283 \ \text{ m.s}^{-2} \\
\sigma_{\text{acc_bias}} &= 3 \times 10^{-3} \times \sqrt{200} = 0.0424 \ \text{ m.s}^{-3}
\end{aligned} \tag{6.15}$$

which were obtained from a noise analysis with the IMU at rest. As noted in [Burri et al., 2016], these parameters may be underestimated, as they do not account for deterministic errors arising during flight from additional sources of uncertainty. Consequently, the true noise levels may be significantly higher than those prescribed by (6.15). This provides an ideal ground for evaluating the performance of our method.

6.4.2 Presentation of the experimental setup

Since the UKF-LG has already been extensively tested and validated on these datasets in [Brossard et al., 2018], notably for the three scenarios (easy, medium, and difficult) depicted in Figure 6.26b, we reused the corresponding implementation (available at this link) and integrated the proposed filter into the existing framework. The considered state

space is $\boldsymbol{SE}_{2+N_l}(3) \times \mathbb{R}^6$ and the discrete dynamical model is given by

$$\begin{aligned}
\text{body state dynamics } \boldsymbol{\chi}_k : & \begin{cases} \boldsymbol{\mathcal{R}}_k = \boldsymbol{\mathcal{R}}_{k-1} \exp([\mathbf{u}_{\text{gyro},k} - \mathbf{b}_{\text{gyro},k-1}]^\wedge \tau) \exp(\mathbf{w}_{\text{gyro},k}^\wedge \tau) \\ \boldsymbol{v}_k = \boldsymbol{v}_{k-1} + \boldsymbol{\mathcal{R}}_k(\mathbf{u}_{\text{acc},k} - \mathbf{b}_{\text{acc},k-1} + \mathbf{w}_{\text{acc},k})\,\tau + \mathbf{g}\,\tau \\ \boldsymbol{x}_k = \boldsymbol{x}_{k-1} + \boldsymbol{v}_k\,\tau \end{cases} \\
\text{IMU biases dynamics } \mathbf{b}_k : & \begin{cases} \mathbf{b}_{\text{gyro},k} = \mathbf{b}_{\text{gyro},k-1} + \mathbf{w}_{\text{gyro_bias},k}\,\tau \\ \mathbf{b}_{\text{acc},k} = \mathbf{b}_{\text{acc},k-1} + \mathbf{w}_{\text{acc_bias},k}\,\tau \end{cases} \\
\text{landmarks dynamics: } & \left\{ \boldsymbol{l}_k^{(i)} = \boldsymbol{l}_{k-1}^{(i)} \quad \forall i = 1, \ldots, N_l \right.
\end{aligned} \tag{6.16}$$

where $\boldsymbol{\mathcal{R}} \in \boldsymbol{SO}(3)$ is the body orientation, $\boldsymbol{v} \in \mathbb{R}^3$ is the body velocity, $\boldsymbol{x} \in \mathbb{R}^3$ is the body position, $\mathbf{g} = 9.81\,\boldsymbol{e}_{\mathcal{I}_3}$ is the Earth gravitational acceleration, $\mathbf{u}_{\text{gyro}} \in \mathbb{R}^3$ is the angular velocity input coming from the gyrometer, $\mathbf{u}_{\text{acc}} \in \mathbb{R}^3$ is the linear acceleration input coming from the accelerometer, and $\boldsymbol{l}^{(i)} \in \mathbb{R}^3$ denotes the i^{th} landmark expressed in the inertial frame. In this context, an element $\boldsymbol{\chi}_k \in \boldsymbol{SE}_{2+N_l}(3)$ is given as

$$\boldsymbol{\chi}_k = \left(\begin{array}{c|cccccc} \boldsymbol{\mathcal{R}}_k & \boldsymbol{v}_k & \boldsymbol{x}_k & \boldsymbol{l}_k^{(1)} & \ldots & \boldsymbol{l}_k^{(N_l)} \\ \hline \mathbf{0}_{(2+N_l)\times 3} & \multicolumn{5}{c}{\mathbf{I}_{(2+N_l)\times(2+N_l)}} \end{array} \right) \tag{6.17}$$

The equivariant uncertainty representation for this space is given by

$$\begin{cases} \boldsymbol{\chi}_k = \hat{\boldsymbol{\chi}}_k \exp(\boldsymbol{\xi}_{\mathfrak{g}_k}^\wedge) \\ \mathbf{b}_k = \hat{\mathbf{b}}_k + \boldsymbol{\xi}_{\mathbf{b}_k} \end{cases} \tag{6.18}$$

where $\hat{\boldsymbol{\chi}}_k$ and $\hat{\mathbf{b}}_k$ represent the mean (estimated) state at time k. The random vector $\boldsymbol{\xi}_{\mathfrak{g}} \in \mathbb{R}^{9+3N_l}$ contains the uncertainties in $\boldsymbol{\mathcal{R}}$, $\boldsymbol{v}$, $\boldsymbol{x}$ and $\boldsymbol{l}^{(i)}$ $(\forall i = 1, \ldots, N_l)$ as

$$\boldsymbol{\xi}_{\mathfrak{g}} = \left[\boldsymbol{\xi}_{\boldsymbol{\omega}}^\top \;\; \boldsymbol{\xi}_{\boldsymbol{v}}^\top \;\; \boldsymbol{\xi}_{\boldsymbol{x}}^\top \;\; \boldsymbol{\xi}_{\boldsymbol{l}^{(1)}}^\top \;\; \ldots \;\; \boldsymbol{\xi}_{\boldsymbol{l}^{(N_l)}}^\top \right], \tag{6.19}$$

while the random vector $\boldsymbol{\xi}_{\mathbf{b}} \in \mathbb{R}^6$ contains the biases uncertainties. These are stacked as

$$\begin{aligned}
[\boldsymbol{\xi}_{\mathfrak{g}}^\top \;\; \boldsymbol{\xi}_{\mathbf{b}}^\top]^\top &= \Big[\boldsymbol{\xi}_{\boldsymbol{\omega}}^\top \underbrace{\boldsymbol{\xi}_{\boldsymbol{v}}^\top \;\; \boldsymbol{\xi}_{\boldsymbol{x}}^\top \;\; \boldsymbol{\xi}_{\boldsymbol{l}^{(1)}}^\top \;\; \ldots \;\; \boldsymbol{\xi}_{\boldsymbol{l}^{(N_l)}}^\top}_{\boldsymbol{\xi}_{\boldsymbol{\rho}}^\top} \;\; \boldsymbol{\xi}_{\mathbf{b}}^\top \Big]^\top \in \mathbb{R}^{15+3N_l} \\
\text{with covariance, } \mathbf{P} &= \begin{pmatrix} \mathbf{P}_{\boldsymbol{\omega}\boldsymbol{\omega}} & \mathbf{P}_{\boldsymbol{\rho}\boldsymbol{\omega}}^\top \\ \mathbf{P}_{\boldsymbol{\rho}\boldsymbol{\omega}} & \mathbf{P}_{\boldsymbol{\rho}\boldsymbol{\rho}} \end{pmatrix} \in \mathbb{R}^{(15+3N_l)\times(15+3N_l)}, \quad \mathbb{E}(\boldsymbol{\xi}_{\mathfrak{g}}) = \mathbb{E}(\boldsymbol{\xi}_{\mathbf{b}}) = \mathbf{0}
\end{aligned} \tag{6.20}$$

where $\mathbf{P}_{\boldsymbol{\omega}\boldsymbol{\omega}} \in \mathbb{R}^{3\times 3}$ relates to the uncertainty in $\boldsymbol{\mathcal{R}}$, and $\mathbf{P}_{\boldsymbol{\rho}\boldsymbol{\rho}} \in \mathbb{R}^{12+3N_l}$ relates to the uncertainty in all of the remaining state variables.

In addition to the IMU measurements used as input ($\mathbf{u}_{\text{gyro}}$ and $\mathbf{u}_{\text{acc}}$) for the dynamics (6.16), the vehicle gets visual information from a calibrated monocular camera. It observes and tracks the N_l landmarks. Landmark $\boldsymbol{l}^{(i)}$ is observed through the camera as

$$\mathbf{y}^{(i)} = \frac{1}{\mathrm{y}_3^{(i)}} \begin{bmatrix} \mathrm{y}_1^{(i)} \\ \mathrm{y}_2^{(i)} \end{bmatrix} + \mathbf{v}^{(i)} \quad \forall i = 1, \ldots, N_l \tag{6.21}$$

where $\mathbf{y}^{(i)}$ is the measured normalized pixel location of the landmark in the camera frame, i.e.,

$$\mathbf{y}^{(i)} = \boldsymbol{\mathcal{R}}_C^\top \left(\boldsymbol{\mathcal{R}}^\top (\boldsymbol{l}^{(i)} - \boldsymbol{x}) - \boldsymbol{x}_C \right) \tag{6.22}$$

where the right term corresponds to the distance from the landmark to the camera expressed in the camera frame. $\boldsymbol{\mathcal{R}}_C$ and $\boldsymbol{x}_C$ are the known rotation matrix and the translation mapping from the body frame onto the camera frame. Finally, $\mathbf{v}^{(i)} \sim \mathcal{N}(\mathbf{v}^{(i)}; \mathbf{0}, \mathbf{R}^{(i)})$ represents the pixel noise. In the following experiments, the pixel noise is always assumed with a Gaussian uncertainty model.

6.5 Validations of the vMG-UKF-LG on real data

For the vMG-UKF-LG, we consider $[\boldsymbol{\xi}_{\boldsymbol{\omega}}^\top \; \boldsymbol{\xi}_{\boldsymbol{\rho}}^\top]^\top \in \mathbb{S}^3 \times \mathbb{R}^{6+3N_l}$ with the uncertainty model $[\boldsymbol{\xi}_{\boldsymbol{\omega}}^\top \; \boldsymbol{\xi}_{\boldsymbol{\rho}}^\top]^\top \sim \mathcal{VMG}(\boldsymbol{\xi}_{\boldsymbol{\omega}}, \boldsymbol{\xi}_{\boldsymbol{\rho}};\; \boldsymbol{\kappa},\, \mathbf{P}_{\boldsymbol{\rho}\boldsymbol{\rho}})$.

As for the different input noise, they are stacked as

$$\mathbf{w} = \Big[\underbrace{\mathbf{w}_{\text{gyro}}^\top}_{\mathbf{w}_{\boldsymbol{\omega}}^\top} \; \underbrace{\mathbf{w}_{\text{acc}}^\top \; \mathbf{w}_{\text{gyro_bias}}^\top \; \mathbf{w}_{\text{acc_bias}}^\top}_{\mathbf{w}_{\boldsymbol{\rho}}^\top} \Big]^\top \in \mathbb{S}^3 \times \mathbb{R}^9 \tag{6.23}$$

with the uncertainty model $\mathbf{w} \sim \mathcal{VMG}(\mathbf{w}_{\boldsymbol{\omega}}, \mathbf{w}_{\boldsymbol{\rho}};\; \boldsymbol{\kappa}_{\mathbf{u}},\, \mathbf{Q}_{\boldsymbol{\rho}})$. To parameterize the concentration coefficients κ_{u_i} of the input noise for the vMG-UKF-LG, we take again the numerical approximation $\kappa_{u_i} = \vartheta(Q_{\omega_i})$ given by (6.7) with the coefficients (6.8), and set $\boldsymbol{\kappa}_{\mathbf{u}}$ from $\mathbf{Q}_{\boldsymbol{\omega}} = \sigma_{\text{gyro}}^2 \mathbf{I}_{3\times 3}$, $\forall i = \{1, 2, 3\}$. Similarly, the concentration parameters for $\boldsymbol{\kappa}$ are extracted from $\mathbf{P}_{\boldsymbol{\omega}\boldsymbol{\omega}}$ using the same approximation model.

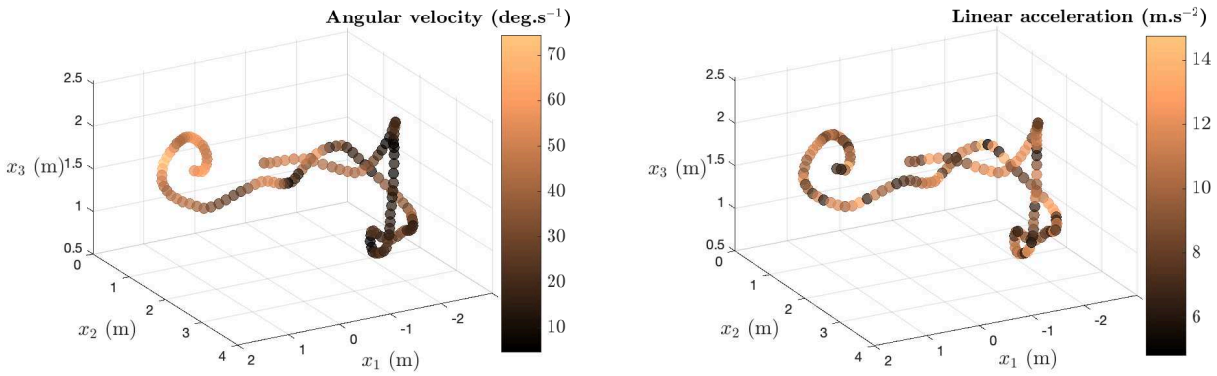
The (Right)-vMG-UKF-LG is evaluated against four benchmark filters: the (Right)-UKF-LG [Brossard et al., 2018], the SE(3)-UKF [Loianno et al., 2016], the IEKF [Barrau and Bonnabel, 2015], and a classical UKF [Van der Merwe and Wan, 2001]. All of these four baselines are parameterized in a conventional manner, assuming Gaussian noise characterized solely by covariance information. While the most direct comparison is naturally with the UKF-LG, it is also informative to assess the performance of the vMG-UKF-LG relative to these other filtering approaches. All filters are initialized at the ground truth with small initial covariance.

6.5.1 Difficult scenario

The difficult scenario features fast motions (cf. Figure 6.27) in a blurred environment. Hence, estimation errors should be ubiquitous in this setting. We perform a single filtering simulation on 20 seconds of flight where all estimators process the same sequences of inputs $\{\mathbf{u}_k\}_{k=1,\dots}$ and outputs $\{\mathbf{y}_k\}_{k=1,\dots}$ for a fair comparison.

Log-rotation errors

For this simulation, the log-rotation errors are shown in Figure 6.28. Overall, the estimation error of all filters increases along the trajectory. We attribute this behavior



(a) Norm of the angular velocity vector $||\mathbf{u}_{\text{gyro},k}||$.

(b) Norm of the linear acceleration vector $||\mathbf{u}_{\text{acc},k}||$.

Figure 6.27: Motion characteristics of the difficult scenario over a window of 20 seconds of flight.

to a likely improper handling of the visual observation measurements in the original implementation provided with [Brossard et al., 2018]. Resolving this issue lies beyond the scope of the present thesis.

Instead, the objective here is to evaluate whether the proposed vMG uncertainty model is able to mitigate the magnitude of the estimation errors produced by the UKF-LG. As illustrated in Figure 6.28, this objective is achieved for attitude estimation. Over most of the trajectory, the standard UKF-LG exhibits significantly larger error variations than the vMG-UKF-LG. This behavior is consistent and reproducible: independent simulations performed with identical hyperparameters but different measurement sequences[3] lead to qualitatively similar results. This is illustrated in Table 6.2, which reports the attitude RMSE over the 20-second flight for three independent simulations. Overall, the vMG-UKF-LG achieves the best performance, with an average RMSE of 0.62° (and ranked first for all three simulations), followed by the UKF-LG with an average RMSE of 1.81°, and the SE(3)-UKF with 1.84°. Interestingly, the standard UKF, with an average RMSE of 3.83°, outperforms the IEKF, whose average RMSE is 5.88°, in this scenario. This apparent advantage, however, is largely driven by an outlier: in simulation 2, the IEKF exhibits a significantly higher RMSE of 11.89°, whereas in simulations 1 and 3, the IEKF achieves lower RMSE values than the UKF.

	Simulation 1	Simulation 2	Simulation 3	Average
vMG-UKF-LG	0.29	0.79	0.78	0.62
UKF-LG	0.84	2.24	2.36	1.81
SE(3)-UKF	0.47	2.76	2.30	1.84
IEKF	0.89	11.89	2.86	5.88
UKF	1.79	3.27	6.42	3.83

Table 6.2: [Difficult] RMSE in rotation (°) for three independent simulations. For each simulation, the best performance is highlighted in green.

[3]Due to the processing of visual measurements, some randomness is present across independent runs.

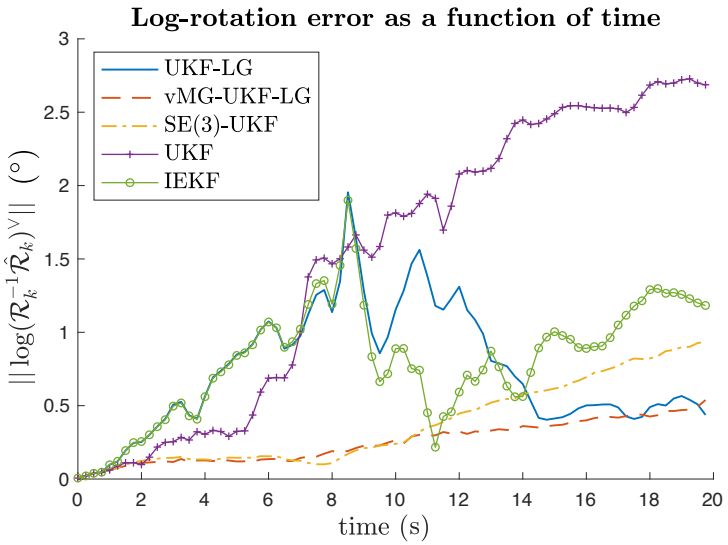


Figure 6.28: [Difficult] Attitude error along 20 seconds of flight.

A more informative analysis consists in examining the log-rotation estimation error for each component ω_i, $\forall i = \{1, 2, 3\}$, together with the corresponding uncertainty bounds expressed as 3σ standard deviations. This allows for a qualitative assessment of filter consistency. Ideally, such an analysis should rely on Monte Carlo runs; however, the runtime required to obtain the results of a single simulation was already significant. The resulting estimation errors are shown in Figure 6.29. A key observation is that the UKF-LG errors lie outside its own 3σ bounds for a large portion of the trajectory, indicating an overconfident uncertainty estimate. In contrast, the vMG-UKF-LG errors remain within the 3σ bounds predicted by the filter itself, suggesting either a more accurate characterization of its uncertainty or a conservative behavior. Another noteworthy aspect is that the 3σ bounds of the vMG-UKF-LG appear to saturate and remain bounded over time, pointing to a controlled and stable uncertainty evolution.

Position errors

The results are less encouraging with respect to the position estimates $\boldsymbol{x}_k$, as illustrated in the Figure 6.30. Overall, the vMG-UKF-LG does not demonstrate a clear improvement over the UKF-LG in terms of positional accuracy. Moreover, the consistency observed for the attitude estimates does not carry over to the position component, since the vMG-UKF-LG position errors occasionally exceed the corresponding 3σ bounds (cf. Figure 6.31). On average, the vMG-UKF-LG yields slightly inferior performance to the UKF-LG in terms of position RMSE (cf. Table 6.3). Overall, the UKF-LG achieves the best results across the three simulations, with an average RMSE of 0.51 m, closely followed by the vMG-UKF-LG with an average RMSE of 0.55 m. The IEKF ranks next with an average RMSE of 0.59 m, followed by the SE(3)-UKF at 0.61 m, and finally the standard UKF with an average RMSE of 0.66 m.

In conclusion, for the difficult scenario, the proposed vMG-UKF-LG substantially reduces attitude errors compared to the standard UKF-LG, thanks to the uncertainty mod-

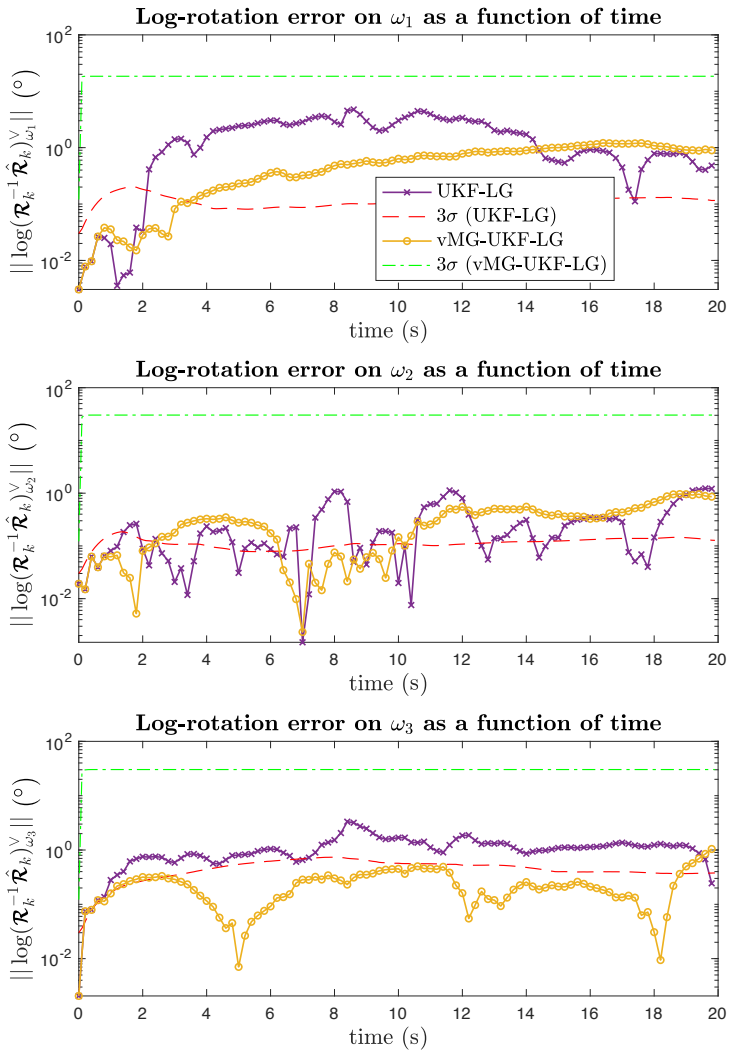


Figure 6.29: [Difficult] Log-rotation error of the variable ω_i, $\forall i = \{1, 2, 3\}$, as a function of time. These errors are those obtained from using the UKF-LG and the vMG-UKF-LG. The dotted lines are the variances given by the filters in terms of 3σ standard deviations.

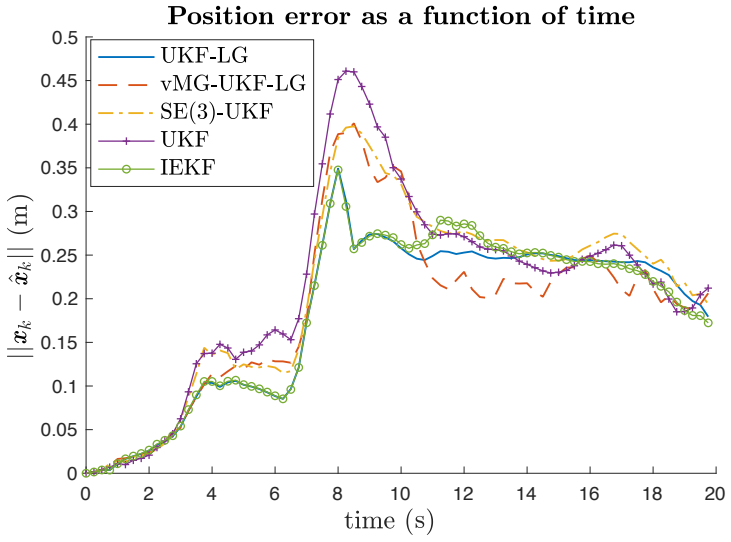


Figure 6.30: [Difficult] Position error along 20 seconds of flight.

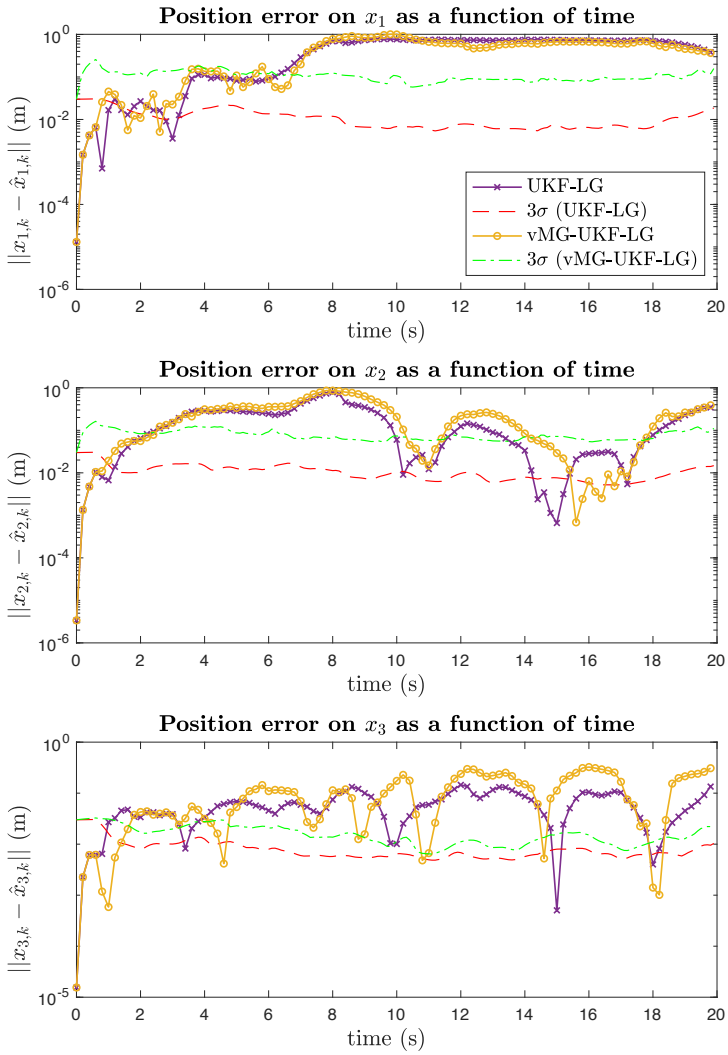


Figure 6.31: Position error of the variable x_i, $\forall i = \{1, 2, 3\}$, as a function of time. These errors are those obtained from using the UKF-LG and the vMG-UKF-LG. The dotted lines are the variances given by the filters in terms of 3σ standard deviations.

	Simulation 1	Simulation 2	Simulation 3	Average
vMG-UKF-LG	0.21	0.64	0.80	0.55
UKF-LG	0.20	0.54	0.78	0.51
SE(3)-UKF	0.23	0.67	0.93	0.61
IEKF	0.20	0.78	0.80	0.59
UKF	0.24	0.69	1.04	0.66

Table 6.3: [Difficult] RMSE in position (m) for three independent simulations. For each simulation, the best performance is highlighted in green.

eling based on the von Mises–Gaussian distribution. In challenging scenarios, filters are more prone to errors, so the uncertainty magnitude may exceed what a full Gaussian assumption can accurately represent, particularly for attitude. Applying a von Mises distribution to the rotational component helps mitigate these errors. However, since a Gaussian model is still used for the translational components (such as positions), this may explain why the proposed method does not improve position accuracy.

6.5.2 Medium scenario

Although less demanding than the difficult scenario, the medium scenario still sufficiently highlights the advantages of our proposed method over the standard UKF-LG.

Log-rotation errors

Table 6.4 presents the RMSE in attitude for three independent simulations in this scenario. While the proposed method achieves the best average performance across these simulations (average RMSE: 1.04°), some realizations of the vMG-UKF-LG do not outperform the UKF-LG, which is the case of the simulation 3, where the RMSE of the UKF-LG is 1.05°, and the RMSE of the vMG-UKF-LG is 1.40°, while the other simulations exhibit improved performance of the vMG filter against its Gaussian counterpart.

	Simulation 1	Simulation 2	Simulation 3	Average
vMG-UKF-LG	1.29	0.44	1.40	1.04
UKF-LG	1.39	1.43	1.05	1.29
SE(3)-UKF	1.90	1.00	2.15	1.68
IEKF	1.38	2.17	0.91	1.49
UKF	1.81	2.70	3.41	2.64

Table 6.4: [Medium] RMSE in rotation (°) for three independent simulations. For each simulation, the best performance is highlighted in green.

Position errors

Table 6.5 reports the position RMSE for the medium-difficulty scenario. The vMG-UKF-LG consistently exhibits slightly lower errors than the UKF-LG across all three simulations, leading to a marginally better average performance. This trend appears robust, as the improvement is observed in each run: 0.20 m versus 0.24 m in simulation 1, 0.08 m versus 0.10 m in simulation 2, and 0.20 m versus 0.22 m in simulation 3 for the vMG-UKF-LG and the UKF-LG, respectively. Nevertheless, the magnitude of these differences remains limited and does not allow for strong or definitive conclusions.

	Simulation 1	Simulation 2	Simulation 3	Average
vMG-UKF-LG	0.20	0.08	0.20	0.16
UKF-LG	0.24	0.10	0.22	0.19
SE(3)-UKF	0.19	0.11	0.18	0.16
IEKF	0.24	0.14	0.22	0.20
UKF	0.18	0.21	0.21	0.20

Table 6.5: [Medium] RMSE in position (m) for three independent simulations. For each simulation, the best performance is highlighted in green.

Comparison of execution time

It is instructive to analyze the computational cost of each filter during the prediction and update steps. To this end, several runs of 20 seconds of flight were conducted, and the average execution time of each filter was measured for the two steps. The results are reported in Table 6.6. On average, the vMG-UKF-LG requires approximately 1.35× more computation time than the UKF-LG during the prediction step and 1.27× more during the update step. This additional cost can mainly be attributed to two factors: the computation of the concentration parameters, and the more elaborate generation of sigma-points and associated weights, which relies on the numerical evaluation of ratios of Bessel $B_\nu(\kappa_i)$, $\forall i = \{1, 2, 3\}$, following the method proposed in [Kurz et al., 2013].

	prediction (sec)	update (sec)
IEKF	0.61	0.24
UKF	11.2	3.5
$SE(3)$-UKF	22.3	3.8
UKF-LG	30.4	6.2
vMG-UKF-LG	41.1	7.9

Table 6.6: Execution times of the different filters for the prediction and update. The indicated values corresponded to the average duration for 20 seconds of flight of the medium scenario.

6.5.3 Easy scenario

The easy scenario is not very challenging as it features slow motions in a bright scene. Nevertheless, we can still see some improvements from a vMG-based UKF-LG compared to a Gaussian-based UKF-LG.

Log-rotation errors

With respect to log-rotation errors (cf. Table 6.7), neither the UKF-LG nor the vMG-UKF-LG achieves the best performance in the easy scenario. In fact, the UKF-LG exhibits the poorest performance, with an average RMSE of 8.80° over the three simulations, even exceeding that of the standard UKF and the IEKF, whose average RMSE values are 6.60° and 6.99°, respectively. By contrast, the vMG-UKF-LG yields a substantial improvement, reducing the average RMSE to 3.10° and ranking second just behind the SE(3)-UKF, which achieves an average RMSE of 2.79°.

Position errors

As for the position RMSE in the easy-difficulty scenario (Table 6.8), our vMG-based filter does not outperform the Gaussian-based filter, which can be ranked first in two out of thee simulations, leading to an average RMSE of 0.40°, yet looses to the IEKF (average RMSE: 0.34°). It turns out that in this case, the vMG-based filter is ranked last among of filters.

	Simulation 1	Simulation 2	Simulation 3	Average
vMG-UKF-LG	2.92	3.24	3.15	3.10
UKF-LG	2.60	12.75	9.06	8.80
SE(3)-UKF	2.31	3.29	2.76	2.79
IEKF	2.63	12.74	3.60	6.99
UKF	5.29	9.45	5.05	6.60

Table 6.7: [Easy] RMSE in rotation (°) for three independent simulations. For each simulation, the best performance is highlighted in green.

	Simulation 1	Simulation 2	Simulation 3	Average
vMG-UKF-LG	0.40	0.64	0.77	0.60
UKF-LG	0.23	0.26	0.72	0.40
SE(3)-UKF	0.30	0.58	0.74	0.54
IEKF	0.24	0.28	0.49	0.34
UKF	0.32	0.48	0.81	0.54

Table 6.8: [Easy] RMSE in position (m) for three independent simulations. For each simulation, the best performance is highlighted in green.

■ Concluding remarks:

In all of the three scenarios, the vMG-based UKF-LG achieved, on average, the best performance (relatively to the Gaussian-based UKF-LG) regarding the accuracy of the attitude estimation. This improvement on the attitude part, does not necessarily carry on the position part, although there exists slight improvements for the medium-difficulty scenario. However, the reduction of attitude error provided by the vMG-based filter is systematic, since almost every independent simulations (for all three scenarios) resulted in an overperformance of the vMG-UKF-LG against the UKF-LG. This is an encouraging result, and demonstrates the interest of using our proposed equivariant von Mises-Gaussian for state estmation of a variable evolving on motion groups. Finally, we also assessed the performance of the OQ-UKF-LG on these datasets. However, the gain performance provided by the optimal quantization approach was only marginal, even for the difficult scenario. Some results can be found in the Appendix (§B.5.2).

🔑 As final conclusions, both algorithms (vMG-UKF-LG and OQ-UKF-LG) have been evaluated on simulated data under their validity assumptions. In this case, they both provide enhanced performances against the benchmark algorithm of the standard UKF-LG for most of the case studied, which is precisely one of the declared objectives of this thesis. Evaluations on real data also validated the proposed approaches, especially the vMG-UKF-LG, which systematically improves the rotational estimation accuracy compared to the standard UKF-LG. A remarkable result showing that the vMG-based method is of particular relevance for orientation estimation.

Chapter 7

Conclusions and Perspectives

Contents

7.1 Main contributions 149
7.1.1 An alternative to the equivariant Gaussian on $\boldsymbol{SE}(n)$ 149
7.1.2 An optimal sigma-point placement 150
7.2 Identified limits and drawbacks 150
7.2.1 Concerning the equivariant von Mises-Gaussian 150
7.2.2 Concerning the optimal sigma-point placement 151
7.3 Perspectives 151

This thesis focused specifically on issues related to the **characterization and propagation of uncertainty on Lie groups**, with the objective of improving the UKF-LG framework. In particular, **two main challenges** have been identified.

1. The so-called "Lie-theoretic" covariance used within the concentrated Gaussian assumption is only locally (or almost-globally) valid. Consequently, this statistical moment may be insufficient if used as the sole measure of dispersion when the underlying distribution is broad.

2. Even when the distribution is sufficiently concentrated to justify a Gaussian assumption, the characterization and propagation of moments through sigma-point placement and propagation may still fail to provide accurate estimates; a limitation well known in classical UKF formulations.

7.1 Main contributions

With regard to the two challenges stated previously, the main contributions of the thesis are summarized below.

7.1.1 An alternative to the equivariant Gaussian on $\boldsymbol{SE}(n)$

The contribution in this field concerns an alternative probabilistic model for characterizing uncertainty on motion groups such as $\boldsymbol{SE}(n)$, and naturally extends to augmented

motion groups $\boldsymbol{SE}_k(n)$ as well as to product manifolds of the form $\boldsymbol{SE}_k(n)\times\mathbb{R}^N$. Referred to as the **equivariant von Mises–Gaussian** (equivariant-vMG), this model combines the structural simplicity of the equivariant framework with the geometric consistency of spherical distributions. As such, it can be interpreted as a principled extension of the equivariant Gaussian, particularly suited to scenarios involving large dispersions. Building on this formulation, the sigma-point transformation on Lie groups has been adapted to match the proposed uncertainty representation. This led to the development of a revised UKF-LG, termed the **vMG-UKF-LG**, which explicitly accounts for both covariance and concentration parameters when generating sigma-points, rather than relying on covariance alone.

7.1.2 An optimal sigma-point placement

Under the assumption that the uncertainty is small enough to consider the Lie-theoretic covariance to remain valid and sufficient, a second contribution of the thesis concerns the accurate placement of sigma-points following a Gaussian distribution. This approach, based on **optimal quantization**, aims to readjust the sigma-points by revisiting a CLVQ algorithm which minimizes a cost criterion. This development has been beneficial to the UT on Lie groups, but can apply to any sort of UT (as long as we consider Gaussian assumptions). Then, the optimal quantization-based UT has been applied in a UKF-LG structure to come up with an **OQ-UKF-LG**, which exhibits several improvements over the standard UKF-LG.

7.2 Identified limits and drawbacks

With regard to these contributions, several limitations have emerged in both the **modeling assumptions** and the **algorithmic implementation**.

7.2.1 Concerning the equivariant von Mises-Gaussian

Fundamentally, the proposed probability still lacks **modeling accuracy** in its definition, in the sense that:

- The von Mises part of the vMG does not model inter-correlations between the different components of $\boldsymbol{\omega}$.
- The vMG on $\mathbb{S}^m\times\mathbb{R}^\ell$ does not correctly model cross-correlations between the angular part $\boldsymbol{\omega}$ and the translational part $\boldsymbol{\rho}$, as it rely on an approximation.

These modeling limitations directly affect the sigma-point generation within the Unscented Transform, which aims at approximating the underlying distribution through moment matching, while some of these moments are actually approximated representations. Consequently, the proposed vMG model remains inherently limited in its ability to accurately capture the true uncertainty of random variables evolving on a spherical–Euclidean state space, which in turn may limit the achievable fidelity of the resulting filtering scheme.

Moreover, to be able to adapt it to a Kalman filter algorithm (whose update equations use only the mean $\hat{\boldsymbol{\chi}}$ and the full covariance $\mathbf{P}$), we assumed a one-to-one relationship between the diagonal components of the covariance of $\boldsymbol{\omega}$ and the concentration parameters $(\kappa_1, \kappa_2, \kappa_3)$, so that we infer them from the updated $\mathbf{P}_{\boldsymbol{\omega}\boldsymbol{\omega}}$. Actually, there exists a one-to-one relationship between the notion of circular variance and the concentration parameter, but we are not sure if this kind of relationship still holds with regard to the standard variance.

In terms of **algorithmic implementation**, the proposed vMG-UKF-LG has several drawbacks:

- It requires more computations, as it aims to estimate additional statistical moments along the covariance and the mean, thereby increasing computation time.
- The generation of the sigma-points and weights requires numerical approximations of the ratios of Bessel, thereby adding computational cost and numerical errors.
- An approximation method has been used to reconstruct the concentration parameters from the propagated sample of the prediction UT, providing further numerical errors.
- At present, no rigorous or analytical correction scheme is available for updating the concentration parameters in the proposed vMG-UKF-LG. In this thesis, a provisional approach is adopted, where these parameters are inferred from the updated covariance matrix (an assumption whose theoretical validity remains open to discussion).

7.2.2 Concerning the optimal sigma-point placement

In terms of **algorithmic implementation**, adding an optimal quantization procedure in the filtering process extends the computational cost. In particular, the revised CLVQ algorithm has additional tuning parameters that impose some limitations:

- The number of steps n_{max} must be chosen large enough so as to make the stochastic gradient descent to converge, but low enough to remain computationally low-cost.
- The CLVQ has as many learning rates γ as there are variables in the state space. Besides, these learning rates must be hand-tuned by trial and error. For the time being, the current method does not propose a solution on how to tune these parameters in a systematic fashion.

In terms of **probabilistic modeling**, the method still aims to capture a Gaussian with prescribed moments and has not been extended to other distributions yet.

7.3 Perspectives

With regard to these two methodological developments, their validations on simulations, and their respective limitations, this thesis naturally gives rise to several research

perspectives, with the objective of overcoming the underlying assumptions and further extending the proposed developments.

A first perspective concerns the **refinement of probabilistic modeling on Lie groups**, notably by improving the equivariant von Mises–Gaussian framework to better capture cross-correlations and higher-order dependencies, as well as by establishing more rigorous links between concentration parameters and statistical moments. For more accurate uncertainty modeling within the vMG framework, one could consider exploiting the multivariate von Mises distribution that we already saw in subsection §3.4.2, in particular through its angular correlation parameter $\mathbf{\Lambda}$. This parameter encodes dependencies between angular variables and could, in principle, be inferred at each Kalman iteration from the off-diagonal entries of the angular covariance matrix $\mathbf{P}_{\omega\omega}$. A key open question, however, concerns how such correlation information should be incorporated into the sigma-point generation at the subsequent iteration. This issue was not addressed in the present thesis, but it constitutes a promising direction for future work. Addressing it would likely require the introduction of additional moment constraints, beyond marginal circular moments, in order to define sigma-points that are consistent with both concentration and correlation properties of the mvM distribution. Another direction of development could be the Matrix Fisher Gaussian approach (cf. §3.4.3), which could potentially address the approximation we made on the rotational/translational cross-correlations. Beyond the particular case of robotic navigation, extending uncertainty representations to richer families of distributions while remaining compatible with Kalman-type filtering schemes constitutes an important avenue. More generally, within the broader framework of Bayesian filtering, adopting probability models derived from the principle of maximum entropy distribution appears as a particularly promising direction.

A second perspective relates to the optimal quantization-based UT and its **algorithmic implementation**: future work should focus on reducing its computational burden, proposing principled strategies for tuning its hyperparameters, and extending the approach to non-Gaussian distributions.

Finally, validating these developments on **larger-scale**, real-world robotic systems and examining how they interact with **control laws** and **decision-making architectures** would help close the gap between theoretical developments and practical implementation. Recent approaches, for instance, exploit the covariance provided by stochastic estimators to explicitly account for uncertainty in control design. These are known as “covariance-steering” control laws. In this regard, enriching probabilistic models with more expressive and accurate uncertainty representations (as suggested in the present thesis) has the potential to significantly enhance the effectiveness and reliability of such control strategies.

Appendix A

Probability Materials

A.1 Usual definitions

Definition A.1 (Probability space)**.** A random vector $\mathbf{x}$ is defined on a probability space $\{\Omega, \mathcal{F}, \mathcal{P}\}$ consisting of a triple of elements:

- A sample space Ω, which is the set of all possible outcomes of $\mathbf{x}$.
- An event space $\mathcal{F}$, which is a set of events, where an event is a subset of outcomes in the sample space.
- A probability density function $\mathcal{P}$, which assigns, to each event in the event space, a probability, which is a number between 0 and 1.

Definition A.2 (Conditional probability)**.** Given two random events A and B, the conditional probability $\mathcal{P}(A|B)$ (the probability of A when B is realized) is expressed as

$$\mathcal{P}(A|B) = \frac{\mathcal{P}(A,B)}{\mathcal{P}(B)} \tag{A.1}$$

This expression can be extended in the case we consider three events A, B, C and looking for the conditional probability $\mathcal{P}(A|B,C)$. In this case, its expression is given by

$$\mathcal{P}(A|B,C) = \frac{\mathcal{P}(A,B|C)}{\mathcal{P}(B|C)} \tag{A.2}$$

Proof. Starting from $\mathcal{P}\big(A|(B,C)\big)$ and using the conditional probability (A.1), we get

$$\mathcal{P}\big(A|(B,C)\big) = \frac{\mathcal{P}\big(A,(B,C)\big)}{\mathcal{P}(B,C)} = \frac{\mathcal{P}\big(A,(B,C)\big)}{\mathcal{P}(B|C)\mathcal{P}(C)}$$

But since $\mathcal{P}\big(A,(B,C)\big) = \mathcal{P}\big((A,B),C\big) = \mathcal{P}\big((A,B)|C\big)\mathcal{P}(C)$, then we have

$$\mathcal{P}(A|B,C) = \frac{\mathcal{P}\big(A,B|C\big)\cancel{\mathcal{P}(C)}}{\mathcal{P}(B|C)\cancel{\mathcal{P}(C)}}$$

□

Definition A.3 (Bayes rule)**.** Given two random events A and B, the conditional probability $\mathcal{P}(A|B)$ (also called *posterior* probability density) is given by Bayes rule which is expressed as

$$\mathcal{P}(A|B) = \frac{\mathcal{P}(B|A)\mathcal{P}(A)}{\mathcal{P}(B)} \tag{A.3}$$

The term $\mathcal{P}(A)$ is the *prior* probability of the event A and $\mathcal{P}(B|A)$ is the *likelihood* of B when A is realized. The term $\mathcal{P}(B)$ is the total probability of B and serves as a normalization factor. The proof is trivial and derives directly from the definition of conditional probability since $\mathcal{P}(A, B) = \mathcal{P}(B, A) = \mathcal{P}(B|A)\mathcal{P}(A)$.

Definition A.4 (Independency)**.** Two random events A and B are said to be *independent* if we have the joint probability

$$\mathcal{P}(A, B) = \mathcal{P}(A)\mathcal{P}(B) \tag{A.4}$$

The proof is trivial and derives directly from the expression of conditional probability. Indeed, from (A.1), we have $\mathcal{P}(A, B) = \mathcal{P}(A|B)\mathcal{P}(B)$ but since event A is said to be independent from event B, then $\mathcal{P}(A|B) = \mathcal{P}(A)$.

This definition generalizes to m random events. The set of random events $(A_1, ..., A_m)$ is said to be *mutually independent* if

$$\mathcal{P}(A_1, ..., A_m) = \prod_{i=1}^{m} \mathcal{P}(A_i) \tag{A.5}$$

Definition A.5 (n^{th}-order moments)**.** For a continuous random variable $x \in I$ that follows a probability distribution $\mathcal{P}(x)$, the n^{th} (raw) moment of x, denoted $\mathbb{E}[x^n]$, is defined as

$$\mathbb{E}[x^n] = \int_I x^n \mathcal{P}(x) dx \tag{A.6}$$

Example: The 1^{st} moment $\mathbb{E}[x] = \int_I x\mathcal{P}(x)dx$ is commonly refered as the *mean* of x.
Instead of the n^{th} raw moment, we can define the n^{th} central moment, denoted $\mathbb{E}\Big[(x - \mathbb{E}[x])^n\Big]$, and provides informations on the shape of the distribution

$$\mathbb{E}\Big[(x - \mathbb{E}[x])^n\Big] = \int_I (x - \mathbb{E}[x])^n \mathcal{P}(x) dx \tag{A.7}$$

Example: The 2^{nd} central moment is referred as the *covariance* of x.
Similar definitions hold for an $\mathbb{R}^d$-valued random vector $\mathbf{x}$.

Definition A.6 (Probability measure)**.** Let $\mathbf{x} \in \mathbb{R}^d$ be a continuous random vector. A probability measure $\mu(\mathbf{x})$ on an interval $I \subset \mathbb{R}^d$ is defined as

$$\mu(\mathbf{x}) := \int_I \mathcal{P}(\mathbf{x}) d\mathbf{x} \tag{A.8}$$

A probability measure returns a value between 0 and 1.

Definition A.7 (Characteristic function)**.** For a d-dimensional random vector $\mathbf{x}$ and $\mathbf{a} \in \mathbb{R}^d$, the characteristic function is defined as

$$\phi_{\mathbf{x}}(\mathbf{a}) := \mathbb{E}[e^{i\mathbf{a}^\top \mathbf{x}}] = \int_{\mathbb{R}^d} e^{i\mathbf{a}^\top \mathbf{x}} \mathcal{P}(\mathbf{x}) d\mathbf{x} \tag{A.9}$$

The characteristic function of any real-valued random vector completely defines its probability distribution in a unique way. Besides, moments of the random vector can be deduced from its characteristic function.

A.2 Proof of the recursive Bayesian filter

The proof of the update equation (2.7) of the recursive Bayesian filter relies on Bayes' rule (Definition A.3) and conditional probability (Definition A.2).

Proof.

$$\begin{aligned}
\mathcal{P}(\mathbf{x}_k|\mathbf{y}_{0:k}) &= \frac{\mathcal{P}(\mathbf{y}_{0:k}|\mathbf{x}_k)\mathcal{P}(\mathbf{x}_k)}{\mathcal{P}(\mathbf{y}_{0:k})} = \frac{\mathcal{P}(\mathbf{y}_k, \mathbf{y}_{0:k-1}|\mathbf{x}_k)\mathcal{P}(\mathbf{x}_k)}{\mathcal{P}(\mathbf{y}_k, \mathbf{y}_{0:k-1})} \\
&= \frac{\mathcal{P}(\mathbf{y}_k|\mathbf{y}_{0:k-1}, \mathbf{x}_k)\mathcal{P}(\mathbf{y}_{0:k-1}|\mathbf{x}_k)\mathcal{P}(\mathbf{x}_k)}{\mathcal{P}(\mathbf{y}_k|\mathbf{y}_{0:k-1})\mathcal{P}(\mathbf{y}_{0:k-1})} \\
&= \frac{\mathcal{P}(\mathbf{y}_k|\mathbf{y}_{0:k-1}, \mathbf{x}_k)}{\mathcal{P}(\mathbf{y}_k|\mathbf{y}_{0:k-1})\cancel{\mathcal{P}(\mathbf{y}_{0:k-1})}} \frac{\mathcal{P}(\mathbf{x}_k|\mathbf{y}_{0:k-1})\cancel{\mathcal{P}(\mathbf{y}_{0:k-1})}}{\cancel{\mathcal{P}(\mathbf{x}_k)}} \cancel{\mathcal{P}(\mathbf{x}_k)} \\
&= \frac{\mathcal{P}(\mathbf{y}_k, \mathbf{y}_{0:k-1}|\mathbf{x}_k)}{\mathcal{P}(\mathbf{y}_{0:k-1}|\mathbf{x}_k)} \frac{\mathcal{P}(\mathbf{x}_k|\mathbf{y}_{0:k-1})}{\mathcal{P}(\mathbf{y}_k|\mathbf{y}_{0:k-1})} \\
&= \frac{\mathcal{P}(\mathbf{y}_k|\mathbf{y}_{0:k-1}, \mathbf{x}_k)\cancel{\mathcal{P}(\mathbf{y}_{0:k-1}|\mathbf{x}_k)}}{\cancel{\mathcal{P}(\mathbf{y}_{0:k-1}|\mathbf{x}_k)}} \frac{\mathcal{P}(\mathbf{x}_k|\mathbf{y}_{0:k-1})}{\mathcal{P}(\mathbf{y}_k|\mathbf{y}_{0:k-1})} \\
&= \frac{\mathcal{P}(\mathbf{y}_k|\mathbf{x}_k)\mathcal{P}(\mathbf{x}_k|\mathbf{y}_{0:k-1})}{\mathcal{P}(\mathbf{y}_k|\mathbf{y}_{0:k-1})}
\end{aligned}$$

where we used the independence of $\mathbf{y}_k$ from past measurements $\mathbf{y}_{0:k-1}$ to get $\mathcal{P}(\mathbf{y}_k|\mathbf{y}_{0:k-1}, \mathbf{x}_k) = \mathcal{P}(\mathbf{y}_k|\mathbf{x}_k)$. This result is non parametric (i.e. it is true for all types of probability density functions). □

Appendix B

Additional details about Lie groups

This appendix provides details on Lie groups that have been overlooked in the main contents of the manuscript. For further details, see [Solà et al., 2021][Chirikjian, 2011][Eade, 2013]. Table B.1 provides a list of usual Lie groups with their respective characteristics.

Manifold $\boldsymbol{G}$	Description	Group operator $\circ$	$\dim(\boldsymbol{G})$
$\mathbb{R}^d$	Euclidean space	addition "+"	d
$\mathbb{S}^1$	Unit complex group whose elements are defined in the 1-sphere	multiplication "$\cdot$"	1
$\mathbb{S}^3$	Unit quaternion group whose elements are in the 3-sphere, also denoted by Sp(1)	multiplication "$\cdot$"	3
$\boldsymbol{GL}(n)$	General Linear group = set of invertible matrices of size $n \times n$	matrix multiplication	n^2
$\boldsymbol{SL}(n)$	Special Linear group = subset of $\boldsymbol{Gl}(n)$ with determinant $+1$	matrix multiplication	$n^2 - 1$
$\boldsymbol{O}(n)$	Orthogonal group = set of orthogonal matrices of size $n \times n$	matrix multiplication	$\frac{n(n-1)}{2}$
$\boldsymbol{SO}(n)$	Rotation (Special Orthogonal) group in an n-dimensional space = subset of $\boldsymbol{O}(n)$ with determinant $+1$	matrix multiplication	$\frac{n(n-1)}{2}$
$\boldsymbol{SE}(n)$	Motion (Special Euclidean) group in an n-dimensional space	matrix multiplication	$\frac{n(n+1)}{2}$

Table B.1: Some examples of Lie groups and their characteristics.

B.1 Demonstration of the group jacobians

Let $\mathbf{x}, \mathbf{y}, \mathbf{z} \in \mathbb{R}^{\dim(\boldsymbol{G})}$. Recall that the BCH formula is given by

$$\begin{aligned} &\log\big(\exp(\mathbf{z}^\wedge)\big)^\vee = \log\big(\exp(\mathbf{x}^\wedge)\exp(\mathbf{y}^\wedge)\big)^\vee = \mathbf{z} = \mathbf{x} + \mathbf{y} + \frac{1}{2}\mathrm{ad}(\mathbf{x}^\wedge)\mathbf{y} \\ &+ \frac{1}{12}\mathrm{ad}(\mathbf{x}^\wedge)\mathrm{ad}(\mathbf{x}^\wedge)\mathbf{y} + \frac{1}{12}\mathrm{ad}(\mathbf{y}^\wedge)\mathrm{ad}(\mathbf{y}^\wedge)\mathbf{x} - \frac{1}{24}\mathrm{ad}(\mathbf{y}^\wedge)\mathrm{ad}(\mathbf{x}^\wedge)\mathrm{ad}(\mathbf{x}^\wedge)\mathbf{y} + \ldots \end{aligned} \tag{B.1}$$

If we assume $\mathbf{y}$ small with respect to $\mathbf{x}$, then the serie can be factorized by $\mathbf{y}$ to yield an approximation of the BCH

$$\mathbf{z} \approx \mathbf{x} + \underbrace{\left(\sum_{k=0}^{+\infty} \frac{B_k}{k!} \big(- \operatorname{ad}(\mathbf{x}^\wedge) \big)^k \right)}_{\boldsymbol{\mathcal{J}}_{\mathsf{R}}^{-1}\left(\operatorname{ad}(\mathbf{x}^\wedge)\right)} \mathbf{y} \tag{B.2}$$

where the $\{B_k\}_{k \geq 0}$ are the *Bernoulli numbers*

$$B_0 = 1,\, B_1 = -\frac{1}{2},\, B_2 = \frac{1}{6},\, B_3 = 0,\, B_4 = -\frac{1}{30},\, B_5 = 0,\, \ldots, \tag{B.3}$$

The sum inside the parentheses is called the inverse of the (right) group jacobian. Inverting it, we get the right jacobian

$$\boldsymbol{\mathcal{J}}_{\mathsf{R}}\big(\operatorname{ad}(\mathbf{x}^\wedge)\big) = \sum_{k=0}^{+\infty} \frac{1}{(k+1)!} \big(- \operatorname{ad}(\mathbf{x}^\wedge) \big)^k \tag{B.4}$$

If we had assumed $\mathbf{x}$ small with respect to $\mathbf{y}$, then we would have obtained

$$\mathbf{z} \approx \underbrace{\left(\sum_{k=0}^{+\infty} \frac{B_k}{k!} \big(\operatorname{ad}(\mathbf{x}^\wedge) \big)^k \right)}_{\boldsymbol{\mathcal{J}}_{\mathsf{L}}^{-1}\left(\operatorname{ad}(\mathbf{x}^\wedge)\right)} \mathbf{x} + \mathbf{y} \tag{B.5}$$

and get the left group jacobian, which is expressed as

$$\boldsymbol{\mathcal{J}}_{\mathsf{L}}\big(\operatorname{ad}(\mathbf{x}^\wedge)\big) = \sum_{k=0}^{+\infty} \frac{1}{(k+1)!} \big(\operatorname{ad}(\mathbf{x}^\wedge) \big)^k \tag{B.6}$$

Hence, we have the following relationship:

$$\boldsymbol{\mathcal{J}}_{\mathsf{R}}\big(\operatorname{ad}(\mathbf{x}^\wedge)\big) = \boldsymbol{\mathcal{J}}_{\mathsf{L}}\big(- \operatorname{ad}(\mathbf{x}^\wedge) \big) \tag{B.7}$$

B.2 Usual matrix Lie groups

We provides further details on the Lie groups of interest in this thesis. In particular, their exponential, logarithmic, Adjoint and small adjoint maps are given in closed-form.

B.2.1 Special Orthogonal groups $\boldsymbol{SO}(n)$

The Special Orthogonal group $\boldsymbol{SO}(n)$ is the set of $n \times n$ symmetric matrices with determinant $+1$.

$$\boldsymbol{SO}(n) = \Big\{ \boldsymbol{\mathcal{R}} \in \mathcal{M}_{n \times n}(\mathbb{R}) \;\Big|\; \boldsymbol{\mathcal{R}}^\top \boldsymbol{\mathcal{R}} = \boldsymbol{I}_{n \times n},\, \det(\boldsymbol{\mathcal{R}}) = 1 \Big\} \tag{B.8}$$

The condition $\boldsymbol{\mathcal{R}}^\top \boldsymbol{\mathcal{R}} = \boldsymbol{I}_{n \times n}$ implies $\boldsymbol{\mathcal{R}}^\top = \boldsymbol{\mathcal{R}}^{-1}$. Its dimension is

$$\dim(\boldsymbol{SO}(n)) = d = \begin{cases} 1 \text{ for } n = 2 \\ 3 \text{ for } n = 3 \end{cases}$$

Its Lie algebra $\mathfrak{so}(n)$ is the set of $n \times n$ skew-symmetric matrices.

$$\mathfrak{so}(n) = \left\{ \boldsymbol{A} \in \mathcal{M}_{n\times n}(\mathbb{R}) \;\middle|\; \boldsymbol{A} = -\boldsymbol{A}^\top \right\} \tag{B.9}$$

The wedge and vee operators are given as follows:

- if $n = 2 \Rightarrow \boldsymbol{\omega} = \theta \in \mathbb{R}$, then $[\cdot]^\wedge : \mathbb{R} \mapsto \mathfrak{so}(2)$ s.t.

$$\boldsymbol{\omega}^\wedge = \begin{pmatrix} 0 & -\theta \\ \theta & 0 \end{pmatrix} \tag{B.10}$$

  inversely, $(\boldsymbol{\omega}^\wedge)^\vee = \theta$

- if $n = 3 \Rightarrow \boldsymbol{\omega} \in \mathbb{R}^3$, then $[\cdot]^\wedge : \mathbb{R}^3 \mapsto \mathfrak{so}(3)$ s.t.

$$\boldsymbol{\omega}^\wedge = \begin{pmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{pmatrix} \tag{B.11}$$

  inversely, $(\boldsymbol{\omega}^\wedge)^\vee = [\omega_1, \omega_2, \omega_3]^\top$

■ Exponential map $\boldsymbol{SO}(n)$

- for $n = 2$:

$$\exp(\boldsymbol{\omega}^\wedge) = \boldsymbol{\mathcal{R}}(\theta) = \begin{pmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{pmatrix} \tag{B.12}$$

- for $n = 3$:

$$\begin{aligned} &\exp(\boldsymbol{\omega}^\wedge) = \boldsymbol{\mathcal{R}}(\boldsymbol{\omega}) = \boldsymbol{I}_{3\times 3} + \frac{\sin(\theta)}{\theta}\boldsymbol{\omega}^\wedge + \frac{\big(1-\cos(\theta)\big)}{\theta^2}(\boldsymbol{\omega}^\wedge)^2 \\ &\text{where } \theta = ||\boldsymbol{\omega}|| = \sqrt{\boldsymbol{\omega}^\top \boldsymbol{\omega}} \end{aligned} \tag{B.13}$$

The demonstrations of (B.12) and (B.13) can be found in [Eade, 2013]. The expression (B.13) is the so-called Rodrigues formula. The exponential map yields a rotation by θ radians around the axis carried by vector the $\boldsymbol{\omega}$.

■ Logarithm map $\boldsymbol{SO}(n)$

- for $n = 2$:

$$\log(\boldsymbol{\mathcal{R}}) = \underbrace{\arctan(\mathcal{R}_{21}, \mathcal{R}_{11})}_{\theta} \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} = \begin{pmatrix} 0 & -\theta \\ \theta & 0 \end{pmatrix} = \boldsymbol{\omega}^\wedge \tag{B.14}$$

- for $n = 3$:

$$\begin{aligned} \theta &= \arccos^{-1}\left(\frac{\mathrm{tr}(\boldsymbol{\mathcal{R}}) - 1}{2}\right) := ||\boldsymbol{\omega}|| \\ \log(\boldsymbol{\mathcal{R}}) &= \frac{\theta}{2\sin(\theta)}(\boldsymbol{\mathcal{R}} - \boldsymbol{\mathcal{R}}^\top) = \boldsymbol{\omega}^\wedge \end{aligned} \tag{B.15}$$

■ Adjoint $\boldsymbol{SO}(n)$

- for $n = 2$:

$$\forall \boldsymbol{\mathcal{R}} \in \boldsymbol{SO}(2), \quad \mathrm{Ad}(\boldsymbol{\mathcal{R}}) = \mathbf{I}_{2\times 2} \tag{B.16}$$

- for $n = 3$:

$$\forall \boldsymbol{\mathcal{R}} \in \boldsymbol{SO}(3), \quad \mathrm{Ad}(\boldsymbol{\mathcal{R}}) = \boldsymbol{\mathcal{R}} \tag{B.17}$$

■ **Small adjoint $\boldsymbol{SO}(n)$**

- for $n = 2$:

$$\forall \theta^\wedge \in \mathfrak{so}(2), \quad \mathrm{ad}(\theta^\wedge) = \mathbf{0}_{2\times 2} \tag{B.18}$$

- for $n = 3$:

$$\forall \boldsymbol{\omega}^\wedge \in \mathfrak{so}(3), \quad \mathrm{ad}(\boldsymbol{\omega}^\wedge) = \boldsymbol{\omega}^\wedge \tag{B.19}$$

B.2.2 Special Euclidean groups $\boldsymbol{SE}(n)$

The Special Euclidean group, also called group of direct isometries, or simply group of rigid-body motions, denoted by $\boldsymbol{SE}(n)$, consist of elements $\boldsymbol{\chi} \in \boldsymbol{SE}(n)$ as defined by

$$\boldsymbol{\chi} = \begin{pmatrix} \boldsymbol{\mathcal{R}} & \boldsymbol{x} \\ \mathbf{0}_{1\times n} & 1 \end{pmatrix}, \quad \boldsymbol{\mathcal{R}} \in \boldsymbol{SO}(n),\ \boldsymbol{x} \in \mathbb{R}^n \tag{B.20}$$

and its inverse is given by

$$\boldsymbol{\chi}^{-1} = \begin{pmatrix} \boldsymbol{\mathcal{R}}^\top & -\boldsymbol{\mathcal{R}}^\top \boldsymbol{x} \\ \mathbf{0}_{1\times n} & 1 \end{pmatrix} \tag{B.21}$$

Its dimension is

$$\dim(\boldsymbol{SE}(n)) = d = \begin{cases} 3 \text{ for } n = 2 \\ 6 \text{ for } n = 3 \end{cases}$$

Therefore, the Lie algebra $\mathfrak{se}(n)$ is a vector space of associated dimension d. An element $\boldsymbol{\xi}^\wedge \in \mathfrak{se}(n)$ is defined as follows

$$\boldsymbol{\xi}^\wedge = \begin{pmatrix} \boldsymbol{\omega}^\wedge & \boldsymbol{\rho} \\ \mathbf{0}_{1\times n} & 0 \end{pmatrix}, \quad \boldsymbol{\omega} \in \begin{cases} \mathbb{R} \text{ (if } n = 2) \\ \mathbb{R}^3 \text{ (if } n = 3) \end{cases}, \quad \boldsymbol{\rho} \in \begin{cases} \mathbb{R}^2 \text{ (if } n = 2) \\ \mathbb{R}^3 \text{ (if } n = 3) \end{cases} \tag{B.22}$$

where $\boldsymbol{\omega}^\wedge$ is the isomorphism defined by (B.10) or (B.11), depending on the value of n. The vector $\boldsymbol{\xi} \in \mathbb{R}^d$ is simply the concatenation of the rotational vector $\boldsymbol{\omega}$ and the translational vector $\boldsymbol{\rho}$:

$$\boldsymbol{\xi} = \begin{bmatrix} \boldsymbol{\omega} \\ \boldsymbol{\rho} \end{bmatrix}$$

- for $n = 2 \rightarrow \boldsymbol{\xi} = [\xi_1, \xi_2, \xi_3]^\top = [\theta, \rho_1, \rho_2]^\top$
- for $n = 3 \rightarrow \boldsymbol{\xi} = [\xi_1, \xi_2, \xi_3, \xi_4, \xi_5, \xi_6]^\top = [\omega_1, \omega_2, \omega_3, \rho_1, \rho_2, \rho_3]^\top$

■ **Exponential map $\boldsymbol{SE}(n)$**

- for $n = 2$:

  First, compute $\boldsymbol{\mathcal{R}}(\theta)$ using expression (B.12). Then, we have

$$\boldsymbol{\chi} = \exp(\boldsymbol{\xi}^\wedge) = \begin{pmatrix} \boldsymbol{\mathcal{R}}(\theta) & \mathbf{J}_\mathsf{L} \begin{bmatrix} \rho_1 \\ \rho_2 \end{bmatrix} \\ \mathbf{0}_{1\times 2} & 1 \end{pmatrix} \tag{B.23}$$

$$\text{where } \mathbf{J}_\mathsf{L} = \frac{1}{\theta} \begin{pmatrix} \sin(\theta) & -1 + \cos(\theta) \\ 1 - \cos(\theta) & \sin(\theta) \end{pmatrix}$$

The matrix $\mathbf{J}_\mathsf{L}$ comes from the power series expansion of the exponential map in closed-form (see section §5.2 of [Eade, 2013]). Moreover, it is called the (left) jacobian of $\boldsymbol{SO}(2)$ ([Chirikjian, 2011], p.36).

- for $n = 3$:

  First, compute $\boldsymbol{\mathcal{R}}(\boldsymbol{\omega})$ using (B.13). Then, we have

$$\begin{aligned} \boldsymbol{\chi} &= \exp(\boldsymbol{\xi}^\wedge) = \begin{pmatrix} \boldsymbol{\mathcal{R}}(\boldsymbol{\omega}) & \mathbf{J}_\mathsf{L}\boldsymbol{\rho} \\ \mathbf{0}_{1\times 3} & 1 \end{pmatrix} \\ \mathbf{J}_\mathsf{L} &= \boldsymbol{I}_{3\times 3} + \Big(\frac{1-\cos(\theta)}{\theta^2}\Big)\boldsymbol{\omega}^\wedge + \Big(\frac{\theta-\sin(\theta)}{\theta^3}\Big)(\boldsymbol{\omega}^\wedge)^2 \\ \theta &= ||\boldsymbol{\omega}|| \end{aligned} \tag{B.24}$$

  where $\mathbf{J}_\mathsf{L}$ is the (left) jacobian of $\boldsymbol{SO}(3)$ (see [Barfoot and Furgale, 2014] or section §3.2 of [Eade, 2013]).

■ Logarithm map $\boldsymbol{SE}(n)$

- for $n = 2$:

  Find $\boldsymbol{\omega}^\wedge$ by solving $\log(\boldsymbol{\mathcal{R}})$ from (B.14). The log map is then expressed as

$$\log(\boldsymbol{\chi}) = \begin{pmatrix} \boldsymbol{\omega}^\wedge & \mathbf{J}_\mathsf{L}^{-1}\boldsymbol{x} \\ \mathbf{0}_{1\times 2} & 1 \end{pmatrix} = \boldsymbol{\xi}^\wedge \tag{B.25}$$

  where a closed-form formula for the inverse of $\mathbf{J}_\mathsf{L}$ is given by

$$\begin{aligned} \mathbf{J}_\mathsf{L}^{-1} &= \frac{1}{A^2+B^2}\begin{pmatrix} A & B \\ -B & A \end{pmatrix} \\ A = \frac{\sin(\theta)}{\theta}\,, &\quad B = \frac{1-\cos(\theta)}{\theta} \end{aligned} \tag{B.26}$$

- for $n = 3$:

  Find $\boldsymbol{\omega}^\wedge$ and θ by solving $\log(\boldsymbol{\mathcal{R}})$ from (B.15). The log map is then expressed as

$$\log(\boldsymbol{\chi}) = \begin{pmatrix} \boldsymbol{\omega}^\wedge & \mathbf{J}_\mathsf{L}^{-1}\boldsymbol{x} \\ \mathbf{0}_{1\times 3} & 1 \end{pmatrix} = \boldsymbol{\xi}^\wedge \tag{B.27}$$

  where a closed-form formula for the inverse of $\mathbf{J}_\mathsf{L}$ is given by

$$\begin{aligned} \mathbf{J}_\mathsf{L}^{-1} &= \boldsymbol{I}_{3\times 3} - \frac{1}{2}\boldsymbol{\omega}^\wedge + \frac{1}{\theta^2}\Big(1 - \frac{A}{2B}\Big)(\boldsymbol{\omega}^\wedge)^2 \\ A = \frac{\sin(\theta)}{\theta}\,, &\quad B = \frac{1-\cos(\theta)}{\theta} \end{aligned} \tag{B.28}$$

  Another closed-form formula is proposed in [Barfoot and Furgale, 2014].

■ **Adjoint $\boldsymbol{SE}(n)$**

If $\boldsymbol{\xi}$ contains the angular variable(s) in first and the translational variables in second as $\boldsymbol{\xi} = [\boldsymbol{\theta}, \boldsymbol{\rho}]^\top$, then the Adjoints are given by:

$$\begin{aligned} \forall \boldsymbol{\chi} \in \boldsymbol{SE}(2), \quad \mathrm{Ad}(\boldsymbol{\chi}) &= \left(\begin{array}{c|c} 1 & \mathbf{0}_{1\times 2} \\ \hline \mathbf{M}\boldsymbol{x} & \boldsymbol{\mathcal{R}} \end{array}\right), \quad \mathbf{M} = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \\ \forall \boldsymbol{\chi} \in \boldsymbol{SE}(3), \quad \mathrm{Ad}(\boldsymbol{\chi}) &= \left(\begin{array}{c|c} \boldsymbol{\mathcal{R}} & \mathbf{0}_{3\times 3} \\ \hline \boldsymbol{x}^\wedge \boldsymbol{\mathcal{R}} & \boldsymbol{\mathcal{R}} \end{array}\right) \end{aligned} \tag{B.29}$$

If one chooses the convention that $\boldsymbol{\xi}$ contains the translational variables in first and the angular variable(s) in second as $\boldsymbol{\xi} = [\boldsymbol{\rho}, \boldsymbol{\theta}]^\top$, then we have

$$\begin{aligned} \forall \boldsymbol{\chi} \in \boldsymbol{SE}(2), \quad \mathrm{Ad}(\boldsymbol{\chi}) &= \left(\begin{array}{c|c} \boldsymbol{\mathcal{R}} & \mathbf{M}\boldsymbol{x} \\ \hline \mathbf{0}_{1\times 2} & 1 \end{array}\right), \quad \mathbf{M} = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \\ \forall \boldsymbol{\chi} \in \boldsymbol{SE}(3), \quad \mathrm{Ad}(\boldsymbol{\chi}) &= \left(\begin{array}{c|c} \boldsymbol{\mathcal{R}} & \boldsymbol{x}^\wedge \boldsymbol{\mathcal{R}} \\ \hline \mathbf{0}_{3\times 3} & \boldsymbol{\mathcal{R}} \end{array}\right) \end{aligned} \tag{B.30}$$

■ **Small adjoint $\boldsymbol{SE}(n)$**

If $\boldsymbol{\xi} = [\boldsymbol{\theta}, \boldsymbol{\rho}]^\top$, then we have

$$\begin{aligned} \forall \boldsymbol{\xi}^\wedge \in \mathfrak{se}(2), \quad \mathrm{ad}(\boldsymbol{\xi}^\wedge) &= \left(\begin{array}{c|c} 0 & \mathbf{0}_{1\times 2} \\ \hline \mathbf{M}\boldsymbol{\rho} & \boldsymbol{\theta}^\wedge \end{array}\right), \quad \mathbf{M} = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \\ \forall \boldsymbol{\xi}^\wedge \in \mathfrak{se}(3), \quad \mathrm{ad}(\boldsymbol{\xi}^\wedge) &= \left(\begin{array}{c|c} \boldsymbol{\theta}^\wedge & \mathbf{0}_{3\times 3} \\ \hline \boldsymbol{\rho}^\wedge & \boldsymbol{\theta}^\wedge \end{array}\right) \end{aligned} \tag{B.31}$$

If one chooses $\boldsymbol{\xi} = [\boldsymbol{\rho}, \boldsymbol{\theta}]^\top$, then

$$\begin{aligned} \forall \boldsymbol{\xi}^\wedge \in \mathfrak{se}(2), \quad \mathrm{ad}(\boldsymbol{\xi}^\wedge) &= \left(\begin{array}{c|c} \boldsymbol{\theta}^\wedge & \mathbf{M}\boldsymbol{\rho} \\ \hline \mathbf{0}_{1\times 2} & 0 \end{array}\right), \quad \mathbf{M} = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \\ \forall \boldsymbol{\xi}^\wedge \in \mathfrak{se}(3), \quad \mathrm{ad}(\boldsymbol{\xi}^\wedge) &= \left(\begin{array}{c|c} \boldsymbol{\theta}^\wedge & \boldsymbol{\rho}^\wedge \\ \hline \mathbf{0}_{3\times 3} & \boldsymbol{\theta}^\wedge \end{array}\right) \end{aligned} \tag{B.32}$$

B.2.3 "Extended" Special Euclidean groups $\boldsymbol{SE}_k(n)$

Apart from $\boldsymbol{SE}(n)$, which describes only attitude and position, there exists an "extended" Special Euclidean group, often called the group of k-direct isometries and denoted $\boldsymbol{SE}_k(n)$, which is able to incorporate additional translational quantities alongside the position vector (for example, velocity vector and/or landmarks position). It was first introduced in [Bonnabel, 2012; Barrau and Bonnabel, 2015] in the context of SLAM.

For a given integer k, its dimension is

$$\dim\left(\boldsymbol{SE}_k(n)\right) = d = \begin{cases} 1 + 2k & \text{for } n = 2 \\ 3 + 3k & \text{for } n = 3 \end{cases}$$

$\boldsymbol{SE}_k(n)$ covers $\boldsymbol{SE}(n)$ with $k = 1$, and $\boldsymbol{SO}(n)$ with $k = 0$.

If we characterize a state containing the attitude $\boldsymbol{\mathcal{R}}$ of a rigid-body, its position $\boldsymbol{x}$, its velocity $\boldsymbol{v}$, as well as N_l landmarks, each of them denoted $\boldsymbol{l}^{(i)}$, $i = 1, \ldots, N_l$. Thus, a matrix $\boldsymbol{\chi} \in \boldsymbol{SE}_{2+N_l}(n)$ is given by

$$\boldsymbol{\chi} = \left(\begin{array}{c|cccc} \boldsymbol{\mathcal{R}} & \boldsymbol{v} & \boldsymbol{x} & \boldsymbol{l}^{(1)} & \ldots \quad \boldsymbol{l}^{(N_l)} \\ \hline \mathbf{0}_{(2+N_l)\times n} & \multicolumn{4}{c}{\mathbf{I}_{(2+N_l)\times(2+N_l)}} \end{array}\right), \quad \boldsymbol{\mathcal{R}} \in \boldsymbol{SO}(n);\ \boldsymbol{v}, \boldsymbol{x}, \boldsymbol{l}^{(i)} \in \mathbb{R}^n \tag{B.33}$$

and its inverse is given by

$$\boldsymbol{\chi} = \left(\begin{array}{c|cccc} \boldsymbol{\mathcal{R}}^\top & -\boldsymbol{\mathcal{R}}^\top(\boldsymbol{v} & \boldsymbol{x} & \boldsymbol{l}^{(1)} & \ldots \quad \boldsymbol{l}^{(N_l)}) \\ \hline \mathbf{0}_{(2+N_l)\times 3} & \multicolumn{4}{c}{\mathbf{I}_{(2+N_l)\times(2+N_l)}} \end{array}\right) \tag{B.34}$$

Hence, the Lie algebra $\mathfrak{se}_{2+N_l}(n)$ is a vector space of associated dimension d. An element $\boldsymbol{\xi}^\wedge \in \mathfrak{se}_{2+N_l}(n)$ is defined as follows

$$\boldsymbol{\xi}^\wedge = \left(\begin{array}{c|cccc} \boldsymbol{\omega}^\wedge & \boldsymbol{\rho}_{\boldsymbol{v}} & \boldsymbol{\rho}_{\boldsymbol{x}} & \boldsymbol{\rho}_{\boldsymbol{l}_1} & \ldots \quad \boldsymbol{\rho}_{\boldsymbol{l}_{N_l}} \\ \hline \mathbf{0}_{(2+N_l)\times 3} & \multicolumn{4}{c}{\mathbf{I}_{(2+N_l)\times(2+N_l)}} \end{array}\right) \tag{B.35}$$

Thus, the vector $\boldsymbol{\xi} \in \mathbb{R}^d$ is simply given by

$$\boldsymbol{\xi} = [\boldsymbol{\omega}^\top, \boldsymbol{\rho}_{\boldsymbol{v}}^\top, \boldsymbol{\rho}_{\boldsymbol{x}}^\top, \boldsymbol{\rho}_{\boldsymbol{l}^{(1)}}^\top, \ldots, \boldsymbol{\rho}_{\boldsymbol{l}^{(N_l)}}^\top]^\top$$

B.3 Additional details for the kinematical model on $\boldsymbol{SE}(2)$

For a matrix Lie group $\boldsymbol{G}$, the discrete kinematical model, describing the evolution of the pose $\boldsymbol{\chi}_{k-1}$ from known velocity inputs $\mathbf{u}_{k-1} \in \mathbb{R}^d$, is given by $\boldsymbol{f} : \boldsymbol{G} \times \mathbb{R}^d \mapsto \boldsymbol{G}$ as

$$\begin{aligned} \boldsymbol{\chi}_k &= \boldsymbol{f}(\boldsymbol{\chi}_{k-1}, \mathbf{u}_{k-1}) = \boldsymbol{\chi}_{k-1}\boldsymbol{U}_{k-1} \\ \text{where,}\ \ \boldsymbol{U}_{k-1} &= \exp(\mathbf{u}_{k-1}^\wedge \tau) \in \boldsymbol{G} \end{aligned} \tag{B.36}$$

and allows us to fully describe the motion of a rigid-body. In the case where $\boldsymbol{G} = \boldsymbol{SE}(2)$, it is possible to see how it is an equivalent formulation for a standard kinematical model defined in the state space $\mathbf{x} = [\theta, \boldsymbol{x}] = [\theta, x_1, x_2]^\top \in \mathbb{R}^3$. For that, we define the prediction model $\mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}) : \mathbb{R}^3 \times \mathbb{R}^3 \mapsto \mathbb{R}^3$ as follows

$$\mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}) = \begin{cases} \theta_k = \theta_{k-1} + u_{\theta,k-1}\tau \\ x_{1,k} = x_{1,k-1} + u_{\rho_1,k-1}\cos(\theta_{k-1})\tau - u_{\rho_2,k-1}\sin(\theta_{k-1})\tau \\ x_{2,k} = x_{2,k-1} + u_{\rho_1,k-1}\sin(\theta_{k-1})\tau + u_{\rho_2,k-1}\cos(\theta_{k-1})\tau \end{cases} \tag{B.37}$$

with $\mathbf{u}_k = [u_{\theta,k}, u_{\rho_1,k}, u_{\rho_2,k}]^\top$ being the velocity input vector expressed in the body-fixed frame. The kinematical model expressed above is similar to a nonholonomic car model (or unicycle model) [Lee et al., 2001] with a longitudinal velocity input u_{ρ_1}, except that additional terms (expressed via u_{ρ_2}) now appear and stand for transversal shift[1]. Thus,

[1] If the translational velocity u_{ρ_2} is taken null at any time instant, we recover the nonholonomic car model.

this model can describe the motion of holonomic ground robots such as three-wheeled omnidirectional robots, provided that we know the velocity input vector $\mathbf{u}_k$. In practice, the angular velocity $u_{\theta,k}$ can be derived from a gyrometer, while the translational velocities $u_{\rho_1,k}$ and $u_{\rho_2,k}$ can be derived from an odometer or an accelerometer.

By explicitly deriving the expression (B.36) using the exponential map of $\boldsymbol{SE}(2)$ (B.23), we get

$$\begin{aligned}
&\boldsymbol{f}(\boldsymbol{\chi}_{k-1}, \mathbf{u}_{k-1}) = \boldsymbol{\chi}_k \\
&= \begin{pmatrix} \boldsymbol{\mathcal{R}}(\theta_{k-1}) & \boldsymbol{x}_{k-1} \\ \mathbf{0}_{1\times 2} & 1 \end{pmatrix} \begin{pmatrix} \boldsymbol{\mathcal{R}}(u_{\theta,k-1}\tau) & \mathbf{J}_{\mathsf{L}} \begin{bmatrix} u_{\rho_1,k-1}\tau \\ u_{\rho_2,k-1}\tau \end{bmatrix} \\ \mathbf{0}_{1\times 2} & 1 \end{pmatrix} \\
&= \begin{pmatrix} \cos(\theta_{k-1}) & -\sin(\theta_{k-1}) & x_{1,k-1} \\ \sin(\theta_{k-1}) & \cos(\theta_{k-1}) & x_{2,k-1} \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \cos(u_{\theta,k-1}\tau) & -\sin(u_{\theta,k-1}\tau) & \mathbf{J}_{\mathsf{L}} \begin{bmatrix} u_{\rho_1,k-1}\tau \\ u_{\rho_2,k-1}\tau \end{bmatrix} \\ \sin(u_{\theta,k-1}\tau) & \cos(u_{\theta,k-1}\tau) & \\ 0 & 0 & 1 \end{pmatrix}
\end{aligned} \tag{B.38}$$

with,

$$\mathbf{J}_{\mathsf{L}} = \frac{1}{u_{\theta,k-1}\tau} \begin{pmatrix} \sin(u_{\theta,k-1}\tau) & -1+\cos(u_{\theta,k-1}\tau) \\ 1-\cos(u_{\theta,k-1}\tau) & \sin(u_{\theta,k-1}\tau) \end{pmatrix} \tag{B.39}$$

Hence, the predicted state matrix $\boldsymbol{\chi}_k$ becomes

$$\boldsymbol{\chi}_k = \begin{pmatrix} \cos(\theta_{k-1}+u_{\theta,k-1}\tau) & -\sin(\theta_{k-1}+u_{\theta,k-1}\tau) & x_{1,k-1}+v_\ell(\mathbf{u}_{k-1})\cos(\theta_{k-1}) - v_t(\mathbf{u}_{k-1})\sin(\theta_{k-1}) \\ \sin(\theta_{k-1}+u_{\theta,k-1}\tau) & \cos(\theta_{k-1}+u_{\theta,k-1}\tau) & x_{2,k-1}+v_\ell(\mathbf{u}_{k-1})\sin(\theta_{k-1}) + v_t(\mathbf{u}_{k-1})\cos(\theta_{k-1}) \\ 0 & 0 & 1 \end{pmatrix} \tag{B.40}$$

with,

$$\begin{aligned}
v_\ell(\mathbf{u}_{k-1}) &= \frac{1}{u_{\theta,k-1}\tau}\Big(u_{\rho_1,k-1}\tau\sin(u_{\theta,k-1}\tau) - u_{\rho_2,k-1}\tau + u_{\rho_2,k-1}\tau\cos(u_{\theta,k-1}\tau)\Big) \\
v_t(\mathbf{u}_{k-1}) &= \frac{1}{u_{\theta,k-1}\tau}\Big(u_{\rho_1,k-1}\tau - u_{\rho_1,k-1}\tau\cos(u_{\theta,k-1}\tau) + u_{\rho_2,k-1}\tau\sin(u_{\theta,k-1}\tau)\Big)
\end{aligned} \tag{B.41}$$

where v_ℓ denotes the longitudinal shift and v_t denotes the transerval shift. Hence, we can see a similar formulation as $\mathbf{f}$. Indeed, by extracting the state variables from $\boldsymbol{\chi}_k$, we have

$$\begin{aligned}
\theta_k &= \theta_{k-1} + u_{\theta,k-1}\tau \\
x_{1,k} &= x_{1,k-1} + v_\ell(\mathbf{u}_{k-1})\cos(\theta_{k-1}) - v_t(\mathbf{u}_{k-1})\sin(\theta_{k-1}) \\
x_{2,k} &= x_{2,k-1} + v_\ell(\mathbf{u}_{k-1})\sin(\theta_{k-1}) + v_t(\mathbf{u}_{k-1})\cos(\theta_{k-1})
\end{aligned} \tag{B.42}$$

which coincides with $\mathbf{f}$ if the quantity $u_{\theta,k-1}\tau$ is assumed small, since in this case, we have

$$\begin{aligned}
v_\ell(\mathbf{u}_{k-1}) &\approx u_{\rho_1,k-1}\tau \\
v_t(\mathbf{u}_{k-1}) &\approx u_{\rho_2,k-1}\tau
\end{aligned} \tag{B.43}$$

In practice, the assumption that $u_{\theta,k-1}\tau$ is small is often encountered in practice, since the propagation is discretized at a quite large frequency.

This example on $\boldsymbol{SE}(2)$ shows that a $\boldsymbol{G}$-valued kinematical model of the form $\boldsymbol{\chi}_{k-1}\boldsymbol{U}_{k-1}$ can totally replace a $\mathbb{R}^d$-valued kinematical model of the form $\mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1})$ in a compact form.

B.4 Evaluation metrics

We present the main metrics used to evaluate the performance of the proposed estimators. At each time instant along the trajectory, the estimation error can be quantified either through group errors (Definition 3.5) or through log-Euclidean errors (Definition 3.6). To measure the discrepancy between an estimated state $\hat{\boldsymbol{\chi}}_k$ and a reference state $\boldsymbol{\chi}_k$, we adopt a left-invariant error formulation,

$$\tilde{\boldsymbol{\chi}}_k = \boldsymbol{\chi}_k^{-1}\hat{\boldsymbol{\chi}}_k \quad \Rightarrow \quad \tilde{\boldsymbol{\xi}}_k = \log(\tilde{\boldsymbol{\chi}}_k)^\vee$$

As a stochastic estimator, a Kalman filter is expected to be unbiased and exhibit statistical consistency. Accordingly, its performance must be evaluated using metrics that reflect both estimation accuracy and uncertainty representation. Commonly used criteria for stochastic estimators include the Root Mean Square Error (RMSE), the Normalized Estimation Error Squared (NEES), and the Cramér–Rao Lower Bound (CRLB). In the following, we introduce these metrics and adapt their definitions to the Lie group setting.

Bias of the estimator

Using the log-Euclidean error definition, it is possible to evaluate the bias of an estimator along a trajectory by computing the average error over N_s Monte-Carlo (performed according to the procedure detailed in subsection §6.1.3),that is $\forall k = 1, \dots, k_{end}$,

$$\begin{aligned}\mathbb{E}[\tilde{\boldsymbol{\xi}}_k] &= \frac{1}{N_s}\sum_{i=1}^{N_s}\tilde{\boldsymbol{\xi}}_k^{(i)} = \frac{1}{N_s}\sum_{i=1}^{N_s}\log(\tilde{\boldsymbol{\chi}}_k^{(i)})^\vee \\ &= \frac{1}{N_s}\sum_{i=1}^{N_s}\log\left((\boldsymbol{\chi}_k^{(i)})^{-1}\hat{\boldsymbol{\chi}}_k^{(i)}\right)^\vee\end{aligned} \tag{B.44}$$

A correct stochastic observer is expected to be unbiased, i.e. $\mathbb{E}(\tilde{\boldsymbol{\xi}}_k) \to \mathbf{0}$ as k increases.

Root Mean Square Error (RMSE)

Let $||\cdot||$ denotes the L^2-norm. Following the definition of left-invariant error, the Root Mean Square (log)-Error (RMSE) of an estimate $\hat{\boldsymbol{\chi}}_k$, with respect to a reference value $\boldsymbol{\chi}_k$, is defined as

$$\mathrm{RMSE}(\hat{\boldsymbol{\chi}}_k, \boldsymbol{\chi}_k) = \sqrt{\mathbb{E}\left[||\log(\boldsymbol{\chi}_k^{-1}\hat{\boldsymbol{\chi}}_k)^\vee||^2\right]} \tag{B.45}$$

The RMSE provides information on the accuracy of the estimator. However, the RMSE is highly sensitive to outliers.

Consistency

In general, *consistency* of an estimator of a random variable $\mathbf{x}$, refers to the property that, as the number of data points increases indefinitely, the resulting sequence of estimates converges in probability to that of $\mathbf{x}$. In this manuscript, the term *consistency*

refers to the capacity to provide an accurate covariance estimate $\mathbf{P}_k$ compared to the covariance of the estimation error $\tilde{\mathbf{P}}_k$. These two covariances are directly comparable if the estimator is unbiased. Indeed, the covariance of the estimation error is defined as

$$\tilde{\mathbf{P}}_k := \mathbb{E}\Big[\big(\tilde{\boldsymbol{\xi}}_k - \mathbb{E}(\tilde{\boldsymbol{\xi}}_k)\big)\big(\tilde{\boldsymbol{\xi}}_k - \mathbb{E}(\tilde{\boldsymbol{\xi}}_k)\big)^\top\Big]. \tag{B.46}$$

If the estimator is unbiased, then $\mathbb{E}(\tilde{\boldsymbol{\xi}}_k) = \mathbf{0}$, and the covariance of the estimation error becomes

$$\tilde{\mathbf{P}}_k = \mathbb{E}[\tilde{\boldsymbol{\xi}}_k\,\tilde{\boldsymbol{\xi}}_k^\top] = \mathbb{E}\Big[\log(\boldsymbol{\chi}_k^{-1}\hat{\boldsymbol{\chi}}_k)^\vee \log(\boldsymbol{\chi}_k^{-1}\hat{\boldsymbol{\chi}}_k)^{\vee\top}\Big] := \mathbf{P}_k \tag{B.47}$$

From N_s Monte-Carlo simulations, and $\forall k = 1, \ldots, k_{end}$, it is possible to compute the empirical covariance of the estimation error $\tilde{\mathbf{P}}_k$ as

$$\begin{aligned}\tilde{\mathbf{P}}_k &= \frac{1}{N_s}\sum_{i=1}^{N_s}\Big(\tilde{\boldsymbol{\xi}}_k^{(i)} - \mathbb{E}[\tilde{\boldsymbol{\xi}}_k]\Big)\Big(\tilde{\boldsymbol{\xi}}_k^{(i)} - \mathbb{E}[\tilde{\boldsymbol{\xi}}_k]\Big)^\top \\ &= \frac{1}{N_s}\sum_{i=1}^{N_s}\left[\log\Big((\boldsymbol{\chi}_k^{(i)})^{-1}\hat{\boldsymbol{\chi}}_k^{(i)}\Big)^\vee - \frac{1}{N_s}\sum_{i=1}^{N_s}\log\Big((\boldsymbol{\chi}_k^{(i)})^{-1}\hat{\boldsymbol{\chi}}_k^{(i)}\Big)^\vee\right]\left[\ldots\right]^\top\end{aligned} \tag{B.48}$$

which is the real precision of the estimator. Then, we compare it with the covariance $\mathbf{P}_k$ given by the estimator, which is the "confidence level" it has in its own estimate $\hat{\boldsymbol{\chi}}_k$. If the diagonal components of both quantities match, then the estimator is said to be consistent, and evaluates well its estimate. If the diagonal components of $\mathbf{P}_k$ are superior to those of $\tilde{\mathbf{P}}_k$, then the estimator is *conservative*: it "overestimates the uncertainty", or equivalently, "underestimate its capabilities" (meaning that it could perform better). If the diagonal components of $\mathbf{P}_k$ are inferior to those of $\tilde{\mathbf{P}}_k$, then the estimator is said *inconsistent*. In general, it is more desirable to be conservative than inconsistent. We can qualitatively check the closeness between these two covariances by plotting their evolution as a function of k, or quantitatively assess the closeness using the next metric.

Normalized Estimation Error Squared (NEES)

The closeness between the $d \times d$ matrices $\mathbf{P}_{k|k}$ and $\tilde{\mathbf{P}}_k$ can be quantitatively checked with a hypothesis test using the Normalized Estimation Error Squared (NEES) metric, which is also the squared *Mahalanobis distance*, and defined as

$$\begin{aligned}\text{NEES}(\mathbf{P}_k, \tilde{\mathbf{P}}_k) &= \tilde{\boldsymbol{\xi}}_k^\top \mathbf{P}_k^{-1}\,\tilde{\boldsymbol{\xi}}_k \\ &= \log(\boldsymbol{\chi}_k^{-1}\hat{\boldsymbol{\chi}}_k)^{\vee\top}\mathbf{P}_k^{-1}\,\log(\boldsymbol{\chi}_k^{-1}\hat{\boldsymbol{\chi}}_k)^\vee\end{aligned} \tag{B.49}$$

The hypothesis test relies on the assumption that the quantity $\text{NEES}(\mathbf{P}_k, \tilde{\mathbf{P}}_k)$ should be χ^2-distributed[2] with d degrees of freedom. Hence, the mean of the NEES should be

$$\mathbb{E}[\text{NEES}] = d \tag{B.50}$$

This can be used to check whether a state estimator provides consistent results or not. The following hypothesis

$$H_0 : \text{the estimation error } \tilde{\boldsymbol{\xi}}_k \text{ is consistent with the filter's covariance } \mathbf{P}_{k|k}$$

[2] [Wikipedia] χ^2 distribution

is accepted if $b_1 \leq \text{NEES} \leq b_2$. The acceptance interval is chosen such that the probability $\mathcal{P}$ that H_0 is accepted is $1-\alpha$, i.e.

$$\mathcal{P}(\text{NEES} \in [b_1, b_2] | H_0) = 1 - \alpha \tag{B.51}$$

where α is a chosen acceptance rate. The confidence interval $[b_1, b_2]$ can be calculated with the inverse cumulative distribution function, denoted CDF^{-1}, of the χ^2 distribution

$$\begin{aligned} b_2 &= CDF^{-1}(1 - \frac{\alpha}{2}, d) \quad \text{(upper bound)} \\ b_1 &= CDF^{-1}(\frac{\alpha}{2}, d) \quad \text{(lower bound)} \end{aligned} \tag{B.52}$$

The interpretation of the NEES is explained thanks to Figure B.1.

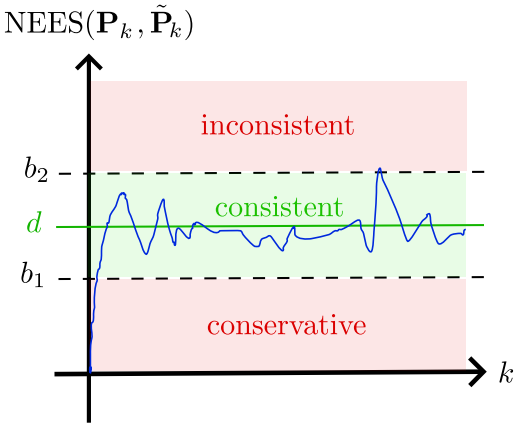


Figure B.1: The NEES is computed along the trajectory (for all k), and quantifies the consistency of the estimated covariance $\mathbf{P}_k$ with respect to the squared log-Euclidean estimation error (indicated by the covariance of the estimation error $\tilde{\mathbf{P}}_k$). If $b_1 < \text{NEES}(\mathbf{P}_k, \tilde{\mathbf{P}}_k) < b_2$, then the filter is consistent: its estimation of the error correctly matches the true error. If $\text{NEES}(\mathbf{P}_k, \tilde{\mathbf{P}}_k) < b_1$, the filter is conservative: the error is estimated larger than it truely is. If $\text{NEES}(\mathbf{P}_k, \tilde{\mathbf{P}}_k) > b_2$, the filter is inconsistent: the filter estimates the error to be lower than it is in reality.

To increase statistical significance, additional N_s independent values $\text{NEES}(\mathbf{P}_k^{(i)}, \tilde{\mathbf{P}}_k^{(i)})$ $(i = 1, \ldots, N_s)$ can be generated and an Average NEES can be computed

$$\text{ANEES}(\mathbf{P}_k, \tilde{\mathbf{P}}_k) = \frac{1}{N_s} \sum_{i=1}^{N_s} \text{NEES}(\mathbf{P}_k^{(i)}, \tilde{\mathbf{P}}_k^{(i)}) \tag{B.53}$$

The $\text{ANEES}(\mathbf{P}_k, \tilde{\mathbf{P}}_k)$ has the property that the expectation converges towards d and that $N_s \times \text{ANEES}(\mathbf{P}_k^{(i)}, \tilde{\mathbf{P}}_k^{(i)})$ is χ^2-distributed with $N_s \times d$ degrees of freedom. The confidence interval must therefore be normalized by N_s, i.e.

$$\begin{aligned} b_2 &= \frac{1}{N_s} CDF^{-1}(1 - \frac{\alpha}{2}, N_s \times d) \quad \text{(upper bound)} \\ b_1 &= \frac{1}{N_s} CDF^{-1}(\frac{\alpha}{2}, N_s \times d) \quad \text{(lower bound)} \end{aligned} \tag{B.54}$$

so that the interpretation of the ANEES remains the same as the NEES.

von Mises Statistic (vMS)

[Horwood and Poore, 2014] introduced the "Mahalanobis von Mises statistic" as a mechanism for testing if some realization $(\boldsymbol{x}, \theta)$ from a GvM distribution is statistically significant. The Mahalanobis von Mises statistic is analogous to the Mahalanobis distance for a multivariate Gaussian random vector. Hence, it is analoguous to the NEES. For a random variable $(\boldsymbol{z}, \phi)$ following a canonical GvM distribution $\mathcal{GVM}(\boldsymbol{z}, \phi; \mathbf{0}, \mathbf{I}, \kappa)$, the Mahalanobis von Mises statistic is given by $\text{MvMS} = \boldsymbol{z}^\top \boldsymbol{z} + 4\kappa \sin^2(\frac{1}{2}\phi)$, where $\boldsymbol{z}^\top \boldsymbol{z}$ is precisely the Mahalnobis distance for a centered-normalized random vector. For an angular variable, we are only interested in the von Mises part. In particular, we can use it to assess if an estimated realization is statistically significant with a von Mises distribution. Hence, for an estimated angular variable ω_k of concentration parameter κ_k at time instant k, we define the von Mises statistic (vMS) as follows

$$\text{vMS}(\kappa_k, \tilde{\theta}_k) = 4\kappa_k \sin^2(\frac{1}{2}\tilde{\theta}_k) \tag{B.55}$$

where $\tilde{\theta}_k$ is the real precision of the estimator. Thus, κ_k is analoguous to $\mathbf{P}_k$, and $\tilde{\theta}_k$ is analoguous to $\tilde{\mathbf{P}}_k$ described previously. If the attitude is described in $\boldsymbol{SO}(3)$ such that we have three angular variables $\omega_{i,k}$ $(i = 1, \ldots, 3)$, then we compute $\tilde{\theta}_{i,k}$ from

$$\begin{bmatrix} \tilde{\theta}_{1,k} \\ \tilde{\theta}_{2,k} \\ \tilde{\theta}_{3,k} \end{bmatrix} = \log(\boldsymbol{\mathcal{R}}_k^{-1} \hat{\boldsymbol{\mathcal{R}}}_k)^\vee \tag{B.56}$$

By extension, we also define an average von Mises statistic (from N_s Monte-Carlo) as

$$\text{AvMS}(\kappa_k, \tilde{\theta}_k) = \frac{1}{N_s} \sum_{i=1}^{N_s} \left(4\kappa_k^{(i)} \sin^2(\frac{1}{2}\tilde{\theta}_k^{(i)}) \right) \tag{B.57}$$

The von Mises statistic has the same interpretation as the NEES. In particular, an estimated realization ω_k (with concentration κ_k) is statistically significant with a von Mises distribution if its vMS converges to a χ^2-distribution of 1 degree of freedom.

Cramér-Rao Lower Bounds (CRLB)

Cramér-Rao Lower Bounds (CRLB) are traditionaly used to assess the *efficiency* of an unbiased estimator. They compute the minimal variance possible. An unbiased estimator that reaches the lower bounds is said to be efficient. Such a solution achieves the lowest possible mean squared error among all unbiased estimators, and is, therefore, the minimum variance unbiased estimator.

In Chahbazian et al. [2024], a method is provided to recursively compute the Cramér-Rao Lower Bounds on Lie groups, denoted J_k $(k = 0, \ldots)$, given a system of the form

$$\begin{aligned} \boldsymbol{\chi}_k &= \boldsymbol{f}(\boldsymbol{\chi}_{k-1}, \mathbf{u}_{k-1}) \exp(\mathbf{w}_{k-1}^\wedge) \in \boldsymbol{G} \\ \boldsymbol{Y}_k &= \boldsymbol{h}(\boldsymbol{\chi}_k) \exp(\mathbf{v}_k^\wedge) \in \boldsymbol{G} \end{aligned} \tag{B.58}$$

with respective covariance $\mathbf{Q}_k$ and $\mathbf{R}_k$ on their noise $\mathbf{w}_k$ and $\mathbf{v}_k$, respectively. Basically, it consists of the same recursive equations as the usual CRLB in Euclidean spaces:

$$\begin{aligned} k = 0: &\quad J_0 = \mathbf{P}_{0|0}^{-1} \\ k > 0: &\quad J_k = D_{k-1}^{12} - D_{k-1}^{21}(J_{k-1} + D_{k-1}^{11})D_{k-1}^{12} \end{aligned} \tag{B.59}$$

where the different matrices D_k^{ij} are given by

$$\begin{aligned} D_{k-1}^{11} &= \mathbf{F}_{k-1}^{\top}\mathbf{Q}_{k-1}^{-1}\mathbf{F}_{k-1} \\ D_{k-1}^{12} &= \mathbf{F}_{k-1}^{\top}\mathbf{Q}_{k-1}^{-1} \\ D_{k-1}^{21} &= (D_{k-1}^{12})^{\top} \\ D_{k-1}^{22} &= \mathbf{Q}_{k-1}^{-1} + \mathbf{H}_k^{\top}\mathbf{R}_k^{-1}\mathbf{H}_k \end{aligned} \tag{B.60}$$

but with $\mathbf{F}_{k-1}$ and $\mathbf{H}_k$ defined as

$$\mathbf{F}_{k-1} = \left.\frac{\partial \log\left(\boldsymbol{f}(\boldsymbol{\chi}_{k-1}\exp(\boldsymbol{\xi}^{\wedge}), \mathbf{u}_{k-1})^{-1}\boldsymbol{\chi}_k\right)^{\vee}}{\partial \boldsymbol{\xi}}\right|_{\boldsymbol{\xi}\to\mathbf{0}} \tag{B.61}$$

$$\mathbf{H}_k = \left.\frac{\partial \log\left(\boldsymbol{h}\left(\boldsymbol{\chi}_k \exp(\boldsymbol{\xi}^{\wedge})\right)^{-1}\boldsymbol{Y}_k\right)^{\vee}}{\partial \boldsymbol{\xi}}\right|_{\boldsymbol{\xi}\to\mathbf{0}} \tag{B.62}$$

In this thesis, since we consider a system model of the form

$$\begin{aligned} \boldsymbol{\chi}_k &= \boldsymbol{\chi}_{k-1}\exp(\mathbf{u}_{k-1}^{\wedge})\exp(\mathbf{w}_{k-1}^{\wedge}) \in \boldsymbol{G} \\ \mathbf{y}_k &= \mathbf{h}(\boldsymbol{\chi}_k, \mathbf{v}_k) \in \mathbb{R}^p \end{aligned} \tag{B.63}$$

which is a particular case of (B.58), the jacobian $\mathbf{F}_k$ writes

$$\mathbf{F}_k = -\boldsymbol{\mathcal{J}}_{\mathsf{L}}\left(\log(\boldsymbol{\chi}_{k-1}^{-1}\boldsymbol{\chi}_k)\right) \tag{B.64}$$

where $\boldsymbol{\mathcal{J}}_{\mathsf{L}}(\cdot)$ denotes the left jacobian operator, and the jacobian $\mathbf{H}_k$ writes

$$\mathbf{H}_k = \left.\frac{\partial\left(\mathbf{y}_k - \mathbf{h}\left(\boldsymbol{\chi}_k\exp(\boldsymbol{\xi}^{\wedge}), \mathbf{0}\right)\right)}{\partial \boldsymbol{\xi}}\right|_{\boldsymbol{\xi}\to\mathbf{0}} \tag{B.65}$$

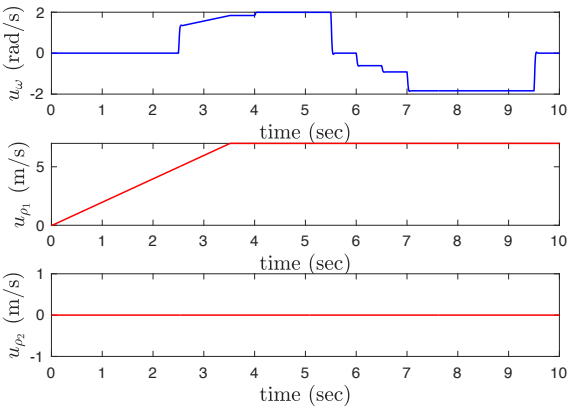
B.5 Additional results on the OQ-UKF-LG

We provide additional results on the optimal quantization-based UKF-LG.

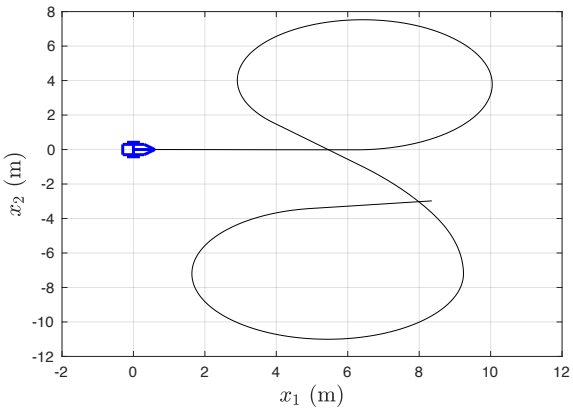
B.5.1 Evaluation for a simulated 2D trajectory

We evaluate the performance of the OQ-UKF-LG for a trajectory evolving in $\boldsymbol{SE}(2)$, following the procedures described in Section §6.1. The nominal trajectory is generated by the velocity inputs $\mathbf{u}_k$, perturbed by white Gaussian noise $\mathbf{w}_k \sim \mathcal{N}(\mathbf{w}_k; \mathbf{0}, \mathbf{Q}_k)$. We also restrict ourselves to the case of constant model noise covariance, i.e. $\mathbf{Q}_k = \mathbf{Q}$, $\forall k$. The nominal trajectory (Figure B.2b) is generated by the sequence of inputs $\mathbf{u}_k$ (Figure B.2a). It corresponds to 10 seconds of simulated trajectory, with $f_\tau = 100$ Hz.

Regarding the measurement acquisition for the Kalman correction step, the default setting is given by $f_{obs} = 10$ Hz, and $\mathcal{N}(\mathbf{v}_k; \mathbf{0}, \mathbf{R}_k)$, with $\mathbf{R}_k = \mathbf{R}$ $\forall k$.



(a) Sequence of nominal inputs $\mathbf{u}$ used to generate the nominal trajectory.



(b) Nominal trajectory in the two-dimensional space. The rigid-body is initialized near the position $[0, 0]^\top$.

Figure B.2: Nominal inputs and trajectory for the OQ-UKF-LG on $\boldsymbol{SE}(2)$.

■ Results for a single run

Let us compare the OQ-(Left)-UKF-LG and (Left)-UKF-LG for a single run. For a fair comparison, both filters process the same sequence of random variables $\{\mathbf{v}_k\}_{k=1,\dots,k_{end}}$, such that they use the same measurement $\mathbf{y}_k$ at a given time instant.

Hyper-parameters:

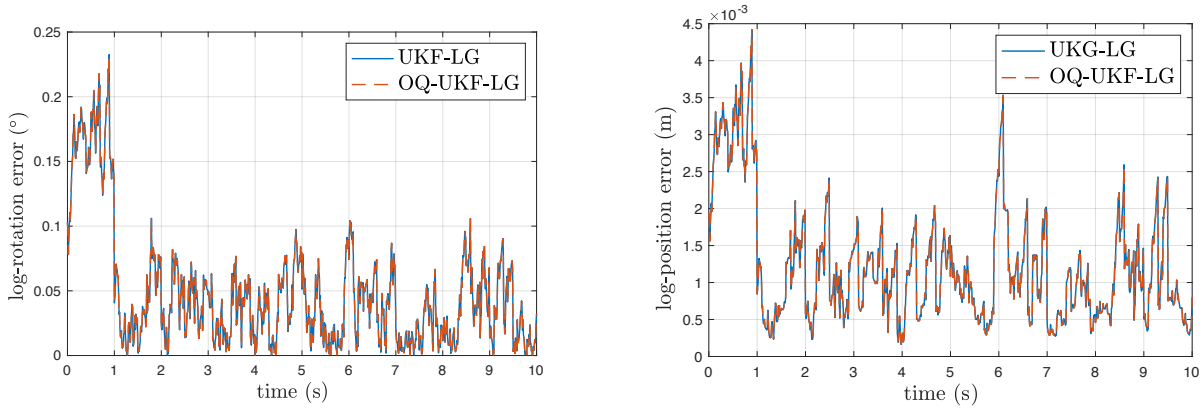
Measurement model: $\mathbf{h}_{\text{gps}}$
Measurement frequency: $f_{obs} = 10$ Hz
Initial covariance: $\mathbf{P}_{0|0} = \text{diag}\left(\left[(\frac{0.1\pi}{180})^2, (0.001)^2, (0.001)^2\right]\right)$
Input noise: $\mathbf{Q} = \text{diag}\left(\left[(\frac{\pi}{180})^2, (0.001)^2, (0.001)^2\right]\right)$
Input uncertainty parameter: $\hat{\mathbf{Q}} = \mathbf{Q}$
Output noise: $\mathbf{R} = \text{diag}\left([0.001^2, 0.001^2]\right)$
Output uncertainty parameter: $\hat{\mathbf{R}} = \mathbf{R}$
CLVQ parameters: $n_{max} = 100$, $\gamma_{\ell,n} = \frac{\frac{1}{10}\sqrt{(\mathbf{P}_{aug})_{\ell\ell}}}{n}$, $n = 1, \dots, n_{max}$, $\forall \ell = 1, \dots, d'$ (for prediction UT and correction UT)

For this set of hyperparameters, the results are plotted in Figure B.3. As we can see, they do not differ for small uncertainties, and whatever the time index k. For this single simulation, we compute the RMSE (over the entire trajectory) to provide a global evaluation of the performance (Table B.2). It seems that for low uncertainties (in the initial

Filter	RMSE in rotation (°)	RMSE in position (m)
UKF-LG	0.064911	0.0010474
OQ-UKF-LG	0.064911	0.0010474

Table B.2: Root Mean (over the entire trajectory) Square Error for the results of Figure B.3.

covariance, in the model, and measurement noise), the OQ-UKF-LG is not of particular relevance. The idea is now to evaluate the sensitivity of the proposed OQ-UKF-LG in different cases, therefore testing its limits and drawbacks, as well as identifying situations



(a) L^2-norm of the log-rotation error

(b) L^2-norm of the log-position error

Figure B.3: L^2-norm of the log-Euclidean errors (in rotation and in position) as a function of time

where using a UKF-LG enhanced by optimal quantization could be relevant.

Sensitivity to an overestimation of Q:

For this set of experiments, we have slightly increased the standard deviations of the noise: $\sqrt{Q_{\rho_2}} = \sqrt{Q_{\rho_3}} = 0.01$ and $\sqrt{R_x} = \sqrt{R_y} = 0.01$. All other hyper-parameters being equal, we propose different values for the uncertainty parameter $\hat{\mathbf{Q}}$, superior to the true covariance $\mathbf{Q}$, and keeping $\hat{\mathbf{R}} = \mathbf{R}$. This implies that the Kalman filter is parameterized such that it is conservative. The global performance, in terms of RMSE, is provided in Table B.3.

Filter	$\hat{\mathbf{Q}}$	RMSE in rotation (°)	RMSE in position (m)
UKF-LG	$10^2 \times \mathbf{Q}$	0.22351	0.0081566
OQ-UKF-LG	$10^2 \times \mathbf{Q}$	0.22319 (-0.15%)	0.008153 (-0.05%)
UKF-LG	$20^2 \times \mathbf{Q}$	0.33149	0.0097266
OQ-UKF-LG	$20^2 \times \mathbf{Q}$	0.33025 (-0.37%)	0.009718 (-0.09%)
UKF-LG	$30^2 \times \mathbf{Q}$	0.4158	0.010876
OQ-UKF-LG	$30^2 \times \mathbf{Q}$	0.41358 (-0.53%)	0.010859 (-0.16%)
UKF-LG	$40^2 \times \mathbf{Q}$	0.48039	0.011742
OQ-UKF-LG	$40^2 \times \mathbf{Q}$	0.47687 (-0.73%)	0.011724 (-0.15%)
UKF-LG	$50^2 \times \mathbf{Q}$	0.52898	0.012393
OQ-UKF-LG	$50^2 \times \mathbf{Q}$	0.52439 (-0.87%)	0.012371 (-0.19%)

Table B.3: RMSE (over the entire trajectory), for different overestimations of the true covariance $\mathbf{Q}$. The value in percentage corresponds to the relative difference between the RMSE of the OQ-UKF-LG and the RMSE of the UKF-LG, taken as the reference.

Regarding these results, the optimal quantization seems to provide a small improvement on the RMSE. Despite this fact, the reduction in error (permitted by the proposed method) becomes all the more significant as the size of $\hat{\mathbf{Q}}$ increases. An interesting feature is that we see more differences with respect to the RMSE in rotation than with respect to the RMSE in position, as pointed out by the relative difference in RMSE between the

two filters (cf. Table B.3), indicating that the OQ-UKF-LG could be relevant in scenarios where no cap measurement is available. Indeed, we considered the case $\mathbf{h}_{\text{gps}}$, such that only cartesian positions are observed. This implies that, in the correction step of the UKF-LG, the posterior variance of the angular variable in $\mathbf{P}_{k|k}$ is less prone to decrease compared to the posterior variance of the translational variables. Hence, in the standard UKF-LG, the variance of the angular variable tends to overestimate the true uncertainty of the cap. Further investigations allow us to show that the CLVQ algorithm tends to make the sigma-points closer to the central point. As a consequence, the CLVQ helps the Kalman filter in such a way that it reduces the overestimation of the uncertainty of non observable state variables. Therefore, the OQ-UKF-LG may be relevant to reduce conservatism. However, this may also imply that using a CLVQ could not provide any help for an already inconsistent parameterization, as will be seen with the next experiment.

Sensitivity to an underestimation of Q

We investigate what happens if the noise parameter $\hat{\mathbf{Q}}$ is underestimated with respect to $\mathbf{Q}$. Taking $\hat{\mathbf{Q}} < \mathbf{Q}$, while keeping $\hat{\mathbf{R}} = \mathbf{R}$ will imply the Kalman filter to be inconsistent. We perform two tests, by setting $\hat{\mathbf{Q}} = \text{diag}\left((\frac{\pi}{180})^2, (0.01)^2, (0.01)^2\right)$ and $\mathbf{Q} = 50^2 \times \hat{\mathbf{Q}}$, or $\mathbf{Q} = 40^2 \times \hat{\mathbf{Q}}$. The results are given in Table B.4.

Filter	$\hat{\mathbf{Q}}$	RMSE in rotation (°)	RMSE in position (m)
UKF-LG	$\frac{1}{40^2} \times \mathbf{Q}$	3.4128	0.12453
OQ-UKF-LG	$\frac{1}{40^2} \times \mathbf{Q}$	3.413 (+0.006%)	0.12453 (±0%)
UKF-LG	$\frac{1}{50^2} \times \mathbf{Q}$	4.2565	0.15544
OQ-UKF-LG	$\frac{1}{50^2} \times \mathbf{Q}$	4.2572 (+0.016%)	0.15546 (+0.013%)

Table B.4: RMSE (over the entire trajectory) for an underestimation of the true covariance $\mathbf{Q}$. The value in percentage corresponds to the relative difference between the RMSE of the OQ-UKF-LG and the RMSE of the UKF-LG, taken as the reference.

The fact that sigma-points tend to tighten after readjustment from the CLVQ may be the reason why it does not provide any help when the covariance is already underestimated. Indeed, for an underestimated prior covariance, the right decision would be to spread the sigma-points instead of rendering them closer than they already are. This is something the CLVQ does not actually perform. Fortunately, the sensitivity to an underestimation of $\mathbf{Q}$ is not as important as the sensitivity to an overestimation of $\mathbf{Q}$, as underlined by the magnitude of the relative differences in Table B.4 compared to those in Table B.3.

Relevancy for a conservative scenario

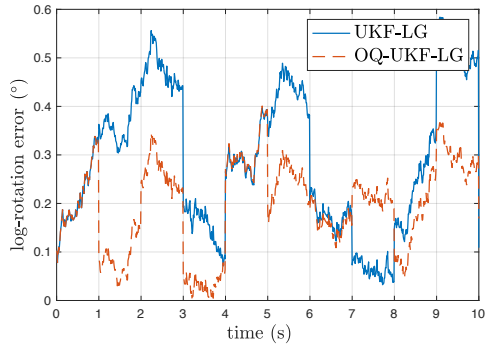
To better see a difference between the two filters when both are conservatives, we modify the hyper-parameters setting while keeping $\mathbf{h}_{\text{gps}}$.

Hyper-parameters:

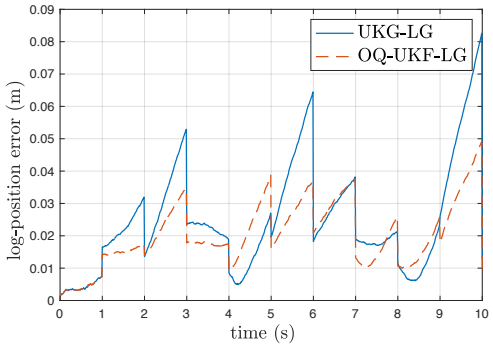
Measurement frequency: $f_{obs} = 1$ Hz
Input noise: $\mathbf{Q} = \text{diag}\left(\left[(\frac{\pi}{180})^2, (0.01)^2, (0.01)^2\right]\right)$

Input uncertainty parameter: $\hat{\mathbf{Q}} = 50^2 \times \mathbf{Q}$
Output noise: $\mathbf{R} = \text{diag}\left([0.01^2, 0.01^2]\right)$
Output uncertainty parameter: $\hat{\mathbf{R}} = \mathbf{R}$

This allows us to overestimate the uncertainty and limit the occurrence to which the covariance is updated by a measurement acquisition. We also slightly modify the CLVQ parameters: only the dimension related to the angle is adjusted, and the learning rate is set to $\gamma_{1,n} = \frac{\sqrt{P_\theta}}{n}$ for a bigger step. A single realization of the simulation is performed, and the associated results are given in Figure B.4. As shown in these figures, the UKF-LG



(a) L^2-norm of the log-rotation error. RMSE (UKF-LG) = 0.32725°. RMSE (OQ-UKF-LG) = 0.21935°.



(b) L^2-norm of the log-position error. RMSE (UKF-LG) = 0.029625 m. RMSE (OQ-UKF-LG) = 0.022542 m.

Figure B.4: L^2-norm of the log-Euclidean errors (in rotation and in position) as a function of time. $f_{obs} = 1$ Hz, $\hat{\mathbf{Q}} = 50^2 \times \mathbf{Q}$, $\hat{\mathbf{R}} = \mathbf{R}$. CLVQ on angular dimension with $\gamma_{1,n} = \frac{\sqrt{P_\theta}}{n}$.

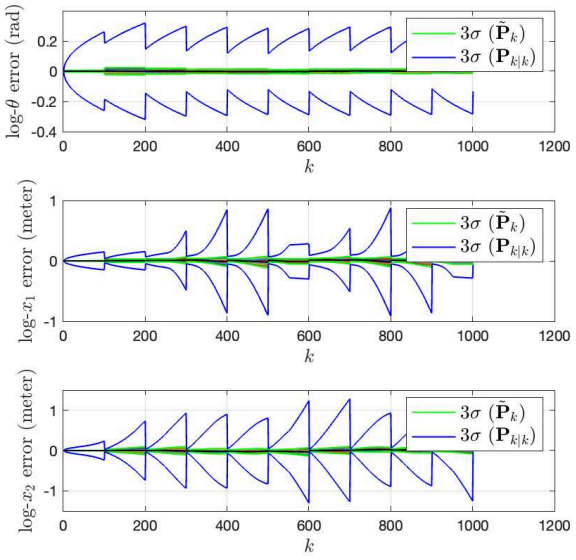
coupled with the CLVQ, gains performance in its ability to reduce the estimation error. During prediction steps only, both filters seem to behave similarly. However, (almost) every time a position measurement is acquired (one observation per second), the log-rotation error of the OQ-UKF-LG is reduced compared to that of the UKF-LG. This reduction is carried out on the log-position error (probably due to correlations between rotational and translational parts). The reason is that, using a default UKF-LG, the correction of the covariance is not (or poorly) performed on the angular part due to position observations only, and that the variance of the angular estimate increases (while it could benefit from decreasing instead). To some extent, enhancing the UKF-LG with the CLVQ allows us to reduce the variance of the state of interest. This enables us to limit the conservatism and provides us with a better estimate.

Consistency results:

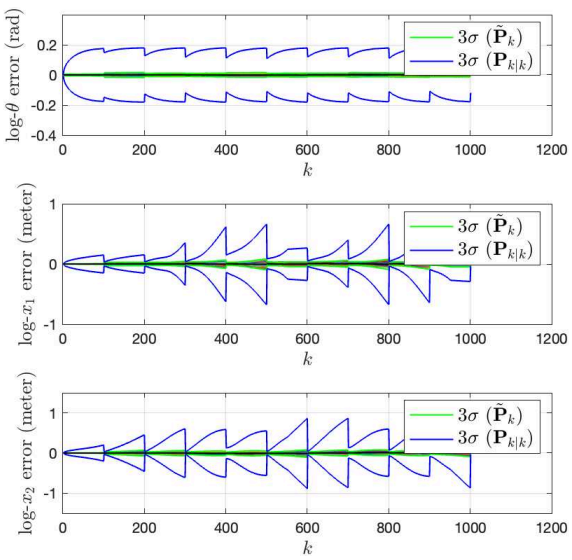
The last results were obtained from a single run. To assess statistical significance, it is necessary to run multiple Monte-Carlo simulations as described in subsection §6.1.3. This will allow us to check for the consistency by computing an empirical covariance of the estimation error $\tilde{\mathbf{P}}_k$ and compare it with the covariance $\mathbf{P}_{k|k}$ given by the Kalman algorithm.

Hence, we perform $N_s = 100$ independent simulations in the same configuration as

previously. All of the log-Euclidean errors, as a function of k, are computed and plotted in Figure B.5, as well as the covariance of the estimation error and the covariance of the filter estimate, both given in terms of 3σ confidence interval. Plots are on the same scale in the y-axis to facilitate the comparison. Globally, if we compare the variances, we can



(a) Results of the UKF-LG. The Root Mean (over all the trajectory) Square Error in rotation and position (averaged over the N_s Monte-Carlos) are respectively given by 0.30° and 0.029 m.



(b) Results of the OQ-UKF-LG. The Root Mean (over all the trajectory) Square Error in rotation and position (averaged over the N_s Monte-Carlos) are respectively given by 0.23° and 0.022 m.

Figure B.5: Log-Euclidean errors on all three state variables θ, x_1 and x_2, as a function of k.

see that both filters are conservative, since $\mathbf{P}_{k|k}$ encompasses $\tilde{\mathbf{P}}_k$, but the OQ-UKF-LG tends to provide a covariance $\mathbf{P}_{k|k}$ that is closer to $\tilde{\mathbf{P}}_k$, making it less conservative than the other filter. This has been beneficial for the estimation performance, as the mean RMSE (averaged over the N_s simulations) has been reduced for both rotation (-23.3%) and position (-24.1%).

If we compute the ANEES for both filters (Figure B.6), we see that they are indeed in the conservative domain for the most part. At the update step (every 100 iterations), the OQ-UKF-LG tends to approach the consistent domain more than the standard UKF-LG, as indicated by the multiple spikes..

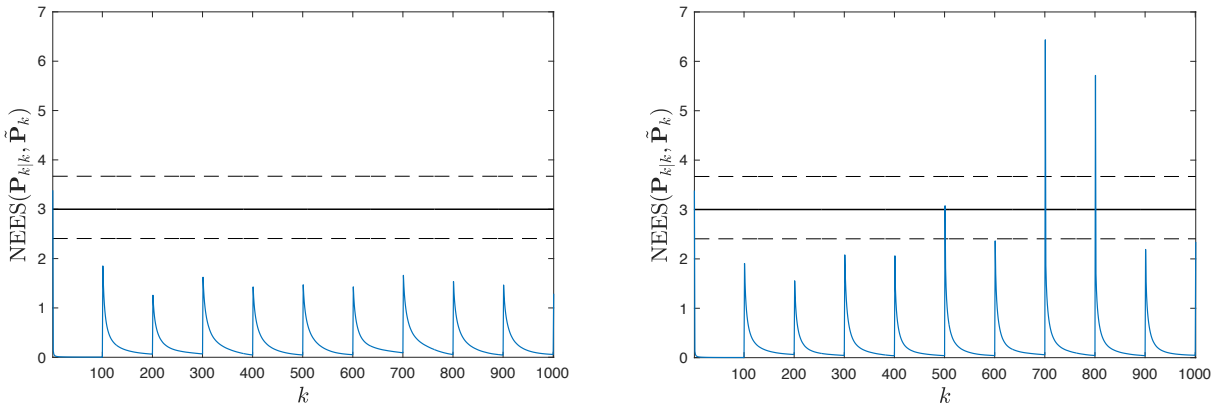
■ Monte-Carlo simulations

We now perform N_s Monte-Carlo runs to assess statistical significance given different parameterizations. In particular, we assess the sensitivity of the OQ-UKF-LG to overestimations of noise parameters $\mathbf{Q}$ and $\mathbf{R}$, and to different CLVQ parameters.

Sensitivity to overestimations of Q and R

Hyper-parameters:

Measurement model: $\mathbf{h}_{\text{gps}}$
Measurement frequency: $f_{obs} = 1$ Hz



(a) ANEES of the UKF-LG as a function of k.

(b) ANEES of the OQ-UKF-LG as a function of k.

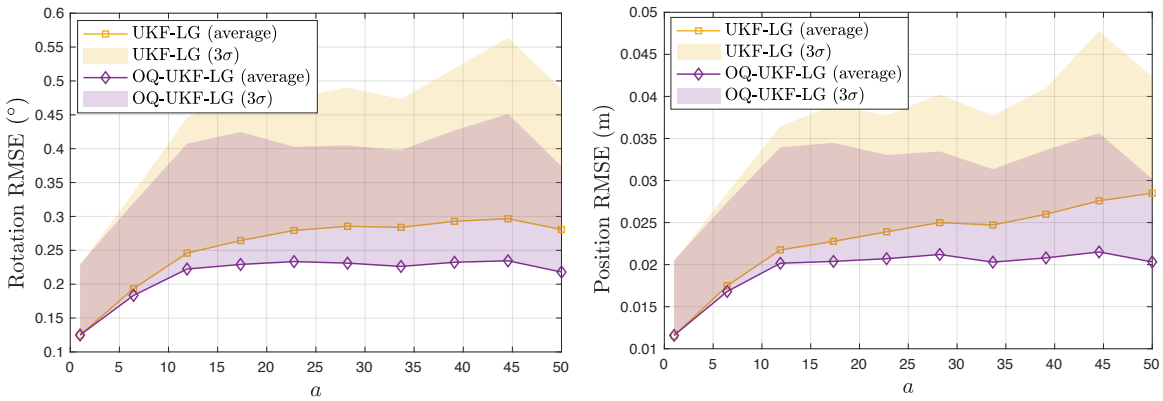
Figure B.6: Comparison of the ANEES between (a) the UKF-LG and (b) the OQ-UKF-LG. $f_{obs} = 1$ Hz, $\hat{\mathbf{Q}} = 50^2 \times \mathbf{Q}$, $\hat{\mathbf{R}} = \mathbf{R}$.

Input noise: $\mathbf{Q} = \text{diag}\left(\left[(\frac{\pi}{180})^2, (0.01)^2, (0.01)^2\right]\right)$
Input uncertainty parameter: $\hat{\mathbf{Q}} = a^2 \times \mathbf{Q}$
Output noise: $\mathbf{R} = \text{diag}\left([0.01^2, 0.01^2]\right)$
Output uncertainty parameter: $\hat{\mathbf{R}} = b^2 \times \mathbf{R}$

For the CLVQ parameters, we only readjust the dimensions corresponding to the three states of interests, and parameterized as follows:

$$n_{max} = 100, \quad \gamma_{1,n} = \frac{\sqrt{P_\theta}}{n}, \quad \gamma_{2,n} = \frac{\frac{1}{10}\sqrt{P_{\rho_1}}}{n}, \quad \gamma_{3,n} = \frac{\frac{1}{10}\sqrt{P_{\rho_2}}}{n}, \quad n = 1, \dots, n_{max}$$

We evaluate the performance in terms of RMSE over the whole trajectory, as a function of the level a or b. The results are given in Figures B.7 and B.8. Regarding the RMSE



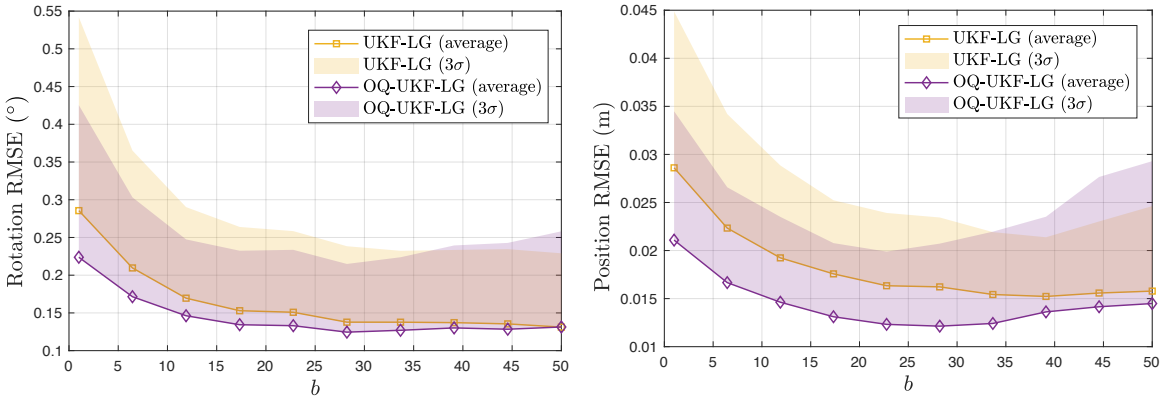
(a) RMSE in rotation as a function of a ($b = 1$).

(b) RMSE in position as a function of a ($b = 1$).

Figure B.7: RMSE of all Monte-Carlo simulations in rotation and in position, as a function of the noise level a, with b fixed. For each noise level a, the average Monte-Carlo of the RMSE is plotted along with its confidence interval (upper-half) in the colored area.

as a function of an overestimation of $\mathbf{Q}$ (with $\hat{\mathbf{R}} = \mathbf{R}$) (cf. Figure B.7), we recover the pattern that we observed previously. Namely, using the CLVQ is beneficial when the input noise $\mathbf{Q}$ is overestimated, and the difference between the UKF-LG and the OQ-UKF-LG

is more pronounced as $\hat{\mathbf{Q}}$ is large compared to $\mathbf{Q}$, for both rotational and translational parts. Regarding the RMSE as a function of an overestimation of $\mathbf{R}$, with $\hat{\mathbf{Q}} = 50^2 \times \mathbf{Q}$



(a) RMSE in rotation, as a function of b ($a = 50$).

(b) RMSE in position, as a function of b ($a = 50$).

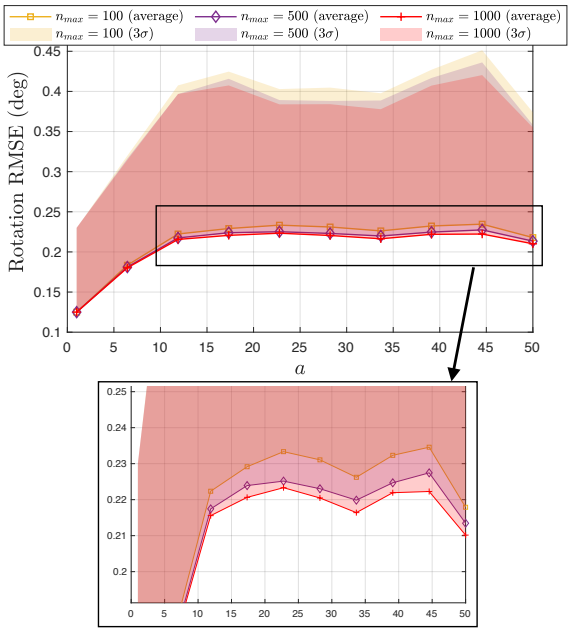
Figure B.8: RMSE of all Monte-Carlo simulations in rotation and in position, as a function of the noise level b, with a fixed. For each noise level b, the average Monte-Carlo of the RMSE is plotted along with its confidence interval (upper-half) in the colored area.

(Figure B.8), we first notice that the inverse trend occurs: the performance decreases as b increases for both filters, and the OQ-UKF-LG provides no significant improvement over the UKF-LG after reaching some value b. In addition, although the average RMSE of the OQ-UKF-LG remains inferior to that of the UKF-LG, there are some realizations of the former that present a less good performance for high values of b, as indicated by the 3σ upper bound.

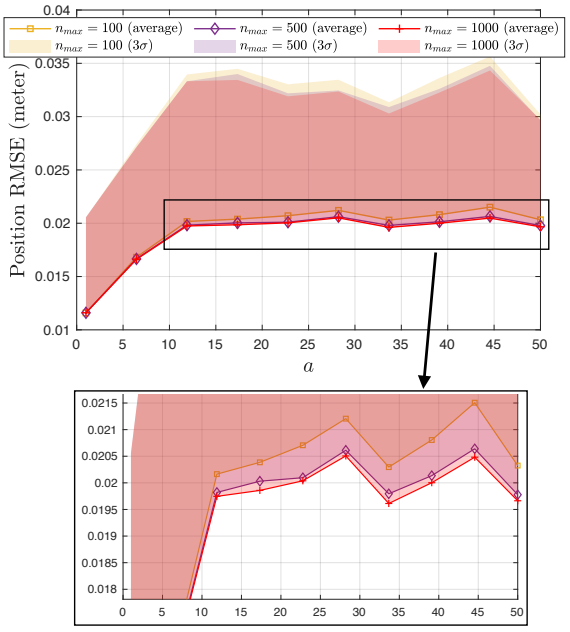
Sensitivity to CLVQ parameters

We now evaluate the sensitivity of the OQ-UKF-LG to other parameterizations of the CLVQ, namely, different parameterizations of the number of iterations n_{max}, and the sequence of learning rates γ.

Taking again $\hat{\mathbf{Q}} = a^2 \times \mathbf{Q}$ and $\hat{\mathbf{R}} = \mathbf{R}$ where $\mathbf{Q}$ and $\mathbf{R}$ are left unchanged, we start by evaluating the sensitivity to different integers $n_{max} = \{100, 500, 1000\}$, all other parameters being the same as before. Similarly, the results are presented in terms of RMSE as a function of a (cf. Figure B.9). A focus on the average RMSE shows that increasing the parameter n_{max} allows us to optimize the performance of the OQ-UKF-LG. This was expected since, according to Theorem 2.1, the stochastic gradient descent performed by the CLVQ algorithm is expected to converge to an optimal solution as n tends to infinity. Hence, the more n_{max} is set to large values, the more we can optimize the quality of the estimate, but at the cost of computational efficiency. For this set of experiments, the time duration of a single run (averaged over the 100 Monte-Carlos) for each n_{max} was: 0.2984 seconds ($n_{max} = 100$), 1.0381 seconds ($n_{max} = 500$) and 1.9866 seconds ($n_{max} = 1000$). The latter is then 6.65× longer than the first duration, for 10× more iterations during the CLVQ loop. In addition, the time duration for the UKF-LG without optimal quantization was 0.0968 seconds in average.



(a) RMSE of the OQ-UKF-LG (in rotation) as a function of a.



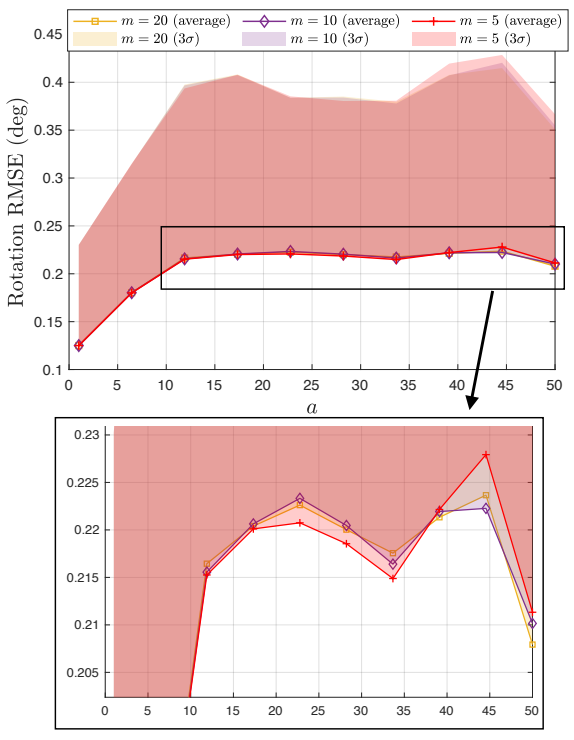
(b) RMSE of the OQ-UKF-LG (in position) as a function of a.

Figure B.9: RMSE of the OQ-UKF-LG as a function of the noise level a, and for different values of n_{max}.

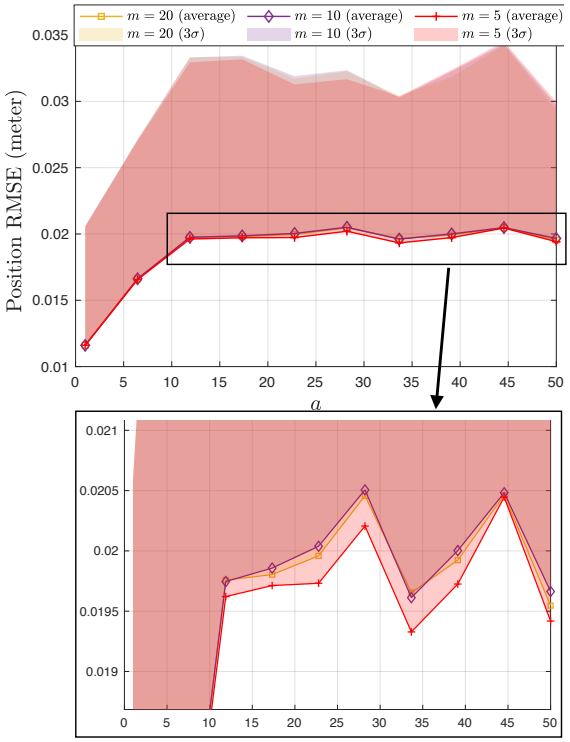
We also assess the sensitivity to the learning rate, or step sequence $\{\frac{\frac{1}{\nu}\sqrt{\mathbf{P}_{\ell\ell}}}{n}\}_{n=1,\dots,n_{max}}$, $\forall \ell = 1, \dots, 3$. Recall that if m is small, then the sigma-points are strongly readjusted during the first steps of the CLVQ loop, since the step is large. In contrast, if m is taken large, the sigma-points are weakly readjusted. For these simulations, we set:

$$n_{max} = 1000, \quad \gamma_{1,n} = \frac{\sqrt{P_\theta}}{n}, \quad \gamma_{2,n} = \frac{\frac{1}{\nu}\sqrt{P_{\rho_1}}}{n}, \quad \gamma_{3,n} = \frac{\frac{1}{\nu}\sqrt{P_{\rho_2}}}{n}, \quad n = 1, \dots, n_{max}$$

and evaluate the performance for three different values $m = \{5, 10, 20\}$, all other things being equal. The results are given in Figure B.10. The results are quite nuanced, since for the position RMSE (Figure B.10b), taking a large step ($m = 5$) enabled an optimization of the performance for all a, while for the rotation RMSE (Figure B.10a), taking $m = 5$ provided better performances for $a < 40$ only. In addition, the sensitivity of the position RMSE to the variation of m has been expected since we varied the learning rates for positions only. Regarding the sensitivity of the rotation RMSE to the variation of m, it may be explained by the existing correlations between the angular and position variables. Between $m = 10$ and $m = 20$, there does not seem to be a regular pattern when comparing the two of them, and the difference in performance is less marked than with $m = 5$. In this case in particular, $m = 20$ seems to provide the best compromise, but that does not mean that it will be the case for experiments with other hyperparameters. In general, the tuning of the learning rates should be performed according to the state variables that we would like to optimize, while paying attention to the correlations between these variables.



(a) RMSE of the OQ-UKF-LG (in rotation) as a function of a.



(b) RMSE of the OQ-UKF-LG (in position) as a function of a.

Figure B.10: RMSE of the OQ-UKF-LG as a function of the noise level a, and for different values of m.

Results for the observation model $\mathbf{h}_{\text{lidar}}$: We evaluate the performance where the measurements are given in the form of processed range-and-bearing observations, as modeled by equations (6.3)-(6.4). We consider the presence of $N_l = 3$ landmarks with the following positions: $\boldsymbol{l}^{(1)} = [4, -2]^\top$, $\boldsymbol{l}^{(2)} = [6, 3]^\top$, $\boldsymbol{l}^{(3)} = [6, -7]^\top$.

The simulation setting is the same as before, i.e. $\hat{\mathbf{Q}} = a^2 \times \mathbf{Q}$ and $\hat{\mathbf{R}}^{(i)} = b^2 \times \mathbf{R}^{(i)}$ for the i^{th} landmark, with $\mathbf{Q} = \text{diag}\Big((\frac{\pi}{180})^2, (0.01)^2, (0.01)^2\Big)$ and $\mathbf{R}^{(i)} = \text{diag}\Big((0.01)^2, (0.01)^2\Big)$ $\forall i = 1, 2, 3$.

For the CLVQ parameters, in contrast to previously, we have set different values for the CLVQ in prediction and the CLVQ in correction. Regarding the first one, we have set:

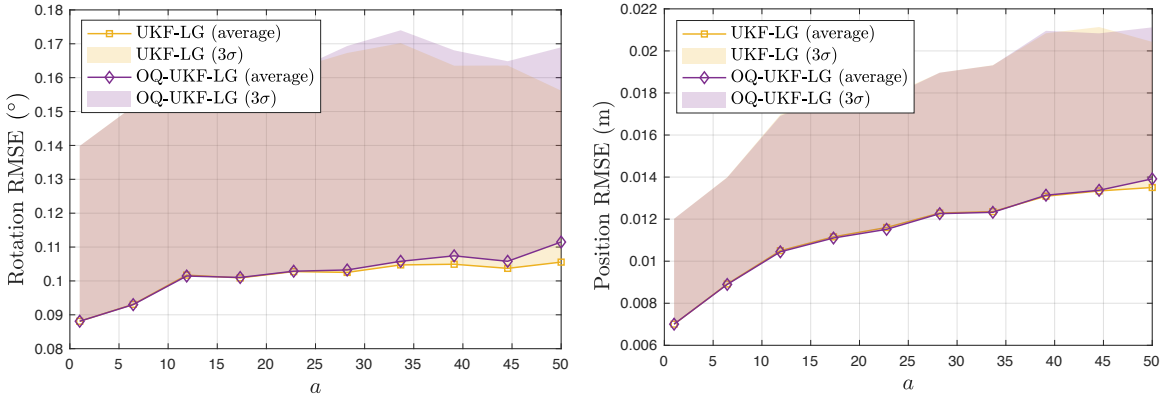
$$n_{max} = 100, \quad \gamma_{1,n} = \frac{\frac{1}{5}\sqrt{P_\theta}}{n}, \quad \gamma_{2,n} = \frac{\frac{1}{5}\sqrt{P_{\rho_1}}}{n}, \quad \gamma_{3,n} = \frac{\frac{1}{5}\sqrt{P_{\rho_2}}}{n}, \quad n = 1, \dots, n_{max}$$

And regarding the second, we have set:

$$n_{max} = 100, \quad \gamma_{1,n} = \frac{\frac{1}{10}\sqrt{P_\theta}}{n}, \quad \gamma_{2,n} = \frac{\frac{1}{10}\sqrt{P_{\rho_1}}}{n}, \quad \gamma_{3,n} = \frac{\frac{1}{10}\sqrt{P_{\rho_2}}}{n}, \quad n = 1, \dots, n_{max}$$

We again performed $N_s = 100$ Monte-Carlo simulations for different noise levels a and b, then plotted the results in Figures B.13 and B.12. As we can see, the CLVQ algorithm is unable to improve the performance of the OQ-UKF-LG over the UKF-LG when we overestimate $\mathbf{Q}$ by setting $a > 1$ while maintaining consistency in $\mathbf{R}$ by setting $b = 1$ (cf. Figure

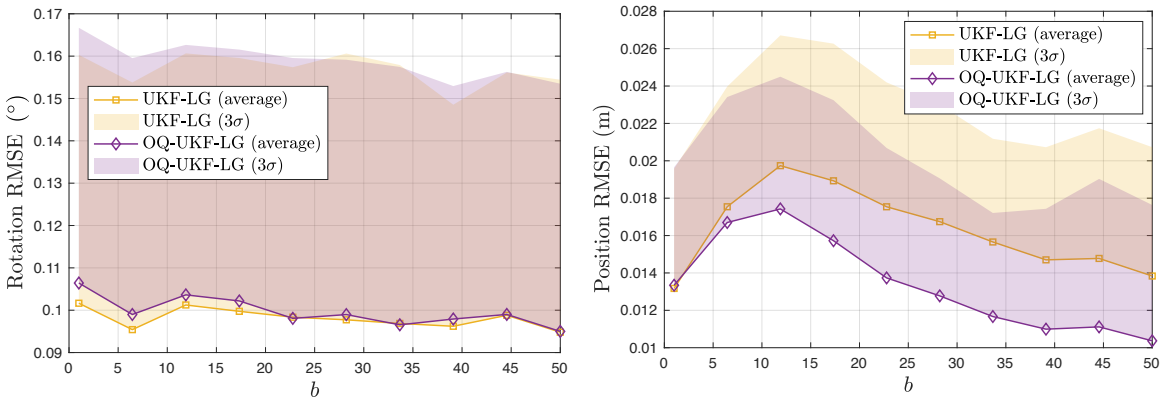
B.13). For a near 50, the OQ-UKF-LG is even less accurate than the standard UKF-LG. However, when both noise are overestimated ($a = 50$ and $b > 1$), it seems possible to greatly decrease the RMSE in position of the OQ-UKF-LG (cf. Figure B.12b), as well as to decrease its RMSE in rotation until it reaches the same performance as the UKF-LG.



(a) RMSE in rotation as a function of a ($b = 1$).

(b) RMSE in position as a function of a ($b = 1$).

Figure B.11: RMSE of all Monte-Carlo simulations in rotation and in position, as a function of the noise level a, with b fixed, and for the observation model $\mathbf{h}_{\text{lidar}}$.



(a) RMSE in rotation, as a function of b ($a = 50$).

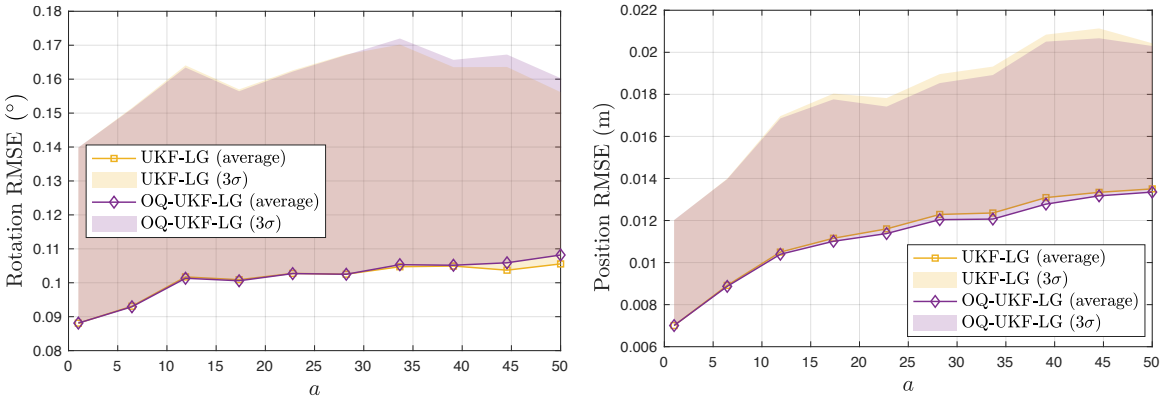
(b) RMSE in position, as a function of b ($a = 50$).

Figure B.12: RMSE of all Monte-Carlo simulations in rotation and in position, as a function of the noise level b, with a fixed, and for the observation model $\mathbf{h}_{\text{lidar}}$.

It is still possible to optimize the OQ-UKF-LG by tuning the CLVQ parameters. For example, the results of Figures B.13 and B.14 have been obtained by modifying the learning rates of the CLVQ (in the prediction step) as follows:

$$\gamma_{1,n} = \frac{\frac{1}{2}\sqrt{P_\theta}}{n}, \quad \gamma_{2,n} = \frac{\frac{1}{2}\sqrt{P_{\rho_1}}}{n}, \quad \gamma_{3,n} = \frac{\frac{1}{2}\sqrt{P_{\rho_2}}}{n},$$

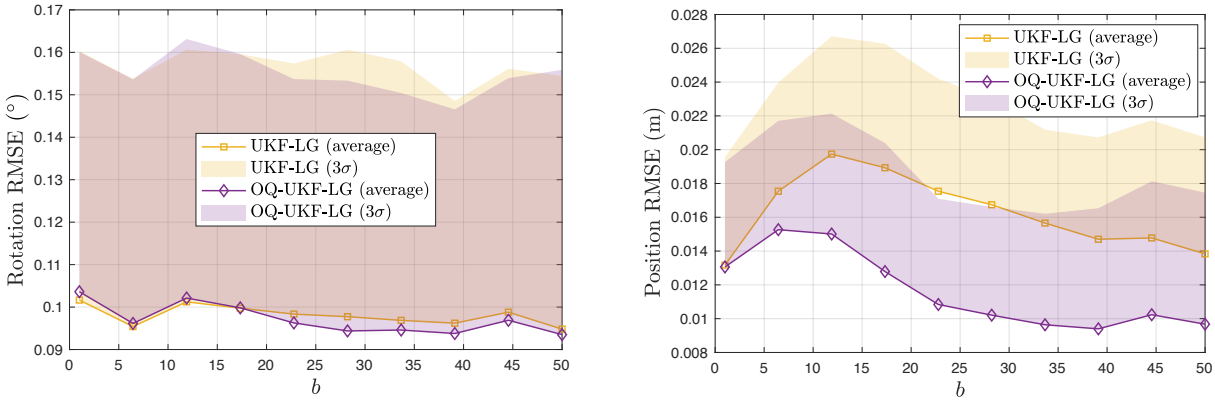
This shows that it can be a matter of tuning and that this tuning may be case-dependent.



(a) RMSE in rotation as a function of a ($b = 1$).

(b) RMSE in position as a function of a ($b = 1$).

Figure B.13: RMSE of all Monte-Carlo simulations in rotation and in position, as a function of the noise level a, with b fixed, and for the observation model $\mathbf{h}_{\text{lidar}}$.



(a) RMSE in rotation, as a function of b ($a = 50$).

(b) RMSE in position, as a function of b ($a = 50$).

Figure B.14: RMSE of all Monte-Carlo simulations in rotation and in position, as a function of the noise level b, with a fixed, and for the observation model $\mathbf{h}_{\text{lidar}}$.

B.5.2 Evaluation of the OQ-UKF-LG on real data

We evaluated the performance of the OQ-UKF-LG on the difficult scenario of the EuRoc datasets (cf. subsection §6.4.1). The filter was parameterized with the following setting:

CLVQ parameters for the prediction and correction UTs:

$$n_{max} = 100$$

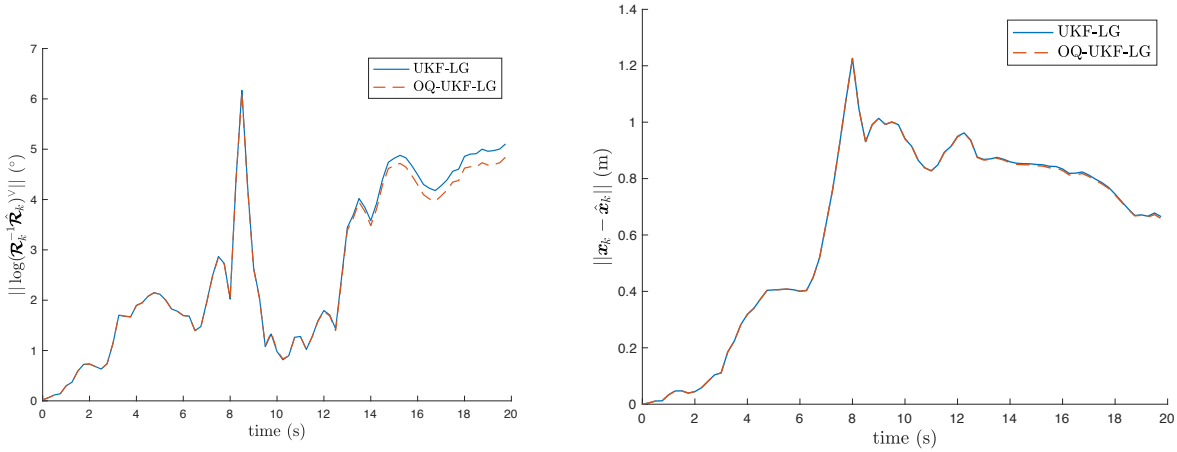
$$\gamma_{\ell,n} = \frac{\frac{1}{5}\sqrt{\mathbf{P}_{\ell\ell}}}{n}, \quad \forall \ell = \{1, 2, 3\}$$

$$\gamma_{\ell,n} = \frac{\frac{1}{10}\sqrt{\mathbf{P}_{\ell\ell}}}{n}, \quad \forall \ell = \{7, 8, 9\}$$

(Hence, we readjusted only the rotation components and position components of the sigma-points)

Results:

Figure B.15 compares the performance of the optimal quantization–based UKF-LG with that of the standard UKF-LG. For most of the trajectory, both filters exhibit similar behavior. However, at a specific time instant ($t \approx 14$ s), the OQ-UKF-LG begins to slightly outperform the UKF-LG. This improvement is primarily reflected in the attitude estimation, while no significant gain is observed in the position estimation.



(a) Log-rotation error as a function of time.

(b) Position error as a function of time.

Figure B.15: Evolution of the error as a function of time.

Appendix C

Résumé étendu en Français 🇫🇷 - French summary

Contexte

Cette thèse, cofinancée par la fédération ONERA–ISAE–ENAC et menée à Toulouse (France), s'inscrit dans une problématique issue du domaine de la navigation robotique terrestre et/ou aérienne, impliquant des systèmes tels que des robots à roues ou des drones. L'essor d'applications toujours plus exigeantes, qu'elles soient civiles (livraison autonome par drone, inspection de zones à risque, opérations de secours, etc.) ou militaires (coordination de flottes de drones, reconnaissance, etc.), impose aux systèmes robotiques un haut niveau d'autonomie dans l'exécution de leurs missions.

Une exigence centrale pour ces systèmes est leur capacité à estimer de manière fiable leur propre état au cours du temps. Pour les robots mobiles, cet état comprend généralement la position et l'orientation (désignées collectivement sous le terme de pose), ainsi que d'autres grandeurs telles que la vitesse ou les biais des capteurs. Une estimation précise de la pose constitue un prérequis pour les tâches de plus haut niveau, comme la planification, le contrôle ou la prise de décision, dans la mesure où les stratégies de commande par retour d'état dépendent directement de la qualité de l'état estimé.

En pratique, l'estimation d'état repose sur des mesures bruitées et imparfaites fournies par les capteurs embarqués. Les contraintes de masse, de volume et de consommation énergétique limitent la nature et la qualité des instruments pouvant être intégrés, en particulier sur les plateformes mobiles et aériennes, souvent équipées d'unités de mesure inertielle (IMU), de lidars, de sonars, de baromètres, etc., chacun ne fournissant qu'une information partielle sur l'état du système. Cela motive le recours à des techniques de fusion de données capables de combiner plusieurs sources d'information tout en tenant explicitement compte des incertitudes associées.

Cadre scientifique

Le cadre scientifique de la thèse se situe à un niveau algorithmique. En particulier, de nombreux algorithmes d'estimation existent déjà pour les applications précédemment évoquées, dont le très connu filtre de Kalman. Ces méthodes sont basées sur des modèles mathématiques décrivant le comportement du système considéré. Or, cette modélisation est souvent imparfaite et sujette à des hypothèses simplificatrices. Couplée aux problèmes relatifs à la qualité des capteurs évoqués précédemment, la reconstruction de l'état s'en trouve dégradée et implique une certaine incertitude quant à la qualité de l'estimation. Capturer cette incertitude est crucial, car elle est souvent utilisée dans les schémas algorithmiques afin de pondérer les corrections de l'estimation. Ainsi, cette thèse met l'accent sur la caractérisation de cette incertitude, au travers d'outils issus des probabilités. On se place spécifiquement dans un cadre bayésien. En particulier, le filtrage de Kalman fournit déjà un cadre privilégié à partir duquel nous pouvons apporter une contribution scientifique pour traiter ces problématiques d'estimation d'état sous incertitude.

Objectifs et méthodologie générale

La finalité de cette thèse est de proposer une ou plusieurs solutions algorithmiques permettant d'estimer correctement la position et l'attitude (orientation) d'un corps rigide par fusion de données, en présence de bruits et d'incertitudes élevés.

Plus précisément, les apports scientifiques (par rapport au cadre adopté) concernent la mise en place d'outils probabilistes permettant de représenter l'incertitude des variables aléatoires de manière adéquate, compte tenu de leur nature. Par ailleurs, on cherche également à estimer correctement la propagation de cette incertitude lorsque les variables aléatoires évoluent à travers des modèles non linéaires.

La méthodologie générale de ce projet de thèse consiste à mettre en place ces outils, puis à les appliquer, en particulier, à une technique de filtrage de Kalman préexistante. L'algorithme d'estimation ainsi développé sera d'abord évalué par des simulations utilisant des données artificiellement bruitées selon différentes lois de probabilité afin d'analyser ses performances dans des conditions contrôlées. Enfin, il fera l'objet d'une validation expérimentale sur des données réelles issues de jeux de données open source.

État de l'art - du filtrage bayésien au filtrage de Kalman

Étant donné le cadre stochastique dans lequel on se place, le filtre bayésien (ou estimateur récursif bayésien) constitue une solution adéquate pour estimer en temps réel l'état d'un système dynamique quelconque, connaissant l'état passé du système ainsi que l'ensemble des mesures passées et actuelles. Cette méthode d'estimation repose sur une formulation probabiliste de l'état, de manière à modéliser le degré d'incertitude issu de la connaissance imprécise des informations disponibles. Elle se décompose en un schéma récursif de prédiction/correction, où la première étape s'appuie sur le modèle dynamique du système afin d'en prédire une estimation brute de son état. La deuxième étape prend

en compte une mesure/observation, via l'information de sortie disponible, et pondère un terme de correction selon la fiabilité accordée à la mesure ou au modèle. Malgré le fait que l'estimation bayésienne puisse offrir une solution optimale et générique au problème d'estimation, elle n'en fournit qu'une solution théorique et repose sur des calculs qui sont, en pratique, inextricables ou difficiles à mettre en œuvre en temps réel. Pour pallier ce problème, de nombreuses méthodes approchées du filtrage bayésien ont vu le jour depuis les travaux de Kalman (1960). Ce dernier est à l'origine de la méthode de filtrage qui porte son nom, initialement utilisée avec succès dans le cadre de la navigation spatiale. Dès lors, cette technique a fait l'objet de nombreux développements dans des domaines variés tels que l'aérospatiale, l'aéronautique, la "cartographie et localisation simultanées" (SLAM), mais aussi l'océanographie, la météorologie ou la finance.

Concrètement, le filtre de Kalman a été décliné en plusieurs versions afin de traiter les non-linéarités présentes dans de nombreux systèmes dynamiques. Deux grandes familles peuvent être distinguées : les filtres de Kalman "étendus" (Extended) et les filtres de Kalman "sans parfum" (Unscented). Tandis que la première repose sur une linéarisation du modèle, avec une hypothèse de distribution gaussienne sur les variables aléatoires afin d'appliquer les équations linéaires du filtre de Kalman, la seconde consiste en une discrétisation de la distribution de probabilité des variables aléatoires en jeu. Cette méthode exploite un nuage de points (les "sigma-points") et propage ce nuage afin de reconstruire une estimation basée sur cet échantillonnage, un processus appelé "Unscented Transformation" (UT). Le travail de thèse s'intéresse en particulier à ce dernier type de méthode car, outre ses avantages par rapport à une approche fondée sur la linéarisation, la caractérisation de l'incertitude par un nuage de points constitue une approche pertinente pour se rapprocher des équations génériques de l'estimateur récursif bayésien. En effet, le nuage de points ne nécessite pas nécessairement de supposer une distribution gaussienne (bien que cette hypothèse reste largement utilisée dans ce cadre). Ainsi, le choix de la densité de probabilité est plus "libre", tout en maintenant certaines contraintes statistiques souhaitées, par exemple en imposant les moments de la distribution (moyenne, covariance, etc.).

Les filtres de Kalman Unscented (UKF) sont ainsi issus de l'idée de générer des sigma-points en fonction des statistiques connues de la distribution, puis de les propager à travers les différentes fonctions intervenant dans la modélisation du système, afin de reconstruire les statistiques à partir de cet échantillon propagé. De ce fait, un élément crucial pour obtenir une estimation précise de l'état du système concerne le placement adéquat des sigma-points (ainsi que de leurs poids associés). Les méthodes permettant cela reposent sur un échantillonnage déterministe, fondé sur des techniques d'intégration numérique de type "quadrature de Gauss". Ces techniques permettent notamment de générer les sigma-points de sorte que les moments statistiques du nuage coïncident avec la moyenne et la covariance de la distribution réelle. Cependant, un problème apparaît après la propagation : ces techniques ne garantissent pas l'exactitude entre les moments statistiques reconstruits à partir des sigma-points propagés et ceux de la distribution réelle propagée,

car dépendent fortement des non-linéarités du modèle et/ou de la dispersion initiale. Ce problème constitue un premier frein à la performance des UKFs.

Une autre famille de méthodes d'estimation, plus proche du filtre bayésien, est le filtrage particulaire. Cette technique repose également sur un échantillonnage, mais dont les points (en plus grand nombre) sont placés de manière aléatoire tout en respectant quasi exactement la distribution réelle des variables aléatoires en jeu. Cette famille de méthodes fournit une estimation très précise, mais coûteuse en temps de calcul. Dans cette thèse, on privilégie les UKFs, moins coûteux en ressources de calcul.

État de l'art - Estimateurs invariants

Outre le choix de porter ce travail de recherche sur une approche de type UKF, le contexte de la navigation robotique incite également à se diriger vers des techniques d'estimation dites "invariantes" (ou préservant les symétries). Ces techniques sont nées de la volonté de préserver les contraintes géométriques inhérentes aux problèmes de navigation. En effet, les modèles physiques décrivant la dynamique d'un corps rigide possèdent des symétries intrinsèques qu'il convient de prendre en compte dans la synthèse d'observateurs. En termes mathématiques, une symétrie est une transformation qui laisse inchangé l'objet sur lequel elle est appliquée. Dans le cadre des modèles dynamiques de corps rigides, il existe un ensemble de changements de variables (implicitement, des groupes de transformations) qui rendent le modèle inchangé sous ce nouveau système de coordonnées. On parle ainsi de symétrie des "modèles", intrinsèquement liée aux propriétés géométriques de l'état du système considéré. Idéalement, il serait préférable de synthétiser un observateur dont les propriétés de symétrie sont préservées tout au long du schéma récursif de prédiction/correction, afin d'obtenir une estimation cohérente avec la géométrie du problème. Un filtre de Kalman standard ne prend pas en compte cet aspect. Étant donné cela, la théorie des observateurs invariants (dont le filtrage de Kalman invariant) s'est fortement développée au cours des vingt dernières années et constitue désormais une approche privilégiée pour une grande partie des problèmes rencontrés en robotique et en navigation inertielle.

Les premiers observateurs invariants ont été développés à partir des travaux de N. Aghannan, P. Rouchon et S. Bonnabel, initialement dans un cadre déterministe (ne prenant pas en compte le bruit). Leurs résultats ont permis d'élaborer une méthode constructive utilisant conjointement la géométrie différentielle et la théorie des groupes pour synthétiser des observateurs non linéaires dédiés aux problèmes d'estimation de l'état d'un système, tout en préservant ses symétries. Leur conception exige, à travers différentes structures de groupe, l'identification de transformations agissant sur le modèle dynamique afin de reformuler le problème dans un cadre mathématique adapté. L'introduction d'une structure géométrique appropriée, reposant notamment sur la construction d'un champ de vecteurs invariants et sur la définition d'une erreur invariante, permet ensuite de finaliser l'élaboration de l'estimateur. Pour en régler la dynamique, les gains sont déterminés par

la linéarisation d'une erreur d'état intrinsèquement non linéaire. Cette linéarisation n'est plus effectuée autour d'un point d'équilibre, mais autour de l'élément neutre du groupe, ce qui étend le domaine de convergence de l'estimation tout en simplifiant l'expression de la dynamique d'erreur. Un autre avantage de ces estimateurs invariants est qu'ils possèdent des propriétés similaires à celles rencontrées dans le cas linéaire. Notamment, la dynamique de l'erreur est autonome, dans le sens où son évolution ne dépend que d'elle-même et non de l'état estimé, ce qui limite les risques de divergence.

Initialement, les observateurs invariants ont été considérés uniquement pour des systèmes invariants par certains types de transformations. Par la suite, la classe de systèmes pour lesquels il est possible de construire des estimateurs invariants a été étendue et correspond désormais aux systèmes dits "groupe affine". Une des particularités de cette classe est qu'elle généralise les fonctions affines sur un espace euclidien. Ces systèmes sont à valeurs dans un groupe de Lie (une variété différentielle munie d'une structure de groupe) et possèdent de manière explicite les propriétés linéaires évoquées précédemment. Depuis lors, un grand nombre d'estimateurs formulés sur ces variétés ont vu le jour, notamment les filtres de Kalman sur groupes de Lie. Dans le domaine de la robotique en particulier, les groupes de Lie matriciels tels que $\boldsymbol{SO}(n)$, $\boldsymbol{SE}(n)$, ou les groupes étendus $\boldsymbol{SE}_k(n)$ jouent un rôle crucial dans la description de l'orientation, de la position et de la vitesse des systèmes robotiques. À partir des travaux de G. Chirikjian, T. D. Barfoot et K. C. Wolfe sur la caractérisation de l'incertitude sur de telles variétés, les filtres de Kalman sur groupes de Lie ont pu être développés, notamment par S. Bonnabel et G. Bourmaud, conduisant à la formulation d'EKF sur groupes de Lie (EKF-LG), puis à des extensions naturelles telles que les UKFs sur groupes de Lie (UKF-LG). Par ailleurs, les filtres particulaires sur groupes de Lie ont aussi commencé à émerger récemment.

Problématiques traitées durant la thèse

Une problématique principale concerne la manière de modéliser l'incertitude d'une variable aléatoire vivant sur un groupe de Lie, au moyen d'une fonction de densité de probabilité (ddp). De manière générale, modéliser une ddp sur une variété non euclidienne (dans le but de résoudre des problèmes d'estimation stochastique) est délicat, car il convient de donner un sens à ses caractéristiques, en particulier à ses moments. En effet, les opérations usuelles utilisées en statistiques euclidiennes ne peuvent pas être directement généralisées aux variétés. Par conséquent, elles doivent être redéfinies en fonction des lois de composition et des distances propres à la variété considérée. Pour les variétés d'intérêt de cette thèse, à savoir les groupes de Lie matriciels, leur structure permet de modéliser l'incertitude à l'aide d'une ddp gaussienne dite concentrée", au sens où la dispersion est suffisamment faible pour considérer une notion de covariance "Lie-théorique" comme valide. Cette densité est ensuite transposée sur le groupe via une opération adaptée à sa structure, on parle ainsi de gaussienne "équivariante". C'est cette hypothèse de forte concentration autour de la moyenne qui est principalement utilisée dans un UKF-LG. Le problème de ce type de modélisation de l'incertitude est qu'elle n'est valable que locale-

ment. Au-delà d'un certain niveau de dispersion, la notion de covariance Lie-théorique perd son sens, et il devient nécessaire de modéliser l'incertitude de manière plus globale, en faisant appel à d'autres notions de moments statistiques. Il s'avère que la nature des variables angulaires, ainsi que leur couplage avec les variables translationnelles, permet de s'appuyer à la fois sur des distributions sphériques et sur une distribution gaussienne. La ddp "hybride" ainsi formulée doit être capable de représenter correctement ce couplage. Dans la littérature, il existe déjà plusieurs ddp capables de le faire (matrice de Fisher-Gauss, Gauss–von Mises, Bingham–Gauss, etc.), mais elles n'ont pas été spécifiquement conçues (ni nécessairement adaptées) pour un UKF-LG. Ainsi, en s'inspirant de ces outils, un objectif de la thèse est de développer une ddp applicable à ce type de filtre de Kalman, remplaçant par la même occasion l'hypothèse de gaussienne concentrée.

Étant donnée une hypothèse arbitraire sur la distribution des variables aléatoires en jeu, la performance d'un UKF (qu'il soit exprimé sur un espace euclidien ou sur un groupe de Lie) dépend fortement de la manière dont les sigma-points et leur pondération sont construits et propagés. C'est pourquoi une seconde problématique consiste à déterminer un placement adéquat des sigma-points sous une hypothèse gaussienne. Comme énoncé précédemment, les techniques de type quadrature de Gauss constituent une approche classique permettant de générer ces quantités à partir de la moyenne et de la covariance supposées. Cependant, après une transformation Unscented au travers d'une fonction non linéaire, il est probable que le nuage de points pondérés ne permette pas de reconstruire exactement les moments propagés réels. Dans cette thèse, l'objectif est de réduire ce type de source d'erreur à l'aide d'une approche numérique.

Développements réalisés

Cette thèse a fait l'objet de deux développements majeurs. Le premier consiste à traiter le problème de la modélisation gaussienne équivariante, en particulier l'utilisation exclusive de la covariance comme mesure de dispersion, susceptible de s'effondrer en présence de fortes incertitudes. Ainsi, ce premier développement se concentre sur une amélioration au niveau de la modélisation probabiliste. Le second développement vise plutôt à traiter l'incapacité des sigma-points à reconstruire exactement les moments statistiques après propagation dans une fonction non linéaire, et ce sous une hypothèse gaussienne valide (sans remettre en question la validité de la notion de covariance Lie-théorique).

Ainsi, deux directions méthodologiques ont été suivies durant la thèse :

1. Une approche basée sur une **formulation hybride gaussienne/von Mises** pour un espace d'état comprenant à la fois des variables de translation et des variables de rotation. Le but de cette approche est en partie de s'affranchir d'une ddp entièrement gaussienne sur l'espace tangent d'un groupe, tout en s'inspirant des modèles issus des statistiques directionnelles.

2. Une approche basée sur la **quantification optimale**, visant à raffiner le placement des sigma-points tout en supposant une hypothèse gaussienne valide. Cette approche

n'a pas pour objectif de remettre en question l'utilisation d'une gaussienne équivariante, mais cherche plutôt à fournir une solution numérique à l'approximation d'une loi normale par les sigma-points.

Ces deux approches offrent différentes manières (potentiellement complémentaires) de répondre aux objectifs scientifiques fixés, à savoir une représentation et une propagation fidèles de l'incertitude d'état, afin d'améliorer les performances des algorithmes de filtrage (dont l'UKF dans le cas présent) pour résoudre des problématiques de navigation inertielle.

1. Approche hybride

Cette approche consiste à modéliser une ddp pour un vecteur $\boldsymbol{\xi}$ vivant dans l'algèbre de Lie (plus précisément sur l'espace vectoriel tangent à l'identité du groupe). Cette ddp remplace partiellement la ddp gaussienne, dans la mesure où l'incertitude est désormais modélisée par une distribution de von Mises (univariée) pour chaque variable angulaire intervenant dans le vecteur $\boldsymbol{\xi}$. Ainsi, au lieu de représenter l'incertitude par une matrice de covariance pour la partie rotation, celle-ci est caractérisée par un coefficient de concentration unique pour chaque variable angulaire. Cependant, la covariance est conservée dans la formalisation de la loi, car elle peut s'avérer utile dans certains cas pratiques. La modélisation probabiliste est complétée par un paramètre de couplage entre les variables de translation et de rotation afin de représenter les corrélations potentielles entre ces variables. Cette ddp, que nous avons appelée distribution "von Mises-Gaussienne", est ensuite transposée sur le groupe de Lie via les opérations adéquates, aboutissant à ce que l'on appelle une distribution "von Mises-Gaussienne équivariante", entièrement caractérisée par une pose moyenne $\hat{\boldsymbol{\chi}} \in \boldsymbol{SE}(n)$, une covariance $\mathbf{P}$, et un ensemble de coefficients de concentration rassemblés dans le vecteur $\boldsymbol{\kappa}$. Les différentes étapes de l'UKF-LG sont ensuite adaptées à cette ddp, de manière à proposer un schéma récursif estimant classiquement la moyenne et la covariance de l'état, tout en intégrant les coefficients de concentration dans le processus d'estimation. Cela a abouti à l'algorithme **vMG-UKF-LG**, présenté dans [Pravong et al., 2025a].

2. Approche quantification optimale

Cette approche repose sur une densité de probabilité gaussienne classique et vise à positionner de manière optimale les sigma-points dans un espace vectoriel. Elle s'appuie sur les principes de la quantification optimale, qui consiste à approximer une densité continue par une densité discrète en optimisant un critère de coût. Elle constitue ainsi une alternative au placement des sigma-points, initialement déterminé par la résolution d'un problème de quadrature de Gauss. L'approche par quantification optimale propose de générer les sigma-points de manière classique, puis de les réagencer de manière non déterministe à l'aide d'un algorithme de gradient stochastique. Ce réajustement optimal des sigma-points, sous hypothèse gaussienne, constitue le cœur de l'approche et peut ensuite être intégré dans un UKF standard, aboutissant à l'algorithme **OQ-UKF** [Pravong et al., 2024]. Par la suite, la méthode a été naturellement étendue à un UKF-LG, tout en démontrant son intérêt dans une architecture de commande LQG (Linear Quadratic Gaussian) [Pravong et al., 2025b].

Les deux algorithmes ainsi développés ont ensuite fait l'objet d'une validation expérimentale. Dans un premier temps, cette validation a été menée sur des jeux de données simulées, permettant de contrôler précisément les hypothèses de bruit sous-jacentes. Dans un second temps, les méthodes ont été évaluées sur des jeux de données réelles disponibles en open source. Les résultats obtenus mettent en évidence des améliorations significatives des performances d'estimation de l'attitude pour le vMG-UKF-LG, ainsi que des gains notables pour l'OQ-UKF-LG, en comparaison avec l'UKF-LG classique.

Conclusion et perspectives

Cette thèse a abouti à l'élaboration de deux versions du filtre de Kalman sans parfum : un **vMG-UKF-LG** et un **OQ-UKF-LG**. La première peut être considérée comme une tentative de s'affranchir, au moins partiellement, d'une ddp gaussienne[1] sur un groupe de Lie, tout en conservant la simplicité calculatoire des UKFs. La seconde, comme son nom l'indique, constitue plutôt une tentative d'optimisation sous une hypothèse de probabilité déjà largement admise dans la littérature.

Les méthodes proposées dans cette thèse restent encore limitées dans leur manière de capturer et de propager l'incertitude. En particulier, l'approche vMG repose sur plusieurs hypothèses simplificatrices et ne permet pas de modéliser de manière exacte certaines corrélations. Quant à l'approche par quantification optimale, elle n'a été développée que dans le cadre gaussien et nécessite le réglage de paramètres supplémentaires, dont l'automatisation n'a pas encore été traitée.

En perspective, cette thèse ouvre plusieurs axes de recherche, tels que l'automatisation du réglage des paramètres de la méthode de quantification optimale, son extension à des lois de probabilité non gaussiennes (pour un potentiel couplage entre les deux approches proposées), ou encore un ajustement du modèle vMG intégrant des caractéristiques négligées au cours des travaux. De plus, ces approches pourraient trouver des applications au-delà du seul cadre de la navigation inertielle, car elles reposent sur des concepts relativement fondamentaux.

[1] Pour un UKF-LG en tout cas. En effet, la littérature est déjà bien fournie en tentatives de ce type, mais avec des algorithmes d'estimation souvent assez différents d'un Kalman sur le plan structurel.

Bibliography

Aghannan, N. and Rouchon, P. (2002). On invariant asymptotic observers. In *Proceedings of the 41st IEEE Conference on Decision and Control, 2002.*, volume 2, pages 1479–1484. IEEE.

Alspach, D. and Sorenson, H. (1972). Nonlinear bayesian estimation using gaussian sum approximations. *IEEE Transactions on Automatic Control*, 17(4):439–448.

Anderson, B. D. and Moore, J. B. (2007). The kalman-bucy filter as a true time-varying wiener filter. *IEEE Transactions on Systems, Man, and Cybernetics*, (2):119–128.

Arasaratnam, I. and Haykin, S. (2009). Cubature kalman filters. *IEEE Transactions on automatic control*, 54(6):1254–1269.

Arasaratnam, I., Haykin, S., and Elliott, R. J. (2007). Discrete-Time Nonlinear Filtering Algorithms Using Gauss–Hermite Quadrature. *Proceedings of the IEEE*, 95(5):953–977. Conference Name: Proceedings of the IEEE.

Athans, M., Wishner, R., and Bertolini, A. (1968). Suboptimal state estimation for continuous-time nonlinear systems from discrete noisy measurements. *IEEE Transactions on Automatic Control*, 13(5):504–514.

Bailey, T., Nieto, J., Guivant, J., Stevens, M., and Nebot, E. (2006). Consistency of the EKF-SLAM algorithm. pages 3562–3568.

Barfoot, T. D. and Furgale, P. T. (2014). Associating uncertainty with three-dimensional poses for use in estimation problems. *IEEE Transactions on Robotics*, 30(3):679–693.

Barrau, A. and Bonnabel, S. (2015). An ekf-slam algorithm with consistency properties. *arXiv preprint: 1510.06263*.

Barrau, A. and Bonnabel, S. (2016). The invariant extended kalman filter as a stable observer. *IEEE Transactions on Automatic Control*, 62(4):1797–1812.

Barrau, A. and Bonnabel, S. (2018). Invariant kalman filtering. *Annual Review of Control, Robotics, and Autonomous Systems*, 1(1):237–257.

Barrau, A. and Bonnabel, S. (2022). The geometry of navigation problems. *IEEE Transactions on Automatic Control*, 68(2):689–704.

Bellantoni, J. and Dodge, K. (1967). A square root formulation of the kalman-schmidt filter. *AIAA journal*, 5(7):1309–1314.

Berens, P. (2009). CircStat: A MATLAB Toolbox for Circular Statistics. *Journal of Statistical Software*, 31:1–21.

Bingham, C. (1974). An antipodally symmetric distribution on the sphere. *The Annals of Statistics*, pages 1201–1225.

Bonnabel, S. (2007a). *Invariant asymptotic observers: theory and examples*. phdthesis, École Nationale Supérieure des Mines de Paris.

Bonnabel, S. (2007b). Left-invariant extended kalman filter and attitude estimation. In *2007 46th IEEE Conference on Decision and Control*, pages 1027–1032. IEEE.

Bonnabel, S. (2012). Symmetries in observer design: Review of some recent results and applications to ekf-based slam. *Robot Motion and Control 2011*, pages 3–15.

Bonnabel, S., Martin, P., and Rouchon, P. (2008). Symmetry-Preserving Observers. *IEEE Transactions on Automatic Control*, 53(11):2514–2526. Conference Name: IEEE Transactions on Automatic Control.

Bouazza, T., Van Goor, P., Mahony, R., and Hamel, T. (2023). Nonlinear constructive observer design for direct homography estimation. *IFAC-PapersOnLine*, 56(2):1655–1660.

Bourmaud, G., Mégret, R., Arnaudon, M., and Giremus, A. (2015). Continuous-Discrete Extended Kalman Filter on Matrix Lie Groups Using Concentrated Gaussian Distributions. *Journal of Mathematical Imaging and Vision*, 51(1):209–228.

Bourmaud, G., Mégret, R., Giremus, A., and Berthoumieu, Y. (2013). Discrete Extended Kalman Filter on Lie groups. In *21st European Signal Processing Conference (EUSIPCO 2013)*, pages 1–5. ISSN: 2076-1465.

Brossard, M. (2020). *Deep learning, inertial measurements units, and odometry : some modern prototyping techniques for navigation based on multi-sensor fusion*. Phd thesis, Université Paris sciences et lettres.

Brossard, M., Barrau, A., and Bonnabel, S. (2020a). AI-IMU Dead-Reckoning. *IEEE Transactions on Intelligent Vehicles*, 5(4):585–595.

Brossard, M., Barrau, A., and Bonnabel, S. (2020b). A Code for Unscented Kalman Filtering on Manifolds (UKF-M). In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5701–5708. ISSN: 2577-087X.

Brossard, M., Barrau, A., Chauchat, P., and Bonnabel, S. (2022). Associating Uncertainty to Extended Poses for on Lie Group IMU Preintegration With Rotating Earth. *IEEE Transactions on Robotics*, 38(2):998–1015.

Brossard, M., Bonnabel, S., and Barrau, A. (2018). Invariant Kalman Filtering for Visual Inertial SLAM. In *2018 21st International Conference on Information Fusion (FUSION)*, pages 2021–2028.

Brossard, M., Bonnabel, S., and Barrau, A. (2020c). A New Approach to 3D ICP Covariance Estimation. *IEEE Robotics and Automation Letters*, 5(2):744–751.

Brossard, M., Bonnabel, S., and Condomines, J.-P. (2017). Unscented Kalman filtering on Lie groups. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2485–2491.

Bucy, R. (1970). Linear and nonlinear filtering. *Proceedings of the IEEE*, 58(6):854–864.

Burri, M., Nikolic, J., Gohl, P., Schneider, T., Rehder, J., Omari, S., Achtelik, M. W., and Siegwart, R. (2016). The EuRoC micro aerial vehicle datasets. *The International Journal of Robotics Research*, 35(10):1157–1163.

Campbell, J. E. (1897). On a law of combination of operators (second paper). *Proceedings of the London Mathematical Society*, 1(1):14–32.

Chahbazian, C. (2023). *Particle Filtering on Lie Groups : Application to Navigation*. Phd thesis, Université Paris-Saclay.

Chahbazian, C., Dahia, K., Merlinge, N., Winter-Bonnet, B., Blanc, A., and Musso, C. (2024). Recursive posterior Cramér–Rao lower bound on Lie groups. *Automatica*, 160:111422.

Chirikjian, G. S. (2011). *Stochastic models, information theory, and Lie groups, volume 2: Analytic methods and modern applications*, volume 2. Springer Science & Business Media.

Condomines, J.-P., Seren, C., and Hattenberger, G. (2013). Nonlinear state estimation using an invariant unscented kalman filter. In *AIAA Guidance, Navigation, and Control (GNC) Conference*, page 4869.

Condomines, J.-P., Seren, C., and Hattenberger, G. (2014). Pi-Invariant Unscented Kalman Filter for sensor fusion. In *53rd IEEE Conference on Decision and Control*, pages 1035–1040. ISSN: 0191-2216.

DeMars, K., Jah, M., Giza, D., and Kelecy, T. (2009). Orbit determination performance improvements for high area-to-mass ratio space object tracking using an adaptive gaussian mixtures estimation algorithm. In *21st International Symposium on Space Flight Dynamics*.

Dhillon, I. S. and Sra, S. (2003). Modeling data using directional distributions. Technical report, Technical Report TR-03-06, Department of Computer Sciences, The University ….

Doucet, A., Godsill, S., and Andrieu, C. (2000). On sequential monte carlo sampling methods for bayesian filtering. *Statistics and computing*, 10(3):197–208.

Eade, E. (2013). Lie groups for 2d and 3d transformations. *URL http://ethaneade.com/lie.pdf, revised Dec*, 117:118.

Falorsi, L., De Haan, P., Davidson, T. R., and Forré, P. (2019). Reparameterizing distributions on lie groups. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 3244–3253. PMLR.

Giza, D., Singla, P., Crassidis, J., Linares, R., Cefola, P., and Hill, K. (2010). Entropy-based space object data association using an adaptive gaussian sum filter. In *AIAA/AAS Astrodynamics Specialist Conference*, page 7526.

Gordon, N. J., Salmond, D. J., and Smith, A. F. (1993). Novel approach to nonlinear/non-gaussian bayesian state estimation. In *IEE proceedings F (radar and signal processing)*, volume 140, pages 107–113. IET.

Graf, S. and Luschgy, H. (2000). *Foundations of quantization for probability distributions*. Springer Science & Business Media.

Hanebeck, U. D., Huber, M. F., and Klumpp, V. (2009). Dirac mixture approximation of multivariate Gaussian densities. In *Proceedings of the 48h IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference*, pages 3851–3858. ISSN: 0191-2216.

Hartley, R., Jadidi, M. G., Grizzle, J. W., and Eustice, R. M. (2018). Contact-Aided Invariant Extended Kalman Filtering for Legged Robot State Estimation. In *Robotics: Science and Systems XIV*.

Hassan, S., Hussin, A., and Zubairi, Y. (2012). Improved efficient approximation of concentration parameter and confidence interval for circular distribution. *ScienceAsia*, 38(1):118.

Hide, C., Moore, T., and Smith, M. (2003). Adaptive kalman filtering for low-cost ins/gps. *the Journal of Navigation*, 56(1):143–152.

Horwood, J. T., Aragon, N. D., and Poore, A. B. (2011). Gaussian sum filters for space surveillance: theory and simulations. *Journal of Guidance, Control, and Dynamics*, 34(6):1839–1851.

Horwood, J. T. and Poore, A. B. (2014). Gauss von Mises Distribution for Improved Uncertainty Realism in Space Situational Awareness. *SIAM/ASA Journal on Uncertainty Quantification*, 2(1):276–304.

Hua, M.-D., Zamani, M., Trumpf, J., Mahony, R., and Hamel, T. (2011). Observer design on the special euclidean group se(3). In *2011 50th IEEE Conference on Decision and Control and European Control Conference*, pages 8169–8175.

Jaynes, E. (1988). The relation of bayesian and maximum entropy methods. In *Maximum-Entropy and Bayesian Methods in Science and Engineering: Foundations*, pages 25–29. Springer.

Jaynes, E. T. (1968). Prior probabilities. *IEEE Transactions on Systems Science and Cybernetics*, 4(3):227–241.

Jazwinski, A. H. (2007). *Stochastic processes and filtering theory*. Courier Corporation.

Julier, S. (2002). The scaled unscented transformation. In *Proceedings of the 2002 American Control Conference (IEEE Cat. No.CH37301)*, volume 6, pages 4555–4559 vol.6.

Julier, S. (2003). The spherical simplex unscented transformation. In *Proceedings of the 2003 American Control Conference, 2003.*, volume 3, pages 2430–2434 vol.3.

Julier, S., Uhlmann, J., and Durrant-Whyte, H. (1995). A new approach for filtering nonlinear systems. In *Proceedings of 1995 American Control Conference - ACC'95*, volume 3, pages 1628–1632.

Julier, S., Uhlmann, J., and Durrant-Whyte, H. (2000). A new method for the nonlinear transformation of means and covariances in filters and estimators. *IEEE Transactions on Automatic Control*, 45(3):477–482.

Julier, S. J. and Uhlmann, J. K. (1997). New extension of the Kalman filter to nonlinear systems. In *Signal Processing, Sensor Fusion, and Target Recognition VI*, volume 3068, pages 182–193. SPIE.

Julier, S. J. and Uhlmann, J. K. (2002). Reduced sigma point filters for the propagation of means and covariances through nonlinear transformations. In *Proceedings of the 2002 American control conference (IEEE Cat. No. CH37301)*, volume 2, pages 887–892. IEEE.

Kalman, R. E. (1960). A New Approach to Linear Filtering and Prediction Problems. *Journal of Basic Engineering*, 82(1):35–45.

Khosravian, A., Trumpf, J., Mahony, R., and Lageman, C. (2015). Observers for invariant systems on lie groups with biased input measurements and homogeneous outputs. *Automatica*, 55:19–26.

Kurz, G., Gilitschenski, I., and Hanebeck, U. D. (2013). Recursive nonlinear filtering for angular data based on circular distributions. In *2013 American Control Conference*, pages 5439–5445. ISSN: 2378-5861.

Kurz, G., Gilitschenski, I., and Hanebeck, U. D. (2014). The partially wrapped normal distribution for SE(2) estimation. In *2014 International Conference on Multisensor Fusion and Information Integration for Intelligent Systems (MFI)*, pages 1–8.

Labsir, S. (2020). *Méthodes statistiques fondées sur les groupes de Lie pour le suivi d'un amas de débris spatiaux.* Phd thesis, Université de Bordeaux.

Labsir, S., Pages, G., and Vivet, D. (2022). Lie Group Modelling for an EKF-Based Monocular SLAM Algorithm. *Remote Sensing*, 14(3):571.

Lee, T. (2016). Global unscented attitude estimation via the matrix Fisher distributions on SO(3). In *2016 American Control Conference (ACC)*, pages 4942–4947. ISSN: 2378-5861.

Lee, T. (2018). Bayesian Attitude Estimation With the Matrix Fisher Distribution on SO(3). *IEEE Transactions on Automatic Control*, 63(10):3377–3392.

Lee, T.-C., Song, K.-T., Lee, C.-H., and Teng, C.-C. (2001). Tracking control of unicycle-modeled mobile robots using a saturation feedback controller. *IEEE transactions on control systems technology*, 9(2):305–318.

Li, T., Bolic, M., and Djuric, P. M. (2015). Resampling Methods for Particle Filtering: Classification, implementation, and strategies. *IEEE Signal Processing Magazine*, 32(3):70–86.

Liang, C., Xue, W., Fang, H., He, X., and Gupta, V. (2023). On consistency and stability of distributed kalman filter under mismatched noise covariance and uncertain dynamics. *Automatica*, 153.

Lin, T.-Y., Zhang, R., Yu, J., and Ghaffari, M. (2021). Legged robot state estimation using invariant kalman filtering and learned contact events. *arXiv preprint arXiv:2106.15713*.

Loianno, G., Watterson, M., and Kumar, V. (2016). Visual inertial odometry for quadrotors on SE(3). In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1544–1551.

Long, A. W., Wolfe, K. C., Mashner, M. J., Chirikjian, G. S., et al. (2013). The banana distribution is gaussian: A localization study with exponential coordinates. *Robotics: Science and Systems VIII*, 265(1).

Lopez, E., Dahia, K., Merlinge, N., Winter-Bonnet, B., Maschiella, A., and Musso, C. (2024). Sequential markov chain monte carlo methods on matrix lie groups. In *2024 27th International Conference on Information Fusion (FUSION)*, pages 1–7. IEEE.

Mahony, R., Hamel, T., and Pflimlin, J.-M. (2005). Complementary filter design on the special orthogonal group so(3). In *Proceedings of the 44th IEEE Conference on Decision and Control*, pages 1477–1484.

Mahony, R., Hamel, T., and Pflimlin, J.-M. (2008). Nonlinear Complementary Filters on the Special Orthogonal Group. *IEEE Transactions on Automatic Control*, 53(5):1203–1218. Conference Name: IEEE Transactions on Automatic Control.

Mangelson, J. G., Ghaffari, M., Vasudevan, R., and Eustice, R. M. (2020). Characterizing the uncertainty of jointly distributed poses in the lie algebra. *IEEE Transactions on Robotics*, 36(5):1371–1388.

Mardia, K. V., Hughes, G., Taylor, C. C., and Singh, H. (2008). A Multivariate Von Mises Distribution with Applications to Bioinformatics. *The Canadian Journal of Statistics / La Revue Canadienne de Statistique*, 36(1):99–109.

Mardia, K. V. and Jupp, P. E. (2009). *Directional statistics*. John Wiley & Sons.

Marković, I., Ćesić, J., and Petrović, I. (2016). On wrapping the Kalman filter and estimating with the SO(2) group. In *2016 19th International Conference on Information Fusion (FUSION)*, pages 2245–2250.

Masoumnezhad, M., Jamali, A., and Nariman-zadeh, N. (2015). Optimal design of symmetrical/asymmetrical sigma-point Kalman filter using genetic algorithms. *Transactions of the Institute of Measurement and Control*, 37(3):425–432.

Meng, L., Li, L., and Veres, S. (2010). Aerodynamic parameter estimation of an unmanned aerial vehicle based on extended kalman filter and its higher order approach. In *2010 2nd international conference on advanced computer control*, volume 5, pages 526–531. IEEE.

Mohamed, A. and Schwarz, K. (1999). Adaptive kalman filtering for ins/gps. *Journal of geodesy*, 73(4):193–203.

Nielsen, K., Svahn, C., Rodriguez-Deniz, H., and Hendeby, G. (2021). UKF Parameter Tuning for Local Variation Smoothing. In *2021 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, pages 1–8.

Olver, P. J. (1999). *Classical invariant theory*. Number 44. Cambridge University Press.

Pagès, G. and Pham, H. (2005). Optimal quantization methods for nonlinear filtering with discrete-time observations. *Bernoulli Society for Mathematical Statistics and Probability*, 11(5):893–932.

Pagès, G., Pham, H., and Printems, J. (2004). Optimal Quantization Methods and Applications to Numerical Problems in Finance. In *Handbook of Computational and Numerical Methods in Finance*, pages 253–297. Birkhäuser.

Pagès, G. and Printems, J. (2003). Optimal quadratic quantization for numerics: the Gaussian case. *Monte Carlo Methods Appl.*, 9(2):135–165.

Pavliotis, G. A. (2014). Stochastic processes and applications. *Texts in applied mathematics*, 60.

Phogat, K. S. and Chang, D. E. (2020). Invariant extended Kalman filter on matrix Lie groups. *Automatica*, 114:108812.

Pravong, V., Condomines, J.-P., Lundin, G. O., and Puechmorel, S. (2024). Unscented Kalman Filter using Optimal Quantization. In *2024 27th International Conference on Information Fusion (FUSION)*, pages 1–8.

Pravong, V., Condomines, J.-P., Lundin, G. Ö., and Puechmorel, S. (2025a). An equivariant von mises-gaussian distribution on se (n) for unscented kalman filtering.

Pravong, V., Condomines, J.-P., Lundin, G. Ö., and Puechmorel, S. (2025b). An optimal quantization-based unscented kalman filter on se(2): Application to trajectory tracking. In *2025 European Control Conference (ECC)*, pages 2012–2018.

Rhudy, M. (2023). Selection of tuning parameters of the unscented kalman filter using analytical truth statistics. In *AIAA SCITECH 2023 Forum*, page 2702.

Sakai, A. and Kuroda, Y. (2010). Discriminative Parameter Training of Unscented Kalman Filter. *IFAC Proceedings Volumes*, 43(18):677–682.

Scandaroli, G. G., Morin, P., and Silveira, G. (2011). A nonlinear observer approach for concurrent estimation of pose, IMU bias and camera-to-IMU rotation. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3335–3341. ISSN: 2153-0866.

Scardua, L. A. and da Cruz, J. J. (2017). Complete offline tuning of the unscented Kalman filter. *Automatica*, 80:54–61.

Sellami, A. (2005). *Méthodes de quantification optimale pour le filtrage et applications à la finance*. PhD thesis, Université Paris Dauphine.

Shannon, C. E. (1948). A mathematical theory of communication. *The Bell system technical journal*, 27(3):379–423.

Simon, D. (2006). *Optimal state estimation: Kalman, H infinity, and nonlinear approaches*. John Wiley & Sons.

Sjøberg, A. M. and Egeland, O. (2022). Lie Algebraic Unscented Kalman Filter for Pose Estimation. *IEEE Transactions on Automatic Control*, 67(8):4300–4307. Conference Name: IEEE Transactions on Automatic Control.

Smyth, G. K. (1998). Numerical integration. *Encyclopedia of biostatistics*, pages 3088–3095.

Solà, J., Deray, J., and Atchuthan, D. (2021). A micro Lie theory for state estimation in robotics.

Steinbring, J., Pander, M., and Hanebeck, U. D. (2015). The smart sampling kalman filter with symmetric samples. *arXiv preprint:1506.03254*.

Straka, O., Duník, J., and Šimandl, M. (2014). Unscented Kalman filter with advanced adaptation of scaling parameter. *Automatica*, 50(10):2657–2664.

Tan, C., Cai, Y., Wang, H., Sun, X., and Chen, L. (2023). Vehicle state estimation combining physics-informed neural network and unscented kalman filtering on manifolds. *Sensors*, 23(15):6665.

Tao, Z., Xue, J., Wang, D., Li, G., and Fang, J. (2022). An adaptive invariant ekf for map-aided localization using 3d point cloud. *IEEE Transactions on Intelligent Transportation Systems*, 23(12):24057–24070.

Van der Merwe, R. and Wan, E. (2001). The square-root unscented Kalman filter for state and parameter-estimation. In *2001 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No.01CH37221)*, volume 6, pages 3461–3464. ISSN: 1520-6149.

van Goor, P., Hamel, T., and Mahony, R. (2023). Constructive equivariant observer design for inertial velocity-aided attitude. *IFAC-PapersOnLine*, 56(1):349–354.

Van Loan, C. F. and Golub, G. (1996). Matrix computations. *Matrix Computations*, 5:32.

Wan, E. and Van Der Merwe, R. (2000). The unscented Kalman filter for nonlinear estimation. In *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No.00EX373)*, pages 153–158.

Wang, W. and Lee, T. (2021). Bingham-Gaussian Distribution on $\mathbb{S}^3 \times \mathbb{R}^n$ for Unscented Attitude Estimation. In *2021 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, pages 1–7.

Wang, W. and Lee, T. (2022). Matrix fisher–gaussian distribution on SO(3) $\times$ $\mathbb{R}^n$ and bayesian attitude estimation. *IEEE Transactions on Automatic Control*, 67(5):2175–2191.

Wang, Y. and Chirikjian, G. (2006). Error propagation on the Euclidean group with applications to manipulator kinematics. *IEEE Transactions on Robotics*, 22(4):591–602. Conference Name: IEEE Transactions on Robotics.

Wang, Y. and Chirikjian, G. S. (2008). Nonparametric Second-order Theory of Error Propagation on Motion Groups. *The International Journal of Robotics Research*, 27(11-12):1258–1273.

Weisstein, E. W. (20XX). Hermite polynomial. From MathWorld–A Wolfram Resource. https://mathworld.wolfram.com/HermitePolynomial.html.

Wolfe, K., Mashner, M., and Chirikjian, G. (2011). Bayesian Fusion on Lie Groups. *Journal of Algebraic Statistics*, 2.

Wonham, W. M. (1964). Some applications of stochastic differential equations to optimal nonlinear filtering. *Journal of the Society for Industrial and Applied Mathematics, Series A: Control*, 2(3):347–369.

Wu, T.-H., Kaufman, E., and Lee, T. (2015). Globally asymptotically stable attitude observer on so(3). In *2015 54th IEEE Conference on Decision and Control (CDC)*, pages 2164–2168.

Ye, J., Jayaraman, A. S., and Chirikjian, G. S. (2025). Uncertainty Propagation on Unimodular Matrix Lie Groups. *IEEE Transactions on Automatic Control*, 70(4):2450–2465.

Zakai, M. (1969). On the optimal filtering of diffusion processes. *Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete*, 11(3):230–243.

Zeng, N., Wang, Z., Li, Y., Du, M., and Liu, X. (2011). A hybrid ekf and switching pso algorithm for joint state and parameter estimation of lateral flow immunoassay models. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 9(2):321–329.

Štecha, J. and Havlena, V. (2012). Unscented Kalman filter revisited — Hermite-Gauss Quadrature approach. In *2012 15th International Conference on Information Fusion*, pages 495–502.

RÉPUBLIQUE FRANÇAISE
Liberté
Égalité
Fraternité

Université de Toulouse

Titre : Estimation d'état sur groupes de Lie pour les systèmes robotiques sous incertitudes

Mots clés : Estimation d'état, Robotique mobile, Navigation, Filtre de Kalman, Observateurs invariants, Densité de probabilité

Résumé : Dans le contexte de la navigation robotique terrestre et/ou aérienne, les différents algorithmes de guidage et de contrôle doivent être robustes face aux situations d'aléas, qu'il s'agisse de pannes (capteurs et/ou actionneurs) ou d'événements exogènes tels que des perturbations aérologiques. Si une solution possible consiste à rendre ces algorithmes insensibles à de tels phénomènes, par exemple en s'orientant vers la synthèse de stratégies de commande robustes, nous nous focalisons dans cette thèse à fournir une estimation fiable de l'état de l'engin, afin que des méthodes de commande de type « retour d'état » puissent, elles aussi, exploiter les informations les plus exactes possible sur l'état courant du système robotique.

Le travail de thèse porte ainsi sur le développement de techniques d'estimation de pose (attitude et position) appliquées à des systèmes de robotique mobile, sur la base de mesures bruitées et imparfaites. Pour cela, les méthodes usuelles de filtrage, comme le filtrage Bayésien (plus exactement les filtres de Kalman) constituent la base méthodologique de cette étude. Néanmoins, ces méthodes souffrent d'imprécisions et d'un manque de cohérence lorsque les modèles sous-jacents sont imprécis et/ou incomplets, ou lorsque les erreurs et pannes des capteurs/actionneurs sont difficilement modélisables. Dans la grande majorité des cas, les approches classiques de la littérature reposent sur l'hypothèse de variables aléatoires gaussiennes pour caractériser les incertitudes agissant sur le système, permettant ainsi de pondérer la correction de l'état estimé. Or, si cette hypothèse est commode pour diverses raisons, elle peut s'effondrer dans de nombreux cas d'application.

En estimation appliquée à la navigation, les avancées récentes ont davantage porté sur une redéfinition de l'état sur des variétés non euclidiennes telles que les groupes de Lie, comme SO(3) ou SE(3), afin de prendre en compte la géométrie sous-jacente des systèmes de navigation. Sur ces espaces, la caractérisation de l'incertitude constitue un élément crucial pour la synthèse de filtres. L'objectif de nos travaux de recherche est ainsi de proposer une méthode de filtrage qui caractérise l'incertitude de l'état de manière plus adéquate, afin d'améliorer les performances de filtres déjà pré-établis.

Title: State estimation on Lie groups for robotic systems under uncertainties

Key words: State estimation, Mobile robot, Navigation, Kalman filter, Symmetry-preserving observers, Probability density

Abstract: In the context of terrestrial and aerial robotic navigation, guidance and control algorithms must be robust to unexpected events, whether for sensor/actuator failures, or exogenous disturbances such as aerodynamic perturbations. While one possible solution is to make these algorithms insensitive to such phenomena, for instance by designing robust control strategies, this thesis focuses instead on providing a reliable estimate of the vehicle's state, so that state-feedback control methods can also benefit from the most qualitative informations about the system's current state.

This thesis therefore concerns the development of pose estimation techniques (attitude and position) for mobile robotic systems, based on noisy and imperfect measurements. Standard filtering methods, such as Bayesian filtering, and more specifically Kalman filters, form the methodological framework of this work. However, these methods suffer from inaccuracies and a lack of consistency when the underlying models are imprecise and/or incomplete, or when sensor and actuator faults are difficult to model. In most cases, classical approaches in the literature rely on the assumption of Gaussian random variables to characterize all sources of uncertainty acting on the system, thereby enabling the weighting of the state correction. Yet, although convenient for several reasons, this assumption often breaks down in practical applications.

In navigation-oriented estimation, recent advances have increasingly focused on redefining the state on non-Euclidean manifolds such as Lie groups, including SO(3) and SE(3), in order to account for the underlying geometry of navigation systems. On such spaces, uncertainty characterization is a key ingredient for the design of filters. The objective of this research work is therefore to propose a filtering method that models state uncertainty in a more suitable manner, with the aim of improving the performance of existing filtering schemes.