

Gait Orientation Estimation and Control of Lower-limb Exoskeletons for Walking Assistance

by

Shilei Li

A Thesis Submitted to
The Hong Kong University of Science and Technology
in Partial Fulfillment of the Requirements for
the Degree of Doctor of Philosophy
in the Department of Electronic and Computer Engineering

August 2022, Hong Kong

Authorization

I hereby declare that I am the sole author of the thesis.

I authorize the Hong Kong University of Science and Technology to lend this thesis to other institutions or individuals for the purpose of scholarly research.

I further authorize the Hong Kong University of Science and Technology to reproduce the thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

Shilei Li

August 2022

Gait Orientation Estimation and Control of Lower-limb Exoskeletons for Walking Assistance

by

Shilei Li

This is to certify that I have examined the above PhD thesis
and have found that it is complete and satisfactory in all respects,
and that any and all revisions required by
the thesis examination committee have been made.

Prof. Ling SHI, Thesis Supervisor
Department of Electronic and Computer Engineering

Prof. Andrew POON, Head of Department
Department of Electronic and Computer Engineering

Thesis Examination Committee

1. Prof. Gary Shueng Han CHAN Department of Electronic and Computer Engineering
(Chairman)
2. Prof. Ling SHI (Supervisor) Department of Electronic and Computer Engineering
3. Prof. Chik Patrick YUE Department of Electronic and Computer Engineering
4. Prof. Qifeng CHEN Department of Electronic and Computer Engineering
5. Prof. Francesco CIUCCI Department of Mechanical and Aerospace Engineering
6. Prof. Chenglong FU Mechanical and Energy Engineering
(External Examiner) Southern University of Science and Technology, China

Department of Electronic and Computer Engineering
The Hong Kong University of Science and Technology

August 2022

Acknowledgments

I would like to express my considerable gratitude to many people with whom I have interacted during my Ph.D. career. Without their kind help and support, I cannot complete my thesis.

Foremost, I would like to express my appreciation to my supervisor, Prof. Ling Shi. I learned a lot from him: enthusiasm for research, optimistic attitude towards life, and carefulness and thoroughness. He gave me substantial support at the beginning of my research, and we kept a regular research-related meeting every two weeks. He also enjoyed listening to my academic ideas and gave his suggestions. Moreover, he liked to share his philosophical point of view and recommended some valuable books to me, which made him both a tutor and a friend to me.

Then, my heartfelt thanks go to all my co-authors: Prof. Dawei Shi, Dr. Peihu Duan, Dr. Pu Duan, Wulin Zou, and Lijing Li. Many good ideas were generated after talking with them. Meanwhile, they helped me a lot in refining my research results and improving the quality of the manuscript.

Further, I would like to thank my thesis committee members, Prof. Gary Shueng Han CHAN, Prof. Chik Patrick YUE, Prof. Francesco CIUCCI, Prof. Qifeng CHEN, and Prof. Chenlong FU. Thank you for your valuable time and helpful comments.

My sincere thanks go to all the current group members in Shi-group: Mengyu Huang, Fan Zhang, Wulin Zou, Xiaomeng Chen, Yuxing Zhong, Andy Yung Yu Yiu, Huiwen Yang, Wei Huo, Nachuan Yang, Ruiqi Zeng, Xingzhou Chen, Zikai Wang, Anshuman Medhi, Yan Ning, Emmett Yim Ying Hing, Fuhai Nan. I also like to express my gratitude to many previous group members. Special thanks to my colleagues when I took an internship at Xeno Dynamics: Pu Duan, Yawen Chen, Zhe Liang, and Ren Dong. Thank you all for the valuable discussions about my research topic we have had.

Last but not least, I would like to give my deepest thanks to my family and girlfriend. Thank you for your endless support and unconditional love.

Contents

Title Page	i
Authorization Page	ii
Signature Page	iii
Acknowledgments	iv
Table of Contents	v
List of Figures	ix
List of Tables	xv
Abstract	xviii
1 Overview	1
1.1 Literature Review	3
1.1.1 Exoskeletons for Gait Rehabilitation and Walking Assistance	4
1.1.2 Correntropy-based Kalman Filter	5
1.1.3 Orientation Estimation of IMUs	7
1.1.4 Disturbance Observer	8
1.1.5 Adaptive Oscillators and Preference-based Control	10
1.2 Thesis Outline and Contributions	11
1.3 List of Publications	17
2 Multi-kernel Maximum Correntropy Kalman Filter	18
2.1 Problem Formulation and Preliminaries	18
2.1.1 Kalman Filter	18
2.1.2 Problem Formulation	19

2.2	Multi-kernel Correntropy	20
2.2.1	Properties of the Multi-kernel Correntropy	21
2.2.2	A Robust Cost for the Kalman Filter	23
2.3	Algorithm Derivation	25
2.4	Disturbance Mitigation	29
2.4.1	Systems with Process Disturbance	30
2.4.2	Systems with Measurement Disturbance	31
2.5	Simulations	31
2.5.1	Example 1	31
2.5.2	Example 2	32
2.5.3	Example 3	36
2.6	Summary	37
2.7	Appendix	37
2.7.1	Proof of Theorem 2.1	37
2.7.2	Proof of Theorem 2.2	38
2.7.3	Proof of Theorem 2.3	39
3	Gait Orientation Estimation	42
3.1	Preliminaries	42
3.1.1	Sensor Models	43
3.1.2	ESKF for Six-axis IMUs and Nine-axis IMUs	44
3.1.3	Effects of External Acceleration and Magnetic Disturbance	46
3.1.4	Bayesian Optimization	47
3.2	Methodology for Six-axis IMUs	48
3.2.1	The Compact Multi-kernel Maximum Correntropy Kalman Filter	49
3.2.2	The Compact Multi-kernel Maximum Correntropy-Based Error State Kalman Filter	49
3.3	Simulations and Experiments for Six-axis IMUs	51
3.3.1	Influence of External Accelerations in Simulation	51
3.3.2	Dynamic Impact Test	53
3.3.3	External Acceleration Test	53
3.3.4	Exoskeleton Orientation Estimation	55
3.4	Methodology for Nine-axis IMUs	56

3.4.1	MKMCKF-OE	57
3.4.2	Kernel Bandwidths Optimization	57
3.5	Experiments	59
3.5.1	Data Collection	59
3.5.2	Kernel Bandwidths Optimization	61
3.5.3	Results	61
3.6	Performance Validation on An Embedded System	63
3.7	Summary	66
3.8	Appendix	67
3.8.1	ESKF for Six-axis IMUs	67
3.8.2	ESKF for Nine-axis IMUs	68
4	Disturbance Observer Using the Generalized Multi-kernel Correntropy	71
4.1	Generalized Multi-kernel Correntropy	71
4.1.1	Problem Formulation	71
4.1.2	Generalized Multi-kernel Correntropy	73
4.1.3	Properties of the Generalized Multi-kernel Correntropy	74
4.1.4	Influence Function of the LMP and GL	75
4.1.5	Relationship with the Kalman Filter	78
4.2	Algorithm Derivation for Disturbance Estimation	80
4.2.1	Algorithm Derivation	80
4.2.2	Disturbance Estimation	82
4.3	Simulations	83
4.3.1	System Modeling	83
4.3.2	Laplace Distribution with Unknown Disturbance	86
4.3.3	Gaussian Distribution with Unknown Disturbance	88
4.4	Summary	88
4.5	Appendix	89
4.5.1	Proof of Theorem 4.1	89
4.5.2	Proof of Theorem 4.2	89
4.5.3	Proof of Theorem 4.3	89
4.5.4	Proof of Theorem 4.4	90
4.5.5	Proof of Theorem 4.5	91

4.5.6 Proof of Theorem 4.6	91
5 Preference-based Assistance Map Learning with Robust Adaptive Oscillators	92
5.1 Materials and Methods	92
5.1.1 Approach Overview	92
5.1.2 Robust Adaptive Oscillators	94
5.1.3 Exoskeleton Controller	97
5.1.4 Assistance Map Learning	98
5.2 Experiments	100
5.2.1 Robust Adaptive Oscillators	100
5.2.2 Torque Controller	102
5.2.3 Parameter Map Learning	103
5.2.4 Investigation of Muscle Activities	106
5.3 Summary	110
6 Conclusion	111
6.1 Concluding Remarks	111
6.2 Future Work	113
Reference	114

List of Figures

1.1	Treadmill-based exoskeletons. (a) Lokomat. (b) LOPES. (c) ALEX.	4
1.2	Ground-based exoskeletons. (a) Indego. (b) HAL. (c) USTC Exoskeleton.	4
1.3	Exoskeletons that have broken “metabolic barrier”. (a) CMU unpowered ankle exoskeleton. (b) Bent-leaf-spring (BLS) unpowered hip exoskeleton. (c) CMU powered ankle exoskeleton. (d) Samsung hip exoskeleton. (e) Harvard hip exosuit. (f) Harvard ankle exosuit.	5
1.4	Exoskeletons for elder persons or post-stroke patients. (a) Harvard portable exosuit. (b) Curara. (c) GEMS-L.	6
2.1	Approximating a mixture Gaussian distribution using a single Gaussian distribution. The target distribution $f(e)$ follows $0.5\mathcal{N}(0, 1) + 0.5\mathcal{N}(0, 100)$. The estimated distribution is obtained by minimizing the mean squared error $E[(f(e) - \hat{f}(e))^2]$.	20
2.2	Contours of $CIM(\mathcal{X}, 0)$ in 2D space with different kernel bandwidths.	22
2.3	Objective function and influence function of the MSE and the MKCL in one dimensional case.	25
2.4	Probability densities of the estimation error using the KF, MCKF, MKMCKF1, MKMCKF2 and PF. The kernel bandwidth for the MCKF is $\sigma = 40$. The bandwidths for MKMCKF1 are $\sigma_p = [1.2, 0.5]^T$ and $\sigma_r = 10^4$, and the bandwidths for MKMCKF2 are $\sigma_p = [0.5, 1.2]^T$ and $\sigma_r = 10^4$.	33

2.5	Error performances of the ASKF, MCKF, MKMCKF1, MKMCKF2 and PF. The white region is the error without process disturbance while the blue region is the error with process disturbance. The kernel bandwidth for MCKF is $\sigma = 100$ while the bandwidths for MKMCKF1 are $\sigma_p = [1.5, 10^4]^T$ and $\sigma_r = 10^4$, and the bandwidths for MKMCKF2 are $\sigma_p = [10^4, 1.5]^T$ and $\sigma_r = 10^4$. The zoom-in plot at time intervals 20 s to 30 s shows that the performances of the ASKF, MCKF, MKMCKF1 and MKMCKF2 are similar without process disturbance.	35
2.6	Error performances of the ASKF, MCKF, MKMCKF, and PF with measurement disturbance. The kernel bandwidth for MCKF is $\sigma = 5$ while the bandwidths for MKMCKF are $\sigma_p = [10^4, 5]^T$ and $\sigma_r = 10^4$. The zoom-in plot shows that the performances of the ASKF, MCKF, and MKMCKF are similar without disturbance.	37
3.1	The gyroscope signals and accelerometer signals in simulation. The impulsive acceleration is added at 1.2 s while the sinusoidal external acceleration is added at the interval of 4.5 s to 8.0 s.	52
3.2	Performances of the different algorithms with impulsive and sinusoidal accelerations. The close-up of the pitch error is shown on the left top of the figure.	52
3.3	The experimental setup of dynamic impact test and the sensor signals.	53
3.4	The rotational angle obtained by an encoder and the angle errors of the GD, ESKF, and CMKMC-ESKF.	54
3.5	The experimental setup of external acceleration test and sensor signals.	54
3.6	Angle errors and position errors of the ESKF and CMKMC-ESKF with external accelerations.	55
3.7	The experimental setup and sensor signals with $f = 0.8$ Hz.	56
3.8	The encoder angle and the corresponding errors of the GD, ESKF, and CMKMC-ESKF with $f = 0.8$ Hz.	57

3.9	The experimental setup and the IMU coordinate system. The movement can be divided into the linear motion in Figure 3.9(a) and the rotation in Figure 3.9(b). The acceleration disturbance A_d is generated by the movement of Xsens itself and the magnetic disturbance M_d is generated by the pliers.	60
3.10	Norms of the accelerometer and magnetometer readings in experiment 8.	62
3.11	Errors of the GD, ESKF, MKMCKF-OE, and IGD in experiment 8.	63
3.12	Norms of the accelerometer and magnetometer readings in experiment 11.	63
3.13	Estimation errors of the GD, ESKF, MKMCKF-OE, and IGD in experiment 11.	64
3.14	Experimental setup. The IMU is installed on the heel of a shoe, which is shown in Figure 3.14(a). The subject walks on designated path which is shown in Figure 3.14(b) and Table 3.6.	64
3.15	Ground truth trajectory and orientation of the foot in experiment 1.	65
3.16	Orientation estimation errors of the MKMCKF-OE, GD, ESKF, and IGD in experiment 1.	65
4.1	Approximating a ε -contaminated mixture model using a Gaussian distribution. The target distribution follows $0.37\mathcal{U}(-5, 5) + 0.63\mathcal{N}(0, 0.5)$. The estimated distribution is obtained by minimizing the mean squared error $E[(f(w_{d,k}) - \hat{f}(w_{d,k}))^2]$.	73
4.2	Contours of $J_{GL}(\mathcal{X}, 0)^{\frac{1}{\alpha}}$ in 2D space with different shape parameters and different bandwidths.	76
4.3	Plots of the J_{LMP} , J_{GL} , ∇J_{LMP} , and ∇J_{GL} in one dimensional case.	78
4.4	Laplace distribution, Gaussian distribution, $p(e)$ in (4.18), and $p(w_{d,k})$ using a ε -contaminated mixture model. The error domain is set to be $\mathcal{E} = [-5, 5]$ in $p(e)$. The noise distribution $p_L(w_{d,k})$ follows $0.2\mathcal{L}(0, 1) + 0.8\mathcal{U}(-5, 5)$ while $p_G(w_{d,k})$ follows $0.2\mathcal{N}(0, 0.5) + 0.8\mathcal{U}(-5, 5)$.	80
4.5	The exoskeleton.	84
4.6	Disturbance errors of different observers. The step-like disturbance is added at $t = 4$ s and disappears at $t = 6$ s.	86
4.7	Tracking errors of different observers.	87

- 5.1 Front and rear view of the hip exoskeleton. An elastic fasten belt is used to avoid the exoskeleton waist flip. 93
- 5.2 Overview of the control architecture. First, gait is processed by a robust adaptive oscillator and its output is gait cycle percentage c and gait features g . Then, the desired torque is generated based on c and its shape is controlled by a human-machine interface. After that, the exoskeleton carries out the command and assists the subject. In this process, the user can adjust the assistance profile and submit his (or her) satisfied parameters based on their proprioceptive feedback. A Gaussian process regression (GPR) module is employed to predict the user's preference and to map the gait features g to a parameter set Ψ . After the learning procedure, the exoskeleton can automatically provide suitable assistance based on the user's gait feature without the need of manipulating the human-robot interface. 94
- 5.3 A graphic illustration of k_w . The black solid dots represent the zero-cross point of q , θ_r and θ_l are the right and left hip angle, ξ_w represents the amplitude, and d_w represents the correction region. 96
- 5.4 Desired torque generator. The profile is controlled by both the speed-independent parameters (i.e., $offset$, ext_{dur}) and the speed-dependent parameters (i.e., ext_{peak} , ext_{max} , fle_{peak} , and fle_{max}). The speed-independent parameters are determined before the training session. At the training session, users select and submit their preferred speed-dependent parameters when walking at different speeds. 97

- 5.5 Performances comparison of the RAO and AO. The subject walks with a hip exoskeleton in three modes: steady walking, speed up, and speed down. The RAO and AO have similar performance with steady walking, while RAO's performance is better than AO with gait transition, such as speed up or speed down. The symbol q , w_{real} , p_{real} represents the input signal, the ground truth frequency, and the ground truth phase, respectively. The subscript rao and ao denotes the corresponding estimates from the RAO and AO, respectively. A close-up with a speed-up gait is shown on the right. It can be seen that the frequency estimation of the RAO converges to the real frequency quickly. Moreover, its phase is aligned with the zero phases automatically. 101
- 5.6 Error performances of the RAO and AO. Three indices including tracking error, frequency error, and phase error are involved in the figure. The red lines represent the performance of the RAO while the dashed black lines are the errors of the AO. 102
- 5.7 Performances of three torque controllers. The signals q are shown on the top while the corresponding torque tracking performances are shown at the bottom. 103
- 5.8 Experimental setup and the interactive screen. Six parameters are involved in the screen. Among them, the offset $offset$ and the extension duration ext_{dur} are determined in the warm-up session and remain unchanged in the following session, while the other parameters (ext_{max} , ext_{peak} , fle_{max} , fle_{peak}) are tuned in the training session. 104
- 5.9 Learned assistance maps for S1, S2, and S3. The stars * represent the user-selected parameters, the predicted values are drawn by mesh figures. 105
- 5.10 Learned assistance maps for S7. The star * represents the user-selected parameters, the predicted values is drawn by mesh figures. 105
- 5.11 Experimental scenario and sEMG sensor placement. Three muscles are involved in this experiment, which are RF, TA, and m.GAS. The desired torque is generated based on the user's selection (or preference). 107
- 5.12 The desired torque and actual torque for 7 participants when walking at 5 km/h using the user-preferred parameters. 107

- 5.13 Muscle activities of seven subjects in three conditions. The amplitudes are normalized to the data in the NW condition. Subjects walk on the treadmill at a speed of 5 km/h. The assistance profile is given by the user's manual selection (or preference). 108
- 5.14 Normalized muscle activities and their changes in the assistance mode compared to ZT mode and NW mode. 109

List of Tables

2.1	Performances of the KF, MCKF, MKMCKF1, MKMCKF2 and PF with non-Gaussian Process Noises.	33
2.2	Performances of the ASKF, MCKF, MKMCKF1, MKMCKF2 and PF with Process Disturbance.	35
2.3	Performances of the ASKF, MCKF, MKMCKF and PF with Measurement Disturbance.	36
3.1	Error Performances of Different Algorithms with Impulsive Accelerations.	53
3.2	Error Performances of Different Algorithms with Different Gait Frequencies.	56
3.3	Experimental Description.	60
3.4	RMSEs of Different Algorithms on the Roll and Pitch.	61
3.5	RMSEs of Different Algorithms on the Yaw.	62
3.6	Experimental Description Using Self-designed IMUs.	65
3.7	Performances of Different Algorithms on the Yaw.	66
3.8	Performances of Different Algorithms on the Pitch and Roll.	66
4.1	Performances of Different Observers with Laplace Noises.	87
4.2	Performances of Different Observers with Gaussian Noises.	88
5.1	Performances of the RAO and AO.	101
5.2	Performances of Different Torque Controllers.	103
5.3	Learning Efficiency Comparisons with Different Methods.	106
5.4	Muscle Activity Comparisons.	109

Acronyms and Notations

Acronyms

CMKMCKF	Compact Multi-kernel Maximum Correntropy Kalman Filter
ESKF	Error State Kalman Filter
GMKMCKF	Generalized Multi-kernel Maximum Correntropy Kalman Filter
KF	Kalman Filter
MCC	Maximum Correntropy Criterion
MCKF	Maximum Correntropy Kalman Filter
MKC	Multi-kernel Correntropy
MKCL	Multi-kernel Correntropy Loss
MKMCKF	Multi-kernel Maximum Correntropy Kalman Filter
MSE	Mean Squared Error

Notations

\mathbb{R}^m	Set of m -dimensional vectors
$\mathbb{R}^{n \times m}$	Set of matrices with n rows and m columns
$\mathcal{N}(\mu, \Sigma)$	Gaussian distribution with mean μ and covariance Σ
\triangleq	Equality by definition

x^+ The *a posteriori* estimate of vector x

x^- The *a priori* estimate of vector x

X^T Transpose of matrix X

Gait Orientation Estimation and Control of Lower-limb Exoskeletons for Walking Assistance

by

Shilei Li

Department of Electronic and Computer Engineering
The Hong Kong University of Science and Technology

ABSTRACT

Recently, lower-limb exoskeletons have demonstrated the ability to enhance human mobility and walking efficiency for both healthy subjects and patients. However, this technology is confined to the laboratory and its performance is unsatisfactory in a community environment due to: 1) as important components of the exoskeletons, the inertial measurement units (IMUs) are vulnerable to external acceleration and magnetic disturbance. 2) the control performance is significantly affected by external disturbance. 3) the user's preference or the individualized walking coordination strategy is rarely considered in the assistance map construction.

This thesis aims to cope with the aforementioned issues. Specifically, the first two issues can be regarded as performance degeneration of the state estimation with some channels containing non-Gaussian noises. In the conventional estimation framework, the mean squared error (MSE) has been widely used as a cost function due to its features of smoothness, convexity, and mathematical tractability. However, the optimality of the MSE is rooted in the Gaussian assumption, and its performance may be unsatisfactory with heavy-tailed noises. Actually, the correntropy is a better metric for non-Gaussian noises since it captures high-order error statistics. However, it is defined for random variables and is incapable of systems with partial non-Gaussian noises. In this thesis, we introduce a multi-kernel correntropy (MKC) which extends the definition of correntropy from random variables to random vectors. Some important properties of the MKC are given, and a multi-kernel correntropy Kalman filter (MKMCKF) is derived based on the MKC. The proposed estimator is robust against non-Gaussian noises and maintains good

performance under Gaussian noises. Simulations verify the effectiveness of the proposed method.

We further apply the MKMCKF to orientation estimation of IMUs and the disturbance estimation of an exoskeleton. Specifically, for six-axis IMUs, we derive a compact multi-kernel maximum correntropy Kalman filter (CMKMCKF) which has a minimal parameter number and has less computation penalty. For nine-axis IMUs, we tune the kernel bandwidths of the MKMCKF using Bayesian optimization. The proposed algorithms are compared with the benchmark methods. Simulations and experiments verify the effectiveness of the proposed algorithms, especially with external acceleration and magnetic disturbance. The MKC is further extended to a generalized multi-kernel correntropy (GMKC) under generalized Gaussian kernels. Comprehensive properties of the GMKC are given and the corresponding generalized multi-kernel correntropy loss (GMKCL) is introduced, which is proven to be more versatile than the traditional least mean p power (LMP) criterion. We reveal that the GMKCL is associated with a certain class of heavy-tailed distributions and is an optimal cost function based on the maximum *a posteriori* probability under some assumptions. Then, a new filter named the generalized multi-kernel maximum correntropy Kalman filter (GMKMCKF) is derived under the GMKCL, and it is utilized as a disturbance observer for a target tracking task using exoskeletons. Simulations show that the proposed disturbance observer outperforms the existing approaches.

To involve the user's preference in the control of the exoskeleton, a robust adaptive oscillator (RAO) is designed to estimate the gait phase and extract gait features. Meanwhile, the participant's preferred assistance parameters and gait features are collected and stored. Then, the Gaussian process regression (GPR) is employed to construct the individualized assistance map based on the historical data. The effectiveness of the proposed method is validated by a hip exoskeleton at a speed of 5 km/h with 7 participants. Three muscles which include rectus femoris, tibialis anterior, and medial gastrocnemius are investigated in three conditions: user-preferred assistance (ASS), zero torque (ZT), and normal walking (NW). Results show that all muscles achieve an activity reduction in the ASS mode compared with the ZT or NW. Meanwhile, there is a statistically significant difference in medial gastrocnemius in the ASS mode with respect to the ZT and NW ($-15.63 \pm 6.51\%$ and $-8.73 \pm 6.40\%$, respectively).

Chapter 1

Overview

Globally, there are 703 million persons aged 65 or over which accounts for 9 percent of the whole population in 2019. This number is projected to be 1.3 billion with the proportion of 16 percent in 2050, so that one in six will be the elderly [1]. The aging society has been a major social issue for many countries or regions, and the number of persons who suffer from mobility dysfunction due to frailty-sarcopenia syndrome is increasing rapidly. The gait of the elderly is characterized by slow speed, small step length, shuffling, decreased arm swing [2], and expensive transport economy [3]. These deficits largely restrain their travel distance, induce fatigue, decrease community involvement, and influence their independence. Stroke is another major reason for mobility impairment. Every year, there are 13.7 million new stroke cases globally [4], and about 2 million new cases in China [5]. The elderly or stroke patients usually have impaired mobility function, accompanied by limited walking speed and distance, reduced endurance and fitness reserve, and a higher risk of falls [6–10]. These gait deficits are observed even in many sedentary older adults [11, 12], which hinders their community reintegration and social activity participation, ultimately contributing to reduced health and life quality [13–15].

Regular daily walking not only contributes to the community involvement of the elderly and the stroke patients but also is beneficial to the prevention of cardiovascular disease, cancer, and heart disease [16, 17]. Crutches, rollators, and orthosis [18] are common devices for the elderly and stroke patients to increase their walking stability. However, these equipment cannot improve their walking economy. Physical therapy can significantly improve stroke patients' motor function, reduce complications, and facilitate their mobility rehabilitation [5]. However, a patient may need several experienced therapists for training, resulting in expensive medical costs and putting a lot of pressure on the public medical system.

The exoskeleton is a promising tool to improve the gait economy and restore the normal gait. Many exoskeletons have been utilized for gait training and rehabilitation with great effectiveness, such as Lokomat [19], LOPES [20], ALEX [21], Indego [22], HAL [23], and USTC [24] exoskeleton. These devices free therapists from repetitive and labor-demanding work, and guide the walking of patients by designing spatiotemporal reference trajectories [25], constructing torque fields or velocity fields [26, 27], or employing the concept of assist-as-needed [28]. Recently, some exoskeletons [29–36] have demonstrated the ability of improving the walking efficiency (reducing the metabolism or muscle activity) using both passive [32, 33] and active devices [29–31, 34–36]. Although the passive device has less mass penalty, the powered one has the potential to provide a versatile force profile and has the ability to accommodate complex environments. Hence, it is much more popular [29–31, 34–36]. Some impressive progress has been achieved by the human-in-the-loop (HIL) or parameter sweeping strategy in a laboratory condition, with 14.79% [35], 17.4% [30], 9.3% [36] net metabolic reduction using a soft exosuit, and 24.2% metabolic reduction [29] and 33.6% muscle activity reduction using an ankle exoskeleton [31].

There is a trend toward designing lightweight exoskeletons to assist the walking of the elderly or stroke patients in a community environment, such as the Harvard portable exosuit [34], SMA [37], H2 [38], GEAR [39], CURARA [40], and GEMS-L [41]. To synchronize the human-robot movement, a real-time gait phase estimator is required. It was obtained by interpolating the gait events (e.g., the heel strike moment or the maximum hip flexion angle moment), employing the periodical signals of gyroscopes [13] or IMUs [35], and using adaptive oscillators (AO) [42–44]. Then, a predefined force profile or an assistance map is constructed based on the gait phase with the purpose of restoring normal gait or improving walking economy [13, 34, 41, 42, 44].

There are three challenges of utilizing the exoskeleton in a complex community environment. The first one is that the sensors deployed for the exoskeletons are largely limited. Some equipment is not accessible when walking in a community, such as foot force plates, optical motion capture systems, indirect calorimetry systems, treadmills, and many others. Only a small number of sensors can be installed on the exoskeletons (e.g., encoders, torque sensors, and IMUs), which can only capture a part of the human-robot information. Even worse, IMUs, as important components of the exoskeletons that are

used for gait orientation estimation [26, 27, 34, 41, 45] or gait assessment [46, 47], are vulnerable to external acceleration and magnetic disturbance [48–51]. These deficits significantly reduce their reliability and have adverse effects on exoskeleton performance. The second challenge is the performance degeneration caused by unknown disturbance. Disturbance widely exists in mechanical systems [52–54] due to unknown friction, unmodeled system dynamics, or abrupt forces from the environment. It deteriorates the controller’s performance significantly and even induces instability. Therefore, it is necessary to design advanced disturbance observer to reject the disturbance and retain the controller’s nominal performance. The final challenge is individualized assistance map construction. This challenge comes from two aspects. Firstly, it is difficult to synchronize the human-exoskeleton movement in a community, where the step length and step time may change dramatically to accommodate the complex environment. Secondly, the popular HIL or parameter sweeping methods [29–31, 33–36] are not applicable to the community assistance, and the empirical assistance map construction ignores the user’s individualized coordination strategy and contributes to the uncomfortable feeling.

This thesis aims to cope with the aforementioned issues. Specifically, we extend the definition of correntropy from random variables to random vectors and derive some multi-kernel correntropy-based estimators. Then, we apply the derived estimators to the orientation estimation of IMUs and disturbance observers of exoskeletons. As for the gait phase estimation and assistance map construction, we construct a robust adaptive oscillator (RAO) to meet the requirement of walking in a community and design the assistance map based on the user’s preferences so as to accommodate individualized coordination strategies.

1.1 Literature Review

In this section, we give a brief review of the literature on exoskeletons, correntropy-based estimators, orientation estimation of IMUs, disturbance observers, adaptive oscillators, and preference-based control.

1.1.1 Exoskeletons for Gait Rehabilitation and Walking Assistance

Over the past decades, many exoskeletons have been designed for gait rehabilitation and walking assistance. For gait rehabilitation, exoskeletons can be divided into treadmill-based exoskeletons and ground-based exoskeletons. Treadmill-based exoskeletons are typically equipped with a body-weight support system (BWSS) for balance support, a screen to visualize gait training, and a sensing system to regulate human-robot interaction. Exoskeletons of this type include the Lokomat [19], LOPES [20], and ALEX [21] (in Figures 1.1(a), 1.1(b), and 1.1(c), respectively), which can provide consistent and intensive gait training for stroke patients in the early stage of stroke.

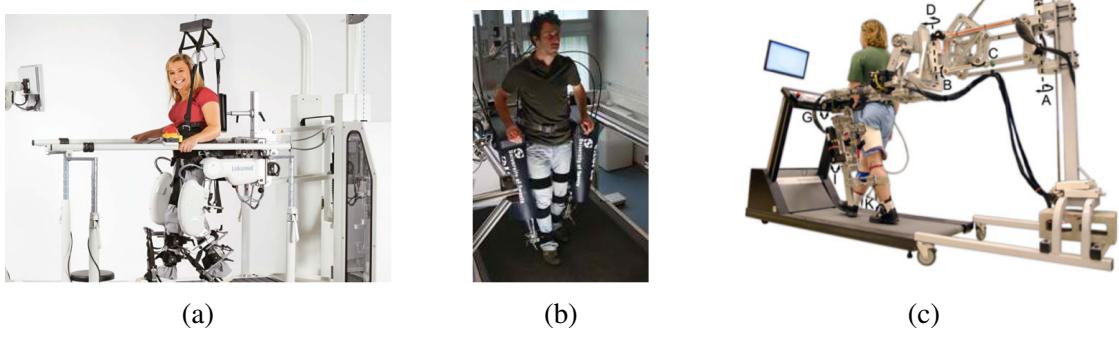


Figure 1.1: Treadmill-based exoskeletons. (a) Lokomat. (b) LOPES. (c) ALEX.

Compared with training on a treadmill, overground training offers a better representation of the community and home ambulation [22], and provides a more realistic gravity perception. Therefore, it can facilitate the patient's balance relearning and is beneficial to the neuromuscular rehabilitation [13, 23, 24]. Exoskeletons for this purpose include Indego [22], HAL [23], and USTC exoskeleton [24] (in Figures 1.2(a), 1.2(b), and 1.2(c), respectively).

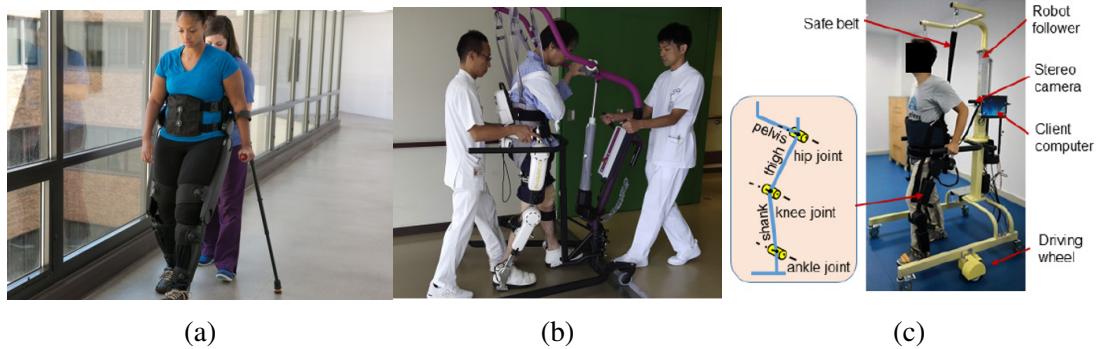


Figure 1.2: Ground-based exoskeletons. (a) Indego. (b) HAL. (c) USTC Exoskeleton.

Recently, some exoskeletons have broken the “metabolic barrier” with both unpowered and powered devices. Exoskeletons of this type include CMU unpowered ankle exoskeleton [55] in Figure 1.3(a), BLS unpowered hip exoskeleton [56] in Figure 1.3(b), CMU powered ankle exoskeleton [29] in Figure 1.3(c), Samsung hip exoskeleton [57] in Figure 1.3(d), Harvard hip exosuit [30] in Figure 1.3(e), and Harvard ankle exosuit [58] in Figure 1.3(f).



Figure 1.3: Exoskeletons that have broken “metabolic barrier”. (a) CMU unpowered ankle exoskeleton. (b) Bent-leaf-spring (BLS) unpowered hip exoskeleton. (c) CMU powered ankle exoskeleton. (d) Samsung hip exoskeleton. (e) Harvard hip exosuit. (f) Harvard ankle exosuit.

There is a trend of assisting the elder persons or stroke patients in the community using powered exoskeletons [34, 37–41]. A predefined force profile or assistance map is constructed based on the estimated gait phase and is utilized to restore normal gait or reduce walking economy. Exoskeletons of this type include the Harvard portable exosuit [34] in Figure 1.4(a), Curara [40] in Figure 1.4(b), and GEMS-L [41] in Figure 1.4(c).

1.1.2 Correntropy-based Kalman Filter

The Kalman filter (KF) is an minimum mean square error estimator (MMSE) for a linear system with Gaussian noise. It has been successfully applied to many areas, including

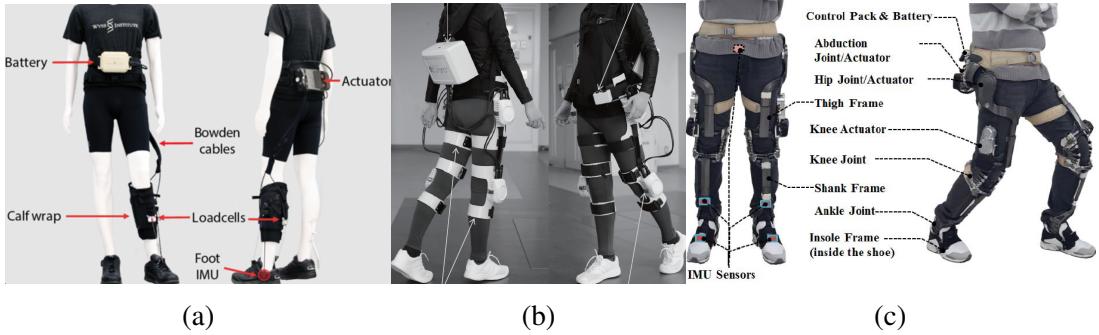


Figure 1.4: Exoskeletons for elder persons or post-stroke patients. (a) Harvard portable exosuit. (b) Curara. (c) GEMS-L.

motion tracking [59], sensor fusion [60], system identification [61], control systems [62], pattern recognition [63], orientation estimation of IMUs [64, 65, 65–67], and disturbance observers [68–70]. Although KF is an attractive estimator due to its closed-form solution, computational feasibility, and familiarity, its performance may be poor when the system encounters outliers or disturbances. The reason is that the KF is derived based on the mean squared error (MSE) based objective function and is sensitive to large residuals.

Many robust techniques have been applied to the KF to increase its robustness, such as the Huber-based KF (HKF) [71, 72], robust Student’s t -based KF [73, 74], the modified influence function-based KF [75], and the τ -divergence based KF [76]. The HKF adopt Hubers function to mitigate the bad effects of outliers; the Student’s t -based KF utilized the heavy-tailed Student’s t -distribution to model the noises; the modified influence function-based KF employed a bounded influence function for algorithm derivation; the τ -divergence based KF was obtained by minimizing the mean square error according to the least favorable model in a ball that bounded by the τ -divergence. Although these methods can increase the robustness of the KF, they assume that the non-Gaussian noises exist in all measurement channels or process channels, and their performances may be poor when only some specific channels are contaminated by non-Gaussian noises.

The correntropy originates from the information-theoretic learning (ITL) [77, 78] and it captures higher-order statistics [79] compared with the second-order moment (i.e., MSE). Therefore, it offers potential improvements compared with MSE-based algorithms. It has been widely used as a robust cost for machine learning [80], adaptive filtering [81], regression [82], and state estimation [79, 83]. A good property of the correntropy is that the correntropy induced metric (CIM) varies from an ℓ_2 norm to an ℓ_0 norm [84]: it is similar to an ℓ_2 norm when the error is small (the distance between two random variables),

it approaches an ℓ_1 norm when the error is bigger, and it behaves like an ℓ_0 norm when the error is extremely large. Since a lower norm is more robust to outliers, the algorithms that are derived based on the maximum correntropy criterion (MCC) are robust to heavy-tailed noises. Using this property, the maximum correntropy KF (MCKF) is derived in [79, 85, 86]. Its sequential form, Chandrasekhar-type recursion, and square-root form are derived in [87–89]. It is also extended to the nonlinear system with MC-EKF [90], MC-UKF [91], and MC-GHKF [92]. Moreover, it is applied to the system with state constraints [93, 94], distributed state estimation [95], and interacting multiple model [96]. In this thesis, we collectively regard these filters as correntropy-based Kalman filters (CKF).

1.1.3 Orientation Estimation of IMUs

Micro-Electro-Mechanical-system (MEMS) based-IMUs have attracted continuing research efforts for their advantages of low cost, real-time communication, portability, and small size. They are found in most smart-phones, virtual reality (VR) headsets, and tablet computers, and have been widely used in the fields of navigation [97], robotics [41, 98], localization [99], motion capture [100], and rehabilitation [101].

IMUs are important components of exoskeletons. They are utilized for gait phase estimation in [34, 45], adopted for gait orientation estimation in [26, 27, 41], and employed for gait assessment in [46, 47]. MEME based-IMUs can be divided into two types: the six-axis IMU composed of accelerometers and gyroscopes, and the nine-axis IMU composed of accelerometers, gyroscopes, and magnetometers. The nine-axis IMU is also referred to as the attitude and heading reference system (AHRS) or the magnetic angular rate and gravity (MARG) system, where gyroscopes measure the angular velocity rate while accelerometers and magnetometers are used to correct the long-term gyroscope drift.

Many algorithms have been developed for orientation estimation using IMUs. Among them, the KF-based method may be the most popular one due to its familiarity and good performance, such as the extended Kalman filter (EKF) in [51, 65], the unscented Kalman filter (UKF) in [66, 67]. Other methods, such as complementary filter [102, 103], gradient descend (GD) [48, 104], and sliding mode observer [105], are also developed with different characteristics and are used for different applications. The KF-based method

has been a good remedy for IMUs without considering external accelerations and magnetic disturbances. Unfortunately, when walking with IMUs, the external acceleration is considerable (especially when attaching IMUs to feet) and the magnetic disturbance may be involved by the surrounding ferromagnetic materials, like iron and magnets. The KF is sensitive to outliers or abrupt disturbances since it is derived based on the MSE criterion. Therefore, the performance of the orientation estimation employing the KF would degenerate significantly when external disturbance is involved in the system.

A conventional way to mitigate the influence of the disturbance is to involve the disturbance as a new state and construct the augmented state KF [106]. However, this method's performance usually is not satisfactory since the accurate disturbance dynamics is difficult to obtain. Li et al. [49] used a Gaussian-Markov model to estimate the external acceleration; Roetenberg et al. [51] employed an adaptive measurement covariance to eliminate the influence of the magnetic disturbance; Madgwick et al. [48] re-normalized the magnetic vector to minimize its influence on inclination portions (i.e., roll and pitch); Seel and Ruppin [50] used a constraint so that the inclination was only affected by accelerometers while the heading was only affected by magnetometers. These methods, in general, can alleviate the bad effects of external accelerations and magnetic disturbances. Their performances, however, are usually unsatisfactory because the estimation errors are sacrificed when without disturbances.

1.1.4 Disturbance Observer

Disturbance widely exists in mechanical systems and aeronautic systems, such as industrial robotic manipulators [52], motion servo systems [53], disk drive systems [54], missiles [107], satellites [108], and many others. For an exoskeleton system, the disturbance comes from both the internal system, such as friction torques, parameter perturbations, cogging torques, and the external system, such as interacting with the environment. The disturbance significantly deteriorates the exoskeleton performance. Hence, disturbance rejection is necessary and important for the controller design of the exoskeleton.

An intuitive method for disturbance attenuation is feedforward control when the disturbance is measurable. However, the sensors may be expensive, and sometimes the disturbance is immeasurable. An alternative method is to design robust controllers. However, there is an intrinsic trade-off between the robustness against disturbance and the nominal

performance (e.g., the tracking accuracy), which is also referred to as a single degree of freedom control structure. For example, robust controllers achieve robustness to disturbance at the price of degraded nominal performance [109]; the involvement of integral action in the proportional-integral-derivative (PID) controller can increase the robustness against disturbance but may increase the overshoot. The disturbance observer (DOB) is a promising technique to address the aforementioned issue. It acts as an add-on component for the baseline controller and can increase the robustness of the controller against disturbance and recover its nominal performance when the disturbance disappears. This property allows the user to design the disturbance observer and the controller independently, thereby largely simplifying the controller design.

Various linear disturbance observers have been designed by different researchers for different applications, which include the frequency domain-based DOB [110], the extended state observer (ESO) in active disturbance rejection control (ADRC) [111, 112], the unknown input observer (UIO) in disturbance accommodation control (DAC) [113], the Kalman filter-based disturbance observer (KF-DOB) [68], the uncertainty disturbance estimator (UDE) [114], the equivalent input disturbance estimator (EID) [115], and so on. The frequency domain-based DOB was proposed by Ohishi and his colleagues [110] where the inverse of the plant model accompanied with a filter was used to estimate the lumped disturbance; the ESO was designed by Han [112] for the purpose of estimating the lumped disturbance; the UIO [113] estimated both the state and the disturbance by assuming that the disturbance dynamics was *a priori* knowledge; the KF-DOB [68] modeled the disturbance as a new state and estimated the state and disturbance simultaneously using a KF recursion; the mechanism of the UDE [114] was quite close to the frequency domain-based DOB where a filter was utilized to make the disturbance estimation implementable; the EID [115] can be regarded as an alternative to the ESO by deliberately selecting the parameters.

Although plenty of linear disturbance observers are available with different characteristics. They usually use a constant gain (or Kalman gain) to estimate the state and the disturbance [68, 110–112, 114, 115], which intrinsically causes a trade-off among the disturbance estimation, state estimation, and noises suppression. For example, in the KF-DOB, the estimation is optimal if the disturbance dynamics is accurately known and

the noises are Gaussian. However, in a practical application, the disturbance dynamics usually is unknown and time-varying, and non-Gaussian noises may be involved. In this case, re-setting the noise covariance matrices can improve the robustness of the KF against disturbance, at the cost of estimation degeneration when disturbance is absent.

The multi-kernel correntropy (MKC) provides a solution for state estimation with disturbances (we will introduce it in Chapter 2). However, it is derived under Gaussian kernels. The Gaussian kernel is not always the best kernel in a practical application. Chen et al. [116] utilized the generalized Gaussian density (GGD) function as kernels and proposed the generalized maximum correntropy criterion (GMCC) for adaptive filtering. After that, this criterion has been utilized for active noise control [117], multiple-hypothesis detection [118], and many others. An important property of the GMCC is that the corresponding generalized correntropy loss (GCL) varies from ℓ_α^α to ℓ_0^0 (α is the shape parameter in GGD) based on the location of the errors, which is more powerful than the least mean p -power (LMP) criterion [119–121]. Unfortunately, the GCL is not a good metric for state estimation with disturbances where only some channels are contaminated by non-Gaussian noises, which motivates us to extend the definition of the MKC under the Gaussian kernel to the generalized multi-kernel correntropy (GMKC) under the GGD.

1.1.5 Adaptive Oscillators and Preference-based Control

The adaptive oscillator (AO) is a popular approach for gait phase estimation and has been successfully utilized in many exoskeletons [41–44, 122]. It was proposed by Righetti et al. [123] to synchronize with periodic or pseudo-periodic signals using the dynamic Hebbian learning rule. Then, it was utilized for human-robot synchronization by extracting elbow motion features and compensating elbow dynamics [122]. To enhance its capability, Ronsse et al. [124] coupled it with kernel-based filters for the purpose of velocity and acceleration estimation, while Seo et al. [42] proposed a particularly-shaped adaptive oscillator (PSAO) to estimate the gait phase with a single hip angle. Notably, AO employs a gradient-based method for phase estimation, hence inevitably owing a phase lag when the input signals are non-steady. To mitigate this problem, Zhang et al. [43] designed an extra phase reset and phase error learning block to identify the zero-phase, while Xue et al. [44] developed a dual-oscillator structure to alleviate the phase error.

The force profile or the assistance map of the exoskeleton is usually built upon the

gait phase and has to be designed deliberately to improve walking economy. One method for force profile construction is parameter sweeping [35, 125, 126] which traverses all possible profiles and finds the most efficient one by trial and error. However, this method is time-consuming, especially when the parameter dimension is high. Recently, human-in-the-loop (HIL) optimization has emerged and attracted a lot of research attention since it is an automatically discovering, customizing, and continuously adapting strategy [29, 30, 127]. However, this approach requires indirect calorimetry equipment, and is time-consuming too (83 ± 14 min, 21.4 ± 1 min, 50 min in [29, 30, 127], respectively). Therefore, it is not suitable for walking assistance in a community. Moreover, it ignores the user’s walking coordination strategy and feelings, and may induce discomfort and speed up fatigue [31].

The preference-based learning [128–130] provides a potential solution to the above issue. This method incorporates the user’s preference into the exoskeleton controller design. Tucker et al. [128] utilized this method for step length and step width optimization using the Atalante exoskeleton. Simulations and experiments revealed that the learning efficiency of this approach was higher than the other methods. After that, it was further applied to a six-dimensional learning task, and its performance was reported to be satisfactory [129]. This method also demonstrated the feasibility of recovering gait utility landscapes from limited human trials [130], which was beneficial for reliably learning the entire preference landscape. Preference-based learning has been intensively investigated in the field of gait optimization for an exoskeleton system. However, to the best of our knowledge, it has not been utilized for walking assistance in a community.

1.2 Thesis Outline and Contributions

In this thesis, we focus on paving the way for walking assistance in a community using a lightweight exoskeleton. To handle the performances degeneration of IMUs and disturbance observers, we formulate them as a state estimation issue with some specific channels contaminated by heavy-tailed noises. To cope with this problem, we propose a novel metric, the MKC, as a robust cost and derive a multi-kernel maximum correntropy Kalman filter (MKMCKF). Then, we apply the MKMCKF to the orientation estimation of six-axis IMUs and nine-axis IMUs. Extensive simulations and experiments verify the

effectiveness of the proposed method. Meanwhile, we extend the MKC under a Gaussian kernel to the generalized multi-kernel correntropy (GMKC) under the generalized Gaussian density (GGD) kernel. Then, we derive a generalized multi-kernel maximum correntropy Kalman filter (GMKCMKF) and apply it to the disturbance observer. Simulations reveal that the proposed method is more versatile than the MKMCKF and outperforms the benchmark methods. To incorporate the user's preference into the exoskeleton controller, we design a robust adaptive oscillator (RAO) for gait phase estimation. Moreover, we construct the assistance map based on the user's preference. Experiments verify the efficacy of the proposed method. The contributions of this thesis are summarized as follows.

In Chapter 2, we consider a state estimation problem where some specific channels are contaminated by heavy-tailed noises. To address this issue, we extend the definition of correntropy from random variables to random vectors and define the MKC. Then, we introduce the corresponding multi-kernel correntropy induced metric (CIM) where its behavior in different channels can be designed independently. Specifically, when applying an infinite bandwidth to a channel, the corresponding CIM is very similar to an ℓ_2 norm; when applying a relatively small bandwidth to a channel, the corresponding CIM changes from an ℓ_2 norm to an ℓ_0 norm. Since a lower norm is more robust to heavy-tailed noises or outliers, using the MKC as a robust cost to replace the conventional MSE should be beneficial. The advantage of the MKC over the traditional correntropy is that its CIM in each channel can be designed independently. The traditional correntropy utilizes a universal bandwidth for all channels, which is not suitable for the situation where only some specific channels are contaminated by non-Gaussian noises. The MKC solves this problem by employing additional kernel bandwidths. Based on the MKC, we derive a MKMCKF which is robust to outliers but not conservative (which means that it retains good performance with Gaussian noises). The contributions of this chapter are summarized as follows:

- 1) The definition of the correntropy is extended from random variables to random vectors. Some important properties of the MKC are given with rigorous proof, which guarantees the feasibility of the MKC as a robust cost.
- 2) The MKMCKF is derived based on the MKC, which actually is a generalization of the KF and MCKF. The proposed algorithm is robust to heavy-tailed noises while not

conservative. Its performance significantly outperforms the KF and the MCKF when some specific channels contain non-Gaussian noises.

- 3) We validate the performance of the proposed algorithm in three cases: systems with process non-Gaussian noises, systems with process disturbance, and systems with measurement disturbance. In each case, we compare it with the KF, the MCKF, and the particle filter (PF). Simulations verify the effectiveness of the proposed method.

In chapter 3, we apply the MKMCKF to orientation estimations using six-axis IMUs and nine-axis IMUs. For six-axis IMUs with accelerometers and gyroscopes, we qualitatively analyze the influence of external accelerations on the conventional ESKF [50, 106] and reveal that the existing algorithm is incapable of accurate orientation tracking with obvious external accelerations. To address this issue and simultaneously minimize the algorithm complexity, we derive a compact multi-kernel maximum correntropy Kalman filter (CMKMCCKF) where all Gaussian channels utilize the infinite bandwidth and all non-Gaussian channels utilize the universal bandwidth σ . Then, we design a compact multi-kernel maximum correntropy-based error state Kalman filter (CMKMC-ESKF) for orientation estimation of six-axis IMUs. Simulations and experiments are conducted to verify the effectiveness of the proposed method, which reveals that the proposed method is significantly better than the gradient descent (GD) [48] and the conventional ESKF [50, 106], especially with fast-varying accelerations.

As for nine-axis IMUs with accelerometers, gyroscopes, and magnetometers, the performance of the traditional ESKF degenerates significantly with external accelerations and magnetic disturbances. To handle this issue, we derive a multi-kernel maximum correntropy Kalman filter for orientation estimation (MKMCKF-OE). Since the disturbances for nine-axis IMUs come from two different sources (e.g., external accelerations and magnetic disturbances) with different characteristics, we need to use two distinct bandwidths for these two disturbances. Generally, we aim to choose a set of kernel bandwidths so that the objective of the MKMCKF-OE is similar to an ℓ_2 norm-based cost without disturbance and is similar to an ℓ_0 or ℓ_1 norm-based cost with disturbance. However, it is difficult to tune the bandwidths manually, especially when the system dimension is high and there are some constraints. To handle this issue, we employ Bayesian optimization (BO) for kernel bandwidth selection, which is an efficient tool for global minimization

where the objective function can be black-box, non-convex, and expensive to be evaluated [131–133]. We validate the performance of the proposed algorithm on the validation set and compare it with the (GD) [48] method, conventional ESKF [51, 106], and improved gradient descent (IGD) [50]. Intensive experiments verify the effectiveness of the proposed algorithm. Contributions of this chapter are summarized as follows:

- 1) The effects of external accelerations on the conventional ESKF for six-axis IMUs are investigated, revealing that the conventional ESKF is incapable of accurate orientation tracking with fast-varying external accelerations.
- 2) A CMKMCKF is designed to suppress the bad effects of disturbances while simultaneously minimizing the algorithm complexity. Then, we apply this algorithm to six-axis IMUs orientation estimation and construct the CMKMC-ESKF.
- 3) The performance of the CMKMC-ESKF is compared with that of the GD and the ESKF in both simulations and experiments. The results reveal that the proposed algorithm has a moderate computation burden and its performance is significantly better than the benchmark methods.
- 4) For the nine-axis IMUs, a novel MKMCKF-OE is designed. Moreover, the BO is utilized to automatically find the optimal kernel bandwidths, which is revealed to be both efficient and convenient.
- 5) The performance of the MKMCKF-OE is validated on the validation set and is shown to be significantly better than the GD and ESKF, and is superior to the IGD on the yaw axis with magnetic disturbance. The MKMCKF-OE is also implemented in an embedded system and its performance is validated by an optical motion capture system.

In Chapter 4, we extend the MKC under the Gaussian kernel to a generalized multi-kernel correntropy (GMKC) under the GGD. Some important properties of the GMKC are given in Theorems 4.1-4.5, which disclose the intrinsic robustness of the GMKC as a cost function. The corresponding generalized multi-kernel correntropy loss (GMKCL) is also provided, which is associated with a class of heavy-tailed distributions and is proven to be an optimal objective under some assumptions based on the maximum *a posteriori* probability (MAP). We derive a generalized multi-kernel maximum correntropy Kalman

filter (GMKMCKF) based on the GMKCL. Then, we apply it to the disturbance estimation and validate its effectiveness on a single degree of freedom exoskeleton. Further, we compare it with the ESO [112], KF-DOB [68], MCKF [79], and particle filter (PF) [134]. The contributions of this chapter are summarized as follows:

- 1) The noise distribution of the disturbance channel is formulated as an ε -contaminated mixture model. Then, we reveal that the conventional Gaussian distribution can not approach this kind of distribution effectively. To cope with this problem, we extend the MKC to the GMKC, providing its properties in Theorem 4.1-4.5 and demonstrating its feasibility as a robust cost.
- 2) The GMKCL is revealed to be an optimal objective function under a certain class of heavy-tailed distributions based on MAP. Meanwhile, a GMKMCKF is derived based on the GMKCL and is applied to the disturbance estimation.
- 3) Simulations are conducted and the performance of the GMKMCKF is compared with that of the ESO, KF-DOB, MCKF, and PF. The results verify the effectiveness of the proposed method.

In Chapter 5, we design a robust adaptive oscillator (RAO) for human-exoskeleton synchronization and construct the assistance map of exoskeletons based on the user's preference. Specifically, we employ the zero-cross moment to identify the zero phases and the gait frequency. Then, we use this information to update the estimated phases and frequencies through nonlinear gains. Both theoretical analysis and experiments support that the RAO has a faster convergence rate compared with the AO. For the assistance map construction, we employ the GPR strategy where the inputs are the user-selected assistance parameters and gait features whereas its outputs are the predicted assistance parameters. The proposed method has some advantages compared with the traditional HIL method. Firstly, it does not require indirect calorimetry equipment. Subjects can explore assistance profiles and evaluate their utility through proprioceptive feedback. This process can be conducted naturally in daily walking. Hence, it is very convenient and does not require any special equipment. Secondly, the learning efficiency of the proposed methods is superior to that of the HIL. One reason for the slow convergence rate of the HIL is that it may disturb the predictive forward model in our motor system [135] since the user cannot predict the changes of the following control strategy. Hence, users

need to spend more time adapting to the machine. Another reason is that signals from the calorimetry device are very noisy and have to be averaged over a few minutes, which hinders the fast convergence of HIL. These drawbacks can be mitigated by the preference-based learning method since the assistance profile is adjusted based on the users' selection and its effect is predictive. Moreover, subjects can evaluate their "preference score" in a few seconds. Therefore, the learning efficiency of our method is higher than the HIL. Thirdly and most importantly, interaction comfort, as an important evaluation standard of the wearing experience, is ignored by HIL. However, it can be accessed by our proprioception system [136–138] naturally. Bertram [139] showed that humans prefer gaits with a smaller metabolic expenditure. This supports the fact that users can 'feel' the biological efforts and try to select the most energy-saving walking strategy. Actually, preference-based learning can be seen as a combination of multi-purpose optimization that includes interaction comfort and biological effort reduction. These combined metrics should be more reasonable and practical than the single metabolism reduction criterion in a real application.

To investigate the metabolic impact of the proposed method, we investigate the lower-limb muscle activity using sEMG sensors under different walking modes. The results show that the proposed method can achieve muscle activity reduction compared with normal walking (without exoskeletons). The contributions of this chapter are summarized as follows:

- 1) A RAO is designed which aligns the zero phase automatically and has a faster convergence rate compared with AO.
- 2) A GPR learning strategy is developed to construct the assistance map, which predicts the assistance parameters based on users' historical preferences. This method has the advantages of not requiring additional equipment, an efficient learning procedure, and interaction comfort.
- 3) The user-selected assistance profile is validated by investigating lower-limb muscle activities at 5 km/h on the treadmill. Results reveal that the user-selected assistance can achieve a muscle activity reduction compared with normal walking without exoskeletons.

1.3 List of Publications

Journal Papers

- **Shilei Li**, Dawei Shi, Wulin Zou, and Ling Shi, “Multi-Kernel Maximum Correntropy Kalman Filter”, *IEEE Control Systems Letters*, vol. 6, pp. 1490-1495, 2021.
- **Shilei Li**, Lijing Li, Dawei Shi, Wulin Zou, Pu Duan and Ling Shi, “Multi-kernel Maximum Correntropy Kalman Filter for Orientation Estimation”, *IEEE Robotics and Automation Letters*, vol. 7, pp. 6693-6700, 2022.
- **Shilei Li**, Peihu Duan, Dawei Shi, Wulin Zou, Pu Duan, and Ling Shi, “Compact Maximum Correntropy-Based Error State Kalman Filter for Exoskeleton Orientation Estimation”, *IEEE Transactions on Control Systems Technology*, accepted.
- **Shilei Li**, Dawei Shi, Wulin Zou, and Ling Shi, “Generalized Multi-kernel Maximum Correntropy Kalman Filter for Disturbance Estimation”, *IEEE Transactions on Signal Processing*, under review.
- **Shilei Li**, Wulin Zou, Pu Duan, and Ling Shi, “Preference-based Assistance Map Learning with Robust Adaptive Oscillators”, *IEEE Transactions on Medical Robotics and Bionics*, under review.
- Wulin Zou, Xiang Chen, **Shilei Li**, Pu Duan, Ningbo Yu and Ling Shi, “Robust Admittance Control for Human Arm Strength Augmentation with Guaranteed Passivity: A Complementary Design”, *IEEE Transactions on Mechatronics*, accepted.

Conference Papers

- **Shilei Li**, Dawei Shi, Wulin Zou, and Ling Shi, “Multi-Kernel Maximum Correntropy Kalman Filter,” *Proceedings of 2022 American Control Conference*, accepted.
- Wulin Zou, Xiang Chen, **Shilei Li**, Pu Duan, Ningbo Yu, Ling Shi, “Multi-Objective Admittance Control: An LMI-Based Method”, *Proceedings of 2022 7th IEEE International Conference on Advanced Robotics and Mechatronics*, accepted.

Chapter 2

Multi-kernel Maximum Correntropy Kalman Filter

In this chapter, we derive a multi-kernel maximum correntropy Kalman filter (MKMCKF) which is a generalization of the KF and MCKF. Specifically, we extend the definition of correntropy from random variables to random vectors and define a multi-kernel correntropy. Then, we provide its properties and reveal that the multi-kernel correntropy loss is a suitable objective function for systems in which some channels contain heavy-tailed noises. Further, we derive the MKMCKF based on the proposed multi-kernel correntropy. Simulations verify the effectiveness of the proposed algorithm.

2.1 Problem Formulation and Preliminaries

2.1.1 Kalman Filter

We consider a linear time-invariant (LTI) system:

$$\begin{aligned} x_{k+1} &= Ax_k + w_k \\ y_k &= Cx_k + v_k \end{aligned} \tag{2.1}$$

where $x_k \in \mathbb{R}^n$ is the state, $y_k \in \mathbb{R}^m$ is the measurement, and w_k and v_k are Gaussian noises with $w_k \sim \mathcal{N}(0, Q_k)$ and $v_k \sim \mathcal{N}(0, R_k)$ where $Q_k \succcurlyeq 0$ and $R_k \succ 0$. Denote the measurement set until time step k as $\{y_k\}_1^k := \{y_1, y_2, \dots, y_k\}$ (for brevity, we use $\{y_k\}$ to denote $\{y_k\}_1^k$). The pair $(A, \sqrt{Q_k})$ is controllable and (A, C) is observable. The initial state $x_0 \sim \mathcal{N}(0, \Pi_0)$ is uncorrelated with w_k and v_k for $k > 0$. Then, the following four estimators coincide [140]:

$$E(x_k | \{y_k\}), \text{conditional mean} \tag{2.2}$$

$$\max_{x_k} p(x_k | \{y_k\}), \text{maximum } a posteriori \text{ (MAP)} \quad (2.3)$$

$$\min_{\hat{x}_k} E(\|x_k - \hat{x}_k\|_2^2), \text{minimum mean square error (MMSE)} \quad (2.4)$$

$$\min_{\hat{x}_k \in \text{span}(\{y_k\})} E(\|x_k - \hat{x}_k\|_2^2), \text{minimum linear expected MSE.} \quad (2.5)$$

The well-known Kalman filter can be derived based on any of the above criteria when the system is linear with Gaussian noises. It can be written as

$$\hat{x}_k^- = A\hat{x}_{k-1}^+ \quad (2.6)$$

$$P_k^- = AP_{k-1}^+A^T + Q_k \quad (2.7)$$

$$K_k = P_k^- C^T (CP_k^- C^T + R_k)^{-1} \quad (2.8)$$

$$\hat{x}_k^+ = \hat{x}_k^- + K_k(y_k - CP_k^- C^T) \quad (2.9)$$

$$P_k^+ = (I - K_k C)P_k^- \quad (2.10)$$

where \hat{x}_k^- is the *a priori* estimate of x_k , \hat{x}_k^+ is the *a posteriori* estimate of x_k , and P_k is the error covariance at time step k .

2.1.2 Problem Formulation

We assume that some channels of the system are contaminated by heavy-tailed noises due to linearization of the nonlinear dynamics, involvement of disturbances, or measurement errors. Actually, non-Gaussian noises are very common in many applications. Mathematically, we can write the process and measurement noise as

$$\begin{aligned} w_k &= \Lambda_p w_{G,k} + (I_{n \times n} - \Lambda_p) w_{nG,k} \\ v_k &= \Lambda_r v_{G,k} + (I_{m \times m} - \Lambda_r) v_{nG,k} \end{aligned} \quad (2.11)$$

where Λ_p and Λ_r are diagonal indicator matrices with diagonal entries set to 1 for Gaussian channels and 0 for non-Gaussian channels, $w_{G,k}$ and $v_{G,k}$ are Gaussian noises, and

$w_{nG,k}$ and $v_{nG,k}$ are non-Gaussian noises. Many non-Gaussian noises can be represented by a mixture of Gaussian models [79]. Unfortunately, this type of heavy-tailed distribution cannot be approximated by a single Gaussian distribution effectively (see Figure 2.1), resulting in the performance degeneration of the KF with non-Gaussian noises.

In this chapter, we propose a multi-kernel correntropy to address this issue.

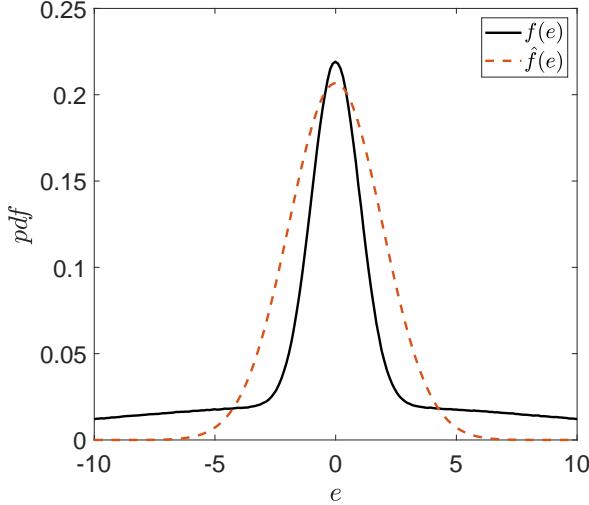


Figure 2.1: Approximating a mixture Gaussian distribution using a single Gaussian distribution. The target distribution $f(e)$ follows $0.5\mathcal{N}(0, 1) + 0.5\mathcal{N}(0, 100)$. The estimated distribution is obtained by minimizing the mean squared error $E[(f(e) - \hat{f}(e))^2]$.

2.2 Multi-kernel Correntropy

The correntropy is a local similarity measure for random variables $X, Y \in \mathbb{R}$ with joint distribution $F_{XY}(x, y)$. It is defined by

$$C(X, Y) = E[\kappa(X, Y)] = \int \kappa(x, y) dF_{XY}(x, y) \quad (2.12)$$

where $\kappa(x, y)$ is a shift-invariant Mercer kernel. A common kernel is the Gaussian kernel with

$$\kappa(x, y) = G_\sigma(x, y) = \exp\left(-\frac{e^2}{2\sigma^2}\right) \quad (2.13)$$

where $e = x - y$ and σ is the kernel bandwidth. In practice, the joint distribution $F_{XY}(x, y)$ is unknown and only samples $x(k)$ and $y(k)$ are available. Then, the correntropy can be obtained by

$$\hat{C}(X, Y) = \frac{1}{N} \sum_{k=1}^N \kappa(x(k), y(k)) = \frac{1}{N} \sum_{k=1}^N G_\sigma(x(k), y(k)) \quad (2.14)$$

where N is the sample number and k is the sample index. The correntropy induced metric (CIM) is defined as

$$CIM(X, Y) = \left(1 - \frac{1}{N} \sum_{k=1}^N G_\sigma(x(k), y(k)) \right)^{1/2} \quad (2.15)$$

In this chapter, we define the multi-kernel correntropy for random vectors $\mathcal{X}, \mathcal{Y} \in \mathbb{R}^l$ as

$$C(\mathcal{X}, \mathcal{Y}) = \sum_{i=1}^l E[\tilde{\kappa}_i(\mathcal{X}_i, \mathcal{Y}_i)] = \sum_{i=1}^l \int \tilde{\kappa}_i(x_i, y_i) dF_{\mathcal{X}_i \mathcal{Y}_i}(x_i, y_i) \quad (2.16)$$

with

$$\tilde{\kappa}_i(x_i, y_i) = 2\sigma_i^2 \kappa_{\sigma_i}(e_i) = 2\sigma_i^2 G_{\sigma_i}(e_i) = 2\sigma_i^2 \exp\left(-\frac{e_i^2}{2\sigma_i^2}\right) \quad (2.17)$$

where \mathcal{X}_i and \mathcal{Y}_i are the i -th element of \mathcal{X} and \mathcal{Y} , σ_i is the kernel bandwidth for i -th channel, and $e_i = x_i - y_i$ is the corresponding error. In a practical application, the joint distribution $F_{\mathcal{X}_i \mathcal{Y}_i}$ is unknown and the correntropy can be estimated by

$$\hat{C}(\mathcal{X}, \mathcal{Y}) = \sum_{i=1}^l 2\sigma_i^2 \hat{C}_{\sigma_i}(\mathcal{X}_i, \mathcal{Y}_i) \quad (2.18)$$

with

$$\hat{C}_{\sigma_i}(\mathcal{X}_i, \mathcal{Y}_i) = \frac{1}{N} \sum_{k=1}^N G_{\sigma_i}(e_i(k)) \quad (2.19)$$

where $e_i(k) = x_i(k) - y_i(k)$ is the error for channel i at time step k . One can see that $\hat{C}(\mathcal{X}, \mathcal{Y})$ actually is a weighted summation of $\hat{C}_{\sigma_i}(\mathcal{X}_i, \mathcal{Y}_i)$ from $i = 1$ to l . The corresponding multi-kernel correntropy loss (MKCL) is defined as

$$J_{MKCL}(\mathcal{X}, \mathcal{Y}) = \left(\sum_{i=1}^l 2\sigma_i^2 (1 - \hat{C}_{\sigma_i}(\mathcal{X}_i, \mathcal{Y}_i)) \right) = \left(\sum_{i=1}^l 2\sigma_i^2 \frac{1}{N} \sum_{k=1}^N (1 - G_{\sigma_i}(e_i(k))) \right). \quad (2.20)$$

One can see that maximizing the multi-kernel correntropy is equivalent to minimizing the MKCL. Similar to (2.15), the multi-kernel CIM is written as

$$CIM(\mathcal{X}, \mathcal{Y}) = (J_{MKCL}(\mathcal{X}, \mathcal{Y}))^{1/2} = \left(\sum_{i=1}^l 2\sigma_i^2 \frac{1}{N} \sum_{k=1}^N (1 - G_{\sigma_i}(e_i(k))) \right)^{1/2}. \quad (2.21)$$

2.2.1 Properties of the Multi-kernel Correntropy

In this section, we provide some properties of the multi-kernel correntropy.

Theorem 2.1. *When setting all $\sigma_i \rightarrow \infty$, the MKCL in (2.20) becomes the expectation of the second order error moments with $\lim_{\sigma_i \rightarrow \infty} J_{MKCL}(\mathcal{X}, \mathcal{Y}) = E\|\mathcal{X} - \mathcal{Y}\|^2$.*

The proof can be found in Appendix 2.7.1.

Theorem 2.2. *The CIM in (2.21) defines a metric in the N-dimensional sample vector space.*

The proof of this theorem can be found in Appendix 2.7.2. The contour plots of $CIM(\mathcal{X}, 0)$ in 2D space with different kernel bandwidths are shown in Figure 2.2. One can see that when applying relatively big bandwidths ($\sigma_1 = 10, \sigma_2 = 10$) to both channels, the behavior of CIM is very similar to an ℓ_2 norm in the whole space. If the bandwidths are relatively small ($\sigma_1 = 1, \sigma_2 = 1$), the space is divided into three regions: it is approximated by an ℓ_2 norm when the error (distance to the origin) is small; and it approaches an ℓ_1 norm with the growth of the error; when the error is very large, it is similar to an ℓ_0 norm. When applying different bandwidths on different channels, e.g., $\sigma_1 = 1, \sigma_2 = 10$, we can obtain a varying norm in \mathcal{X}_1 channel and an ℓ_2 norm in \mathcal{X}_2 channel. Because the ℓ_2 norm-based objective function is optimal for Gaussian noises from the MAP perspective (we will explain this in the next section), and a lower norm-based objective function is more robust to heavy-tailed noises, the multi-kernel correntropy is an appropriate cost for systems where some specific channels are contaminated by non-Gaussian noises.

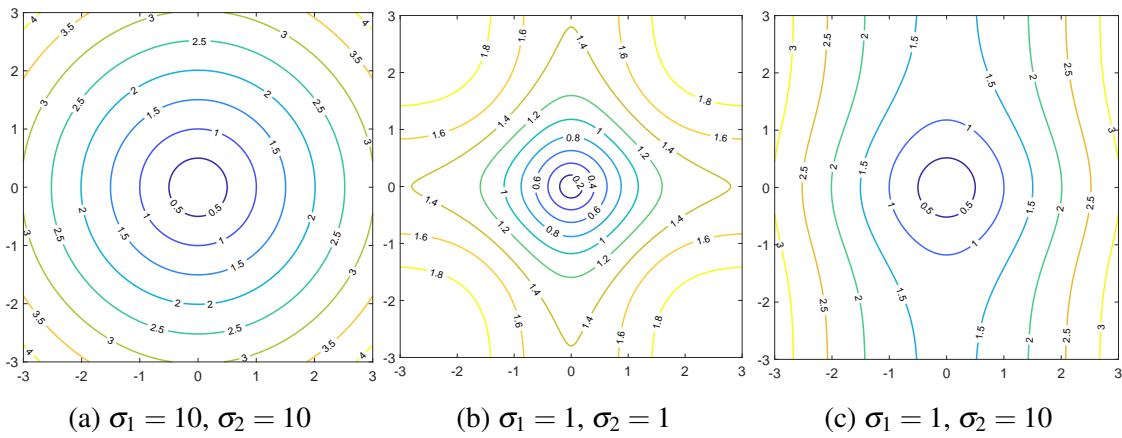


Figure 2.2: Contours of $CIM(\mathcal{X}, 0)$ in 2D space with different kernel bandwidths.

2.2.2 A Robust Cost for the Kalman Filter

For a linear system with Gaussian noises described in equation (2.1), based on Bayes' theorem, the *a posteriori* probability of x_k with measurement set $\{y_k\}$ has

$$\begin{aligned} p(x_k|\{y_k\}) &= p(x_k|\{y_{k-1}\}, y_k) \\ &= \frac{p(x_k, \{y_{k-1}\}, y_k)}{p(\{y_{k-1}\}, y_k)} \\ &= \frac{p(y_k|x_k, \{y_{k-1}\})p(x_k|\{y_{k-1}\})p(\{y_{k-1}\})}{p(y_k|\{y_{k-1}\})p(\{y_{k-1}\})} \\ &\propto p(y_k|x_k)p(x_k|\{y_{k-1}\}) \end{aligned} \quad (2.22)$$

where $p(y_k|x_k)$ is the probability of y_k conditional on the *a priori* estimate of x_k , and $p(x_k|\{y_{k-1}\})$ is the *a priori* estimate of x_k with $\{y_{k-1}\}$. From the perspective of maximum *a posteriori* probability, we obtain

$$\arg \max_{x_k} p(x_k|\{y_k\}) = \arg \max_{x_k} p(y_k|x_k)p(x_k|\{y_{k-1}\}). \quad (2.23)$$

Since w_k and v_k are Gaussian, and the initial state $x_0 \sim \mathcal{N}(0, \Pi_0)$ is uncorrelated with w_k and v_k , $p(y_k|x_k)$ and $p(x_k|\{y_{k-1}\})$ should also follow the Gaussian distribution (see [141]) with

$$\begin{aligned} p(y_k|x_k) &= \frac{\exp(-(y_k - C\hat{x}_k)^T R_k^{-1} (y_k - C\hat{x}_k))}{\sqrt{(2\pi)^m |R_k|}} \\ p(x_k|\{y_{k-1}\}) &= \frac{\exp(-(\hat{x}_k - A\hat{x}_{k-1})^T (P_k^-)^{-1} (\hat{x}_k - A\hat{x}_{k-1}))}{\sqrt{(2\pi)^n |P_k^-|}} \end{aligned} \quad (2.24)$$

where \hat{x}_k is the *a posteriori* estimate of x_k , $|R_k|$ is the determinant of R_k , P_k^- is the *a priori* estimate of error covariance, and $|P_k^-|$ is the determinant of P_k^- . It follows

$$\begin{aligned} \arg \max_{\hat{x}_k} p(x_k|\{y_k\}) &= \exp(-(y_k - C\hat{x}_k)^T R_k^{-1} (y_k - C\hat{x}_k)) \\ &\times \exp(-(\hat{x}_k - A\hat{x}_{k-1})^T (P_k^-)^{-1} (\hat{x}_k - A\hat{x}_{k-1})) \end{aligned} \quad (2.25)$$

where the normalization constants are ignored. It is equivalent to minimizing the negative log:

$$\begin{aligned} \arg \min_{\hat{x}_k} J_{MSE} &= \|R_k^{-1/2} (y_k - C\hat{x}_k)\|^2 + \\ &\| (P_k^-)^{-1/2} (\hat{x}_k - A\hat{x}_{k-1}) \|^2 \end{aligned} \quad (2.26)$$

which actually is a MSE criterion.

Theorem 2.3. *The Kalman filter in (2.6)–(2.10) can be derived by the MSE criterion in (2.26).*

The proof of this theorem is shown in Appendix 2.7.3. It reveals that the MSE criterion (an ℓ_2 norm-based objective function) is optimal for Gaussian noise from the MAP perspective. However, when some channels of the system are contaminated by heavy-tailed noises, the probability density function (2.24) does not hold. In this case, the ℓ_2 norm-based cost function is not the best, and a lower norm-based objective function is more suitable. To proceed, we define the process error $e_{p,k}$ and measurement error $e_{r,k}$ as

$$\begin{aligned} e_{p,k} &\triangleq (P_k^-)^{-1/2}(\hat{x}_{k|k} - A_k \hat{x}_{k-1}) \\ e_{r,k} &\triangleq R_k^{-1/2}(y_k - C_k \hat{x}_k). \end{aligned} \quad (2.27)$$

Then, a robust cost using the MKCL can be written as:

$$\arg \min_{\hat{x}_k} J_{MKCL} = J_{MKCL, \sigma_p}(e_{p,k}) + J_{MKCL, \sigma_r}(e_{r,k}). \quad (2.28)$$

where $\sigma_p = [\sigma_1, \sigma_2, \dots, \sigma_n]^T$ is the process bandwidth vector, $\sigma_r = [\sigma_{n+1}, \sigma_{n+2}, \dots, \sigma_{n+m}]^T$ is the measurement bandwidth vector. Based on Theorem 2.1, when setting all bandwidths to infinity, we have $\lim_{\sigma_i \rightarrow \infty} J_{MKCL} = J_{MSE}$. Since the behavior of the MKCL in each channel can be controlled independently by the kernel bandwidth, the MKCL is a more powerful cost function when some channels contains non-Gaussian noises. In one dimensional case, we have $J_{MSE}(e) = \|e\|^2$ and $J_{MKCL}(e) = 2\sigma^2(1 - \exp(-\frac{\|e\|^2}{2\sigma^2}))$. Figures of $J_{MSE}(e)$ and $J_{MKCL}(e)$ (with different bandwidths) are shown in Figure 2.3(a), and their influence functions $\nabla J_{MSE}(e)$ and $\nabla J_{MKCL}(e)$ are shown in Figure 2.3(b). One can see that when using a big bandwidth (i.e., $\sigma = 100$), $J_{MKCL}(e)$ is nearly identical to $J_{MSE}(e)$. On the contrary, when using relative small bandwidths (i.e., $\sigma = 1$ or $\sigma = 2$), $J_{MKCL}(e)$ goes toward saturation and its influence function $\nabla J_{MKCL}(e)$ goes towards zero for large residuals. This implies that a larger error contributes less to the $J_{MKCL}(e)$, hence it is robust to outliers. The maxima and minima points of $\nabla J_{MKCL}(e)$ are drawn by pentagrams, which corresponds to errors with $e = \pm\sigma$. This implies that the redescending property of $\nabla J_{MKCL}(e)$ is solely controlled by the kernel bandwidths.

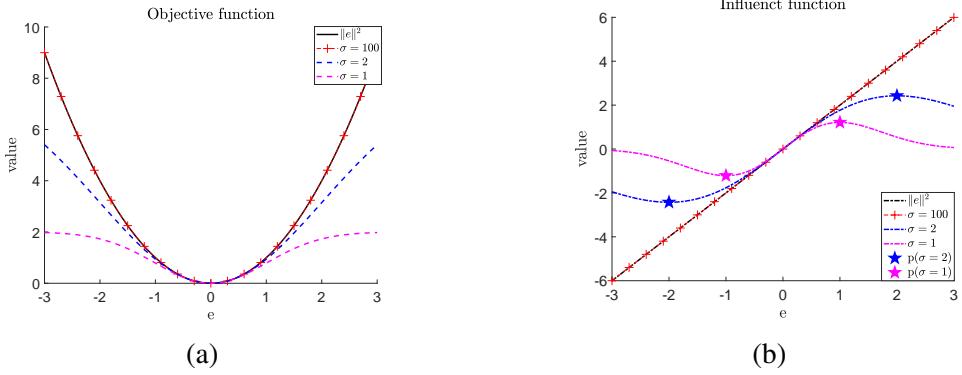


Figure 2.3: Objective function and influence function of the MSE and the MKCL in one dimensional case.

2.3 Algorithm Derivation

The system dynamics in (2.1) can be rewritten as

$$\begin{pmatrix} x_k^- \\ y_k \end{pmatrix} = \begin{pmatrix} I \\ C \end{pmatrix} x_k + v_k \quad (2.29)$$

where x_k^- is the *a priori* estimate of state x_k . The noise v_k has

$$v_k = \begin{pmatrix} x_k^- - x_k \\ v_k \end{pmatrix} \quad (2.30)$$

with

$$E(v_k v_k^T) = \begin{pmatrix} P_k^- & 0 \\ 0 & R_k \end{pmatrix} = \begin{pmatrix} B_{p,k} B_{p,k}^T & 0 \\ 0 & B_{r,k} B_{r,k}^T \end{pmatrix} = B_k B_k^T \quad (2.31)$$

where P_k^- is the *a priori* error covariance, and $B_{p,k}$ and $B_{r,k}$ can be obtained by Cholesky decomposition. Left multiplying B_k^{-1} in both sides of (2.29), we obtain

$$T_k = W_k x_k + \xi_k \quad (2.32)$$

where

$$T_k = B_k^{-1} \begin{pmatrix} x_k^- \\ y_k \end{pmatrix}, W_k = B_k^{-1} \begin{pmatrix} I \\ C \end{pmatrix}, \xi_k = B_k^{-1} v_k \quad (2.33)$$

The noise term ξ_k is white since $E(\xi_k \xi_k^T) = I$. We use the MKCL in (2.28) as a cost function, which is equivalent with

$$\arg \min_{x_k} J_{MKCL} = \sum_{i=1}^{n+m} 2\sigma_i^2 (1 - G_{\sigma_i}(e_{i,k})). \quad (2.34)$$

where $e_{i,k} = t_{i,k} - w_{i,k}x_k$ is the error at time step k , $t_{i,k}$ is the i -th element of T_k , $w_{i,k}$ is the i -th row of W_k , and σ_i are kernel bandwidth. It follows that

$$\arg \min_{x_k} J_{MKCL} = \arg \max_{x_k} J_{MKC} \quad (2.35)$$

with

$$J_{MKC} = \sum_{i=1}^{n+m} 2\sigma_i^2 G_{\sigma_i}(e_{i,k})$$

which reveals that minimizing the multi-kernel correntropy loss J_{MKCL} is equivalent to maximizing the multi-kernel correntropy J_{MKC} . Equation (2.35) can be solved by

$$\frac{\partial J_{MKC}}{\partial x_k} = 0. \quad (2.36)$$

It follows that

$$\sum_{i=1}^{n+m} G_{\sigma_i}(e_{i,k}) w_{i,k}^T (t_{i,k} - w_{i,k}x_k) = 0. \quad (2.37)$$

Then, we have

$$x_k = \left(\sum_{i=1}^{n+m} G_{\sigma_i}(e_{i,k}) w_{i,k}^T w_{i,k} \right)^{-1} \left(\sum_{i=1}^{n+m} G_{\sigma_i}(e_{i,k}) w_{i,k}^T t_{i,k} \right). \quad (2.38)$$

Since $e_{i,k} = t_{i,k} - w_{i,k}x_k$, both sides of (2.38) contains x_k which actually is a fixed-point function. It can be rewritten as

$$x_k = (W_k^T M_k W_k)^{-1} W_k^T M_k T_k \quad (2.39)$$

with

$$M_k = \begin{pmatrix} M_{p,k} & 0 \\ 0 & M_{r,k} \end{pmatrix}$$

$$M_{p,k} = \text{diag}[G_{\sigma_p}(e_{p,k})], M_{r,k} = \text{diag}[G_{\sigma_r}(e_{r,k})] \quad (2.40)$$

$$e_{p,k} = B_{p,k}^{-1}(x_k^- - x_k), e_{r,k} = B_{r,k}^{-1}(y_k - Cx_k)$$

Substituting (2.31) and (2.33) into (2.39), we arrive at

$$(W_k^T M_k W_k)^{-1} = [(B_{p,k}^{-1})^T M_{p,k} B_{p,k}^{-1} + C^T (B_{r,k}^{-1})^T M_{r,k} B_{r,k}^{-1} C]^{-1}. \quad (2.41)$$

Then, using matrix inversion lemma, we obtain

$$\begin{aligned} (W_k^T M_k W_k)^{-1} &= B_{p,k} M_{p,k}^{-1} B_{p,k}^T - B_{p,k} M_{p,k}^{-1} B_{p,k}^T C^T \\ &\times (B_{r,k} M_{r,k}^{-1} B_{r,k}^T + C B_{p,k} M_{p,k}^{-1} B_{p,k}^T C^T)^{-1} C B_{p,k} M_{p,k}^{-1} B_{p,k}^T. \end{aligned} \quad (2.42)$$

Further, we have

$$W_k^T M_k T_k = (B_{p,k}^{-1})^T M_{p,k} B_{p,k}^{-1} x_k^- + C^T (B_{r,k}^{-1})^T M_{r,k} B_{r,k}^{-1} y_k. \quad (2.43)$$

Substituting (2.42) and (2.43) into (2.39), we obtain

$$x_k = x_k^- + \tilde{K}_k (y_k - C x_k^-) \quad (2.44)$$

with

$$\begin{aligned} \tilde{K}_k &= \tilde{P}_k^- C^T (C \tilde{P}_k^- C^T + \tilde{R}_k)^{-1} \\ \tilde{P}_k^- &= B_{p,k} M_{p,k}^{-1} B_{p,k}^T, \quad \tilde{R}_k = B_{r,k} M_{r,k}^{-1} B_{r,k}^T. \end{aligned} \quad (2.45)$$

Finally, the *a posteriori* error covariance can be updated with

$$P_k^+ = (I - \tilde{K}_k C) P_k^- (I - \tilde{K}_k C)^T + \tilde{K}_k R_k \tilde{K}_k^T. \quad (2.46)$$

Notably, (2.44) is a fixed-point equation since \tilde{P}_k^- and \tilde{R}_k are related with x_k . Hence, a fixed-point algorithm can be utilized to solve it. To avoid the numerical singularity of $M_{p,k}$ and $M_{r,k}$, we introduce the following constraints:

$$M_{p,k} \succcurlyeq \alpha I_{n \times n}, \quad M_{r,k} \succcurlyeq \beta I_{m \times m} \quad (2.47)$$

where α and β are two small positive numbers. In a practical application, if the entries of diagonal matrices $M_{p,k}$ and $M_{r,k}$ are smaller than α or β , they will be replaced by scalar α or β , and the fixed-point equation (2.44) will stop iteration. The detailed algorithm is summarized in Algorithm 2.1.

Remark 2.1. We use a threshold ε and constraints (2.47) as a stop condition. This can avoid the numerical matrix singularity in implementation.

Remark 2.2. As discussed in Section 2.1, we use a large bandwidth for the Gaussian channel (e.g., 10^8), and a relative small bandwidth for the heavy-tailed channel, so that the CIM in Gaussian channel is similar to an ℓ_2 norm, while the CIM in non-Gaussian channel changes from an ℓ_2 norm to an ℓ_0 norm.

Theorem 2.4. The MKMCKF is identical to the traditional KF if we set all bandwidths $\sigma_i \rightarrow \infty$ and is equivalent to the MCKF if we set all bandwidths to be the same with $\sigma_i = \sigma$.

Algorithm 2.1 MKMCKF

- 1: **Step 1: Initialization:**
- 2: Choose bandwidth vectors σ_p, σ_r , lower bound values α, β , and a threshold ϵ .
- 3: **Step 2: State Prediction:**
- 4: $\hat{x}_k^- = A\hat{x}_{k-1}^+$
- 5: $P_k^- = AP_{k-1}^+A^T + Q_k$
- 6: Obtain $B_{p,k}$ with $P_k^- = B_{p,k}B_{p,k}^T$
- 7: Obtain $B_{r,k}$ with $R_k = B_{r,k}B_{r,k}^T$
- 8: **Step 3: State Update:**
- 9: $\hat{x}_{k,0}^+ = \hat{x}_k^-$
- 10: **while** $\frac{\|\hat{x}_{k,t}^+ - \hat{x}_{k,t-1}^+\|}{\|\hat{x}_{k,t-1}^+\|} > \epsilon, M_{p,k} \succcurlyeq \alpha I_{n \times n}, M_{r,k} \succcurlyeq \beta I_{m \times m}$ **do** $\triangleright t$ starts from 1
- 11: $\hat{x}_{k,t}^+ = \hat{x}_k^- + \tilde{K}_{k,t}(y_k - C\hat{x}_k^-)$
- 12: $\tilde{K}_{k,t} = \tilde{P}_k^- C^T (C\tilde{P}_k^- C^T + \tilde{R}_k)^{-1}$
- 13: $\tilde{P}_k^- = B_{p,k} M_{p,k}^{-1} B_{p,k}^T$
- 14: $\tilde{R}_k = B_{r,k} M_{r,k}^{-1} B_{r,k}^T$
- 15: $M_{p,k} = \text{diag}(G_{\sigma_p}(e_{p,k}))$
- 16: $M_{r,k} = \text{diag}(G_{\sigma_r}(e_{r,k}))$
- 17: $e_{p,k} = B_{p,k}^{-1} \hat{x}_k^- - B_{p,k}^{-1} \hat{x}_{k,t-1}^+$
- 18: $e_{r,k} = B_{r,k}^{-1} y_k - B_{r,k}^{-1} C \hat{x}_{k,t-1}^+$
- 19: $t = t + 1$
- 20: **end while**
- 21: $P_k^+ = (I - \tilde{K}_k C) P_k^- (I - \tilde{K}_k C)^T + \tilde{K}_k R_k \tilde{K}_k^T$

Proof. As all $\sigma_i \rightarrow \infty$, we have $M_k = I$. Then, the MKMCKF is equivalent to the KF. As $\sigma_i = \sigma$, we have $M_{p,k} = \text{diag}[G_\sigma(e_{p,k})]$ and $M_{r,k} = \text{diag}[G_\sigma(e_{r,k})]$. Then, the MKMCKF is identical to the MCKF. \square

Theorem 2.5. Denote the measurement matrix as $C = [C1_{m \times f}, C2_{m \times l}]$ where $n = f + l$ and denote the process error as $e_{p,k} = [e_{f,k}^T, e_{l,k}^T]^T$. In the case that $C2_{m \times l}$ is a zero matrix, the error partition $e_{l,k}$ is always a zero vector in the MKMCKF.

Proof. Based on line 17 in Algorithm 2.1, we have

$$e_{p,k} = B_{p,k}^{-1} \hat{x}_k^- - B_{p,k}^{-1} \hat{x}_{k,t-1}^+ \quad (2.48)$$

where t is the iteration number at time step k . If $t = 1$, we have $\hat{x}_{k,0}^+ = \hat{x}_k^-$. Then, e_p is a

zero vector and $\tilde{P}_k^- = P_k^-$. If $t > 1$, based on line 6, 11, 12, 17 in Algorithm 2.1, we have

$$\begin{aligned}
e_{p,k} &= B_{p,k}^{-1}(\hat{x}_k^- - \hat{x}_{k,t-1}^+) \\
&= -B_{p,k}^{-1}\tilde{K}_{k,t-1}(y_k - C\hat{x}_k^-) \\
&= -B_{p,k}^{-1}\tilde{P}_k^-C^T(C\tilde{P}_k^-C^T + \tilde{R}_k)^{-1}(y_k - C\hat{x}_k^-) \\
&= -M_{p,k}^{-1}B_{p,k}^TC^T(C\tilde{P}_k^-C^T + \tilde{R}_k)^{-1}(y_k - C\hat{x}_k^-)
\end{aligned} \tag{2.49}$$

Since $M_{p,k}^{-1}$ is diagonal, $B_{p,k}^T$ is an upper triangular matrix, and C^T contains l row zero vectors, it follows that $M_{p,k}^{-1}B_{p,k}^TC^T$ can be written as a block matrix with

$$M_{p,k}^{-1}B_{p,k}^TC^T = \begin{pmatrix} \Lambda_{f \times m} \\ 0_{l \times m} \end{pmatrix}. \tag{2.50}$$

Denote $\eta = (C\tilde{P}_k^-C^T + \tilde{R}_k)^{-1}(y_k - C\hat{x}_k^-) \in \mathbb{R}^m$. Then, we arrive at

$$e_{p,k} = \begin{pmatrix} e_{f,k} \\ e_{l,k} \end{pmatrix} = \begin{pmatrix} \Lambda_{f \times m} \\ 0_{l \times m} \end{pmatrix} \eta = \begin{pmatrix} e_{f,k} \\ 0_{l,k} \end{pmatrix} \tag{2.51}$$

where $e_{l,k} = 0_{l,k}$. This completes the proof. \square

Remark 2.3. In the situation of Theorem 2.5, we denote the process kernel vector as $\sigma_p = [\sigma_f^T, \sigma_l^T]^T$. Then, the selection of σ_l becomes meaningless because we always have $e_{l,k} = 0$. With this special measurement matrix, the performance of the MKMCKF will degenerate. To avoid this situation, we can apply the state transformation techniques, exchange the state position (see Section 2.4.1), or use other decomposition methods (such as principal square root decomposition) in Algorithm 2.1.

2.4 Disturbance Mitigation

In many applications, systems contain unknown process disturbance or measurement disturbance. These disturbances are usually difficult to be estimated due to the lack of accurate disturbance models. In this section, we employ the MKMCKF to address this problem.

2.4.1 Systems with Process Disturbance

We consider the following linear model with unknown process disturbance:

$$\begin{aligned} x_{k+1} &= Ax_k + \Gamma d_k + w_k \\ y_k &= Cx_k + v_k \end{aligned} \tag{2.52}$$

where d_k is the unknown disturbance, matrix Γ maps the disturbance to the state, and w_k and v_k are Gaussian noises. To eliminate the influence of d_k on state x_k , we augment the state as $\bar{x}_k = [x_k^T, d_k^T]^T$ and construct an augmented state Kalman filter (ASKF). Since we have no prior knowledge about the disturbance, a conventional way is to assume that $d_{k+1} = d_k$. Correspondingly, the system dynamics can be written as

$$\begin{aligned} \bar{x}_{k+1} &= \bar{A}\bar{x}_k + \bar{w}_k \\ y_k &= \bar{C}\bar{x}_k + \bar{v}_k \end{aligned} \tag{2.53}$$

with

$$\bar{A} = \begin{bmatrix} A & \Gamma \\ 0 & I \end{bmatrix}, \quad \bar{C} = \begin{bmatrix} C & 0 \end{bmatrix} \tag{2.54}$$

where \bar{w}_k and \bar{v}_k are Gaussian noises with covariance matrices

$$Q_k = E[\bar{w}_k \bar{w}_k^T], \quad R_k = E[\bar{v}_k \bar{v}_k^T]. \tag{2.55}$$

Notably, the modeling of d_k is not accurate and the unmodeled dynamics can be seen as non-Gaussian process noises. According to Theorem 2.5, the performance of the MKM-CKF will degenerate due to the special form of \bar{C} . To this end, we exchange the position of the state and the disturbance and write the new state as $\bar{x}_k^* = [d_k^T, x_k^T]^T$. Then, the system dynamics can be written as

$$\begin{aligned} \bar{x}_{k+1}^* &= \bar{A}^*\bar{x}_k^* + w_k^* \\ y_k &= \bar{C}^*\bar{x}_k^* + v_k^* \end{aligned} \tag{2.56}$$

with

$$\bar{A}^* = \begin{bmatrix} I & 0 \\ \Gamma & A \end{bmatrix}, \quad \bar{C}^* = \begin{bmatrix} 0 & C \end{bmatrix} \tag{2.57}$$

where w_k^* is the process noise and v_k^* is the measurement noise. With this new equation, the situation of Theorem 2.5 can be avoided. A performance comparison of the MKMCKF using equation (2.53) and (2.56) is shown in simulations 2.5.1 and 2.5.2.

2.4.2 Systems with Measurement Disturbance

We consider the following linear system with unknown measurement disturbance:

$$\begin{aligned} x_{k+1} &= Ax_k + w_k \\ y_k &= Cx_k + \Omega d_k + v_k \end{aligned} \tag{2.58}$$

where d_k is unknown measurement disturbance and Ω maps the disturbance to the measurement. To eliminate the influence of d_k , we augment the disturbance as a new state $\bar{x}_k = [x_k^T, d_k^T]^T$. Moreover, we assume that the disturbance has a decayed first-order Markov model with a coefficient $0 < c < 1$. Thus, we have

$$\begin{aligned} \bar{x}_{k+1} &= \bar{A}\bar{x}_k + \bar{w}_k \\ y_k &= \bar{C}\bar{x}_k + \bar{v}_k \end{aligned} \tag{2.59}$$

with

$$\bar{A} = \begin{bmatrix} A & 0 \\ 0 & c \cdot I \end{bmatrix}, \quad \bar{C} = \begin{bmatrix} C & \Omega \end{bmatrix}$$

where \bar{w}_k is the process noise and \bar{v}_k is the measurement noise. Since the modeling of the disturbance $d_{k+1} = c \cdot d_k$ is not accurate, the performance of the ASKF will degenerate. However, this effect can be mitigated by the MKMCKF. The comparisons of the ASKF, MCKF, MKMCKF, and particle filter (PF) are shown in 2.5.3.

2.5 Simulations

In this section, we give three examples to demonstrate the performance of the MKMCKF on rejecting non-Gaussian process noises, eliminating process disturbance, and suppressing measurement disturbance.

2.5.1 Example 1

We consider the following velocity tracking problem with non-Gaussian process noises:

$$\begin{aligned} \begin{bmatrix} x_{1,k+1} \\ x_{2,k+1} \end{bmatrix} &= \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_{1,k} \\ x_{2,k} \end{bmatrix} + \begin{bmatrix} w_{1,k} \\ w_{2,k} \end{bmatrix} \\ y_k &= \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_{1,k} \\ x_{2,k} \end{bmatrix} + v_k \end{aligned} \tag{2.60}$$

where k is time index, T is sampling time with $T = 0.1$ seconds, and $x_k = [x_{1,k}, x_{2,k}]^T$ is the target velocity and acceleration at time step k . The process and measurement noises are given as

$$\begin{aligned} w_{1,k} &\sim 0.9\mathcal{N}(0, 0.01) + 0.1\mathcal{N}(0, 4) \\ w_{2,k} &\sim 0.9\mathcal{N}(0, 0.01) + 0.1\mathcal{N}(0, 100) \\ v_k &\sim \mathcal{N}(0, 0.04). \end{aligned} \quad (2.61)$$

Based on Theorem 2.5, the MKMCKF with equation (2.60) cannot suppress non-Gaussian process noises effectively. To handle this problem, we exchange the position of $x_{1,k}$ and $x_{2,k}$ and write a new state as $x_k^* = [x_{2,k}, x_{1,k}]^T$. Then, the system dynamics can be rewritten as

$$\begin{aligned} \begin{bmatrix} x_{2,k+1} \\ x_{1,k+1} \end{bmatrix} &= \begin{bmatrix} 1 & 0 \\ T & 1 \end{bmatrix} \begin{bmatrix} w_{2,k} \\ w_{1,k} \end{bmatrix} + \begin{bmatrix} w_{2,k} \\ w_{1,k} \end{bmatrix} \\ y_k &= \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} x_{2,k} \\ x_{1,k} \end{bmatrix} + v_k. \end{aligned} \quad (2.62)$$

Five simulations are conducted to compare the performances of the KF, MCKF, MKMCKF1, MKMCKF2, and PF. The former three filters use the dynamic equation (2.60) while the fourth employs (2.62). The initial state, process covariance, measurement covariance, and initial error covariance are set to be the same for the first four filters. The particle number for the PF is $N = 1000$ while the resampling method is systematic resample. As for the kernel bandwidths for the MKMCKF1 and MKMCKF2, we select them based on the properties of the CIM and tune them by trial and error. A total of 500 independent Monte Carlo runs are conducted with 1000 time steps at each run. The error probability density functions of the KF, the MCKF, the MKMCKF1, and the MKMCKF2 are drawn in Figure 2.4. The corresponding RMSEs are summarized in Table 2.1. It can be seen that the MKMCKF2 outperforms the others with non-Gaussian process noises, even including the heavy computation required PF. In addition, one can see that the performance of the MKMCKF2 is better than that of the MKMCKF1, which verifies Remark 2.3.

2.5.2 Example 2

We consider a tracking problem where the target is occasionally disturbed by an unknown force f_k . The target mass is assumed to be $m = 1$ kg and the force follows $f_{k+1} = f_k$

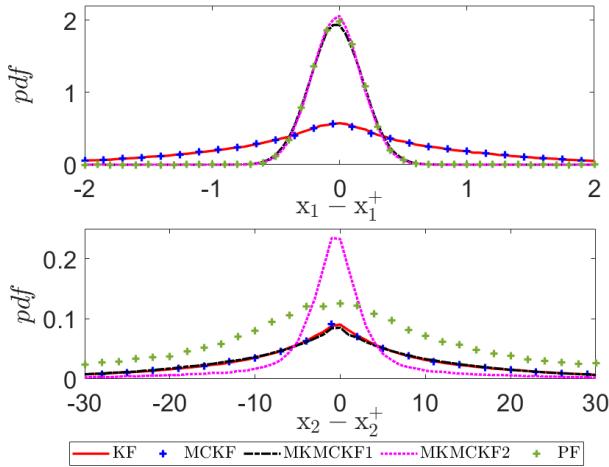


Figure 2.4: Probability densities of the estimation error using the KF, MCKF, MKMCKF1, MKMCKF2 and PF. The kernel bandwidth for the MCKF is $\sigma = 40$. The bandwidths for MKMCKF1 are $\sigma_p = [1.2, 0.5]^T$ and $\sigma_r = 10^4$, and the bandwidths for MKMCKF2 are $\sigma_p = [0.5, 1.2]^T$ and $\sigma_r = 10^4$.

Table 2.1: Performances of the KF, MCKF, MKMCKF1, MKMCKF2 and PF with non-Gaussian Process Noises.

Filters	RMSE of x_1 (m/s)	RMSE of x_2 (m/s 2)
KF	1.072	8.377
MCKF	1.056	8.426
MKMCKF1	0.2050	8.355
MKMCKF2	0.1968	5.003
PF	0.2026	6.3108

since we do not have prior knowledge about the force dynamics. Then, we construct the state as $x_k = [x_{1,k}, x_{2,k}]^T = [v_k, f_k]^T$ where $x_{1,k}$ is the velocity and $x_{2,k}$ is the force. The system dynamics can be written as

$$\begin{aligned} \begin{bmatrix} x_{1,k+1} \\ x_{2,k+1} \end{bmatrix} &= \begin{bmatrix} 1 & \frac{T}{m} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_{1,k} \\ x_{2,k} \end{bmatrix} + \begin{bmatrix} w_{1,k} \\ w_{2,k} \end{bmatrix} \\ y_k &= \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_{1,k} \\ x_{2,k} \end{bmatrix} + v_k. \end{aligned} \quad (2.63)$$

The process noises and measurement noises are both Gaussian with

$$w_{1,k} \sim \mathcal{N}(0, 0.01), w_{2,k} \sim \mathcal{N}(0, 0.01), v(k) \sim \mathcal{N}(0, 0.04). \quad (2.64)$$

In simulation, the unknown force follows

$$f_k = \begin{cases} 20 \sin(0.4\pi kT) + w_{2,k}, & 500 \leq k \leq 600 \\ w_{2,k}, & \text{otherwise} \end{cases} \quad (2.65)$$

with $T = 0.1$ seconds. According to Theorem 2.5, the performance of the MKMCKF will degenerate with $\bar{C} = [1, 0]$ in equation (2.63). To handle this problem, we rewrite the system dynamics by analog to (2.56). Then, we have

$$\begin{aligned} \begin{bmatrix} x_{2,k+1} \\ x_{1,k+1} \end{bmatrix} &= \begin{bmatrix} 1 & 0 \\ \frac{T}{m} & 1 \end{bmatrix} \begin{bmatrix} x_{2,k} \\ x_{1,k} \end{bmatrix} + \begin{bmatrix} w_{2,k} \\ w_{1,k} \end{bmatrix} \\ y_k &= \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} x_{2,k} \\ x_{1,k} \end{bmatrix} + v_k. \end{aligned} \quad (2.66)$$

The new state is $x_k^* = [x_{2,k}, x_{1,k}]^T$ and the situation of Theorem 2.5 is avoided. Five simulations are conducted to verify the performances of the ASKF, MCKF, MKMCKF1, MKMCKF2 and PF. The former three filters utilize dynamic equation (2.63) while the fourth uses (2.66). The process covariance, measurement covariance, initial state, and initial error covariance are set to be the same for the first four filters. The particle number for the PF is $N = 1000$ while the resampling method is systematic resample. The estimate error in a run is shown in Figure 2.5. After 500 independent Monte Carlo runs, the RMSEs are summarized in Table 2.2. It can be seen that the MKMCKF2 outperforms the others remarkably on both the estimate of state $x_{1,k}$ and the disturbance $x_{2,k}$.

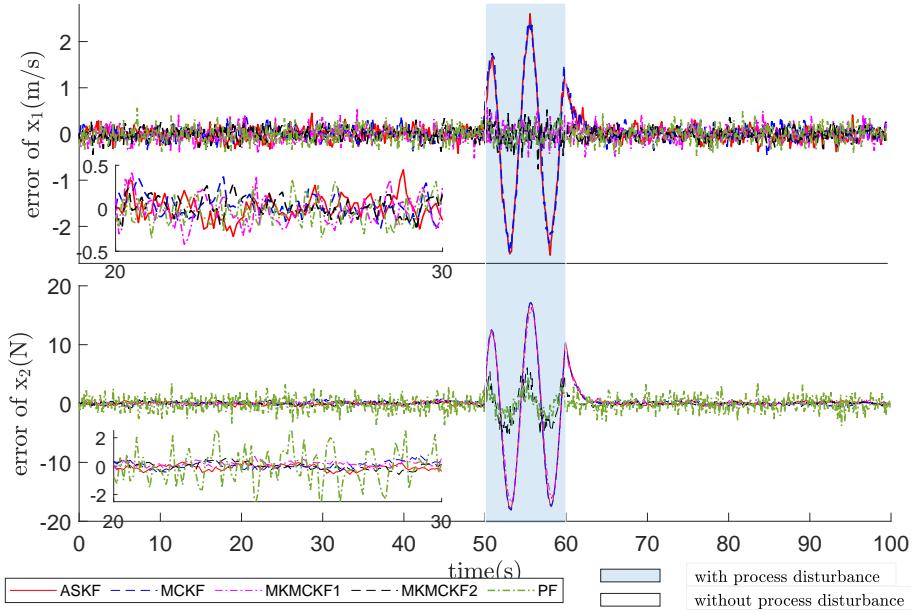


Figure 2.5: Error performances of the ASKF, MCKF, MKMCKF1, MKMCKF2 and PF. The white region is the error without process disturbance while the blue region is the error with process disturbance. The kernel bandwidth for MCKF is $\sigma = 100$ while the bandwidths for MKMCKF1 are $\sigma_p = [1.5, 10^4]^T$ and $\sigma_r = 10^4$, and the bandwidths for MKMCKF2 are $\sigma_p = [10^4, 1.5]^T$ and $\sigma_r = 10^4$. The zoom-in plot at time intervals 20 s to 30 s shows that the performances of the ASKF, MCKF, MKMCKF1 and MKMCKF2 are similar without process disturbance.

Table 2.2: Performances of the ASKF, MCKF, MKMCKF1, MKMCKF2 and PF with Process Disturbance.

Filter	RMSE of x_1 (m/s)	RMSE of x_2 (N)
ASKF	0.5077	3.6949
MCKF	0.5065	3.6916
MKMCKF1	0.1534	4.3119
MKMCKF2	0.1450	1.0390
PF	0.1715	1.2744

2.5.3 Example 3

We consider the following velocity tracking problem where the measurement is contaminated by unknown disturbance:

$$\begin{aligned} \begin{bmatrix} x_{1,k+1} \\ x_{2,k+1} \end{bmatrix} &= \begin{bmatrix} 1 & 0 \\ 0 & c \end{bmatrix} \begin{bmatrix} x_{1,k} \\ x_{2,k} \end{bmatrix} + \begin{bmatrix} w_{1,k} \\ w_{2,k} \end{bmatrix} \\ y_k &= \begin{bmatrix} 1 & 1 \end{bmatrix} \begin{bmatrix} x_{1,k} \\ x_{2,k} \end{bmatrix} + v_k \end{aligned} \quad (2.67)$$

where $x_k = [x_{1,k}, x_{2,k}]^T = [v_k, d_k]^T$ represents the target velocity and the disturbance velocity. The measurement signal contains both the target velocity and the disturbance, and the disturbance dynamics are assumed to be $d_{k+1} = c \cdot d_k$ with $c = 0.8$.

In simulation, the disturbance has the following expression:

$$d_k = \begin{cases} 10 \sin(0.4\pi kT) + w_{2,k}, & 500 \leq k \leq 600 \\ w_{2,k}, & \text{otherwise} \end{cases} \quad (2.68)$$

with $T = 0.1$ seconds. The initial state, initial error covariance, process covariance, and measurement covariance are set to be the same for the ASKF, MCKF, and MKMCKF. Again, the particle number for the PF is $N = 1000$ while the resampling method is systematic resample. The estimation error of these filters in a run is shown in Figure 2.6. After 500 independent Monte Carlo runs, the RMSEs of these filters are summarized in Table 2.3. It can be seen that though the MCKF estimates the state x_1 accurately, its performance on disturbance x_2 is poor. On the contrary, the MKMCKF estimates both the state and the disturbance accurately.

Table 2.3: Performances of the ASKF, MCKF, MKMCKF and PF with Measurement Disturbance.

Filter	RMSE of x_1 (m/s)	RMSE of x_2 (m/s)
ASKF	1.946	1.9925
MCKF	0.5809	2.2284
MKMCKF	0.4111	0.4120
PF	0.5739	0.7847

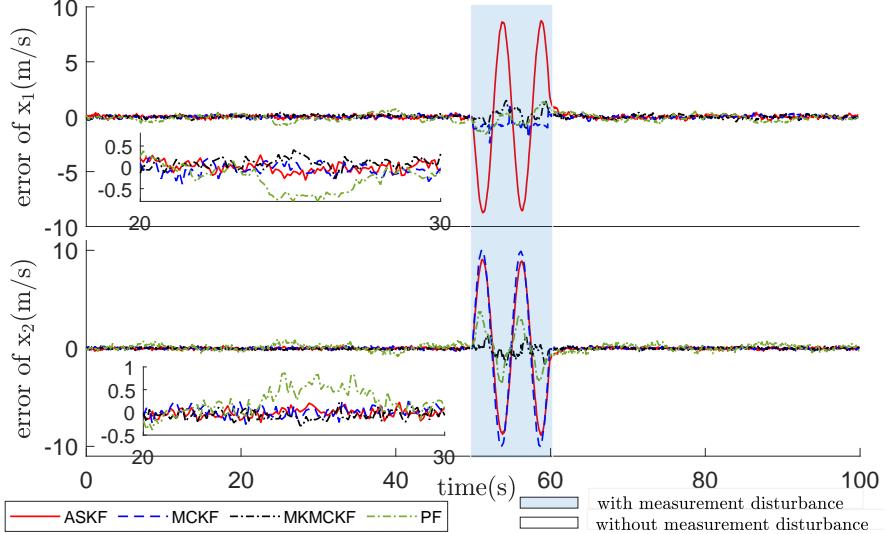


Figure 2.6: Error performances of the ASKF, MCKF, MKMCKF, and PF with measurement disturbance. The kernel bandwidth for MCKF is $\sigma = 5$ while the bandwidths for MKMCKF are $\sigma_p = [10^4, 5]^T$ and $\sigma_r = 10^4$. The zoom-in plot shows that the performances of the ASKF, MCKF, and MKMCKF are similar without disturbance.

2.6 Summary

In this chapter, we extend the definition of correntropy from a single kernel to a multi-kernel version to address an estimation issue that some channels contain heavy-tailed noises. Specifically, we present a multi-kernel correntropy loss (MKCL) and reveal that it is a better objective function for systems partially containing heavy-tailed noises. Moreover, we prove that the MKCL is identical to the MSE when setting all kernel bandwidths to infinity, which establishes a connection between state estimation under Gaussian noises and under non-Gaussian noises. Further, we derive a novel MKMCKF which utilizes MKCL as a loss function. We apply this method to non-Gaussian process noise suppression, process disturbance mitigation, and measurement disturbance elimination. Simulations verify the effectiveness of the proposed algorithm.

2.7 Appendix

2.7.1 Proof of Theorem 2.1

Proof. Equation (2.20) can be rewritten as

$$J_{MKCL}(\mathcal{X}, \mathcal{Y}) = \sum_{i=1}^l 2\sigma_i^2 (1 - E[G_{\sigma_i}(x_i, y_i)]) \quad (2.69)$$

Taking Taylor series expansion of the Gaussian density function, we have

$$J_{MKCL}(\mathcal{X}, \mathcal{Y}) = \sum_{i=1}^l 2\sigma_i^2 \left(1 - E \left[\sum_{n=0}^{\infty} \frac{(-1)^n}{2^n \sigma_i^{2n} n!} |x_i - y_i|^{2n} \right] \right). \quad (2.70)$$

When $\sigma_i \rightarrow \infty$, we obtain

$$\lim_{\sigma_i \rightarrow \infty} J_{MKCL}(\mathcal{X}, \mathcal{Y}) = \sum_{i=1}^l E[|x_i - y_i|^2] = E[\|\mathcal{X} - \mathcal{Y}\|^2] \quad (2.71)$$

This completes the proof. \square

2.7.2 Proof of Theorem 2.2

Proof. Equation (2.21) can be rewritten as

$$CIM(\mathcal{X}, \mathcal{Y}) = \left(\sum_{i=1}^l 2\sigma_i^2 \frac{1}{N} \sum_{k=1}^N (1 - \kappa_{\sigma_i}(e_i(k))) \right)^{1/2}. \quad (2.72)$$

with $\kappa_{\sigma_i}(e_i(k)) = \exp(-\frac{e_i^2(k)}{2\sigma_i^2})$. To be a metric, *CIM* has to fulfill the following properties:

- 1) Negativity: $CIM(\mathcal{X}, \mathcal{Y}) \geq 0$. This can be proved by $0 < \kappa_{\sigma_i}(e_i(k)) \leq 1$.
- 2) Identities of indiscernibles: $CIM(\mathcal{X}, \mathcal{Y}) = 0$ if and only if $\mathcal{X} = \mathcal{Y}$. This can be proved by $\kappa_{\sigma_i}(e_i(k)) = 0$ if and only if $e_i(k) = x_i(k) - y_i(k) = 0$.
- 3) Symmetry: $CIM(\mathcal{X}, \mathcal{Y}) = CIM(\mathcal{Y}, \mathcal{X})$. This can be proved by $\kappa_{\sigma_i}(x_i, y_i) = \kappa_{\sigma_i}(y_i, x_i)$.
- 4) Triangle inequity: $CIM(\mathcal{X}, \mathcal{Z}) \leq CIM(\mathcal{X}, \mathcal{Y}) + CIM(\mathcal{Y}, \mathcal{Z})$. The Gaussian kernel induces a mapping function Φ from the input space to an infinite dimensional reproducing kernel Hilbert space (RKHS) \mathcal{F} with $\kappa_{\sigma}(x_i, y_i) = \langle \Phi(x_i), \Phi(y_i) \rangle_{\mathcal{F}}$. For random pairs $\mathcal{X}_i, \mathcal{Y}_i$, we construct new vectors $\tilde{\mathcal{X}}_i = [\Phi(x_i(1)), \Phi(x_i(2)), \dots, \Phi(x_i(N))]^T$, $\tilde{\mathcal{Y}}_i = [\Phi(y_i(1)), \Phi(y_i(2)), \dots, \Phi(y_i(N))]^T$ in Hilbert space \mathcal{F}^N . Then, the Euclidean distance $D(\tilde{\mathcal{X}}_i, \tilde{\mathcal{Y}}_i)$ has

$$\begin{aligned} D(\tilde{\mathcal{X}}_i, \tilde{\mathcal{Y}}_i) &= (\langle \tilde{\mathcal{X}}_i - \tilde{\mathcal{Y}}_i, \tilde{\mathcal{X}}_i - \tilde{\mathcal{Y}}_i \rangle)^{1/2} \\ &= (\langle \tilde{\mathcal{X}}_i, \tilde{\mathcal{X}}_i \rangle - 2\langle \tilde{\mathcal{X}}_i, \tilde{\mathcal{Y}}_i \rangle + \langle \tilde{\mathcal{Y}}_i, \tilde{\mathcal{Y}}_i \rangle)^{1/2} \\ &= \left(\sum_{k=1}^N \kappa_{\sigma_i}(0) - \sum_{k=1}^N \kappa_{\sigma_i}(x_i(k), y_i(k)) + \sum_{k=1}^N \kappa_{\sigma_i}(0) \right)^{1/2} \\ &= \sqrt{2N} (1 - \hat{C}_{\sigma_i}(\mathcal{X}_i, \mathcal{Y}_i))^{1/2} \end{aligned} \quad (2.73)$$

Then, we have $1 - \hat{C}_{\sigma_i}(\tilde{\mathcal{X}}_i, \tilde{\mathcal{Y}}_i) = \frac{D^2(\tilde{\mathcal{X}}_i, \tilde{\mathcal{Y}}_i)}{2N}$. Correspondingly, we obtain

$$\begin{aligned}
CIM(\mathcal{X}, \mathcal{Z}) &= \left(\sum_{i=1}^l 2\sigma_i^2 (1 - \hat{C}_{\sigma_i}(\tilde{\mathcal{X}}_i, \tilde{\mathcal{Z}}_i)) \right)^{1/2} \\
&= \left(\sum_{i=1}^l \frac{\sigma_i^2}{N} D^2(\tilde{\mathcal{X}}_i, \tilde{\mathcal{Z}}_i) \right)^{1/2} \\
&\leq \left(\sum_{i=1}^l \frac{\sigma_i^2}{N} D^2(\tilde{\mathcal{X}}_i, \tilde{\mathcal{Y}}_i) + D^2(\tilde{\mathcal{Y}}_i, \tilde{\mathcal{Z}}_i) \right)^{1/2} \\
&\leq \left(\sum_{i=1}^l \frac{\sigma_i^2}{N} D^2(\tilde{\mathcal{X}}_i, \tilde{\mathcal{Y}}_i) \right)^{1/2} + \left(\sum_{i=1}^l \frac{\sigma_i^2}{N} D^2(\tilde{\mathcal{Y}}_i, \tilde{\mathcal{Z}}_i) \right)^{1/2} \\
&= CIM(\mathcal{X}, \mathcal{Y}) + CIM(\mathcal{Y}, \mathcal{Z}).
\end{aligned} \tag{2.74}$$

The first inequality utilizes the Euclidean distance property of $D(\tilde{\mathcal{X}}_i, \tilde{\mathcal{Z}}_i) \leq D(\tilde{\mathcal{X}}_i, \tilde{\mathcal{Y}}_i) + D(\tilde{\mathcal{Y}}_i, \tilde{\mathcal{Z}}_i)$ and the second inequality employs the triangle inequality of the ℓ_2 norm $\|p+q\|_2 \leq \|p\|_2 + \|q\|_2$. This completes the proof.

□

2.7.3 Proof of Theorem 2.3

Proof. To derive the KF in (2.6)- (2.10), in equation (2.26), let

$$\frac{\partial J_{KF}}{\partial \hat{x}_k} = 0. \tag{2.75}$$

It follows that

$$-C^T R_k^{-1} (y_k - C\hat{x}_k) + (P_k^-)^{-1} (\hat{x}_k - A\hat{x}_{k-1}) = 0. \tag{2.76}$$

Collecting the terms, we have

$$((P_k^-)^{-1} + C^T R_k^{-1} C) \hat{x}_k = (P_k^-)^{-1} A \hat{x}_{k-1} + C^T R_k^{-1} y_k. \tag{2.77}$$

Right adding and subtracting $C^T R_k^{-1} C \hat{x}_k^-$ and using $\hat{x}_k^- = A \hat{x}_{k-1}$, we arrive at

$$\begin{aligned}
&((P_k^-)^{-1} + C^T R_k^{-1} C) \hat{x}_k \\
&= (P_k^-)^{-1} \hat{x}_k^- + C^T R_k^{-1} y_k + C^T R_k^{-1} C \hat{x}_k^- - C^T R_k^{-1} C \hat{x}_k^- \\
&= ((P_k^-)^{-1} + C^T R_k^{-1} C) \hat{x}_k^- + C^T R_k^{-1} (y_k - C \hat{x}_k^-).
\end{aligned} \tag{2.78}$$

It is equivalent to

$$\hat{x}_k = \hat{x}_k^- + K_k (y_k - C \hat{x}_k^-) \tag{2.79}$$

where $K_k = ((P_k^-)^{-1} + C^T R_k^{-1} C)^{-1} C^T R_k^{-1}$. We use \hat{x}_k^+ to denote the *a posteriori* estimate of x_k . Then, equation (2.79) can be rewritten as

$$\hat{x}_k^- = A\hat{x}_{k-1}^+ \quad (2.80)$$

$$\hat{x}_k^+ = \hat{x}_k^- + K_k(y_k - C\hat{x}_k^-) \quad (2.81)$$

$$K_k = ((P_k^-)^{-1} + C^T R_k^{-1} C)^{-1} C^T R_k^{-1} \quad (2.82)$$

The propagation of error covariance matrices has

$$P_k^- = AP_{k-1}^+A^T + Q_k \quad (2.83)$$

$$P_k^+ = (I - K_k C)P_k^- (I - K_k C)^T + K_k R_k K_k^T \quad (2.84)$$

The *a posteriori* estimate covariance matrix update is the Joseph form of the covariance update equation. Then, we want to prove that (2.8) and (2.82) are equivalent. Using the matrix inversion lemma:

$$(\mathbf{A} + \mathbf{U}\mathbf{C}\mathbf{V})^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{U}(\mathbf{C}^{-1} + \mathbf{V}\mathbf{A}^{-1}\mathbf{U})^{-1}\mathbf{V}\mathbf{A}^{-1}. \quad (2.85)$$

with $\mathbf{A} \rightarrow ((P_k^-)^{-1}$, $\mathbf{U} \rightarrow C^T$, and $\mathbf{C} \rightarrow R_k^{-1}$, and $\mathbf{V} \rightarrow C$. We reformulate (2.82) as

$$\begin{aligned} K_k &= ((P_k^-)^{-1} + C^T R_k^{-1} C)^{-1} C^T R_k^{-1} \\ &= (P_k^- - P_k^- C^T (CP_k^- C^T + R_k)^{-1} CP_k^-) C^T R_k^{-1} \\ &= P_k^- C^T (R_k^{-1} - (CP_k^- C^T + R_k)^{-1} CP_k^- C^T R_k^{-1}) \end{aligned} \quad (2.86)$$

Further, we use the following lemma (see [37], equation (22)-(26))

$$(\mathbf{A} + \mathbf{B})^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{B}(\mathbf{A} + \mathbf{B})^{-1}. \quad (2.87)$$

with $\mathbf{A} \rightarrow R_k$, $\mathbf{B} \rightarrow CP_k^- C^T + R_k$, we obtain

$$\begin{aligned} &(R_k + CP_k^- C^T)^{-1} \\ &= R_k^{-1} - R_k^{-1}(CP_k^- C^T)(R_k + CP_k^- C^T)^{-1}. \end{aligned} \quad (2.88)$$

Due to the symmetry of R_k , R_k^{-1} , $CP_k^- C^T$, and $(R_k + CP_k^- C^T)^{-1}$, we obtain

$$\begin{aligned} &(R_k + CP_k^- C^T)^{-1} = ((R_k + CP_k^- C^T)^{-1})^T \\ &= R_k^{-1} - (R_k + CP_k^- C^T)^{-1}(CP_k^- C^T)R_k^{-1} \end{aligned} \quad (2.89)$$

Substituting (2.89) into (2.86), we have

$$K_k = P_k^- C^T (C P_{k|k-1} C^T + R_k)^{-1} \quad (2.90)$$

Finally, we substitute (2.90) into (2.84), we obtain

$$\begin{aligned} P_k^+ &= P_k^- - P_k^- C^T K_k^T - K_k C P_k^- + K_k (R_k + C P_k^- C^T) K_k^T \\ &= P_k^- - P_k^- C^T K_k^T - K_k C P_k^- + P_k^- C^T K_k^T \\ &= (I - K_k C) P_k^- \end{aligned} \quad (2.91)$$

This completes the proof. \square

Chapter 3

Gait Orientation Estimation

In this chapter, we estimate the orientation of IMUs using the MKMCKF. Specifically, for six-axis IMUs with accelerometers and gyroscopes, we first derive a compact multi-kernel maximum correntropy Kalman filter (CMKMCKF) to minimize the parameter numbers and the computation burden. Then, we design a compact multi-kernel maximum correntropy-based error state Kalman filter (CMKMC-ESKF) for orientation estimation. We apply this algorithm to an embedded system for orientation estimation. Both simulations and experiments verify the effectiveness of the proposed method. For nine-axis IMUs with accelerometers, gyroscopes, and magnetometers, we construct a multi-kernel maximum correntropy Kalman filter for orientation estimation (MKMCKF-OE). To find the optimal kernel bandwidths to suppress external accelerations and magnetic disturbances, we use Bayesian optimization (BO) for this black box optimization task. Experiments verify the superiority of the proposed algorithm. The MKMCKF-OE is also implemented in a low-cost embedded system and its performance is verified by an optical motion capture system in different walking tasks.

3.1 Preliminaries

In this section, we give a brief introduction to sensor models of IMUs, the traditional error state Kalman filter (ESKF) for six-axis IMUs and nine-axis IMUs, effects of external accelerations and magnetic disturbances, and Bayesian optimization (BO).

3.1.1 Sensor Models

We give sensor models of gyroscopes, accelerometers, and magnetometers. Firstly, the gyroscope signal $y_{G,k} \in \mathbb{R}^3$ at time step k is composed of the angular velocity w_k , the offset b_k , and the white noise $v_{G,k}$. The offset is modeled as a Markov process with an additional Gaussian noise $v_{b,k}$. Then, we have

$$\begin{aligned} y_{G,k} &= w_k + b_k + v_{G,k} \\ b_k &= b_{k-1} + v_{b,k} \end{aligned} \tag{3.1}$$

where $v_{G,k} \sim \mathcal{N}(0, Q_{vG})$ and $v_{b,k} \sim \mathcal{N}(0, Q_{vb})$. The accelerometer signal $y_{A,k} \in \mathbb{R}^3$ is composed of the gravity vector in the sensor frame ${}^S g_{A,k}$, the external acceleration ${}^S a_k$ in the sensor frame, and an additional noise $v_{A,k}$. The external acceleration in the global frame ${}^G a_k$ is modeled as a decayed Markov process with coefficient η_a and the noise $v_{a,k}$. Thus, we obtain

$$\begin{aligned} y_{A,k} &= -{}^S g_{A,k} + {}^S a_k + v_{A,k} \\ {}^G a_k &= \eta_a ({}^G a_{k-1}) + v_{a,k} \end{aligned} \tag{3.2}$$

where $0 < \eta_a < 1$, $v_{A,k} \sim \mathcal{N}(0, Q_{vA})$, $v_{a,k} \sim \mathcal{N}(0, Q_{va})$. The free acceleration in the global frame and the sensor frame can be transferred by a rotation matrix ${}^G S R_k$ with

$${}^G a_k = {}^G S R_k ({}^S a_k). \tag{3.3}$$

Finally, the magnetometer signal $y_{M,k} \in \mathbb{R}^3$ is a summation of the magnetic vector ${}^S m_k$, the magnetic disturbance ${}^S d_k$ and the noise $v_{M,k}$. The magnetic disturbance in the global frame ${}^G d_k$ is modeled as a Markov process with a coefficient η_m and an additional noise $v_{d,k}$. Hence, one has

$$\begin{aligned} y_{M,k} &= {}^S m_k + {}^S d_k + v_{M,k} \\ {}^G d_k &= \eta_m ({}^G d_{k-1}) + v_{d,k} \end{aligned} \tag{3.4}$$

where $0 < \eta_m < 1$, $v_{M,k} \sim \mathcal{N}(0, Q_{vM})$ and $v_{d,k} \sim \mathcal{N}(0, Q_{vd})$. Similarly, the disturbance in the sensor frame and the global frame can be transferred by

$${}^G d_k = {}^G S R_k ({}^S d_k). \tag{3.5}$$

One can refer to [51, 106] for more information.

3.1.2 ESKF for Six-axis IMUs and Nine-axis IMUs

The conventional ESKF models the error state rather than the true state. For six-axis IMUs, the error state $x_{\varepsilon,k}$ includes the error orientation $\theta_{\varepsilon,k} \in \mathbb{R}^{3 \times 1}$ (Euler angle), the error offset $b_{\varepsilon,k} \in \mathbb{R}^{3 \times 1}$ and the error acceleration ${}^S a_{\varepsilon,k} \in \mathbb{R}^{3 \times 1}$. The error state transition model has

$$x_{\varepsilon,k} = \Phi_k x_{\varepsilon,k-1} + w_{x,k} \quad (3.6)$$

with

$$x_{\varepsilon,k} = \begin{bmatrix} \theta_{\varepsilon,k} \\ b_{\varepsilon,k} \\ {}^S a_{\varepsilon,k} \end{bmatrix}, w_{x,k} = \begin{bmatrix} w_{\theta,k} \\ w_{b,k} \\ w_{a,k} \end{bmatrix} \quad (3.7)$$

where $w_{\theta,k}$, $w_{b,k}$, $w_{a,k}$ are the orientation noises, offset noises, and acceleration noises, respectively. The process covariance is Q . The error state transition matrix Φ_k is a zero matrix since the *a priori* estimate of $x_{\varepsilon,k}^-$ does not depend on the previous *a posteriori* estimate of $x_{\varepsilon,k-1}^+$. A more detailed description can be found in [51, 106].

The measurement signal $z_{\varepsilon,k} \in \mathbb{R}^{3 \times 1}$ describes the difference of the *a priori* estimate of the gravity vector obtained by the accelerometer ${}^S \widehat{g}_{A,k}$ and by the *a priori* orientation ${}^S \widehat{g}_{G,k}$. Thus, we have

$$z_{\varepsilon,k} = {}^S \widehat{g}_{A,k} - {}^S \widehat{g}_{G,k} = H_k x_{\varepsilon,k} + v_k \quad (3.8)$$

with

$$\begin{aligned} H_k &= \left[-{}^S \widehat{g}_{G,k} \times, \delta t({}^S \widehat{g}_{G,k} \times), I_{3 \times 3} \right] \\ {}^S \widehat{g}_{A,k}^- &= -y_{A,k} + {}^S \widehat{a}_k^- \\ {}^S \widehat{g}_{G,k}^- &= \hat{R}_k^- g \end{aligned} \quad (3.9)$$

where ${}^S \widehat{a}_k^-$ is the *a priori* estimate of the external acceleration, \hat{R}_k^- is the *a priori* estimate of the orientation matrix, g is the universal gravity vector, δt is the sampling time, H_k is the measurement matrix, and v_k is the measurement noise with a covariance matrix Z_k . The orientation $\theta_k \in \mathbb{R}^{3 \times 1}$ can be expressed as $q_k \in \mathbb{R}^{4 \times 1}$ in a quaternion form or $R_k \in \mathbb{R}^{3 \times 3}$ in a direction cosine matrix form. The error covariance at step k is denoted as P_k . The detailed algorithm of ESKF is summarized in Appendix 3.8.1, and more detailed description can be found in [51, 106].

For nine-axis IMUs, the error state transition model has

$$x_{\varepsilon,k+1} = \Phi_k x_{\varepsilon,k} + w_k \quad (3.10)$$

with

$$x_{\varepsilon,k} = \begin{pmatrix} \theta_{\varepsilon,k} \\ b_{\varepsilon,k} \\ {}^S a_{\varepsilon,k} \\ {}^S d_{\varepsilon,k} \end{pmatrix}, w_k = \begin{pmatrix} w_{\theta,k} \\ w_{b,k} \\ w_{a,k} \\ w_{d,k} \end{pmatrix} \quad (3.11)$$

where the subscript ε denotes the difference of the estimated state and the nominal state, the error state $x_{\varepsilon,k}$ includes the error orientation $\theta_{\varepsilon,k} \in \mathbb{R}^3$ (Euler angle), error gyroscope offset $b_{\varepsilon,k} \in \mathbb{R}^3$, error external acceleration in the sensor frame ${}^S a_{\varepsilon,k} \in \mathbb{R}^3$, and error magnetic disturbance in the sensor frame ${}^S d_{\varepsilon,k} \in \mathbb{R}^3$. The noise w_k is assumed to be Gaussian with covariance Q . Notably, the error state transition matrix $\Phi_k \in \mathbb{R}^{12 \times 12}$ is a zero matrix since the *a priori* estimate of $x_{\varepsilon,k}^-$ does not depend on the previous *a posteriori* estimate of $x_{\varepsilon,k-1}^+$ (the knowledge about previous errors has been incorporated in the current nominal state.). A full treatment of this model can refer to [106].

The measurement signals are related with the error state through

$$\begin{aligned} z_{\varepsilon,k} &= \begin{pmatrix} {}^S \hat{g}_{A,k}^- - {}^S \hat{g}_{G,k}^- \\ {}^S \hat{m}_{M,k}^- - {}^S \hat{m}_{G,k}^- \end{pmatrix} + v_k \\ &= H_k x_{\varepsilon,k} + v_k \end{aligned} \quad (3.12)$$

with

$$\begin{aligned} H_k &= \left[\begin{array}{c} -{}^S \hat{g}_{G,k}^- \times, \delta t({}^S \hat{g}_{G,k}^- \times), I_{3 \times 3}, 0_{3 \times 3} \\ -{}^S \hat{m}_{G,k}^- \times, \delta t({}^S \hat{m}_{G,k}^- \times), 0_{3 \times 3}, -I_{3 \times 3} \end{array} \right] \\ {}^S \hat{g}_{A,k}^- &= -y_{A,k} + {}^S \hat{a}_k^-, \quad {}^S \hat{g}_{G,k}^- = \hat{R}_k^-({}^G g_k) \\ {}^S \hat{m}_{M,k}^- &= y_{M,k} - {}^S \hat{d}_k^-, \quad {}^S \hat{m}_{G,k}^- = \hat{R}_k^-({}^G m_k) \end{aligned} \quad (3.13)$$

where ${}^S \hat{g}_{A,k}^-$ and ${}^S \hat{g}_{G,k}^-$ denotes the *a priori* estimate of gravity vector obtained by the accelerometer and by the coordinate transformation, ${}^S \hat{m}_{M,k}^-$ and ${}^S \hat{m}_{G,k}^-$ denotes the *a priori* estimate of geomagnetic vector obtained by the magnetometer model and by the coordinate transformation, ${}^S \hat{a}_k^-$ and ${}^S \hat{d}_k^-$ is the *a priori* estimate of external acceleration and the *a priori* estimate of magnetic disturbance, ${}^G g_k$ and ${}^G m_k$ is the constant gravity vector and

the geomagnetic vector, \hat{R}_k^- is the *a priori* estimate of orientation matrix. The measurement noise is v_k with covariance Z_k , δt is the sampling time, and P_k is the corresponding error covariance matrix. One can refer to [106] for more information.

The orientation $\theta_k \in \mathbb{R}^3$ can be written as a rotation matrix $R_k \in \mathbb{R}^{3 \times 3}$ or a quaternion $q_k \in \mathbb{R}^4$. The detailed algorithm is summarized in Appendix 3.8.2.

3.1.3 Effects of External Acceleration and Magnetic Disturbance

The performance of the ESKF relies heavily on the first-order external acceleration dynamics and the magnetic disturbance in (3.2) and (3.4). For the ESKF in six-axis IMUs, based on equations (3.8), (3.9) and Line 5, 8, 15 in Algorithm 3.4, the *a posteriori* estimate of the external acceleration has

$$\begin{aligned} {}^S\hat{a}_k^+ &= {}^S\hat{a}_k^- - {}^S\hat{a}_{\varepsilon,k}^+ \\ &= \eta_a({}^S\hat{a}_{k-1}^+) - {}^S\hat{a}_{\varepsilon,k}^+ \\ &= \eta_a({}^S\hat{a}_{k-1}^+) - K_k({}^S a_\varepsilon)({}^S\hat{g}_{A,k}^- - {}^S\hat{g}_{G,k}^-) \\ &= \eta_a({}^S\hat{a}_{k-1}^+) - K_k({}^S a_\varepsilon)(-y_{A,k} + \eta_a({}^S\hat{a}_{k-1}^+) - {}^S\hat{g}_{G,k}^-) \\ &= (I - K_k({}^S a_\varepsilon))\eta_a({}^S\hat{a}_{k-1}^+) + K_k({}^S a_\varepsilon)({}^S\hat{g}_{G,k}^- + y_{A,k}) \end{aligned} \quad (3.14)$$

where $K_k({}^S a_\varepsilon)$ is the Kalman gain partition with respect to the external acceleration. One can see that this estimator relies heavily on nominal acceleration model ${}^S\hat{a}_k^- = \eta_a({}^S\hat{a}_{k-1}^+)$. However, in a practical application, the external acceleration dynamics is time-varying with ${}^S a_k = f({}^S a_{k-1}) + w_{a,k}$ where $f(\cdot)$ is an unknown function and $w_{a,k}$ is the noise. In this case, the optimal estimator should be ${}^S\hat{a}_k^+ = (I - K_k({}^S a_\varepsilon))f({}^S\hat{a}_{k-1}^+) + K_k({}^S a_\varepsilon)({}^S\hat{g}_{G,k}^- + y_{A,k})$. A tremendous estimation error can be caused by (3.14) when the discrepancy $f({}^S\hat{a}_{k-1}^+) - \eta_a({}^S\hat{a}_{k-1}^+)$ is large.

Similarly, for the ESKF in nine-axis IMUs, based on equations (3.12), (3.13), and Line 4, 9, 17 in Algorithm 3.5, the *a posteriori* estimate of the magnetic disturbance has

$$\begin{aligned} {}^S\hat{d}_k^+ &= {}^S\hat{d}_k^- - {}^S\hat{d}_{\varepsilon,k}^+ \\ &= \eta_m({}^S\hat{d}_{k-1}^+) - {}^S\hat{d}_{\varepsilon,k}^+ \\ &= \eta_m({}^S\hat{d}_{k-1}^+) - K_k({}^S d_\varepsilon)({}^S\hat{m}_{M,k}^- - {}^S\hat{m}_{G,k}^-) \\ &= \eta_m({}^S\hat{d}_{k-1}^+) - K_k({}^S d_\varepsilon)(y_{M,k} - \eta_m({}^S\hat{d}_{k-1}^+) - {}^S\hat{m}_{G,k}^-) \\ &= (I + K_k({}^S d_\varepsilon))\eta_m({}^S\hat{d}_{k-1}^+) + K_k({}^S d_\varepsilon)({}^S\hat{m}_{G,k}^- - y_{M,k}) \end{aligned} \quad (3.15)$$

where $K_k({}^S d_\varepsilon)$ is the Kalman gain partition with respect to the magnetic disturbance. In a practical application, the magnetic disturbance dynamics is time-varying with ${}^S d_k = f({}^S d_{k-1}) + w_{d,k}$ where $f({}^S d_{k-1})$ is an unknown function and $w_{d,k}$ is the noise. In this case, the optimal estimator should be ${}^S \hat{d}_k^+ = (I + K_k({}^S d_\varepsilon))f({}^S \hat{d}_{k-1}^+) + K_k({}^S d_\varepsilon)({}^S \hat{m}_{G,k}^- - y_{M,k})$. An obvious estimation error can be caused by (3.15) when $f({}^S d_{k-1})$ is far away from $\eta_m {}^S d_{k-1}$.

The unmodeled dynamics of the external acceleration (i.e., $f({}^S \hat{a}_{k-1}^+) - \eta_a {}^S \hat{a}_{k-1}^+$) and the magnetic disturbance (i.e., $f({}^S d_{k-1}) - \eta_m {}^S d_{k-1}$) can be regarded as heavy-tailed noises. In the next section, we will employ the MKMCKF to mitigate these issues.

3.1.4 Bayesian Optimization

We use the BO to explore the optimal kernel bandwidths for the MKMCKF. Suppose we have a function $f : \mathcal{X} \subset \mathbb{R}^l \rightarrow \mathbb{R}$ with $x \in \mathcal{X}$. We try to find the minima point x^* on a domain \mathcal{X} , i.e.,

$$x^* = \arg \min_{x \in \mathcal{X}} f(x). \quad (3.16)$$

In the framework of BO, the distribution f is assumed to be Gaussian with $p(f) = \mathbb{GP}(f; \mu, K)$ where μ is the mean and K is the covariance. With N finite samples $\mathbf{x}_{1:N}$, the Gaussian Process (GP) model is a joint Gaussian $\mathbf{f}(\mathbf{x}_{1:N}) \sim \mathcal{N}(\mu(\mathbf{x}_{1:N}), \mathbf{K})$ where $\mathbf{K}_{i,j} = \kappa(x_i, x_j)$ and $\kappa(\cdot, \cdot)$ is the covariance function. For a new data x_{N+1} , we consider the joint distribution over f with both the old data and new data:

$$\begin{pmatrix} \mathbf{f}(\mathbf{x}_{1:N}) \\ f(x_{N+1}) \end{pmatrix} \sim \mathcal{N}\left(\begin{array}{c} \mu(\mathbf{x}_{1:N}) \\ \mu(x_{N+1}) \end{array}, \begin{bmatrix} \mathbf{K} & \mathbf{k} \\ \mathbf{k}^T & \kappa(x_{N+1}, x_{N+1}) \end{bmatrix}\right) \quad (3.17)$$

where $\mathbf{k} = \kappa(\mathbf{x}_{1:N}, x_{N+1})$. We denote the training set as $\mathcal{D}_N := \{\mathbf{x}_{1:N}, \mathbf{f}(\mathbf{x}_{1:N})\}$. Then, using the standard conditioning rules for Gaussian random variables, we have

$$f(x_{N+1}) | \mathcal{D}_N, x_{N+1} \sim \mathcal{N}(\hat{\mu}(x_{N+1}), \hat{\Sigma}^2(x_{N+1})) \quad (3.18)$$

with

$$\begin{aligned} \hat{\mu}(x_{N+1}) &= \mu(x_{N+1}) + \mathbf{k}^T \mathbf{K}^{-1} (\mathbf{f}(\mathbf{x}_{1:N}) - \mu(\mathbf{x}_{1:N})) \\ \hat{\Sigma}^2(x_{N+1}) &= \kappa(\mathbf{x}_{N+1}, \mathbf{x}_{N+1}) - \mathbf{k}^T \mathbf{K}^{-1} \mathbf{k}. \end{aligned} \quad (3.19)$$

To use a GP, one has to specify the mean function and the covariance function. The mean function is usually set to be zero and the covariance function can be set as Gaussian kernel

with $\kappa(x, x') = \exp(\frac{1}{2}(x - x')^T M^{-1}(x - x'))$. The kernel parameters M can be obtained by empirical Bayesian methods [142]; or see [143, 144] for more information.

Expected improvement (EI) is a common acquisition function to determine which point should be evaluated at the next iteration. Suppose that f^* is the minimum value of f observed so far. Essentially, EI tries to find a candidate point x that improves upon f^* the most with a utility function

$$u(x) = \max_{x \in \mathcal{X}} (0, f^* - f(x)). \quad (3.20)$$

The acquisition function can be written as

$$a_{EI}(x) = \hat{\Sigma}(x) \left(\frac{f^* - \hat{\mu}(x)}{\hat{\Sigma}(x)} \Phi\left(\frac{f^* - \hat{\mu}(x)}{\hat{\Sigma}(x)}\right) + \phi\left(\frac{f^* - \hat{\mu}(x)}{\hat{\Sigma}(x)}\right) \right) \quad (3.21)$$

where Φ is the cumulative distribution function and ϕ is the standard normal probability density function; see [131, 144] for details. Naturally, $a_{EI}(x)$ balances between the exploitation and the exploration. The next candidate point x_{nc} can be selected as

$$x_{nc} = \arg \max_{x \in \mathcal{X}} a_{EI}(x). \quad (3.22)$$

In the applications with the inequity constraint $c(x) \leq \lambda$, we adopt the method in [131] which writes the constrained utility function as

$$u_c(x) = \Delta(x) \max_{x \in \mathcal{X}} (0, f^* - f(x)) = \Delta(x) u(x). \quad (3.23)$$

where $\Delta(x) = \{0, 1\}$ is a feasibility indicator and $\Delta(x) = 1$ if $c(x) \leq \lambda$, and $\Delta(x) = 0$ otherwise. Then, the constrained acquisition function can be written as

$$a_{CEI}(x) = Pr(c(x) \leq \lambda) \cdot a_{EI}(x) \quad (3.24)$$

with

$$Pr(c(x) \leq \lambda) = \int_{-\infty}^{\lambda} p(c(x) | x, \mathcal{D}_N) dc(x) \quad (3.25)$$

The next candidate point can be obtained by

$$x_{nc} = \arg \max_{x \in \mathcal{X}} a_{CEI}(x). \quad (3.26)$$

3.2 Methodology for Six-axis IMUs

To mitigate the bad effects of the external acceleration for six-axis IMUs and simultaneously reduce the parameter numbers of the MKMCKF, we derive a CMKMCKF which

uses infinite bandwidths for Gaussian channels and bandwidths σ for non-Gaussian channels. Then, we apply this method to the conventional ESKF and obtain a CMKMC-ESKF for orientation estimation.

3.2.1 The Compact Multi-kernel Maximum Correntropy Kalman Filter

For a LTI system described in equation (2.1) with some channels containing non-Gaussian noises in (2.11), we use the MKCL as a cost function and derive a novel MKM-CKF in Algorithm 2.1. To avoid involving too much kernel bandwidths, in this section, we apply the infinite bandwidth σ_{inf} for all Gaussian channels, and the bandwidth σ for all non-Gaussian channels. In this situation, equation (2.40) can be replaced by

$$\begin{aligned} M_k &= \begin{pmatrix} M_{p,k} & 0 \\ 0 & M_{r,k} \end{pmatrix} \\ M_{p,k} &= \Lambda_p I_{n \times n} + (I_{n \times n} - \Lambda_p) \text{diag}(G_\sigma(e_p)) \\ M_{r,k} &= \Lambda_r I_{m \times m} + (I_{m \times m} - \Lambda_r) \text{diag}(G_\sigma(e_r)) \\ e_{p,k} &= B_{p,k}^{-1} (x_k^- - x_k), \quad e_{r,k} = B_{r,k}^{-1} (y_k - Cx_k) \end{aligned} \tag{3.27}$$

where Λ_p and Λ_r are diagonal indicator matrices where the diagonal entries are set to 1 for Gaussian channels and 0 for non-Gaussian channels. Then, we can obtain the CMKM-CKF in Algorithm 3.1.

3.2.2 The Compact Multi-kernel Maximum Correntropy-Based Error State Kalman Filter

As illustrated in (3.14), the first order acceleration assumption cannot model the practical acceleration dynamics accurately. The unmodeled dynamics can be regarded as non-Gaussian noises. To mitigate its bad effects, we apply the CMKMCKF to the orientation estimation of six-axis IMUs. The process kernel vector and measurement kernel vector are designed as

$$\begin{aligned} \sigma_p &= [\sigma_{\text{inf}} I_{1 \times 3}, \sigma_{\text{inf}} I_{1 \times 3}, \sigma I_{1 \times 3}]^T \\ \sigma_r &= [\sigma_{\text{inf}} I_{1 \times 3}]^T. \end{aligned} \tag{3.28}$$

Algorithm 3.1 CMKMCKF

```

1: Initialization:
2: Choose a kernel bandwidth  $\sigma$ , a small positive number  $\varepsilon$ 
3: State Prediction:
4:  $\hat{x}_k^- = A\hat{x}_{k-1}^+$ 
5:  $P_k^- = AP_{k-1}^+A^T + Q_k$ 
6: Obtain  $B_{p,k}$  and  $B_{r,k}$  by  $P_k^- = B_{p,k}B_{p,k}^T$  and  $R_k = B_{r,k}B_{r,k}^T$ 
7: State Update:
8:  $\hat{x}_{k,0}^+ = \hat{x}_k^-$ 
9: while  $\frac{\|\hat{x}_{k,t}^+ - \hat{x}_{k,t-1}^+\|}{\|\hat{x}_{k,t-1}^+\|} > \varepsilon$  do  $\triangleright t$  starts from 1
10:  $\hat{x}_{k,t}^+ = \hat{x}_k^- + \tilde{K}_k(y_k - C\hat{x}_k^-)$ 
11:  $\tilde{K}_k = \tilde{P}_k^- C^T (C\tilde{P}_k^- C^T + \tilde{R}_k)^{-1}$ 
12:  $\tilde{P}_k^- = B_{p,k} M_{p,k}^{-1} B_{p,k}^T$ 
13:  $\tilde{R}_k = B_{r,k} M_{r,k}^{-1} B_{r,k}^T$ 
14:  $M_{p,k} = \Lambda_p I_{n \times n} + (I_{n \times n} - \Lambda_p) \text{diag}(G_\sigma(e_{p,k}))$ 
15:  $M_{r,k} = \Lambda_r I_{m \times m} + (I_{m \times m} - \Lambda_r) \text{diag}(G_\sigma(e_{r,k}))$ 
16:  $e_{p,k} = B_{p,k}^{-1} \hat{x}_k^- - B_{p,k}^{-1} \hat{x}_{k,t-1}^+$ 
17:  $e_{r,k} = B_{r,k}^{-1} y_k - B_{r,k}^{-1} C \hat{x}_{k,t-1}^+$ 
18:  $t = t + 1$ 
19: end while
20:  $P_k^+ = (I - \tilde{K}_k C) P_k^- (I - \tilde{K}_k C)^T + \tilde{K}_k R \tilde{K}_k^T$ 

```

since the non-Gaussian noises only exist in the external acceleration channels. Then, we construct a CMKMC-ESKF for orientation estimation and the details are summarized in Algorithm 3.2. One can see that the calculation of $B_{r,k}$, $e_{r,k}$, and $\tilde{Z}_k = B_{r,k} M_{r,k}^{-1} B_{r,k}^T$ can be omitted in Algorithm 3.2 due to the designated kernel bandwidths in (3.28), which reduces the computation complexity of the CMKMC-ESKF and is beneficial to implementation.

Remark 3.1. *The KF is a well-known optimal estimator for linear systems with Gaussian noises. However, its performance would degenerate with non-Gaussian noises. On the contrary, the conventional MCKF can suppress non-Gaussian noises. However, its performance on the Gaussian channels is unsatisfactory (see the simulation in Section 3.3.1). The CMKMCKF combines the advantages of the KF and the MCKF, which employs σ_{inf} to suppress Gaussian noises and σ to reject non-Gaussian noises.*

Algorithm 3.2 CMKMC-ESKF

```

1: Initialization:
2: Choose a bandwidth  $\sigma$  and a small positive number  $\varepsilon$ 
3: State Prediction:
4:  $\hat{q}_k^- = \hat{q}_{k-1}^+ \Delta q(\hat{\omega}_k^- \delta t)$ 
5:  $\hat{\omega}_k^- = y_{G,k} - \hat{b}_k^-$ 
6:  $\hat{b}_k^- = \hat{b}_{k-1}^+$ 
7:  ${}^s\hat{a}_k^- = \eta_a({}^s\hat{a}_{k-1}^+)$ 
8: Error State Propagation:
9:  $P_k^- = f(P_{k-1}^+, Q)$  ▷ details in (3.32)
10: Obtain  $B_{p,k}$  by  $B_{p,k}B_{p,k}^T = P_k^-$ 
11: Initialize  $x_{\varepsilon,k,0}^+ = x_{\varepsilon,k}^- = 0$ 
12: while  $\frac{x_{\varepsilon,k,t}^+ - x_{\varepsilon,k,t-1}^+}{x_{\varepsilon,k,t-1}^+} \leq \varepsilon$  do ▷ t starts from 1
13:  $x_{\varepsilon,k,t}^+ = x_{\varepsilon,k}^- + \tilde{K}_k z_{\varepsilon,k}$  ▷  $z_{\varepsilon,k}$  is shown in (3.8)
14:  $\tilde{K}_k = \tilde{P}_k^- H_k^T S_k^{-1}$  ▷  $H_k$  is shown in (3.9)
15:  $S_k = H_k \tilde{P}_k^- H_k^T + Z_k$ 
16:  $\tilde{P}_k^- = B_{p,k} M_{p,k}^{-1} B_{p,k}^T$ 
17:  $M_{p,k} = \Lambda_p I_{n \times n} + (I_{n \times n} - \Lambda_p) \text{diag}(G_\sigma(e_{p,k}))$ 
18:  $e_{p,k} = B_{p,k}^{-1} x_{\varepsilon,k}^- - B_{p,k}^{-1} x_{\varepsilon,k,t-1}^+$ 
19:  $t = t + 1$ 
20: end while
21: State Update:
22:  $\hat{q}_k^+ = \hat{q}_k^- \Delta q(-\hat{\theta}_{\varepsilon,k}^+)$ 
23:  $\hat{b}_k^+ = \hat{b}_k^- - \hat{b}_{\varepsilon,k}^+$ 
24:  ${}^s\hat{a}_k^+ = {}^s\hat{a}_k^- - {}^s\hat{a}_{\varepsilon,k}^+$ 
25:  $P_k^+ = (I - \tilde{K}_k H_k) P_k^-$ 

```

3.3 Simulations and Experiments for Six-axis IMUs

Some simulations and experiments are conducted to verify the effectiveness of the CMKMC-ESKF.

3.3.1 Influence of External Accelerations in Simulation

A simulation is designed to investigate the influence of impulsive and sinusoidal external accelerations. Assume the IMU is static with its z axis downward. The external acceleration is added to the y axis. The gyroscope readings and the accelerometer readings are shown in Figure 3.1. The pitch is the most affected by the external acceleration. The corresponding pitch errors and the acceleration errors using the GD, ESKF, MCKF, and CMKMC-ESKF are shown in Figure 3.2. The kernel bandwidths for the MCKF and

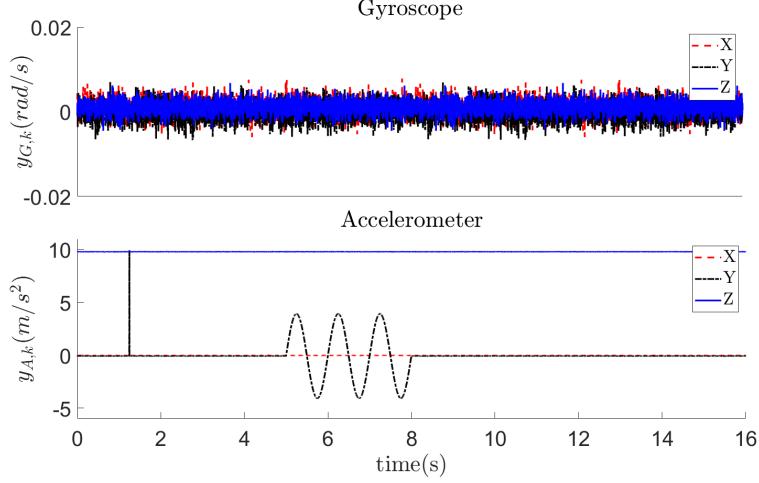


Figure 3.1: The gyroscope signals and accelerometer signals in simulation. The impulsive acceleration is added at 1.2 s while the sinusoidal external acceleration is added at the interval of 4.5 s to 8.0 s.

CMKMC-ESKF are the same with $\sigma = 0.3$. Note that the GD does not model the external acceleration. Therefore, it does not estimate the acceleration error. In Figure 3.2, one can see that though MCKF performs well for the orientation estimation, its performance on the acceleration estimation is the worst since it uses a universal kernel bandwidth for all channels, which verifies Remark 3.1. On the contrary, the CMKMC-ESKF performs well on both estimations. In addition, although the robustness of ESKF can be strengthened by using bigger measurement covariance, it decreases the convergence rate significantly (see the close-up of Figure 3.2), which indicates that the conventional ESKF is incapable of this problem.

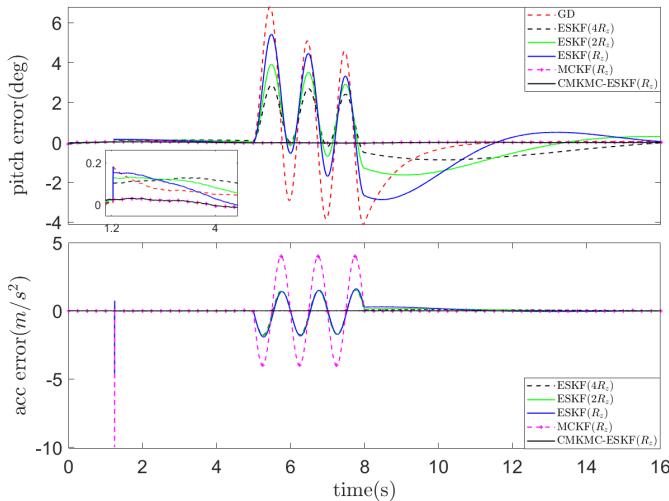


Figure 3.2: Performances of the different algorithms with impulsive and sinusoidal accelerations. The close-up of the pitch error is shown on the left top of the figure.

3.3.2 Dynamic Impact Test

Xsens MTI-670 is utilized to test the effectiveness of the CMKMC-ESKF with impulsive accelerations. The IMU is mounted to a rotating shaft driven by a motor. A rod is used to restrict the rotational range and the impulsive acceleration is generated by the link hitting the rod. The experimental setup is shown in Figure 3.3(a) and the sensor signals are shown in Figure 3.3(b).

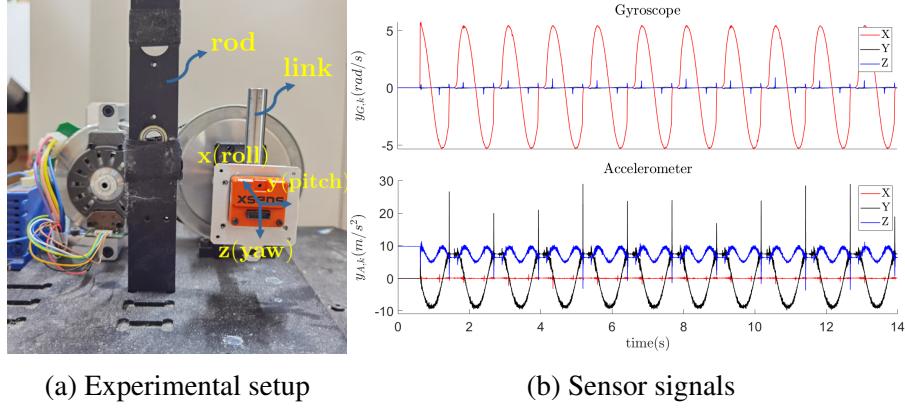


Figure 3.3: The experimental setup of dynamic impact test and the sensor signals.

The true angle is obtained by the optical encoder and the corresponding error performances are shown in Figure 3.4. The root-mean-square errors (RMSEs) for the GD and ESKF are 0.65 deg and 0.67 deg, while it is a mere 0.34 deg for the CMKMC-ESKF. The RMSEs and maximum errors (MEs) of different algorithms are summarized in Table 3.1. One can see that the CMKMC-ESKF is significantly better than the GD and ESKF.

Table 3.1: Error Performances of Different Algorithms with Impulsive Accelerations.

Algorithm	GD	ESKF	CMKMC-ESKF
RMSE (deg)	0.6520	0.6734	0.3425
ME (deg)	1.4303	1.3371	0.8323

3.3.3 External Acceleration Test

The IMU is mounted on the end effector of a linear actuator. The commanded trajectory is sinusoidal with an amplitude of 4 mm and a frequency of 2 Hz. The experimental setup is shown in Figure 3.5(a) while the corresponding sensor signals are shown in Figure 3.5(b).

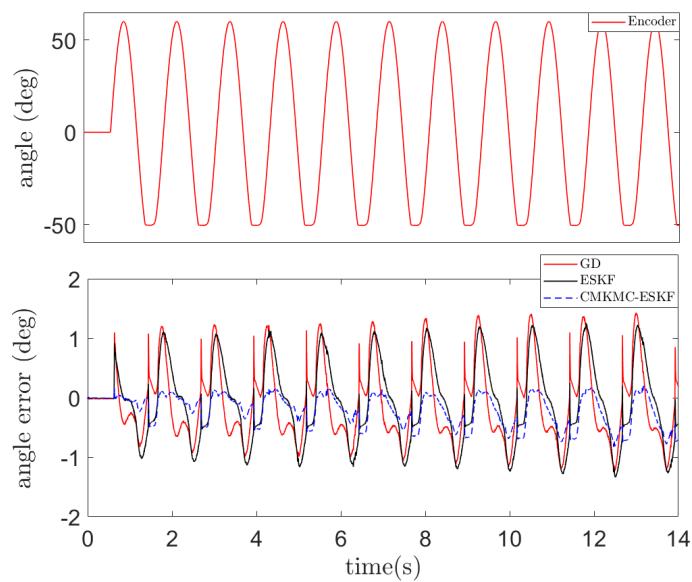


Figure 3.4: The rotational angle obtained by an encoder and the angle errors of the GD, ESKF, and CMKMC-ESKF.

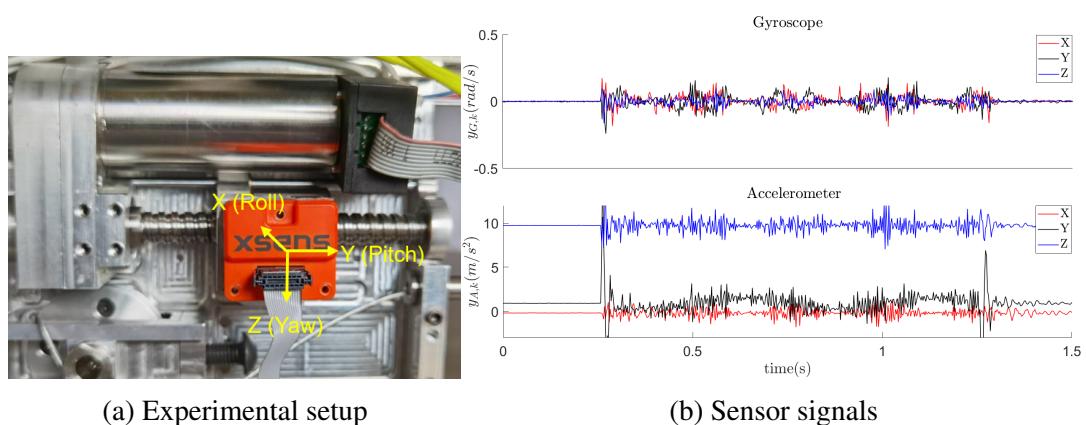


Figure 3.5: The experimental setup of external acceleration test and sensor signals.

The true orientation on the y axis is zero since there is no rotation during the experiment. The ground truth translation distance is obtained by an encoder while the estimated distance by IMUs is obtained by double integration of the external acceleration. The corresponding angle error and position error are shown in Figure 3.6 (note that the GD does not model the external acceleration), where one can see that the CMKMC-ESKF outperforms the GD and ESKF significantly.

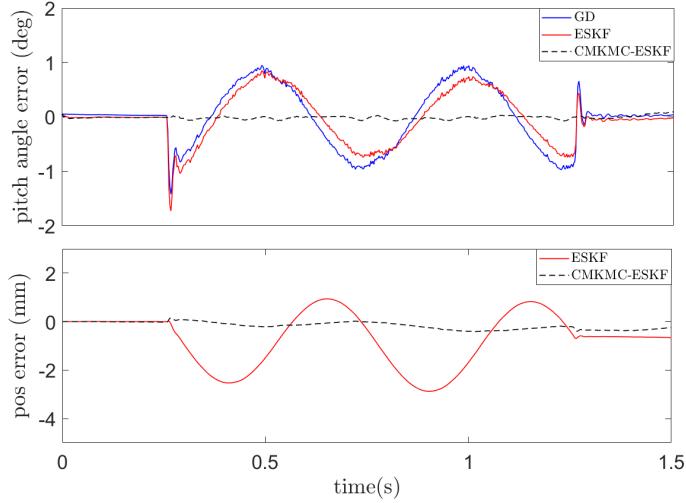


Figure 3.6: Angle errors and position errors of the ESKF and CMKMC-ESKF with external accelerations.

3.3.4 Exoskeleton Orientation Estimation

To further investigate the performance of the proposed algorithm on a microprocessor for orientation estimation. We apply our algorithm to a self-made low-cost IMU. To minimize the computation cost, the maximum iteration number in Algorithm 3.2 is set to 2 (from Line 12 to Line 20). The IMU is attached to the shank of an exoskeleton robot and the exoskeleton gait is designed to imitate human walking. The external acceleration grows with gait frequency f since the IMU has both rotations and translations when walking. A total of 5 frequencies ($f = 0.05, 0.2, 0.4, 0.6$, and 0.8 Hz) are involved in the experiments to investigate the performance of the CMKMC-ESKF with different levels of external accelerations. The experimental setup is shown in Figure 3.7(a) and the sensor signals at $f = 0.8$ Hz are shown in Figure 3.7(b). The sampling frequency of the IMU is 500 Hz.

The ground truth angle is obtained by encoders of the exoskeleton. When $f = 0.8$ Hz,

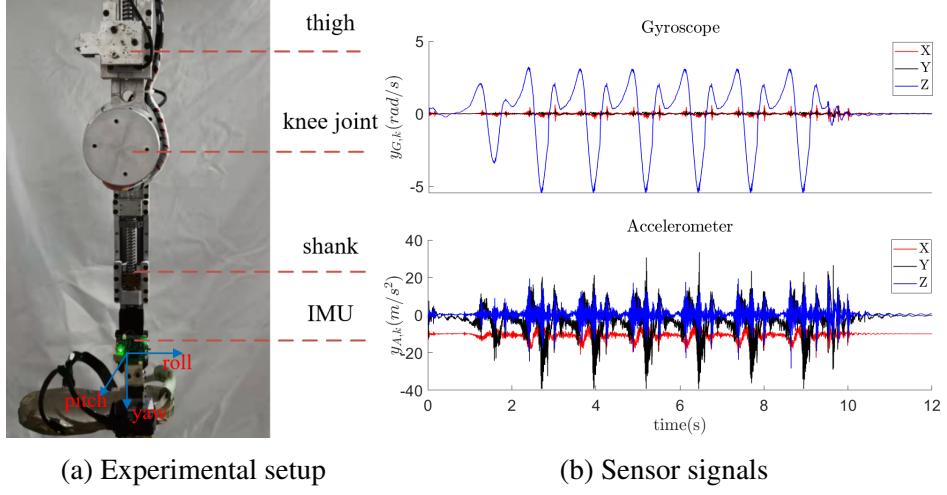


Figure 3.7: The experimental setup and sensor signals with $f = 0.8$ Hz.

Table 3.2: Error Performances of Different Algorithms with Different Gait Frequencies.

f (Hz)	RMSE (deg)			ME (deg)		
	GD	ESKF	CMKMC-ESKF	GD	ESKF	CMKMC-ESKF
$f = 0.05$	0.7452	0.5290	0.4786	1.1416	1.1089	0.9867
$f = 0.2$	0.7321	0.3862	0.3268	1.5536	0.9738	0.8233
$f = 0.4$	0.9707	0.7006	0.1977	2.5580	1.9172	0.5786
$f = 0.6$	1.9972	1.3515	0.3609	5.2197	3.7455	1.0701
$f = 0.8$	1.8236	1.2210	0.3641	4.7885	3.6346	1.1747

the ground truth angle and the corresponding angle error of different algorithms are shown in Figure 3.8. We can see that there is an obvious error for the GD (4.79 deg) and ESKF (3.63 deg), while it is less than 1.17 deg for the CMKMC-ESKF. The error performances with different frequencies are summarized in Table 3.2. One can see that the performance of the CMKMC-ESKF is better than the GD and the ESKF in all frequencies, especially with obvious external accelerations ($f \geq 0.4$ Hz).

3.4 Methodology for Nine-axis IMUs

To alleviate the influence of the external acceleration and magnetic disturbance on nine-axis IMUs, in this section, we derive a novel MKMCKF-OE. Further, we tune the hyper-parameters (i.e., kernel bandwidths) with BO to minimize the orientation error.

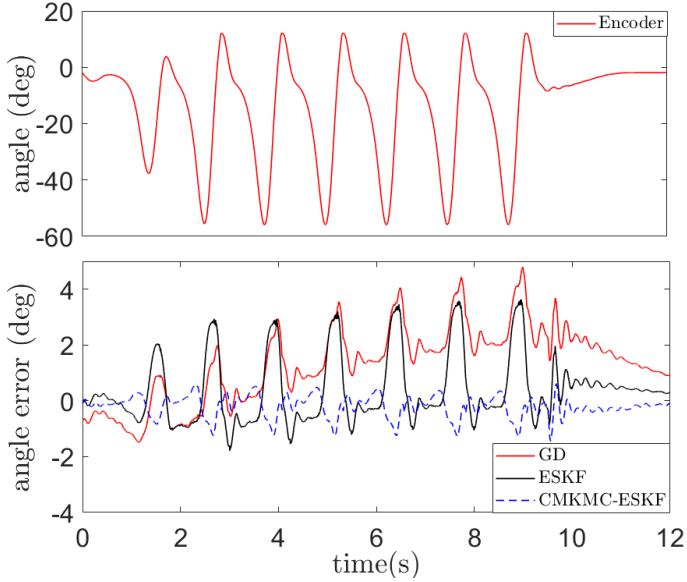


Figure 3.8: The encoder angle and the corresponding errors of the GD, ESKF, and CMKMC-ESKF with $f = 0.8$ Hz.

3.4.1 MKMCKF-OE

According to the MKMCKF in Algorithm 2.1 and the ESKF for nine-axis IMUs in Algorithm 3.5, we can construct the MKMCKF-OE which is shown in Algorithm 3.3. The difference between our algorithm and the traditional ESKF [51, 106] is that we employ the MKCL as a cost function to increase its robustness to disturbance.

3.4.2 Kernel Bandwidths Optimization

In Algorithm 3.3, we have to choose kernel bandwidth vectors σ_p and σ_r . Based on Theorem 2.4, the MKMCKF is identical to the traditional KF if we set all kernel bandwidths $\sigma_i \rightarrow \infty$. Since the KF is optimal for a linear system with Gaussian noises, we can use a unified large kernel bandwidth, i.e., $\sigma_{\text{inf}} = 10^8$, for Gaussian channels. As for non-Gaussian channels, we tune them by BO. In nine-axis IMUs, the non-Gaussian noises exit in the external acceleration and magnetic disturbance channels due to the first-order Markov dynamics assumption in equations (3.2) and (3.4). Therefore, we have

$$\begin{aligned}\sigma_p &= (\sigma_{\text{inf}} I_{1 \times 3}, \sigma_{\text{inf}} I_{1 \times 3}, \sigma_1 I_{1 \times 3}, \sigma_2 I_{1 \times 3})^T \\ \sigma_r &= (\sigma_{\text{inf}} I_{1 \times 3}, \sigma_{\text{inf}} I_{1 \times 3})^T\end{aligned}\tag{3.29}$$

where $I_{1 \times 3}$ is the identity vector, σ_{inf} is a constant, and σ_1 and σ_2 are two parameters to be optimized. The objective function of BO is to minimize the RMSE on all axes with a

Algorithm 3.3 MKMCKF-OE

- 1: **Initialization:**
 - 2: Choose kernel bandwidth vector $\sigma_p \in \mathbb{R}^{12}$, $\sigma_r \in \mathbb{R}^6$.
 - 3: **State Prediction:**
 - 4: $\hat{b}_k^- = \hat{b}_{k-1}^+$
 - 5: ${}^S\hat{a}_k^- = \eta_a({}^S\hat{a}_{k-1}^+)$ ▷ derived from (3.2)
 - 6: ${}^S\hat{d}_k^- = \eta_m({}^S\hat{d}_{k-1}^+)$ ▷ derived from (3.4)
 - 7: $\hat{\omega}_k^- = y_{G,k} - \hat{b}_k^-$
 - 8: $\hat{q}_k^- = \hat{q}_{k-1}^+ \Delta q(\hat{\omega}_k^- \delta t)$
 - 9: **Error State Propagation:**
 - 10: $P_k^- = f(P_{k-1}^+, Q)$, details in (3.34)
 - 11: Obtain $B_{p,k}$ and $B_{r,k}$ by $B_{p,k}B_{p,k}^T = P_k^-$ and $B_{r,k}B_{r,k}^T = Z_k$
 - 12: Initialize $x_{\varepsilon,k}^- = x_{\varepsilon,k,0}^+ = 0$
 - 13: **while** $\frac{x_{\varepsilon,k,t}^+ - x_{\varepsilon,k,t-1}^+}{x_{\varepsilon,k,t-1}^+} \leq \varepsilon$ or $t = 1$ **do** ▷ t starts from 1
 - 14: $x_{\varepsilon,k,t}^+ = x_{\varepsilon,k}^- + \tilde{K}_k z_{\varepsilon,k}$ ▷ $z_{\varepsilon,k}$ is obtained by (3.12)
 - 15: $\tilde{K}_k = \tilde{P}_k^- H_k^T S_k^{-1}$ ▷ H_k is shown in equation (3.13)
 - 16: $S_k = H_k \tilde{P}_k^- H_k^T + \tilde{Z}_k$
 - 17: $\tilde{P}_k^- = T_{p,k} M_{p,k}^{-1} T_{p,k}^T$
 - 18: $\tilde{Z}_k^- = T_{r,k} M_{r,k}^{-1} T_{r,k}^T$
 - 19: $M_{p,k} = \text{diag}(G_{\sigma_p}(e_{p,k}))$
 - 20: $M_{r,k} = \text{diag}(G_{\sigma_r}(e_{r,k}))$
 - 21: $e_{p,k} = B_{p,k}^{-1} x_{\varepsilon,k}^- - B_{p,k}^{-1} x_{\varepsilon,k,t-1}^+$
 - 22: $e_{r,k} = B_{r,k}^{-1} (z_{\varepsilon,k} - H_k x_{\varepsilon,k,t-1}^+)$
 - 23: **end while**
 - 24: **State Update:**
 - 25: Obtain $\hat{\theta}_{\varepsilon,k}^+$, ${}^S\hat{a}_{\varepsilon,k}^+$, $\hat{b}_{\varepsilon,k}^+$, ${}^S\hat{d}_{\varepsilon,k}^+$ from $x_{\varepsilon,k,t}^+$ ▷ based on (3.11)
 - 26: $\hat{q}_k^+ = \hat{q}_k^- \Delta q(-\hat{\theta}_{\varepsilon,k}^+)$
 - 27: $\hat{b}_k^+ = \hat{b}_k^- - \hat{b}_{\varepsilon,k}^+$
 - 28: ${}^S\hat{a}_k^+ = {}^S\hat{a}_k^- - {}^S\hat{a}_{\varepsilon,k}^+$
 - 29: ${}^S\hat{d}_k^+ = {}^S\hat{d}_k^- - {}^S\hat{d}_{\varepsilon,k}^+$ ▷ if $M_J = 0$ in (3.35)
 - 30: update ${}^G m_k$ using (3.36), (3.37) ▷ if $M_J = 0$ in (3.35)
 - 31: $P_k^+ = (I - \tilde{K}_k H_k) P_k^- (I - \tilde{K}_k H_k)^T + \tilde{K}_k Z_k \tilde{K}_k^T$
-

constraint on the yaw:

$$\begin{aligned} \arg \min_{\sigma_1, \sigma_2 \in \mathcal{Z}} J_{BO} &= \arg \min_{\sigma_1, \sigma_2 \in \mathcal{Z}} \frac{1}{3m} \sum_{i=1}^3 \sum_{j=1}^m \underbrace{\frac{1}{n} \sqrt{\sum_{k=1}^n (\hat{\theta}_{i,j,k}(\sigma_1, \sigma_2) - \theta_{i,j,k})^2}}_{RMSE} \\ s.t. \quad &\frac{1}{m} \sum_{j=1}^m \frac{1}{n} \sqrt{\sum_{k=1}^n (\hat{\theta}_{3,j,k}(\sigma_1, \sigma_2) - \theta_{3,j,k})^2} \leq \lambda \end{aligned} \quad (3.30)$$

where m is the number of training sets, \mathcal{Z} is the domain of kernel bandwidths, $\hat{\theta}_{i,j,k}(\sigma_1, \sigma_2)$ represents the estimated angle under i axis (roll, pitch and yaw) at j -th experiment and time step k , $\theta_{i,j,k}$ represents the corresponding ground truth angle (Euler angle). The constant λ gives an upper RMSE bound on the yaw ($i = 3$) axis. In total, we conduct m experiments to involve different situations, i.e., with or without acceleration disturbance, with or without magnetic disturbance, and different types of movement (linear motion or rotation).

3.5 Experiments

In this section, we first obtain the optimal kernel bandwidths through BO. Then, we validate the performance of MKMCKF-OE in the testing set and compare it with the GD [48], ESKF [51, 106], and IGD [50]. The implementations of the GD, ESKF, and MKMCKF-OE are available at [145].

3.5.1 Data Collection

We use Xsens MTI-670 to collect the raw data. The sampling frequency for accelerometers and gyroscopes is 400 Hz while the value for magnetometers is 100 Hz. The MKMCKF-OE is executed at 100 Hz. The experimental setup is shown in Figure 3.9(a) and Figure 3.9(b) which corresponds to the linear motion and the rotation, respectively. In Figure 3.9(a), the acceleration disturbance A_d is generated by moving the IMU along the rail manually, while the magnetic disturbance M_d is generated by the approaching of the ferromagnetic pliers. In Figure 3.9(b), the IMU is fixed to the shank of an exoskeleton robot where its leg swings to imitate the gait of a human. The gait frequency is denoted by f and the acceleration disturbance grows with f . On the right side of Figure 3.9(b), the magnetic disturbance M_d is added when the pliers are close to the IMU.

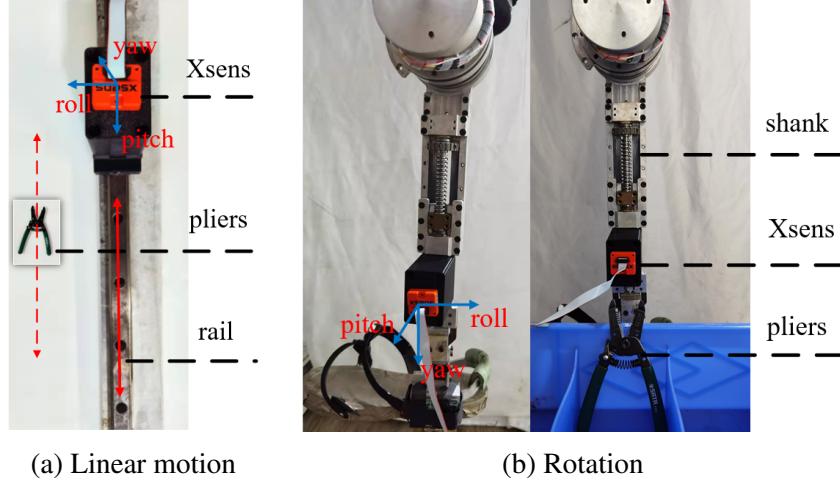


Figure 3.9: The experimental setup and the IMU coordinate system. The movement can be divided into the linear motion in Figure 3.9(a) and the rotation in Figure 3.9(b). The acceleration disturbance A_d is generated by the movement of Xsens itself and the magnetic disturbance M_d is generated by the pliers.

The ground truth orientation for the linear motion is set to be the initial mean angle when the Xsens is static without disturbance. The ground truth pitch angle for rotation is obtained by encoders of the exoskeleton robot, while the roll angle and the yaw angle are obtained by the initial mean angles of the corresponding axes when the Xsens is static since there is no rotation at these axes. In total, we conduct 12 experiments which contain two experiments for linear motion and 10 experiments for rotation. A detailed description of the experiments is summarized in Table 3.3.

Table 3.3: Experimental Description.

Index	Mode	Description	Usage
1	linear motion	A_d, M_d	training
2	rotation	$f = 0.1$ Hz	training
3	rotation	$f = 0.5$ Hz	training
4	rotation	$f = 0.1$ Hz, M_d	training
5	rotation	$f = 0.5$ Hz, M_d	training
6	linear motion	A_d, M_d	testing
7	rotation	$f = 0.2$ Hz	testing
8	rotation	$f = 0.3$ Hz	testing
9	rotation	$f = 0.4$ Hz	testing
10	rotation	$f = 0.2$ Hz, M_d	testing
11	rotation	$f = 0.3$ Hz, M_d	testing
12	rotation	$f = 0.4$ Hz, M_d	testing

3.5.2 Kernel Bandwidths Optimization

A total of 5 experiments are involved in the training set which are experiments 1-5 in Table 3.3. Among them, experiment 5 has the fastest dynamics with both obvious acceleration disturbance and magnetic disturbance. Since the magnetic disturbance affects the yaw angle the most, our objective is to minimize the overall RMSE with a constraint of the yaw error, i.e., $m = 5$ and $\lambda = 2.0$ deg in (3.30). The kernel bandwidth domain is set to be $\mathcal{Z} = [0.1, 10]$, which is capable of majority applications. The initial candidate points are randomly selected and the initial points number is 2. The maximum iteration number for BO is 100. The optimization procedure is conducted on MATLAB 2019b based on equations (3.16)-(3.26). The learned optimal kernel bandwidths are $\sigma_1^* = 1.6188$ and $\sigma_2^* = 0.4234$.

3.5.3 Results

We validate the performance of MKMCKF-OE on the testing set (experiments 6-12) and compared it with the GD [48], ESKF [51], and IGD [50]. The corresponding RMSEs are summarized in Tables 3.4 and 3.5 (we use MKMC to denote MKMCKF-OE). The detailed experimental description can be found in Table 3.3. One can see that the overall performance of the MKMCKF-OE is better than the others, especially on the yaw axis with obvious magnetic disturbance in experiments 6, 10, 11, and 12.

Table 3.4: RMSEs of Different Algorithms on the Roll and Pitch.

Index	Roll (deg)				Pitch (deg)			
	MKMC	GD	ESKF	IGD	MKMC	GD	ESKF	IGD
6	0.3683	1.7992	1.4010	0.7596	0.4257	0.4549	0.9942	0.2891
7	0.1381	0.4985	0.5447	0.1104	0.2833	1.3211	0.8961	0.5396
8	0.2199	0.4077	0.5463	0.1867	0.3811	1.4246	0.9958	0.6671
9	0.1827	0.5400	0.4274	0.4394	0.4629	1.7052	1.3178	1.4895
10	0.1133	1.5521	2.1813	0.0744	0.4228	1.9076	3.3010	0.4441
11	0.2093	1.2445	1.9321	0.1305	0.3022	1.6302	2.9933	0.4670
12	0.1759	1.2293	2.0358	0.1809	0.2420	1.7925	3.2558	0.7003

The Euclidean norm of accelerometer readings and magnetometer readings can reflect the disturbance level of A_d and M_d where their values should be equal to the acceleration of gravity and the local geomagnetic field strength when without disturbance. In experiment 8, the accelerometer norm and the magnetometer norm are shown in Figure 3.10.

Table 3.5: RMSEs of Different Algorithms on the Yaw.

Index	Yaw (deg)			
	MKMC	GD	ESKF	IGD
6	0.3581	4.4078	7.8615	5.1642
7	0.2726	1.9509	1.3227	1.3768
8	0.2604	1.6842	1.0041	1.2250
9	0.4098	1.7093	1.0888	1.0490
10	1.4697	19.7168	20.5097	17.1180
11	1.4911	20.8336	21.4068	18.0540
12	1.1269	24.4531	24.9301	22.2111

One can see that the IMU encounters big acceleration disturbance while small magnetic disturbance. The corresponding orientation errors are depicted in Figure 3.11. One can see that the overall performance of the MKMCKF-OE is superior to the GD, ESKF, and IGD, especially along the yaw and pitch axes.

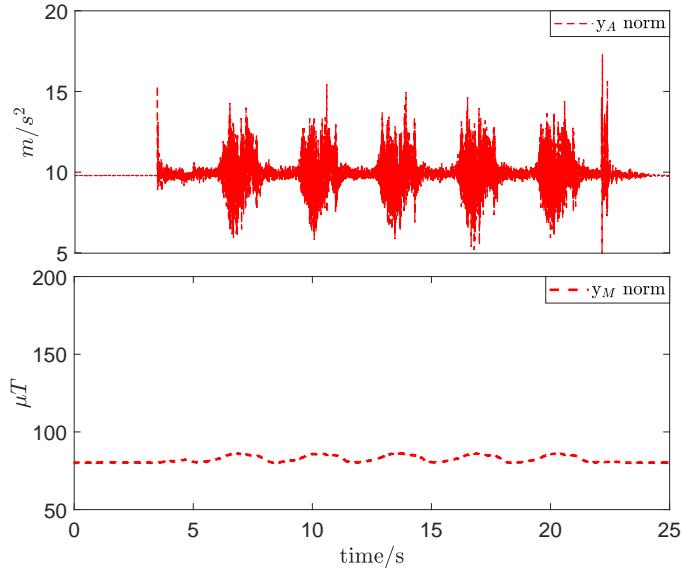


Figure 3.10: Norms of the accelerometer and magnetometer readings in experiment 8.

We also visualize the norm of accelerometer readings and magnetometer readings in experiment 11 where the time interval is divided into regions of no disturbance, static with M_d , rotation with M_d and A_d (see Figure 3.12). The corresponding error performance is shown in Figure 3.13. One can see that a very big yaw error is caused by the magnetic disturbance in the blue region by the GD, ESKF, and IGD, while it is avoided by the MKMCKF-OE. In the yellow region where the IMU rotates with both M_d and A_d , the performance of the MKMCKF-OE is again significantly better than the others, which reveals that our algorithm is very robust to both acceleration disturbance and magnetic

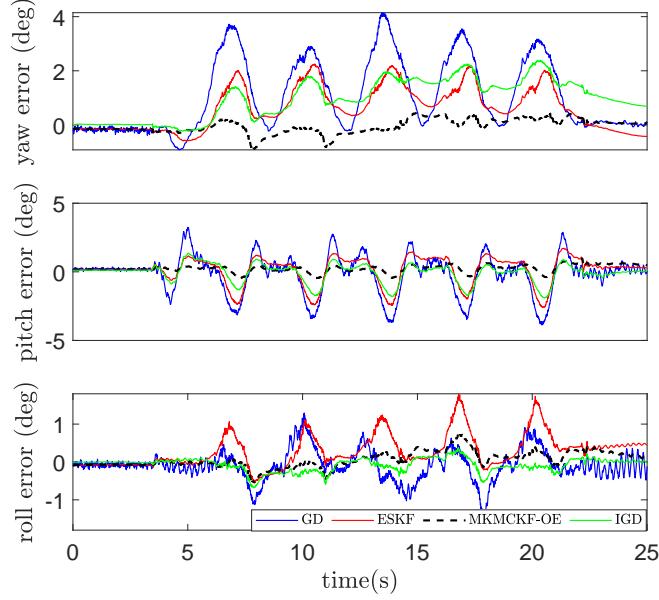


Figure 3.11: Errors of the GD, ESKF, MKMCKF-OE, and IGD in experiment 8.

disturbance. The IGD achieves good performance for the inclination estimation (the roll and pitch). However, their performances on the yaw are not satisfactory when there is magnetic disturbance.

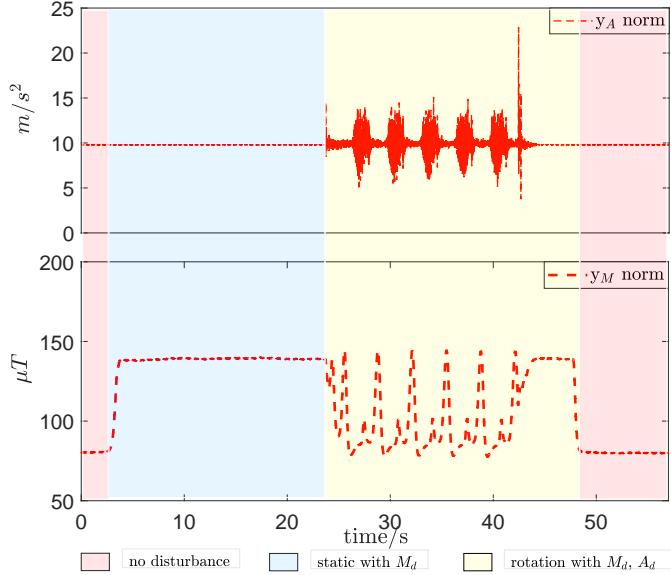


Figure 3.12: Norms of the accelerometer and magnetometer readings in experiment 11.

3.6 Performance Validation on An Embedded System

We also apply the MKMCKF-OE to a low-cost IMU that we built and test its performance using the Vicon motion capture system with 8 cameras. The experimental setup

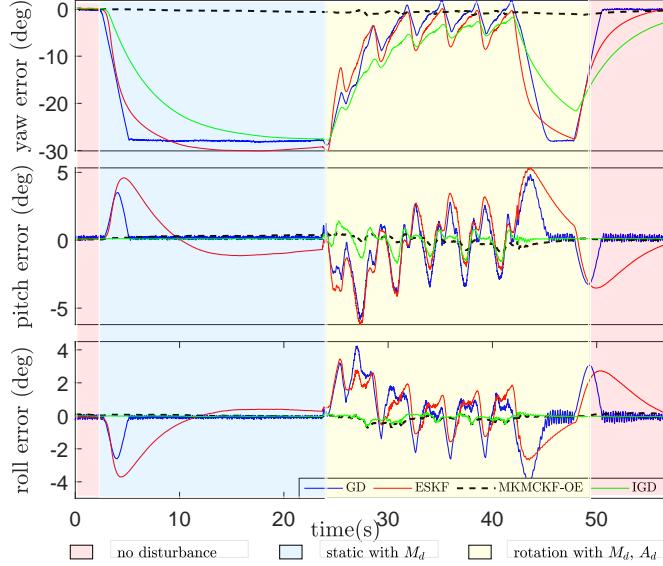


Figure 3.13: Estimation errors of the GD, ESKF, MKMCKF-OE, and IGD in experiment 11.

is depicted in Figure 3.14. Four experiments with different walking tasks are designed to validate the proposed algorithm's performance. A detailed description of the experiments is shown in Table 3.6.

The ground truth foot trajectory in experiment 1 is depicted in Figure 3.15 based on the output of the Vicon system. The corresponding orientation errors of the MKMCKF-OE (we denote it as the MKMC in the tables for brevity.), GD, ESKF, and IGD are shown in Figure 3.16. One can see that the proposed algorithm outperforms the others. The RMSEs of various algorithms in various experiments are summarized in Table 3.7 and 3.8, where we can see that the RMSEs on the yaw in the MKMCKF-OE are less than 2.04 deg, and the RMSEs on the roll and pitch are smaller than 0.98 deg, which is significantly better than the other algorithms.

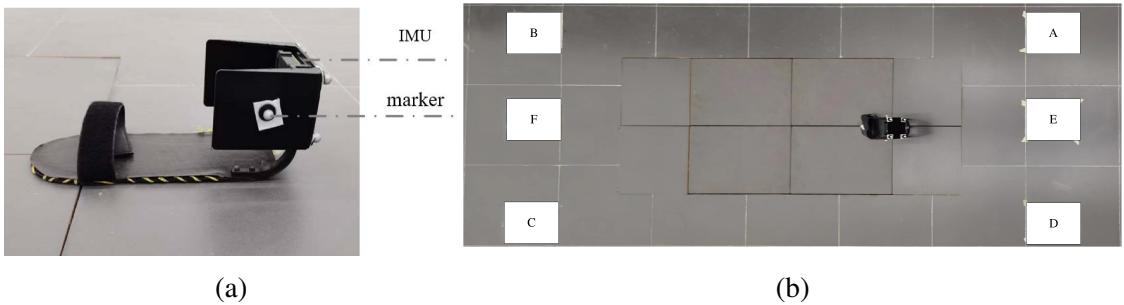


Figure 3.14: Experimental setup. The IMU is installed on the heel of a shoe, which is shown in Figure 3.14(a). The subject walks on designated path which is shown in Figure 3.14(b) and Table 3.6.

Table 3.6: Experimental Description Using Self-designed IMUs.

Index	Description	Route
1	walking with turn	A→B→C→D→A, 1 lap
2	walking with turn	A→B→C→D→A, 2 laps
3	straightforward walking	E→F
4	straightforward walking and return	E→F→E

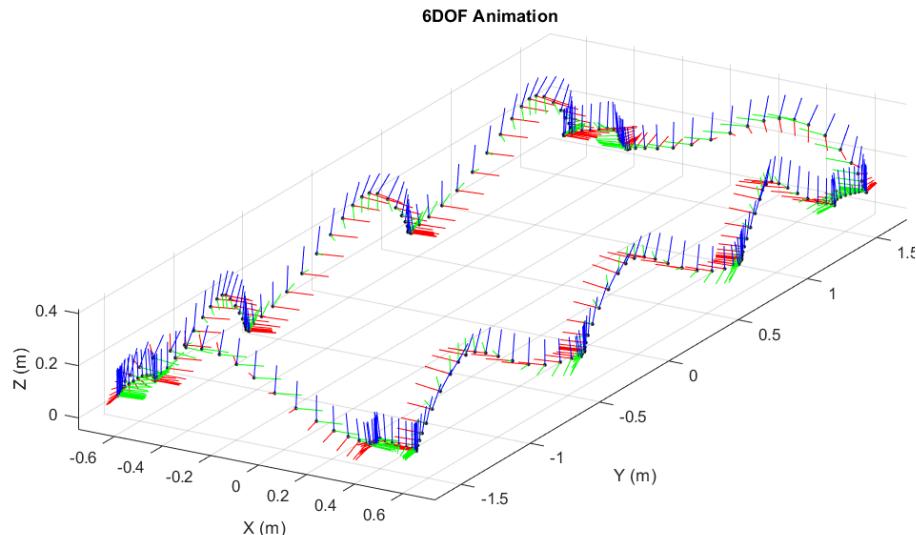


Figure 3.15: Ground truth trajectory and orientation of the foot in experiment 1.

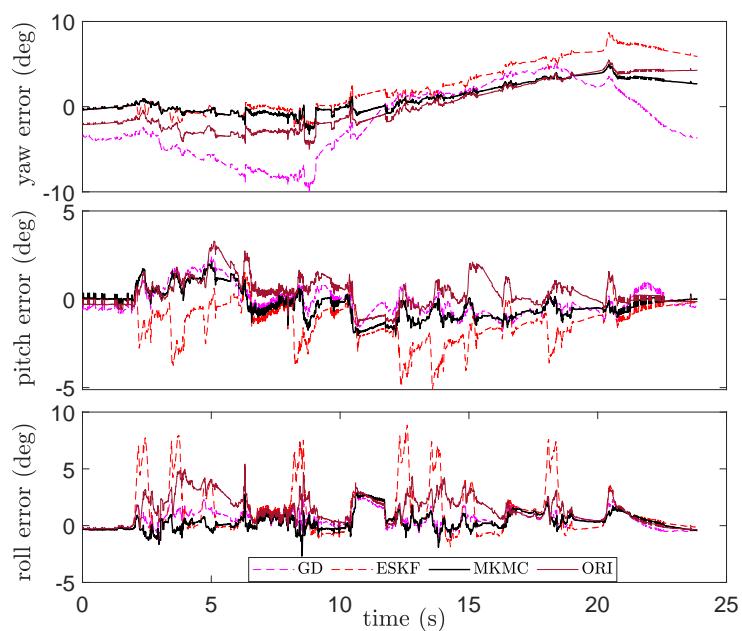


Figure 3.16: Orientation estimation errors of the MKMCKF-OE, GD, ESKF, and IGD in experiment 1.

Table 3.7: Performances of Different Algorithms on the Yaw.

Index	Yaw (deg)			
	GD	IGD	ESKF	MKMC
1	4.4097	2.8135	3.8104	2.0408
2	3.8460	4.0368	4.9117	1.9250
3	5.9588	3.7199	1.1534	0.8668
4	5.7493	2.1855	1.4132	0.6380

Table 3.8: Performances of Different Algorithms on the Pitch and Roll.

Index	Pitch (deg)				Roll (deg)			
	GD	IGD	ESKF	MKMC	GD	IGD	ESKF	MKMC
1	0.8045	0.9171	1.6405	0.8576	0.9479	1.8768	2.4608	0.8435
2	0.9688	0.9260	1.7712	0.9779	1.0060	2.0990	2.4495	0.8606
3	0.7844	0.8889	2.1457	0.6321	1.1826	3.2221	3.2018	0.5356
4	0.8354	0.8999	1.5070	0.8844	1.0928	2.0933	2.5015	0.6888

3.7 Summary

In this chapter, we investigate the influence of external acceleration and magnetic disturbance on the conventional ESKF for both six-axis IMUs and nine-axis IMUs. To mitigate the negative effects of external acceleration for six-axis IMUs while simultaneously reducing computation complexity and parameter number, we propose a CMKMC-ESKF for orientation estimation with an infinite kernel bandwidth σ_{inf} applied to all Gaussian channels and a constant bandwidth σ applied to the external acceleration channels. Intensive simulations and experiments verify the effectiveness of the proposed method. In the case of nine-axis IMUs, we develop a novel MKMCKF-OE that rejects both external acceleration and magnetic disturbance effectively. Further, we employ BO to obtain the optimal kernel bandwidths which is both efficient and convenient. The effectiveness of the proposed method is validated by a set of experiments and compared with the benchmark methods. Experiments show that our proposed method outperforms the GD and the ESKF significantly and is superior to the IGD when there is magnetic disturbance. The MKMCKF-OE is further implemented in a low-cost embedded system, and its performance is verified by an optical motion capture system.

3.8 Appendix

3.8.1 ESKF for Six-axis IMUs

The ESKF for six-axis IMUs is summarized in Algorithm 3.4. The *a priori* estimate

Algorithm 3.4 ESKF for Six-axis IMUs

1: **State Prediction:**

$$2: \hat{q}_k^- = \hat{q}_{k-1}^+ \Delta q(\hat{\omega}_k^- \delta t)$$

$$3: \hat{\omega}_k^- = y_{G,k} - \hat{b}_k^-$$

$$4: \hat{b}_k^- = \hat{b}_{k-1}^+$$

$$5: {}^S\hat{a}_k^- = \eta_a {}^S\hat{a}_{k-1}^+$$

6: **Error State Propagation:**

$$7: P_k^- = f(P_{k-1}^+, Q_{vb}, Q_{vG}, Q_{va}), \text{details in (3.32)}$$

$$8: \hat{x}_{\epsilon,k}^+ = K_k z_{\epsilon,k} \quad \triangleright z_{\epsilon,k} \text{ is shown in (3.8)}$$

$$9: K_k = P_k^- H_k^T S_k^{-1} \quad \triangleright H_k \text{ is shown in (3.9)}$$

$$10: S_k = H_k P_k^- H_k^T + Z_k$$

11: **State Update:**

$$12: \text{Obtain } \hat{\theta}_{\epsilon,k}^+, {}^S\hat{a}_{\epsilon,k}^+, \hat{b}_{\epsilon,k}^+ \text{ from } x_{\epsilon,k}^+ \quad \triangleright \text{based on (3.7)}$$

$$13: \hat{q}_k^+ = \hat{q}_k^- \Delta q(-\hat{\theta}_{\epsilon,k}^+)$$

$$14: \hat{b}_k^+ = \hat{b}_k^- - \hat{b}_{\epsilon,k}^+$$

$$15: {}^S\hat{a}_k^+ = {}^S\hat{a}_k^- - {}^S\hat{a}_{\epsilon,k}^+$$

$$16: P_k^+ = (I - K_k H_k) P_k^-$$

of error covariance P_k^- has

$$P_k^- = \begin{bmatrix} P_k^-(\theta_\epsilon, \theta_\epsilon), P_k^-(\theta_\epsilon, b_\epsilon), P_k^-(\theta_\epsilon, a_\epsilon) \\ P_k^-(b_\epsilon, \theta_\epsilon), P_k^-(b_\epsilon, b_\epsilon), P_k^-(b_\epsilon, a_\epsilon) \\ P_k^-(a_\epsilon, \theta_\epsilon), P_k^-(a_\epsilon, b_\epsilon), P_k^-(a_\epsilon, a_\epsilon) \\ P_k^-(d_\epsilon, \theta_\epsilon), P_k^-(d_\epsilon, b_\epsilon), P_k^-(d_\epsilon, a_\epsilon) \end{bmatrix} \quad (3.31)$$

with

$$\left\{ \begin{array}{l} P_k^-(\theta_\varepsilon, \theta_\varepsilon) = P_{k-1}^+(\theta_\varepsilon, \theta_\varepsilon) + \delta t^2 [P_{k-1}^+(b_{\varepsilon,k}, b_{\varepsilon,k}) + Q_{vb} + Q_{vG}] \\ P_k^-(\theta_\varepsilon, b_\varepsilon) = -\delta t [P_{k-1}^+(\theta_\varepsilon, b_\varepsilon) + Q_{vb}] \\ P_k^-(\theta_\varepsilon, a_\varepsilon) = 0_{3 \times 3} \\ P_k^-(b_\varepsilon, \theta_\varepsilon) = -\delta t [P_{k-1}^+(b_\varepsilon, \theta_\varepsilon) + Q'_{vb}] \\ P_k^-(b_\varepsilon, b_\varepsilon) = P_{k-1}^+(b_\varepsilon, b_\varepsilon) + Q_{vb} \\ P_k^-(b_\varepsilon, a_\varepsilon) = 0_{3 \times 3} \\ P_k^-(a_\varepsilon, \theta_\varepsilon) = 0_{3 \times 3} \\ P_k^-(a_\varepsilon, b_\varepsilon) = 0_{3 \times 3} \\ P_k^-(a_\varepsilon, a_\varepsilon) = \eta^2 P_{k-1}^+(a_\varepsilon, a_\varepsilon) + Q_{va} \end{array} \right. \quad (3.32)$$

where Q_{vb} , Q_{vG} , and Q_{va} are the covariance matrices of the gyroscope offset, the gyroscope signals, and the linear accelerations. One can refer to [106] for more information.

3.8.2 ESKF for Nine-axis IMUs

The ESKF for nine-axis IMUs is summarized in Algorithm 3.5. The *a priori* estimate

Algorithm 3.5 ESKF for Nine-axis IMUs

- 1: **State Prediction:**
 - 2: $\hat{b}_k^- = \hat{b}_{k-1}^+$
 - 3: ${}^S\hat{a}_k^- = \eta_a({}^S\hat{a}_{k-1}^+)$ \triangleright derived from (3.2)
 - 4: ${}^S\hat{d}_k^- = \eta_m({}^S\hat{d}_{k-1}^+)$ \triangleright derived from (3.4)
 - 5: $\hat{\omega}_k^- = y_{G,k} - \hat{b}_k^-$
 - 6: $\hat{q}_k^- = \hat{q}_{k-1}^+ \Delta q(\hat{\omega}_k^- \delta t)$
 - 7: **Error State Propagation:**
 - 8: $P_k^- = f(P_{k-1}^+, Q)$, details in (3.34)
 - 9: $x_{\varepsilon,k,t}^+ = x_{\varepsilon,k}^- + K_k z_{\varepsilon,k}$ $\triangleright z_{\varepsilon,k}$ obtained by (3.12)
 - 10: $K_k = P_k^- H_k^T S_k^{-1}$ $\triangleright H_k$ is shown in equation (3.13)
 - 11: $S_k = H_k P_k^- H_k^T + Z_k$
 - 12: **State Update:**
 - 13: Obtain $\hat{\theta}_{\varepsilon,k}^+$, ${}^S\hat{a}_{\varepsilon,k}^+$, $\hat{b}_{\varepsilon,k}^+$, ${}^S\hat{d}_{\varepsilon,k}^+$ from $x_{\varepsilon,k}^+$ \triangleright based on (3.11)
 - 14: $\hat{q}_k^+ = \hat{q}_k^- \Delta q(-\hat{\theta}_{\varepsilon,k}^+)$
 - 15: $\hat{b}_k^+ = \hat{b}_k^- - \hat{b}_{\varepsilon,k}^+$
 - 16: ${}^S\hat{a}_k^+ = {}^S\hat{a}_k^- - {}^S\hat{a}_{\varepsilon,k}^+$
 - 17: ${}^S\hat{d}_k^+ = {}^S\hat{d}_k^- - {}^S\hat{d}_{\varepsilon,k}^+$ \triangleright if $M_J = 0$ in (3.35)
 - 18: update Gm_k using (3.36), (3.37) \triangleright if $M_J = 0$ in (3.35)
 - 19: $P_k^+ = (I - K_k H_k) P_k^- (I - K_k H_k)^T + K_k Z_k K_k^T$
-

of error covariance can be written as

$$P_k^- = \begin{bmatrix} P_k^-(\theta_\varepsilon, \theta_\varepsilon), P_k^-(\theta_\varepsilon, b_\varepsilon), P_k^-(\theta_\varepsilon, a_\varepsilon), P_k^-(\theta_\varepsilon, d_\varepsilon) \\ P_k^-(b_\varepsilon, \theta_\varepsilon), P_k^-(b_\varepsilon, b_\varepsilon), P_k^-(b_\varepsilon, a_\varepsilon), P_k^-(b_\varepsilon, d_\varepsilon) \\ P_k^-(a_\varepsilon, \theta_\varepsilon), P_k^-(a_\varepsilon, b_\varepsilon), P_k^-(a_\varepsilon, a_\varepsilon), P_k^-(a_\varepsilon, d_\varepsilon) \\ P_k^-(d_\varepsilon, \theta_\varepsilon), P_k^-(d_\varepsilon, b_\varepsilon), P_k^-(d_\varepsilon, a_\varepsilon), P_k^-(d_\varepsilon, d_\varepsilon) \end{bmatrix} \quad (3.33)$$

with

$$\left\{ \begin{array}{l} P_k^-(\theta_\varepsilon, \theta_\varepsilon) = P_{k-1}^+(\theta_\varepsilon, \theta_\varepsilon) + \delta t^2 [P_{k-1}^+(b_{\varepsilon,k}, b_{\varepsilon,k}) + Q_{vb} + Q_{vG}] \\ P_k^-(\theta_\varepsilon, b_\varepsilon) = -\delta t [P_{k-1}^+(\theta_\varepsilon, b_\varepsilon) + Q_{vb}] \\ P_k^-(\theta_\varepsilon, a_\varepsilon) = P_k^-(\theta_\varepsilon, d_\varepsilon) = 0_{3 \times 3} \\ P_k^-(b_\varepsilon, \theta_\varepsilon) = P_k^-(\theta_\varepsilon, b_\varepsilon)^T \\ P_k^-(b_\varepsilon, b_\varepsilon) = P_{k-1}^+(b_\varepsilon, b_\varepsilon) + Q_{vb} \\ P_k^-(b_\varepsilon, a_\varepsilon) = P_k^-(b_\varepsilon, d_\varepsilon) = 0_{3 \times 3} \\ P_k^-(a_\varepsilon, \theta_\varepsilon) = P_k^-(a_\varepsilon, b_\varepsilon) = P_k^-(a_\varepsilon, d_\varepsilon) = 0_{3 \times 3} \\ P_k^-(a_\varepsilon, a_\varepsilon) = \eta_a^2 P_{k-1}^+(a_\varepsilon, a_\varepsilon) + Q_{va} \\ P_k^-(d_\varepsilon, \theta_\varepsilon) = P_k^-(d_\varepsilon, b_\varepsilon) = P_k^-(d_\varepsilon, a_\varepsilon) = 0_{3 \times 3} \\ P_k^-(d_\varepsilon, d_\varepsilon) = \eta_m^2 P_{k-1}^+(d_\varepsilon, d_\varepsilon) + Q_{vd} \end{array} \right. \quad (3.34)$$

where Q_{vb} , Q_{vG} , Q_{va} , Q_{vd} are the covariance matrices of the gyroscope offset, gyroscope signals, linear accelerations, and magnetic disturbance [51, 106].

To detect magnetic jamming, we apply the following threshold-based classifier:

$$M_J = \begin{cases} 1, \|\hat{d}_{\varepsilon,k}^+\|^2 > \tau^2 \\ 0, \text{otherwise} \end{cases} \quad (3.35)$$

where $\hat{d}_{\varepsilon,k}^+$ is the error magnetic disturbance, τ is a threshold, $\|\cdot\|$ is the Euclidean norm.

Then, if there is magnetic jamming with $M_J = 1$, it implies that the magnetometer signals are not reliable. In this case, the error state should be updated only based on the accelerometer readings and the gyroscope readings (see [106, p.19] for details). Moreover, the update of $\hat{d}_{\varepsilon,k}^+$ and Gm_k (Line 17 and 18) in algorithm 3.3 should be ignored. Otherwise, we reconstruct Gm_k to minimize the influence of magnetic disturbance on the pitch and the roll as follows

$$\varphi_{mag,k} = \tan^{-1} \left(\frac{Gm_{z,k-1} - G\hat{d}_{\varepsilon,z,k}^+}{Gm_{x,k-1} - G\hat{d}_{\varepsilon,x,k}^+} \right) \quad (3.36)$$

where $\varphi_{mag,k}$ is the inclination angle at time step k , and the subscript x and z denotes the

first and the third element of the corresponding vector, respectively. Then, the geomagnetic vector is updated as

$${}^G m_k = \begin{pmatrix} M_{sth} \cos(\varphi_{mag,k}) \\ 0 \\ M_{sth} \sin(\varphi_{mag,k}) \end{pmatrix}. \quad (3.37)$$

where M_{sth} is the local geomagnetic field strength. A full treatment of this model can be found in [106, p.19].

Chapter 4

Disturbance Observer Using the Generalized Multi-kernel Correntropy

In this chapter, we propose a generalized multi-kernel correntropy under the generalized Gaussian density (GGD) function as kernels. We provide its properties and demonstrate its feasibility as a robust cost. Further, we associate the generalized multi-kernel correntropy loss (GMKCL) with a type of heavy-tailed distribution and derive a generalized multi-kernel maximum correntropy Kalman filter (GMKMCKF). Finally, we apply this method to disturbance estimation. Simulations verify the effectiveness of the proposed algorithm.

4.1 Generalized Multi-kernel Correntropy

In this section, we consider an estimation problem with unknown disturbance. To suppress unknown disturbance, we introduce a generalized multi-kernel correntropy. Then, we provide its properties and compare the GMKCL with the LMP criterion.

4.1.1 Problem Formulation

In many practical applications, systems contain unknown process disturbance, i.e.,

$$\begin{aligned} x_{k+1} &= Ax_k + \Gamma d_k + w_{x,k} \\ y_k &= Cx_k + v_k \end{aligned} \tag{4.1}$$

where $d_k \in \mathbb{R}^l$ is the unknown disturbance, and $w_{x,k}$ and v_k are noises. To estimate the disturbance, we treat the disturbance as a new state and construct the augmented state as

$\bar{x}_k = [d_k^T, x_k^T]^T$. We assume that disturbance dynamics follows

$$d_{k+1} = Id_k + w_{d,k} \quad (4.2)$$

since we do not have the *a priori* knowledge about the disturbance dynamics. Then, we obtain

$$\begin{aligned} \bar{x}_{k+1} &= \bar{A}\bar{x}_k + \bar{w}_k \\ y_k &= \bar{C}\bar{x}_k + \bar{v}_k \end{aligned} \quad (4.3)$$

with

$$\bar{A} = \begin{bmatrix} I & 0 \\ \Gamma & A \end{bmatrix}, \bar{C} = \begin{bmatrix} 0 & C \end{bmatrix} \quad (4.4)$$

where $\bar{w}_k = [w_{d,k}^T, w_{x,k}^T]^T$ and $\bar{v}_k = v_k$. In the conventional Kalman filter, the noises are assumed to be Gaussian with

$$w_{d,k} \sim \mathcal{N}(0, Q_d), w_{x,k} \sim \mathcal{N}(0, Q_x), v_k \sim \mathcal{N}(0, R). \quad (4.5)$$

However, this assumption does not hold for disturbance channels. In a practical application, the disturbance dynamics is time-varying with $d_{k+1} = f(d_k) + w_k^*$ where $f(d_k)$ is unknown dynamics and w_k^* is the noise. We use the nominal model (4.2) for implementation since we are not accessible to the practical dynamics $f(d_k)$. In this case, $w_{d,k} = f(d_k) - Id_k + w_k^*$ which is characterized by a heavy-tailed distribution since it contains both the modeling mismatch $f(d_k) - Id_k$ and the noise w_k^* . A possible representation of this kind of distribution is the ε -contaminated mixture model with

$$w_{d,k} \sim \varepsilon \mathcal{U}(a, b) + (1 - \varepsilon) \mathcal{N}(0, Q_w), \quad 0 < \varepsilon < 1 \quad (4.6)$$

where ε is a constant, $\mathcal{U}(a, b)$ is a uniform distribution with bounds a and b which is responsible for the modeling mismatch part $f(d_k) - Id_k$, and $\mathcal{N}(0, Q_w)$ is a Gaussian distribution which is responsible for the noise part w_k^* . Unfortunately, this mixture model cannot be approximated by a single Gaussian distribution effectively (see Figure 4.1). This reveals that KF is not a good estimator for this type of noise from the modeling perspective. Further, in some cases, the noises for the state or the measurement may follow other types of distributions (e.g., heavy-tailed distribution in [146], Laplace distribution in [147]). All these factors deteriorate the estimation accuracy of the KF.

Remark 4.1. *For a state estimation problem with unknown disturbance, the noise in the disturbance channel usually is heavy-tailed. This type of distribution cannot be estimated*

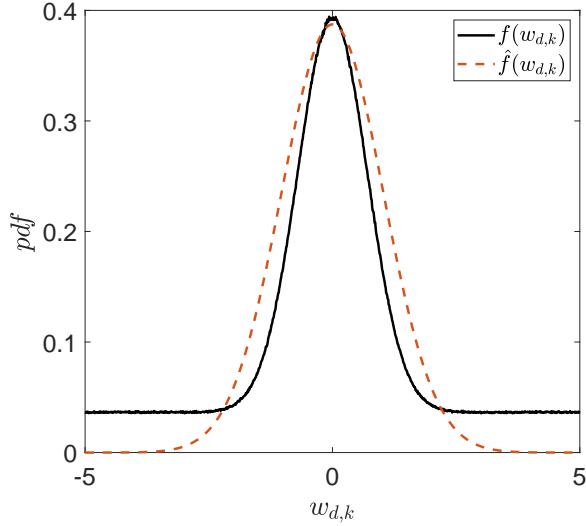


Figure 4.1: Approximating a ε -contaminated mixture model using a Gaussian distribution. The target distribution follows $0.37\mathcal{U}(-5, 5) + 0.63\mathcal{N}(0, 0.5)$. The estimated distribution is obtained by minimizing the mean squared error $E[(f(w_{d,k}) - \hat{f}(w_{d,k}))^2]$.

by a Gaussian distribution effectively, resulting in the performance degeneration of the KF. A conventional way for this problem is to enlarge the nominal covariance. However, this would deteriorate the estimation performance when the modeling mismatch is small (i.e., $f(d_k) \approx Id_k$).

4.1.2 Generalized Multi-kernel Correntropy

The correntropy is a robust measure for random variables. It is originally defined as a local similarity measure of two random variables $X, Y \in \mathbb{R}$ with joint distribution $F_{XY}(x, y)$ and its calculation is shown in (2.12)-(2.14).

In this chapter, we adopt the GGD as kernels with

$$\kappa(x, y) = G_{\alpha, \beta}(x, y) = \exp(-|e/\beta|^\alpha) \quad (4.7)$$

where $e = x - y$ is the error, $\alpha > 0$ is the shape, and $\beta > 0$ is the bandwidth. Under the GGD, we **define** the generalized multi-kernel correntropy for random vectors $\mathcal{X}, \mathcal{Y} \in \mathbb{R}^l$ as follows:

$$C(\mathcal{X}, \mathcal{Y}) = \sum_{i=1}^l E[\tilde{\kappa}_i(\mathcal{X}_i, \mathcal{Y}_i)] = \sum_{i=1}^l \int \tilde{\kappa}_i(x_i, y_i) dF_{\mathcal{X}_i \mathcal{Y}_i}(x_i, y_i) \quad (4.8)$$

with

$$\tilde{\kappa}_{\alpha, \beta_i}(x_i, y_i) = \beta_i^\alpha G_{\alpha, \beta_i}(x_i, y_i) = \beta_i^\alpha \exp(-|e_i/\beta_i|^\alpha)$$

where $e_i = x_i - y_i$ is the realization error of \mathcal{X}_i and \mathcal{Y}_i , α is the universal shape parameter, and β_i is the i -th bandwidth for \mathcal{X}_i and \mathcal{Y}_i . In a practical application, the joint distribution $F_{\mathcal{X}_i \mathcal{Y}_i}(x_i, y_i)$ is not available. In this case, we can estimate the multi-kernel correntropy using a mean estimator

$$\hat{C}(\mathcal{X}, \mathcal{Y}) = \sum_{i=1}^l \beta_i^\alpha C_{\alpha, \beta_i}(\mathcal{X}_i, \mathcal{Y}_i) \quad (4.9)$$

with

$$C_{\alpha, \beta_i}(\mathcal{X}_i, \mathcal{Y}_i) = \frac{1}{N} \sum_{k=1}^N G_{\alpha, \beta_i}(x_i(k), y_i(k)) \quad (4.10)$$

where $C_{\alpha, \beta_i}(\mathcal{X}_i, \mathcal{Y}_i)$ is the correntropy for $\mathcal{X}_i, \mathcal{Y}_i$ under α and β_i , and $x_i(k)$ and $y_i(k)$ is the k -th sample of random variables \mathcal{X}_i and \mathcal{Y}_i . Correspondingly, the generalized multi-kernel correntropy (we denote it as GL for brevity) is defined as

$$\begin{aligned} J_{GL}(\mathcal{X}, \mathcal{Y}) &= \sum_{i=1}^l \beta_i^\alpha (1 - C_{\alpha, \beta_i}(\mathcal{X}_i, \mathcal{Y}_i)) \\ &= \sum_{i=1}^l \beta_i^\alpha \left(1 - \frac{1}{N} \sum_{k=1}^N G_{\alpha, \beta_i}(x_i(k), y_i(k)) \right). \end{aligned} \quad (4.11)$$

4.1.3 Properties of the Generalized Multi-kernel Correntropy

In this section, we provide some important properties of the generalized multi-kernel correntropy.

Theorem 4.1. *In the case of $0 < \alpha \leq 2$, the generalized multi-kernel correntropy in (4.9) can be regarded as a weighted summation of the second-order statistic in the mapped feature space.*

The proof of this theorem is shown in Appendix 4.5.1.

Theorem 4.2. *When setting $\beta_i^\alpha \rightarrow \infty$, the GL in (4.11) becomes the expectation of α -order absolute moments with $\lim_{\beta_i^\alpha \rightarrow \infty} J_{GL}(\mathcal{X}, \mathcal{Y}) = E \|\mathcal{X} - \mathcal{Y}\|_\alpha^\alpha$.*

The proof of this theorem is shown in Appendix 4.5.2. It reveals that when setting the kernel parameters as $\beta_i^\alpha \rightarrow \infty$, the GL becomes the traditional least mean p -power (LMP) criterion with $\alpha = p$.

Theorem 4.3. *Denote the correntropy induced metric as $\text{GCIM}(\mathcal{X}, \mathcal{Y}) = (J_{GL}(\mathcal{X}, \mathcal{Y}))^{\frac{1}{2}}$. Then, it defines a metric in the N -dimensional sample vector space when $0 < \alpha \leq 2$.*

The proof of this theorem is shown in Appendix 4.5.3. The contour plots of $J_{GL}(\mathcal{X}, 0)^{\frac{1}{\alpha}}$ in 2D space with different shape parameters α and different bandwidths β_i are shown in Figure 4.2. One can see that when selecting big bandwidths for all channels in Figures 4.2(a), 4.2(d) and 4.2(g), $J_{GL}(\mathcal{X}, 0)^{\frac{1}{\alpha}}$ behaves like an ℓ_α norm; when using relative small bandwidths in Figures 4.2(b), 4.2(e), and 4.2(h), $J_{GL}(\mathcal{X}, 0)^{\frac{1}{\alpha}}$ varies from an ℓ_α norm to an ℓ_0 norm; when using different bandwidths for different channels, $J_{GL}(\mathcal{X}, 0)^{\frac{1}{\alpha}}$ behaves differently on different channels, i.e., it changes from an ℓ_α norm to an ℓ_0 norm on the \mathcal{X}_1 channel, and behaves like an ℓ_α norm on the \mathcal{X}_2 channel when setting $\beta_1 = 1$ and $\beta_2 = 100$. From Figure 4.2, one can see that the shape of $J_{GL}(\mathcal{X}, 0)^{\frac{1}{\alpha}}$ can be adjusted flexibly among the ℓ_α to ℓ_0 norm by the kernel parameters α and β_i , which provides a potential improvement for algorithms when using the GL as a loss function to replace the LMP criterion.

4.1.4 Influence Function of the LMP and GL

The influence function measures the derivation of the loss function with respect to the residual [140], and gives a straightforward view of how the residual influences the objective function. For the LMP [119] criterion, we have

$$J_{LMP}(e) = \|e\|_p^p = \sum_{i=1}^l |e_i|^p. \quad (4.12)$$

For the GL in (4.11) with $N = 1$, we have

$$J_{GL}(e) = \sum_{i=1}^l \beta_i^\alpha (1 - G_{\alpha, \beta_i}(e_i)). \quad (4.13)$$

where $e \in \mathbb{R}^l$ and $e_i = x_i - y_i$ is the i -th error. The influence functions can be obtained by calculating the gradients

$$\begin{aligned} \nabla J_{LMP}(e) &= \frac{\partial J_{LMP}}{\partial e} = [\rho_1, \rho_2, \dots, \rho_l]^T \\ \nabla J_{GL}(e) &= \frac{\partial J_{GL}}{\partial e} = [\gamma_1, \gamma_2, \dots, \gamma_l]^T \end{aligned} \quad (4.14)$$

with

$$\begin{aligned} \rho_i &= p \frac{|e_i|^p}{e_i}, i = 1, 2, \dots, l \\ \gamma_i &= \frac{\alpha \exp^{-\frac{|e_i|^\alpha}{\beta_i^\alpha}} |e_i|^\alpha}{e_i}, i = 1, 2, \dots, l. \end{aligned}$$

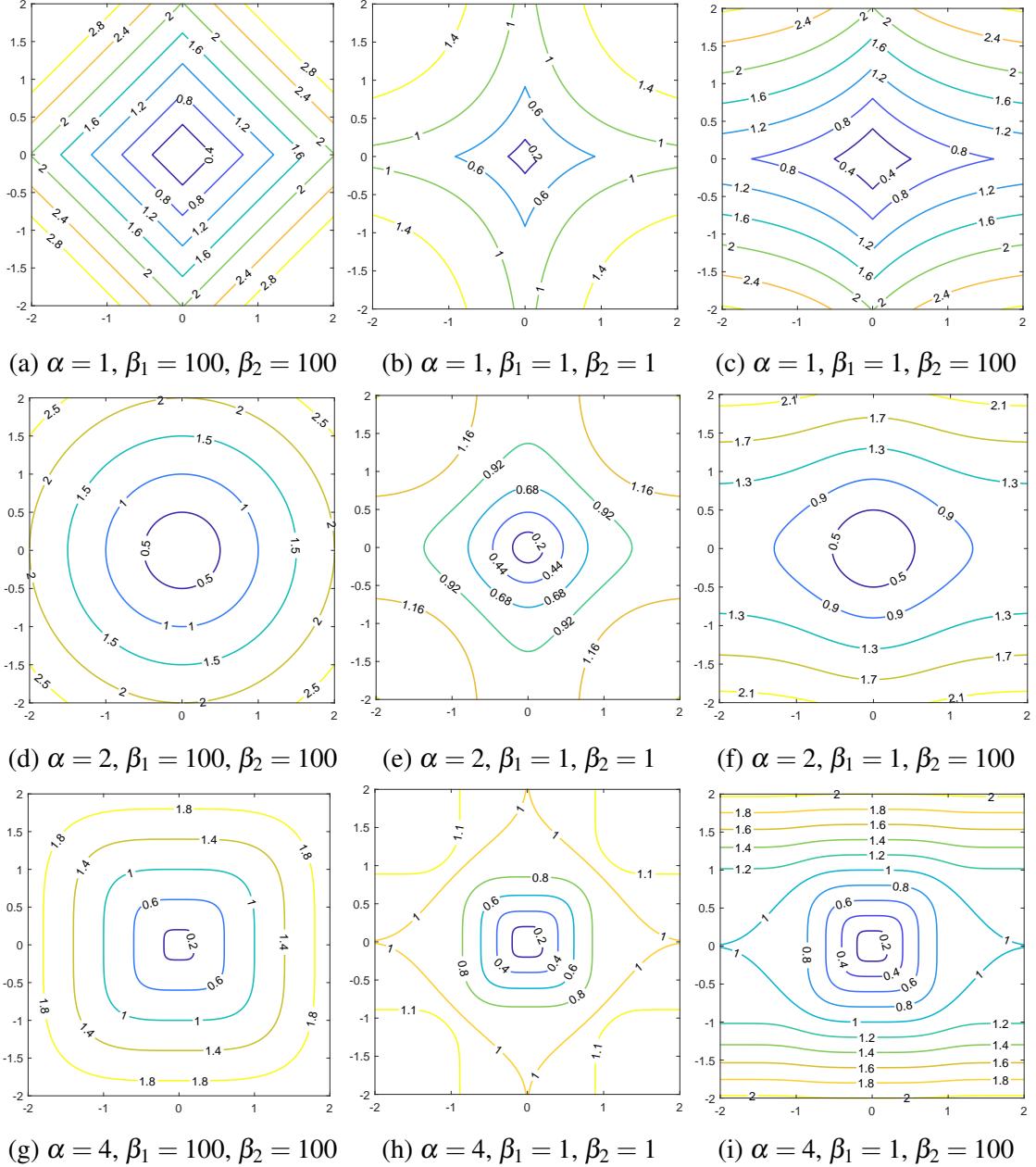


Figure 4.2: Contours of $J_{GL}(\mathcal{X}, 0)^{\frac{1}{\alpha}}$ in 2D space with different shape parameters and different bandwidths.

Theorem 4.4. In the case of $N = 1$, the GL in (4.11) is a differential invex function of e with $\alpha > 1$ and $e_i \leq \varphi$ ($i = 1, 2, \dots, l$) where $\varphi \in \mathbb{R}^+$ is an arbitrary positive number.

The proof of this theorem is shown in Appendix 4.5.4.

Theorem 4.5. The $J_{GL}(e)$ is identical to the $J_{LMP}(e)$ when $\alpha = p$ and $\beta_i^\alpha \rightarrow \infty$. Moreover, in the case of $0 < \alpha \leq 1$, $J_{GL}(e)$ is concave with $e \neq 0$; in the case of $\alpha > 1$, $J_{GL}(e)$ is convex with the region $|e_i| \leq (\frac{\alpha-1}{\alpha})^{\frac{1}{\alpha}} \beta_i$.

The proof of this theorem can be found in Appendix 4.5.5. In one dimensional case with $e \in \mathbb{R}$, we have $J_{LMP}(e) = |e|^p$, $J_{GL}(e) = \beta^\alpha(1 - G_{\alpha,\beta}(e))$, $\nabla J_{LMP}(e) = p \frac{|e|^{p-1}}{e}$, and $\nabla J_{GL}(e) = \frac{\alpha \exp^{-\frac{|e|^\alpha}{\beta^\alpha}} |e|^\alpha}{e}$. The plots of $J_{LMP}(e)$ and $J_{GL}(e)$ with respect to e are shown in Figures 4.3(a) and 4.3(b), and the corresponding influence functions are shown in Figures 4.3(c) and 4.3(d). When $p = 2$ in Figure 4.3(a), the LMP becomes the well-known MSE criterion. One can see that the corresponding influence function grows linearly with the error (see Figure 4.3(c)). This indicates that a big error contributes much to the objective function. On the contrary, a low order p in $J_{LMP}(e)$ (e.g., $p = 0.2$ in Figure 4.3(c)) is more robust to outliers. Generally, a lower order error moment is more suitable for heavy-tailed noises while a higher order moment is more appropriate for light-tailed noises. However, the fixed order error moment cannot suppress mixed noise distributions effectively since different noise distributions may require different orders of error moments. For example, for a mixed noise distribution with a high probability of Gaussian noise and a low probability of outliers, $p = 2$ is optimal for Gaussian noises while a lower p is more suitable for outliers. In this case, a trade-off between the Gaussian noise elimination and the outlier mitigation is unavoidable using a fixed order error moment.

This limitation can be eased by the GL. As shown in Figure 4.3(b), the GL approaches $|e|^\alpha$ when setting $\beta = 100$ which varies Theorem 4.2. The corresponding influence function ∇J_{GL} with $\beta = 1$ and $\alpha > 1$ is shown in Figure 4.3(d). One can see when the error is bigger than $(\frac{\alpha-1}{\alpha})^{\frac{1}{\alpha}} \beta$ (see Appendix 4.5.5), ∇J_{GL} decreases continually and goes towards zero when the error is extremely large. On the contrary, when the error is small, ∇J_{GL} is close to ∇J_{LMP} . Due to this property, the GL is an attractive cost function for noises with a mixture model distribution since its behavior varies with the instantaneous error. For example, for the aforementioned mixed distribution in (4.6), the error is expected to be small for the Gaussian noise while being large for the uniform noise. In this situation,

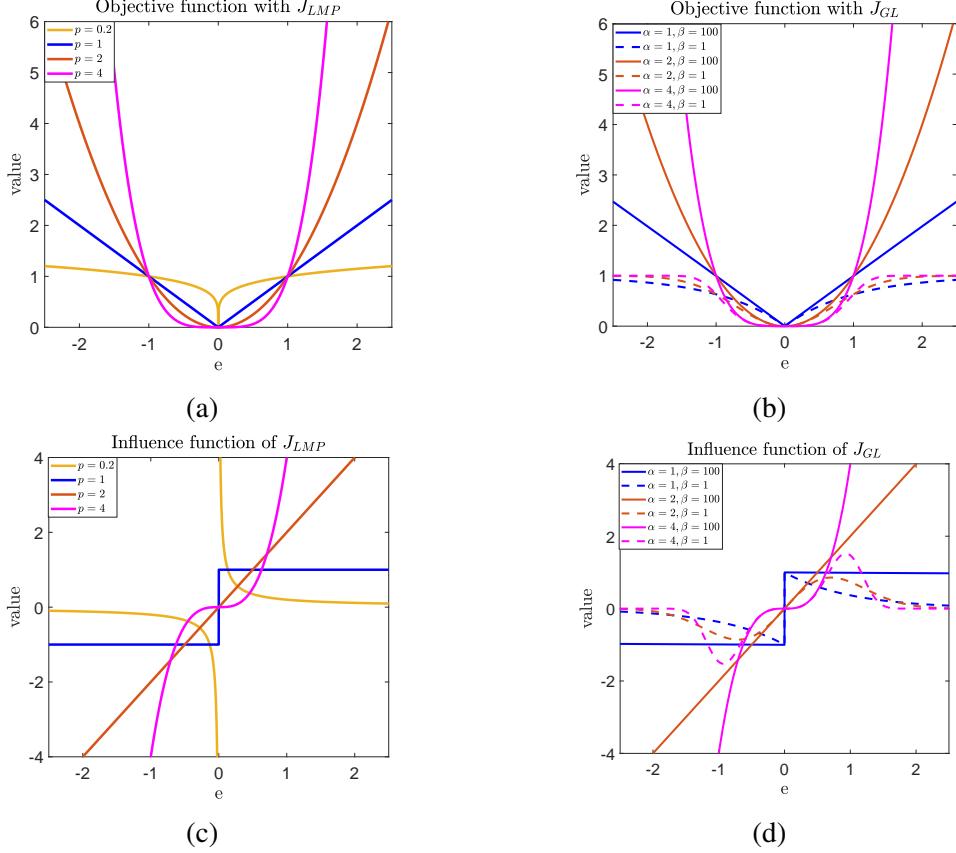


Figure 4.3: Plots of the J_{LMP} , J_{GL} , ∇J_{LMP} , and ∇J_{GL} in one dimensional case.

both the Gaussian noise and the uniform noise can be mitigated effectively by selecting proper kernel parameters.

4.1.5 Relationship with the Kalman Filter

As shown in equations (2.22)–(2.26), the KF is optimal for a linear system with Gaussian noises. However, when the noises w_k and v_k are **non-Gaussian**, the probability density function (2.24) does not hold. In this case, the ℓ_2 norm-based cost function in (2.26) is not the best. The definition of the measurement error $e_{r,k}$ and process error $e_{p,k}$ is shown in (2.27). In contrast to the Gaussian assumption, we **assume** that $p(y_k|x_k)$ and $p(x_k|\{y_{k-1}\})$ have the following distributions:

$$p(y_k|x_k) = \begin{cases} c_r \exp(-J_{GL,\alpha,\beta_r}(e_{r,k})), & e_{r,k} \in \mathcal{E}_r \\ 0, & \text{otherwise} \end{cases} \quad (4.15)$$

$$p(x_k|\{y_{k-1}\}) = \begin{cases} c_p \exp(-J_{GL,\alpha,\beta_p}(e_{p,k})), & e_{p,k} \in \mathcal{E}_p \\ 0, & \text{otherwise} \end{cases}$$

where J_{GL,α,β_r} and J_{GL,α,β_p} are the GL with $N = 1$, $\alpha \in \mathbb{R}$, $\beta_r \in \mathbb{R}^m$, and $\beta_p \in \mathbb{R}^n$. The symbol \mathcal{E}_r is the domain of $e_{r,k}$, \mathcal{E}_p is the domain of $e_{p,k}$, and c_r and c_p are two constants so that the integral of $p(y_k|x_k)$ and $p(x_k|y_{k-1})$ over the entire space is equal to 1. The errors $e_{r,k}$ and $e_{p,k}$ are assumed to be bounded so that $p(y_k|x_k)$ and $p(x_k|y_{k-1})$ are probability density functions. Compared with the Gaussian distribution in (2.24), equation (4.15) represents a type of heavy-tailed distribution (see Figure 4.4). By this assumption and analogy to equations (2.23)–(2.26), we have

$$\arg \max_{\hat{x}_k} p(x_k|\{y_k\}) = \arg \min_{\hat{x}_k} J_{GL,KF} \quad (4.16)$$

with

$$J_{GL,KF} = J_{GL,\alpha,\beta_r}(e_{r,k}) + J_{GL,\alpha,\beta_p}(e_{p,k}). \quad (4.17)$$

One can see that the ℓ_2 -norm based objective function in (2.26) is replaced by the GL in (4.17). Equations (4.15)–(4.17) indicate that the GL is an optimal cost function when the noises follows (4.15) based on MAP. To visualize the distributions in (4.15), in one dimensional case, we have

$$p(e) = \begin{cases} c \exp(-J_{GL,\alpha,\beta}(e)), e \in \mathcal{E} \\ 0, \text{otherwise} \end{cases}. \quad (4.18)$$

The comparisons of the $p(e)$, the Laplace distribution, the Gaussian distribution, and the ε -contaminated mixture model are shown in Figure 4.4. One can see that $p(e)$ approaches $\mathcal{L}(0, 1)$ with $\alpha = 1$ and $\beta = 100$ in Figure 4.4(a), and is close to $\mathcal{N}(0, 0.5)$ with $\alpha = 2$ and $\beta = 100$ in Figure 4.4(b). Moreover, when selecting a proper bandwidth, it can approach a ε -contaminated mixture model effectively (see the magenta and the dot blue lines in Figure 4.4(a) and Figure 4.4(b)). Actually, $p(e)$ approaches the α -order exponential distribution when setting $\beta \rightarrow \infty$ since $\lim_{\beta \rightarrow \infty} c \exp(-J_{GL,\alpha,\beta}(e)) = c \exp(-e^\alpha)$ and it represents a type of heavy-tailed distribution when using relative small β . This implies that the shape of $p(e)$ can be controlled by the bandwidth flexibly.

Remark 4.2. When the kernel bandwidth β is very small, the distribution (4.18) is very similar to a uniform distribution. When the bandwidth $\beta = 0$, it becomes a Dirac delta function. In practice, the error domain \mathcal{E} does not have to be declared explicitly. It can be implicitly selected as a sufficient large value so that all errors are bounded by the domain \mathcal{E} .

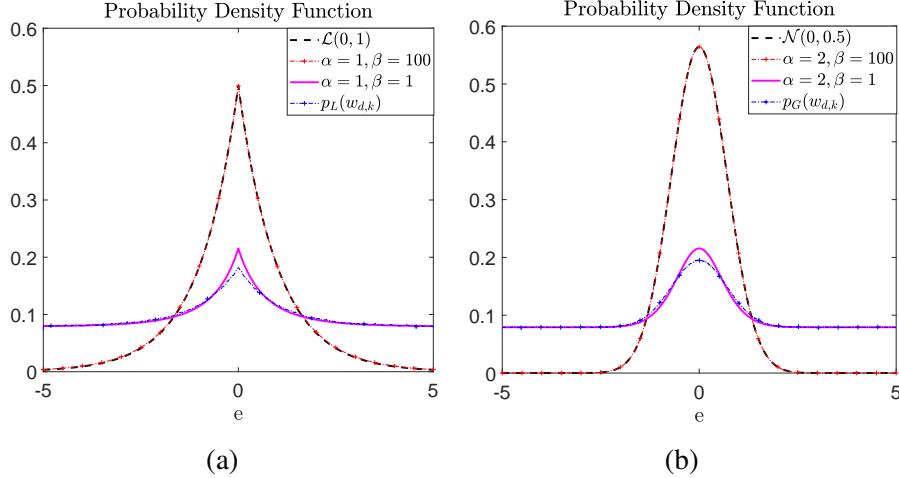


Figure 4.4: Laplace distribution, Gaussian distribution, $p(e)$ in (4.18), and $p(w_{d,k})$ using a ε -contaminated mixture model. The error domain is set to be $\mathcal{E} = [-5, 5]$ in $p(e)$. The noise distribution $p_L(w_{d,k})$ follows $0.2\mathcal{L}(0, 1) + 0.8\mathcal{U}(-5, 5)$ while $p_G(w_{d,k})$ follows $0.2\mathcal{N}(0, 0.5) + 0.8\mathcal{U}(-5, 5)$.

4.2 Algorithm Derivation for Disturbance Estimation

In this section, we derive the GMKMCKF. Then, we apply it to disturbance estimation.

4.2.1 Algorithm Derivation

The system dynamics in (2.1) can be rewritten as

$$\begin{pmatrix} x_k^- \\ y_k \end{pmatrix} = \begin{pmatrix} I \\ C \end{pmatrix} x_k + v_k \quad (4.19)$$

where x_k^- is the *a priori* estimate of state x_k . The noise v_k has

$$v_k = \begin{pmatrix} x_k^- - x_k \\ v_k \end{pmatrix} \quad (4.20)$$

with

$$E(v_k v_k^T) = \begin{pmatrix} P_k^- & 0 \\ 0 & R_k \end{pmatrix} = \begin{pmatrix} B_{p,k} B_{p,k}^T & 0 \\ 0 & B_{r,k} B_{r,k}^T \end{pmatrix} = B_k B_k^T \quad (4.21)$$

where P_k^- is the *a priori* error covariance, and B_p and B_r can be obtained by Cholesky decomposition. Left multiplying B_k^{-1} in both sides of (4.19), we obtain

$$T_k = W_k x_k + \xi_k \quad (4.22)$$

where

$$T_k = B_k^{-1} \begin{pmatrix} x_k^- \\ y_k \end{pmatrix}, W_k = B_k^{-1} \begin{pmatrix} I \\ C \end{pmatrix}, \xi_k = B_k^{-1} v_k \quad (4.23)$$

The noise term ξ_k is white since $E(\xi_k \xi_k^T) = I$. We use the GL in (4.17) as the cost function, which is equivalent with

$$\arg \min_{x_k} J_{GL,KF} = \sum_{i=1}^{n+m} \beta_i^\alpha (1 - G_{\alpha, \beta_i}(e_{i,k})) . \quad (4.24)$$

where $e_{i,k} = t_{i,k} - w_{i,k}x_k$ is the error at time step k , $t_{i,k}$ is the i -th element of T_k , $w_{i,k}$ is the i -th row of W_k , and α and β_i are kernel parameters. It follows that

$$\arg \min_{x_k} J_{GL,KF} = \arg \max_{x_k} J_{GC,KF} \quad (4.25)$$

with

$$J_{GC,KF} = \sum_{i=1}^{n+m} \beta_i^\alpha G_{\alpha, \beta_i}(e_{i,k}) .$$

Equation (4.25) can be solved by

$$\frac{\partial J_{GC,KF}}{\partial x_k} = 0 . \quad (4.26)$$

It follows that

$$\sum_{i=1}^{n+m} \alpha w_{i,k}^T (t_{i,k} - w_{i,k}x_k) |e_{i,k}|^{\alpha-2} \exp(-\beta_i^{-\alpha} |e_{i,k}|^\alpha) = 0 . \quad (4.27)$$

We denote $|e_{i,k}|^{\alpha-2} \exp(-\beta_i^{-\alpha} |e_{i,k}|^\alpha)$ as $g_C(e_{i,k})$. Then, we have

$$\sum_{i=1}^{n+m} w_{i,k}^T g_C(e_{i,k}) t_{i,k} = \sum_{i=1}^{n+m} w_{i,k}^T g_C(e_{i,k}) w_{i,k} x_k . \quad (4.28)$$

It follows that

$$x_k = \left(\sum_{i=1}^{n+m} w_{i,k}^T g_C(e_{i,k}) w_{i,k} \right)^{-1} \left(\sum_{i=1}^{n+m} w_{i,k}^T g_C(e_{i,k}) t_{i,k} \right) . \quad (4.29)$$

One can see that the above question is a fix-point equation since both sides of (4.29) contain x_k ($e_{i,k} = t_{i,k} - w_{i,k}x_k$). It can be expressed as

$$x_k = (W_k^T M_k W_k)^{-1} (W_k^T M_k T_k) \quad (4.30)$$

with

$$M_k = \begin{bmatrix} M_{p,k} & 0 \\ 0 & M_{r,k} \end{bmatrix} \quad (4.31)$$

where $M_{p,k} = \text{diag}(g(e_{1,k}), \dots, g(e_{n,k}))$ and $M_{r,k} = \text{diag}(g(e_{n+1,k}), \dots, g(e_{n+m,k}))$. Substituting the expression of W_k in (4.22) into (4.30), we have

$$(W_k^T M_k W_k)^{-1} = [(B_{p,k}^{-1})^T M_{p,k} B_{p,k}^{-1} + C^T (B_{r,k}^{-1})^T M_{r,k} B_{r,k}^{-1} C]^{-1} . \quad (4.32)$$

Using the matrix inversion lemma, we arrive at

$$\begin{aligned} (W_k^T M_k W_k)^{-1} &= B_{p,k} M_{p,k}^{-1} B_{p,k}^T - B_{p,k} M_{p,k}^{-1} B_{p,k}^T C^T (B_{r,k} M_{r,k}^{-1} B_{r,k}^T \\ &\quad + C B_{p,k} M_{p,k}^{-1} B_{p,k}^T C^T)^{-1} C B_{p,k} M_{p,k}^{-1} B_{p,k}^T. \end{aligned} \quad (4.33)$$

Further, we have

$$W_k^T M_k T_k = (B_{p,k}^{-1})^T M_{p,k} B_{p,k}^{-1} x_k^- + C^T (B_{r,k}^{-1})^T M_{r,k} B_{r,k}^{-1} y_k. \quad (4.34)$$

Substitute the (4.33) and (4.34) into (4.30), we have

$$x_k = x_k^- + \tilde{K}_k (y_k - C x_k^-) \quad (4.35)$$

with

$$\begin{aligned} \tilde{K}_k &= \tilde{P}_k^- C^T (C \tilde{P}_k^- C^T + \tilde{R}_k)^{-1} \\ \tilde{P}_k^- &= B_{p,k} M_{p,k}^{-1} B_{p,k}^T, \quad \tilde{R}_k = B_{r,k} M_{r,k}^{-1} B_{r,k}^T. \end{aligned} \quad (4.36)$$

The *a posteriori* error covariance can be given as

$$P_k^+ = (I - \tilde{K}_k C) P_k^- (I - \tilde{K}_k C)^T + \tilde{K}_k R_k \tilde{K}_k^T. \quad (4.37)$$

The detailed algorithm of the GMKMCKF is summarized in Algorithm 4.1.

Theorem 4.6. *The GMKMCKF is identical to the KF when $\alpha = 2$ and $\beta_i \rightarrow \infty$. It is equivalent to the traditional MCKF [79] when $\alpha = 2$ and $\beta_i = \sigma$. Moreover, it becomes the MKMCKF [148] when $\alpha = 2$ and $\beta_i = \sigma_i$.*

The proof of this theorem is shown in Appendix 4.5.6

4.2.2 Disturbance Estimation

We consider a disturbance estimation problem formulated in (4.1)-(4.5). The nominal noises are assumed to be Gaussian with $E(\bar{w}_k^T \bar{w}_k) = Q_k \succcurlyeq 0$, $E(\bar{v}_k^T \bar{v}_k) = R_k \succ 0$. Since the modeling of the disturbance is not accurate, the unmodeled part can be seen as non-Gaussian noises. Then, we can employ the GMKMCKF to estimate both the system state and the disturbance.

Remark 4.3. *As discussed in Section 4.1.4 and Section 4.1.5, the kernel parameters can be selected based on the shape of the influence function or the noise distribution. Generally, if w_k and v_k are heavy-tailed, we can select $1 < \alpha < 2$; if they are Gaussian, we*

Algorithm 4.1 GMKMCKF

```

1: Step 1: Initialization
2: Choose  $\alpha, \beta_1, \beta_2, \dots, \beta_{n+m}$ , maximum iteration number  $m_{iter}$ , and a threshold  $\epsilon$ .
3: Step 2: State Prediction
4:  $\hat{x}_k^- = A\hat{x}_{k-1}^+$ 
5:  $P_k^- = AP_{k-1}^+A^T + Q_k$ 
6: Obtain  $B_{p,k}$  with  $P_k^- = B_{p,k}B_{p,k}^T$ 
7: Obtain  $B_{r,k}$  with  $R_k = B_{r,k}B_{r,k}^T$ 
8: Step 3: State Update
9:  $\hat{x}_{k,0}^+ = \hat{x}_k^-$ 
10: while  $\frac{\|\hat{x}_{k,t}^+ - \hat{x}_{k,t-1}^+\|}{\|\hat{x}_{k,t-1}^+\|} > \epsilon$  or  $t \leq m_{iter}$  do  $\triangleright t$  starts from 1
11:  $\hat{x}_{k,t}^+ = \hat{x}_k^- + \tilde{K}_{k,t}(y_k - H\hat{x}_k^-)$ 
12:  $\tilde{K}_{k,t} = \tilde{P}_k^- H^T (H\tilde{P}_k^- H^T + \tilde{R}_k)^{-1}$ 
13:  $\tilde{P}_k^- = B_p M_p^{-1} B_p^T$ 
14:  $\tilde{R}_k = B_r M_r^{-1} B_r^T$ 
15:  $M_{p,k} = diag(g(e_{k,1}), \dots, g(e_{k,n}))$ 
16:  $M_{r,k} = diag(g(e_{k,n+1}), \dots, g(e_{k,n+m}))$ 
17:  $e_{i,k} = t_{i,k} - w_{i,k}\hat{x}_{k,t-1}^+$ 
18:  $t = t + 1$ 
19: end while
20:  $P_k^+ = (I - \tilde{K}_k H)P_k^- (I - \tilde{K}_k H)^T + \tilde{K}_k R_k \tilde{K}_k^T$ 

```

can use $\alpha = 2$; if w_k and v_k are light-tailed, we can use $\alpha > 2$. As for the bandwidth β_i , we employ a large bandwidth (e.g., $\beta_i = 10^8$) for channels x_k and y_k , and use a relatively small bandwidth for d_k since this channel contains heavy-tailed noise due to modeling mismatch.

4.3 Simulations

In this section, we employ the GMKMCKF as a disturbance observer for an angle tracking task using an exoskeleton. Moreover, we compared its performance with the ESO [111], KF-DOB [68], MCKF [79], and PF [134].

4.3.1 System Modeling

We consider a tracking problem using one-degree of freedom exoskeleton which is depicted in Figure 4.5. The purpose of the exoskeleton is to track a predefined angle θ_d with or without disturbance d .

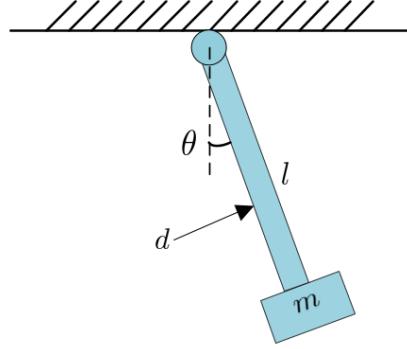


Figure 4.5: The exoskeleton.

The system dynamics of the exoskeleton can be written as

$$I_m \ddot{\theta} + b_m \dot{\theta} + k_m \theta + mgl \sin(\theta) = \tau + d \quad (4.38)$$

where I_m is the inertia, m is the mass, l is the link length, b_m is the damping, k_m is the stiffness, θ is the angle, g is the gravity constant, τ is the motor output, and d is the disturbance caused by unknown friction or the environment. To eliminate the nonlinear term in (4.38), we use the feedback linearization technique [149] by applying the control input $u_g = mgl \sin(\hat{\theta})$ ($\hat{\theta}$ is the estimated angle). In this case, the new model becomes

$$I_m \ddot{\theta} + b_m \dot{\theta} + k_m \theta = \bar{\tau} + d. \quad (4.39)$$

where $\bar{\tau} = \tau - u_g$. Then, equation (4.39) can be rewritten as a discrete state-space form by Euler discretization method

$$\begin{aligned} x_{k+1} &= Ax_k + Fu_k + w_k \\ y_k &= Cx_k + v_k \end{aligned} \quad (4.40)$$

with

$$\begin{aligned} A &= \begin{bmatrix} 1 & 0 & 0 \\ \frac{T}{I_m} & 1 - \frac{b_m T}{I_m} & -\frac{k_m T}{I_m} \\ 0 & T & 1 \end{bmatrix} \\ F &= \begin{bmatrix} 0 \\ \frac{T}{I_m} \\ 0 \end{bmatrix}, C = \begin{bmatrix} 0, 0, 1 \end{bmatrix} \end{aligned} \quad (4.41)$$

where $u_k = \bar{\tau}_k$, $x_k = [d_k, \dot{\theta}_k, \theta_k]^T$ including the disturbance, angular velocity, and angular acceleration, w_k is the process noise, v_k is the measurement noise, and T is the sampling time.

In simulation, the desired angle follows

$$\theta_{d,k} = 15 \sin(0.4\pi kT). \quad (4.42)$$

As for the controller, we use a feedforward term $u_{ff,k}$ to compensate the system dynamics, a feedback controller $u_{fb,k}$ to stabilize the plant, and a disturbance compensator $u_{d,k}$ to counteract the disturbance. The feedback controller is the PD controller [150]. The control signal has

$$u_k = u_{ff,k} + u_{d,k} + u_{fb,k} \quad (4.43)$$

with

$$\begin{cases} u_{ff,k} = I_m \ddot{\theta}_{d,k} + b \dot{\theta}_{d,k} + k \theta_{d,k} \\ u_{d,k} = -\hat{d}_k \\ u_{fb,k} = k_p (\theta_{d,k} - \hat{\theta}_k) + k_d (\dot{\theta}_{d,k} - \hat{\dot{\theta}}_k) \end{cases} \quad (4.44)$$

where $\ddot{\theta}_{d,k}$ is the desired angular acceleration, $\dot{\theta}_{d,k}$ is the desired angular velocity, $\theta_{d,k}$ is the desired angle, $\hat{\theta}_k$, $\hat{\dot{\theta}}_k$, and \hat{d}_k are the states and disturbance estimated by the observer, and k_p and k_d are control gains. The overall motor output is $\tau = u_k + u_g = u_k + mgl \sin \hat{\theta}$. Without considering the external disturbance, the stability of the controller (4.43) is proved in [150].

In simulation, the disturbance is step-like and follows

$$d_k = \begin{cases} 50 + w_{d,k}, 400 \leq k \leq 600 \\ w_{d,k}, \text{otherwise.} \end{cases} \quad (4.45)$$

The exoskeleton inertia is $I_m = 0.1 \text{ N}\cdot\text{m}\cdot\text{s}^2/\text{deg}$, the damping coefficient is $b_m = 1.0 \text{ N}\cdot\text{m}\cdot\text{s}/\text{deg}$, the stiffness coefficient is $k_m = 0.1 \text{ N}\cdot\text{m}/\text{deg}$, and the sampling time is $T = 0.01$ seconds. We compare the performances of the controller (4.43) using the GMKMCKF, KF-DOB, MCKF, ESO, and PF as an observer in two situations: 1) the state noises $w_{x,k}$ and measurement noises v_k are Laplacian; 2) the state noises $w_{x,k}$ and measurement noises v_k are Gaussian. In these two cases, we use the same measurement covariance, process covariance, and the initial error covariance for the KF-DOB, MCKF, and GMKMCKF. The particle number for the PF is $N = 1000$ while the resampling method is the systematic resample. To investigate the error performances of different observers, we conduct 100 independent Monte Carlo runs for each observer.

4.3.2 Laplace Distribution with Unknown Disturbance

For system dynamics in (4.40) with Laplace distribution, we assume that

$$\begin{aligned} w_{1,k} &\sim \mathcal{L}\left(0, \frac{0.1\sqrt{2}}{2}\right), w_{2,k} \sim \mathcal{L}\left(0, \frac{0.01\sqrt{2}}{2}\right) \\ w_{3,k} &\sim \mathcal{L}\left(0, \frac{0.01\sqrt{2}}{2}\right), v_k \sim \mathcal{L}\left(0, \frac{0.01\sqrt{2}}{2}\right). \end{aligned} \quad (4.46)$$

Since the modeling of the disturbance in (4.40) is not accurate, in the GMKMCKF, we select $\beta_1 = 1$ to suppress the heavy-tailed noises in the disturbance channel. As for other channels, we use $\beta_2 = \beta_3 = \beta_4 = 10^8$. We employ the shape parameter $\alpha = 1.6$ for the GMKMCKF1 and $\alpha = 2$ for the GMKMCKF2. The maximum iteration number in each sample interval is set to be $m_{iter} = 3$. The disturbance error using different observers is shown in Figure 4.6. The corresponding tracking angle error is shown in Figure 4.7. The root mean squared errors (RMSEs) of the x_1 (disturbance, Nm), x_2 (angular velocity, deg/s), x_d (angle, deg), and $\theta_d - \theta_a$ (tracking error, deg) are summarized in Table 4.1. One can see that the GMKMCKF1 outperforms the others observers, which reveals that $\alpha < 2$ is more suitable for heavy-tailed distributions.

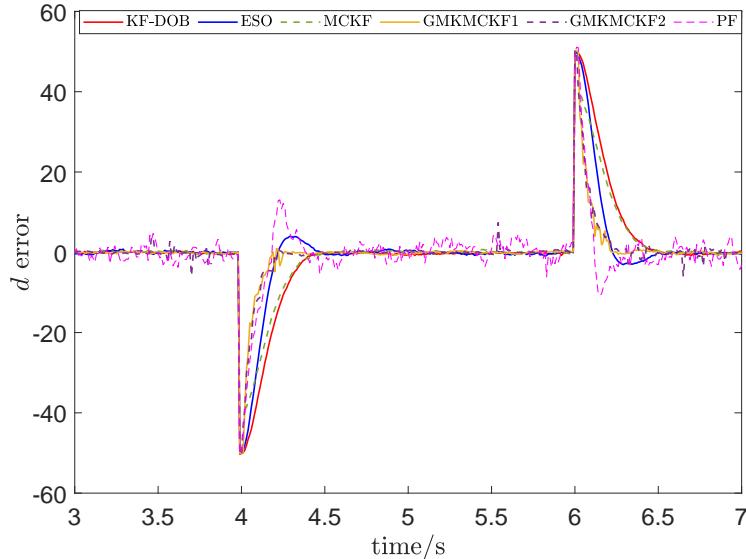


Figure 4.6: Disturbance errors of different observers. The step-like disturbance is added at $t = 4$ s and disappears at $t = 6$ s.

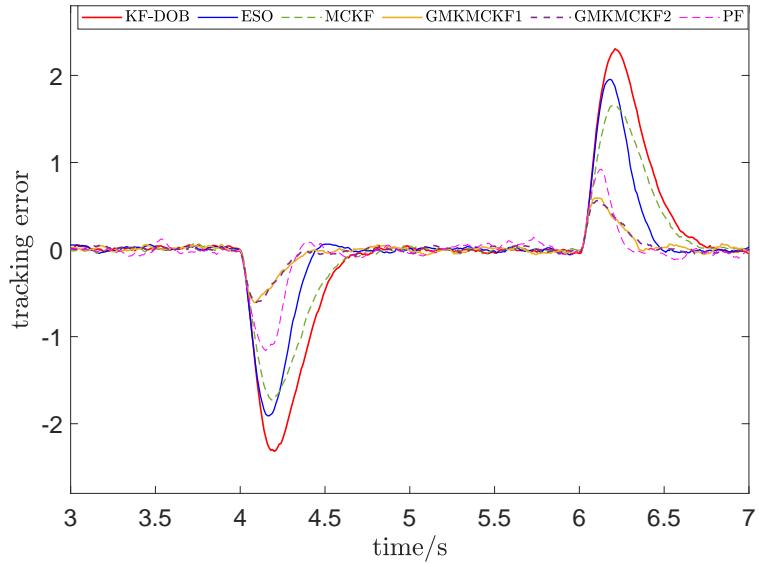


Figure 4.7: Tracking errors of different observers.

Table 4.1: Performances of Different Observers with Laplace Noises.

Observer	x_1 (N·m)	x_2 (deg/s)	x_3 (deg)	$\theta_d - \theta_a$ (deg)
KF-DOB	8.0460	4.1637	0.0241	0.4951
ESO	6.8374	3.3209	0.0198	0.3606
MCKF	6.5352	3.4596	0.0249	0.3648
GMKMCKF1	4.9361	0.7900	0.0083	0.1085
GMKMCKF2	5.0331	0.9480	0.0088	0.1086
PF	5.3955	1.5472	0.0100	0.1473

4.3.3 Gaussian Distribution with Unknown Disturbance

We consider the Gaussian noises for (4.40) as follows

$$\begin{aligned} w_{1,k} &\sim \mathcal{N}(0, 0.01), w_{2,k} \sim \mathcal{N}(0, 0.0001) \\ w_{3,k} &\sim \mathcal{N}(0, 0.0001), v_k \sim \mathcal{N}(0, 0.0001). \end{aligned} \tag{4.47}$$

Similarly, we apply $\beta_1 = 1$ for the disturbance channel, and $\beta_2 = \beta_3 = \beta_4 = 10^8$ for other channels in the GMKMCKF. Moreover, We employ the shape parameter $\alpha = 1.6$ for the GMKMCKF1 and $\alpha = 2$ for the GMKMCKF2. The maximum iteration number is set to be $m_{iter} = 3$. The RMSEs of different observers are summarized in Table 4.2. One can see the GMKMCKF2 outperforms the others which indicates that $\alpha = 2$ is a better option for Gaussian distributions.

Table 4.2: Performances of Different Observers with Gaussian Noises.

Observer	x_1 (N·m)	x_2 (deg/s)	x_3 (deg)	$\theta_d - \theta_a$ (deg)
KF-DOB	8.0397	4.1571	0.0241	0.4350
ESO	6.8507	3.3302	0.0198	0.3182
MCKF	6.5286	3.4494	0.0250	0.3626
GMKMCKF1	4.9886	0.8224	0.0082	0.0835
GMKMCKF2	4.9292	0.7536	0.0082	0.0780
PF	5.2685	1.3566	0.0097	0.1005

4.4 Summary

In this chapter, we formulate the unknown disturbance dynamics as the summation of a nominal model and a ε -contaminated mixture noise. To suppress the heavy-tailed noises, we present a generalized multi-kernel correntropy, providing its properties and comparing it with the LMP criterion. Further, we derive a novel GMKMCKF for disturbance estimation, which is an extension of the MKMCKF when some channels of a system contain heavy-tailed noises and can achieve excellent performance both with and without disturbance. Simulations on an exoskeleton verify the effectiveness of the proposed method.

4.5 Appendix

4.5.1 Proof of Theorem 4.1

Proof. In the case of $0 < \alpha \leq 2$, the GGD $\kappa_{\alpha, \beta_i}(x_i, y_i) = G_{\alpha, \beta_i}(x_i, y_i)$ is positive definite (see [151], p434), which induces a mapping function Φ from input space to infinite dimensional reproducing kernel Hilbert space (RKHS) with $\kappa_{\alpha, \beta_i}(x_i, y_i) = \Phi(x_i)^T \Phi(y_i)$. For random pairs $\mathcal{X}_i, \mathcal{Y}_i$ in (4.10), we obtain $C_{\alpha, \beta_i}(\mathcal{X}_i, \mathcal{Y}_i) = \frac{1}{N} \sum_{k=1}^N G_{\alpha, \beta_i}(x_i(k), y_i(k)) = E[\Phi(x_i)^T \Phi(y_i)]$. Thus, $\hat{C}(\mathcal{X}, \mathcal{Y}) = \sum_{i=1}^l \beta_i^\alpha C_{\alpha, \beta_i}(\mathcal{X}_i, \mathcal{Y}_i) = \sum_{i=1}^l \beta_i^\alpha E[\Phi(x_i)^T \Phi(y_i)]$. This completes the proof. \square

4.5.2 Proof of Theorem 4.2

Proof. Equation (4.11) can be rewritten as

$$J_{GL}(\mathcal{X}, \mathcal{Y}) = \sum_{i=1}^l \beta_i^\alpha (1 - E[G_{\sigma, \beta_i}(x_i, y_i)]) \quad (4.48)$$

Taking Taylor series expansion of the generalized Gaussian density function $G_{\sigma, \beta_i}(x_i, y_i)$, we have

$$J_{GL}(\mathcal{X}, \mathcal{Y}) = \sum_{i=1}^l \beta_i^\alpha \left(1 - E \left[\sum_{n=0}^{\infty} \frac{(-1)^n}{\beta_i^{\alpha n} n!} |x_i - y_i|^{\alpha n} \right] \right). \quad (4.49)$$

When $\beta_i^\alpha \rightarrow \infty$, we obtain

$$\lim_{\beta_i^\alpha \rightarrow \infty} J_{GL}(\mathcal{X}, \mathcal{Y}) = \sum_{i=1}^l E[|x_i - y_i|^\alpha] = E[\|\mathcal{X} - \mathcal{Y}\|_\alpha^\alpha] \quad (4.50)$$

This completes the proof. \square

4.5.3 Proof of Theorem 4.3

Proof. When $0 < \alpha \leq 2$, the kernel $\kappa_{\alpha, \beta_i}(x_i, y_i) = G_{\alpha, \beta_i}(x_i, y_i) = \exp\left(-\left|\frac{e_i}{\beta}\right|^\alpha\right)$ is a Mercer kernel (see [151], p434) with $e_i = x_i - y_i$, which induces a mapping function Φ from input space to infinite dimensional RKHS with $\kappa_{\alpha, \beta_i}(x_i, y_i) = \langle \Phi(x_i), \Phi(y_i) \rangle_{\mathcal{F}}$. Then, it is clear that GCIM satisfies: 1) Negativity: $\text{GCIM}(\mathcal{X}, \mathcal{Y}) \geq 0$; 2) Identities of indiscernibles: $\text{GCIM}(\mathcal{X}, \mathcal{Y}) = 0$ iff $\mathcal{X} = \mathcal{Y}$; 3) Symmetry: $\text{GCIM}(\mathcal{X}, \mathcal{Y}) = \text{GCIM}(\mathcal{Y}, \mathcal{X})$. For the triangle inequality: $\text{GCIM}(\mathcal{X}, \mathcal{Z}) \leq \text{GCIM}(\mathcal{X}, \mathcal{Y}) + \text{GCIM}(\mathcal{Y}, \mathcal{Z})$. We construct vectors $\tilde{\mathcal{X}}_i = [\Phi(x_i(1)), \Phi(x_i(2)), \dots, \Phi(x_i(N))]^T$ and $\tilde{\mathcal{Y}}_i =$

$[\Phi(y_i(1)), \Phi(y_i(2)), \dots, \Phi(y_i(N))]^T$ in Hilbert space \mathcal{F}^N for random pairs $\tilde{\mathcal{X}}_i, \tilde{\mathcal{Y}}_i$. Then, the square of the Euclidean distance $D(\tilde{\mathcal{X}}_i, \tilde{\mathcal{Y}}_i)$ has

$$\begin{aligned} D^2(\tilde{\mathcal{X}}_i, \tilde{\mathcal{Y}}_i) &= \langle \tilde{\mathcal{X}}_i - \tilde{\mathcal{Y}}_i, \tilde{\mathcal{X}}_i - \tilde{\mathcal{Y}}_i \rangle \\ &= \langle \tilde{\mathcal{X}}_i, \tilde{\mathcal{X}}_i \rangle - 2\langle \tilde{\mathcal{X}}_i, \tilde{\mathcal{Y}}_i \rangle + \langle \tilde{\mathcal{Y}}_i, \tilde{\mathcal{Y}}_i \rangle \\ &= \sum_{k=1}^N \kappa_{\alpha, \beta_i}(0) - \sum_{k=1}^N \kappa_{\alpha, \beta_i}(x_i(k), y_i(k)) + \sum_{k=1}^N \kappa_{\alpha, \beta_i}(0) \\ &= 2N(1 - C_{\alpha, \beta_i}(\tilde{\mathcal{X}}_i, \tilde{\mathcal{Y}}_i)) \end{aligned} \quad (4.51)$$

Then, based on the property of Euclidean distance $D(\tilde{\mathcal{X}}_i, \tilde{\mathcal{Z}}_i) \leq D(\tilde{\mathcal{X}}_i, \tilde{\mathcal{Y}}_i) + D(\tilde{\mathcal{Y}}_i, \tilde{\mathcal{Z}}_i)$, and using the Minkowski inequality [152], we have

$$\begin{aligned} \left(\sum_{i=1}^p D^2(\tilde{\mathcal{X}}_i, \tilde{\mathcal{Z}}_i) \right)^{1/2} &\leq \left(\sum_{i=1}^p (D(\tilde{\mathcal{X}}_i, \tilde{\mathcal{Y}}_i) + D(\tilde{\mathcal{Y}}_i, \tilde{\mathcal{Z}}_i))^2 \right)^{1/2} \\ &\leq \left(\sum_{i=1}^p D^2(\tilde{\mathcal{X}}_i, \tilde{\mathcal{Y}}_i) \right)^{1/2} + \left(\sum_{i=1}^p D^2(\tilde{\mathcal{Y}}_i, \tilde{\mathcal{Z}}_i) \right)^{1/2} \end{aligned} \quad (4.52)$$

Substitute (4.51) into $\text{GCIM}(\mathcal{X}, \mathcal{Z})$, we have

$$\begin{aligned} \text{GCIM}(\mathcal{X}, \mathcal{Z}) &= \left(\sum_{i=1}^l \beta_i^\alpha \left(\frac{D^2(\tilde{\mathcal{X}}_i, \tilde{\mathcal{Z}}_i)}{2N} \right) \right)^{1/2} \\ &\leq \left(\sum_{i=1}^l \beta_i^\alpha \left(\frac{D^2(\tilde{\mathcal{X}}_i, \tilde{\mathcal{Y}}_i)}{2N} \right) \right)^{1/2} + \left(\sum_{i=1}^l \beta_i^\alpha \left(\frac{D^2(\tilde{\mathcal{Y}}_i, \tilde{\mathcal{Z}}_i)}{2N} \right) \right)^{1/2} \\ &= \text{GCIM}(\mathcal{X}, \mathcal{Y}) + \text{GCIM}(\mathcal{Y}, \mathcal{Z}). \end{aligned} \quad (4.53)$$

This completes the proof. \square

4.5.4 Proof of Theorem 4.4

Proof. A differentiable function $f: \mathbb{R}^l \rightarrow \mathbb{R}$ is said to be invex, if and only if

$$f(x_2) \geq f(x_1) + q(x_1, x_2)^T \nabla f(x_1). \quad (4.54)$$

In the case of $N = 1$, the GL and its gradient are shown in (4.13) and (4.14). Then, we have $J_{GL}(e) > 0$ for any $e \neq \mathbf{0}$ and $J_{GL}(\mathbf{0}) = 0$, which indicate $J_{GL}(\mathbf{0})$ is a global minima of J_{GL} . We construct the vector valued function $q(e_1, e_2)$ as follow:

$$q(e_1, e_2) = \begin{cases} \frac{J_{GL}(e_2) - J_{GL}(e_1)}{\nabla J_{GL}(e_1)^T \nabla J_{GL}(e_1)} \nabla J_{GL}(e_1), & e_1 \neq \mathbf{0} \\ \mathbf{0}, & e_1 = \mathbf{0} \end{cases}. \quad (4.55)$$

Then, it holds that

$$J_{GL}(e_2) \geq J_{GL}(e_1) + q(e_1, e_2)^T \nabla J_{GL}(e_1) \quad (4.56)$$

for any $e_1, e_2 \in \mathbb{R}^l$. This completes the proof. \square

4.5.5 Proof of Theorem 4.5

Proof. Taking Taylor series expansion of $G_{\alpha,\beta_i}(e_i)$, one has

$$G_{\alpha,\beta_i}(e_i) = \sum_{n=0}^{\infty} \frac{(-1)^n}{\beta_i^{\alpha n} n!} |e_i|^{\alpha n}. \quad (4.57)$$

Then, we have

$$\begin{aligned} \lim_{\beta_i^\alpha \rightarrow \infty} J_{GL}(e) &= \lim_{\beta_i^\alpha \rightarrow \infty} \sum_{i=1}^l \beta_i^\alpha (1 - G_{\alpha,\beta_i}(e_i)) \\ &= \sum_{i=1}^l |e_i|^\alpha \\ &= \|e\|_\alpha^\alpha. \end{aligned} \quad (4.58)$$

In this case, $J_{GL}(e)$ is identical to $J_{LMP}(e)$ with $\alpha = p$. The Hessian matrix of $J_{GL}(e)$ has

$$\begin{aligned} H(J_{GL}) &= \frac{\partial \nabla J_{GL}}{\partial e} \\ &= \begin{bmatrix} \zeta_1, & 0, & \dots, 0 \\ 0, & \zeta_1, & \dots, 0 \\ \vdots, & \vdots, & \dots, \vdots \\ 0, & 0, & \dots, \zeta_l \end{bmatrix} \end{aligned} \quad (4.59)$$

with

$$\zeta_i = -\frac{\alpha e^{-\frac{|e_i|^\alpha}{\beta_i^\alpha}} |e_i|^\alpha (\alpha |e_i|^\alpha - (\alpha - 1)\beta_i^\alpha)}{\beta_i^\alpha e_i^2}, i = 0, 1, \dots, l. \quad (4.60)$$

One can see that $H(J_{GL})$ is a diagonal matrix. When $0 < \alpha \leq 1$, we have $H(J_{GL}) \prec 0$ for any $e \neq 0$. Thus, $H(J_{GL})$ is concave in this case. When $\alpha > 1$ and $|e_i| \leq (\frac{\alpha-1}{\alpha})^{\frac{1}{\alpha}} \beta_i$, we have $H(J_{GL}) \succ 0$. Then, $H(J_{GL})$ is convex in this situation. This completes the proof. \square

4.5.6 Proof of Theorem 4.6

Proof. When $\alpha = 2$, and $\sigma_i \rightarrow \infty$, we have $\tilde{M}_p = I_{n \times n}$ and $\tilde{M}_r = I_{m \times m}$. Then, GMKM-CKF is equal to KF. As $\sigma_i = 2$ and $\beta_i = \sigma$, we have $M_p = \text{diag}[G_\sigma(e_{1,k}, \dots, e_{n,k})]$, $M_r = \text{diag}[G_\sigma(e_{n+1,k}, \dots, e_{n+m,k})]$. Then, it is identical to the MCKF. When $\alpha = 2$ and $\beta_i = \sigma_i$, we have $M_p = \text{diag}[G_{\sigma_p}(e_{1,k}, \dots, e_{n,k})]$, $M_r = \text{diag}[G_{\sigma_r}(e_{n+1,k}, \dots, e_{n+m,k})]$ with $\sigma_p = [\sigma_1, \dots, \sigma_n]^T$ and $\sigma_r = [\sigma_{n+1}, \dots, \sigma_{n+m}]^T$. In this case, it becomes the MKMCK-F. \square

Chapter 5

Preference-based Assistance Map Learning with Robust Adaptive Oscillators

In this chapter, we present a robust adaptive oscillator (RAO) to synchronize the human-robot movement and extract gait features. Then, we use the Gaussian process regression (GPR) to map the preference-based assistance parameters to gait features. Experiments show that RAO has a faster convergence rate and is more accurate compared with AO. Meanwhile, the learning efficiency of the GBR is superior to the HIL. The effectiveness of the preference-based assistance is validated by a hip exoskeleton through walking at a speed of 5 km/h.

5.1 Materials and Methods

5.1.1 Approach Overview

A hip exoskeleton prototype provided by Xeno Dynamics Co. Ltd is used. The control board is installed on the back of the device, which collects absolute joint angle and interactive torque from sensors at the hip joints. Belts at the shoulders, waist, and thighs are employed to avoid slack and assure better torque transfer. The mass of the exoskeleton concentrates on the waist and its weight is 4.5 Kg. A diagram of the hip exoskeleton is shown in Figure 5.1.

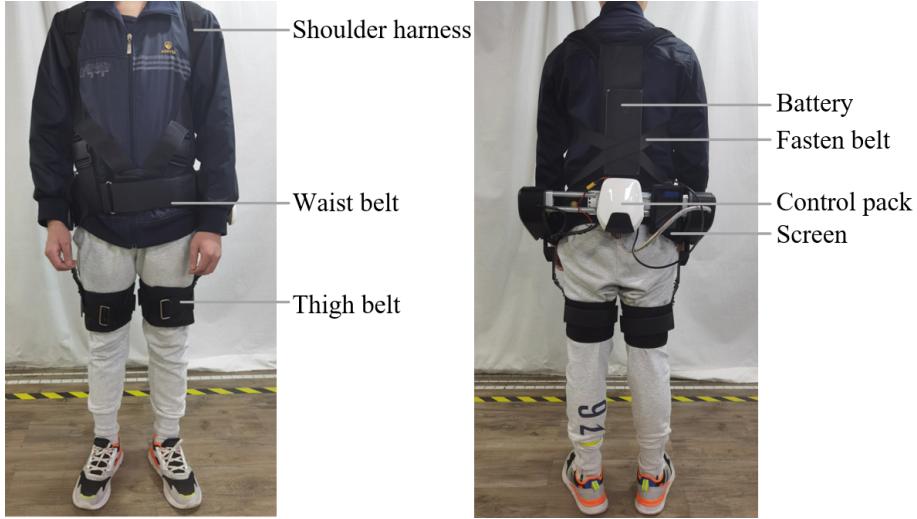


Figure 5.1: Front and rear view of the hip exoskeleton. An elastic fasten belt is used to avoid the exoskeleton waist flip.

The overall control architecture is shown in Figure 5.2. It comprises three components. The first component is the gait perception module, where a RAO is deployed to obtain the gait cycle percentage c (i.e., the percentage of a stride) and extract the gait features g . The input of this module is the difference between two hip angles $q = \theta_r - \theta_l$ where θ_r and θ_l is the right and left hip angle, respectively. As for the exoskeleton controller module, the desired torque is designed based on gait cycle percentage and is controlled by parameter set $\Psi = \{ext_{max}, ext_{peak}, fle_{max}, fle_{peak}\}$. Then, a low-level torque controller is designed to track the desired torque. The final component is the assistance map learning module which includes a training session and a learning session. In the training session, users are required to walk at different speeds on a treadmill with different gait features (i.e., different gait amplitude and frequencies). Then, they explore and submit their preferred assistance parameter set Ψ through a human-robot interaction interface. The GPR predicts the user-preferred assistance profiles based on historical data. The objective of this module is to minimize the difference between the human selected parameter set Ψ and the learned parameter set $\bar{\Psi}$ based on the training set. After the learning procedure, the exoskeleton can automatically provide an individualized assistance profile when walking at different speeds.

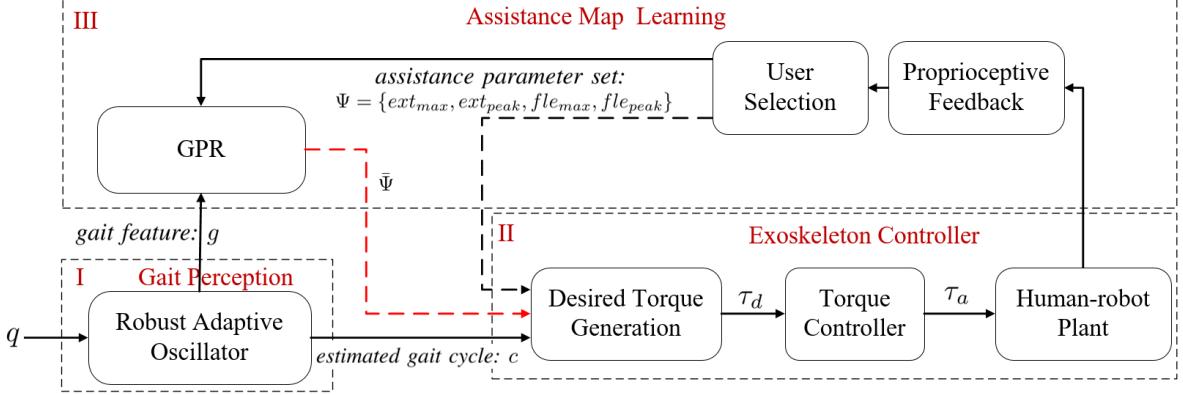


Figure 5.2: Overview of the control architecture. First, gait is processed by a robust adaptive oscillator and its output is gait cycle percentage c and gait features g . Then, the desired torque is generated based on c and its shape is controlled by a human-machine interface. After that, the exoskeleton carries out the command and assists the subject. In this process, the user can adjust the assistance profile and submit his (or her) satisfied parameters based on their proprioceptive feedback. A Gaussian process regression (GPR) module is employed to predict the user's preference and to map the gait features g to a parameter set Ψ . After the learning procedure, the exoskeleton can automatically provide suitable assistance based on the user's gait feature without the need of manipulating the human-robot interface.

5.1.2 Robust Adaptive Oscillators

In the conventional AO, a periodic signal q is estimated by a pool of sinusoidal curves with

$$\hat{q} = \sum_{i=0}^K \alpha_i \sin(\phi_i) = \sum_{i=0}^K \alpha_i \sin(iwt + \beta_i) \quad (5.1)$$

where K denotes the order of the harmonics, w is the fundamental frequency, α_i is the amplitude for the i -th order harmonic, and β_i is the offset phase. Note that ϕ_0 is set to be $\pi/2$ so that α_0 is a bias term. The learning rule is shown as follows.

$$\begin{cases} \dot{\phi}_i(t) = iw(t) + v_\phi \frac{e(t)}{\sum_{i=0}^K \alpha_i(t)} \cos \phi_i(t) \\ \dot{\alpha}_i(t) = \eta e(t) \sin \phi_i(t) \\ \dot{w}(t) = v_w \frac{e(t)}{\sum_{i=0}^K \alpha_i(t)} \cos \phi_1(t) \end{cases} \quad (5.2)$$

with

$$e(t) = q(t) - \hat{q}(t)$$

where $q(t)$ is the input signal, $\hat{q}(t)$ is the estimated signal obtained by (5.1), $e(t)$ is the tracking error, and v_ϕ , v_w , and η are learning gains.

Ronsse et al. [124] showed that the traditional AO needs about 2–5 periods for convergence. Inevitably, a gait phase estimation error may be caused by gait transition, e.g., switching from a slow walking pattern to a rapid walking pattern. To handle this problem, we use the difference between the right and left hip angle as input signals ($q = \theta_r - \theta_l$) and employ the zero-cross point ($q = 0, \dot{q} > 0$) to identify the zero phase (see Figure 5.3). Then, we estimate the gait frequency by

$$\bar{w}(t) = \frac{2\pi}{T_k} = \frac{2\pi}{t_{z,k} - t_{z,k-1}} \quad (5.3)$$

where $\bar{w}(t)$ is the estimated frequency, $t_{z,k}$ and $t_{z,k-1}$ are the latest two adjacent zero-cross moments, and T_k is the gait period. We use the following new frequency update rule:

$$\dot{w}(t) = v_w \frac{e(t)}{\sum_{i=0}^K \alpha_i(t)} \cos \phi_1(t) + k_w (\bar{w}(t) - w(t)). \quad (5.4)$$

with

$$k_w = \begin{cases} \xi_w \exp(-\frac{q^2}{2\sigma_w^2}) & 0 \leq q \leq d_w, \dot{q} > 0 \\ 0, & \text{otherwise} \end{cases} \quad (5.5)$$

where k_w is a nonlinear gain, ξ_w is the amplitude, σ_w is the bandwidth, d_w is the correction region, q is the difference of two hip angles, and \dot{q} is the derivative of q . A graphic illustration of k_w is shown in Figure 5.3. One can see that k_w has a maximum value at the zero-cross moment, and then decreases rapidly. Compared with a constant gain in [41], our proposed nonlinear gain is more reasonable since the estimated frequency $\bar{w}(t)$ achieves a best estimation at the zero-cross point. Then, its accuracy declines rapidly and the frequency estimation is dominated by the original learning rules.

In order to obtain a continuous gait phase between two adjacent zero-cross points, we integrate the gait frequency w and obtain the auxiliary phase as follows:

$$\phi_{aux} = \begin{cases} 0, t = t_{z,k} \\ \int w(t) dt, \text{ otherwise} \end{cases} \quad (5.6)$$

where $t_{z,k}$ is the zero-cross moment. Note that ϕ_{aux} is different from the fundamental harmonic phase ϕ_1 since ϕ_{aux} resets at the zero-cross point. The auxiliary phase ϕ_{aux} is *not* involved in the system dynamics so that it does not affect the oscillator stability. Note that the endpoint of ϕ_{aux} in a period may not be aligned with 2π . To cope with this issue, we correct it as follows:

$$\phi_{aux,c} = \phi_{aux} + k_\phi (2\pi - \phi_{aux}) \quad (5.7)$$

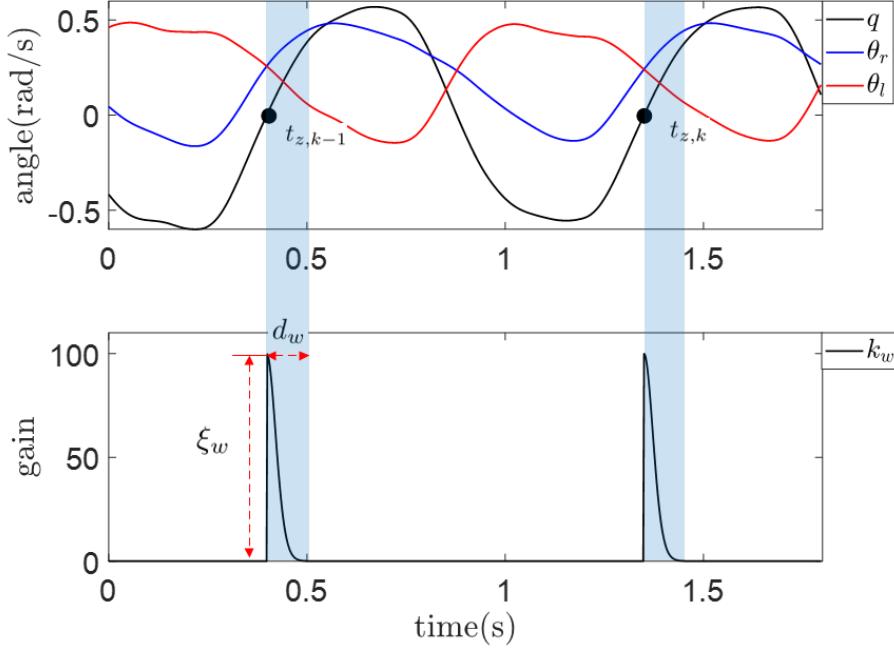


Figure 5.3: A graphic illustration of k_w . The black solid dots represent the zero-cross point of q , θ_r and θ_l are the right and left hip angle, ξ_w represents the amplitude, and d_w represents the correction region.

with

$$k_\phi = \begin{cases} \xi_\phi \exp\left(-\frac{q^2}{2\sigma_\phi^2}\right) & -d_\phi \leq q \leq 0, \dot{q} > 0 \\ 0, & \text{otherwise} \end{cases} \quad (5.8)$$

where ϕ_{aus} is the auxiliary phase, $\phi_{aux,c}$ is the corrected auxiliary phase, k_ϕ is a nonlinear gain, ξ_ϕ is the amplitude, d_ϕ is the correction window length, and σ_ϕ is the bandwidth. One can see that $\phi_{aux,c}$ is dominated by ϕ_{aus} when q is far away from the next zero-cross point, and is dominated by 2π when it approaches to the zero-cross point. Finally, the gait cycle percentage is estimated by

$$c = \frac{\mod(\phi_{aux,c} - \pi/2, 2\pi)}{2\pi} \quad (5.9)$$

where $c \in [0, 1)$ is the gait cycle percentage, \mod is the modulo operation, and $\pi/2$ is an offset term (note that the zero phase belongs to the mid-swing phase, and $\pi/2$ is the estimated heel strike moment). The gait feature g is defined as

$$g = [w, \alpha_1, w\alpha_1]^T \quad (5.10)$$

where w is the estimated fundamental frequency, α_1 is the amplitude for the fundamental harmonic. By involving two nonlinear gains in equations (5.5) and (5.8), our RAO has a faster convergence rate compared with the AO with gait transition. A performance comparison is shown in Section 5.2.1.

In a practical application, it is necessary to classify walking and standing. The binary classifier is shown as follows:

$$\begin{cases} r = (q^2 + \gamma \dot{q}^2)^{0.5} > r_{start}, & \text{walking} \\ r = (q^2 + \gamma \dot{q}^2)^{0.5} < r_{stop}, & \text{standing} \end{cases} \quad (5.11)$$

where r is a distance metric, γ is a constant coefficient, r_{start} and r_{stop} are two thresholds. To avoid misclassification caused by disturbance, q and \dot{q} are smoothed by a Butterworth filter. If the output of the classifier is walking, an assisted torque will be applied to the subject. Otherwise, the exoskeleton works at a zero torque mode.

5.1.3 Exoskeleton Controller

We use the pointwise minimum jerk curve [153] for the commanded torque generation. The torque is designed based on the gait cycle percentage which is shown in Figure 5.4. Six parameters control the shape of the profile. Among them, the offset $offset$ and extension duration ext_{dur} are set to be user-specific but do not change with walking speed. Other parameters, i.e., the extension peak time ext_{peak} , the maximum extension torque ext_{max} , the flexion peak time fle_{peak} and maximum flexion torque fle_{max} , are speed-dependent and can be tuned in the training procedure.

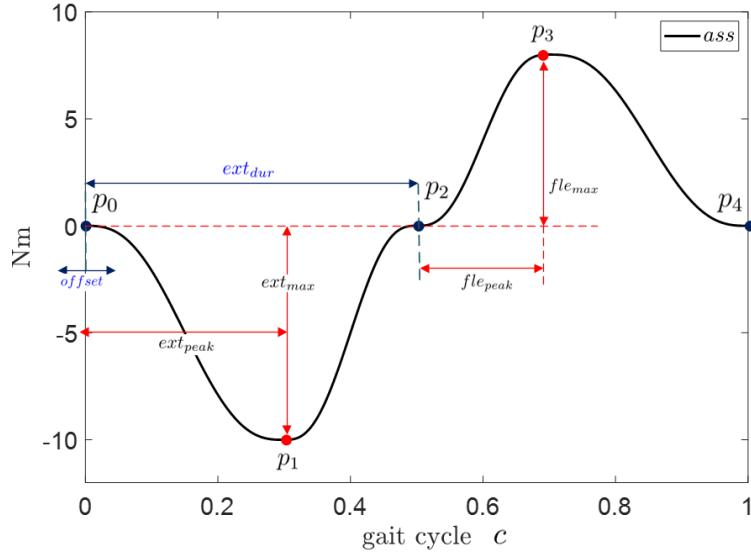


Figure 5.4: Desired torque generator. The profile is controlled by both the speed-independent parameters (i.e., $offset$, ext_{dur}) and the speed-dependent parameters (i.e., ext_{peak} , ext_{max} , fle_{peak} , and fle_{max}). The speed-independent parameters are determined before the training session. At the training session, users select and submit their preferred speed-dependent parameters when walking at different speeds.

The torque controller is designed as

$$u = -\tilde{d} + k_p(\tau_d - \tau_a) + k_d \dot{\tau}_d \quad (5.12)$$

where \tilde{d} is the estimated disturbance obtained by a nonlinear disturbance observer (N-DOB) [52], τ_d is the desired torque, τ_a is the actual torque measured by torque sensor, k_p is a proportional term, and k_d is a feedforward term. The performance of this controller is shown in Section 5.2.2.

5.1.4 Assistance Map Learning

The block diagram of the assistance map learning is shown on the top of Figure 5.2. It has two parts: a training part and a learning part. In the training part, the subject manipulates the exoskeleton and submits the preferred assistance parameters based on the proprioceptive feedback. Meanwhile, gait features are extracted by the RAO and paired with the preferred parameters. In the learning part, the GPR is utilized to construct assistance maps based on the historical data. After the learning procedure, the exoskeleton can automatically provide assistance.

The GPR's inputs are the gait feature g extracted by a RAO and the user selected parameter set Ψ . Its output is the predicted assistance parameter set $\bar{\Psi} = [\bar{ext}_{max}, \bar{ext}_{peak}, \bar{fle}_{max}, \bar{fle}_{peak}]^T$. We assume that the output elements are independent with each other. Then, we estimate the output elements one by one. For brevity, we use y to denote any output element. Then, we have

$$\begin{cases} f(g) = \Phi(g)^T \mathbf{w} \\ y = f(g) + \epsilon \end{cases} \quad (5.13)$$

where $g \in \mathbb{R}^3$ is the gait feature vector, $f(g)$ maps g to a scalar, $\Phi(g)$ is a kernel function, \mathbf{w} is the weight, and $\epsilon \sim \mathcal{N}(0, \sigma^2 I)$ is the Gaussian noise with zero mean and covariance σ^2 .

We denote the training set as $\mathbb{D} = \{(g_i, y_i) | i = 1, 2, \dots, n\} = \{G, \mathbf{y}\}$ where $G \in \mathbb{R}^{3 \times n}$ represents the input matrix, $\mathbf{y} \in \mathbb{R}^{n \times 1}$ represents target vector, and n is the number of samples. Then, the probability density of the observation set \mathbf{y} given G and \mathbf{w} can be

written as:

$$\begin{aligned}
p(\mathbf{y}|G, \mathbf{w}) &= \prod_{i=1}^n p(y_i|g_i, \mathbf{w}) \\
&= \prod_{i=1}^n \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(y_i - \Phi(g_i)^T \mathbf{w})^2}{2\sigma^2}\right) \\
&= \frac{1}{(2\pi\sigma^2)^{1/2}} \exp\left(-\frac{1}{2\sigma^2} |\mathbf{y} - \Phi(G)^T \mathbf{w}|^2\right) \\
&\sim \mathcal{N}(\Phi(G)^T \mathbf{w}, \sigma^2 I_{n \times n})
\end{aligned} \tag{5.14}$$

where $\Phi(G)$ is the aggregation of columns $\Phi(g_i)$. Based on Bayes' rule, we have

$$p(\mathbf{w}|\mathbf{y}, G) = \frac{p(\mathbf{y}|G, \mathbf{w})p(\mathbf{w})}{p(\mathbf{y}|G)} \tag{5.15}$$

where the prior weight is assumed to be Gaussian with $\mathbf{w} \sim \mathcal{N}(0, \Sigma)$. Since the term $p(\mathbf{y}|G)$ is independent of \mathbf{w} , we have

$$\begin{aligned}
p(\mathbf{w}|\mathbf{y}, G) &\propto p(\mathbf{y}|G, \mathbf{w})p(\mathbf{w}) \\
&\propto \exp\left(-\frac{1}{2\sigma^2} (\mathbf{y} - \Phi(G)^T \mathbf{w})^T (\mathbf{y} - \Phi(G)^T \mathbf{w})\right) \exp\left(-\frac{1}{2} \mathbf{w}^T \Sigma \mathbf{w}\right) \\
&\propto \exp\left(-\frac{1}{2} (\mathbf{w} - \bar{\mathbf{w}})^T \left(\frac{1}{\sigma_2} \Phi(G) \Phi(G)^T + \Sigma^{-1}\right) (\mathbf{w} - \bar{\mathbf{w}})\right) \\
&\sim \mathcal{N}(\bar{\mathbf{w}}, A^{-1}).
\end{aligned} \tag{5.16}$$

with

$$A = \sigma^{-2} \Phi(G) \Phi(G)^T + \Sigma^{-1}$$

$$\bar{\mathbf{w}} = \sigma^{-2} A^{-1} \Phi(G) \mathbf{y}$$

For a new input g_* , the predictive function $f_* \triangleq f(g_*)$ can be derived as an average over all possible weights. Thus, we have

$$\begin{aligned}
p(f_*|g_*, G, \mathbf{y}) &= \int p(f_*|g_*, \mathbf{w}) p(\mathbf{w}|G, \mathbf{y}) d\mathbf{w} \\
&= \mathcal{N}(\sigma^{-2} \Phi(g_*)^T A^{-1} \Phi(G) \mathbf{y}, \Phi(g_*)^T A^{-1} \Phi(g_*)).
\end{aligned} \tag{5.17}$$

To avoid the inversion of A matrix with size $n \times n$ which may be not convenient if n is large, equation (5.17) can be rewritten as

$$\begin{aligned}
p(f_*|g_*, G, \mathbf{y}) &= \mathcal{N}(K_*^T (K + \sigma I)^{-1} \mathbf{y}, \\
&\quad K_{**} - K_*^T (K + \sigma I)^{-1} K_*)
\end{aligned} \tag{5.18}$$

where we define $K = \Phi^T \Sigma \Phi$, $K_* = \Phi^T \Sigma \Phi_*$, $K_{**} = \Phi_*^T \Sigma \Phi_*$, $\Phi_* = \Phi(g_*)$, and $\Phi = \Phi(G)$.

Thus, we have

$$\begin{aligned}
\bar{f}_* &= K_*^T (K + \sigma I)^{-1} \mathbf{y} \\
V(f_*) &= K_{**} - K_*^T (K + \sigma I)^{-1} K_*
\end{aligned} \tag{5.19}$$

where \bar{f}_* is the mean of f_* and $V(f_*)$ is the variance. More details can be found in [143, p.12].

The prediction probability in equation (5.18) is highly related to the covariance function K . In this paper, we adopt the squared exponential covariance function as

$$k(g_p, g_q) = \sigma_f^2 \exp\left(-\frac{1}{2}(g_p - g_q)^T \Sigma_l (g_p - g_q)\right) \quad (5.20)$$

where g_p, g_q are two input vectors, σ_f is the amplitude, Σ_l is a diagonal matrix. To optimize the hyperparameters $\boldsymbol{\theta} = (\sigma_f, \Sigma_l, \sigma)$, we calculate the log *marginal likelihood* as

$$\begin{aligned} \log p(\mathbf{y}|G, \boldsymbol{\theta}) &= -\frac{1}{2}\mathbf{y}^T(K + \sigma^2 I)^{-1}\mathbf{y} \\ &\quad - \frac{1}{2}\log|K + \sigma^2 I| - \frac{n}{2}\log 2\pi. \end{aligned} \quad (5.21)$$

Then, the optimal hyperparameters $\boldsymbol{\theta}$ can be obtained by solving the following equation

$$\begin{aligned} \frac{\partial}{\partial \boldsymbol{\theta}_j} \log(\mathbf{y}|G, \boldsymbol{\theta}) &= \frac{1}{2}\mathbf{y}^T K^{-1} \frac{\partial K}{\partial \boldsymbol{\theta}_j} K^{-1} \mathbf{y} - \frac{1}{2} \text{tr}(K^{-1} \frac{\partial K}{\partial \boldsymbol{\theta}_j}) \\ &= 0 \end{aligned} \quad (5.22)$$

where $\theta_j \in \boldsymbol{\theta}$ is a parameter to be optimized. Equation (5.22) can be solved by the modified conjugate gradient (CG) method [154]. After obtaining these hyperparameters, a predictive assistance parameter can be calculated based on (5.18) and the exoskeleton can automatically provide the assistance without manipulating the human-machine interface.

5.2 Experiments

5.2.1 Robust Adaptive Oscillators

To test the performance of the RAO, a healthy subject (male, aged 28, weight 60 Kg, height 170 cm) walks on the ground with a hip exoskeleton in three modes: steady walking, speed up, and speed down. The input signal q is obtained by the difference of two hip encoders with $q = \theta_r - \theta_l$ and the tracking performances of the RAO and AO are shown in Figure 5.5. The real phase is obtained by offline linear interpolation and the zero phases are identified by the zero-cross point ($q = 0, \dot{q} > 0$). The real frequency is assumed to be constant among a stride, which is calculated by the reciprocal of time between two adjacent zero phases. It can be seen that the performances of the RAO and AO are similar in the steady walking. However, the RAO has a faster convergence rate and owns a more

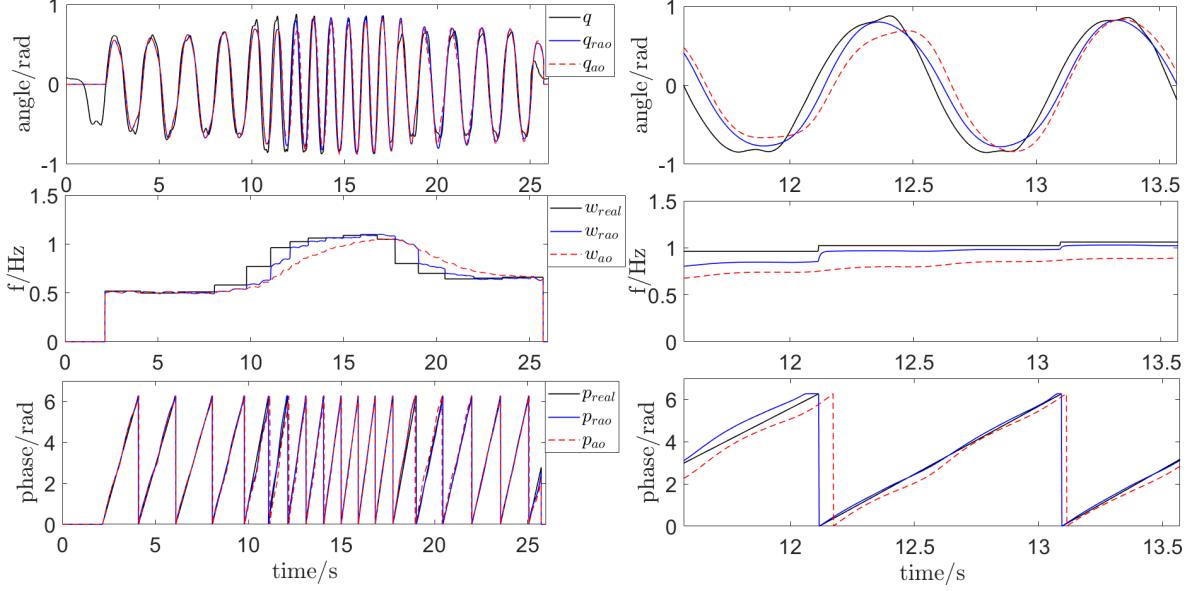


Figure 5.5: Performances comparison of the RAO and AO. The subject walks with a hip exoskeleton in three modes: steady walking, speed up, and speed down. The RAO and AO have similar performance with steady walking, while RAO's performance is better than AO with gait transition, such as speed up or speed down. The symbol q , w_{real} , p_{real} represents the input signal, the ground truth frequency, and the ground truth phase, respectively. The subscript rao and ao denotes the corresponding estimates from the RAO and AO, respectively. A close-up with a speed-up gait is shown on the right. It can be seen that the frequency estimation of the RAO converges to the real frequency quickly. Moreover, its phase is aligned with the zero phases automatically.

accurate phase estimation in the non-steady gait. A close-up on the right of the figure supports this statement.

The error performances of the RAO and AO are drawn in Figure 5.6, in which we can see that the tracking error e_{qrao} , frequency error e_{wrao} , and phase error e_{prao} in RAO are remarkably smaller than that of values in AO. The RMSEs of these three indices are calculated and summarized in Table 5.1. It can be seen that improvements of 20.46%, 36.82%, 34.28% are achieved by RAO compared with AO in terms of tracking error e_q , frequency error e_f , and phase error e_p .

Table 5.1: Performances of the RAO and AO.

Items	RAO	AO	Reduction
RMSE of e_q (rad)	0.1245	0.1565	20.46%
RMSE of e_f (Hz)	0.0779	0.1232	36.82%
RMSE of e_p (rad)	0.1566	0.2383	34.28%

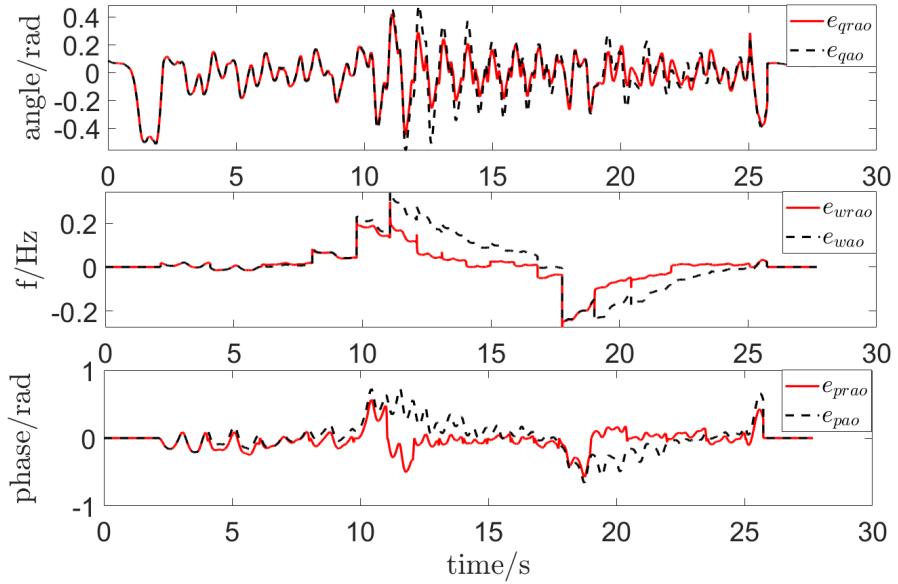


Figure 5.6: Error performances of the RAO and AO. Three indices including tracking error, frequency error, and phase error are involved in the figure. The red lines represent the performance of the RAO while the dashed black lines are the errors of the AO.

5.2.2 Torque Controller

As described in equation (5.12), the torque controller contains three parts: a NDOB term \tilde{d} , a proportional term $k_p(\tau_d - \tau_a)$ and a feedforward term $k_d \dot{\tau}_d$. Three experiments are conducted to verify the effectiveness of the NDOB term and the feedforward term. In the NDOB-KP-KD experiment, the torque controller is identical to equation (5.12). In the NDOB-KP experiment, the KD term is eliminated. In the KP-KD experiment, the NDOB term is excluded. In all experiments, the desired torque is the same and the subject walks on a treadmill at a speed of 3 km/h. The tracking performances of these three controllers are shown in Figure 5.7. It can be seen that the original controller performs the best while the third controller is the worst. This indicates that the NDOB term can significantly improve the tracking accuracy. By comparing the results of the first experiment with the second experiment, one can see that the tracking delay can be compensated effectively by the feedforward term. Notably, in all experiments, there is an obvious tracking error around the heel strike moment. This is caused by the slack of the belt in the reducer (the reducer is composed of gear and belt).

Two metrics, maximum error (ME), and root mean squared error (RMSE), are calculated and summarized in Table 5.2. It can be seen that the RMSE of the proposed controller is a mere 0.51 N·m, while it is 0.63 N·m and 1.42 N·m in the NDOB-KP controller

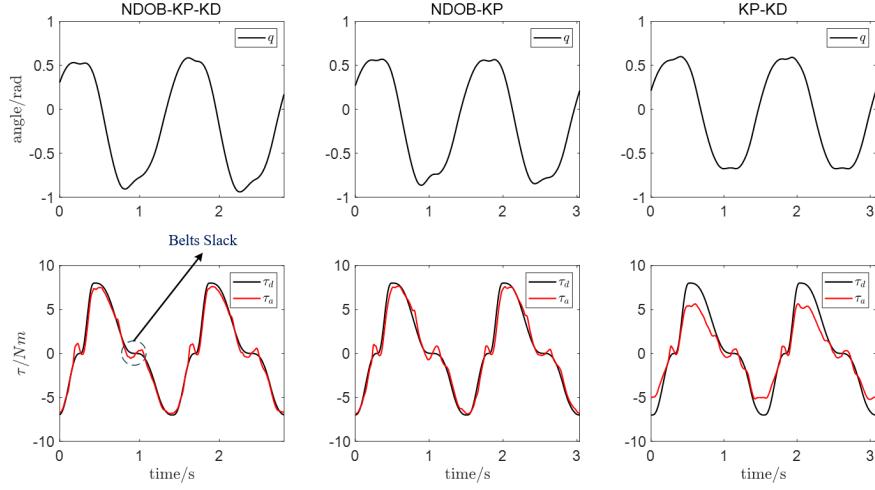


Figure 5.7: Performances of three torque controllers. The signals q are shown on the top while the corresponding torque tracking performances are shown at the bottom.

and the KP-KD controller. As for the ME value, it is 1.4 N·m in the original controller, while is as larger as 1.85 N·m and 2.83 N·m in the NDOB-KP controller and the KP-KD controller. These comparisons reveal that both the NDOB term and the feedforward term can improve the torque tracking performance.

Table 5.2: Performances of Different Torque Controllers.

Controller	ME (N·m)	RMSE (N·m)
NDOB-KP-KD	1.40	0.51
NDOB-KP	1.85	0.63
KP-KD	2.83	1.42

5.2.3 Parameter Map Learning

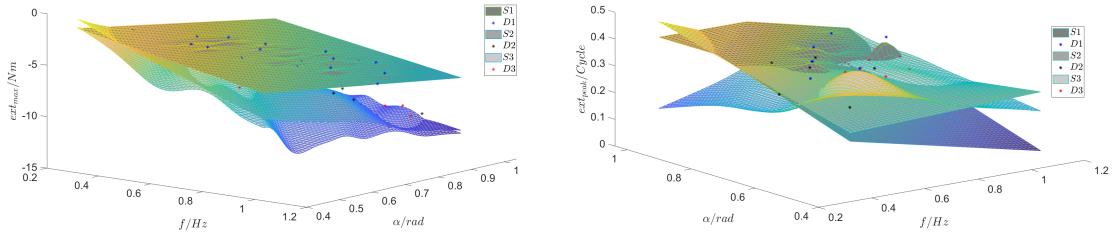
Seven subjects (aged 28.1 ± 4.1 ; weight 65.9 ± 10.9 kg ; height 171.6 ± 3.5 cm; mean \pm S.D.) participate in the experiment. The experiment contains two sessions: a warm-up session, and a test session. The warm-up session is arranged on the first day and participants are guided on how to manipulate the exoskeleton. In this session, the participants are required to select the offset variable and extension duration variable so that the extension torque appears at the heel strike moment and flexion torque appears at the toe-off moment. The test session is conducted on the following day. In this session, subjects are required to walk on the treadmill with a hip exoskeleton from 1.5 km/h to 5.0 km/h at an

increment of 0.5 km/h and select their preferred ext_{max} , ext_{peak} , fle_{max} and fle_{peak} parameters. At each speed, three gaits are required to be conducted which include a natural gait, a small amplitude gait, and a big amplitude gait. Subjects can adjust the assistance by an interactive screen and submit their preferred parameters using the following criteria: the exoskeleton can reduce their biological effort effectively; the interaction force is comfortable. The experimental setup and the interactive screen are shown in Figure 5.8. Notably, subjects can submit either one group of parameters or multiple groups of parameters in each gait. These data, together with the gait features, will be used for assistance map learning.

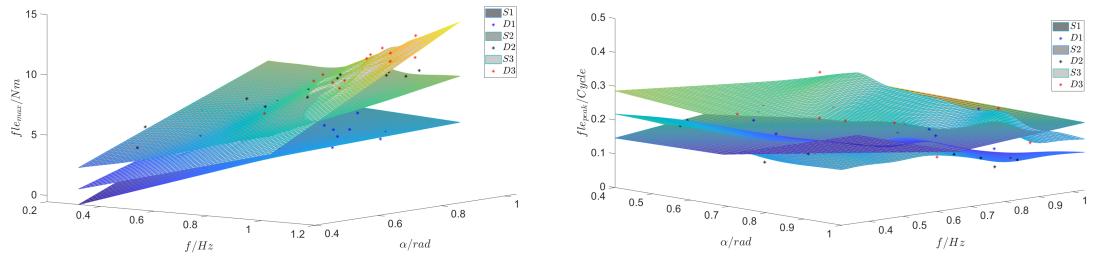


Figure 5.8: Experimental setup and the interactive screen. Six parameters are involved in the screen. Among them, the offset $offset$ and the extension duration ext_{dur} are determined in the warm-up session and remain unchanged in the following session, while the other parameters (ext_{max} , ext_{peak} , fle_{max} , fle_{peak}) are tuned in the training session.

The assistance map is constructed using equations (5.13)–(5.22) in MATLAB offline. After the learning procedure, the learned assistance maps are downloaded to the microprocessor of the exoskeleton. Then, the exoskeleton can provide adaptive individualized assistance when walking at different speeds. The maps of ext_{max} , ext_{peak} , fle_{max} , fle_{peak} for three subjects (S1, S2 and S3) are drawn in Figure 5.9. The maps for S7 are shown in Figure 5.10. From these figures, one can see that there is a substantial difference on assistance maps among subjects, which indicates that one subject's preferred assistance may be disliked by another. Generally, the values of maximum extension torque ext_{max} and maximum flexion torque fle_{max} increase with the growth of gait frequency f and amplitude α . As for the extension flexion peak time ext_{peak} and flexion peak time fle_{peak} , they are more irregular and are different from person to person. A common characteristic

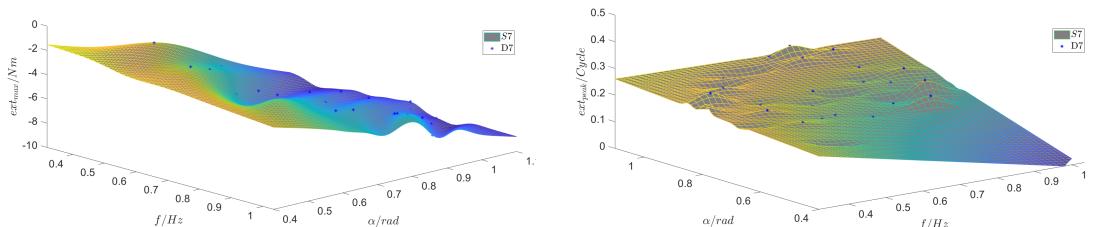


(a) Maximum extension torque maps of three subjects.
(b) Extension peak time maps of three subjects.



(c) Maximum flexion torque maps of three subjects.
(d) Flexion peak time maps of three subjects.

Figure 5.9: Learned assistance maps for S1, S2, and S3. The stars * represent the user-selected parameters, the predicted values are drawn by mesh figures.



(a) Maximum extension torque for S7.
(b) Extension peak time for S7.

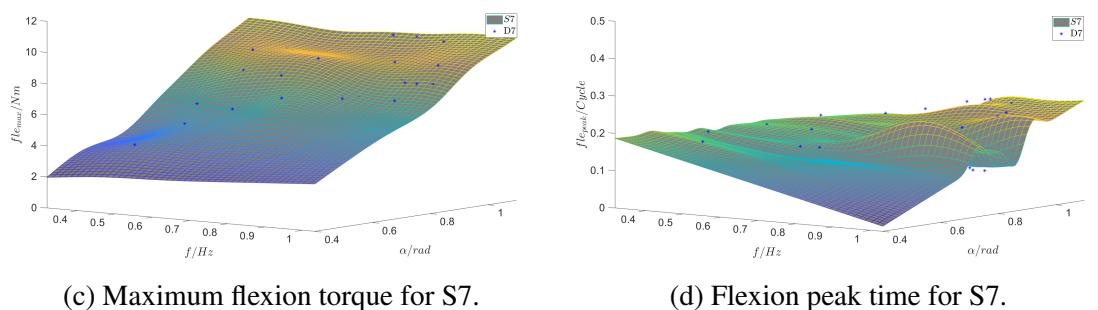


Figure 5.10: Learned assistance maps for S7. The star * represents the user-selected parameters, the predicted values is drawn by mesh figures.

of ext_{peak} and fle_{peak} is that they are statistically located around the midpoint of extension duration and flexion duration, revealing that subjects dislike a sharp assistance transition.

It is worth noting that the proposed method significantly improves the learning efficiency when compared with traditional HIL optimization. In our experiment, participants spend 20.1 ± 2.5 minutes to learn a total of 24 kinds of gaits, which is remarkably less than 83 ± 14 and 21.4 ± 1 minutes for a single gait in [29] and [30], respectively (see Table. 5.3). In addition, our proposed method does not require specific instruments and the training procedure can be conducted at any place (the training process does not necessarily need a treadmill). Hence it is more convenient compared with HIL optimization.

Table 5.3: Learning Efficiency Comparisons with Different Methods.

Methods	Parameters	Gaits	Time Cost
CMA-ES [29]	4	1	83 ± 14 min
Bayesian Optimization [30]	2	1	21.4 ± 1 min
Body-in-the-loop [127]	1	1	50 min
Proposed	4	24	20.1 ± 2.5 min

5.2.4 Investigation of Muscle Activities

To investigate the performance of the user-selected assistance profile, an experiment is conducted to measure muscle activities with sEMG sensors (Trigno, Delsys, USA). Three muscles are considered in this experiment, which include rectus femoris (RF), tibialis anterior (TA), and medial gastrocnemius (m.GAS). Seven participants participate in the experiments and the desired torque is generated based on the user's preference. The experimental scenario and the sensor placement are shown in Figure 5.11.

The experiment has been authorized by the HKUST Ethics Committee. The subjects are required to walk on the treadmill at a pace of 5 km/h in three different conditions: zero torque (ZT), assistance (ASS), and normal walking (NW). Each condition lasts 2 minutes to achieve a steady-state gait and a 2 minutes-break is given between two conditions to avoid fatigue. The sampling frequency of sEMG sensors is 2148 Hz. The raw signal is first butter filtered (cutoff frequency 20 Hz-450 Hz), then offset removed. The sEMG envelope is obtained by its root mean square (RMS) values and the corresponding window

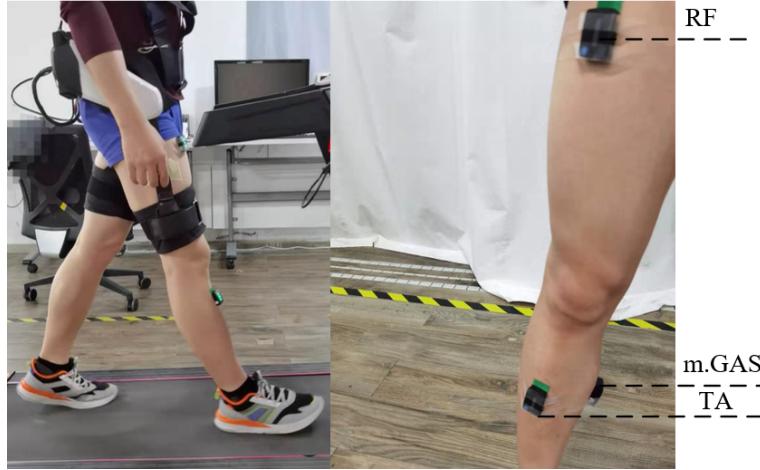


Figure 5.11: Experimental scenario and sEMG sensor placement. Three muscles are involved in this experiment, which are RF, TA, and m.GAS. The desired torque is generated based on the user's selection (or preference).

length is 0.125 s. The orientations of the thigh and shank are collected by inertial measurement units (IMUs) integrated into the sEMG sensors. The gait cycle is segmented by the maximum hip angle obtained by IMUs and the sEMG envelope is averaged and normalized to a stride. We use the last-minute data of each walking condition for calculation.

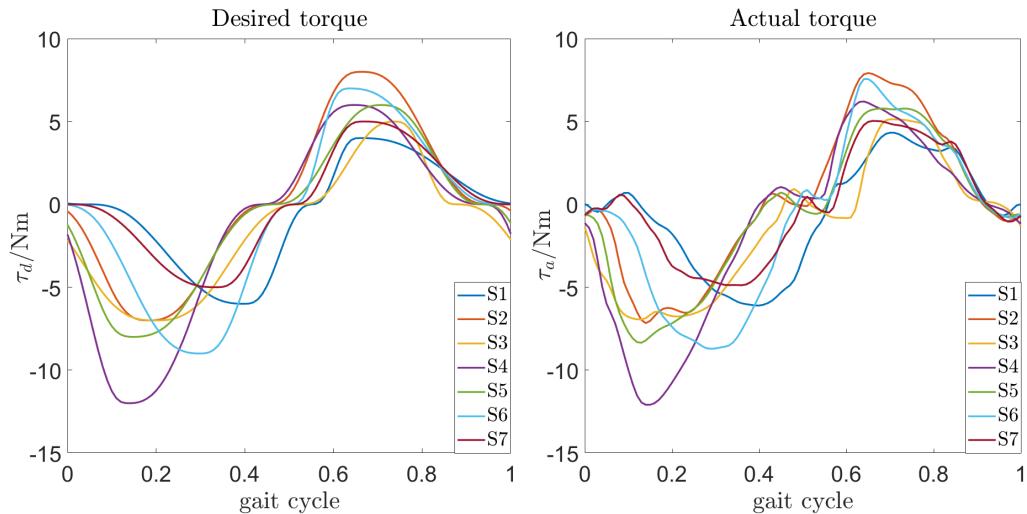
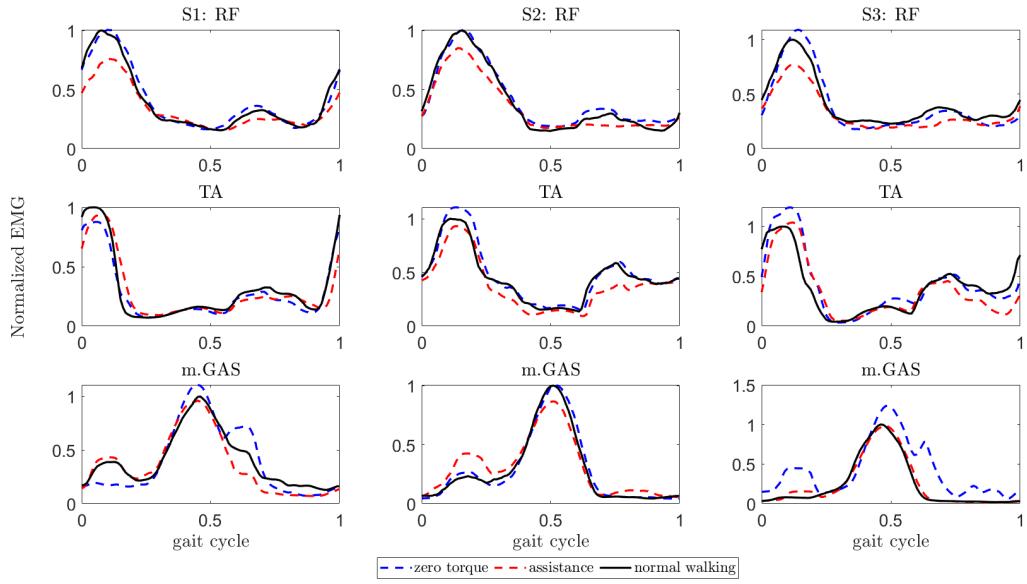
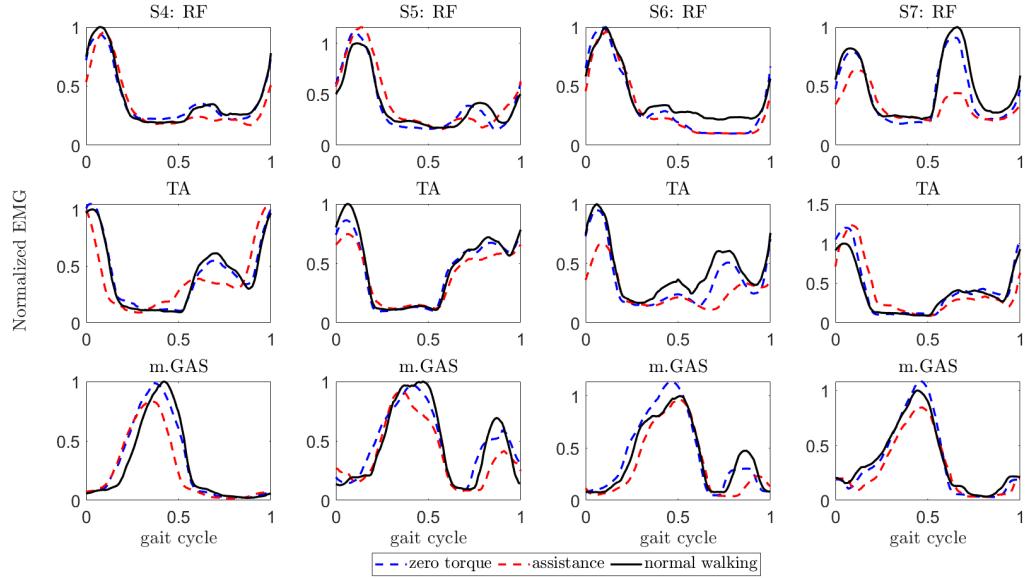


Figure 5.12: The desired torque and actual torque for 7 participants when walking at 5 km/h using the user-preferred parameters.



(a) sEMG signals of S1, S2, and S3.



(b) sEMG signals of S4, S5, S6, and S7.

Figure 5.13: Muscle activities of seven subjects in three conditions. The amplitudes are normalized to the data in the NW condition. Subjects walk on the treadmill at a speed of 5 km/h. The assistance profile is given by the user's manual selection (or preference).

The user-selected torque and the actual torque measured by a torque sensor are shown in Figure 5.12. The envelope muscle activities of seven subjects in different conditions are shown in Figure 5.13. The sEMG amplitudes are normalized to the averaged peak values of the data in the NW condition. It can be seen that there is an obvious muscle activity reduction in the ASS mode compared with ZT mode, which indicates that the user-selected assistance profile can reduce the biological efforts.

We use the RMS sEMG envelope values over a stride for the comparison of muscle activities among different walking modes. It can be seen that all muscle activities are reduced in the ASS mode compared to ZT mode or NW mode (see Figure 5.14). Meanwhile, m.GAS has a statistically significant difference in the ASS compared with ZT and NW, and the corresponding reductions are $15.63 \pm 6.51\%$ and $-8.73 \pm 6.4\%$. The detailed comparison was summarized in Table 5.4 where the superscript * denotes statistically significant differences with respect to ZT, while ** denotes statistically significant differences with respect to NW ($p < 0.05$).

Table 5.4: Muscle Activity Comparisons.

Muscle	ASS-ZT (<i>mean</i> \pm <i>S.D.</i> %)	ASS-NW (<i>mean</i> \pm <i>S.D.</i> %)
<i>RF</i>	$-12.70 \pm 12.59^*$	-13.49 ± 14.85
<i>TA</i>	$-11.40 \pm 10.16^*$	-11.24 ± 14.43
<i>m.GAS</i>	$-15.63 \pm 6.51^*$	$-8.73 \pm 6.40^{**}$

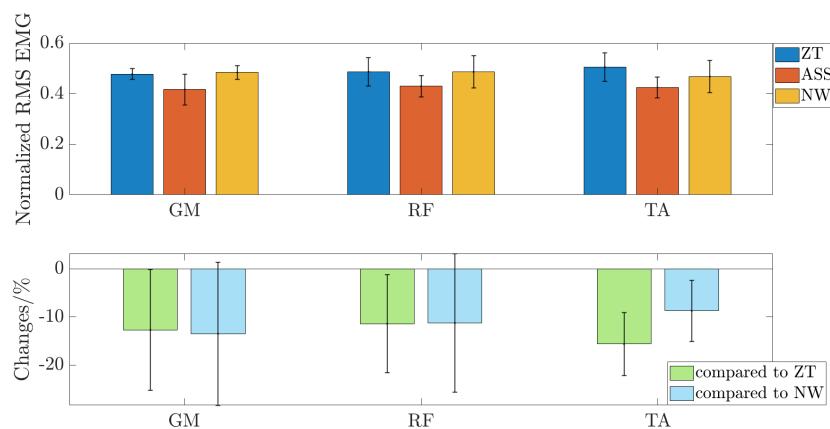


Figure 5.14: Normalized muscle activities and their changes in the assistance mode compared to ZT mode and NW mode.

5.3 Summary

We present a RAO for human-robot synchronization. We reveal that the proposed method has a faster convergence rate and smaller phase error compared with the conventional AO. Moreover, we develop a GPR strategy for assistance map construction, which is both convenient and efficient. An experiment is conducted to verify the performance of the user-selected profiles using sEMG. Results show that the user-selected assistance can achieve an obvious biological effort reduction compared with the zero torque mode.

Our experiment also reveals that the user-selected assistance map is highly personalized and is different from person to person. This indicates that unified assistance is not attractive for users since different subjects prefer different assistance profiles. On the contrary, our work provides a possible solution to integrate the users' preferences into the exoskeleton control.

Chapter 6

Conclusion

6.1 Concluding Remarks

In this thesis, we pave the way for walking assistance in a community environment using a lower-limb exoskeleton. We focus on the following three questions: 1) gait orientation estimation using IMUs; 2) disturbance rejection for an exoskeleton; 3) individualized assistance map construction. Specifically, we generalize the first two questions as state estimation with some channels contaminated by non-Gaussian noises. To address this issue, we extend the definition of correntropy from random variables to random vectors and define the MKC. Then, we provide the corresponding MKCL and demonstrate its feasibility as a robust cost. The MKMCKF and GMKMCKF are derived based on the MKCL under the Gaussian kernel and the GGD. Further, they are utilized for orientation estimation of IMUs and disturbance estimation of an exoskeleton. As for the individualized assistance map construction, we design it by employing the users' preferences together with the gait features. Experiments show that the proposed method can achieve a biological effort reduction when walking at 5 km/h.

In Chapter 2, we consider a state estimation problem with some channels containing non-Gaussian noises. To cope with this issue, we introduce the MKC where different kernel bandwidths are applied to different channels. Then, we use the corresponding MKCL as a cost function to replace the original MSE and derive a MKMCKF which can be regarded as a generalization of the KF. Extensive simulations verify the effectiveness of the proposed method.

In Chapter 3, we apply the MKMCKF to the orientation estimation of IMUs. For six-axis IMUs, to reduce the parameter number of the MKMCKF and decrease its complexity,

we derive a CMKMCKF where an infinite bandwidth is applied to all Gaussian channels and a unified bandwidth is applied to all non-Gaussian channels. Then, we apply this method to orientation estimation and construct the CMKMC-ESKF. We compare it with the GD and the traditional ESKF. Simulations and experiments verify the superiority of the proposed algorithm. For nine-axis IMUs, we derive a MKMCKF-OE and employ BO for kernel bandwidth selection. The proposed method is implemented in an embedded system and compared with the GD, ESKF, and IGD. Experiments using an optical motion capture system verify its efficacy.

In Chapter 4, we consider a disturbance estimation issue. To handle this problem, we first model the disturbance dynamics as a summation of the nominal model and a ϵ -contaminated mixture model. Then, we extend the MKC to the GMKC where the Gaussian kernel is replaced by the GGD. We provide some important properties of the GMKC and demonstrate its versatility and feasibility as a robust cost. Further, we derive a GMKMCKF based on the GMKC which is proved to be a generalization of the MKM-CKF. The proposed method is applied to the disturbance observer. Simulations using an exoskeleton verify the effectiveness of the proposed method.

In Chapter 5, we consider an assistance map construction problem with a hip exoskeleton. To maximize the user's comfort while simultaneously reducing the biological effort, we construct the assistance map based on the user's preference. Specifically, we design a RAO to extract gait features. Then, we record the user's preferred assistance parameters under different gait features. Finally, we construct the assistance map using the GPR. The effectiveness of the preference-based assistance is validated by a hip exoskeleton at a speed of 5 km/h with 7 participants. Three muscles which include rectus femoris, tibialis anterior, and medial gastrocnemius are investigated in three conditions: ASS,ZT, NW. Results show that all muscles achieve an activity reduction in the ASS mode compared with the ZT or NW. Meanwhile, there is a statistically significant difference on medial gastrocnemius in the ASS mode with respect to both the ZT and NW ($-15.63 \pm 6.51\%$ and $-8.73 \pm 6.40\%$, respectively).

6.2 Future Work

For the MKMCKF and GMKMCKF introduced in Chapter 2 and Chapter 4, their variants, such as the multi-kernel correntropy-based sequential Kalman filter, Chandrasekhar Kalman filter, extended Kalman filter, unscented Kalman filter, and distributed Kalman filter are needed to be derived with their characteristics to be explored. Furthermore, the smoother using the multi-kernel correntropy as a cost is worth investigating too. Moreover, the bandwidth selection is still demanding for the multi-kernel correntropy-based estimators. Therefore, it is meaningful to explore some adaptive kernel bandwidth methods.

For the orientation estimation methodology introduced in Chapter 3, although excellent performance has been achieved by the multi-kernel correntropy-based Kalman filter, it may be interesting to re-derive the gradient descend method for orientation estimation using the concept of the MKC. The derived algorithm should be robust to disturbances and own less computation burden compared with the KF-like algorithms, which allow a higher sampling frequency and can prolong the working hours of the battery when implemented on an embedded system.

For the assistance map construction introduced in Chapter 5, we only conduct experiments on a hip exoskeleton and investigate its biological effects by investigating muscle activities. In the future, we may conduct experiments on a hip-knee exoskeleton and investigate both muscle activities and metabolic consumption. Further, a higher-dimensional preference-based assistance map may be considered so as to systematically investigate the relationship between the preferred assistance and the gait feature.

References

- [1] U. N. D. of Economic and S. Affairs, *World Population Ageing 2019*. United Nations, 2020.
- [2] M. Y. Osoba, A. K. Rao, S. K. Agrawal, and A. K. Lalwani, “Balance and gait in the elderly: A contemporary review,” *Laryngoscope Investigative Otolaryngology*, vol. 4, no. 1, pp. 143–153, 2019.
- [3] D. Malatesta, D. Simar, Y. Dauvilliers, R. Candau, F. Borrani, C. Préfaut, and C. Caillaud, “Energy cost of walking and gait instability in healthy 65-and 80-year-olds,” *Journal of Applied Physiology*, vol. 95, no. 6, pp. 2248–2256, 2003.
- [4] C. O. Johnson, M. Nguyen, G. A. Roth, E. Nichols, T. Alam, D. Abate, F. Abd-Allah, A. Abdelalim, H. N. Abraha, and N. M. Abu-Rmeileh, “Global, regional, and national burden of stroke, 1990-2016: a systematic analysis for the global burden of disease study 2016,” *The Lancet Neurology*, vol. 18, no. 5, pp. 439–458, 2019.
- [5] T. Zhang, J. Zhao, X. Li, Y. Bai, B. Wang, Y. Qu, B. Li, and S. Zhao, “Chinese stroke association guidelines for clinical management of cerebrovascular disorders: executive summary and 2019 update of clinical management of stroke rehabilitation,” *Stroke and Vascular Neurology*, 2020.
- [6] S. J. Olney and C. Richards, “Hemiparetic gait following stroke. Part I: Characteristics,” *Gait & Posture*, vol. 4, no. 2, pp. 136–148, 1996.
- [7] C. L. Richards, F. Malouin, and C. Dean, “Gait in stroke: assessment and rehabilitation,” *Clinics in Geriatric Medicine*, vol. 15, no. 4, pp. 833–856, 1999.

- [8] S. A. Combs, E. L. Dugan, E. N. Ozimek, and A. B. Curtis, “Effects of body-weight supported treadmill training on kinetic symmetry in persons with chronic stroke,” *Clinical Biomechanics*, vol. 27, no. 9, pp. 887–892, 2012.
- [9] A. Hall, M. Bowden, S. Kautz, and R. Neptune, “Biomechanical variables related to walking performance 6-months following post-stroke rehabilitation,” *Clinical Biomechanics*, vol. 27, no. 10, pp. 1017–1022, 2012.
- [10] M. G. Bowden, A. L. Behrman, R. R. Neptune, C. M. Gregory, and S. A. Kautz, “Locomotor rehabilitation of individuals with chronic stroke: difference between responders and nonresponders,” *Archives of Physical Medicine and Rehabilitation*, vol. 94, no. 5, pp. 856–862, 2013.
- [11] C. English, P. J. Manns, C. Tucak, and J. Bernhardt, “Physical activity and sedentary behaviors in people with stroke living in the community: a systematic review,” *Physical Therapy*, vol. 94, no. 2, pp. 185–196, 2014.
- [12] K. M. Michael, J. K. Allen, and R. F. Macko, “Reduced ambulatory activity after stroke: the role of balance, gait, and cardiovascular fitness,” *Archives of Physical Medicine and Rehabilitation*, vol. 86, no. 8, pp. 1552–1556, 2005.
- [13] L. N. Awad, J. Bae, K. Odonnell, S. M. De Rossi, K. Hendron, Sloot *et al.*, “A soft robotic exosuit improves walking in patients after stroke,” *Science Translational Medicine*, vol. 9, no. 400, pp. 9084–9096, 2017.
- [14] P. H. Lee, H. Nan, Y.-Y. Yu, I. McDowell, G. M. Leung, and T. Lam, “For non-exercising people, the number of steps walked is more strongly associated with health than time spent walking,” *Journal of Science and Medicine in Sport*, vol. 16, no. 3, pp. 227–230, 2013.
- [15] D. Rand, J. J. Eng, P.-F. Tang, C. Hung, and J.-S. Jeng, “Daily physical activity and its contribution to the health-related quality of life of ambulatory individuals with chronic stroke,” *Health and Quality of Life Outcomes*, vol. 8, no. 1, pp. 1–8, 2010.
- [16] W. Zhao, S. Ukawa, T. Kawamura, K. Wakai, M. Ando, K. Tsushita, and A. Tamakoshi, “Health benefits of daily walking on mortality among younger-elderly men with or without major critical diseases in the new integrated suburban

- seniority investigation project: a prospective cohort study,” *Journal of Epidemiology*, vol. 25, no. 10, pp. 609–616, 2015.
- [17] L. DiPietro, “Physical activity in aging: changes in patterns and their relationship to health and function,” *The Journals of Gerontology Series A: Biological Sciences and Medical Sciences*, vol. 56, no. 2, pp. 13–22, 2001.
- [18] A. Frizera-Neto, R. Ceres, E. Rocon, and J. L. Pons, “Empowering and assisting natural human mobility: The symbiosis walker,” *International Journal of Advanced Robotic Systems*, vol. 8, no. 3, pp. 29–46, 2011.
- [19] A. Duschau-Wicke, A. Caprez, and R. Riener, “Patient-cooperative control increases active participation of individuals with SCI during robot-aided gait training,” *Journal of NeuroEngineering and Rehabilitation*, vol. 7, no. 1, pp. 43–56, 2010.
- [20] J. F. Veneman, R. Kruidhof, E. E. G. Hekman, R. Ekkelenkamp, E. H. F. Van Asseldonk, and H. van der Kooij, “Design and evaluation of the lopes exoskeleton robot for interactive gait rehabilitation,” *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 15, no. 3, pp. 379–386, 2007.
- [21] P. Stegall, K. Winfree, D. Zanotto, and S. K. Agrawal, “Rehabilitation exoskeleton design: Exploring the effect of the anterior lunge degree of freedom,” *IEEE Transactions on Robotics*, vol. 29, no. 4, pp. 838–846, 2013.
- [22] A. Martnez, B. Lawson, and M. Goldfarb, “A controller for guiding leg movement during overground walking with a lower limb exoskeleton,” *IEEE Transactions on Robotics*, vol. 34, no. 1, pp. 183–193, 2018.
- [23] H. Kawamoto, T. Hayashi, T. Sakurai, K. Eguchi, and Y. Sankai, “Development of single leg version of HAL for hemiplegia,” in *2009 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 2009, pp. 5038–5043.
- [24] D. Wei, Z. Li, Q. Wei, H. Su, B. Song, W. He, and J. Li, “Human-in-the-loop control strategy of unilateral exoskeleton robots for gait rehabilitation,” *IEEE Transactions on Cognitive and Developmental Systems*, vol. 13, no. 1, pp. 57–66, 2021.

- [25] H. Vallery, J. Veneman, E. Van Asseldonk, R. Ekkelenkamp, M. Buss, and H. Van Der Kooij, “Compliant actuation of rehabilitation robots,” *IEEE Robotics & Automation Magazine*, vol. 15, no. 3, pp. 60–69, 2008.
- [26] A. Martinez, B. Lawson, and M. Goldfarb, “A controller for guiding leg movement during overground walking with a lower limb exoskeleton,” *IEEE Transactions on Robotics*, vol. 34, no. 1, pp. 183–193, 2017.
- [27] A. Martinez, B. Lawson, C. Durrough, and M. Goldfarb, “A velocity-field-based controller for assisting leg movement during walking with a bilateral hip and knee lower limb exoskeleton,” *IEEE Transactions on Robotics*, vol. 35, no. 2, pp. 307–316, 2018.
- [28] S. Hussain, P. K. Jamwal, M. H. Ghayesh, and S. Q. Xie, “Assist-as-needed control of an intrinsically compliant robotic gait training orthosis,” *IEEE Transactions on Industrial Electronics*, vol. 64, no. 2, pp. 1675–1685, 2016.
- [29] J. Zhang, P. Fiers, K. A. Witte, R. W. Jackson, K. L. Poggensee, C. G. Atkeson, and S. H. Collins, “Human-in-the-loop optimization of exoskeleton assistance during walking,” *Science*, vol. 356, no. 6344, pp. 1280–1284, 2017.
- [30] Y. Ding, M. Kim, S. Kuindersma, and C. J. Walsh, “Human-in-the-loop optimization of hip assistance with a soft exosuit during walking,” *Science Robotics*, vol. 3, no. 15, pp. 5438–5446, 2018.
- [31] W. Wang, J. Chen, Y. Ji, W. Jin, J. Liu, and J. Zhang, “Evaluation of lower leg muscle activities during human walking assisted by an ankle exoskeleton,” *IEEE Transactions on Industrial Informatics*, vol. 16, no. 11, pp. 7168–7176, 2020.
- [32] S. H. Collins, M. B. Wiggin, and G. Sawicki, “Reducing the energy cost of human walking using an unpowered exoskeleton,” *Nature*, vol. 522, pp. 212–215, 2015.
- [33] R. Nasiri, A. Ahmadi, and M. N. Ahmadabadi, “Reducing the energy cost of human running using an unpowered exoskeleton,” *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 26, no. 10, pp. 2026–2032, 2018.
- [34] J. Bae, C. Siviy, M. Rouleau, N. Menard, K. O’Donnell, I. Geliana, M. Athanasius, D. Ryan, C. Bibeau, L. Sloot, P. Kudzia, T. Ellis, L. Awad, and C. J. Walsh,

“A lightweight and efficient portable soft exosuit for paretic ankle assistance in walking after stroke,” in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, 2018, pp. 2820–2827.

- [35] B. T. Quinlivan, S. Lee, P. Malcolm, D. M. Rossi, M. Grimmer, C. Siviy, N. Karavas, D. Wagner, A. Asbeck, I. Galiana *et al.*, “Assistance magnitude versus metabolic cost reductions for a tethered multiarticular soft exosuit,” *Science Robotics*, vol. 2, no. 2, pp. 4416–4426, 2017.
- [36] J. Kim, G. Lee, R. Heimgartner, D. Arumukhom Revi, N. Karavas, D. Nathanson, I. Galiana, A. Eckert-Erdheim, P. Murphy, D. Perry *et al.*, “Reducing the metabolic rate of walking and running with a versatile, portable exosuit,” *Science*, vol. 365, no. 6454, pp. 668–672, 2019.
- [37] C. Buesing, G. Fisch, M. O'Donnell, I. Shahidi, L. Thomas, C. K. Mummidisetti, K. J. Williams, H. Takahashi, W. Z. Rymer, and A. Jayaraman, “Effects of a wearable exoskeleton stride management assist system (SMA®) on spatiotemporal gait characteristics in individuals after stroke: a randomized controlled trial,” *Journal of Neuroengineering and Rehabilitation*, vol. 12, no. 1, pp. 1–14, 2015.
- [38] M. Bortole, A. Venkatakrishnan, F. Zhu, J. C. Moreno, G. E. Francisco, J. L. Pons, and J. L. Contreras-Vidal, “The H2 robotic exoskeleton for gait rehabilitation after stroke: early findings from a clinical study,” *Journal of Neuroengineering and Rehabilitation*, vol. 12, no. 1, pp. 1–14, 2015.
- [39] S. Hirano, H. Kagaya, E. Saitoh, S. Sonoda, S. Tanabe, M. Katoh, J. Yamada, G. Tanino, A. Suzuki, and N. Itoh, “Effectiveness of gait exercise assist robot (GEAR) for stroke patients with hemiplegia,” *Japanese Journal of Comprehensive Rehabilitation Science*, vol. 8, pp. 71–76, 2017.
- [40] N. Mizukami, S. Takeuchi, M. Tetsuya, A. Tsukahara, K. Yoshida, A. Matsushima, Y. Maruyama, K. Tako, and M. Hashimoto, “Effect of the synchronization-based control of a wearable robot having a non-exoskeletal structure on the hemiplegic gait of stroke patients,” *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 26, no. 5, pp. 1011–1016, 2018.

- [41] K. Seo, K. Kim, Y. J. Park, J.-K. Cho, J. Lee, B. Choi, B. Lim, Y. Lee, and Y. Shim, “Adaptive oscillator-based control for active lower-limb exoskeleton and its metabolic impact,” in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, 2018, pp. 6752–6758.
- [42] Keehong Seo, SeungYong Hyung, Byung Kwon Choi, Younbaek Lee, and Youngbo Shim, “A new adaptive frequency oscillator for gait assistance,” in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 5565–5571.
- [43] E. Zheng, S. Manca, T. Yan, A. Parri, N. Vitiello, and Q. Wang, “Gait phase estimation based on noncontact capacitive sensing and adaptive oscillators,” *IEEE Transactions on Biomedical Engineering*, vol. 64, no. 10, pp. 2419–2430, 2017.
- [44] Tao Xue, Ziwei Wang, Tao Zhang, Ou Bai, Meng Zhang, and Bin Han, “A new delayless adaptive oscillator for gait assistance,” in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020, pp. 3459–3464.
- [45] Y. Ding, I. Galiana, C. Siviy, F. A. Panizzolo, and C. Walsh, “IMU-based iterative control for hip extension assistance with a soft exosuit,” in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, 2016, pp. 3501–3508.
- [46] L. Wang, Y. Sun, Q. Li, T. Liu, and J. Yi, “Two shank-mounted IMUs-based gait analysis and classification for neurological disease patients,” *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 1970–1976, 2020.
- [47] V. Joukov, M. Karg, and D. Kulic, “Online tracking of the lower body joint angles using imus for gait rehabilitation,” in *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 2014, pp. 2310–2313.
- [48] S. O. H. Madgwick, A. J. L. Harrison, and R. Vaidyanathan, “Estimation of IMU and MARG orientation using a gradient descent algorithm,” in *IEEE International Conference on Rehabilitation Robotics*, 2011, pp. 1–7.
- [49] L. Xiang and L. Qun, “External acceleration elimination for complementary attitude filter,” in *IEEE International Conference on Information and Automation (ICIA)*, 2017, pp. 208–212.

- [50] T. Seel and S. Ruppin, “Eliminating the effect of magnetic disturbances on the inclination estimates of inertial sensors,” *IFAC*, vol. 50, no. 1, pp. 8798–8803, 2017, 20th IFAC World Congress.
- [51] D. Roetenberg, H. Luinge, C. Baten, and P. Veltink, “Compensation of magnetic disturbances improves inertial and magnetic sensing of human body segment orientation,” *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 13, no. 3, pp. 395–405, 2005.
- [52] W.-H. Chen, D. J. Ballance, P. J. Gawthrop, and J. O'Reilly, “A nonlinear disturbance observer for robotic manipulators,” *IEEE Transactions on Industrial Electronics*, vol. 47, no. 4, pp. 932–938, 2000.
- [53] S. Komada, N. Machii, and T. Hori, “Control of redundant manipulators considering order of disturbance observer,” *IEEE Transactions on Industrial Electronics*, vol. 47, no. 2, pp. 413–420, 2000.
- [54] J. Ishikawa and M. Tomizuka, “A novel add-on compensator for cancellation of pivot nonlinearities in hard disk drives,” *IEEE Transactions on Magnetics*, vol. 34, no. 4, pp. 1895–1897, 1998.
- [55] S. H. Collins, M. B. Wiggin, and G. S. Sawicki, “Reducing the energy cost of human walking using an unpowered exoskeleton,” *Nature*, vol. 522, no. 7555, pp. 212–215, 2015.
- [56] R. Nasiri, A. Ahmadi, and M. N. Ahmadabadi, “Reducing the energy cost of human running using an unpowered exoskeleton,” *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 26, no. 10, pp. 2026–2032, 2018.
- [57] K. Seo, J. Lee, Y. Lee, T. Ha, and Y. Shim, “Fully autonomous hip exoskeleton saves metabolic cost of walking,” in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, 2016, pp. 4628–4635.
- [58] R. Nuckols, S. Lee, K. Swaminathan, D. Orzel, R. Howe, and C. Walsh, “Individualization of exosuit assistance based on measured muscle dynamics during versatile walking,” *Science Robotics*, vol. 6, no. 60, pp. 1362–1374, 2021.

- [59] G. Ligorio and A. M. Sabatini, “A novel Kalman filter for human motion tracking with an inertial-based dynamic inclinometer,” *IEEE Transactions on Biomedical Engineering*, vol. 62, no. 8, pp. 2033–2043, 2015.
- [60] A. Assa and F. Janabi-Sharifi, “A Kalman filter-based framework for enhanced sensor fusion,” *IEEE Sensors Journal*, vol. 15, no. 6, pp. 3281–3292, 2015.
- [61] M. Wu and A. W. Smyth, “Application of the unscented Kalman filter for real-time nonlinear structural system identification,” *Structural Control and Health Monitoring*, vol. 14, no. 7, pp. 971–990, 2007.
- [62] J. M. Kanieski, R. Cardoso, H. Pinheiro, and H. A. Gründling, “Kalman filter-based control system for power quality conditioning devices,” *IEEE Transactions on Industrial Electronics*, vol. 60, no. 11, pp. 5214–5227, 2012.
- [63] O. Ondel, E. Boutleux, E. Blanco, and G. Clerc, “Coupling pattern recognition with state estimation using Kalman filter for fault diagnosis,” *IEEE Transactions on Industrial Electronics*, vol. 59, no. 11, pp. 4293–4300, 2011.
- [64] M. Kok, J. D. Hol, and T. B. Schön, “Using inertial sensors for position and orientation estimation,” *arXiv preprint arXiv:1704.06053*, 2017.
- [65] A. M. Sabatini, “Quaternion-based extended Kalman filter for determining orientation by inertial and magnetic sensing,” *IEEE Transactions on Biomedical Engineering*, vol. 53, no. 7, pp. 1346–1356, 2006.
- [66] P. Zhang, J. Gu, E. E. Milios, and P. Huynh, “Navigation with IMU/GPS/digital compass with unscented Kalman filter,” in *IEEE International Conference Mechatronics and Automation, 2005*, vol. 3, 2005, pp. 1497–1502.
- [67] P. Bernal-Polo and H. Martínez-Barberá, “Kalman filtering for attitude estimation with quaternions and concepts from manifold theory,” *Sensors*, vol. 19, no. 1, p. 149, 2019.
- [68] C. Mitsantisuk, K. Ohishi, S. Urushihara, and S. Katsura, “Kalman filter-based disturbance observer and its applications to sensorless force control,” *Advanced Robotics*, vol. 25, no. 3-4, pp. 335–353, 2011.

- [69] T. T. Phuong, C. Mitsantisuk, and K. Ohishi, “High performance force sensing based on Kalman-filter-based disturbance observer utilizing fpga,” *IEEJ Transactions on Industry Applications*, vol. 131, no. 3, pp. 334–342, 2011.
- [70] F. Wang, D. Tian, and Y. Wang, “High accuracy inertial stabilization via Kalman filter based disturbance observer,” in *2016 IEEE International Conference on Mechatronics and Automation*, 2016, pp. 794–802.
- [71] L. Chang, K. Li, and B. Hu, “Hubers M-estimation-based process uncertainty robust filter for integrated INS/GPS,” *IEEE Sensors Journal*, vol. 15, no. 6, pp. 3367–3374, 2015.
- [72] L. Chang, B. Hu, G. Chang, and A. Li, “Huber-based novel robust unscented Kalman filter,” *IET Science, Measurement & Technology*, vol. 6, no. 6, pp. 502–509, 2012.
- [73] Y. Huang, Y. Zhang, N. Li, Z. Wu, and J. A. Chambers, “A novel robust Student’s t -based Kalman filter,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 53, no. 3, pp. 1545–1554, 2017.
- [74] Y. Huang, Y. Zhang, N. Li, and J. Chambers, “Robust students t based nonlinear filter and smoother,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 52, no. 5, pp. 2586–2596, 2016.
- [75] C. Masreliez and R. Martin, “Robust Bayesian estimation for the linear model and robustifying the Kalman filter,” *IEEE Transactions on Automatic Control*, vol. 22, no. 3, pp. 361–371, 1977.
- [76] M. Zorzi, “Robust Kalman filtering under model perturbations,” *IEEE Transactions on Automatic Control*, vol. 62, no. 6, pp. 2902–2907, 2016.
- [77] J. C. Principe, *Information Theoretic Learning: Renyi’s Entropy and Kernel Perspectives*. Springer Science & Business Media, 2010.
- [78] B. Chen, Y. Zhu, J. Hu, and J. C. Principe, *System Parameter Identification: Information Criteria and Algorithms*. Newnes, 2013.

- [79] B. Chen, X. Liu, H. Zhao, and J. C. Principe, “Maximum correntropy Kalman filter,” *Automatica*, vol. 76, pp. 70–77, 2017.
- [80] B. Chen, X. Wang, N. Lu, S. Wang, J. Cao, and J. Qin, “Mixture correntropy for robust learning,” *Pattern Recognition*, vol. 79, pp. 318–327, 2018.
- [81] A. Singh and J. C. Principe, “Using correntropy as a cost function in linear adaptive filters,” in *2009 International Joint Conference on Neural Networks*, 2009, pp. 2950–2955.
- [82] X. Liu, Z. Ren, H. Lyu, Z. Jiang, P. Ren, and B. Chen, “Linear and nonlinear regression-based maximum correntropy extended Kalman filtering,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 51, no. 5, pp. 3093–3102, 2019.
- [83] X. Liu, H. Qu, J. Zhao, and P. Yue, “Maximum correntropy square-root cubature Kalman filter with application to SINS/GPS integrated systems,” *ISA transactions*, vol. 80, pp. 195–202, 2018.
- [84] W. Liu, P. P. Pokharel, and J. C. Principe, “Correntropy: Properties and applications in non-Gaussian signal processing,” *IEEE Transactions on signal processing*, vol. 55, no. 11, pp. 5286–5298, 2007.
- [85] R. Izanloo, S. A. Fakoorian, H. S. Yazdi, and D. Simon, “Kalman filtering based on the maximum correntropy criterion in the presence of non-Gaussian noise,” in *2016 Annual Conference on Information Science and Systems (CISS)*, 2016, pp. 500–505.
- [86] B. Hou, Z. He, X. Zhou, H. Zhou, D. Li, and J. Wang, “Maximum correntropy criterion Kalman filter for alpha-jerk tracking model with non-Gaussian noise,” *Entropy*, vol. 19, no. 12, 2017.
- [87] M. Kulikova, “Sequential maximum correntropy Kalman filtering,” *Asian Journal of Control*, vol. 22, no. 1, pp. 25–33, 2020.
- [88] M. Kulikova, “Chandrasekhar-based maximum correntropy Kalman filtering with the adaptive kernel size selection,” *IEEE Transactions on Automatic Control*, vol. 65, no. 2, pp. 741–748, 2020.

- [89] M. Kulikova, “Square-root algorithms for maximum correntropy estimation of linear discrete-time systems in presence of non-Gaussian noise,” *Systems & Control Letters*, vol. 108, pp. 8–15, 2017.
- [90] X. Liu, H. Qu, J. Zhao, and B. Chen, “Extended Kalman filter under maximum correntropy criterion,” in *2016 International Joint Conference on Neural Networks (IJCNN)*, 2016, pp. 1733–1737.
- [91] X. Liu, B. Chen, B. Xu, Z. Wu, and P. Honeine, “Maximum correntropy unscented filter,” *International Journal of Systems Science*, vol. 48, no. 8, pp. 1607–1615, 2017.
- [92] W. Qin, X. Wang, and N. Cui, “Maximum correntropy sparse Gauss–Hermite quadrature filter and its application in tracking ballistic missile,” *IET Radar, Sonar & Navigation*, vol. 11, no. 9, pp. 1388–1396, 2017.
- [93] X. Liu, B. Chen, H. Zhao, J. Qin, and J. Cao, “Maximum correntropy Kalman filter with state constraints,” *IEEE Access*, vol. 5, pp. 25 846–25 853, 2017.
- [94] S. Fakoorian, M. Moosavi, R. Izanloo, V. Azimi, and D. Simon, “Maximum correntropy criterion constrained Kalman filter,” in *Dynamic Systems and Control Conference*, 2017.
- [95] G. Wang, R. Xue, and J. Wang, “A distributed maximum correntropy Kalman filter,” *Signal Processing*, vol. 160, pp. 247–251, 2019.
- [96] X. Fan, G. Wang, J. Han, and Y. Wang, “Interacting multiple model based on maximum correntropy Kalman filter,” *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 68, no. 8, pp. 3017–3021, 2021.
- [97] J. Wendel, O. Meister, C. Schlaile, and G. F. Trommer, “An integrated GPS/MEMS-IMU navigation system for an autonomous helicopter,” *Aerospace Science and Technology*, vol. 10, no. 6, pp. 527–533, 2006.
- [98] W. Huo, S. Mohammed, Y. Amirat, and K. Kong, “Fast gait mode detection and assistive torque control of an exoskeletal robotic orthosis for walking assistance,” *IEEE Transactions on Robotics*, vol. 34, no. 4, pp. 1035–1052, 2018.

- [99] M. Hao, K. Chen, and C. Fu, “Smoothen-based 3-D foot trajectory estimation using inertial sensors,” *IEEE Transactions on Biomedical Engineering*, vol. 66, no. 12, pp. 3534–3542, 2019.
- [100] I. Prayudi and D. Kim, “Design and implementation of IMU-based human arm motion capture system,” in *IEEE International Conference on Mechatronics and Automation*, 2012, pp. 670–675.
- [101] F. Wittmann, O. Lambery, R. R. Gonzenbach, M. A. van Raai, R. Hver, J. Held, M. L. Starkey, A. Curt, A. Luft, and R. Gassert, “Assessment-driven arm therapy at home using an IMU-based virtual reality system,” in *IEEE International Conference on Rehabilitation Robotics (ICORR)*, 2015, pp. 707–712.
- [102] R. Mahony, T. Hamel, and J.-M. Pflimlin, “Nonlinear complementary filters on the special orthogonal group,” *IEEE Transactions on Automatic Control*, vol. 53, no. 5, pp. 1203–1218, 2008.
- [103] K. Kanjanapas, Y. Wang, W. Zhang, L. Whittingham, and M. Tomizuka, “A human motion capture system based on inertial sensing and a complementary filter,” in *ASME 2013 Dynamic Systems and Control Conference*, 2013.
- [104] M. Admiraal, S. Wilson, and R. Vaidyanathan, “Improved formulation of the IMU and MARG orientation gradient descent algorithm for motion tracking in human-machine interfaces,” in *IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, 2017, pp. 403–410.
- [105] A. El Hadri and A. Benallegue, “Sliding mode observer to estimate both the attitude and the gyro-bias by using low-cost sensors,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2009, pp. 2867–2872.
- [106] M. S. Mark Pedley and Z. Baranski, “Freescale sensor fusion Kalman filter,” pp. 24–26, 2014. [Online]. Available: <https://github.com/memsindustrygroup/Open-Source-Sensor-Fusion/tree/master/docs>
- [107] C.-K. Lin, “Mixed H₂/H_∞ autopilot design of bank-to-turn missiles using fuzzy basis function networks,” *Fuzzy Sets and Systems*, vol. 158, no. 20, pp. 2268–2287, 2007.

- [108] H. Yoon and B. N. Agrawal, “Adaptive control of uncertain hamiltonian multi-input multi-output systems: With application to spacecraft control,” *IEEE Transactions on Control Systems Technology*, vol. 17, no. 4, pp. 900–906, 2009.
- [109] S. Li, J. Yang, W.-H. Chen, and X. Chen, *Disturbance Observer-based Control: Methods and Applications*. CRC press, 2014.
- [110] K. Ohishi, M. Nakao, K. Ohnishi, and K. Miyachi, “Microprocessor-controlled dc motor for load-insensitive position servo system,” *IEEE Transactions on Industrial Electronics*, no. 1, pp. 44–49, 1987.
- [111] Y. Huang and W. Xue, “Active disturbance rejection control: Methodology and theoretical analysis,” *ISA Transactions*, vol. 53, no. 4, pp. 963–976, 2014.
- [112] J. Han, “From PID to active disturbance rejection control,” *IEEE Transactions on Industrial Electronics*, vol. 56, no. 3, pp. 900–906, 2009.
- [113] C. Johnson, “Further study of the linear regulator with disturbances—the case of vector disturbances satisfying a linear differential equation,” *IEEE Transactions on Automatic Control*, vol. 15, no. 2, pp. 222–228, 1970.
- [114] Q.-C. Zhong, A. Kuperman, and R. Stobart, “Design of UDE-based controllers from their two-degree-of-freedom nature,” *International Journal of Robust and Nonlinear Control*, vol. 21, no. 17, pp. 1994–2008, 2011.
- [115] J.-H. She, X. Xin, and Y. Pan, “Equivalent-input-disturbance approachanalysis and application to disturbance rejection in dual-stage feed drive control system,” *IEEE/ASME Transactions on Mechatronics*, vol. 16, no. 2, pp. 330–340, 2010.
- [116] B. Chen, L. Xing, H. Zhao, N. Zheng, J. C. Pri *et al.*, “Generalized correntropy for robust adaptive filtering,” *IEEE Transactions on Signal Processing*, vol. 64, no. 13, pp. 3376–3387, 2016.
- [117] Y. Zhu, H. Zhao, X. Zeng, and B. Chen, “Robust generalized maximum correntropy criterion algorithms for active noise control,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 1282–1292, 2020.

- [118] S. Hakimi and G. Abed Hodtani, “Generalized maximum correntropy detector for non-Gaussian environments,” *International Journal of Adaptive Control and Signal Processing*, vol. 32, no. 1, pp. 83–97, 2018.
- [119] S. Pei and C. Tseng, “Least mean p-power error criterion for adaptive fir filter,” *IEEE Journal on Selected Areas in Communications*, vol. 12, no. 9, pp. 1540–1547, 1994.
- [120] A. Y. Aravkin, B. M. Bell, J. V. Burke, and G. Pillonetto, “An ℓ_1 -Laplace robust Kalman smoother,” *IEEE Transactions on Automatic Control*, vol. 56, no. 12, pp. 2898–2911, 2011.
- [121] J. Wiora, “Least L_p-norm low-pass filter,” *Automatica*, vol. 133, p. 109854, 2021.
- [122] R. Ronsse, N. Vitiello, T. Lenzi, J. van den Kieboom, M. C. Carrozza, and A. J. Ijspeert, “Human-robot synchrony: Flexible assistance using adaptive oscillators,” *IEEE Transactions on Biomedical Engineering*, vol. 58, no. 4, pp. 1001–1012, 2011.
- [123] Ludovic Righetti, Jonas Buchli, and Auke Jan Ijspeert, “Dynamic hebbian learning in adaptive frequency oscillators,” *Physica D: Nonlinear Phenomena*, vol. 216, no. 2, pp. 269–281, 2006.
- [124] R. Ronsse, S. M. M. De Rossi, N. Vitiello, T. Lenzi, M. C. Carrozza, and A. J. Ijspeert, “Real-time estimate of velocity and acceleration of quasi-periodic signals using adaptive oscillators,” *IEEE Transactions on Robotics*, vol. 29, no. 3, pp. 783–791, 2013.
- [125] A. J. Young, J. Foss, H. Gannon, and D. P. Ferris, “Influence of power delivery timing on the energetics and biomechanics of humans wearing a hip exoskeleton,” *Frontiers in Bioengineering and Biotechnology*, vol. 5, pp. 4–15, 2017.
- [126] J. Lee, K. Seo, B. Lim, J. Jang, K. Kim, and H. Choi, “Effects of assistance timing on metabolic cost, assistance power, and gait parameters for a hip-type exoskeleton,” in *2017 International Conference on Rehabilitation Robotics (ICORR)*, 2017, pp. 498–504.

- [127] J. Koller, D. Gates, D. P. Ferris, and C. D. Remy, “Body-in-the-loop optimization of assistive robotic devices: A validation study,” in *Robotics: Science and Systems*, 2016.
- [128] M. Tucker, E. Novoseller, C. Kann, Y. Sui, Y. Yue, J. W. Burdick, and A. D. Ames, “Preference-based learning for exoskeleton gait optimization,” in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 2351–2357.
- [129] M. Tucker, M. Cheng, E. Novoseller, R. Cheng, Y. Yue, J. W. Burdick, and A. D. Ames, “Human preference-based learning for high-dimensional optimization of exoskeleton walking gaits,” in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020, pp. 3423–3430.
- [130] K. Li, M. Tucker, E. Bıyük, E. Novoseller, J. W. Burdick, Y. Sui, D. Sadigh, Y. Yue, and A. D. Ames, “Roial: Region of interest active learning for characterizing exoskeleton gait preference landscapes,” in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 3212–3218.
- [131] J. R. Gardner, M. J. Kusner, Z. E. Xu, K. Q. Weinberger, and J. P. Cunningham, “Bayesian optimization with inequality constraints,” in *ICML*, vol. 2014, 2014, pp. 937–945.
- [132] K. Kawaguchi, L. P. Kaelbling, and T. Lozano-Perez, “Bayesian optimization with exponential convergence,” in *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 2*. MIT Press, 2015, p. 28092817.
- [133] J. Snoek, H. Larochelle, and R. P. Adams, “Practical Bayesian optimization of machine learning algorithms,” *Advances in Neural Information Processing Systems*, vol. 25, 2012.
- [134] F. Gustafsson, “Particle filter theory and practice with positioning applications,” *IEEE Aerospace and Electronic Systems Magazine*, vol. 25, no. 7, pp. 53–82, 2010.
- [135] P. R. Davidson and D. M. Wolpert, “Widespread access to predictive models in the motor system: a short review,” *Journal of Neural Engineering*, vol. 2, no. 3, pp. S313–S319, aug 2005.

- [136] C. Basu, Q. Yang, D. Hungerman, M. Sinahal, and A. D. Draqan, “Do you want your autonomous car to drive like you?” in *2017 12th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, 2017, pp. 417–425.
- [137] O. Chapelle, T. Joachims, F. Radlinski, and Y. Yue, “Large-scale validation and analysis of interleaved search evaluation,” *ACM Trans. Inf. Syst.*, vol. 30, no. 1, mar 2012.
- [138] M. Tucker, E. Novoseller, C. Kann, Y. Sui, Y. Yue, J. W. Burdick, and A. D. Ames, “Preference-based learning for exoskeleton gait optimization,” in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 2351–2357.
- [139] J. E. A. Bertram, “Constrained optimization in human walking: cost minimization and gait plasticity,” *Journal of Experimental Biology*, vol. 208, no. 6, pp. 979–991, 2005.
- [140] A. Aravkin, J. V. Burke, L. Ljung, A. Lozano, and G. Pillonetto, “Generalized Kalman smoothing: Modeling and algorithms,” *Automatica*, vol. 86, pp. 63–86, 2017.
- [141] D. Simon, *Optimal State Estimation: Kalman, H_{infinity}, and Nonlinear Approaches*. John Wiley & Sons, 2006.
- [142] K. P. Murphy, *Machine Learning: A Probabilistic Perspective*. MIT press, 2012.
- [143] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. The MIT Press, 2005.
- [144] P. I. Frazier, “A tutorial on Bayesian optimization,” *arXiv preprint arXiv:1807.02811*, 2018.
- [145] “Code and data for MKMCKF-OE,” <https://github.com/lsl-zsj/MKMCKF-OE>, 2022.
- [146] Y. Fan, Y. Zhang, G. Wang, X. Wang, and N. Li, “Maximum correntropy based unscented particle filter for cooperative navigation with heavy-tailed measurement noises,” *Sensors*, vol. 18, no. 10, p. 3183, 2018.

- [147] G. Wang, C. Yang, and X. Ma, “A novel robust nonlinear Kalman filter based on multivariate laplace distribution,” *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 68, no. 7, pp. 2705–2709, 2021.
- [148] S. Li, D. Shi, W. Zou, and L. Shi, “Multi-kernel maximum correntropy Kalman filter,” *IEEE Control Systems Letters*, vol. 6, pp. 1490–1495, 2021.
- [149] H. J. Marquez, *Nonlinear Control Systems: Analysis and Design*. John Wiley & Sons, 2003.
- [150] R. M. Murray, Z. Li, and S. S. Sastry, *A Mathematical Introduction to Robotic Manipulation*. CRC press, 2017.
- [151] V. N. Vapnik, “An overview of statistical learning theory,” *IEEE Transactions on Neural Networks*, vol. 10, no. 5, pp. 988–999, 1999.
- [152] P. S. Bullen, *Handbook of Means and Their Inequalities*. Springer Science & Business Media, 2013, vol. 560.
- [153] A. Piazzoli and A. Visioli, “Global minimum-jerk trajectory planning of robot manipulators,” *IEEE Transactions on Industrial Electronics*, vol. 47, no. 1, pp. 140–149, 2000.
- [154] T. Steihaug, “The conjugate gradient method and trust regions in large scale optimization,” *SIAM Journal on Numerical Analysis*, vol. 20, no. 3, pp. 626–637, 1983.