Czech Technical University in Prague

Faculty of Electrical Engineering
Department of Cybernetics
Multi-robot Systems

State Estimation of an Unmanned Surface Vehicle by an Unmanned Multirotor Helicopter

Master's Thesis

Filip Novák

Prague, May 2022

Study programme: Cybernetics and Robotics

Supervisor: Ing. Tomáš Báča, Ph.D.

Author statement:
I declare that the presented work was developed independently and that I have listed all sources of information used within it in accordance with the methodical instructions for observing the ethical principles in the preparation of university theses.

Prague, date............................ ....
Signature

MASTER‘S THESIS ASSIGNMENT

I. Personal and study details

Student's name:	**Novák Filip**	Personal ID number:	**474570**
Faculty / Institute:	**Faculty of Electrical Engineering**		
Department / Institute:	**Department of Cybernetics**		
Study program:	**Cybernetics and Robotics**		
Branch of study:	**Cybernetics and Robotics**		

II. Master’s thesis details

Master’s thesis title in English:

State Estimation of an Unmanned Surface Vehicle by an Unmanned Multirotor Helicopter

Master’s thesis title in Czech:

Estimace stavů lodi z bezpilotní helikoptéry

Guidelines:

This thesis is motivated by the problem of autonomous deployment of multirotor Unmanned Aerial Vehicles (UAVs) above a water surface. Such deployment will require the UAV to be able to dock on top of a small water Unmanned Surface Vehicle (USV) for recharging. However, since the motion of a USV on top of a wavy water surface is nonlinear, state estimation needs to be employed in order to aid the controller of the UAV. This work will focus on the sub-problem of state estimation and state prediction of the USV from the point of view of a multirotor UAV. We presume that the USV is cooperating and can be equipped with custom hardware, including GNSS receiver and visual markers, to aid with the otherwise very complex problem of precise detection from onboard of the UAV. The thesis will tackle the following points:
- Familiarize yourself with the MRS UAV System [1] for control, estimation, and simulation of multirotor helicopters.
- Study the topic of dynamic system modeling of an Unmanned Surface Vehicle on a water surface [3].
- Study the topic of real-time state observers such as Linear Kalman filter, Unscented Kalman filter, and choose the appropriate tool for estimating the states of an Unmanned Surface Vehicle [2, 3].
- Define the required onboard equipment of the USV and UAV that is needed for real-time state estimation of the USV from the point of view of the UAV.
- Prepare a simulation environment in Gazebo/ROS simulator with a catamaran USV and a quadrotor UAV including the previously defined onboard equipment.
- Design and implement a system for state estimation of the USV from the UAV.
- Verify the proposed system using the simulation environment. If possible, given the current social situation, and if the hardware allows, conduct experiments using a real-world robotic setup.

Bibliography / sources:

[1] Tomas Baca, Matej Petrlik, Matous Vrba, Vojtech Spurny, Robert Penicka, Daniel Hert and Martin Saska. The MRS UAV System: Pushing the Frontiers of Reproducible Research, Real-world Deployment, and Education with Autonomous Unmanned Aerial Vehicles. Journal of Intelligent & Robotic Systems 102(26):1–28, May 2021.
[2] Tomas Baca, Petr Stepan, Vojtech Spurny, Daniel Hert, Robert Penicka, Martin Saska, Justin Thomas, Giuseppe Loianno and Vijay Kumar. Autonomous Landing on a Moving Vehicle with an Unmanned Aerial Vehicle. Journal of Field Robotics 36(5):874-891, 2019.
[3] Fossen, Thor I., and Tristan Perez. "Kalman filtering for positioning and heading control of ships and offshore rigs." IEEE control systems magazine 29.6 (2009): 32-46.

Name and workplace of master’s thesis supervisor:

Ing. Tomáš Báča, Ph.D. Multi-robot Systems FEE

Name and workplace of second master’s thesis supervisor or consultant:

Date of master’s thesis assignment: **25.01.2022** Deadline for master's thesis submission: **20.05.2022**

Assignment valid until: **30.09.2023**

Ing. Tomáš Báča, Ph.D. Supervisor’s signature	prof. Ing. Tomáš Svoboda, Ph.D. Head of department’s signature	prof. Mgr. Petr Páta, Ph.D. Dean’s signature

III. Assignment receipt

The student acknowledges that the master’s thesis is an individual work. The student must produce his thesis without the assistance of others, with the exception of provided consultations. Within the master’s thesis, the author must state the names of consultants and include a list of references.

.

Date of assignment receipt Student’s signature

Acknowledgments

Firstly, I would like to thank my supervisor Ing. Tomáš Báča, Ph.D. for his excellent support throughout this thesis. Furthermore, my thanks goes to all the members of Multi-robot Systems group who helped me with this thesis, especially when performing real-world experiments. Special thanks belongs to my family, that supports me throughout my life. My last thanks is for the rest of my colleagues and friends who helped me complete this thesis

Abstract

This thesis deals with design, implementation, and verification of a system for state estimation of an Unmanned Surface Vehicle (USV) known as a boat by an Unmanned Aerial Vehicle (UAV). First, linear and nonlinear mathematical models of the boat extended by wave dynamics are presented. Introduced mathematical models of the boat are used by Kalman filters that perform estimation of boat states using data from UAV onboard sensors. Kalman filters also use data received from sensors placed on the boat, which is sent to the UAV by a wireless communication link. The result of this thesis is a UAV onboard system that provides state estimation of the boat moving on a wavy water surface. The estimation system of boat states is implemented into the UAV control system. The presented estimation system was tested in a realistic robotic simulator Gazebo, and estimation results were analyzed in detail. Then, the presented estimation system was verified by conducting real-world experiments. Using the proposed estimation system, the UAV followed the boat and landed on it successfully.

Keywords: Unmanned Aerial Vehicles, Unmanned Surface Vehicle, State estimation, Kalman filter, Mathematical boat model, Sensor fusion

Abstrakt

Tato práce se zabývá návrhem, implementací a ověřením systému pro estimaci stavů lodi z bezpilotní helikoptéry. Představen je lineární a nelineární matematický model lodi, který je rozšířen o dynamiku vln. Představené matematické modely lodi jsou využity v Kalmanově filtru, který pro estimaci stavů lodi používá data z palubních senzorů bezpilotní helikoptéry. Kalmanův filtr též využívá data ze senzorů umístěných na lodi, která jsou bezdrátově posílána do bezpilotní helikoptéry. Výsledkem práce je palubní systém pro bezpilotní helikoptéru poskytující estimace stavů lodi pohybující se v prostředí s vlnami. Vytvořený systém pro estimaci stavů lodi je začleněn do řídícího systému bezpilotní helikoptéry. Představený systém je testován v realistickém robotickém simulátoru Gazebo a výsledky estimace stavů lodi jsou detailně analyzovány. Systém estimace stavů lodi byl také ověřen při reálných experimentech. Bezpilotní helikoptéra využívající představený systém následovala loď, na kterou i posléze přistála.

Klíčová slova: Bezpilotní prostředky, Dron, Loď, Estimace stavů, Kalmanův filtr, Matematický model lodi, Senzorická fúze

Contents

List of Figures

List of Tables

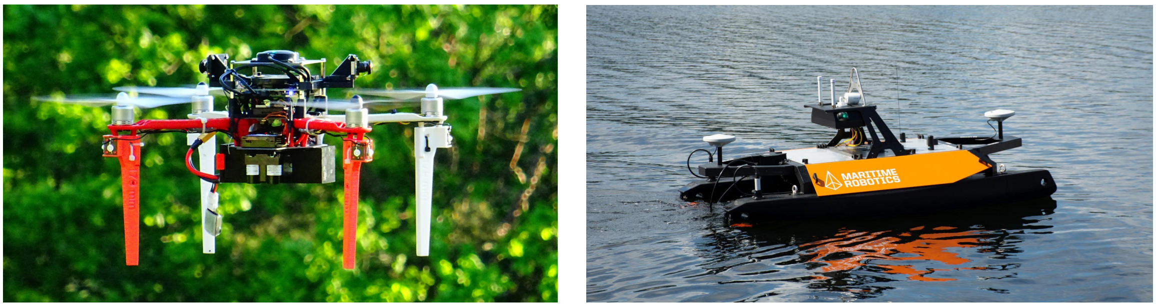
Chapter 1

Introduction

In recent years there has been a significant development in the field of mobile robotics. Key technologies for their development and production are becoming more and more affordable. One of the robot types, whose popularity is rapidly growing every year, is the aerial vehicle. This kind of robot is often called an Unmanned Aerial Vehicle (UAV) or a Micro Aerial Vehicle (MAV), though to the general public it is commonly known as a drone. An example of a UAV is shown in Fig. 1.1a. Typical representatives of aerial robots are multirotor helicopters. Their construction usually consists of several fixed pitch angle propellers mounted on a rigid body that carries necessary electronics. The dimensions of these helicopters can range from centimeters to meters. Thanks to their propellers, multirotor helicopters can take off and land vertically. Their great advantage is also the ability to hover in place.

The popularity of multirotor helicopters is growing due to their wide usage [1]. The UAVs can be used to explore unknown spaces [2], take a photos of a hard-to-reach places [3], capture videos of events such as festivals [4], deliver packages [5], localize fires and aid with firefighting [6], search for lost people in the forest or at the sea after a natural disaster [7], detect the objects in water such as garbage [8], monitor marine mammals [9], survey marine fauna [10]. However, research has started to focus on using a heterogeneous team of robots to accomplish a given task [11]–[13].

One of the currently active research fields of multi-robot systems is cooperation between UAVs and Unmanned Surface Vehicles (USVs) known as ships or boats [14] (see an example of such vehicle in Fig. 1.1b). The USVs operate on the water surface, where they are used to look for and remove garbage from the water [15], assist in dealing with the aftermath of disasters [16], transport materials and objects from one place to another [17], cooperate with rescue services [18], compile data about water environment [19], protect ports [20]. In many cases mentioned above, cooperation between different types of robots results in a better solution to the desired task [12], [16], [21].



(a) An example of UAV.

(b) An example of USV [22].

Figure 1.1: Example of UAV and USV mobile robots.

Important aspect of a robot cooperation system is mutual localization [23]. The UAV needs to know the current state of the USV that typically consists of position, orientation, velocity, and angular velocity [24]. Using this information, the UAV is able to plan its next steps. The UAV can follow the USV, explore different part of the environment than the USV or land on the USV [25], which is the primary motivation of this thesis. In order to land on the USV, the UAV requires precise localization of the landing platform placed on the USV [26]. The estimation of USV states is vital for the UAV to cooperate with the USV in many application scenarios.

The ability to land on the USV is very beneficial for the UAV as the USV can provide services to the UAV. An example of such service is recharging of the UAV battery, because the UAV flight time is mainly limited by battery capacity [27]. Other examples are to unload cargo carried by the UAV [28] and transportation of the UAV [29]. The USV can also provide a safe docking spot for the UAV in case of inclement weather [27]. However, as mentioned above, the key element for a safe landing or any cooperation between the USV and the UAV is a system for estimation of USV states [25], [30], whose design is the main content of this thesis.

1.1 State of the art

The USV described in [14], [24], [31] has become the subject of research in the broad scientific community. The typical usage of the USV is water environment monitoring [32]–[35]. Pollutant tracking using the USV is proposed in [36]. Another usage of the USV is port protection presented in [20]. The USV is also useful for bathymetric measurements providing water depth and information about the floor of oceans, rivers or lakes [37], [38]. As the USVs provide data from the top of the water surface, the UAVs [1], [2], [39] collect information from the air.

The usage of UAVs near a water surface is becoming frequent [8], [9], [40], [41]. One of the tasks of the UAV is conducting marine fauna surveys [10], [40], [42]. The study in [41] focuses on mapping and classification of ecologically sensitive marine habitats using the UAV. The UAV enables us to monitor animals as presented in [9]. Another task is to find and monitor garbage (e.g., plastic marine debris) in the water [8] or to detect oil pollution from ships [43]. Although the UAV and the USV can accomplish the given task on their own, better results can be potentially achieved through their cooperation [29].

The cooperation between the UAV and the USV is very useful especially in marine search and rescue operations [7]. In 2005, a team of USVs and UAVs assisted during rescue and monitoring works after the calamity caused by Hurricane Wilma [11], [16]. The sea rescue system based on a coordinated group of robots consisting of USVs and UAVs is described in [18], where the UAV provided data about castaways position to the USV. In case of a flooded environment, the UAV can build a ground map to plan a path for a rescue boat [21]. A team of USV and UAV is also used to inspect littoral environments and monitor the surroundings of a ship [12], [13]. System for monitoring water pollution based on a combination of USV and UAV is proposed in [15]. The cooperative USV-UAV measurement approach is presented in [19], where low-cost robots are used to collect hydrologic data.

As a result of the increasing usage of the UAV-USV team, the UAV needs to detect the USV and be able to land on it. The USV-UAV platform which enables docking of the UAV on the USV is presented in [29]. The study in [27] introduced a concept of an autonomous solar USV that is designed for autonomous launch and recovery of a UAV. The autonomous landing of a UAV on a USV is proposed in several publications [25], [26], [30], [44]. Most of the landing approaches require detection of a USV and estimation of its states — position, orientation, velocity, and angular velocity.

The approaches to state estimation can be separated into two groups. The first group consists of systems implemented onboard that typically use Global Navigation Satellite System (GNSS) (e.g., Global Positioning System (GPS)) and Inertial Measurement Unit (IMU) to estimate the states [45]–[48]. However, this thesis aims to provide USV states to the UAV to plan the subsequent steps, such as landing or following. Therefore, the estimated states of the USV should be sent to the UAV via a reliable communication link with sufficient frequency and bandwidth. However, the required communication link is challenging to achieve in real-world deployment [49].

State estimation methods in the second group use vision-based relative localization systems that can be placed on another robot to provide information about desired target [23], [50]–[55]. The concept of these systems enables them to be put onboard UAVs that use them to estimate USV states without the need of any communication link. However, these relative localization systems require a target in a detectable position. Usually, the system must be close to the target, and the vision sensor, e.g., the camera, must see the target in its frame. Therefore, in case of a relatively long distance between the UAV and the USV, these relative localization systems do not provide any measurements. Some of them need sufficient light conditions to work properly [50]–[52], i.e. they are not useful in poor lighting conditions such as at night.

Sensor data is then processed in real time to estimate the desired states. Most of the approaches address state estimation using the Kalman filter [56]–[59]. The Kalman filter fuses measurements from different sensors to obtain precise state estimation of a given system [57], [60]. Usage of the Kalman filter to estimate states of the USV is presented in many scientific studies [14], [24], [61]–[63]. The Kalman filter used as a state estimator requires a mathematical model of the system whose states should be estimated.

The mathematical models of the USV are described in [14], [24], [31]. These books present different models that are suitable in many application scenarios. Simulation models are used to precisely capture and reproduce the system's behavior. In order to design a system to control the USV, a simplified model of the USV is used. The most important model for this thesis is observer design model, which contains all the necessary dynamics of the USV for state estimation. More precise model provides better state estimation results and can also be used to predict future states of the system.

1.2 Contributions

This work presents a complex robust system for the USV state estimation from onboard the UAV. Usage of Kalman filter as state observer allows to fuse data from multiple sensors, resulting in accurate estimation. The selected sensors provide sufficient data to estimate all USV dynamics, and thanks to them, the system can be used at any time of day or at a relatively long distance between the USV and the UAV. Moreover, a detailed USV model is proposed to capture the motion of the USV precisely in different environmental conditions such as wavy water surfaces. The presented system also provides a prediction of future USV states using the current estimate and the mathematical model. The system has been implemented into a realistic robotic simulator for further use by other colleagues and verified in many simulations. The proposed system has also been deployed to the real world in different scenarios.

1.3 Outline

This thesis is structured as follows. After the Introduction chapter (Chap. 1), the different types of Kalman filters used as states estimators are presented in Chap. 2. The chapter also contains methods used to verify the performance of presented filters. Chap. 3 proposes a detailed description of a non-linear mathematical USV model, which is then simplified to obtain a linear model. The sensors used to estimate states of the USV are proposed in Chap. 4. They are divided into two groups — sensors placed on the USV and UAV onboard sensors. Chap. 5 presents tools used to simulate the system presented in this thesis. The verification of the designed approach is the content of Chap. 6. The performed real-world experiments are proposed in Chap. 7. Finally, the conclusion of this thesis is written in Chap. 8, where the achieved results are discussed together with proposals of future work.

1.4 Mathematical notation

The mathematical notation used in this thesis is summarized in Table 1.1.

Symbol	Example	Description
lower or uppercase letter	y, Y	scalar
bold lowercase letter	$\boldsymbol{x}$	column vector
bold number **0**	**0**	zero column vector
bold uppercase letter	$\boldsymbol{X}$	matrix
bold uppercase letter $\boldsymbol{I}$	$\boldsymbol{I}$	identity matrix
bold uppercase letter $\boldsymbol{O}$	$\boldsymbol{O}_{x \times y}$	zero matrix of x rows and y columns
symbol $\mathcal{E}$	$\mathcal{E}\{\cdot\}$	mean value of term inside brackets
upper index T	$\boldsymbol{x}^T, \boldsymbol{X}^T$	transposition of vector $\boldsymbol{x}$ or matrix $\boldsymbol{X}$
letter k in brackets after $\boldsymbol{x}$	$\boldsymbol{x}(k)$	vector $\boldsymbol{x}$ at the sample k
lowercase letter accented by a dot	$\dot{x}, \dot{\boldsymbol{x}}$	1^{st} time derivative of $x,\ \boldsymbol{x}$
text *diag* before symbols in brackets	$diag\{x,\ y\}$	diagonal matrix with $x,\ y$ on the diagonal
symbol $\mathbb{R}$	$\mathbb{R}$	set of real numbers
symbol $\mathbb{Z}$	$\mathbb{Z}$	set of integers

Table 1.1: Mathematical notation, nomenclature and notable symbols.

Chapter 2

Kalman filters

Kalman filter, member of the Bayesian filters family, is a standard method of solving optimal filtering problems [57]. The filter was introduced in the year 1960 [56] and has since become one of the most widely used methods for state estimation. Originally the Kalman filter was defined for a discrete Linear time-invariant (LTI) system. Nowadays, there exist many extensions of Kalman filters such as Extended Kalman filter [60] or Unscented Kalman filter [58] that are also usable for nonlinear systems. The Kalman filter can be briefly described as an optimal algorithm that processes data recursively [64].

The filter uses a model of system dynamics as well as models of sensors. However, in a lot of applications, models are simplified and do not describe the reality accurately [65]. Therefore uncertainty of the dynamics models is considered in the Kalman filter. Another essential knowledge relates to the statistical properties of a system and measurement noise. Finally, any information about initial conditions of the variables of interest improves the filter performance [64]. Typical Kalman filter application in control systems is shown in Fig. 2.1.

As mentioned above, the Kalman filter is a recursive algorithm. Only previous estimate and new input data are needed for a new estimate update. Therefore there is no need to store all past observed data. As a result, the Kalman filter is more efficient in terms of computation of new estimates than estimation methods using all past data at each update [57]. It also reduces memory requirements as only the current estimate has to be stored in memory.

The process of Kalman filtering consists of two phases. The first phase is called the prediction step. In the prediction step, the current estimate is propagated through the system model together with the current value of the system input. The second phase of the filtering is known as the data-update step. The current estimate is updated using a new measurement during the data-update step. These two phases are repeated over and over again. Flowchart of a Kalman filter is shown in Fig. 2.2. In the following sections (Sec. 2.1–2.3), different types of Kalman filters are presented.

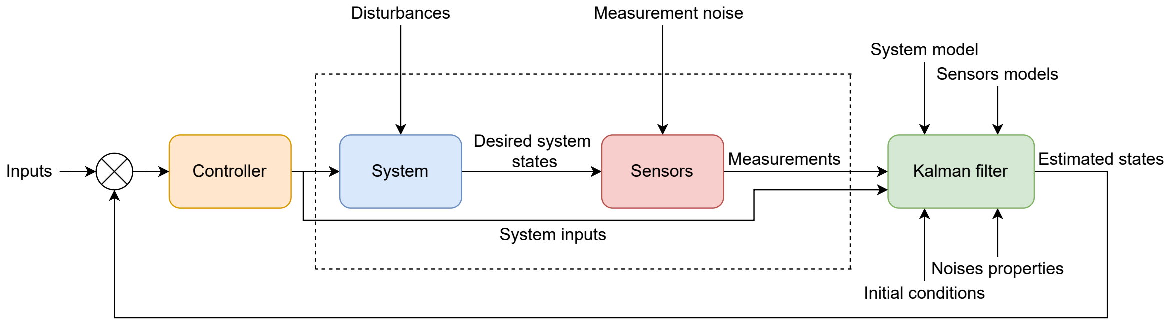


Figure 2.1: Diagram of typical Kalman filter application.

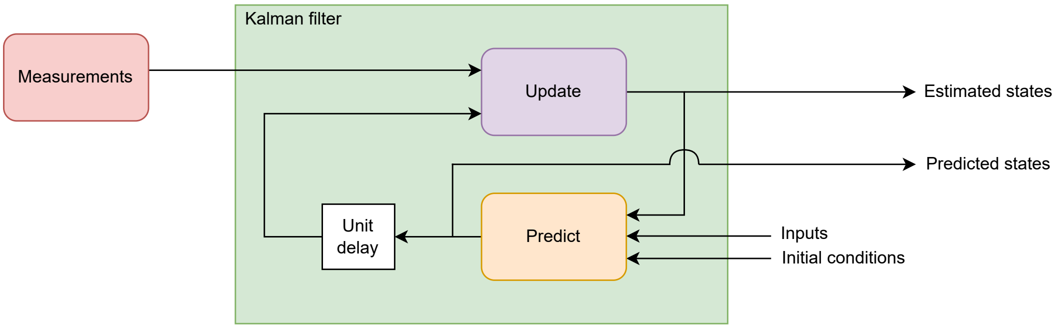


Figure 2.2: Flowchart of Kalman filtering algorithm.

2.1 Linear Kalman filter

According to [56], the Linear Kalman filter was first used on a linear discrete-time stochastic system whose state-space equations are as follows

$$\boldsymbol{x}(k+1) = \boldsymbol{A}\boldsymbol{x}(k) + \boldsymbol{B}\boldsymbol{u}(k) + \boldsymbol{v}(k), \tag{2.1}$$

$$\boldsymbol{y}(k) = \boldsymbol{C}\boldsymbol{x}(k) + \boldsymbol{D}\boldsymbol{u}(k) + \boldsymbol{e}(k). \tag{2.2}$$

State $\boldsymbol{x}(k)$ is defined as a set of variables that fully describe the system behavior in discrete time k [57]. In other words, the state contains all necessary information from past behavior of the system in order to predict its future behavior. State transition matrix $\boldsymbol{A}$ applied on state $\boldsymbol{x}(k)$ and input matrix $\boldsymbol{B}$ applied on input $\boldsymbol{u}(k)$ is used to determine the new state $\boldsymbol{x}(k+1)$. Vector $\boldsymbol{v}(k)$ is process noise. The goal of the filter is to estimate the unknown state $\boldsymbol{x}(k)$. Typically, it is not possible to get the state $\boldsymbol{x}(k)$ directly. However, data from sensors $\boldsymbol{y}(k)$ are available. Measurement $\boldsymbol{y}(k)$ depends on desired state $\boldsymbol{x}(k)$ through matrix $\boldsymbol{C}$ and can also depend on system input $\boldsymbol{u}(k)$ via matrix $\boldsymbol{D}$. The term $\boldsymbol{e}(k)$ is measurement noise. Measurement equation (2.2) varies for different types of sensors.

The process noise $\boldsymbol{v}(k)$ can be intuitively described as an uncertainty of the system model or disturbance acting on the system. The measurement noise $\boldsymbol{e}(k)$ can represent the sensor's inaccuracy. As described in [57], it is assumed that both noises are additive, white and they are normally distributed with zero mean

$$\mathcal{E}\left\{\begin{pmatrix}\boldsymbol{v}(k)\\ \boldsymbol{e}(k)\end{pmatrix}\right\} = \boldsymbol{0} \tag{2.3}$$

and covariances

$$\mathcal{E}\left\{\begin{pmatrix}\boldsymbol{v}(k)\\ \boldsymbol{e}(k)\end{pmatrix}\begin{pmatrix}\boldsymbol{v}(k)\\ \boldsymbol{e}(k)\end{pmatrix}^T\right\} = \begin{pmatrix}\boldsymbol{Q} & \boldsymbol{S}\\ \boldsymbol{S}^T & \boldsymbol{R}\end{pmatrix}. \tag{2.4}$$

If $\boldsymbol{v}(k)$ and $\boldsymbol{e}(k)$ are independent, the matrix $\boldsymbol{S}$ is a zero matrix and prediction step and data-update step can be done separately.

The algorithm of the Linear Kalman filter is described in [57]. Firstly, the estimate of state $\hat{\boldsymbol{x}}(0)$ and its covariance $\boldsymbol{P}(0)$ have to be initialized

$$\hat{\boldsymbol{x}}(0) = \hat{\boldsymbol{x}}_0, \tag{2.5}$$

$$\boldsymbol{P}(0) = \boldsymbol{P}_0. \tag{2.6}$$

Then in each prediction update step, the model of the system is used to propagate the state to the next discrete time step

$$\hat{\boldsymbol{x}}(k+1) = \boldsymbol{A}\hat{\boldsymbol{x}}(k) + \boldsymbol{B}\boldsymbol{u}(k), \tag{2.7}$$

$$\boldsymbol{P}(k+1) = \boldsymbol{A}\boldsymbol{P}(k)\boldsymbol{A}^T + \boldsymbol{Q}. \tag{2.8}$$

In order to use new incoming measurement, the Kalman gain $\boldsymbol{G}(k)$ is computed

$$\boldsymbol{G}(k) = \boldsymbol{P}(k)\boldsymbol{C}^T(k)\left(\boldsymbol{C}\boldsymbol{P}(k)\boldsymbol{C}^T + \boldsymbol{R}\right)^{-1}. \tag{2.9}$$

The estimate of the state is updated together with its covariance

$$\hat{\boldsymbol{x}}(k) = \hat{\boldsymbol{x}}(k) + \boldsymbol{G}(k)(\boldsymbol{y}(k) - \boldsymbol{C}\hat{\boldsymbol{x}}(k)), \tag{2.10}$$

$$\boldsymbol{P}(k) = (\boldsymbol{I} - \boldsymbol{G}(k)\boldsymbol{C})\boldsymbol{P}(k), \tag{2.11}$$

where $\boldsymbol{I}$ is the identity matrix.

2.2 Extended Kalman filter

The Linear Kalman filter presented in Sec. 2.1 is applicable only on linear systems. However, in many applications, the system dynamics and sensors models are nonlinear, and therefore the Linear Kalman filter cannot be used as it is defined. A possible solution to this problem is the Extended Kalman filter [66]. The Extended Kalman filter linearizes the nonlinear model of the system and then applies the same equations as used in the Linear Kalman filter (Sec. 2.1).

The nonlinear discrete-time stochastic system is described by following equations

$$\boldsymbol{x}(k+1) = \boldsymbol{f}(\boldsymbol{x}(k), \boldsymbol{u}(k), \boldsymbol{v}(k), k), \tag{2.12}$$

$$\boldsymbol{y}(k) = \boldsymbol{h}(\boldsymbol{x}(k), \boldsymbol{u}(k), \boldsymbol{e}(k), k), \tag{2.13}$$

where $\boldsymbol{f}(\boldsymbol{x}(k), \boldsymbol{u}(k), \boldsymbol{v}(k), k)$ represents a nonlinear state-space mapping from state $\boldsymbol{x}(k)$ to state $\boldsymbol{x}(k+1)$ and $\boldsymbol{h}(\boldsymbol{x}(k), \boldsymbol{u}(k), \boldsymbol{e}(k), k)$ is a nonlinear model of sensors. The other symbols represent the same variables as in Sec. 2.1. It is assumed that the distribution of state $\boldsymbol{x}(k)$ is Gaussian. Therefore the distribution can be parameterized by mean and covariance.

As mentioned in [66], the nonlinear system described in (2.12) and (2.13) is linearized in each discrete time step k according to the following equations

$$\boldsymbol{A}_D(k) = \left.\frac{\partial \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{v}, k)}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=\boldsymbol{x}(k), \boldsymbol{u}=\boldsymbol{u}(k), \boldsymbol{v}=\boldsymbol{0}}, \tag{2.14}$$

$$\boldsymbol{\Gamma}_{\boldsymbol{v}}(k) = \left.\frac{\partial \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{v}, k)}{\partial \boldsymbol{v}}\right|_{\boldsymbol{x}=\boldsymbol{x}(k), \boldsymbol{u}=\boldsymbol{u}(k), \boldsymbol{v}=\boldsymbol{0}}, \tag{2.15}$$

$$\boldsymbol{C}_D(k) = \left.\frac{\partial \boldsymbol{h}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{e}, k)}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=\boldsymbol{x}(k), \boldsymbol{u}=\boldsymbol{u}(k), \boldsymbol{e}=\boldsymbol{0}}, \tag{2.16}$$

$$\boldsymbol{\Gamma}_{\boldsymbol{e}}(k) = \left.\frac{\partial \boldsymbol{h}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{e}, k)}{\partial \boldsymbol{e}}\right|_{\boldsymbol{x}=\boldsymbol{x}(k), \boldsymbol{u}=\boldsymbol{u}(k), \boldsymbol{e}=\boldsymbol{0}}. \tag{2.17}$$

The assumption of linearization in the Extended Kalman filter (2.14)–(2.17) is differentiable functions $\boldsymbol{f}(\boldsymbol{x}(k), \boldsymbol{u}(k), \boldsymbol{v}(k), k)$ and $\boldsymbol{h}(\boldsymbol{x}(k), \boldsymbol{u}(k), \boldsymbol{e}(k), k)$. Similar to the Linear Kalman filter, the Extended Kalman filter is firstly initialized

$$\hat{\boldsymbol{x}}(0) = \hat{\boldsymbol{x}}_0, \tag{2.18}$$

$$\boldsymbol{P}(0) = \boldsymbol{P}_0. \tag{2.19}$$

The prediction update step of the Extended Kalman filter is performed as

$$\hat{\boldsymbol{x}}(k+1) = \boldsymbol{f}(\boldsymbol{x}(k), \boldsymbol{u}(k), \boldsymbol{0}, k), \tag{2.20}$$

$$\boldsymbol{P}(k+1) = \boldsymbol{A}_D(k)\boldsymbol{P}(k)\boldsymbol{A}_D^T(k) + \boldsymbol{\Gamma}_{\boldsymbol{v}}(k)\boldsymbol{Q}\boldsymbol{\Gamma}_{\boldsymbol{v}}^T(k). \tag{2.21}$$

The following equations define the data-update step of the Extended Kalman filter using incoming measurement $\boldsymbol{y}(k)$

$$\boldsymbol{G}_D(k) = \boldsymbol{P}(k)\boldsymbol{C}_D^T(k)\left(\boldsymbol{C}_D(k)\boldsymbol{P}(k)\boldsymbol{C}_D^T(k) + \boldsymbol{\Gamma}_{\boldsymbol{e}}(k)\boldsymbol{R}\boldsymbol{\Gamma}_{\boldsymbol{e}}^T(k)\right)^{-1}, \tag{2.22}$$

$$\hat{\boldsymbol{x}}(k) = \hat{\boldsymbol{x}}(k) + \boldsymbol{G}_D(k)\left(\boldsymbol{y}(k) - \boldsymbol{h}(\boldsymbol{x}(k), \boldsymbol{u}(k), \boldsymbol{0}, k)\right), \tag{2.23}$$

$$\boldsymbol{P}(k) = \boldsymbol{P}(k) - \boldsymbol{G}_D(k)\left(\boldsymbol{C}_D(k)\boldsymbol{P}(k)\boldsymbol{C}_D^T(k) + \boldsymbol{\Gamma}_{\boldsymbol{e}}(k)\boldsymbol{R}\boldsymbol{\Gamma}_{\boldsymbol{e}}^T(k)\right)\boldsymbol{G}_D^T(k), \tag{2.24}$$

where matrix $\boldsymbol{G}_D(k)$ is known as Kalman gain for the Extended Kalman filter.

The Extended Kalman filter performance deteriorates in case of strongly nonlinear functions (2.12) and (2.13) [66]. A more accurate estimate can be obtained using Iterative Extended Kalman filter [66]. The prediction update step remains the same as presented in (2.20) and (2.21). The data-update step is modified into the following algorithm (Alg. 1), where ϵ_{iekf} represents threshold value and max_i is maximum number of iterations.

Algorithm 1 Data-update step of Iterative Extended Kalman filter

```
$\hat{\boldsymbol{x}}^{(0)}(k) \leftarrow \hat{\boldsymbol{x}}(k)$
$i \leftarrow 0$
repeat
    $\boldsymbol{C}_D^{(i)}(k) \leftarrow \left.\frac{\partial \boldsymbol{h}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{e}, k)}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=\hat{\boldsymbol{x}}^{(i)}(k), \boldsymbol{u}=\boldsymbol{u}(k), \boldsymbol{e}=\boldsymbol{0}}$
    $\boldsymbol{\Gamma}_{\boldsymbol{e}}^{(i)}(k) \leftarrow \left.\frac{\partial \boldsymbol{h}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{e}, k)}{\partial \boldsymbol{e}}\right|_{\boldsymbol{x}=\hat{\boldsymbol{x}}^{(i)}(k), \boldsymbol{u}=\boldsymbol{u}(k), \boldsymbol{e}=\boldsymbol{0}}$
    $\boldsymbol{G}_D^{(i)}(k) \leftarrow \boldsymbol{P}(k)\boldsymbol{C}_D^{(i)^T}(k)\left(\boldsymbol{C}_D^{(i)}(k)\boldsymbol{P}(k)\boldsymbol{C}_D^{(i)^T}(k) + \boldsymbol{\Gamma}_{\boldsymbol{e}}^{(i)}(k)\boldsymbol{R}\boldsymbol{\Gamma}_{\boldsymbol{e}}^{(i)^T}(k)\right)^{-1}$
    $\hat{\boldsymbol{x}}^{(i+1)}(k) \leftarrow \hat{\boldsymbol{x}}(k) + \boldsymbol{G}_D^{(i)}(k)\left(\boldsymbol{y}(k) - \boldsymbol{h}(\hat{\boldsymbol{x}}^{(i)}(k), \boldsymbol{u}(k), \boldsymbol{0}, k) - \boldsymbol{C}_D^{(i)}(k)(\hat{\boldsymbol{x}}(k) - \hat{\boldsymbol{x}}^{(i)}(k))\right)$
    $i \leftarrow i + 1$
until $\left|\hat{\boldsymbol{x}}^{(i-1)}(k) - \hat{\boldsymbol{x}}^{(i)}(k)\right| < \epsilon_{iekf}$ or $i > max_i$
$\hat{\boldsymbol{x}}(k) \leftarrow \hat{\boldsymbol{x}}^{(i)}(k)$
$\boldsymbol{P}(k) \leftarrow \boldsymbol{P}(k) - \boldsymbol{G}_D^{(i-1)}(k)\left(\boldsymbol{C}_D^{(i-1)}(k)\boldsymbol{P}(k)\boldsymbol{C}_D^{(i-1)^T}(k) + \boldsymbol{\Gamma}_{\boldsymbol{e}}^{(i-1)}(k)\boldsymbol{R}\boldsymbol{\Gamma}_{\boldsymbol{e}}^{(i-1)^T}(k)\right)\boldsymbol{G}_D^{(i-1)^T}(k)$
```

2.3 Unscented Kalman filter

In practice, the Extended Kalman filter has well-known problems presented in [58]:

1. Linearization can produce highly unstable filters if the assumptions of local linearity are violated.
2. The derivation of the Jacobian matrices is nontrivial in most applications and often leads to significant implementation difficulties.

Another possible method for the nonlinear system estimation called an Unscented Kalman filter is presented in [58]. The idea behind this filter is to use weighted points to approximate the mean and the covariance of probability distribution together with the new component called unscented transformation. The main advantage of this filter is that no linearization step is required. As a result, the derivation of the Jacobian matrices is not needed. Moreover, the accuracy of the Unscented Kalman filter is better than the accuracy of the Extended Kalman filter as presented in [58], [67]. Therefore, the Linear Kalman filter and the Unscented Kalman filter are selected as states estimators in this thesis.

The unique method that does not need linearization of the system is called unscented transformation. The transformation is used to calculate statistics of random variables on which nonlinear function is applied [58]. According to [58], [67], the fundamental step of the transformation is to approximate mean $\hat{\boldsymbol{x}}$ of the n-dimensional random variable $\boldsymbol{x}$ and its covariance $\boldsymbol{P_{xx}}$ using weighted points called sigma points $\boldsymbol{\chi}$ as follows

$$\boldsymbol{\chi}_0 = \hat{\boldsymbol{x}}, \tag{2.25}$$

$$\boldsymbol{\chi}_i = \hat{\boldsymbol{x}} + \left(\sqrt{(n+\nu)\boldsymbol{P_{xx}}}\right)_{(i)}, \tag{2.26}$$

$$\boldsymbol{\chi}_{i+n} = \hat{\boldsymbol{x}} - \left(\sqrt{(n+\nu)\boldsymbol{P_{xx}}}\right)_{(i-n)}, \tag{2.27}$$

$$w_0 = \frac{\nu}{n+\nu}, \tag{2.28}$$

$$w_i = \frac{1}{2(n+\nu)}, \tag{2.29}$$

$$w_{i+n} = \frac{1}{2(n+\nu)}, \tag{2.30}$$

where $i = 1, \ldots, n$ and $\nu \in \mathbb{R}$. The term w_j is the weight of a corresponding sigma point $\boldsymbol{\chi}_j$ and $(\sqrt{(n+\nu)\boldsymbol{P_{xx}}})_{(i)}$ is the i-th row of the matrix $\sqrt{(n+\nu)\boldsymbol{P_{xx}}}$.

Consider $\boldsymbol{g}(\boldsymbol{x})$ as a nonlinear function that is applied to random variable $\boldsymbol{x}$. The steps of the unscented transformation are defined in [58], [67] as follows

$$\boldsymbol{\mathcal{Y}}_j = \boldsymbol{g}(\boldsymbol{\chi}_j), \tag{2.31}$$

$$\hat{\boldsymbol{y}} = \sum_j w_j \boldsymbol{\mathcal{Y}}_j, \tag{2.32}$$

$$\boldsymbol{P_{yy}} = \sum_j w_j \left(\boldsymbol{\mathcal{Y}}_j - \hat{\boldsymbol{y}}\right)\left(\boldsymbol{\mathcal{Y}}_j - \hat{\boldsymbol{y}}\right)^T, \tag{2.33}$$

where $j = 0, \ldots, 2n$. The term $\hat{\boldsymbol{y}}$ is the mean value of transformed sigma points $\boldsymbol{\mathcal{Y}}_j$ and $\boldsymbol{P_{yy}}$ is their covariance.

The properties of unscented transformation are the subject of many scientific studies [58], [68], [69]. In comparison with the Extended Kalman filter, the Unscented Kalman filter provides higher order of accuracy for mean value thanks to the unscented transformation [58]. The unscented transformation approximates the mean in second-order accuracy. However, the Extended Kalman filter algorithm is able to capture the mean in first-order accuracy [68]. The second order of accuracy for covariance is the same for both approaches [69]. To compute square root of the covariance matrix in (2.26) and (2.27) the Cholesky decomposition can be used as numerical stable method [70]. As shown in equations (2.25)–(2.33), only standard matrix operations such as matrix addition and multiplication are needed to compute the mean and covariance [58]. Against the Extended Kalman filter, the Unscented Kalman filter is faster because there is no need to evaluate the Jacobian matrix [68].

The algorithm of the Unscented Kalman filter for states estimation is presented in [58], [68] under the assumption that $\boldsymbol{v}(k)$ and $\boldsymbol{e}(k)$ are independent. Firstly, the filter is initialized as follows

$$\hat{\boldsymbol{x}}(0) = \hat{\boldsymbol{x}}_0, \tag{2.34}$$

$$\boldsymbol{P}(0) = \boldsymbol{P}_0. \tag{2.35}$$

Then the set of sigma points $\boldsymbol{\chi}^a(k)$ using equations (2.25)–(2.30) is computed in each discrete time step k

$$\boldsymbol{\chi}^a(k) = \{\boldsymbol{\chi}_0(k),\ \boldsymbol{\chi}_i(k),\ \boldsymbol{\chi}_{i+n}(k)\}. \tag{2.36}$$

The prediction update step in discrete time step k using sigma points $\boldsymbol{\chi}^a_j(k)$ from the set $\boldsymbol{\chi}^a(k)$ (2.36) is defined as follows

$$\boldsymbol{\chi}_j(k) = \boldsymbol{f}(\boldsymbol{\chi}^a_j(k-1), \boldsymbol{u}(k), \boldsymbol{0}, k), \tag{2.37}$$

$$\hat{\boldsymbol{x}}(k) = \sum_{j=0}^{2n} w_j \boldsymbol{\chi}_j(k), \tag{2.38}$$

$$\boldsymbol{P}(k) = \sum_{j=0}^{2n} w_j \left(\boldsymbol{\chi}_j(k) - \hat{\boldsymbol{x}}(k)\right)\left(\boldsymbol{\chi}_j(k) - \hat{\boldsymbol{x}}(k)\right)^T, \tag{2.39}$$

where $\boldsymbol{f}(\boldsymbol{\chi}^a_j(k-1), \boldsymbol{u}(k), \boldsymbol{0}, k)$ represents the nonlinear model of system (2.12). The data update step using measurement $\boldsymbol{y}(k)$ proceeds according to the following equations

$$\boldsymbol{\mathcal{Y}}_j(k) = \boldsymbol{h}\left(\boldsymbol{\chi}_j(k), \boldsymbol{u}(k), \boldsymbol{0}, k\right), \tag{2.40}$$

$$\hat{\boldsymbol{y}}(k) = \sum_{j=0}^{2n} w_j \boldsymbol{\mathcal{Y}}_j(k), \tag{2.41}$$

$$\boldsymbol{P}_{\tilde{\boldsymbol{y}}\tilde{\boldsymbol{y}}}(k) = \sum_{j=0}^{2n} w_j \left(\boldsymbol{\mathcal{Y}}_j(k) - \hat{\boldsymbol{y}}(k)\right)\left(\boldsymbol{\mathcal{Y}}_j(k) - \hat{\boldsymbol{y}}(k)\right)^T + \boldsymbol{R}, \tag{2.42}$$

$$\boldsymbol{P}_{\boldsymbol{x}\boldsymbol{y}}(k) = \sum_{j=0}^{2n} w_j \left(\boldsymbol{\chi}_j(k) - \hat{\boldsymbol{x}}(k)\right)\left(\boldsymbol{\mathcal{Y}}_j(k) - \hat{\boldsymbol{y}}(k)\right)^T, \tag{2.43}$$

$$\boldsymbol{G}(k) = \boldsymbol{P}_{\boldsymbol{x}\boldsymbol{y}}(k)\boldsymbol{P}^{-1}_{\tilde{\boldsymbol{y}}\tilde{\boldsymbol{y}}}(k), \tag{2.44}$$

$$\hat{\boldsymbol{x}}(k) = \hat{\boldsymbol{x}}(k) + \boldsymbol{G}(\boldsymbol{y}(k) - \hat{\boldsymbol{y}}(k)), \tag{2.45}$$

$$\boldsymbol{P}(k) = \boldsymbol{P}(k) - \boldsymbol{G}(k)\boldsymbol{P}_{\tilde{\boldsymbol{y}}\tilde{\boldsymbol{y}}}(k)\boldsymbol{G}^T(k), \tag{2.46}$$

where $\boldsymbol{h}(\boldsymbol{\chi}_j(k), \boldsymbol{u}(k), \boldsymbol{0}, k)$ represents nonlinear mapping from states $\hat{\boldsymbol{x}}(k)$ and inputs $\boldsymbol{u}(k)$ to measurements (2.13).

2.4 Verification of Kalman filter

The algorithms for states estimation have been presented. An important step of their development is to verify their performance. One of the most common methods is Root Mean Square Error (RMSE) [71], [72]. The RMSE is computed as

$$RMSE = \sqrt{\frac{\sum_{t=1}^{T} \left(\hat{\boldsymbol{x}}(t) - \boldsymbol{x}_{GT}(t)\right)^T \left(\hat{\boldsymbol{x}}(t) - \boldsymbol{x}_{GT}(t)\right)}{T}}, \tag{2.47}$$

where $\boldsymbol{x}_{GT}(t)$ is ground-truth of estimated state $\hat{\boldsymbol{x}}(t)$ and T is time window in which the RMSE is computed. However, it is usually challenging to obtain the ground-truth $\boldsymbol{x}_{GT}(t)$, especially in real-world conditions.

Another possible method to verify the performance of the Kalman filter is to check the consistency of the filter using innovations [73], [74]. Innovation $\boldsymbol{\zeta}(k)$ is defined as

$$\boldsymbol{\zeta}(k) = \boldsymbol{y}(k) - \hat{\boldsymbol{y}}(k), \tag{2.48}$$

where $\boldsymbol{y}(k)$ is a measurement obtained from the sensors in time step k and $\hat{\boldsymbol{y}}(k)$ is expected measurement with respect to the current estimated state. The covariance of innovation $\boldsymbol{S}(k)$ has the following form

$$\boldsymbol{S}(k) = \mathcal{E}\left\{\boldsymbol{\zeta}(k)\boldsymbol{\zeta}^T(k)\right\}. \tag{2.49}$$

If the filter is working correctly, then the mean of innovation $\boldsymbol{\zeta}(k)$ is zero, and the matrix $\boldsymbol{S}(k)$ is its covariance matrix. For the Linear Kalman filter, the equations (2.48) and (2.49) change to the following form

$$\boldsymbol{\zeta}(k) = \boldsymbol{y}(k) - \boldsymbol{C}\hat{\boldsymbol{x}}(k), \tag{2.50}$$

$$\boldsymbol{S}(k) = \boldsymbol{C}\boldsymbol{P}(k)\boldsymbol{C}^T + \boldsymbol{R}, \tag{2.51}$$

where the $\boldsymbol{P}(k)$ is covariance of the current estimate $\hat{\boldsymbol{x}}(k)$ before data-update step according to the measurement $\boldsymbol{y}(k)$. For the Unscented Kalman filter, the innovation covariance $\boldsymbol{S}(k)$ corresponds to the matrix $\boldsymbol{P}_{\tilde{\boldsymbol{y}}\tilde{\boldsymbol{y}}}(k)$ (2.42).

Test 1 - Innovation magnitude bound test

The first test is to check whether the innovation $\boldsymbol{\zeta}(k)$ is consistent with its covariance $\boldsymbol{S}(k)$. According to [73], [74], the innovation is consistent with its covariance if approximately 95% of the innovation values are within the 95% confidence ellipsoid defined by the matrix $\boldsymbol{S}(k)$. In the scalar case, the test means that approximately 95% of the innovation values lie within bounds $\pm 2\sqrt{S(k)}$.

Test 2 - Normalized innovation squared χ^2 test

The second test is to prove the unbiasedness of the innovation [73], [74]. To perform the test, firstly, the normalized innovation squared $\boldsymbol{q}(k)$ is computed as follows

$$\boldsymbol{q}(k) = \boldsymbol{\zeta}(k)\boldsymbol{S}^{-1}(k)\boldsymbol{\zeta}(k). \tag{2.52}$$

Then the mean value of $\boldsymbol{q}(k)$ is computed

$$\bar{q} = \frac{1}{N}\sum_{k=1}^{N}\boldsymbol{q}(k). \tag{2.53}$$

To pass this test, the $\bar{q}$ should lie in confidence interval $[r_1,\ r_2]$ that is characterized by the hypothesis H_0. The hypothesis H_0 is defined as follows [74]: $N\bar{q}$ is χ^2_{Nm} distributed with probability $P = 1-\alpha$, where m is a dimension of the measurement vector and α defines the confidence region, e.g., $\alpha = 0.05$ specified 95% confidence region,

$$P(N\bar{q} \in [r_1,\ r_2]|H_0) = 1-\alpha. \tag{2.54}$$

Test 3 - Innovation whiteness (autocorrelation) test

The last test tries to show the whiteness of the innovation [74]. The time-averaged correlation is computed during the test

$$corr(\tau) = \frac{1}{N}\sum_{k_r=0}^{N-\tau-1}\zeta(k_r)^T\zeta(k_r+\tau), \tag{2.55}$$

which can be normalized by $corr(0)$. The idea of the test is that for large enough N, $corr(\tau)$ is assumed to be normally distributed with zero mean and variance $\frac{1}{N}$ [74]. Therefore at least 95% of the values of $corr(\tau)$ should be in confidence region defined as $\pm\frac{2}{\sqrt{N}}$.

In all the tests presented above, the performance of the Kalman filter is measured with the assumption that the system's model and noises' statistics are perfectly known. However, the Kalman filter can work incorrectly if the model of the system or statistics of noises is wrong [74]. According to [74], typically, two types of error can happen:

- Error in the statistics of process and measurement noise,
- Error in the model of the system.

If the covariance of the process noise or the measurement noise is under-estimated, more values than 5% fall outside of the confidence ellipsoid (test 1). Moreover, in test 2, the mean of normalized innovation squared does not fit the confidence interval defined by hypothesis H_0 because the mean is larger. The meaning of these results is that the combined process and measurement noises levels are too low [74].

Another possibility is that process noise's covariance, or measurement noise's covariance is over-estimated. In such a case, the innovation is within the required confidence ellipsoid (test 1). However, the mean of the normalized innovation squared is smaller than the confidence interval defined by hypothesis H_0 (test 2). This means that the combined process and measurement noise levels are too high [74].

The second error called the mis-matched filter problem [74] happens when the model of the measured system is wrong. The consequence is that mean of normalized innovation squared values contains drift. This type of error is nicely seen in all presented tests. To reduce the effects of mis-matched filter problem, the covariance matrix of process noise $\boldsymbol{Q}$ can be boosted in magnitude. This leads to an increased influence of measurements on state estimation, i.e. the estimates follow measurements more closely [74].

Chapter 3

Mathematical USV models

The mathematical USV model is one of the most important components of the state estimator designed in this thesis. The USV model is used to fuse multiple sensors data (Chap. 4) and also to predict the future states of the USV that correspond to the USV model. In this chapter, a full nonlinear model of the USV is presented. The nonlinear model is then simplified to obtain a linear model that can be used for linear state estimators such as the Linear Kalman filter (Chap. 2). The approach in this thesis focuses on state estimation of the USV moving in water areas with waves, e.g., sea or ocean [75]. Therefore, the interaction of the waves with the USV model is introduced to be used in the state estimator to obtain better precision.

The USV moves in six Degrees of Freedom (DOFs), which means six independent coordinates are required to fully describe the orientation and position of the USV. The DOFs are composed of position displacements and rotations with respect to three axes in body frame of the USV (see Fig. 3.1). The longitudinal motion of the USV in direction of the x_b axis is called a *surge*. The rotation around the x_b axis is known as a *roll*. A *sway* is lateral motion, also called sideways motion, in the direction of the y_b axis. The rotation corresponding to the y_b axis is named *pitch*. The last motion known as a *heave* takes place in a vertical direction with respect to the z_b axis, and a *yaw* is the corresponding rotation. The notation of presented DOFs of the USV together with their forces, moments, and time derivatives representing linear and angular velocities is summarized in Table 3.1.

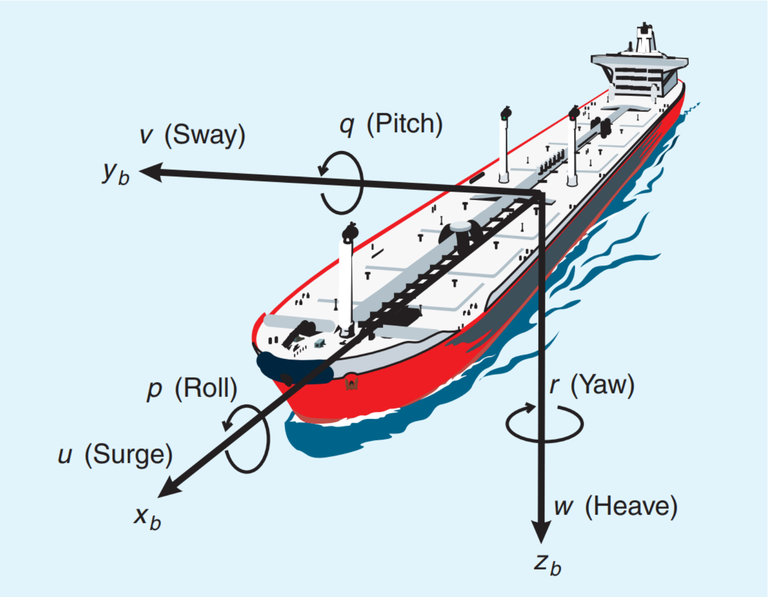


Figure 3.1: Motion of the USV in six DOFs [61, p. 36].

DOF	meaning	Forces and moments	Linear and angular velocities	Positions and Euler angles
1	surge	X	u	x
2	sway	Y	v	y
3	heave	Z	w	z
4	roll	K	p	ϕ
5	pitch	M	q	θ
6	yaw	N	r	ψ

Table 3.1: The notation of DOFs for the USV.

The USV models can be classified into three different types according to their usage [14]. The first class consists of **simulation models**. A simulation model should describe the system most accurately. Therefore, the model uses all DOFs of the system to capture the complex behavior of the system. For the USV, it means that the simulation model contains USV dynamics, actuators, and sensors models, environmental disturbances such as wind, waves, water current, etc. The main goal of the simulation model is to provide the same time response as the real system. However, these models are often very complex. To control the motion of the USV, the simulation model is simplified, and the number of DOFs can be reduced to get the **control design model**. The one DOF model (yaw) is suitable for designing a heading autopilot (yaw controller). The three DOF model consisting of the surge, sway, and yaw is sufficient for a path-following system.

The third type of model is called **observer design model** [14] which this chapter focuses on. Unlike the control design model, the observer design model captures the additional dynamics, i.e. disturbances, sensors, navigation systems, etc. However, the observer design model is still simpler than the simulation one as it captures only dynamics relevant to the desired task, e.g., three DOFs — surge, sway and yaw in the path-following approach. The observer design model of the USV very often incorporates models of waves, wind, and water currents in order to provide precise filtering and motion prediction of desired states. The model's diagram is shown in Fig. 3.2 with a marked observer design model that is used in this thesis.

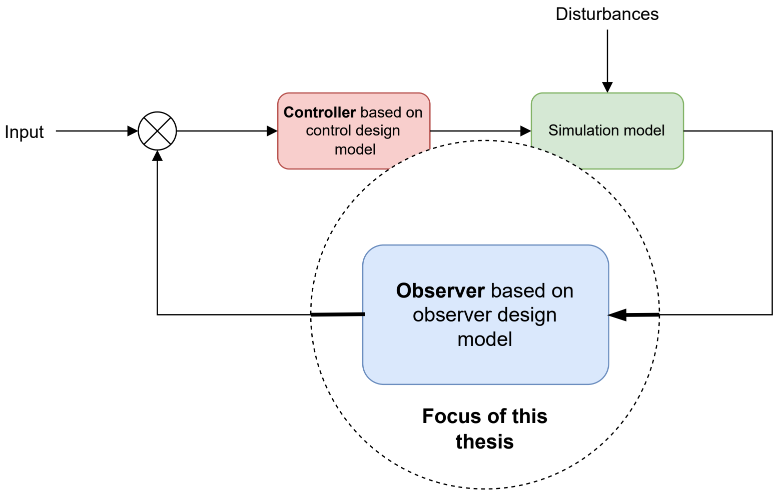

Disturbances
Input
Controller based on control design model
Simulation model
Observer based on observer design model
Focus of this thesis

Figure 3.2: Three different types of models in control system application.

The motion of the USV is modeled using two theories — the **Maneuvering theory** and the **Seakeeping theory** [14]. Maneuvering theory studies the motion of the USV at constant positive speed U in calm water where wave disturbances are not presented, e.g., harbor or sheltered waters. The main assumption of this theory is the frequency independence of hydrodynamic (maneuvering) coefficients, which means no wave excitation. As a result, the added mass and damping due to motion in water can be modeled by constant parameters [14]. The second theory — Seakeeping theory — studies the motion of the USV at a constant heading and zero or constant speed in the water with waves. The hydrodynamic coefficients depend on the frequency of wave excitation where the distribution of mass and geometry of the hull are taken into account [14]. The unification of these two theories results in a model describing the motion of the USV in a seaway where the most important terms of both theories are preserved [14].

3.1 Kinematics

Geometrical aspects of the USV motion are studied using kinematic analysis. The kinematic transformation has to be defined to describe the motion of the USV in different coordinate frames [24]. During analysis of the USV motion, two coordinate frames shown in Fig. 3.3 are usually defined [31]. The first one is fixed with body frame of the USV (see Fig. 3.1). The origin of this frame is typically put into the center of gravity. The body-fixed frame is transformed into a global coordinate frame whose origin is fixed on the Earth surface. It is assumed, that position and orientation of the USV is expressed in a global coordinate frame while linear and angular velocities are described in a body-fixed coordinate frame [31]. According to the coordinate frames presented above, the following vectors describe the USV motion

$$\boldsymbol{\eta} = (\boldsymbol{p}^T,\ \boldsymbol{\Theta}^T)^T = (x,\ y,\ z,\ \phi,\ \theta,\ \psi)^T, \tag{3.1}$$

$$\boldsymbol{\nu} = (\boldsymbol{v}^T,\ \boldsymbol{\omega}^T)^T = (u,\ v,\ w,\ p,\ q,\ r)^T, \tag{3.2}$$

$$\boldsymbol{\tau} = (\boldsymbol{\tau}_1^T,\ \boldsymbol{\tau}_2^T)^T = (X,\ Y,\ Z,\ K,\ M,\ N)^T, \tag{3.3}$$

where $\boldsymbol{\eta}$ denotes position $\boldsymbol{p} = (x,\ y,\ z)^T$ and orientation $\boldsymbol{\Theta} = (\phi,\ \theta,\ \psi)^T$ in terms of intrinsic Euler angles [76] in a global coordinate frame. The vector $\boldsymbol{\nu}$ denotes linear velocity $\boldsymbol{v} = (u,\ v,\ w)^T$ and angular velocity $\boldsymbol{\omega} = (p,\ q,\ r)^T$ in a body-fixed coordinate frame. The term $\boldsymbol{\tau}$ represents forces $\boldsymbol{\tau}_1 = (X,\ Y,\ Z)^T$ and moments $\boldsymbol{\tau}_2 = (K,\ M,\ N)^T$ relative to body-fixed coordinate frame.

The transformation of body-fixed frame variables to global variables is given by

$$\dot{\boldsymbol{p}} = \boldsymbol{J}_1(\boldsymbol{\Theta})\boldsymbol{v}, \tag{3.4}$$

$$\dot{\boldsymbol{\Theta}} = \boldsymbol{J}_2(\boldsymbol{\Theta})\boldsymbol{\omega}, \tag{3.5}$$

where $\boldsymbol{J}_1(\boldsymbol{\Theta})$ and $\boldsymbol{J}_2(\boldsymbol{\Theta})$ are the transformations depending on the current Euler angles $\boldsymbol{\Theta}$. The $\dot{\boldsymbol{p}}$ denotes velocity of the USV in global frame and $\dot{\boldsymbol{\Theta}}$ denotes angular rates of Euler angles. The inverse transformation is as follows

$$\boldsymbol{v} = \boldsymbol{J}_1^{-1}(\boldsymbol{\Theta})\dot{\boldsymbol{p}}, \tag{3.6}$$

$$\boldsymbol{\omega} = \boldsymbol{J}_2^{-1}(\boldsymbol{\Theta})\dot{\boldsymbol{\Theta}}. \tag{3.7}$$

The transformation $\boldsymbol{J}_1(\boldsymbol{\Theta})$ is commonly composed of three rotations $\boldsymbol{R}_{\boldsymbol{\Theta}_i}$ [31]. Each rotation is described by one Euler angle according to the following matrices

$$\boldsymbol{R}_\phi = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi & \cos\phi \end{pmatrix}, \tag{3.8}$$

$$\boldsymbol{R}_\theta = \begin{pmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{pmatrix}, \tag{3.9}$$

$$\boldsymbol{R}_\psi = \begin{pmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{pmatrix}. \tag{3.10}$$

Beware of using rotations $\boldsymbol{R}_{\boldsymbol{\Theta}_i}$ mentioned above. A correctly performed transformation depends on the order in which these rotations are applied. Usually, the intrinsic zyx–convention is used to transform variables from a global frame to a body-fixed frame [14]. This means that the system in a global frame is first rotated by yaw angle ψ about the z axis. Then, the obtained system is rotated by pitch angle θ about the new y axis. Finally, the system resulting from previous two steps is rotated by roll angle ϕ about the new x axis. The transformation from a body-fixed frame to a global frame is mathematically written as follows

$$\boldsymbol{J}_1(\boldsymbol{\Theta}) = \boldsymbol{R}_\psi \boldsymbol{R}_\theta \boldsymbol{R}_\phi. \tag{3.11}$$

Thanks to the orthogonality of the rotation matrices, the inverse transformation $\boldsymbol{J}_1^{-1}(\boldsymbol{\Theta})$ is expressed by (intrinsic zyx–convention)

$$\boldsymbol{J}_1^{-1}(\boldsymbol{\Theta}) = \boldsymbol{R}_\phi^T \boldsymbol{R}_\theta^T \boldsymbol{R}_\psi^T = \boldsymbol{J}_1^T(\boldsymbol{\Theta}). \tag{3.12}$$

The transformation of angular velocities $\boldsymbol{\omega}$ from a body-fixed frame to Euler angle rates $\dot{\boldsymbol{\Theta}}$ in a global frame is given by matrix $\boldsymbol{J}_2(\boldsymbol{\Theta})$. According to [14], the transformation $\boldsymbol{J}_2^{-1}(\boldsymbol{\Theta})$ is derived as follows

$$\boldsymbol{\omega} = \begin{pmatrix} \dot{\phi} \\ 0 \\ 0 \end{pmatrix} + \boldsymbol{R}_\phi^T \begin{pmatrix} 0 \\ \dot{\theta} \\ 0 \end{pmatrix} + \boldsymbol{R}_\phi^T \boldsymbol{R}_\theta^T \begin{pmatrix} 0 \\ 0 \\ \dot{\psi} \end{pmatrix} = \boldsymbol{J}_2^{-1}(\boldsymbol{\Theta})\dot{\boldsymbol{\Theta}}. \tag{3.13}$$

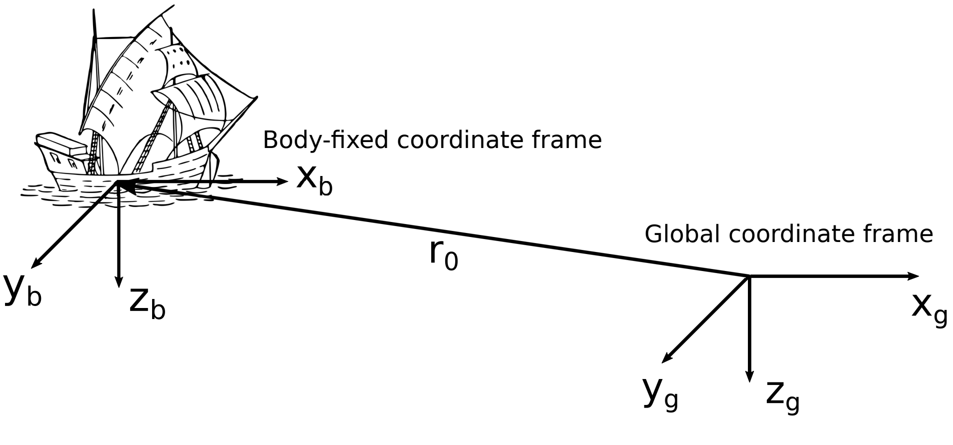


Figure 3.3: Two coordinate frames used in analysis of the USV motion.

The (3.13) can be written in matrix form as follows

$$\boldsymbol{\omega} = \begin{pmatrix} 1 & 0 & -\sin\theta \\ 0 & \cos\phi & \sin\phi\cos\theta \\ 0 & -\sin\phi & \cos\phi\cos\theta \end{pmatrix} \dot{\boldsymbol{\Theta}}. \tag{3.14}$$

To obtain desired transformation $\boldsymbol{J}_2(\boldsymbol{\Theta})$ from a body-fixed frame to a global frame, the inversion of matrix $\boldsymbol{J}_2^{-1}(\boldsymbol{\Theta})$ defined in (3.14) is computed

$$\boldsymbol{J}_2(\boldsymbol{\Theta}) = \left(\boldsymbol{J}_2^{-1}(\boldsymbol{\Theta})\right)^{-1} = \begin{pmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \dfrac{\sin\phi}{\cos\theta} & \dfrac{\cos\phi}{\cos\theta} \end{pmatrix}. \tag{3.15}$$

However, transformation matrix $\boldsymbol{J}_2(\boldsymbol{\Theta})$ is not defined for $\theta = \frac{\pi}{2} + k\pi,\ k \in \mathbb{Z}$. The USV moving on the water surface will probably not get into such a situation. Nevertheless, the solution to such a problem is to use two conventions of the Euler angles representations that have different singularities. If the USV is close to the singularity point in one convention, it is switched to another convention [14].

To summarize, the kinematic transformation is mathematically described as follows

$$\begin{pmatrix} \dot{\boldsymbol{p}} \\ \dot{\boldsymbol{\Theta}} \end{pmatrix} = \begin{pmatrix} \boldsymbol{J}_1(\boldsymbol{\Theta}) & \boldsymbol{O}_{3\times3} \\ \boldsymbol{O}_{3\times3} & \boldsymbol{J}_2(\boldsymbol{\Theta}) \end{pmatrix} \begin{pmatrix} \boldsymbol{v} \\ \boldsymbol{\omega} \end{pmatrix}, \tag{3.16}$$

or, equivalently,

$$\dot{\boldsymbol{\eta}} = \boldsymbol{J}(\boldsymbol{\eta})\boldsymbol{\nu}. \tag{3.17}$$

3.2 Nonlinear USV model

The nonlinear six DOF USV model is presented in [14], [24], [31] as

$$\dot{\boldsymbol{\eta}} = \boldsymbol{J}(\boldsymbol{\eta})\boldsymbol{\nu}, \tag{3.18}$$

$$\boldsymbol{M}\dot{\boldsymbol{\nu}} + \boldsymbol{C}(\boldsymbol{\nu})\boldsymbol{\nu} + \boldsymbol{D}(\boldsymbol{\nu})\boldsymbol{\nu} + \boldsymbol{g}(\boldsymbol{\eta}) = \boldsymbol{\tau}, \tag{3.19}$$

where

- $\boldsymbol{M}$ represents inertia matrix,
- $\boldsymbol{C}(\boldsymbol{\nu})$ denotes matrix of Coriolis and centripetal terms,
- $\boldsymbol{D}(\boldsymbol{\nu})$ is damping matrix,
- $\boldsymbol{g}(\boldsymbol{\eta})$ represents gravitational forces and moments,
- $\boldsymbol{\tau}$ denotes vector forces acting on the USV, e.g., wave forces, wind forces, and control inputs.

The inertia $\boldsymbol{M}$ is composed of two matrices

$$\boldsymbol{M} = \boldsymbol{M}_{RB} + \boldsymbol{M}_{A}. \tag{3.20}$$

The matrix $\boldsymbol{M}_{RB}$ is the positive definite rigid-body mass matrix that is written as

$$\boldsymbol{M}_{RB} = \begin{pmatrix} m\boldsymbol{I}_{3\times3} & \boldsymbol{O}_{3\times3} \\ \boldsymbol{O}_{3\times3} & \boldsymbol{I}_b \end{pmatrix} = \begin{pmatrix} m & 0 & 0 & 0 & 0 & 0 \\ 0 & m & 0 & 0 & 0 & 0 \\ 0 & 0 & m & 0 & 0 & 0 \\ 0 & 0 & 0 & I_x & -I_{xy} & -I_{xz} \\ 0 & 0 & 0 & -I_{yx} & I_y & -I_{yz} \\ 0 & 0 & 0 & -I_{zx} & -I_{zy} & I_z \end{pmatrix}, \tag{3.21}$$

where m is the mass of the USV and $\boldsymbol{I}_{3\times3}$ is the identity matrix. The $\boldsymbol{I}_b$ is the inertia matrix, whose components $I_x,\ I_y,\ I_z$ are the moments of inertia about corresponding body-fixed frame axes $x_b,\ y_b,\ z_b$ and $I_{xy} = I_{yx}$, $I_{xz} = I_{zx}$, $I_{yz} = I_{zy}$ are the products of inertia [14] determined as

$$I_x = \int_V (y^2 + z^2)\rho_m \mathrm{d}V, \tag{3.22}$$

$$I_y = \int_V (x^2 + z^2)\rho_m \mathrm{d}V, \tag{3.23}$$

$$I_z = \int_V (x^2 + y^2)\rho_m \mathrm{d}V, \tag{3.24}$$

$$I_{xy} = \int_V xy\rho_m \mathrm{d}V = \int_V yx\rho_m \mathrm{d}V = I_{yx}, \tag{3.25}$$

$$I_{xz} = \int_V xz\rho_m \mathrm{d}V = \int_V zx\rho_m \mathrm{d}V = I_{zx}, \tag{3.26}$$

$$I_{yz} = \int_V yz\rho_m \mathrm{d}V = \int_V zy\rho_m \mathrm{d}V = I_{zy}, \tag{3.27}$$

where ρ_m is a mass density and V is the volume of the body. The matrix $\boldsymbol{M}_A$ is the virtual hydrodynamic added mass defined in [14] as *"a virtual mass added to a system because an accelerating or decelerating body must move some volume of the surrounding fluid as it moves through it. Moreover, the object and fluid cannot occupy the same physical space simultaneously."* The matrix $\boldsymbol{M}_A$ can be mathematically expressed as

$$\boldsymbol{M}_A = \begin{pmatrix} X_{\dot{u}} & X_{\dot{v}} & X_{\dot{w}} & X_{\dot{p}} & X_{\dot{q}} & X_{\dot{r}} \\ Y_{\dot{u}} & Y_{\dot{v}} & Y_{\dot{w}} & Y_{\dot{p}} & Y_{\dot{q}} & Y_{\dot{r}} \\ Z_{\dot{u}} & Z_{\dot{v}} & Z_{\dot{w}} & Z_{\dot{p}} & Z_{\dot{q}} & Z_{\dot{r}} \\ K_{\dot{u}} & K_{\dot{v}} & K_{\dot{w}} & K_{\dot{p}} & K_{\dot{q}} & K_{\dot{r}} \\ M_{\dot{u}} & M_{\dot{v}} & M_{\dot{w}} & M_{\dot{p}} & M_{\dot{q}} & M_{\dot{r}} \\ N_{\dot{u}} & N_{\dot{v}} & N_{\dot{w}} & N_{\dot{p}} & N_{\dot{q}} & N_{\dot{r}} \end{pmatrix}, \tag{3.28}$$

where elements $X_{\dot{u}},\ X_{\dot{v}}, \dots,\ N_{\dot{r}}$ are coefficients of hydrodynamic added mass.

The Coriolis and centripetal matrix $\boldsymbol{C}(\boldsymbol{\nu})$ is similarly to inertia matrix $\boldsymbol{M}$ (3.20) composed of two matrices

$$\boldsymbol{C}(\boldsymbol{\nu}) = \boldsymbol{C}_{RB}(\boldsymbol{\nu}) + \boldsymbol{C}_A(\boldsymbol{\nu}). \tag{3.29}$$

The Coriolis and the centripetal term is a result of the rotation of the body-fixed reference frame [14]. The $\boldsymbol{C}_{RB}$ stands for rigid-body Coriolis and centripetal matrix, which can be expressed according to [14] as follows

$$\boldsymbol{C}_{RB} = \begin{pmatrix} 0 & 0 & 0 & 0 & mw & -mv \\ 0 & 0 & 0 & -mw & 0 & mu \\ 0 & 0 & 0 & mv & -mu & 0 \\ 0 & mw & -mv & 0 & -I_{yz}q - I_{xz}p + I_z r & I_{yz}r + I_{xy}p - I_y q \\ -mw & 0 & mu & I_{yz}q + I_{xz}p - I_z r & 0 & -I_{xz}r - I_{xy}q + I_x p \\ mv & -mu & 0 & -I_{yz}r - I_{xy}p + I_y q & I_{xz}r + I_{xy}q - I_x p & 0 \end{pmatrix}. \tag{3.30}$$

In order to move the rigid body in a fluid, the hydrodynamic Coriolis and centripetal matrix $\boldsymbol{C}_A(\boldsymbol{\nu})$ is defined in [14] as

$$\boldsymbol{C}_A(\boldsymbol{\nu}) = \begin{pmatrix} 0 & 0 & 0 & 0 & -Ca_3 & Ca_2 \\ 0 & 0 & 0 & Ca_3 & 0 & -Ca_1 \\ 0 & 0 & 0 & -Ca_2 & Ca_1 & 0 \\ 0 & -Ca_3 & Ca_2 & 0 & -Cb_3 & Cb_2 \\ Ca_3 & 0 & -Ca_1 & Cb_3 & 0 & -Cb_1 \\ -Ca_2 & Ca_1 & 0 & -Cb_2 & Cb_1 & 0 \end{pmatrix}, \tag{3.31}$$

where the individual elements of the matrix $\boldsymbol{C}_A(\boldsymbol{\nu})$ are defined as

$$Ca_1 = X_{\dot{u}}u + X_{\dot{v}}v + X_{\dot{w}}w + X_{\dot{p}}p + X_{\dot{q}}q + X_{\dot{r}}r, \tag{3.32}$$

$$Ca_2 = Y_{\dot{u}}u + Y_{\dot{v}}v + Y_{\dot{w}}w + Y_{\dot{p}}p + Y_{\dot{q}}q + Y_{\dot{r}}r, \tag{3.33}$$

$$Ca_3 = Z_{\dot{u}}u + Z_{\dot{v}}v + Z_{\dot{w}}w + Z_{\dot{p}}p + Z_{\dot{q}}q + Z_{\dot{r}}r, \tag{3.34}$$

$$Cb_1 = K_{\dot{u}}u + K_{\dot{v}}v + K_{\dot{w}}w + K_{\dot{p}}p + K_{\dot{q}}q + K_{\dot{r}}r, \tag{3.35}$$

$$Cb_2 = M_{\dot{u}}u + M_{\dot{v}}v + M_{\dot{w}}w + M_{\dot{p}}p + M_{\dot{q}}q + M_{\dot{r}}r, \tag{3.36}$$

$$Cb_3 = N_{\dot{u}}u + N_{\dot{v}}v + N_{\dot{w}}w + N_{\dot{p}}p + N_{\dot{q}}q + N_{\dot{r}}r. \tag{3.37}$$

The term $\boldsymbol{D}(\boldsymbol{\nu})$ represents the damping of the system. As presented in [14], the nonlinear damping $\boldsymbol{D}(\boldsymbol{\nu})$ can be expressed as linear damping

$$\boldsymbol{D}(\boldsymbol{\nu}) \approx \boldsymbol{D} = \begin{pmatrix} X_u & 0 & 0 & 0 & 0 & 0 \\ 0 & Y_v & 0 & Y_p & 0 & Y_r \\ 0 & 0 & Z_w & 0 & Z_q & 0 \\ 0 & K_v & 0 & K_p & 0 & K_r \\ 0 & 0 & M_w & 0 & M_q & 0 \\ 0 & N_v & 0 & N_p & 0 & N_r \end{pmatrix}, \tag{3.38}$$

where the coefficients X_u, Y_v, Y_p, Y_r, Z_w, Z_q, K_v, K_p, K_r, M_w, M_q, N_v, N_p, N_r are known as hydrodynamic derivatives. There are several causes of the damping — potential damping, skin friction, wave drift damping, damping due to vortex shedding, and lifting forces [14].

The restoring forces $\boldsymbol{g}(\boldsymbol{\eta})$ for the USV are based on Archimedes principle where weight W_{USV} is equal to buoyancy B_{USV}

$$W_{USV} = mg = \rho g \nabla = B_{USV}, \tag{3.39}$$

where g is the acceleration of gravity (positive downwards), ρ is the fluid (water) density, and ∇ is a fluid volume that the body displaces. According to [14], the function $\boldsymbol{g}(\boldsymbol{\eta})$ can be rewritten using linear approximation as

$$\boldsymbol{g}(\boldsymbol{\eta}) \approx \boldsymbol{G}\boldsymbol{\eta}, \tag{3.40}$$

where matrix $\boldsymbol{G}$ has the following form

$$\boldsymbol{G} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -Z_z & 0 & -Z_\theta & 0 \\ 0 & 0 & 0 & -K_\phi & 0 & 0 \\ 0 & 0 & -M_z & 0 & -M_\theta & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}. \tag{3.41}$$

The coefficients of the matrix $\boldsymbol{G}$ are determined using formulas presented in [14].

The vector of forces $\boldsymbol{\tau}$ acting on the USV is often difficult to determine. The wave effects are presented in a separate section (Sec. 3.4) as it is one of the challenges for states estimation that is dealt with in this thesis. Under assumption no acting forces $\boldsymbol{\tau}$, the nonlinear model of the USV is expressed as

$$\dot{\boldsymbol{\eta}} = \boldsymbol{J}(\boldsymbol{\eta})\boldsymbol{\nu}, \tag{3.42}$$

$$\dot{\boldsymbol{\nu}} = \boldsymbol{M}^{-1} \left(-\boldsymbol{C}(\boldsymbol{\nu})\boldsymbol{\nu} - \boldsymbol{D}(\boldsymbol{\nu})\boldsymbol{\nu} - \boldsymbol{G}\boldsymbol{\eta} \right). \tag{3.43}$$

Let's define $\boldsymbol{x}_{USV} = (\boldsymbol{\eta}^T,\ \boldsymbol{\nu}^T)^T$, the nonlinear model of the USV defined in (3.42) and (3.43) can be written as

$$\dot{\boldsymbol{x}}_{USV} = \boldsymbol{f}_{USV}(\boldsymbol{x}_{USV}). \tag{3.44}$$

The nonlinear USV model in (3.44) defines derivatives of USV states. To obtain new state $\boldsymbol{x}_{USV}(t+h)$ after time step h from current state $\boldsymbol{x}_{USV}(t)$, the numerical integration of function $\boldsymbol{f}_{USV}(\boldsymbol{x}_{USV})$ has to be performed [77]. One of the simplest algorithms for this task is known as Forward Euler integration

$$\boldsymbol{x}_{USV}(t+h) = \boldsymbol{x}_{USV}(t) + \boldsymbol{f}_{USV}(\boldsymbol{x}_{USV})h. \tag{3.45}$$

However, the Forward Euler method becomes inaccurate for fast or lightly damped systems. The decreasing time step h, i.e. to use faster sampling, is not a sufficient adjustment for all possible situations [78]. The higher-order approximation accuracy can be obtained using Runge-Kutta methods. The well-known and often used algorithm is the Runge-Kutta method of fourth order

$$\dot{\boldsymbol{x}}_{USV}(t) = \boldsymbol{f}_{USV}(\boldsymbol{x}_{USV}(t)), \tag{3.46}$$

$$\boldsymbol{x}_{USV}^{P1} = \boldsymbol{x}_{USV}(t) + \frac{h}{2}\dot{\boldsymbol{x}}_{USV}(t), \tag{3.47}$$

$$\dot{\boldsymbol{x}}_{USV}^{P1} = \boldsymbol{f}_{USV}(\boldsymbol{x}_{USV}^{P1}), \tag{3.48}$$

$$\boldsymbol{x}_{USV}^{P2} = \boldsymbol{x}_{USV}(t) + \frac{h}{2}\dot{\boldsymbol{x}}_{USV}^{P1}, \tag{3.49}$$

$$\dot{\boldsymbol{x}}_{USV}^{P2} = \boldsymbol{f}_{USV}(\boldsymbol{x}_{USV}^{P2}), \tag{3.50}$$

$$\boldsymbol{x}_{USV}^{P3} = \boldsymbol{x}_{USV}(t) + h\dot{\boldsymbol{x}}_{USV}^{P2}, \tag{3.51}$$

$$\dot{\boldsymbol{x}}_{USV}^{P3} = \boldsymbol{f}_{USV}(\boldsymbol{x}_{USV}^{P3}), \tag{3.52}$$

$$\boldsymbol{x}_{USV}(t+h) = \boldsymbol{x}_{USV}(t) + \frac{h}{6}\left(\dot{\boldsymbol{x}}_{USV}(t) + 2\dot{\boldsymbol{x}}_{USV}^{P1} + 2\dot{\boldsymbol{x}}_{USV}^{P2} + \dot{\boldsymbol{x}}_{USV}^{P3} \right). \tag{3.53}$$

In order to use numerical integration with fixed time step h, it is possible to rewrite continuous states transition (3.44) in term of fixed step k as follows

$$\boldsymbol{x}_{USV}(k+1) = \boldsymbol{f}_d(\boldsymbol{x}_{USV}(k), k). \tag{3.54}$$

3.3 Linear USV model

The nonlinear USV state space model defined in (3.44) can be linearized if the following assumptions are considered [14]. Firstly, the roll angle ϕ and pitch angle θ are assumed to be small. This assumption holds for a USV whose roll and pitch motions are limited. The (3.42) can be reformulated using first assumption as

$$\dot{\boldsymbol{\eta}} = \boldsymbol{J}(\boldsymbol{\eta})\boldsymbol{\nu} \overset{\phi=\theta=0}{\approx} \boldsymbol{J}_\psi(\psi)\boldsymbol{\nu}, \tag{3.55}$$

where transformation matrix $\boldsymbol{J}_\psi(\psi)$ is defined as

$$\boldsymbol{J}_\psi(\psi) = \begin{pmatrix} \boldsymbol{R}_\psi & \boldsymbol{O}_{3\times3} \\ \boldsymbol{O}_{3\times3} & \boldsymbol{I}_{3\times3} \end{pmatrix}. \tag{3.56}$$

Using (3.55), the Vessel parallel coordinate system is defined according to [14] as follows

$$\boldsymbol{\eta}_L = \boldsymbol{J}_\psi^T(\psi)\boldsymbol{\eta}, \tag{3.57}$$

where $\boldsymbol{\eta}_L$ denotes the position and orientation in global coordinate frame expressed in body-fixed coordinate frame and $\boldsymbol{J}_\psi^T(\psi)\boldsymbol{J}_\psi(\psi) = \boldsymbol{I}_{6\times6}$. According to the [14], the time derivative of $\boldsymbol{\eta}_L$ can be expressed as

$$\dot{\boldsymbol{\eta}}_L = \dot{\boldsymbol{J}}_\psi^T(\psi)\boldsymbol{\eta} + \boldsymbol{J}_\psi^T(\psi)\dot{\boldsymbol{\eta}}. \tag{3.58}$$

After substitution of term $\boldsymbol{\eta} = \boldsymbol{J}_\psi(\psi)\boldsymbol{\eta}_L$ and $\dot{\boldsymbol{\eta}} \approx \boldsymbol{J}_\psi(\psi)\boldsymbol{\nu}$, the equation (3.58) becomes

$$\dot{\boldsymbol{\eta}}_L = \dot{\boldsymbol{J}}_\psi^T(\psi)\boldsymbol{J}_\psi(\psi)\boldsymbol{\eta}_L + \boldsymbol{J}_\psi^T(\psi)\boldsymbol{J}_\psi(\psi)\boldsymbol{\nu} = r\boldsymbol{S}\boldsymbol{\eta}_L + \boldsymbol{\nu}, \tag{3.59}$$

where r is yaw angular velocity and

$$\boldsymbol{S} = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}. \tag{3.60}$$

Considering $r \approx 0$ [14], the equation (3.59) can be written as follows

$$\dot{\boldsymbol{\eta}}_L \approx \boldsymbol{\nu}. \tag{3.61}$$

The term $\boldsymbol{G\eta}$ (3.40) representing the gravitational and buoyancy forces might also be expressed using Vessel parallel coordinate system according to [14] as

$$\boldsymbol{G\eta} \overset{\phi=\theta=0}{\approx} \boldsymbol{G\eta}_L. \tag{3.62}$$

The nonlinear damping term $\boldsymbol{D}(\boldsymbol{\nu})$ can be converted to a linear form as presented in (3.38). The last nonlinear term represents Coriolis and centripetal forces $\boldsymbol{C}(\boldsymbol{\nu})$. Assuming that $\phi = \theta = 0$ and that in low-speed applications $\boldsymbol{\nu} \approx \boldsymbol{0}$ [14], the term $\boldsymbol{C}(\boldsymbol{\nu})$ becomes zero matrix

$\boldsymbol{C}(\boldsymbol{\nu}) = \boldsymbol{O}_{6\times6}$. Finally, the nonlinear state space equations (3.42) and (3.43) is transformed into a linear form as follows

$$\dot{\boldsymbol{\eta}}_L = \boldsymbol{\nu}, \tag{3.63}$$

$$\dot{\boldsymbol{\nu}} = -\boldsymbol{M}^{-1}\boldsymbol{D}\boldsymbol{\nu} - \boldsymbol{M}^{-1}\boldsymbol{G}\boldsymbol{\eta}_L. \tag{3.64}$$

The LTI state space model is expressed in matrix form as

$$\dot{\boldsymbol{x}}_{VP} = \boldsymbol{A}_{VP}\boldsymbol{x}_{VP}, \tag{3.65}$$

where $\boldsymbol{x}_{VP} = (\boldsymbol{\eta}_L^T,\ \boldsymbol{\nu}^T)^T$ and

$$\boldsymbol{A}_{VP} = \begin{pmatrix} \boldsymbol{O}_{6\times6} & \boldsymbol{I}_{6\times6} \\ -\boldsymbol{M}^{-1}\boldsymbol{G} & -\boldsymbol{M}^{-1}\boldsymbol{D} \end{pmatrix}. \tag{3.66}$$

The global position $\boldsymbol{\eta}$ is computed from $\boldsymbol{\eta}_L$ as

$$\boldsymbol{\eta} = \boldsymbol{J}_\psi(\psi)\boldsymbol{\eta}_L. \tag{3.67}$$

3.4 Wave model

The motion of the USV is significantly influenced by the wave forces. The wave forces are separated into two groups according to their effects on the USV motion [14]. The first group is called First-order wave-induced forces. The results of their action are oscillatory motions with zero mean. The second group is called Second-order wave-induced forces, which represents wave drift forces. These forces are modeled as nonzero components that slowly vary in time.

The characteristics of sea waves are captured in their spectrum $S(\omega_k)$. There exist many spectrum models that are used to derive wave state-space models or wave filtering [14]. The well-known wave spectra are the Neumann spectrum, Bretschneider spectrum, Pierson–Moskowitz spectrum, and JONSWAP spectrum. The amplitude A_k of wave component k is related to wave spectrum $S(\omega_k)$ as

$$\frac{1}{2}A_k^2 = S(\omega_k)\Delta\omega_k, \tag{3.68}$$

where $\Delta\omega_k$ denotes a constant difference between the frequencies of component k and $k-1$. The wave elevation can be expressed as a sum of N harmonic components

$$\zeta = \sum_{k=1}^{N} A_k \cos(\omega_k + \epsilon_k), \tag{3.69}$$

where ϵ_k is a random phase angle of wave component k.

To use wave elevation in the nonlinear USV model, a system generating harmonic signal as one wave component is needed. Inspired by a model described in [79], the system of one wave component is designed as

$$\dot{x}_{\omega_{N1}} = x_{\omega_{N2}}, \tag{3.70}$$

$$\dot{x}_{\omega_{N2}} = -x_{\omega_{N3}}\sin(x_{\omega_{N1}}) - \gamma x_{\omega_{N2}}, \tag{3.71}$$

$$\dot{x}_{\omega_{N3}} = 0, \tag{3.72}$$

$$y_{\omega_N} = x_{\omega_{N2}}, \tag{3.73}$$

where $x_{\omega_{N1}}$, $x_{\omega_{N2}}$ and $x_{\omega_{N3}}$ are state variables of the system, y_{ω_N} is an output signal of one wave component, γ is a damping term of the wave component. The state $x_{\omega_{N3}}$ corresponds to the frequency of the wave component that does not evolve in time. However, the state $x_{\omega_{N3}}$ can be updated in the Kalman filter like any other state. A prior knowledge about the frequency of the wave component can be used to initialize the state $x_{\omega_{N3}}$. The system defined by equations (3.70)–(3.73) can be expressed using $\boldsymbol{x}_{\omega_N} = (x_{\omega_{N1}},\ x_{\omega_{N2}},\ x_{\omega_{N3}})^T$ as

$$\dot{\boldsymbol{x}}_{\omega_N} = \boldsymbol{f}_{\omega_N}(\boldsymbol{x}_{\omega_N}), \tag{3.74}$$

$$y_{\omega_N} = \boldsymbol{g}_{\omega_N}(\boldsymbol{x}_{\omega_N}). \tag{3.75}$$

The N_{nc} individual components y_{ω_N} defined in (3.75) are summed together to obtain complex wave motion

$$\dot{\boldsymbol{x}}_{\omega_{N1}} = \boldsymbol{f}_{\omega_N}(\boldsymbol{x}_{\omega_{N1}}), \tag{3.76}$$

$$y_{\omega_{N1}} = \boldsymbol{g}_{\omega_N}(\boldsymbol{x}_{\omega_{N1}}), \tag{3.77}$$

$$\dot{\boldsymbol{x}}_{\omega_{N2}} = \boldsymbol{f}_{\omega_N}(\boldsymbol{x}_{\omega_{N2}}), \tag{3.78}$$

$$y_{\omega_{N2}} = \boldsymbol{g}_{\omega_N}(\boldsymbol{x}_{\omega_{N2}}), \tag{3.79}$$

$$\vdots$$

$$\dot{\boldsymbol{x}}_{\omega_{NN_{nc}}} = \boldsymbol{f}_{\omega_N}(\boldsymbol{x}_{\omega_{NN_{nc}}}), \tag{3.80}$$

$$y_{\omega_{NN_{nc}}} = \boldsymbol{g}_{\omega_N}(\boldsymbol{x}_{\omega_{NN_{nc}}}), \tag{3.81}$$

$$y_{wave} = y_{\omega_{N1}} + y_{\omega_{N2}} + \ldots + y_{\omega_{NN_{nc}}}, \tag{3.82}$$

which can be simplified using $\boldsymbol{x}_{wave} = (\boldsymbol{x}^T_{\omega_{N1}}, \ldots,\ \boldsymbol{x}^T_{\omega_{NN_{nc}}})^T$ into

$$\dot{\boldsymbol{x}}_{wave} = \boldsymbol{f}_{wave}(\boldsymbol{x}_{wave}), \tag{3.83}$$

$$y_{wave} = \boldsymbol{g}_{wave}(\boldsymbol{x}_{wave}). \tag{3.84}$$

The y_{wave} (3.84) is used to create new state vector $\boldsymbol{\nu}_{wave}$ as

$$\boldsymbol{\nu}_{wave} = (y_{wave,u},\ y_{wave,v},\ y_{wave,w},\ y_{wave,p},\ y_{wave,q},\ y_{wave,r})^T, \tag{3.85}$$

where each element $y_{wave,u}$, $y_{wave,v}$, $y_{wave,w}$, $y_{wave,p}$, $y_{wave,q}$, $y_{wave,r}$ corresponds to a one wave system defined in (3.83) and (3.84). The nonlinear model of the USV (3.42) and (3.43) is rewritten as

$$\dot{\boldsymbol{\eta}} = \boldsymbol{J}(\boldsymbol{\eta})\boldsymbol{\nu}, \tag{3.86}$$

$$\dot{\boldsymbol{\nu}} = \boldsymbol{M}^{-1}\left(-\boldsymbol{C}(\boldsymbol{\nu})\boldsymbol{\nu} - \boldsymbol{D}(\boldsymbol{\nu})\boldsymbol{\nu} - \boldsymbol{G}\boldsymbol{\eta})\right) + \boldsymbol{\nu}_{wave}, \tag{3.87}$$

$$\dot{\boldsymbol{x}}_{wave,u} = \boldsymbol{f}_{wave}(\boldsymbol{x}_{wave,u}), \tag{3.88}$$

$$\dot{\boldsymbol{x}}_{wave,v} = \boldsymbol{f}_{wave}(\boldsymbol{x}_{wave,v}), \tag{3.89}$$

$$\dot{\boldsymbol{x}}_{wave,w} = \boldsymbol{f}_{wave}(\boldsymbol{x}_{wave,w}), \tag{3.90}$$

$$\dot{\boldsymbol{x}}_{wave,p} = \boldsymbol{f}_{wave}(\boldsymbol{x}_{wave,p}), \tag{3.91}$$

$$\dot{\boldsymbol{x}}_{wave,q} = \boldsymbol{f}_{wave}(\boldsymbol{x}_{wave,q}), \tag{3.92}$$

$$\dot{\boldsymbol{x}}_{wave,r} = \boldsymbol{f}_{wave}(\boldsymbol{x}_{wave,r}). \tag{3.93}$$

The wave system defined in (3.83) and (3.84) cannot be used for the linear USV model as the wave system is nonlinear. However, the waves can also be represented by linear state-space model [14]. The simple state-space model of a one wave component including two states $x_{\omega L1},\ x_{\omega L2}$ is as follows

$$\begin{pmatrix} \dot{x}_{\omega L1} \\ \dot{x}_{\omega L2} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -\omega_{0_L}^2 & -2\lambda_L \omega_{0_L} \end{pmatrix} \begin{pmatrix} x_{\omega L1} \\ x_{\omega L2} \end{pmatrix}, \tag{3.94}$$

$$y_{\omega_L} = \begin{pmatrix} 0 & 1 \end{pmatrix} \begin{pmatrix} x_{\omega L1} \\ x_{\omega L2} \end{pmatrix}, \tag{3.95}$$

where ω_{0_L} represents the frequency of the wave component and λ_L is the damping of the wave component. The wave component defined in (3.94) and (3.95) is expressed in matrix form as

$$\dot{\boldsymbol{x}}_{\omega_L} = \boldsymbol{A}_{\omega_L} \boldsymbol{x}_{\omega_L}, \tag{3.96}$$

$$y_{\omega_L} = \boldsymbol{C}_{\omega_L} \boldsymbol{x}_{\omega_L}. \tag{3.97}$$

To achieve complex wave motion composed of several harmonics, the N_{lc} state-space wave components y_{ω_L} can be joined together with different parameters ω_{0_L} and λ_L

$$\dot{\boldsymbol{x}}_{\omega L1} = \boldsymbol{A}_{\omega L1} \boldsymbol{x}_{\omega L1}, \tag{3.98}$$

$$y_{\omega L1} = \boldsymbol{C}_{\omega L1} \boldsymbol{x}_{\omega L1}, \tag{3.99}$$

$$\dot{\boldsymbol{x}}_{\omega L2} = \boldsymbol{A}_{\omega L2} \boldsymbol{x}_{\omega L2}, \tag{3.100}$$

$$y_{\omega L2} = \boldsymbol{C}_{\omega L2} \boldsymbol{x}_{\omega L2}, \tag{3.101}$$

$$\vdots$$

$$\dot{\boldsymbol{x}}_{\omega LN_{lc}} = \boldsymbol{A}_{\omega LN_{lc}} \boldsymbol{x}_{\omega LN_{lc}}, \tag{3.102}$$

$$y_{\omega LN_{lc}} = \boldsymbol{C}_{\omega LN_{lc}} \boldsymbol{x}_{\omega LN_{lc}}, \tag{3.103}$$

$$y_{\text{wave}_L} = y_{\omega L1} + y_{\omega L2} + \ldots + y_{\omega LN_{lc}}. \tag{3.104}$$

The system described in (3.98)–(3.104) can be expressed as a one linear system

$$\dot{\boldsymbol{x}}_{wave_L} = \boldsymbol{A}_{wave_L} \boldsymbol{x}_{wave_L}, \tag{3.105}$$

$$y_{\text{wave}_L} = \boldsymbol{C}_{wave_L} \boldsymbol{x}_{wave_L}, \tag{3.106}$$

where

$$\boldsymbol{x}_{wave_L} = (\boldsymbol{x}_{\omega L1}^T,\ \boldsymbol{x}_{\omega L2}^T, \ldots,\ \boldsymbol{x}_{\omega LN_{lc}}^T)^T, \tag{3.107}$$

$$\boldsymbol{A}_{wave_L} = \operatorname{diag}\{\boldsymbol{A}_{\omega L1},\ \boldsymbol{A}_{\omega L2}, \ldots,\ \boldsymbol{A}_{\omega LN_{lc}}\}, \tag{3.108}$$

$$\boldsymbol{C}_{wave_L} = \begin{pmatrix} \boldsymbol{C}_{\omega L1} & \boldsymbol{C}_{\omega L2} & \cdots & \boldsymbol{C}_{\omega LN_{lc}} \end{pmatrix}, \tag{3.109}$$

where diag$\{\cdot\}$ means a block diagonal matrix of given elements $\boldsymbol{A}_{\omega L1},\ \boldsymbol{A}_{\omega L2}, \ldots,\ \boldsymbol{A}_{\omega LN_{lc}}$.

The one wave system in (3.105) and (3.106) influences one USV state of $\boldsymbol{\nu}$ (3.2). Therefore, the complex wave system for the USV states $\boldsymbol{\nu}$ is defined as

$$\dot{\boldsymbol{x}}_{wave,\boldsymbol{\nu}} = \boldsymbol{A}_{wave,\boldsymbol{\nu}} \boldsymbol{x}_{wave,\boldsymbol{\nu}}, \tag{3.110}$$

$$\boldsymbol{y}_{wave,\boldsymbol{\nu}} = \boldsymbol{C}_{wave,\boldsymbol{\nu}} \boldsymbol{x}_{wave,\boldsymbol{\nu}}, \tag{3.111}$$

$$\boldsymbol{A}_{wave,\boldsymbol{\nu}} = \mathrm{diag}\{\boldsymbol{A}_{wave_L},\ \boldsymbol{A}_{wave_L},\ \boldsymbol{A}_{wave_L},\ \boldsymbol{A}_{wave_L},\ \boldsymbol{A}_{wave_L},\ \boldsymbol{A}_{wave_L}\}, \tag{3.112}$$

$$\boldsymbol{C}_{wave,\boldsymbol{\nu}} = \begin{pmatrix} \boldsymbol{C}_{wave_L} & \boldsymbol{O}_{1\times 2N_{lc}} & \boldsymbol{O}_{1\times 2N_{lc}} & \boldsymbol{O}_{1\times 2N_{lc}} & \boldsymbol{O}_{1\times 2N_{lc}} & \boldsymbol{O}_{1\times 2N_{lc}} \\ \boldsymbol{O}_{1\times 2N_{lc}} & \boldsymbol{C}_{wave_L} & \boldsymbol{O}_{1\times 2N_{lc}} & \boldsymbol{O}_{1\times 2N_{lc}} & \boldsymbol{O}_{1\times 2N_{lc}} & \boldsymbol{O}_{1\times 2N_{lc}} \\ \boldsymbol{O}_{1\times 2N_{lc}} & \boldsymbol{O}_{1\times 2N_{lc}} & \boldsymbol{C}_{wave_L} & \boldsymbol{O}_{1\times 2N_{lc}} & \boldsymbol{O}_{1\times 2N_{lc}} & \boldsymbol{O}_{1\times 2N_{lc}} \\ \boldsymbol{O}_{1\times 2N_{lc}} & \boldsymbol{O}_{1\times 2N_{lc}} & \boldsymbol{O}_{1\times 2N_{lc}} & \boldsymbol{C}_{wave_L} & \boldsymbol{O}_{1\times 2N_{lc}} & \boldsymbol{O}_{1\times 2N_{lc}} \\ \boldsymbol{O}_{1\times 2N_{lc}} & \boldsymbol{O}_{1\times 2N_{lc}} & \boldsymbol{O}_{1\times 2N_{lc}} & \boldsymbol{O}_{1\times 2N_{lc}} & \boldsymbol{C}_{wave_L} & \boldsymbol{O}_{1\times 2N_{lc}} \\ \boldsymbol{O}_{1\times 2N_{lc}} & \boldsymbol{O}_{1\times 2N_{lc}} & \boldsymbol{O}_{1\times 2N_{lc}} & \boldsymbol{O}_{1\times 2N_{lc}} & \boldsymbol{O}_{1\times 2N_{lc}} & \boldsymbol{C}_{wave_L} \end{pmatrix}, \tag{3.113}$$

where $\boldsymbol{A}_{wave,\boldsymbol{\nu}}$ is a block diagonal matrix of this complex wave system and

$$\boldsymbol{x}_{wave,\boldsymbol{\nu}} = (\boldsymbol{x}^T_{wave_L,u},\ \boldsymbol{x}^T_{wave_L,v},\ \boldsymbol{x}^T_{wave_L,w},\ \boldsymbol{x}^T_{wave_L,p},\ \boldsymbol{x}^T_{wave_L,q},\ \boldsymbol{x}^T_{wave_L,r})^T. \tag{3.114}$$

A linear model of the USV (3.65) is then modified as

$$\dot{\boldsymbol{x}}_{VP,waves} = \boldsymbol{A}_{VP,waves} \boldsymbol{x}_{VP,waves}, \tag{3.115}$$

where $\boldsymbol{x}_{VP,waves} = (\boldsymbol{\eta}_L^T,\ \boldsymbol{\nu}^T,\ \boldsymbol{x}^T_{wave,\boldsymbol{\nu}})^T$ and

$$\boldsymbol{A}_{VP,waves} = \begin{pmatrix} \boldsymbol{O}_{6\times 6} & \boldsymbol{I}_{6\times 6} & \boldsymbol{O}_{6\times 12N_{lc}} \\ -\boldsymbol{M}^{-1}\boldsymbol{G} & -\boldsymbol{M}^{-1}\boldsymbol{D} & \boldsymbol{C}_{wave,\boldsymbol{\nu}} \\ \boldsymbol{O}_{12N_{lc}\times 6} & \boldsymbol{O}_{12N_{lc}\times 6} & \boldsymbol{A}_{wave,\boldsymbol{\nu}} \end{pmatrix}. \tag{3.116}$$

Chapter 4

Sensors

The main goal of this thesis is to estimate the USV states. The crucial component for a state estimator is to have enough sensors whose measurement data will cover all parts of the USV model (Chap. 3). The Kalman filter is selected as a state estimator (Chap. 2). In order to apply the filter, the observation model that defines mapping from the USV states to the measurements for individual sensors has to be defined together with its measurement covariance [80]. The sensors used to estimate the USV states are presented in this chapter. The sensors are divided into two groups. The first one contains sensors that are directly placed on the USV. The UAV onboard sensors form the second group.

4.1 Sensors placed on USV

In many real-world scenarios, the UAV is at such a distance that it is not possible to use the onboard sensors. However, it is assumed that the communication link between the UAV and the USV is present. As our objective is autonomous UAV landing on the USV, it is necessary to get closer to the USV where the UAV onboard sensors can be used together with the sensors placed on the USV. To move the UAV towards the USV, at least the raw position of the USV has to be estimated. The GPS sensor [81] has been selected as a device that measures the global position of the USV which is subsequently sent to the UAV.

Another sensor that is placed on the USV is the IMU. The IMU is a device used to measure orientation, angular velocity, and linear acceleration [82]. Typically IMU consists of an accelerometer and a gyroscope. However, a magnetometer can be added to obtain more accurate measurement [82]. The IMU and the GPS are part of the Multi-robot Systems (MRS) boat unit that contains necessary electronic equipment for the landing platform. A picture of the landing platform is shown in Fig. 4.1.

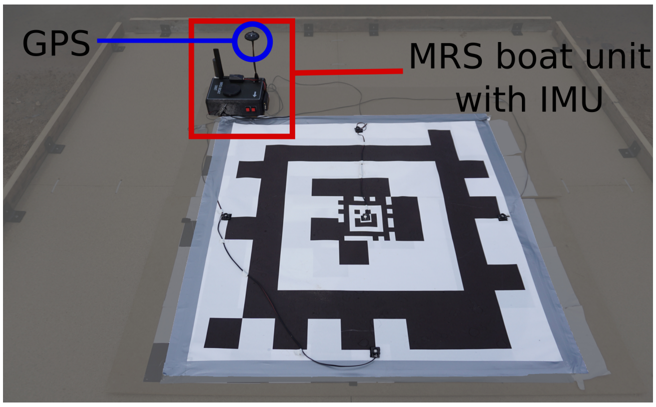


Figure 4.1: The GPS and the IMU placed on landing platform.

4.1.1 GPS

The GPS is a type of GNSS that is well-known and widely used to estimate the position in several application scenarios [39], [47], [48], [83]–[86]. The usage of GPS is motivated by an estimation of the USV and the UAV position in common frame. The global position obtained from the GPS device placed on the USV is sent to the UAV via a wireless communication link. The UAV uses this received data to estimate the position of the USV, and the control system takes the UAV to the proximity of the USV, where onboard sensors are used to update estimate to be more precise (Sec. 4.2).

The GPS device is a standard outdoor sensor that needs an open area without obstacles disturbing its GPS signal to estimate the position properly. Water surface typically satisfies these requirements. Therefore usage of the GPS is reasonable on the USV moving on water surface [45], [46]. However, the GPS signal can still be influenced by natural interference such as ionospheric anomalies, and intentional attacks such as jamming and spoofing [87]. These situations can significantly reduce the accuracy of the GPS position estimation or cause the GPS data to be completely unavailable The error of the GPS position estimation is approximately in meters [81] which is insufficient to estimate the USV states precisely for a landing approach. In order to obtain better accuracy, the Real-time Kinematics (RTK)-GPS method can be used [88]. However, this method requires a calibrated base station and a reliable GPS signal.

The UAV uses the received GPS data $\boldsymbol{y}_{GPS}$ from the USV containing Universal Transverse Mercator (UTM) coordinates in the data-update step of Kalman filters (Chap. 2)

$$\boldsymbol{y}_{GPS} = (x_{UTM},\ y_{UTM},\ z_{UTM})^T. \tag{4.1}$$

To perform the data-update step in the Linear Kalman filter (Sec. 2.1) based on the linear USV model (Sec. 3.3 and 3.4), the sensor matrix $\boldsymbol{C}_{GPS}$ for GPS data is as follows

$$\boldsymbol{C}_{GPS} = \begin{pmatrix} 1 & 0 & 0 & \boldsymbol{O}_{1\times(9+12N_{lc})} \\ 0 & 1 & 0 & \boldsymbol{O}_{1\times(9+12N_{lc})} \\ 0 & 0 & 1 & \boldsymbol{O}_{1\times(9+12N_{lc})} \end{pmatrix}. \tag{4.2}$$

The Unscented Kalman filter (Sec. 2.3) requires the function $\boldsymbol{h}_{GPS}(\boldsymbol{x}_{USV})$ that defines model of sensor for the nonlinear model of USV (Sec. 3.2 and 3.4) to provide the data-update step of the filter

$$\boldsymbol{h}_{GPS}(\boldsymbol{x}_{USV}) = \boldsymbol{p} = (x,\ y,\ z)^T. \tag{4.3}$$

4.1.2 Inertial Measurement Unit

The IMU provides the linear acceleration, angular rotation, and orientation data in high update rates [89]. Similar to the GPS data, the IMU data is sent to the UAV via a wireless communication link. The motivation to add the IMU sensor on the USV landing platform is to obtain a better estimation of motion caused by the waves (Sec. 3.4). Another reason is that the IMU and the GPS can be used together to estimate more accurate states of the system [89], [90]. However, the IMU suffers from time accumulating of bias in the sensor readings [89].

The IMU data $\boldsymbol{y}_{IMU}$ used in the Kalman filters is in the following form

$$\boldsymbol{y}_{IMU} = (\phi_{IMU},\ \theta_{IMU},\ \psi_{IMU},\ p_{IMU},\ q_{IMU},\ r_{IMU})^T, \tag{4.4}$$

where ϕ_{IMU}, θ_{IMU} and ψ_{IMU} are the Euler angles corresponding to states ϕ, θ, ψ of the USV. Variables p_{IMU}, q_{IMU} and r_{IMU} are the angular velocities in case of the USV states p, q, r. The sensor model is needed to use Kalman filters presented in Chap. 2. Using the linear model of USV (Sec. 3.3 and 3.4) the Linear Kalman filter (Sec. 2.1) applies IMU measurement data in the data-update step via sensor matrix $\boldsymbol{C}_{IMU}$

$$\boldsymbol{C}_{IMU} = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \boldsymbol{O}_{1\times(12N_{lc})} \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \boldsymbol{O}_{1\times(12N_{lc})} \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & \boldsymbol{O}_{1\times(12N_{lc})} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & \boldsymbol{O}_{1\times(12N_{lc})} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & \boldsymbol{O}_{1\times(12N_{lc})} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & \boldsymbol{O}_{1\times(12N_{lc})} \end{pmatrix}. \tag{4.5}$$

The Unscented Kalman filter (Sec. 2.3) needs to have the sensor model function $\boldsymbol{h}_{IMU}(\boldsymbol{x}_{USV})$ of the nonlinear USV model (Sec. 3.2 and 3.4) defined in order to provide the data-update step. For the IMU measurement data, the function $\boldsymbol{h}_{IMU}(\boldsymbol{x}_{USV})$ is defined as follows

$$\boldsymbol{h}_{IMU}(\boldsymbol{x}_{USV}) = (\boldsymbol{\Theta}^T,\ \boldsymbol{\omega}^T)^T = (\phi,\ \theta,\ \psi,\ p,\ q,\ r)^T. \tag{4.6}$$

4.2 UAV onboard sensors

In Sec. 4.1 sensors placed on the USV are presented. The data from these sensors have to be sent to the UAV via a communication link. However, data exchange between the USV and the UAV can be difficult to achieve in real-world environments [49]. Also, precision of the GPS sensor is insufficient to perfectly estimate the USV states that have to be used to land on the USV [81]. The precise GPS position can be obtained under a special system, e.g. RTK-GPS [88] which requires a reliable GPS signal and calibrated base station.

To handle these issues, the UAV is equipped with onboard vision systems that allow estimation of the USV states without the need for communication or common reference frame of the USV and the UAV. To increase the redundancy and ensure a properly working system in different real-world conditions, two onboard sensors are used: an UltraViolet Direction And Ranging (UVDAR) system [23], [53], [91], [92] and an AprilTag detector [50]–[52]. Both the vision systems require special markers placed on the target. Therefore a custom landing platform was designed to satisfy these preconditions. The description of both systems is given in the following subsections (Sec. 4.2.1 and 4.2.2).

4.2.1 UVDAR system

The UVDAR system is an onboard vision-based relative localization system developed by the MRS group. One of the great advantages of this system is that no communication is needed. As mentioned above, the UVDAR system requires markers placed on the target. The marker is composed of an UltraViolet (UV) LED that blinks a unique binary signal code. The blinking UV LEDs are captured by UV sensitive cameras (Fig. 4.2) placed onboard on the UAV.

The principle of the UVDAR system is presented in detail in [23], [53], [91], [92]. The onboard UV sensitive cameras take images of the surrounding area. The captured images are passed to the algorithm that extracts the blinking UV markers, whereas the targets are

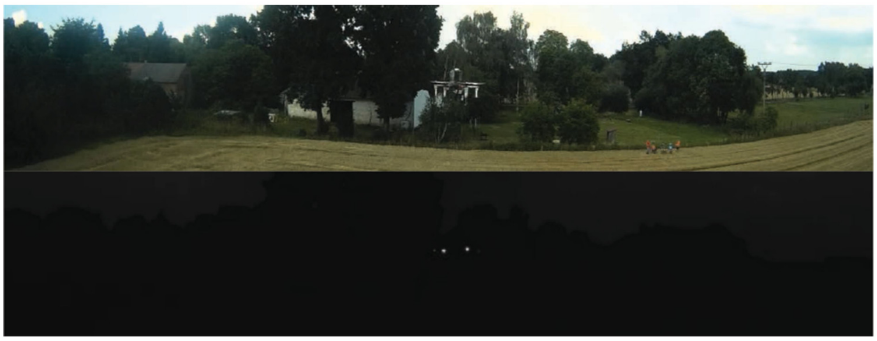


Figure 4.2: Comparison between the visible and UV camera footage from UVDAR, collected during an experiment. The UV image is significantly easier to process to retrieve information on the observed MAV [91, p. 2638].

distinguished by the binary signal code of the UV LEDs and a pattern created from these LEDs. The position and orientation estimation rely on knowledge of the exact location of the UV LEDs on the target. Extracted positions of the UV markers from an image are used to compute the final position and orientation of the target with respect to onboard UV cameras [23], [91].

Typical usage of the UVDAR system is for mutual localization of UAVs in a swarm [93]–[95]. The layout of UV LEDs is defined according to a UAV type. However, the UVDAR system used to detect the horizontal landing platform is a novel application of this system. For this purpose, the novel pattern of UV LEDs layout has to be defined. The five UV LEDs form a plus sign on the landing platform as shown in Fig. 4.3. In order to unambiguously define the orientation of UV LEDs pattern with respect to UV cameras, the two UV LEDs have different signal codes than the others. The distance between UV LEDs on the edge and the middle UV LED is 0.4 m.

The data $\boldsymbol{y}_{UR}$ obtained from the UVDAR system contains position $(x_{UR},\ y_{UR},\ z_{UR})$ and orientation $(\phi_{UR},\ \theta_{UR},\ \psi_{UR})$

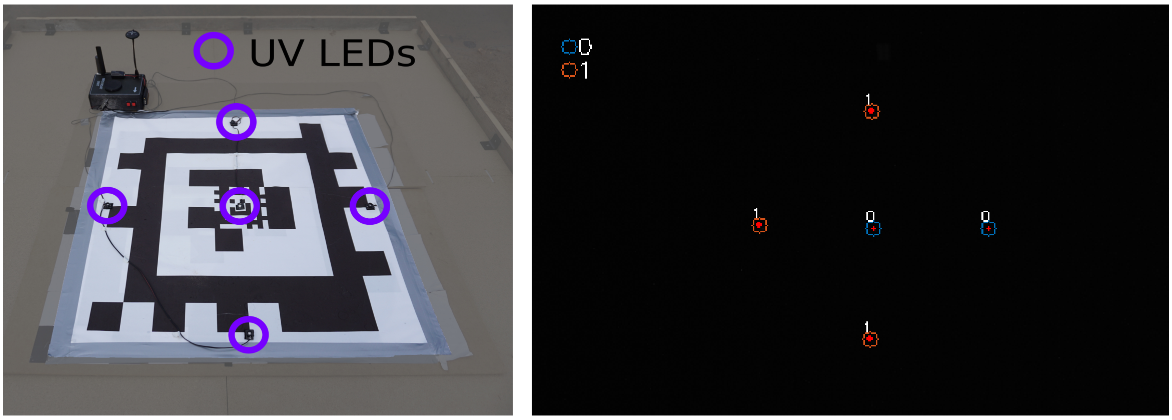
$$\boldsymbol{y}_{UR} = (x_{UR},\ y_{UR},\ z_{UR},\ \phi_{UR},\ \theta_{UR},\ \psi_{UR})^T. \tag{4.7}$$

To use the data measured by the UVDAR system, the sensor model has to be defined for a desired Kalman filter described in Chap. 2. For the linear model of the USV (Sec. 3.3 and 3.4), the sensor matrix $\boldsymbol{C}_{UR}$ for the Linear Kalman filter (Sec. 2.1) is defined as follows

$$\boldsymbol{C}_{UR} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & \boldsymbol{O}_{1\times(6+12N_{lc})} \\ 0 & 1 & 0 & 0 & 0 & 0 & \boldsymbol{O}_{1\times(6+12N_{lc})} \\ 0 & 0 & 1 & 0 & 0 & 0 & \boldsymbol{O}_{1\times(6+12N_{lc})} \\ 0 & 0 & 0 & 1 & 0 & 0 & \boldsymbol{O}_{1\times(6+12N_{lc})} \\ 0 & 0 & 0 & 0 & 1 & 0 & \boldsymbol{O}_{1\times(6+12N_{lc})} \\ 0 & 0 & 0 & 0 & 0 & 1 & \boldsymbol{O}_{1\times(6+12N_{lc})} \end{pmatrix}. \tag{4.8}$$

To use measured data in the data-update step of the Unscented Kalman filter (Sec. 2.3) based on the nonlinear model of USV (Sec. 3.2 and 3.4), the function $\boldsymbol{h}_{UR}(\boldsymbol{x}_{USV})$ that defines model of sensor has the following form

$$\boldsymbol{h}_{UR}(\boldsymbol{x}_{USV}) = (\boldsymbol{p}^T,\ \boldsymbol{\Theta}^T)^T = (x,\ y,\ z,\ \phi,\ \theta,\ \psi)^T. \tag{4.9}$$



(a) UV LEDs placed on the landing platform.

(b) Detected pattern in UV camera image.

Figure 4.3: UV LEDs pattern placed on the landing platform together with detected UV LEDs in UV camera image.

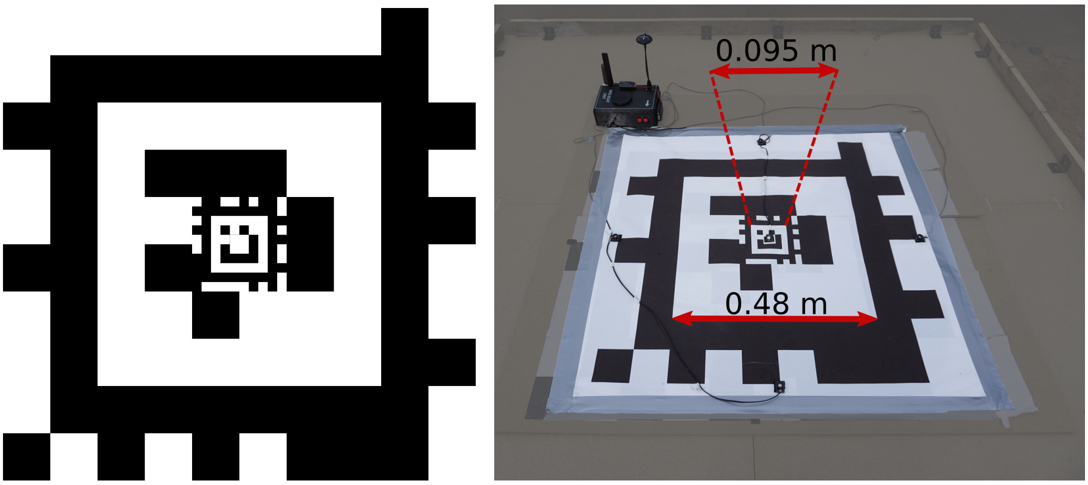
4.2.2 AprilTag detector

The AprilTag detector is a visual fiducial system that detects artificial landmarks known as AprilTags [50], [51]. The AprilTags are black-and-white tags defined by a binary code that can have flexible layouts if some restrictions are satisfied [52]. The AprilTag detector consists of the tag detector and the coding system. The purpose of the tag detector is to find candidates for AprilTags in the image. Found candidates for AprilTags are then put into the coding system, which determines whether the candidates are valid or not, and finally, positions and orientations of the detected AprilTags are estimated [50], [51].

A camera taking images of surroundings is needed to use the AprilTag detector. The taken image is converted to a binary image using an adaptive threshold [51]. The segmentation algorithm is applied to find connected black and white components to find edges in the binary images. Then the quads are fitted to boundary points of the segmented components and put to the decoding algorithm as candidates for AprilTags. The decoding algorithm selects valid candidates for AprilTags. Finally, the position and orientation of detected a AprilTag is estimated using the knowledge of AprilTag binary code, and its size [50], [51].

One of the advantages of using the AprilTag detector is a flexible layout of its markers [52]. The custom tag layout is a special AprilTag layout containing empty space in the middle, which allows to recursively put a smaller tag inside. This configuration of an AprilTag is suitable to localize the landing spots [44], [52]. However, other flexible tag layouts have been successfully used for landing approaches [59], [96]. The AprilTag layout placed on the landing platform designed in this thesis is shown in Fig. 4.4a. The custom tag layout is used, hence a smaller AprilTag is placed in the empty space of the bigger AprilTag. The size of an AprilTag configuration on real landing platform is shown in Fig. 4.4b. This configuration of an AprilTag allows its detection from short and long distances.

A single camera with a processing unit is sufficient to detect an AprilTag on a target. However, the AprilTags are passive markers, meaning they do not broadcast any signal or emit light. This method is therefore only useful under sufficient image taking light conditions. If the UAV has to land in the dark, the AprilTag detector will not produce a reliable position and orientation estimation. Although the AprilTag detector should be robust to light conditions



(a) AprilTag layout used on the landing platform.

(b) AprilTag placed on the landing platform.

Figure 4.4: AprilTag used on landing platform.

(e.g., the landing platform can be lit) [50], [51], the UVDAR system (Sec. 4.2.1) is able to work properly at any time of day as it uses active blinking markers. This is another reason to use two onboard detection systems since robustness is one of the main goals for the system designed in this thesis.

The data $\boldsymbol{y}_{AG}$ measured by the AprilTag detector consists of position $(x_{AG},\ y_{AG},\ z_{AG})$ and orientation $(\phi_{AG},\ \theta_{AG},\ \psi_{AG})$

$$\boldsymbol{y}_{AG} = (x_{AG},\ y_{AG},\ z_{AG},\ \phi_{AG},\ \theta_{AG},\ \psi_{AG})^T. \tag{4.10}$$

Same as for the previously presented sensors, the sensor model has to be defined to use AprilTag detector measurements. For the linear model of the USV (Sec. 3.3 and 3.4) used in the Linear Kalman filter (Sec. 2.1) the sensor matrix $\boldsymbol{C}_{AG}$ of the AprilTag detector is defined as follows

$$\boldsymbol{C}_{AG} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & \boldsymbol{O}_{1\times(6+12N_{lc})} \\ 0 & 1 & 0 & 0 & 0 & 0 & \boldsymbol{O}_{1\times(6+12N_{lc})} \\ 0 & 0 & 1 & 0 & 0 & 0 & \boldsymbol{O}_{1\times(6+12N_{lc})} \\ 0 & 0 & 0 & 1 & 0 & 0 & \boldsymbol{O}_{1\times(6+12N_{lc})} \\ 0 & 0 & 0 & 0 & 1 & 0 & \boldsymbol{O}_{1\times(6+12N_{lc})} \\ 0 & 0 & 0 & 0 & 0 & 1 & \boldsymbol{O}_{1\times(6+12N_{lc})} \end{pmatrix}. \tag{4.11}$$

The Unscented Kalman filter (Sec. 2.3) using the nonlinear model of the USV (Sec. 3.2 and 3.4) requires the function $\boldsymbol{h}_{AG}(\boldsymbol{x}_{USV})$ which defines model of sensor in the following form

$$\boldsymbol{h}_{AG}(\boldsymbol{x}_{USV}) = (\boldsymbol{p}^T,\ \boldsymbol{\Theta}^T)^T = (x,\ y,\ z,\ \phi,\ \theta,\ \psi)^T. \tag{4.12}$$

Chapter 5

Development tools

5.1 Software platform

The system designed in this thesis is based on a software platform called Robot Operating System (ROS) [97]. The ROS[1] is the free open-source framework that enables writing of modular software for robot applications. The software implemented according to this framework consists of individual packages. A package is a basic unit containing source codes of processes, libraries, configuration files, and other files related to the package.

The concept of ROS framework is to use the following tools: nodes, messages, topics, and services. The computational processes are performed using nodes that typically represent one robotic software module, e.g., one node reads data from a sensor, and the second node fuses and filters this data. In order to share the data between individual nodes, the messages are defined as a strongly-typed data structure. The atomic units of messages are basic types (e.g., integer, float, boolean, etc.) that can be organized into arrays. The new message can also be composed of previously defined individual messages or their arrays.

The communication between nodes is performed via topics or services using defined messages. The node sends data to desired topics specified by name and message type. The other nodes can connect to this topic and receive the available data. Multiple nodes can publish and subscribe to this topic. However, the request-reply model of communication is often needed in real applications. This communication model is ensured via services defined by their name and pair of messages. One message defines a request of the client node. The other message characterizes the reply of the target server node. The ROS structure of the system presented in this thesis is shown as a diagram in Fig. 5.1.

The system designed in this thesis is implemented into the MRS UAV system [1]. This open-source system[2] is based on the ROS framework. The MRS UAV system proposes tools to control and estimate UAVs in various applications scenarios, such as outdoor or indoor flying, GNSS or GNSS-denied environments, etc. The tools enable a realistic simulation of the UAVs, sensors, and localization systems. The MRS UAV system was used in many real-world applications [2], [39], [53], [55], [92]–[95].

5.2 Simulation environment

The initial validation of the proposed USV estimation system was performed using the realistic robotic simulator Gazebo [98]. The Gazebo simulator[3] is an open-source project

[1] https://www.ros.org/

[2] https://github.com/ctu-mrs/mrs_uav_system

[3] http://gazebosim.org/

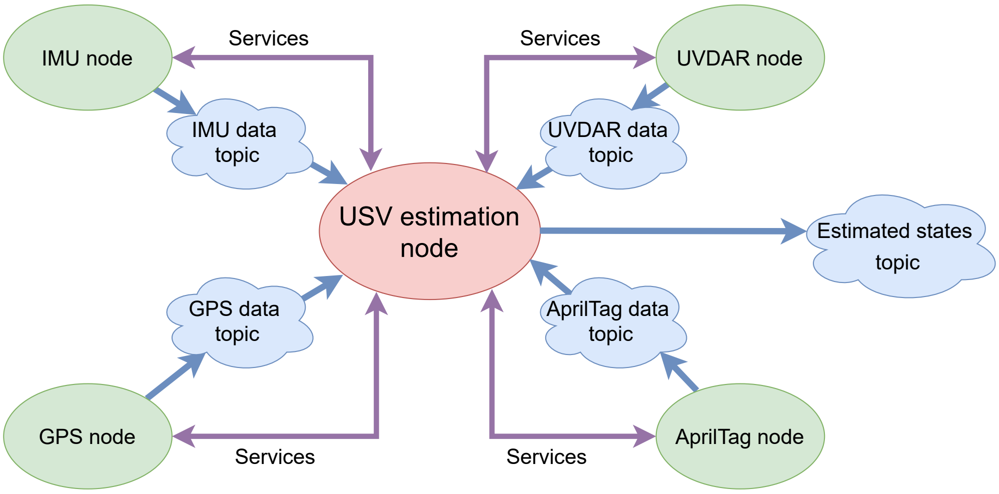


Figure 5.1: Diagram of the system presented in this thesis in the ROS framework.

enabling realistic simulation of robots in different complex environments. The simulator offers multiple high-performance physics engines (e.g., ODE, Bullet), high-quality graphics, and convenient programmatic and graphical interfaces. Both the ROS system and the MRS UAV system are implemented into the Gazebo simulator. In order to simulate the USV on a wavy water surface, the Virtual RobotX (VRX) simulator is used as an extension of the Gazebo simulator.

The VRX simulator[4] is an open-source project designed to support the development, testing, and evaluation of USVs that operate on wavy water surface [99]. The VRX simulator adds new features into the Gazebo simulator, such as a water surface including a wave generator based on different ocean spectra that influences the USV motion. The USV motion is also affected by the wind. To achieve a real USV behavior, the six DOF model of a surface vessel is included that can be controlled via a propulsion system. The screenshots from the Gazebo simulator extended by the VRX simulator and the MRS UAV system are shown in Fig. 5.2.

The USV model in the VRX simulator is based on equations presented in [14]. The USV observer design model presented in Chap. 3 is created on the same basis. However, the model used in the VRX simulator belongs to the simulation models that capture complex behavior of the system to achieve the same response as the real system. The equations are as follows

$$\boldsymbol{M}_{RB}\dot{\boldsymbol{\nu}} + \boldsymbol{C}_{RB}(\boldsymbol{\nu})\boldsymbol{\nu} + \boldsymbol{M}_{A}\dot{\boldsymbol{\nu}}_r + \boldsymbol{C}_{A}(\boldsymbol{\nu}_r)\boldsymbol{\nu}_r + \boldsymbol{D}(\boldsymbol{\nu}_r)\boldsymbol{\nu}_r + \boldsymbol{g}(\boldsymbol{\eta}) = \boldsymbol{\tau}_{propulsion} + \boldsymbol{\tau}_{wind} + \boldsymbol{\tau}_{waves}. \tag{5.1}$$

The relative velocity $\boldsymbol{\nu}_r = \boldsymbol{\nu} - \boldsymbol{\nu}_c$ represents the motion of the USV with respect to the water current velocity $\boldsymbol{\nu}_c$. The $\boldsymbol{\tau}_{propulsion}$ is force generated by the USV propulsion system. The winds and waves forces are denoted as $\boldsymbol{\tau}_{wind}$ and $\boldsymbol{\tau}_{waves}$. The other terms are more thoroughly described in Chap. 3.

The waves are modeled using the Gerstner Waves approach [99]. The approach represents the water surface as a trochoidal shape. The water surface is modeled as a summation

[4] `https://github.com/osrf/vrx`

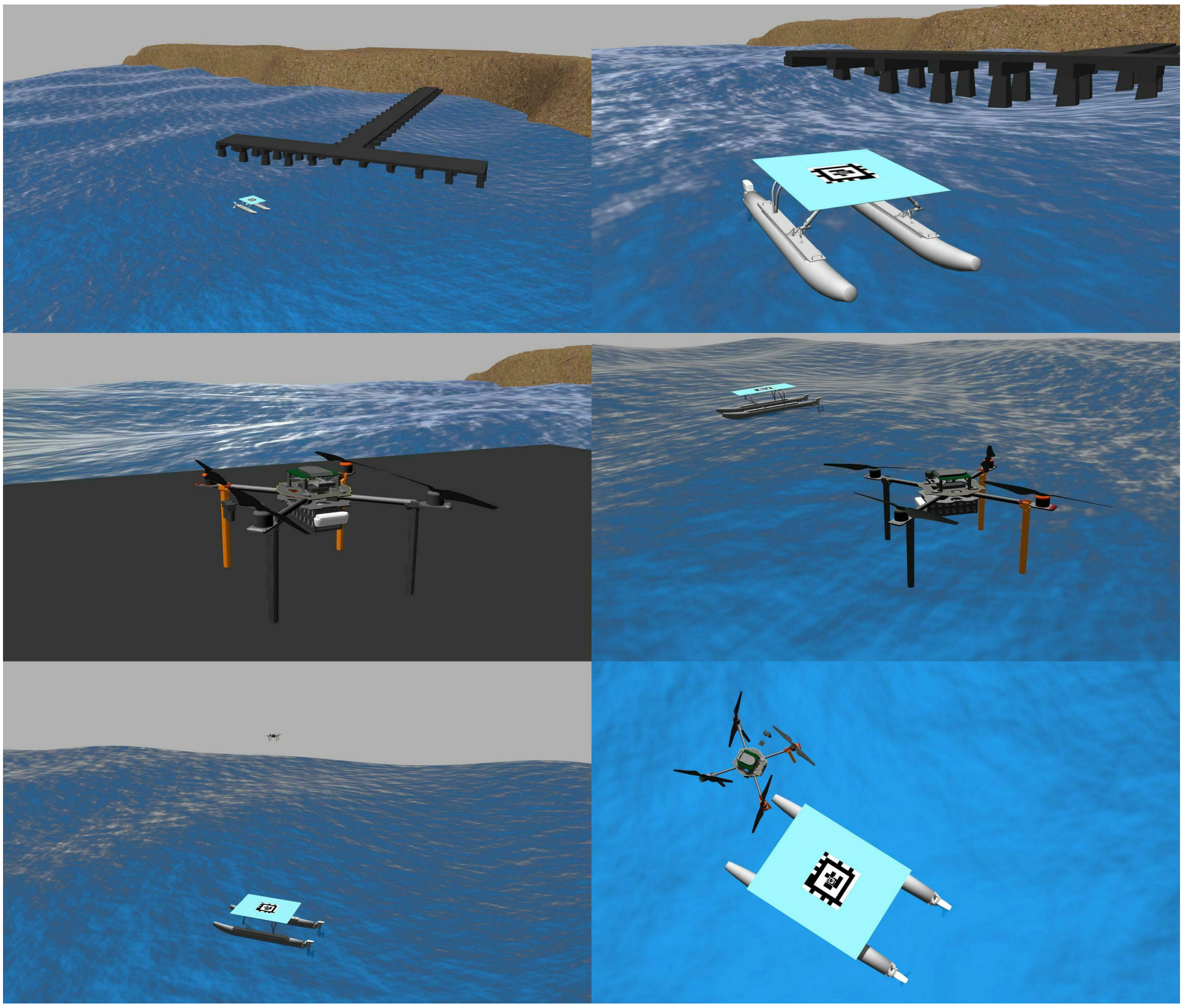


Figure 5.2: The screenshots from the Gazebo simulator extended by the VRX simulator and the MRS UAV system.

of Gerstner waves according to the following equations

$$\boldsymbol{x}(\boldsymbol{x}_0, t) = \boldsymbol{x}_0 - \sum_{i=1}^{N} q_i \frac{\boldsymbol{k}_i}{k_i} A_i \sin\left(\boldsymbol{k}_i^T \boldsymbol{x}_0 - \omega_i t + \phi_{wave,i}\right), \tag{5.2}$$

$$\zeta(\boldsymbol{x}_0, t) = \sum_{i=1}^{N} A_i \cos(\boldsymbol{k}_i^T \boldsymbol{x}_0 - \omega_i t + \phi_{wave,i}), \tag{5.3}$$

where horizontal $\boldsymbol{x}$ and vertical $\boldsymbol{\zeta}$ displacement describe the wave field according to the undisturbed horizontal location $\boldsymbol{x}_0 = (x_0, y_0)^T$ and the vertical height $\zeta_0 = 0$. The other terms are amplitude A_i, steepness q_i, wave-number k_i, angular frequency ω_i, time t and random phase $\phi_{wave,i}$. The wave-vector $\boldsymbol{k_i}$ represents a direction of travel with magnitude k_i.

In order to achieve a simple wave field, the selection of small set of wave components in (5.2) and (5.3) is enough to obtain realistic wavy water surface. However, the wave components can be obtained using a wave spectrum $S(\omega)$ that captures the characteristics of the sea or the ocean. One of the wave spectra is the Pierson-Moskowitz spectrum describing fully developed

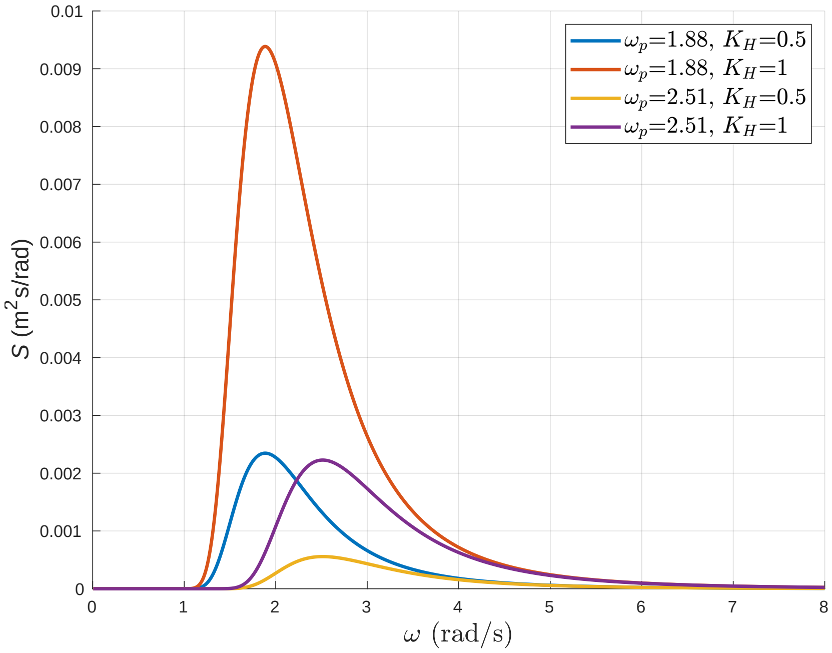


Figure 5.3: The Pierson-Moskowitz spectrum for different K_H and ω_p values.

sea in deep water. The expression of such spectrum used in the VRX simulator is as follows

$$S(\omega) = (K_H)^2 \frac{\alpha g^2}{\omega^5} \exp\left(-\frac{5}{4}\left(\frac{\omega_p}{\omega}\right)^4\right), \tag{5.4}$$

where ω_p is the peak angular frequency, K_H is the non-dimensional gain value, g is the acceleration of gravity and $\alpha = 8.1 \cdot 10^{-3}$. The Pierson-Moskowitz spectrum for different K_H and ω_p values is shown in Fig. 5.3. The individual wave components are obtained from the spectrum using the following equation

$$A_i^2 = 2S(\omega_i)\Delta_{\omega_i}, \tag{5.5}$$

where ω_i is the sample location along the spectrum and Δ_{ω_i} is the width of the frequency band of an individual sample.

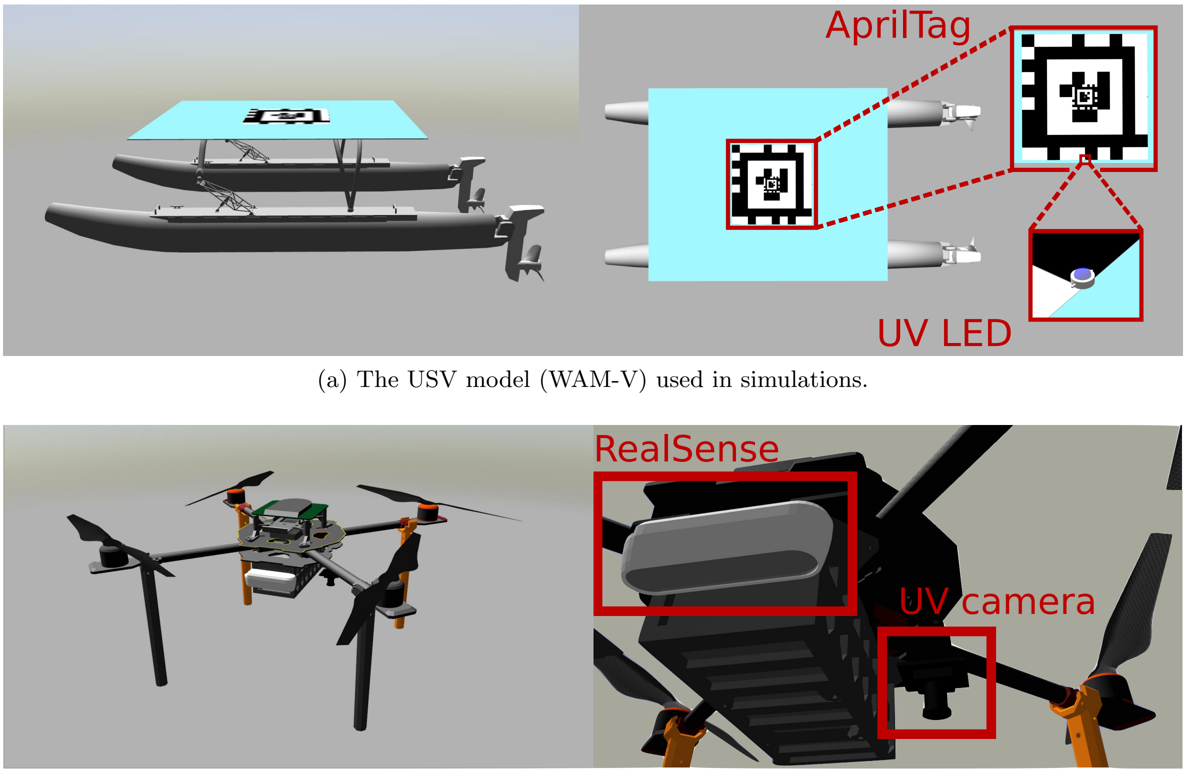
The wind disturbance can significantly influence the object on the water surface [99]. The total wind speed $V_\omega(t)$ is defined as follows

$$V_\omega = \bar{v} + v_g(t). \tag{5.6}$$

The $\bar{v}$ is constant mean wind speed and the $v_g(t)$ is varying wind speed with zero mean. The $v_g(t)$ component is modeled using approximation of the Harris spectrum that can be expressed in the following transfer function

$$h(s) = \frac{K_\omega}{1 + \tau_g s}, \tag{5.7}$$

where K_ω is the response magnitude and τ_g is a time constant. The models of waves and wind are then used to compute τ_{waves} and τ_{wind} acting on the USV according to the algorithm presented in [99].



(a) The USV model (WAM-V) used in simulations.

(b) The UAV model (Tarot t650) used in simulations.

Figure 5.4: Simulation model of the USV (WAM-V) and the UAV (Tarrot t650) used in the Gazebo simulator.

The USV model used in simulations is called Wave Adaptive Modular Vessel (WAM-V). The WAM-V is constructed as a catamaran with two parallel hulls on which the deck with required equipment is located. In order to use the estimation system designed in this thesis, the WAM-V deck is modified to carry necessary sensor equipment (Chap. 4). The AprilTag is put on top of the WAM-V deck and represents the spot for the UAV landing. On the AprilTag, the UV LEDs used by UVDAR system are placed according to the pattern described in Sec. 4.2.1. Finally, the GPS and the IMU sensors[5] are added on the WAM-V deck as the USV onboard sensors. The WAM-V model is shown in Fig. 5.4a. On the left part of the Fig. 5.4a, the side view of WAM-V is presented. On the right part, the top view of the WAM-V is given with a marked AprilTag and a UV LED.

The system for estimation of the USV states is designed to be onboard of the UAV. The used UAV model is called Tarot t650 designed by the MRS group[2]. The Tarot is a quad-rotor helicopter that is able to carry necessary the sensor equipment for the presented estimation system, i.e. GPS sensor, UV camera for UVDAR system, and RealSense D435 camera[6]. The RealSense D435 camera is used to provide images to the AprilTag detector. The Tarot t650 used in simulations is shown in Fig. 5.4b.

[5] http://wiki.ros.org/hector_gazebo_plugins

[6] https://www.intelrealsense.com/depth-camera-d435/

Chapter 6

Verification

This chapter demonstrates the functionality of the estimation system designed in this thesis. Firstly, the estimation system was tested in the realistic robotic simulator Gazebo. A large number of simulations were performed to tune the parameters of the estimation system and prepare it for real-world deployment. After successfully testing in the Gazebo simulator, the system was verified by conducting real-world experiments. The UAV was cooperating with the USV using estimations and predictions given by the estimation system proposed in this thesis. During the verification, the estimation system provides state estimates at 10 Hz.

The USV state estimations are obtained using the Kalman filter (Chap. 2). The estimated states are used to predict the future states using the mathematical model of the USV. The prediction step of the Kalman filter is repeatedly applied to compute the desired number of predictions. Therefore, the estimation system proposes the estimation of current USV states and also predicts the future states, that can be used by planning systems of the UAV. In simulations, the UAV takes off from the pier and flies above position of the USV estimated using the received data from the USV onboard sensors (Fig. 5.2). Then, the UAV follows the USV using the Model Predictive Control (MPC) as a UAV trajectory planning algorithm [39]. The UAV also uses its onboard sensors to update the USV state estimations.

In order to verify the Kalman filters, the innovation tests presented in Sec. 2.4 for individual sensors were performed. The comparison of the estimated and predicted states with the Ground Truth (GT) is given by the RMSE value (2.47). The test 1 called Innovation magnitude bound test checks the consistency of the innovation $\boldsymbol{\zeta}(k)$ (2.48) with its covariance matrix $\boldsymbol{S}(k)$ (2.49). The Normalized innovation squared χ^2 test (test 2) tries to prove unbiasedness of the innovation by the computation of a sum of normalized innovation squared $\boldsymbol{q}(k)$ (2.53) which should lie in the interval defined by the hypothesis H_0 (2.54) for $\alpha = 0.05$. Test 3 uses a normalized time-averaged innovation correlation to show the whiteness of the innovation. Therefore, test 3 is known as the Innovation whiteness test.

6.1 Linear Kalman filter verification in Gazebo simulator

The Linear Kalman filter (Sec. 2.1) verified in this section uses the linear USV model (Sec. 3.3) extended by wave model (Sec. 3.4). The linear USV model uses wave model with predefined frequencies (3.94). Thus the frequencies of waves used in the Gazebo simulator have to be determined. The USV motion influenced by the waves is shown in Fig. 6.5 and 6.6. Waves used in the Gazebo simulator have a frequency spectrum shown in Fig. 6.1 obtained from the Discrete Fourier transform [100]. The wave frequency spectrum is an approximation of the Pierson-Moskowitz spectrum (5.4) using the parameters $\omega_p = 0.18$ Hz and $K_H = 0.025$. The five most important frequencies according to their magnitude have been selected as wave components in the linear USV model $\omega_{0_L} \in \{0.1792,\ 0.2160,\ 0.1930,\ 0.1976,\ 0.1884\}$ Hz. The method to obtain the wave frequencies is proposed as

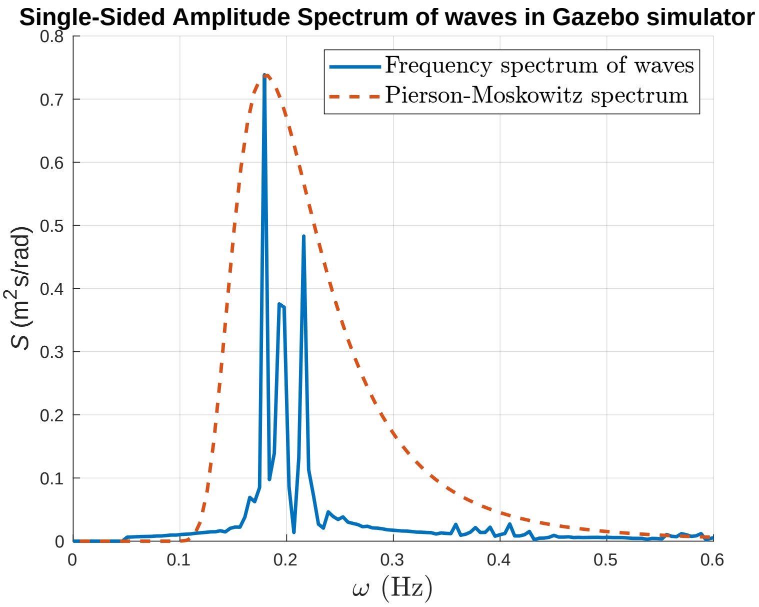


Figure 6.1: Fourier spectrum of waves used in the Gazebo simulator with Pierson-Moskowitz spectrum for $\omega_p = 0.18$ Hz and $K_H = 0.025$.

1. select the number of wave components N_{lc} in the linear USV model,
2. initialize the wave component frequencies with arbitrary values,
3. estimate USV motion in waves for at least 60 s,
4. use Discrete Fourier transform to obtain a spectrum of estimated states as a time signal,
5. select the most important frequency components according to their magnitude,
6. set the obtained frequencies into the linear USV model,
7. repeat from step 3 until convergence of frequencies is reached.

6.1.1 Innovation tests of Linear Kalman filter

The innovation tests (Sec. 2.4) applied to the Linear Kalman filter on the USV states $\boldsymbol{\eta}$ (3.1) are summarized in Table 6.1. The USV states $\boldsymbol{\eta}$ (3.1) consist of $x,\ y,\ z$ positions and roll ϕ, pitch θ, and yaw ψ angles. The tests are divided according to the individual sensors with respect to the measured USV states (Table 3.1). Only the one selected sensor was used in the Linear Kalman filter in its verification using the innovation tests. The percentage value of test 1 should be approximately 95%. The q value of test 2 has to lie inside the interval $[q_{min},\ q_{max}]$. The value of test 3 should be at least 95%.

All the innovation tests of the GPS and the IMU sensor are satisfied (Table 6.1). The test 1 and test 3 for the GPS sensor are shown in Fig. 6.2. The states $(x,\ y,\ z)$ estimated using the GPS sensors are shown in Fig. A.1. The GT is shown blue, the sensor data is in yellow and the estimated values are in red named as *Estimated* (LKF). The Fig. A.3 shows the test 1 and test 2 for the IMU sensor. Result of the estimation of states $(\phi,\ \theta,\ \psi)$ using the IMU is shown in Fig. A.2. The graphs and test results for the GPS and the IMU prove good estimation performance.

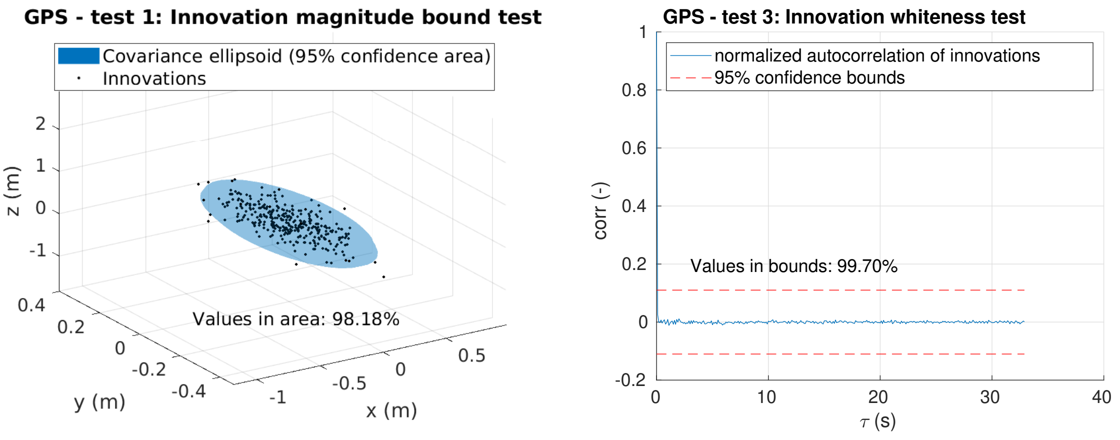
The test 1 of the UVDAR for states $(x,\ y,\ z)$ reaches 100% (Table 6.1). Fig. 6.3a shows that covariance ellipsoid is much bigger than mutual distances between individual innovation samples. That is caused by an over-estimated UVDAR sensor covariance matrix $\boldsymbol{R}$ that has to handle peaks in the UVDAR measurements (see Fig. 6.4). As the UVDAR sensor covariance matrix $\boldsymbol{R}$ is over-estimated, the value q in test 2 lies outside the interval defined by hypothesis H_0. The peaks in measurement also occur for states $(\phi,\ \theta,\ \psi)$ measured by the UVDAR system. The test 3 for states $(x,\ y,\ z)$ and $(\phi,\ \theta,\ \psi)$ in case of the UVDAR measurement passed successfully. Graphs showing the estimation of states $(x,\ y,\ z)$ (Fig. 6.4) and states $(\phi,\ \theta,\ \psi)$ (Fig. A.4) demonstrate performance of the Liner Kalman filter using the UVDAR system suffering from peaks in measurement.

The AprilTag detector measuring states $(x,\ y,\ z)$ and $(\phi,\ \theta,\ \psi)$ successfully passes all the tests (Table 6.1). Test 1 and test 3 of the AprilTag detector for states $(x,\ y,\ z)$ and $(\phi,\ \theta,\ \psi)$ are shown in Fig. A.7 and in Fig. A.9 respectively. The estimation of states $(x,\ y,\ z)$ using the AprilTag detector is presented in Fig. A.6. The AprilTag measurements of states $(\phi,\ \theta,\ \psi)$ are noisy as shown in Fig. A.8. However, the Linear Kalman filter estimates the states $(\phi,\ \theta,\ \psi)$ precisely which is demonstrated in Fig. A.8.

Performed innovation tests verified that the Linear Kalman filter is consistent with the obtained measurements from individual sensors. The covariance matrices of individual sensors $\boldsymbol{R}$ and the covariance matrix of the model $\boldsymbol{Q}$ have been determined according to the results of innovation tests (Sec. 2.4). As the Linear Kalman filter estimates the USV states using an individual sensor, the next Sec. 6.1.2 presents the results of the estimation using the Linear Kalman filter in which data of all the sensors is fused into one precise estimation.

sensor	states	test 1	test 2 (q of $[q_{min},\ q_{max}]$)	test 3
GPS	$(x,\ y,\ z)$	98.18%	644.67 of [590.70, 733.08]	99.70%
IMU	$(\phi,\ \theta,\ \psi)$	93.68%	1008.73 of [866.48, 1037.31]	98.53%
UVDAR	$(x,\ y,\ z)$	100.0%	683.56 of [824.48, 991.31]	96.91%
UVDAR	$(\phi,\ \theta,\ \psi)$	96.47%	830.38 of [824.48, 991.31]	99.34%
AprilTag	$(x,\ y,\ z)$	96.66%	8701.39 of [8486.71, 9005.08]	97.27%
AprilTag	$(\phi,\ \theta,\ \psi)$	95.46%	8800.21 of [8486.71, 9005.08]	98.46%

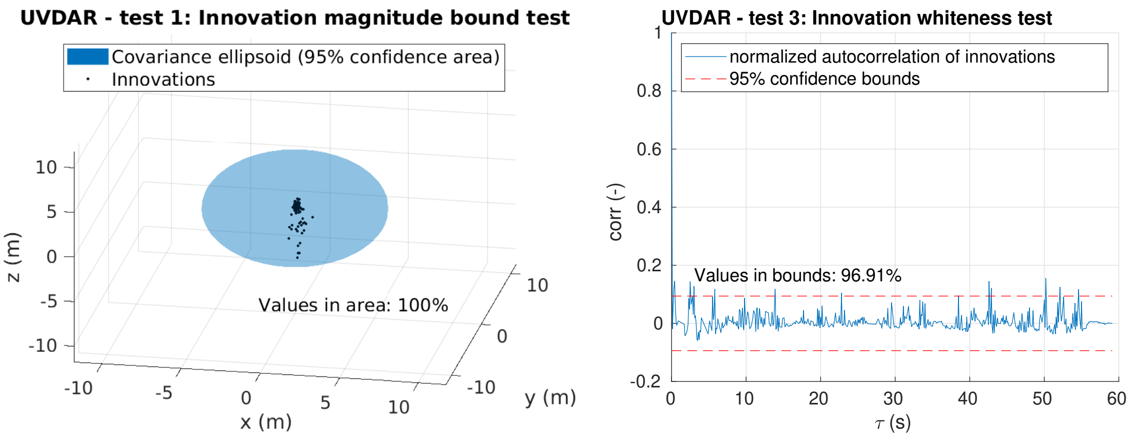
Table 6.1: Innovation tests applied to the Linear Kalman filter according to the individual sensors.



(a) Innovation test 1.

(b) Innovation test 3.

Figure 6.2: Innovation test 1 and test 3 of the Linear Kalman filter for states $(x,\ y,\ z)$ using the GPS measurements.



(a) Innovation test 1.

(b) Innovation test 3.

Figure 6.3: Innovation test 1 and test 3 of the Linear Kalman filter for states $(x,\ y,\ z)$ using the UVDAR measurements.

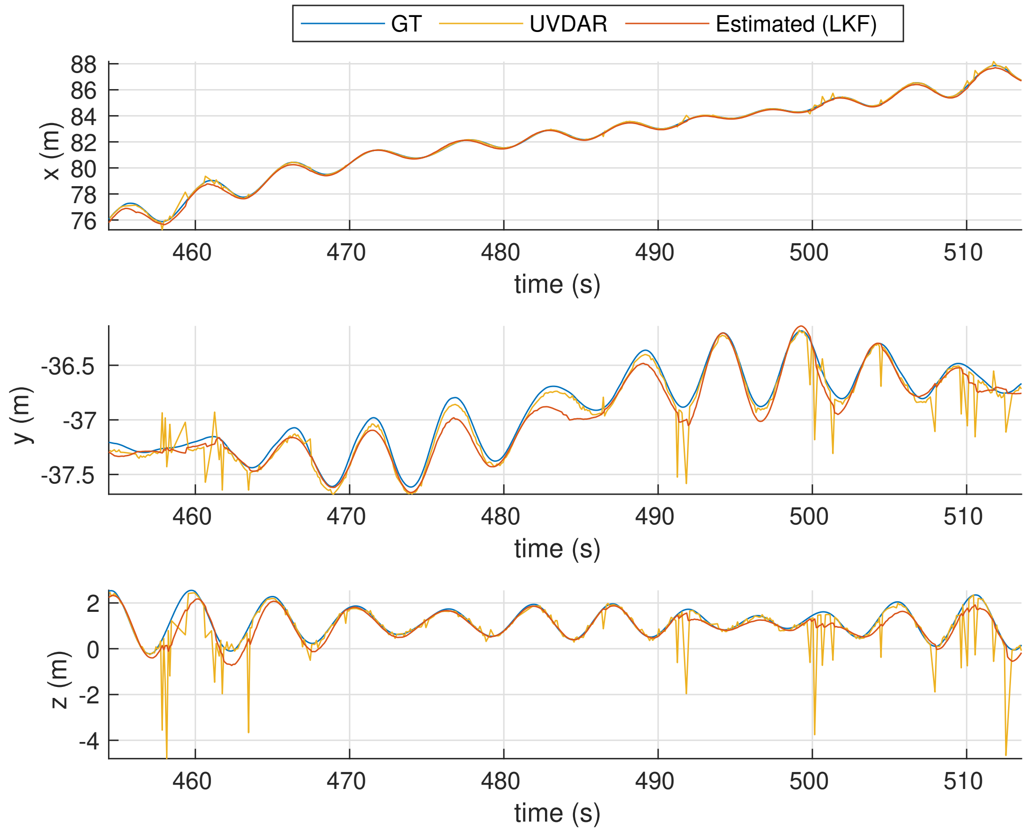


Figure 6.4: Estimated states $(x,\ y,\ z)$ by the Linear Kalman filter using the UVDAR measurements.

6.1.2 Estimation of USV states using Linear Kalman filter

The Table 6.2 presents the RMSE with respect to the sensors used for estimation of the desired USV states. The best estimation of states ($\phi,\ \theta,\ \psi$) is achieved using the IMU data as it has the smallest RMSE. The estimation of states ($x,\ y,\ z$) and corresponding linear velocities ($u,\ v,\ w$) using GPS provides the results with the largest RMSE. A better estimation than the GPS for states ($x,\ y,\ z$) and ($u,\ v,\ w$) is obtained using the UVDAR system. The RMSE for states ($x,\ y,\ z$) is twice smaller using the UVDAR system than when using the GPS and for states ($u,\ v,\ w$) the RMSE using the UVDAR system is a slightly smaller than when using the GPS sensor. However for the states ($\phi,\ \theta,\ \psi$) and corresponding angular velocities ($p,\ q,\ r$) the UVDAR has the largest RMSE from all the sensors. The smallest RMSE for ($x,\ y,\ z$) and ($u,\ v,\ w$) is reached using the AprilTag detector. The estimation of states ($\phi,\ \theta,\ \psi$) and ($p,\ q,\ r$) using the AprilTag provides two times smaller RMSE in comparison with the UVDAR system. However, RMSE of the AprilTag for states ($\phi,\ \theta,\ \psi$) is still four times larger than RMSE of the IMU for the same states.

Fusing data from all the sensors in the Linear Kalman filter provides a nice estimation of the USV states as shown in Fig. 6.5 and 6.6. The graphs show that the GT of all the desired USV states ($x,\ y,\ z,\ u,\ v,\ w,\ \phi,\ \theta,\ \psi,\ p,\ q,\ r$) is captured by the estimated states using the Linear Kalman filter. The RMSE of estimation using all the sensors in the Linear Kalman filter is provided in Table 6.2 in the last row. The RMSE of all the USV states is the lowest for fusing all the data from sensors in comparison with the onboard measurement systems (AprilTag and UVDAR) as they measure position ($x,\ y,\ z$) and orientation ($\phi,\ \theta,\ \psi$) of the USV. The RMSE of all states for the IMU and the GPS is not provided because the GPS is applicable only for the estimation of position ($x,\ y,\ z$) and corresponding linear velocities ($u,\ v,\ w$), and the IMU is usable only for orientation ($\phi,\ \theta,\ \psi$) and corresponding angular velocities ($p,\ q,\ r$). Overall, the estimation of all the USV states using data fusion from all the sensors in the Linear Kalman filter provides better results than estimation using a single sensor.

sensor	RMSE ($x,\ y,\ z$)	RMSE ($\phi,\ \theta,\ \psi$)	RMSE ($u,\ v,\ w$)	RMSE ($p,\ q,\ r$)	RMSE all states
GPS	0.768	-	0.985	-	-
IMU	-	0.795	-	0.114	-
UVDAR	0.330	8.275	0.977	0.220	8.339
AprilTag	0.068	3.579	0.848	0.129	3.680
all sensors	0.090	1.437	0.171	0.050	1.451

Table 6.2: RMSE of estimated USV states using the Linear Kalman filter according to the individual sensors.

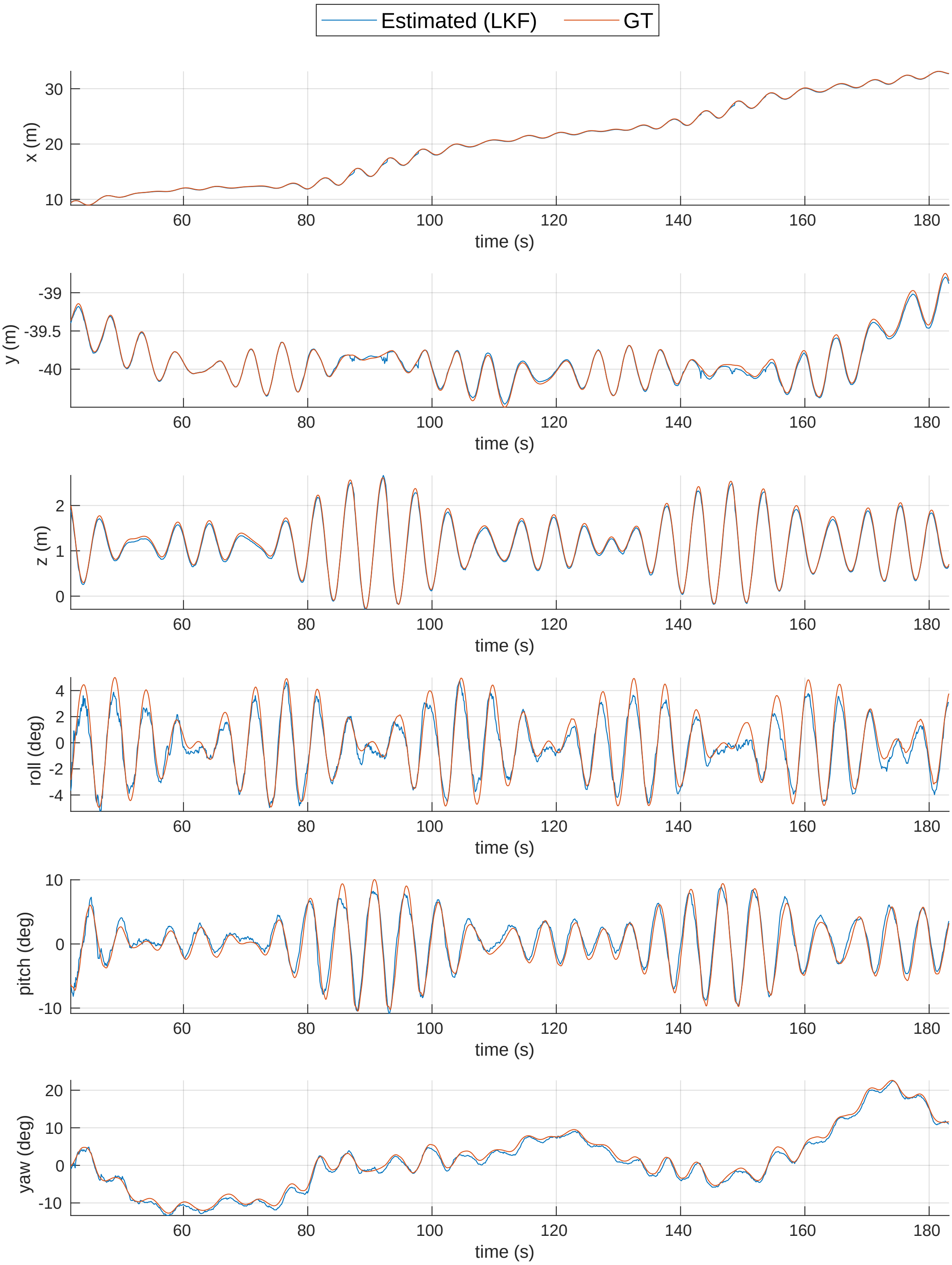


Figure 6.5: Estimated position $(x,\ y,\ z)$ and orientation $(\phi,\ \theta,\ \psi)$ of the USV using the Linear Kalman filter.

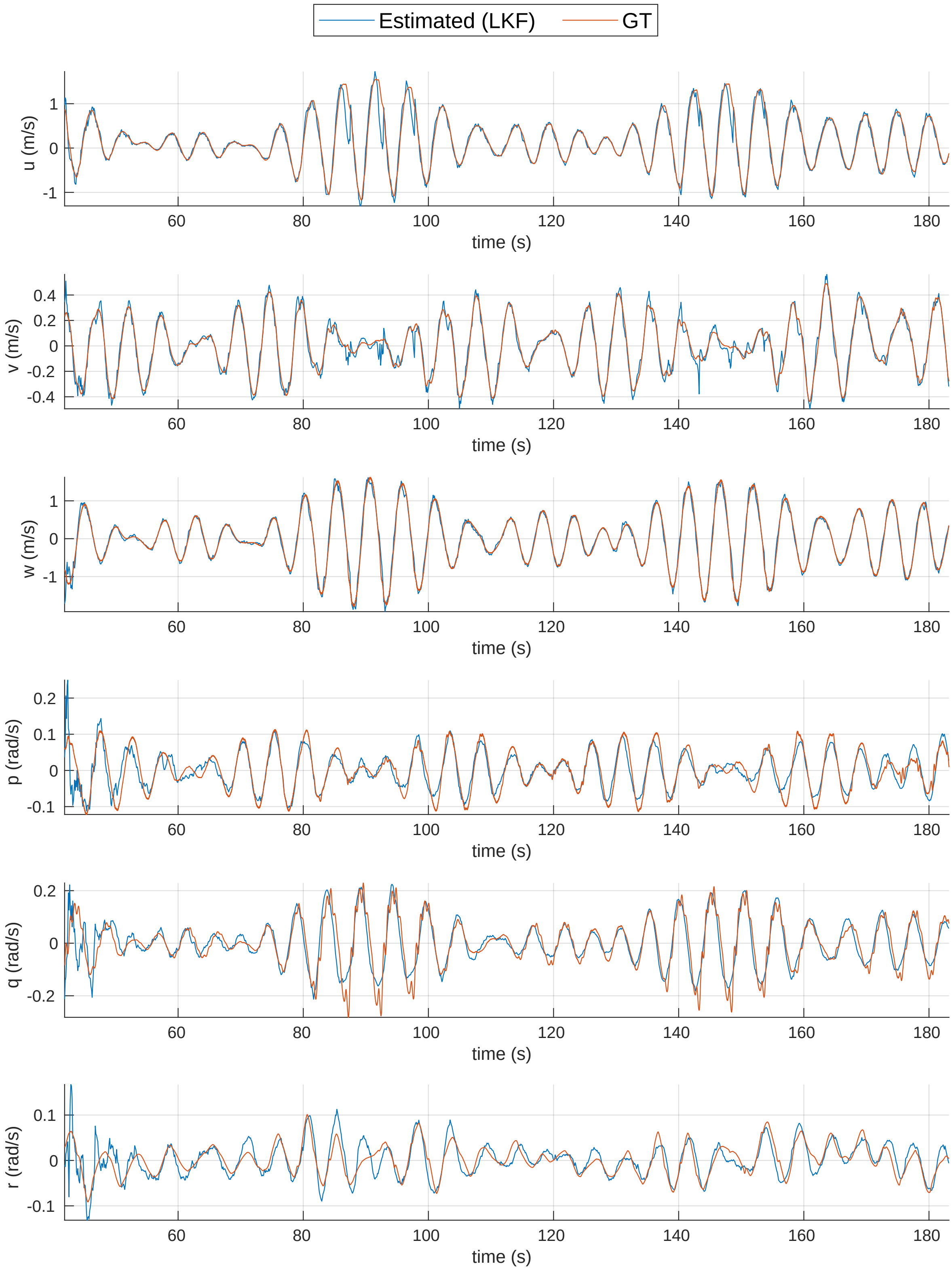


Figure 6.6: Estimated linear $(u,\ v,\ w)$ and angular $(p,\ q,\ r)$ velocities of the USV using the Linear Kalman filter.

6.1.3 Predictions using Linear Kalman filter

The previous Sec. 6.1.2 presents precise estimation of the USV states using the Linear Kalman filter. In this section, the estimated USV states using the Linear Kalman filter and the linear USV model are used to predict the future states of the USV. The two seconds predictions of the USV states are computed every two seconds. Resulted predictions of states $(x,\ y,\ z)$ and $(\phi,\ \theta,\ \psi)$ are shown in Fig. 6.7 and 6.8. The figures show that the predictions deviate more from the GT at the beginning of the simulation. However, the predictions of the USV states are more precise with the increasing time of estimation.

The RMSE of the predicted states is proposed in Table 6.3 in comparison with RMSE of the subsequent estimates. The predictions of states $(x,\ y,\ z,\ \phi,\ \theta,\ \psi)$ have ten times larger RMSE than the estimated states. However, the RMSE of predictions for states $(x,\ y,\ z)$ corresponds to RMSE of estimation of these states using only GPS sensor (Table 6.2). The RMSE of predictions for states $(\phi,\ \theta,\ \psi)$ corresponds to RMSE of these states estimated using UVDAR measurements (Table 6.2). These results prove the applicability of computed predictions to UAV planning algorithm.

USV states	RMSE $(x,\ y,\ z)$	RMSE $(\phi,\ \theta,\ \psi)$	RMSE $(x,\ y,\ z,\ \phi,\ \theta,\ \psi)$
predicted states	0.737	11.194	11.218
estimated states	0.090	1.437	1.440

Table 6.3: RMSE of predicted and estimated USV states using the Linear Kalman filter.

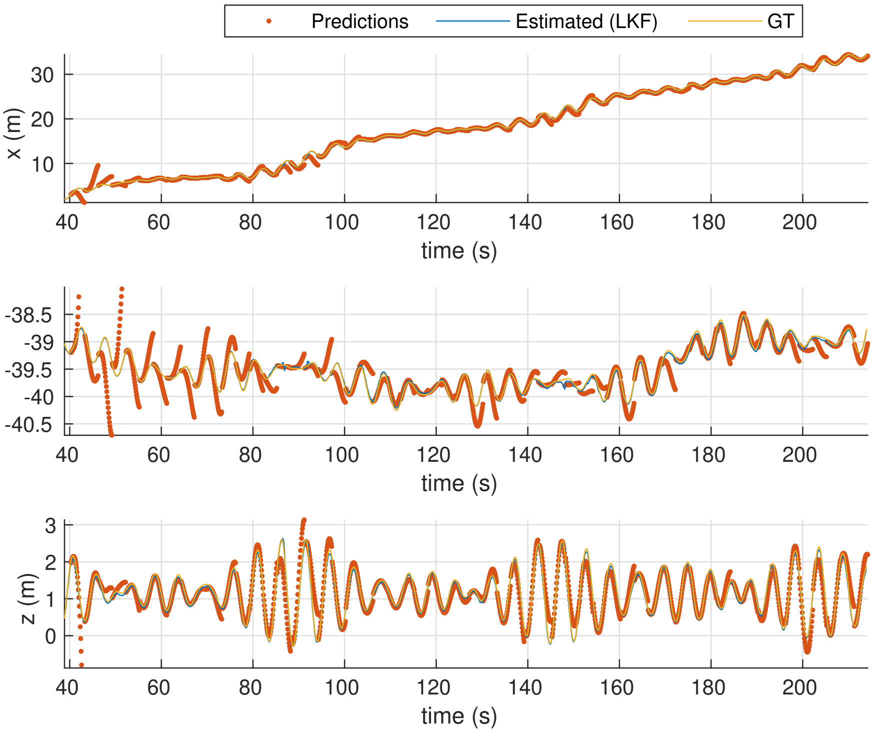


Figure 6.7: Predicted and estimated position $(x,\ y,\ z)$ of the USV using the Linear Kalman filter.

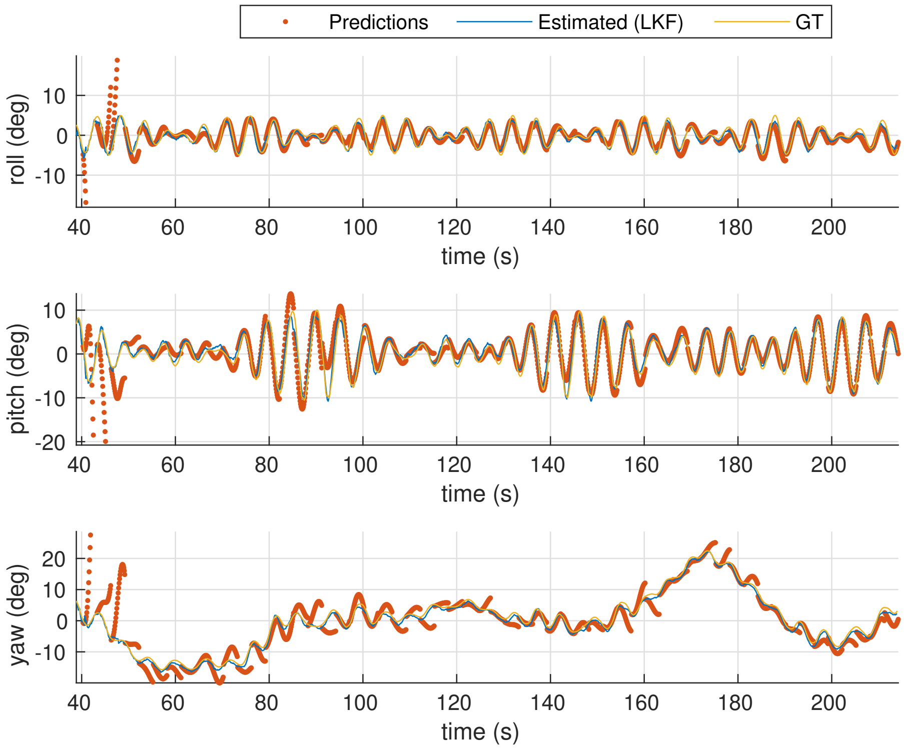


Figure 6.8: Predicted and estimated orientation (ϕ, θ, ψ) of the USV using the Linear Kalman filter.

6.2 Unscented Kalman filter verification in Gazebo simulator

This section presents a verification of the Unscented Kalman filter (Sec. 2.3) that uses the nonlinear USV model (Sec. 3.2) containing wave dynamics (Sec. 3.4). In contrast with the Linear Kalman filter (Sec. 6.1), the nonlinear USV model does not need set frequencies of individual components of the wave model (Sec. 3.4). The frequencies of wave model components are directly influenced by the state $x_{\omega_{N3}}$ of the wave model (Sec. 3.4) that is estimated during update steps of the Unscented Kalman filter. The Sec. 6.2.1 presents innovation tests of the Unscented Kalman filter using individual sensors. The Sec. 6.2.2 proposes results of an estimation of the USV states. The Sec. 6.2.3 demonstrates the usability of prediction steps of the Unscented Kalman filter to predict future USV states.

6.2.1 Innovation tests of Unscented Kalman filter

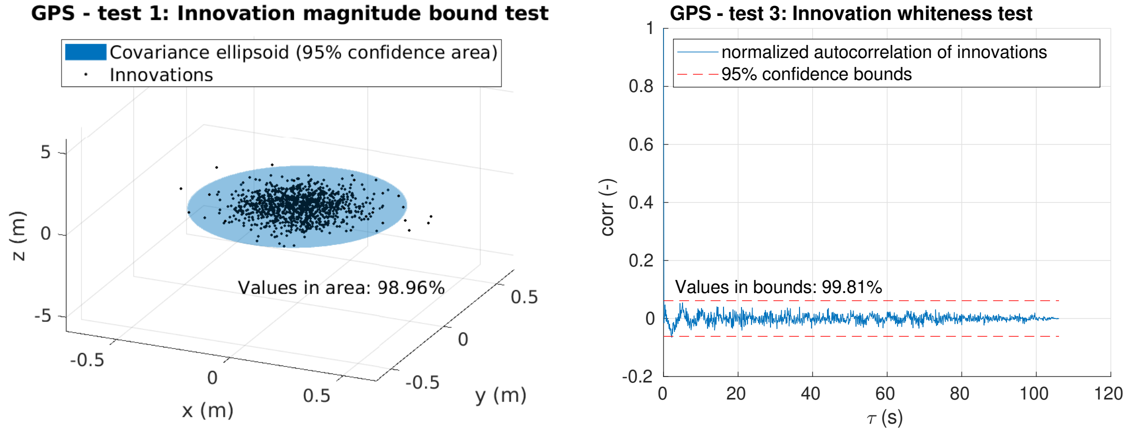
The Table 6.4 contains results of the innovation tests applied to the Unscented Kalman filter on USV states $\boldsymbol{\eta}$ (3.1) consisting of position x, y, z and angles: roll ϕ, pitch θ and yaw ψ. The innovation tests are done in the same manner as for the Linear Kalman filter (Sec. 6.1.1). The innovation tests are performed for individual sensors with respect to the USV states that the sensors measure. In graphs, the estimated USV states using the Unscented Kalman filter are called *Estimated* (UKF). The results in Table 6.4 show that the GPS innovation passes all innovation tests. The innovation test 1 and test 3 of the GPS are shown in Fig. 6.9 and the estimated USV states (x, y, z) are in Fig. A.10. The IMU innovation also passes all the tests (Table 6.4). The Fig. A.12 displays the innovation test 1 and test 3 if the IMU is used. Estimated states $(\phi,\ \theta,\ \psi)$ using the IMU are proposed in Fig. A.11.

The UVDAR sensor is applied to estimate $(x,\ y,\ z)$ as shown in Fig. A.13 and $(\phi,\ \theta,\ \psi)$ as shown in Fig. A.15. All the tests are satisfied when the UVDAR sensor is used (Table 6.4). The innovation test 1 and test 3 for states $(x,\ y,\ z)$ measured using the UVDAR are shown in Fig. A.14. The Fig. A.16 presents the UVDAR innovation test 1 and test 3 for states $(\phi,\ \theta,\ \psi)$. The estimations of $(x,\ y,\ z)$ and $(\phi,\ \theta,\ \psi)$ using the AprilTag are presented in Fig. A.17 and 6.11. As well as for previous sensors, all innovation tests for the AprilTag innovations passed. The test 1 and test 3 of the AprilTag for states $(x,\ y,\ z)$ are proposed in Fig. A.18. The Fig. 6.10 shows the test 1 and test 3 of the AprilTag for states $(\phi,\ \theta,\ \psi)$.

The results of the innovation tests for all the sensors demonstrate that covariance matrices $\boldsymbol{R}$ for individual sensors and covariance matrix $\boldsymbol{Q}$ of the nonlinear USV model are determined correctly (Sec. 2.4). Against the Linear Kalman filter (Sec. 6.1.1), the test 1 of the Unscented Kalman filter for states $(x,\ y,\ z)$ using the UVDAR does not reach 100%. The sensor covariance matrix $\boldsymbol{R}$ for the UVDAR in case of the Unscented Kalman filter does not have to be over-estimated as for the Linear Kalman filter.

sensor	states	test 1	test 2 (q of $[q_{min},\ q_{max}]$)	test 3
GPS	$(x,\ y,\ z)$	98.96%	2162.16 of [1998.16, 2253.63]	99.81%
IMU	$(\phi,\ \theta,\ \psi)$	93.10%	1040.71 of [893.23, 1066.56]	94.83%
UVDAR	$(x,\ y,\ z)$	96.42%	571.28 [547.23, 684.56]	95.77%
UVDAR	$(\phi,\ \theta,\ \psi)$	98.37%	590.67 of [547.23, 684.56]	99.02%
AprilTag	$(x,\ y,\ z)$	99.06%	8568.12 of [8510.35, 9029.43]	95.30%
AprilTag	$(\phi,\ \theta,\ \psi)$	93.57%	8725.63 of [8510.35, 9029.43]	97.67%

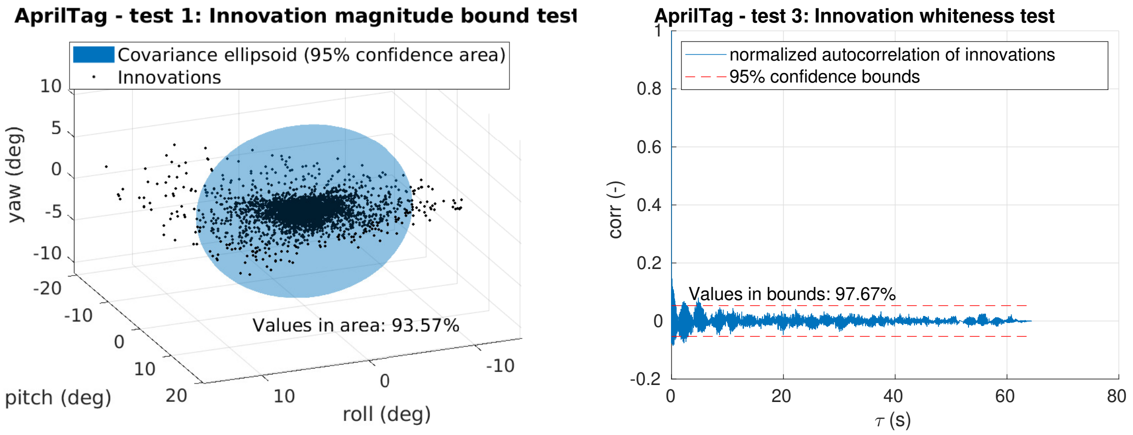
Table 6.4: Innovation tests applied to the Unscented Kalman filter according to the individual sensors.



(a) Innovation test 1.

(b) Innovation test 3.

Figure 6.9: Innovation test 1 and test 3 of the Unscented Kalman filter for states $(x,\ y,\ z)$ using the GPS measurements.



(a) Innovation test 1.

(b) Innovation test 3.

Figure 6.10: Innovation test 1 and test 3 of the Unscented Kalman filter for states (ϕ, θ, ψ) using the AprilTag measurements.

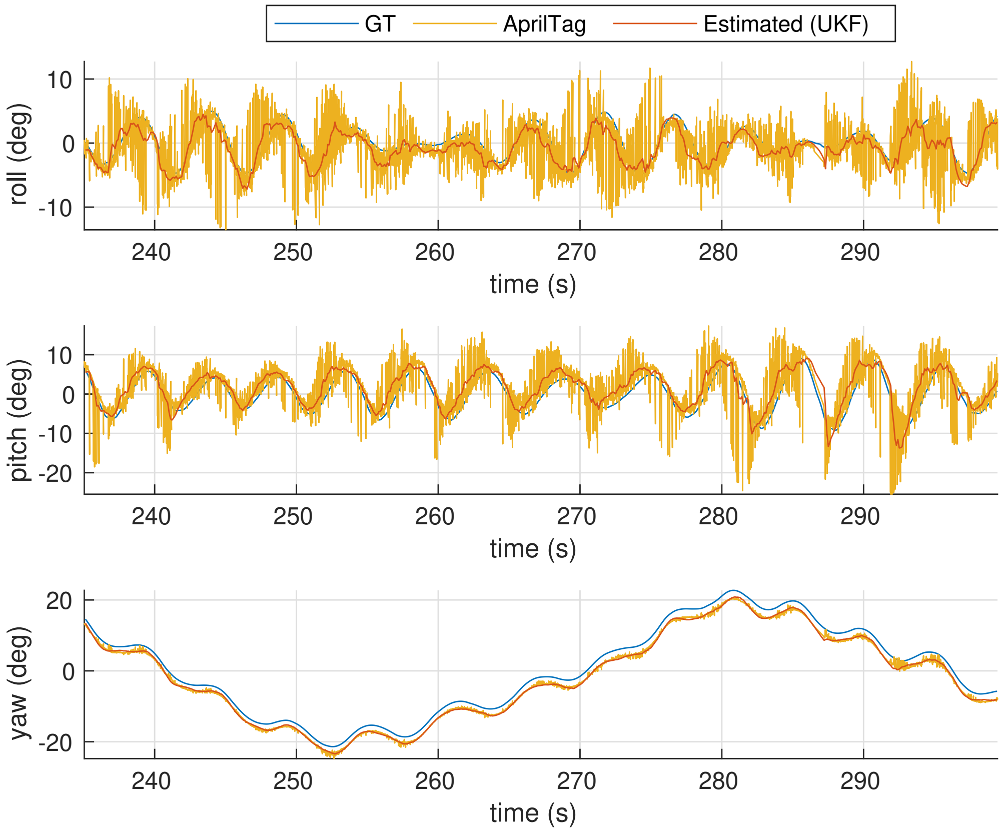


Figure 6.11: Estimated states (ϕ, θ, ψ) by the Unscented Kalman filter using the AprilTag measurements.

6.2.2 Estimation of USV states using Unscented Kalman filter

The RMSE according to the individual sensors used to estimate the desired USV states is proposed in Table 6.5. The worst result from all the sensors in case of RMSE for states $(x,\ y,\ z)$ and $(u,\ v,\ w)$ is provided by estimation using the GPS sensor. In contrast, the estimations of states $(\phi,\ \theta,\ \psi)$ and $(p,\ q,\ r)$ using the IMU sensor achieve the best results from all the sensors as the values of IMU RMSE for states $(\phi,\ \theta,\ \psi)$ and $(p,\ q,\ r)$ are the lowest. The estimated USV states $(x,\ y,\ z)$ and $(u,\ v,\ w)$ using the UVDAR system have twice smaller RMSE than states estimated by the GPS. However, the states $(\phi,\ \theta,\ \psi)$ estimated using the UVDAR system have ten times greater RMSE then estimations using the IMU. The RMSE for states $(p,\ q,\ r)$ is three times smaller for the IMU in comparison with the UVDAR. The smallest RMSE for states $(x,\ y,\ z)$ and $(u,\ v,\ w)$ from all the sensors is achieved using the AprilTag. However, the RMSE of the AprilTag for $(\phi,\ \theta,\ \psi)$ is five times higher than RMSE of the IMU. The RMSE for states $(p,\ q,\ r)$ using the AprilTag is slightly higher than RMSE for these states using the IMU.

The estimation of all USV states using the Unscented Kalman filter is presented in Fig. 6.12 (states $(x,\ y,\ z,\ \phi,\ \theta,\ \psi)$) and Fig. 6.13 (states $(u,\ v,\ w,\ p,\ q,\ r)$). The estimated values nicely catch the GT data as the Unscented Kalman filter fuses data from all sensors. The last row of Table 6.5 provides RMSE of the estimated USV states using the Unscented Kalman filter. The RMSE for all states has the lowest value for fusion data from all sensors in comparison with onboard measurement systems — the AprilTag detector and the UVDAR system.

The RMSE of estimated USV states using the Unscented Kalman filter (Table 6.5) is lower than RMSE for the Linear Kalman filter (Table 6.2). Therefore, the Unscented Kalman filter provides better estimation than the Linear Kalman filter. That is caused by the nonlinear USV model (Sec. 3.2) used in the Unscented Kalman filter. The Linear Kalman filter uses the linear USV model (Sec. 3.3) that comes as linearization of the nonlinear USV model (Sec. 3.2). Thus, the linear USV model is less accurate when trying to capture dynamics of real USV.

sensor	RMSE	RMSE	RMSE	RMSE	RMSE
	$(x,\ y,\ z)$	$(\phi,\ \theta,\ \psi)$	$(u,\ v,\ w)$	$(p,\ q,\ r)$	all states
GPS	0.681	-	0.837	-	-
IMU	-	0.653	-	0.087	-
UVDAR	0.289	7.055	0.385	0.248	7.075
AprilTag	0.046	3.517	0.146	0.092	3.634
all sensors	0.097	0.930	0.157	0.025	0.948

Table 6.5: RMSE of estimated USV states using the Unscented Kalman filter according to the individual sensors.

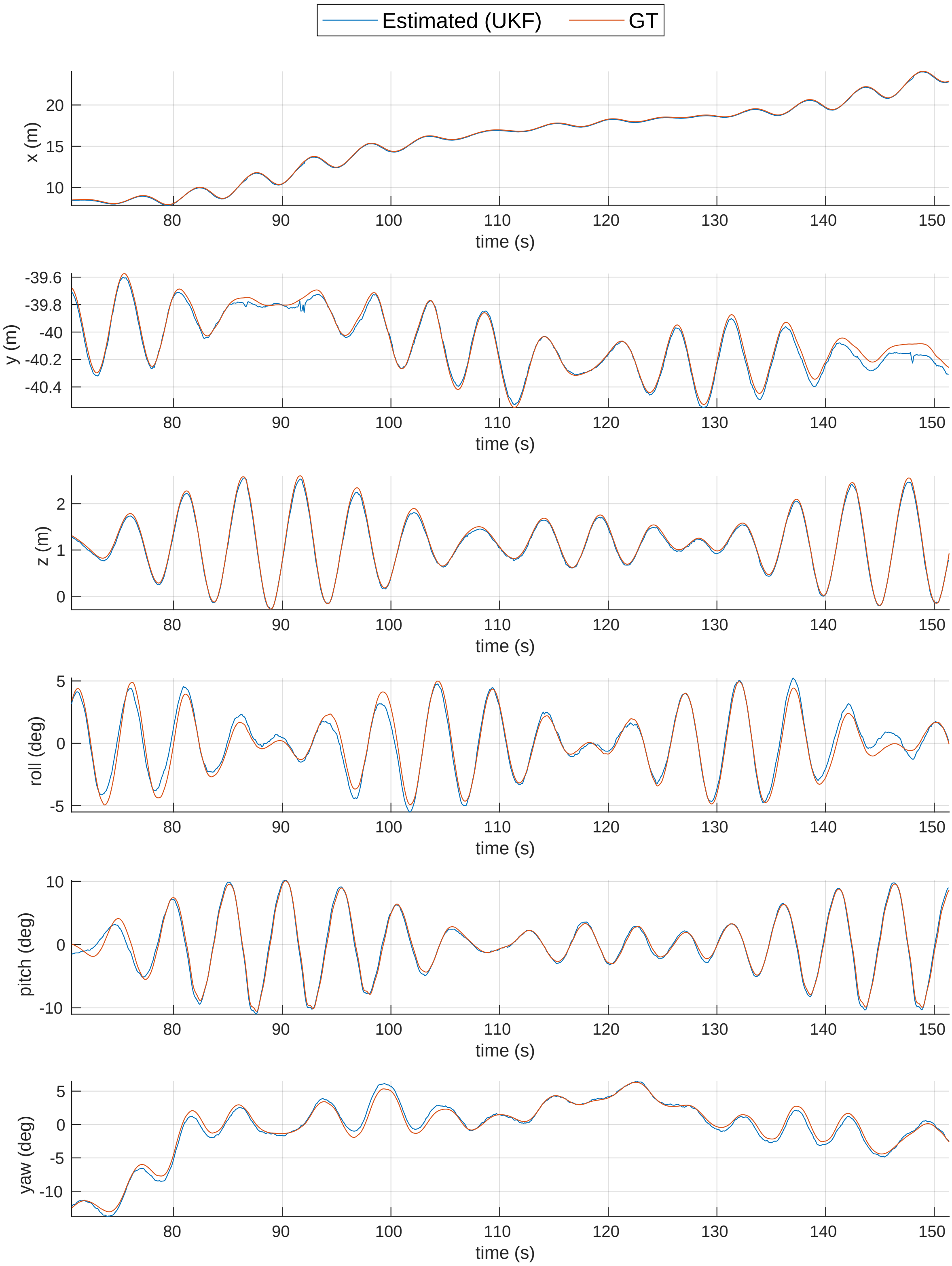


Figure 6.12: Estimated position $(x,\ y,\ z)$ and orientation $(\phi,\ \theta,\ \psi)$ of the USV using the Unscented Kalman filter.

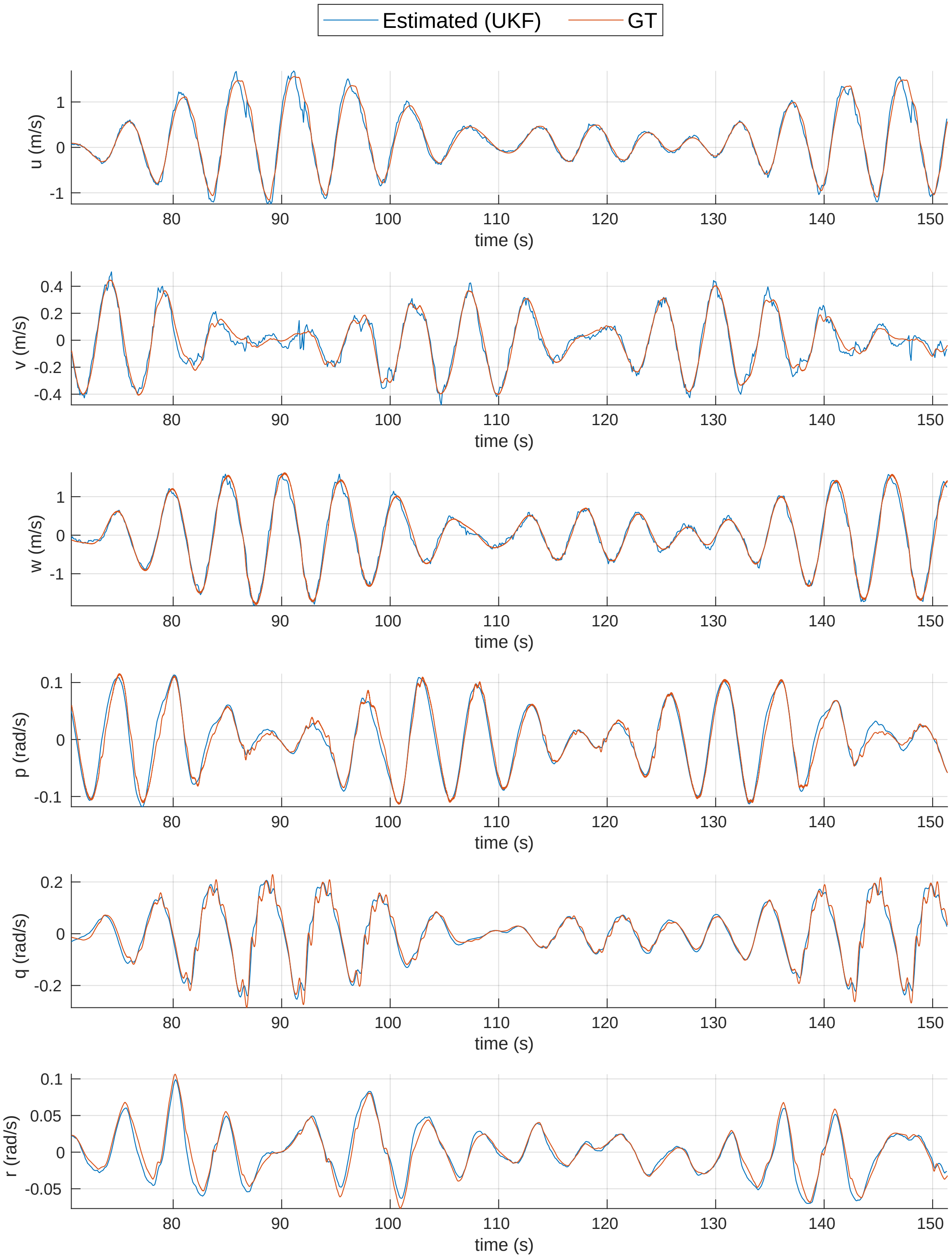


Figure 6.13: Estimated linear $(u,\ v,\ w)$ and angular $(p,\ q,\ r)$ velocities of the USV using the Unscented Kalman filter.

6.2.3 Predictions using Unscented Kalman filter

The Unscented Kalman filter provides precise estimation of the USV states as presented in Sec. 6.2.2. The estimated states can be used to predict future USV states applying the prediction steps of the Unscented Kalman filter. Every two seconds, the USV states are predicted for two seconds. The predicted USV states $(x,\ y,\ z,\ \phi,\ \theta,\ \psi)$ are shown in Fig. 6.14 and 6.15. The graphs in Fig. 6.14 and 6.15 demonstrate, that the Unscented Kalman filter is able to estimate and predict the motion of the USV.

The RMSE of estimated and predicted USV states is presented in Table 6.6. The RMSE of states $(x,\ y,\ z,\ \phi,\ \theta,\ \psi)$ is five times larger for predicted states than for estimated ones. However, the RMSE of predicted states $(x,\ y,\ z)$ corresponds to the RMSE of estimated states $(x,\ y,\ z)$ using only the GPS sensor (Table 6.5). The RMSE of predicted states $(\phi,\ \theta,\ \psi)$ is smaller than RMSE of estimated states $(\phi,\ \theta,\ \psi)$ using the UVDAR (Table 6.5). However, RMSE of predicted states $(\phi,\ \theta,\ \psi)$ is still larger than the RMSE of these states estimated by the AprilTag detector (Table 6.5).

USV states	RMSE $(x,\ y,\ z)$	RMSE $(\phi,\ \theta,\ \psi)$	RMSE $(x,\ y,\ z,\ \phi,\ \theta,\ \psi)$
predicted states	0.647	4.613	4.659
estimated states	0.097	0.930	0.935

Table 6.6: RMSE of predicted and estimated USV states using the Unscented Kalman filter.

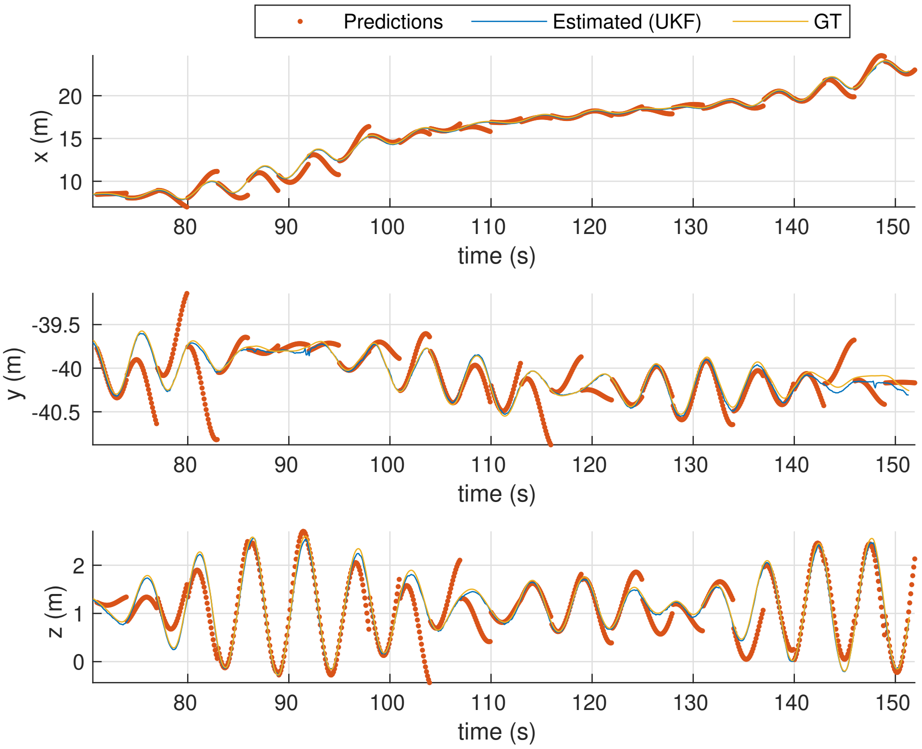


Figure 6.14: Predicted and estimated position $(x,\ y,\ z)$ of the USV using the Unscented Kalman filter.

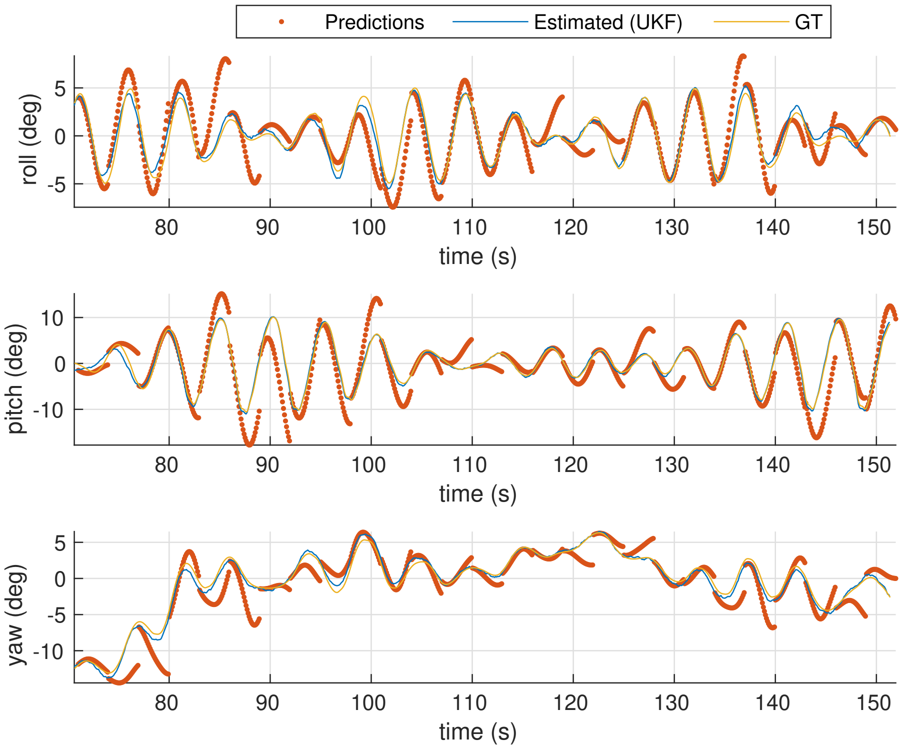


Figure 6.15: Predicted and estimated orientation (ϕ, θ, ψ) of the USV using the Unscented Kalman filter.

6.3 Comparison of Linear and Unscented Kalman filter

The main difference between the Linear Kalman filter and Unscented Kalman filter is in the USV model. The Unscented Kalman filter uses the nonlinear USV model extended by the nonlinear wave model (Sec. 3.2 and 3.4). The Linear Kalman filter uses the linear USV model obtained by linearization of the nonlinear USV model (Sec. 3.3). The linear USV model is extended by the linear wave model (Sec. 3.4). The Sec. 2.1 presents the algorithm of the Linear Kalman filter as a set of matrix equations. The Unscented Kalman filter uses the unscented transformation that enables to use the nonlinear USV model. Algorithm of the Unscented Kalman filter is presented in Sec. 2.3.

The innovation tests of Linear Kalman filter are presented in Sec. 6.1.1. The innovation tests demonstrate that estimations using the Linear Kalman filter are consistent with the sensor's measurements. However, test 1 of the UVDAR system for states (x, y, z) reaches a value 100%, but this value should be around 95%. The sensor covariance matrix $\boldsymbol{R}$ in the Linear Kalman filter for states (x, y, z) measured by the UVDAR system is over-estimated. The reason is high peaks in UVDAR measurements (Fig. 6.4) which must not affect the estimation. The innovation tests of the Unscented Kalman filter proposed in Sec. 6.2.1 prove that estimations using the Unscented Kalman filter are consistent with the sensor's measurements. The UVDAR measurements still contain the peaks (Fig. A.13). However, the sensor covariance matrix $\boldsymbol{R}$ of the UVDAR system in the Unscented Kalman filter for states (x, y, z) is not over-estimated as the value of test 1 is 96.42%.

The RMSE of the Linear Kalman filter using all sensors for the estimation is 1.451 (Table 6.2). However, the Unscented Kalman filter achieves better results as its RMSE using all sensors is 0.948 (Table 6.5). The difference is mainly caused by estimating orientation ($\phi,\ \theta,\ \psi$). The Linear Kalman filter has RMSE of the orientation 1.437. However, the RMSE of orientation estimated by the Unscented Kalman filter is 0.930. Therefore, the Unscented Kalman filter provides a more precise estimation of the USV orientation ($\phi,\ \theta,\ \psi$). The reason is that the Unscented Kalman filter uses the nonlinear USV model that better captures the USV motion dynamics.

The predictions computed using the Linear Kalman filter are presented in Sec. 6.1.3. The RMSE of predicted states using the Linear Kalman filter is 11.218 (Table 6.3). However, the RMSE of predictions using the Unscented Kalman filter is 4.659 (Table 6.6), which is lower than for the Linear Kalman filter. The difference between these two values of RMSE is again caused by RMSE of the USV orientation ($\phi,\ \theta,\ \psi$). The RMSE of predicted states ($\phi,\ \theta,\ \psi$) using the Unscented Kalman filter is 4.613 (Table 6.6). However, the RMSE of predicted states ($\phi,\ \theta,\ \psi$) using the Linear Kalman filter is 11.194 (Table 6.3).

Chapter 7

Real-world experiments

After a successful verification in the realistic robotic simulator Gazebo, the estimation system designed in this thesis was deployed in real-world experiments. The experiments took place in the Czech republic by the Vltava river. The place of real-world experiments is shown in Fig. 7.1. The estimation system designed in this thesis was implemented into MRS system containing the MPC controller [39] that is used as a trajectory planning algorithm for the UAV. Videos documenting real-world experiments are available at `http://mrs.felk.cvut.cz/theses/novak2022`.

In the first scenario (Sec. 7.1), the UAV followed the moving USV. The USV moved in the water arbitrarily, and its motion also included a small wave excitation. The proposed estimation system of the USV states used data from sensors presented in Chap. 4 and computed the estimations and prediction of the USV states. The trajectory planning algorithm received the estimations and predictions of the USV states. Then the trajectory planning algorithm aligned the UAV trajectory with the current estimated USV states and the subsequent predictions of the USV states.

The second scenario (Sec. 7.2) demonstrated the ability of the UAV to take off from the ground and fly above the USV using an estimation of the USV states from data taken by sensors which are located onboard USV. The estimations and following predictions were then improved using data from the UAV onboard measurement systems — the AprilTag detector and the UVDAR system. The trajectory planning algorithm running onboard the UAV created a trajectory to land on the USV based on predicted states of the USV.

The third scenario (Sec. 7.3) was a combination of the previous two scenarios. The UAV took off and flew above the USV whose states were estimated using received data from the USV onboard measurement systems. Then the UAV followed the moving USV using estimated and predicted states of the USV. During the USV following, the UAV used the predicted USV states to select the moment for landing. Finally, the UAV landed on the estimated landing platform located on the USV.

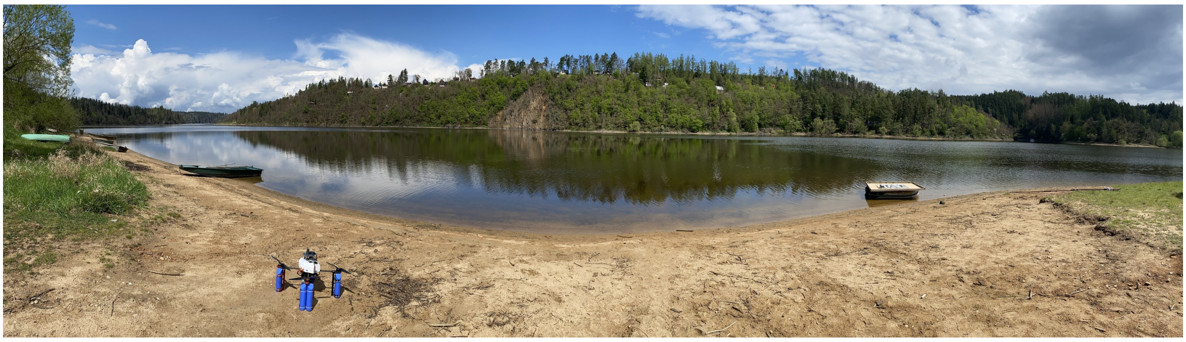


Figure 7.1: Place of real-world experiments in the Czech republic by the Vltava river.

The UAV used in real-world experiments is presented in Fig. 7.2. The UAV is Tarot t650 constructed by the MRS group that is the same type as used in simulations (Sec. 5.2). The UAV is equipped with the necessary sensors for the estimation system of USV states presented in this thesis. The RealSense for the AprilTag detector and the UV camera for the UVDAR system, both pointing down, are located below the case with the computing unit as shown in Fig. 7.2. The UAV also carries a GPS module to obtain its position in GPS coordinates that is used to include incoming USV GPS data into the estimation system of the USV states.

The Fig. 7.3 shows the USV with a landing platform used in real-world experiments. The sensors placed on the landing platform are defined in Chap. 4. The AprilTag is put in the center of the landing platform. The four UV LEDs are located around the perimeter of the AprilTag, and one UV LED is placed in the center of AprilTag (Sec. 4.2.1). Next to the AprilTag with UV LEDs, the MRS boat unit is installed. The MRS boat unit contains a GPS module and a IMU sensor.

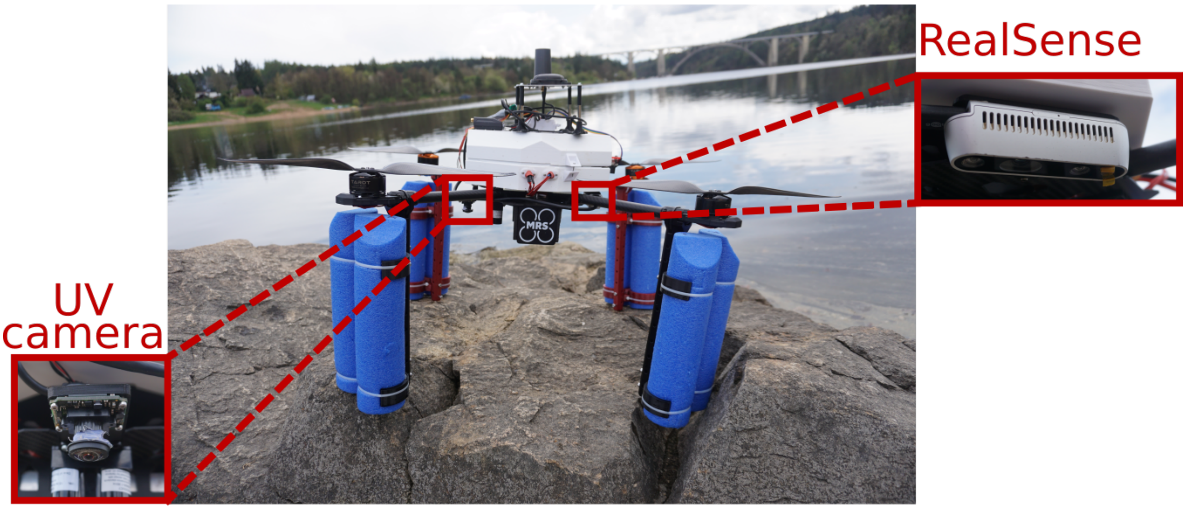


Figure 7.2: The UAV used in real-world experiments.

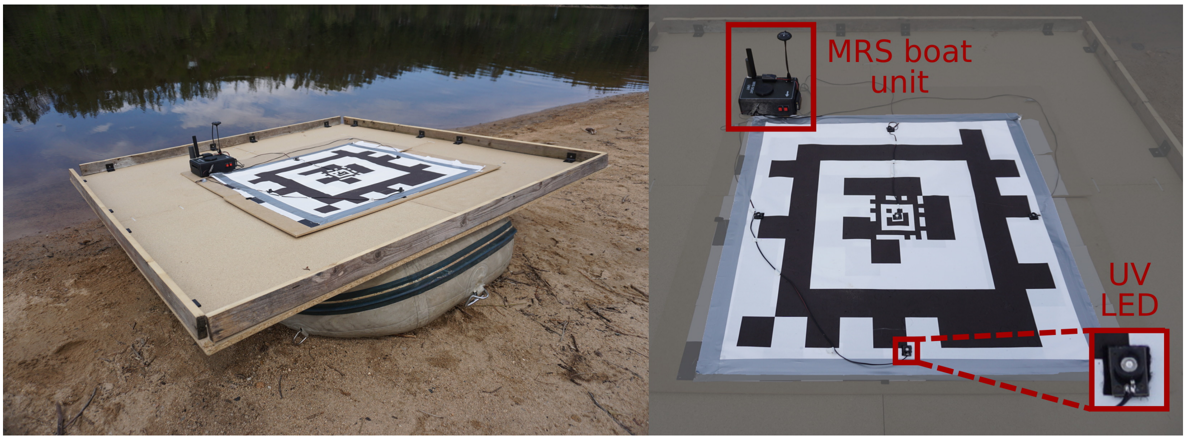
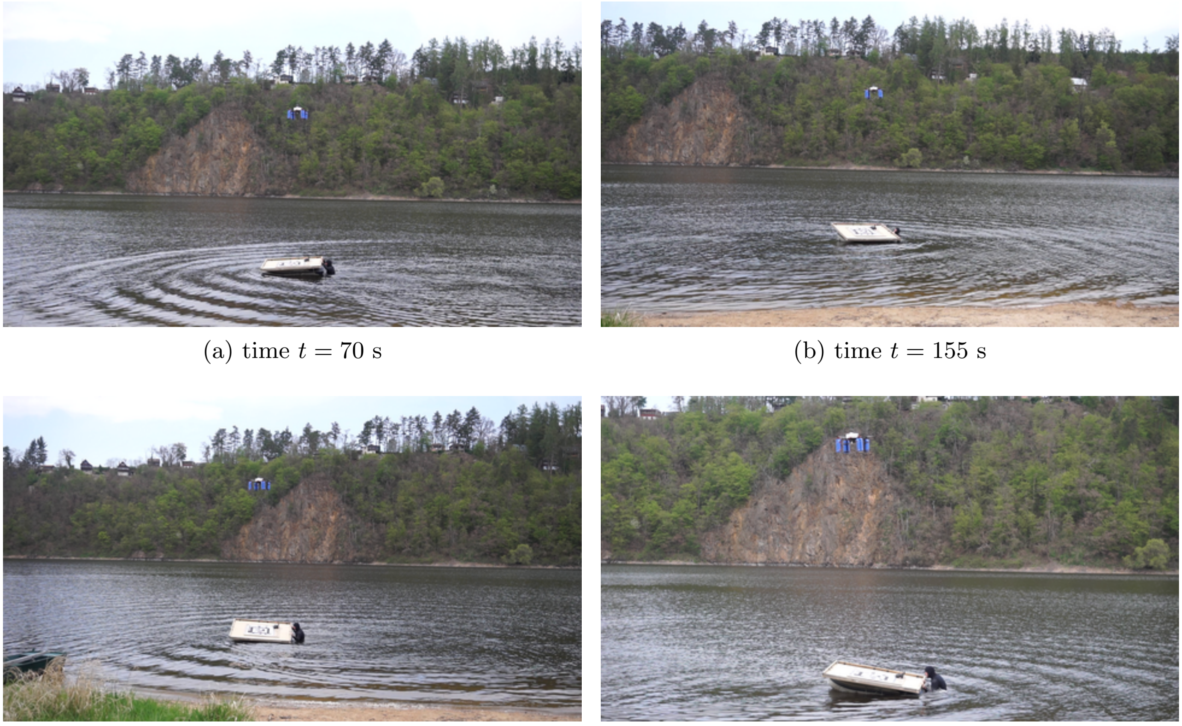


Figure 7.3: The USV with the landing platform used in real-world experiments.

7.1 Following USV

Estimation of the USV states during the process of the USV being followed by the UAV is presented in this section. Firstly, the UAV moved above the USV while the USV states are estimated using received GPS and IMU data from the USV. Then, the UAV onboard sensors improved the estimation of the USV states. The USV was moving while the UAV flew above it. The UAV followed the USV at a constant altitude. As the estimation system of the USV states provides the predictions of future USV states, the UAV planned trajectory according to predicted USV states to keep the USV in the proximity of the UAV onboard sensors. The USV was manually influenced to imitate waves acting on the USV (see video[1]). Snapshots from this real-world experiment are shown in Fig. 7.4.

The Fig. 7.5 presents the estimated USV states $x,\ y,\ z$, roll ϕ, pitch θ, and yaw ψ using the Linear and Unscented Kalman filter. The onboard UAV sensors provided the first data at time $t = 30$ s that improved the estimation of the USV states, especially in the z position. The graphs of roll ϕ and pitch θ contain the motion caused by the person imitating wave excitation. The pitch was influenced from 37 s to 130 s. Then the roll was affected from 130 s to 200 s. The pitch was again influenced from 200 s to 264 s. Both Kalman filters captured these motions in their estimations. The graphs of roll, pitch, and yaw do not contain the UVDAR and AprilTag measurements because these measurements are very noisy. The graphs of roll, pitch, and yaw including UVDAR and AprilTag measurements are proposed in Fig. B.1.



(a) time $t = 70$ s

(b) time $t = 155$ s

(c) time $t = 160$ s

(d) time $t = 218$ s

Figure 7.4: Snapshots from the real-world experiment in which the UAV followed the USV.

[1] http://mrs.felk.cvut.cz/theses/novak2022

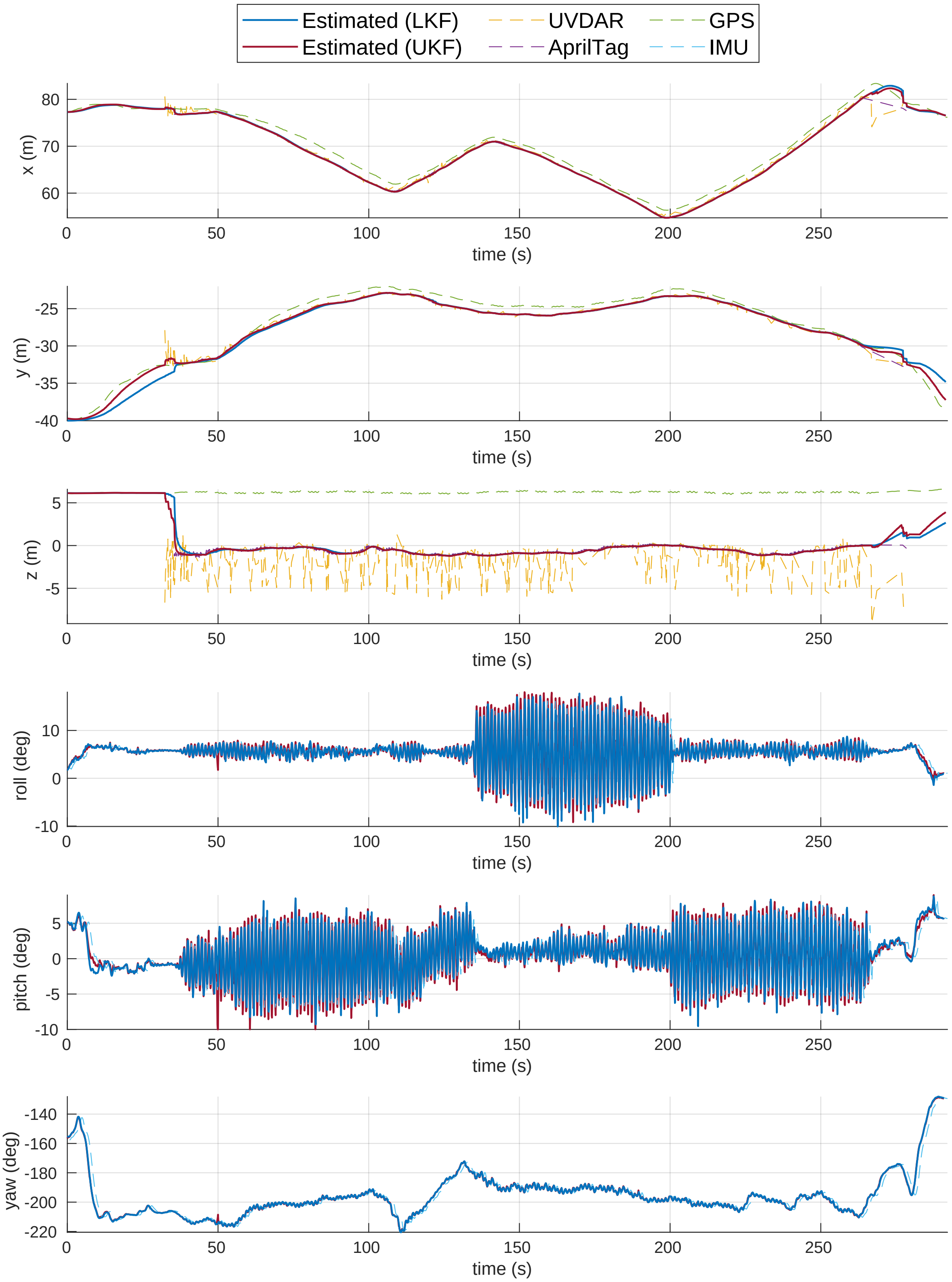
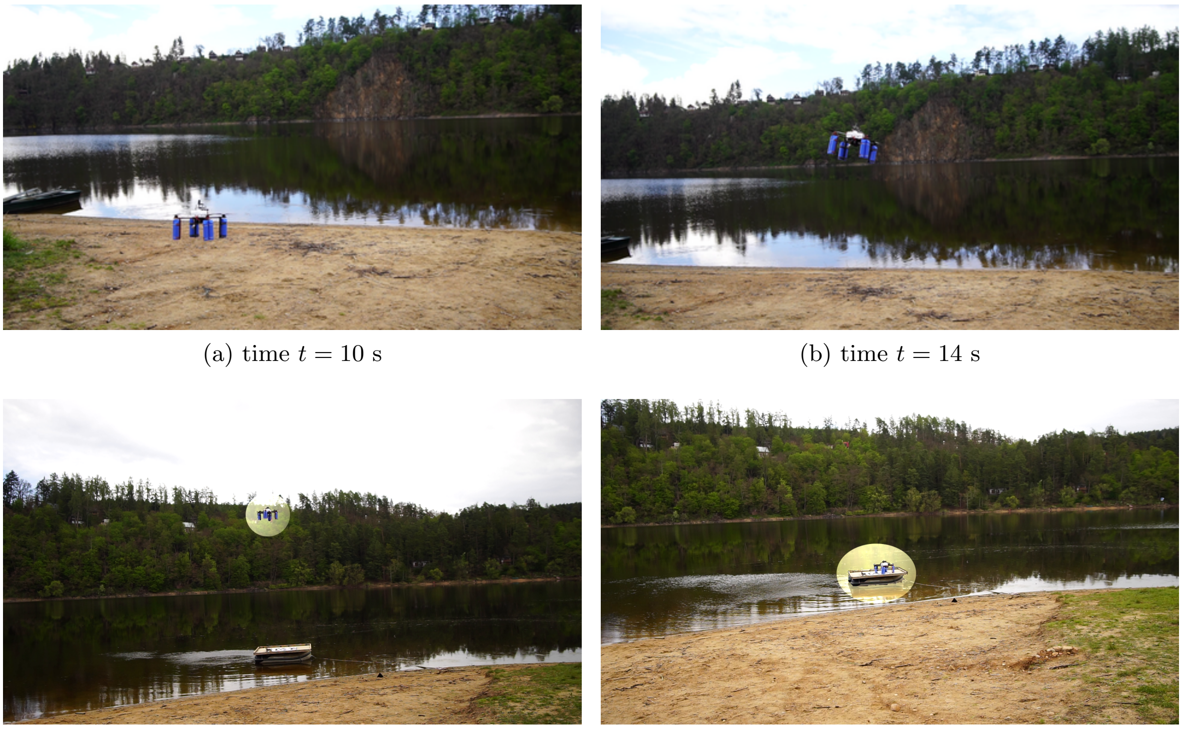


Figure 7.5: Estimation of the USV states (x, y, z, ϕ, θ, ψ) using the Linear Kalman filter (LKF) and the Unscented Kalman filter (UKF) while the UAV followed the USV.

7.2 Landing on USV

In this real-world experiment, the UAV autonomously took off and flew to the altitude of 8 m to x, y positions of the USV estimated using data received from the USV onboard sensors — the GPS and the IMU. After time $t = 20$ s (Fig. 7.7) the UAV got close to the USV and the estimated states were updated using measurements from the UAV onboard sensors — the UVDAR and the AprilTag. The USV moved only slightly as it was anchored on the water surface. The UAV hovered above the USV for 20 s. Thereafter the UAV successfully landed in the center of the landing platform on the USV using estimated USV states (see video[1]).

The graphs of USV states x, y, z, roll ϕ, pitch θ and yaw ψ estimated using the Linear and Unscented Kalman filter are proposed in Fig. 7.7. The graphs of position (x, y, z) prove the importance of using onboard UAV sensors to estimate USV states in case of a landing. Data of the GPS drifted up to 2 m compared to the onboard sensors. The graphs in Fig. 7.7 also show that the UVDAR measurements of z, roll, pitch, yaw and AprilTag measurements of roll and pitch were very noisy. However, the Linear and Unscented Kalman filter handled the situation well. The UAV landed on the USV at time $t = 53$ s. After this moment, the UAV onboard sensors did not propose measurements as they did not see desired markers in their camera frames. Therefore, the estimated USV states converged to measurements obtained from the GPS and the IMU placed on the USV. The Fig. 7.6 proposes the snapshots of this real-world experiment.



(a) time $t = 10$ s

(b) time $t = 14$ s

(c) time $t = 40$ s

(d) time $t = 53$ s

Figure 7.6: Snapshots from the real-world experiment in which the UAV landed on the USV.

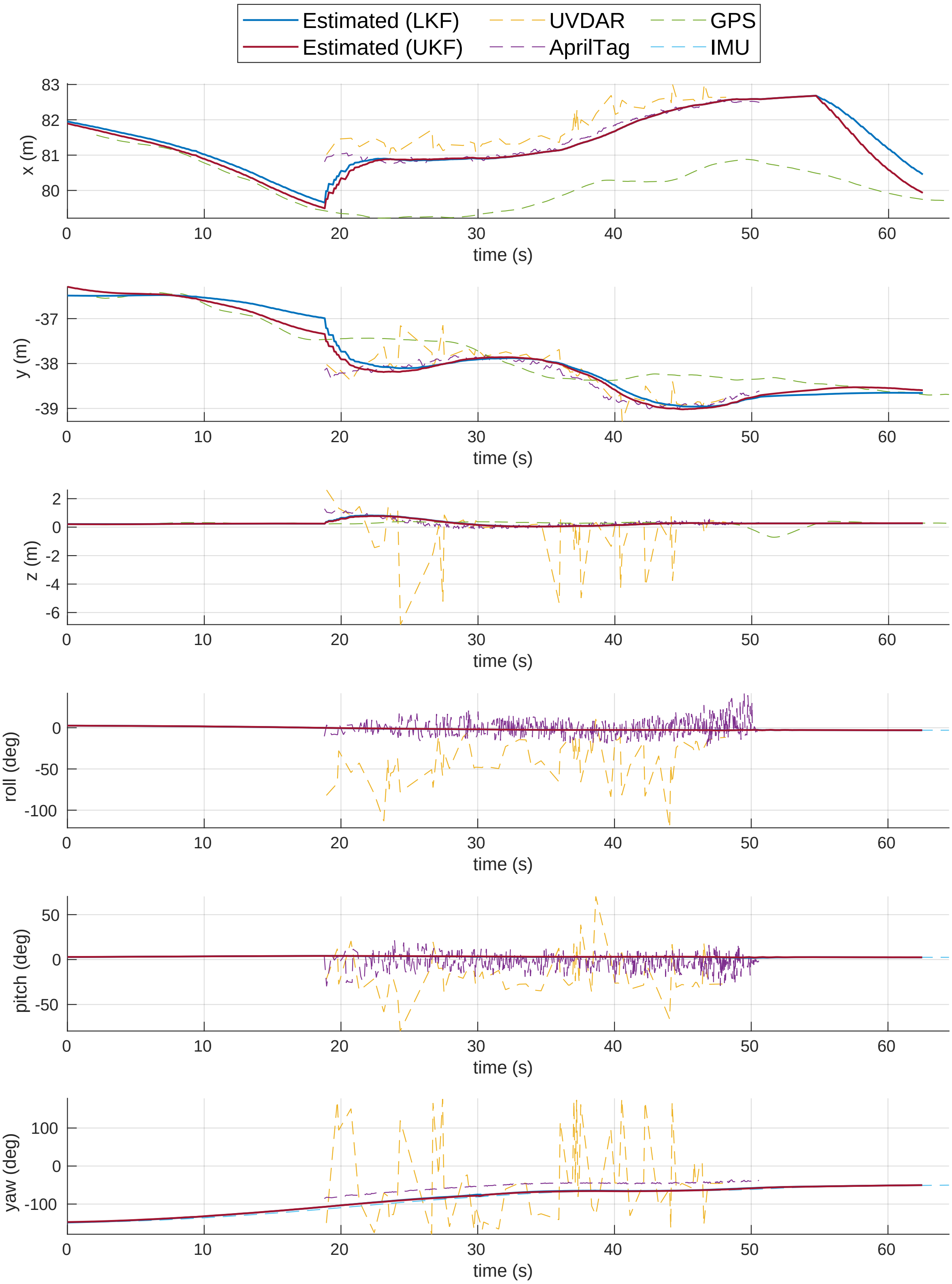
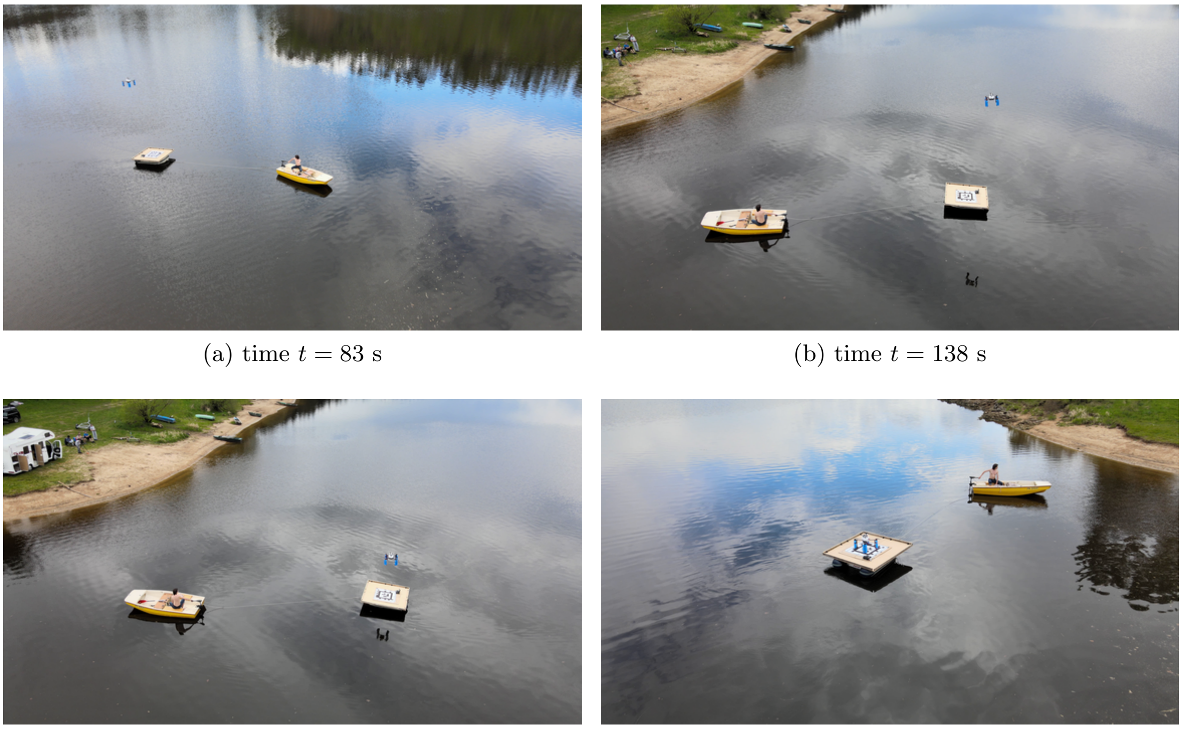


Figure 7.7: Estimation of the USV states ($x,\ y,\ z,\ \phi,\ \theta,\ \psi$) using the Linear Kalman filter (LKF) and Unscented Kalman filter (UKF) while the UAV landed on the USV.

7.3 Following and landing

This real-world experiment is combination of previous two scenarios Sec. 7.1 and 7.2. Firstly, the UAV flew above the USV and estimated the USV states only using data received from the USV onboard sensors — the GPS and the IMU. After the UAV onboard sensors (UVDAR system and AprilTag detector) detected desired markers in their camera frames, the measurements from these UAV onboard sensors improved the estimations of the USV states. The UAV followed the USV for 98 s using the estimated and predicted USV states while the USV was moving. The USV was slowly towed by another boat. While following the USV, the UAV was looking for good conditions to land on the USV and canceled the landing maneuver whenever the landing was inconvenient for the UAV. At time $t = 142$ s, the UAV successfully landed on the USV. The whole experiment is captured on video[1].

The estimations of the USV states during this real-world experiment are presented in Fig. 7.9. The graphs of the USV states show that the Linear and Unscented Kalman filter estimated the motion of the USV nicely. The UAV onboard sensors started providing data at time $t = 45$ s and improved an estimation of the USV states which is obvious especially in graph of the USV state z. As in the previous two real-world experiments (Sec. 7.1 and 7.2), the UVDAR measurements of z, roll, pitch, yaw and AprilTag measurements of roll and pitch were very noisy. However, the Linear and Unscented Kalman filter filtered the noise out and provided smooth estimation of the USV states. The snapshots of this real-world experiment are shown in Fig. 7.8.



(a) time $t = 83$ s

(b) time $t = 138$ s

(c) time $t = 141$ s

(d) time $t = 162$ s

Figure 7.8: Snapshots from the real-world experiment in which the UAV followed the USV and then landed on it.

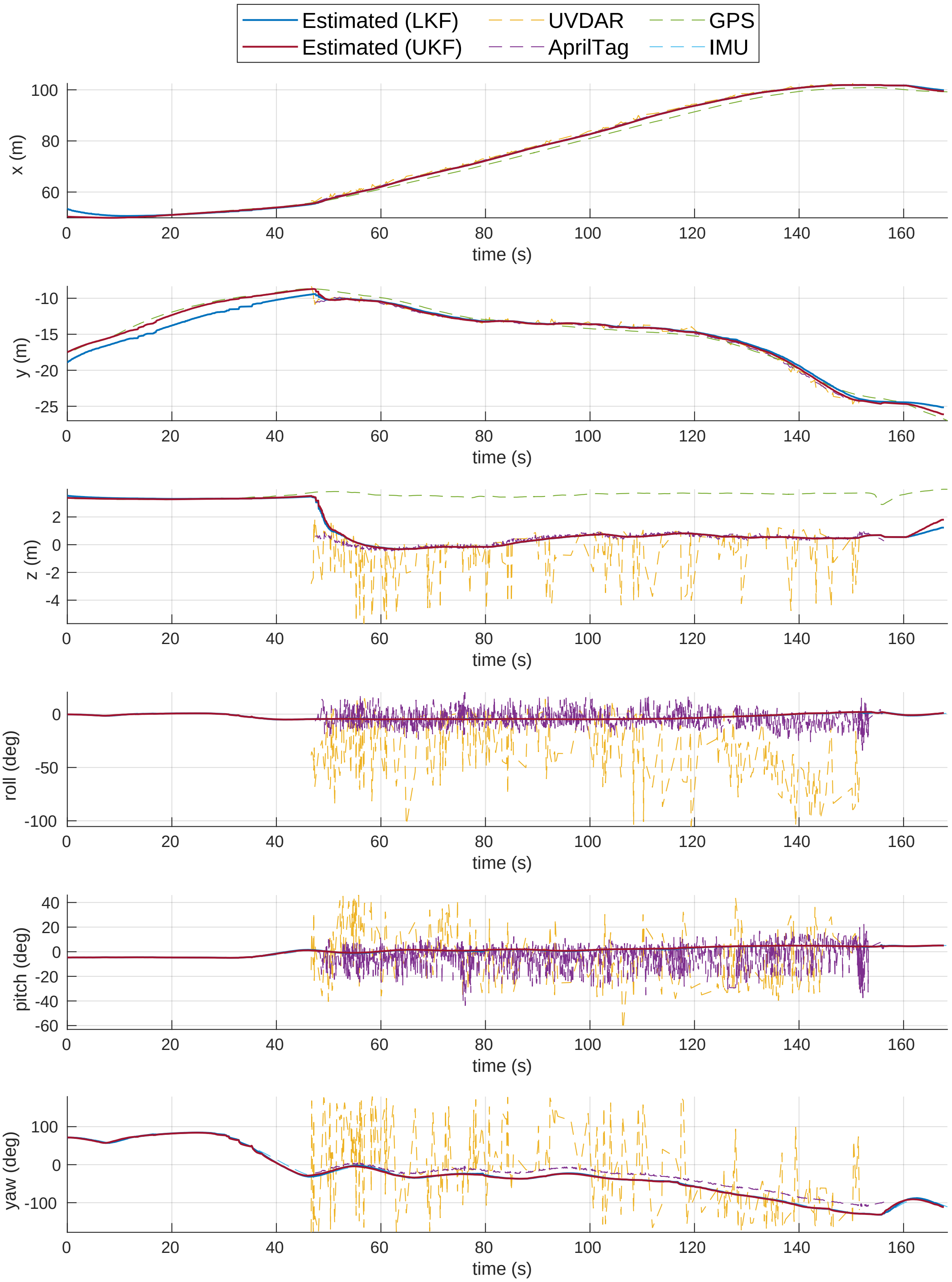


Figure 7.9: Estimation of the USV states (x, y, z, ϕ, θ, ψ) using the Linear Kalman filter (LKF) and Unscented Kalman filter (UKF) while the UAV followed the USV and then landed on it.

Chapter 8

Conclusion

In this thesis, an UAV onboard estimation system of the USV states has been developed. The estimation system is able to estimate and predict the USV states on a wavy water surface. Two types of Kalman filters were selected as state estimators — the Linear Kalman filter and the Unscented Kalman filter. The Unscented Kalman filter uses the designed nonlinear USV model extended by a nonlinear model of waves. The nonlinear USV model, including the wave model, was linearized and provided to the Linear Kalman filter. A precise robust estimation is performed by fusing data from multiple sensors. The first group of sensors consisting of the GPS and the IMU is placed directly on the USV. The data from these sensors is sent to the UAV using a wireless communication link. Two sensors — the UVDAR system and the AprilTag detector — located onboard the UAV form the second group. The selected sensors enable estimations of all considered USV states. The estimation system was integrated into a UAV control system developed by the Multi-robot Systems group at Czech Technical University (CTU) in Prague. Performance of the presented estimation system was verified in the realistic robotic simulator Gazebo. The results were analyzed in detail, and the estimation performances of the Linear Kalman filter and the Unscented Kalman filter were compared. The presented mathematical USV model and the estimated USV states were also used to predict future USV states. The developed estimation system was also verified by conducting real-world experiments. The performance of the estimation system was demonstrated in different application scenarios in which the UAV followed the USV and then landed on it. The multimedia materials complementing this thesis are available at: `http://mrs.felk.cvut.cz/theses/novak2022`.

The assignment of this thesis has been successfully completed. To summarize, the following points have been accomplished.

- The Linear Kalman filter and the Unscented Kalman filter selected as state estimators have been studied in Chap. 2. This chapter also contains methods to verify each filter's performance.
- A nonlinear model of the USV used by the Unscented Kalman filter is presented in Chap. 3. A linear model of the USV used by the Linear Kalman filter is derived from the presented nonlinear model. Both models are then extended by the wave model.
- The selected sensors used by the estimation system are presented in Chap. 4.
- The Chap. 5 focuses on the integration of the estimation system into the MRS system. This chapter also described the simulation environment in the Gazebo simulator used for verification.
- The verification of the designed estimation system is presented in Chap. 6. The estimation performance of the Linear Kalman filter and the Unscented Kalman filter are analyzed and compared.
- The estimation system was also verified by conducting real-world experiments that are presented in Chap. 7.

8.1 Future work

Future research can head in several directions. For instance, different state estimation methods can be used, e.g., Cubature Kalman filter, Particle filter, and Adaptive filter. Researchers can also focus on data-driven state estimation methods based on neural network learning. The estimation system can be extended by a real-time system to determine the wave parameters that can be directly put into the wave model. A different USV model influenced by waves can be designed to describe reality more precisely and predict future states more accurately.

Chapter 9

References

[1] T. Baca, M. Petrlik, M. Vrba, *et al.*, "The mrs uav system: Pushing the frontiers of reproducible research, real-world deployment, and education with autonomous unmanned aerial vehicles," *Journal of Intelligent & Robotic Systems*, vol. 102, no. 26, pp. 1–28, 1 May 2021.

[2] M. Petrlik, T. Baca, D. Hert, M. Vrba, T. Krajnik, and M. Saska, "A robust uav system for operations in a constrained environment," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2169–2176, Apr. 2020.

[3] P. Petracek, V. Kratky, and M. Saska, "Dronument: System for reliable deployment of micro aerial vehicles in dark areas of large historical monuments," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2078–2085, Apr. 2020.

[4] D. C. Duives, W. Daamen, and S. Hoogendoorn, "Trajectory analysis of pedestrian crowd movements at a dutch music festival," in *Pedestrian and Evacuation Dynamics 2012*, U. Weidmann, U. Kirsch, and M. Schreckenberg, Eds., Springer International Publishing, 2014, pp. 151–166.

[5] J. Grzybowski, K. Latos, and R. Czyba, "Low-cost autonomous uav-based solutions to package delivery logistics," in *Advanced, Contemporary Control*, A. Bartoszewicz, J. Kabzinski, and J. Kacprzyk, Eds., Springer International Publishing, 2020, pp. 500–507.

[6] V. Spurny, V. Pritzl, V. Walter, *et al.*, "Autonomous firefighting inside buildings by an unmanned aerial vehicle," *IEEE Access*, vol. 9, pp. 15 872–15 890, Jan. 2021.

[7] S. P. Yeong, L. M. King, and S. S. Dol, "A review on marine search and rescue operations using unmanned aerial vehicles," *International Journal of Marine and Environmental Sciences*, vol. 9, no. 2, pp. 396–399, 2015.

[8] S. Kako, S. Morita, and T. Taneda, "Estimation of plastic marine debris volumes on beaches using unmanned aerial vehicles and image processing based on deep learning," *Marine Pollution Bulletin*, vol. 155, p. 111 127, 2020.

[9] A. S. Aniceto, M. Biuw, U. Lindstrom, S. A. Solbo, F. Broms, and J. Carroll, "Monitoring marine mammals using unmanned aerial vehicles: Quantifying detection certainty," *Ecosphere*, vol. 9, no. 3, e02122, 2018.

[10] A. Hodgson, N. Kelly, and D. Peel, "Unmanned aerial vehicles (uavs) for surveying marine fauna: A dugong case study," *PLOS ONE*, vol. 8, no. 11, Nov. 2013.

[11] R. Murphy, S. Stover, K. Pratt, and C. Griffin, "Cooperative damage inspection with unmanned surface vehicle and micro unmanned aerial vehicle at hurricane wilma," in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2006, pp. 9–9.

[12] M. Lindemuth, R. Murphy, E. Steimle, *et al.*, "Sea robot-assisted inspection," *IEEE Robotics Automation Magazine*, vol. 18, no. 2, pp. 96–107, 2011.

[13] J. Ross, J. Lindsay, E. Gregson, A. Moore, J. Patel, and M. Seto, "Collaboration of multi-domain marine robots towards above and below-water characterization of floating targets," in *2019 IEEE International Symposium on Robotic and Sensors Environments (ROSE)*, 2019, pp. 1–7.

[14] T. I. Fossen, *Handbook of Marine Craft Hydrodynamics and Motion Control*, First Edition. United Kingdom: John Wiley & Sons, 2011.

[15] Y. Han and W. Ma, "Automatic monitoring of water pollution based on the combination of uav and usv," in *2021 IEEE 4th International Conference on Electronic Information and Communication Technology (ICEICT)*, 2021, pp. 420–424.

[16] R. R. Murphy, E. Steimle, C. Griffin, C. Cullins, M. Hall, and K. Pratt, "Cooperative use of unmanned sea surface and micro aerial vehicles at hurricane wilma," *Journal of Field Robotics*, vol. 25, no. 3, pp. 164–180, 2008.

[17] D. Steenken, T. Winter, and U. T. Zimmermann, "Stowage and transport optimization in ship planning," in *Online Optimization of Large Scale Systems*, M. Grotschel, S. O. Krumke, and J. Rambau, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2001, pp. 731–745.

[18] F. F. Ramirez, D. S. Benitez, E. B. Portas, and J. A. L. Orozco, "Coordinated sea rescue system based on unmanned air vehicles and surface vessels," in *OCEANS 2011 IEEE - Spain*, 2011, pp. 1–10.

[19] S. Young, J. Peschel, G. Penny, S. Thompson, and V. Srinivasan, "Robot-assisted measurement for hydrologic understanding in data sparse regions," *Water*, vol. 9, no. 7, 2017.

[20] T. Pastore and V. Djapic, "Improving autonomy and control of autonomous surface vehicles in port protection and mine countermeasure scenarios," *Journal of Field Robotics*, vol. 27, no. 6, pp. 903–914, 2010.

[21] M. F. Ozkan, L. R. G. Carrillo, and S. A. King, "Rescue boat path planning in flooded urban environments," in *2019 IEEE International Symposium on Measurement and Control in Robotics (ISMCR)*, 2019, B2-2-1-B2-2–9.

[22] *The otter*. [Online]. Available: `https://www.maritimerobotics.com/otter` (visited on 03/14/2022).

[23] V. Walter, N.Staub, M. Saska, and A. Franchi, "Mutual localization of uavs based on blinking ultraviolet markers and 3d time-position hough transform," in *14th IEEE International Conference on Automation Science and Engineering (CASE 2018)*, 2018.

[24] T. I. Fossen, *Marine Control Systems Guidance, Navigation, and Control of Ships, Rigs and Underwater Vehicles*, 1st edition. Trondheim: Marine Cybernetics, 2002.

[25] R. Polvara, S. Sharma, J. Wan, A. Manning, and R. Sutton, "Vision-based autonomous landing of a quadrotor on the perturbed deck of an unmanned surface vehicle," *Drones*, vol. 2, no. 2, 2018.

[26] V. Djapic, C. Prijic, and F. Bogart, "Autonomous takeoff & landing of small uas from the usv," in *OCEANS 2015 - MTS/IEEE Washington*, 2015, pp. 1–8.

[27] M. Aissi, Y. Moumen, J. Berrich, T. Bouchentouf, M. Bourhaleb, and M. Rahmoun, "Autonomous solar usv with an automated launch and recovery system for uav: State of the art and design," in *2020 IEEE 2nd International Conference on Electronics, Control, Optimization and Computer Science (ICECOCS)*, 2020, pp. 1–6.

[28] C. G. Grlj, N. Krznar, and M. Pranjic, "A decade of uav docking stations: A brief overview of mobile and fixed landing platforms," *Drones*, vol. 6, no. 1, 2022.

[29] G. Shao, Y. Ma, R. Malekian, X. Yan, and Z. Li, "A novel cooperative platform design for coupled usv–uav systems," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 9, pp. 4913–4922, 2019.

[30] L. Persson and B. Wahlberg, "Model predictive control for autonomous ship landing in a search and rescue scenario," in *AIAA Scitech 2019 Forum*.

[31] T. I. Fossen, *Guidance and control of ocean vehicles*. New York: Wiley, c1994.

[32] M. Caccia, R. Bono, G. Bruzzone, *et al.*, "Sampling sea surfaces with sesamo: An autonomous craft for the study of sea-air interactions," *IEEE Robotics Automation Magazine*, vol. 12, no. 3, pp. 95–105, 2005.

[33] J. Curcio, J. Leonard, and A. Patrikalakis, "Scout - a low cost autonomous surface platform for research in cooperative autonomy," in *Proceedings of OCEANS 2005 MTS/IEEE*, 2005, 725–729 Vol. 1.

[34] P. Mahacek, R. Kobashigawa, A. Schooley, and C. Kitts, "The wasp: An atonomous surface vessel for the university of alaska," vol. 2005, Feb. 2005, 2282–2291 Vol. 3.

[35] P. F. Rynne and K. D. von Ellenrieder, "A wind and solar-powered autonomous surface vehicle for sea surface measurements," in *OCEANS 2008*, 2008, pp. 1–6.

[36] W. Naeem, T. Xu, R. Sutton, and J. Chudley, "Design of an unmanned catamaran with pollutant tracking and surveying capabilities," in *2006 UKACC Control 2006 Mini Symposia*, 2006, pp. 99–113.

[37] C. Specht, E. Switalski, and M. Specht, "Application of an autonomous/unmanned survey vessel (asv/usv) in bathymetric measurements," *Polish Maritime Research*, vol. 24, no. 3, pp. 36–44, 2017.

[38] P. Mahacek, T. Berk, A. Casanova, C. Kitts, W. Kirkwood, and G. Wheat, "Development and initial testing of a swath boat for shallow-water bathymetry," in *OCEANS 2008*, 2008, pp. 1–6.

[39] T. Baca, D. Hert, G. Loianno, M. Saska, and V. Kumar, "Model predictive trajectory tracking and collision avoidance for reliable outdoor deployment of unmanned aerial vehicles," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2018, pp. 1–8.

[40] A. P. Colefax, P. A. Butcher, and B. P. Kelaher, "The potential for unmanned aerial vehicles (UAVs) to conduct marine fauna surveys in place of manned aircraft," *ICES Journal of Marine Science*, vol. 75, no. 1, pp. 1–8, Jun. 2017.

[41] D. Ventura, A. Bonifazi, M. F. Gravina, A. Belluscio, and G. Ardizzone, "Mapping and classification of ecologically sensitive marine habitats using unmanned aerial vehicle (uav) imagery and object-based image analysis (obia)," *Remote Sensing*, vol. 10, no. 9, 2018.

[42] C. A. Trasvina-Moreno, R. Blasco, A. Marco, R. Casas, and A. Trasvina-Castro, "Unmanned aerial vehicle based wireless sensor network for marine-coastal environment monitoring," *Sensors*, vol. 17, no. 3, 2017.

[43] F. Muttin, "Umbilical deployment modeling for tethered uav detecting oil pollution from ship," *Applied Ocean Research*, vol. 33, no. 4, pp. 332–343, 2011.

[44] Z.-C. Xu, B.-B. Hu, B. Liu, X. Wang, and H.-T. Zhang, "Vision-based autonomous landing of unmanned aerial vehicle on a motional unmanned surface vessel," in *2020 39th Chinese Control Conference (CCC)*, 2020, pp. 6845–6850.

[45] M. H. Assaf, E. M. Petriu, and V. Groza, "Ship track estimation using gps data and kalman filter," in *2018 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, 2018, pp. 1–6.

[46] A. Ccolque-Churquipa, J. C. Cutipa-Luque, and D. Y. Aco-Cardenas, "Implementation of a measurement system for the attitude, heading and position of a usv using imus and gps," in *2018 IEEE ANDESCON*, 2018, pp. 1–6.

[47] A. Deep, M. Mittal, and V. Mittal, "Application of kalman filter in gps position estimation," in *2018 IEEE 8th Power India International Conference (PIICON)*, 2018, pp. 1–5.

[48] J. Kwak and Y. Sung, "Autonomous uav flight control for gps-based navigation," *IEEE Access*, vol. 6, pp. 37 947–37 955, 2018.

[49] Q. V. Tran and H.-S. Ahn, "Multi-agent localization of a common reference coordinate frame: An extrinsic approach," *IFAC-PapersOnLine*, vol. 52, no. 20, pp. 67–72, 2019, 8th IFAC Workshop on Distributed Estimation and Control in Networked Systems NECSYS 2019.

[50] E. Olson, "AprilTag: A robust and flexible visual fiducial system," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, May 2011, pp. 3400–3407.

[51] J. Wang and E. Olson, "AprilTag 2: Efficient and robust fiducial detection," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct. 2016.

[52] M. Krogius, A. Haggenmiller, and E. Olson, “Flexible layouts for fiducial tags,” in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct. 2019.

[53] V. Walter, M. Saska, and A. Franchi, “Fast mutual relative localization of uavs using ultraviolet led markers,” in *2018 International Conference on Unmanned Aircraft System (ICUAS 2018)*, 2018.

[54] X. Xiao, J. Dufek, T. Woodbury, and R. Murphy, “Uav assisted usv visual navigation for marine mass casualty incident response,” in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, pp. 6105–6110.

[55] M. Vrba and M. Saska, “Marker-less micro aerial vehicle detection and localization using convolutional neural networks,” *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2459–2466, Apr. 2020.

[56] R. E. Kalman, “A new approach to linear filtering and prediction problems,” *Journal of Basic Engineering*, vol. 82, no. 1, pp. 35–45, 1960.

[57] S. Haykin, *Kalman Filtering and Neural Networks.* New York, USA: John Wiley & Sons, Inc, 2001.

[58] S. J. Julier and J. K. Uhlmann, “New extension of the Kalman filter to nonlinear systems,” in *Signal Processing, Sensor Fusion, and Target Recognition VI*, I. Kadar, Ed., International Society for Optics and Photonics, vol. 3068, SPIE, 1997, pp. 182–193.

[59] D. Malyuta, C. Brommer, D. Hentzen, T. Stastny, R. Siegwart, and R. Brockers, “Long-duration fully autonomous operation of rotorcraft unmanned aerial systems for remote-sensing data acquisition,” *Journal of Field Robotics*, arXiv:1908.06381, Aug. 2019.

[60] M. I. Ribeiro, “Kalman and extended kalman filters: Concept, derivation and properties,” *Institute for Systems and Robotics*, vol. 43, p. 46, 2004.

[61] T. I. Fossen and T. Perez, “Kalman filtering for positioning and heading control of ships and offshore rigs,” *IEEE Control Systems Magazine*, vol. 29, no. 6, pp. 32–46, 2009.

[62] D. S. dos Santos, C. L. Nascimento, and W. C. Cunha, “Autonomous navigation of a small boat using imu/gps/digital compass integration,” in *2013 IEEE International Systems Conference (SysCon)*, 2013, pp. 468–474.

[63] W. Liu, Y. Liu, and R. Bucknall, “A robust localization method for unmanned surface vehicle (usv) navigation using fuzzy adaptive kalman filtering,” *IEEE Access*, vol. 7, pp. 46 071–46 083, 2019.

[64] P. S. Maybeck, *Stochastic models, estimation, and control*, ser. Mathematics in Science and Engineering. 1979, vol. 141.

[65] E. Wit, E. van den Heuvel, and J.-W. Romeijn, “‘all models are wrong...’ An introduction to model uncertainty,” *Statistica Neerlandica*, vol. 66, no. 3, pp. 217–236, 2012.

[66] A. H. Jazwinski, *Stochastic processes and filtering theory*, 1st. Mineola: Dover Publications, 2007.

[67] E. Wan and R. V. D. Merwe, “The unscented kalman filter for nonlinear estimation,” in *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No.00EX373)*, IEEE, 2000, pp. 153–158.

[68] S. J. Julier and J. K. Uhlmann, “A general method for approximating nonlinear transformations of probability distributions,” 1996.

[69] S. J. Julier, J. K. Uhlmann, and H. F. Durrant-Whyte, “A new method for the nonlinear transformation of means and covariances in filters and estimators,” *IEEE Trans. Autom. Control.*, vol. 45, pp. 477–482, 2000.

[70] W. Press, W. Vetterling, S. A. Teukolsky, and B. P. Flannery, *Numerical recipes in C the art of scientific computing*, 2nd. Cambridge: University Press, 1996.

[71] H. Kaur and J. S. Sahambi, “Vehicle tracking in video using fractional feedback kalman filter,” *IEEE Transactions on Computational Imaging*, vol. 2, no. 4, pp. 550–561, 2016.

[72] V. Sangwan, R. Kumar, and A. K. Rathore, “State-of-charge estimation for li-ion battery using extended kalman filter (ekf) and central difference kalman filter (cdkf),” in *2017 IEEE Industry Applications Society Annual Meeting*, 2017, pp. 1–6.

[73] A. Evangelidis and D. Parker, “Quantitative verification of kalman filters,” *Formal Aspects of Computing*, vol. 33, no. 4-5, pp. 669–693, 2021.

[74] I. Reidr, “Lecture notes - estimation ii,” 2001. [Online]. Available: `https://www.robots.ox.ac.uk/~ian/Teaching/Estimation/LectureNotes2.pdf` (visited on 03/14/2022).

[75] P. H. LeBlond and L. A. Mysak, *Waves in the Ocean.* Elsevier, 1981.

[76] J. Diebel, “Representing attitude: Euler angles, unit quaternions, and rotation vectors,” *Matrix*, vol. 58, no. 15-16, pp. 1–35, 2006.

[77] F. E. Cellier and E. Kofman, *Continuous system simulation.* New York: Springer, c2006.

[78] Z. Hurak, “Introduction to numerical simulation,” 2019. [Online]. Available: `https://moodle.fel.cvut.cz/pluginfile.php/210391/mod_resource/content/4/msd_11_intro_to_numerical_simulation_of_ODE.pdf` (visited on 03/30/2022).

[79] G. D. Quiroga and P. A. Ospina-Henao, “Dynamics of damped oscillations: Physical pendulum,” *European Journal of Physics*, vol. 38, no. 6, p. 065 005, Oct. 2017.

[80] B. D. O. Anderson and J. B. Moore, *Optimal filtering.* Englewood Cliffs, N.J.: Prentice-Hall, 1979.

[81] N. M. Drawil, H. M. Amar, and O. A. Basir, “Gps localization accuracy classification: A context-based approach,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 1, pp. 262–273, 2013.

[82] N. Ahmad, R. A. Raja Ghazilla, N. Khairi, and V. Kasi, “Reviews on various inertial measurement unit (imu) sensor applications,” *International Journal of Signal Processing Systems*, vol. 1, pp. 256–262, Jan. 2013.

[83] 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.

[84] A. R. Lopez, “Gps landing system reference antenna,” *IEEE Antennas and Propagation Magazine*, vol. 52, no. 1, pp. 104–113, 2010.

[85] N. Abdelkrim, N. Aouf, A. Tsourdos, and B. White, “Robust nonlinear filtering for ins/gps uav localization,” in *2008 16th Mediterranean Conference on Control and Automation*, 2008, pp. 695–702.

[86] W. Lewandowski, J. Azoubib, and W. Klepczynski, “Gps: Primary tool for time transfer,” *Proceedings of the IEEE*, vol. 87, no. 1, pp. 163–172, 1999.

[87] J. Han, Y. Cho, and J. Kim, “Coastal slam with marine radar for usv operation in gps-restricted situations,” *IEEE Journal of Oceanic Engineering*, vol. 44, no. 2, pp. 300–309, 2019.

[88] K. M. Ng, J. Johari, S. A. C. Abdullah, A. Ahmad, and B. N. Laja, “Performance evaluation of the rtk-gnss navigating under different landscape,” in *2018 18th International Conference on Control, Automation and Systems (ICCAS)*, 2018, pp. 1424–1428.

[89] S. Sukkarieh, E. Nebot, and H. Durrant-Whyte, “A high integrity imu/gps navigation loop for autonomous land vehicle applications,” *IEEE Transactions on Robotics and Automation*, vol. 15, no. 3, pp. 572–578, 1999.

[90] F. Caron, E. Duflos, D. Pomorski, and P. Vanheeghe, “Gps/imu data fusion using multisensor kalman filtering: Introduction of contextual aspects,” *Information Fusion*, vol. 7, no. 2, pp. 221–230, 2006.

[91] V. Walter, N. Staub, A. Franchi, and M. Saska, "Uvdar system for visual relative localization with application to leader–follower formations of multirotor uavs," *IEEE Robotics and Automation Letters*, vol. 4, no. 3, pp. 2637–2644, Jul. 2019.

[92] V. Walter, M. Vrba, and M. Saska, "On training datasets for machine learning-based visual relative localization of micro-scale UAVs," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, Aug. 2020, pp. 10 674–10 680.

[93] F. Novak, V. Walter, P. Petracek, T. Baca, and M. Saska, "Fast collective evasion in self-localized swarms of unmanned aerial vehicles," *Bioinspiration & Biomimetics*, vol. 16, no. 6, p. 066 025, Nov. 2021.

[94] P. Petracek, V. Walter, T. Baca, and M. Saska, "Bio-inspired compact swarms of unmanned aerial vehicles without communication and external localization," *Bioinspiration & Biomimetics*, vol. 16, no. 2, p. 026 009, Dec. 2020.

[95] A. Ahmad, V. Walter, P. Petracek, *et al.*, "Autonomous aerial swarming in gnss-denied environments with high obstacle density," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, Jun. 2021, pp. 570–576.

[96] C. Brommer, D. Malyuta, D. Hentzen, and R. Brockers, "Long-duration autonomy for small rotorcraft UAS including recharging," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, IEEE, Oct. 2018, arXiv:1810.05683.

[97] M. Quigley, K. Conley, B. Gerkey, *et al.*, "Ros: An open-source robot operating system," in *ICRA workshop on open source software*, Kobe, Japan, vol. 3, 2009, p. 5.

[98] N. Koenig and A. Howard, "Design and use paradigms for gazebo, an open-source multi-robot simulator," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Sendai, Japan, Sep. 2004, pp. 2149–2154.

[99] B. Bingham, C. Aguero, M. McCarrin, *et al.*, "Toward maritime robotic simulation in gazebo," in *Proceedings of MTS/IEEE OCEANS Conference*, Seattle, WA, Oct. 2019.

[100] D. Sundararajan, *The discrete Fourier transform: theory, algorithms and applications.* World Scientific, 2001.

Appendices

Appendix A

Verification

GPS – Linear Kalman filter verification – position

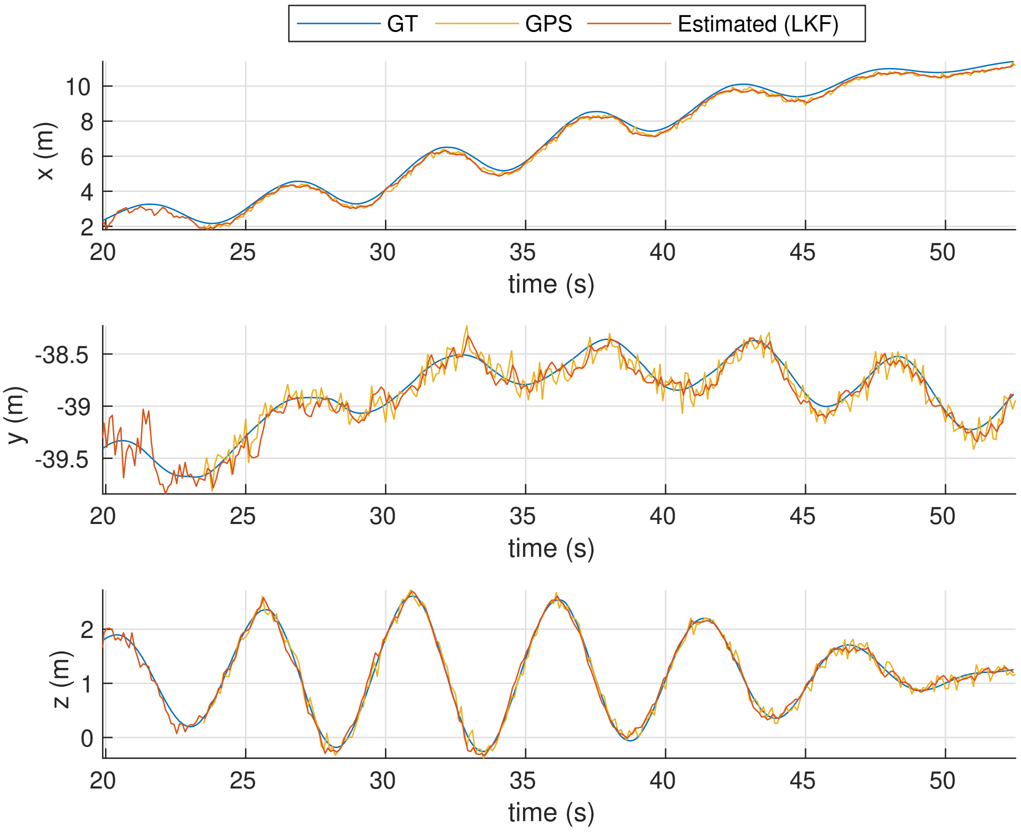


Figure A.1: Estimated states $(x,\ y,\ z)$ by the Linear Kalman filter using the GPS measurements.

IMU – Linear Kalman filter verification – orientation

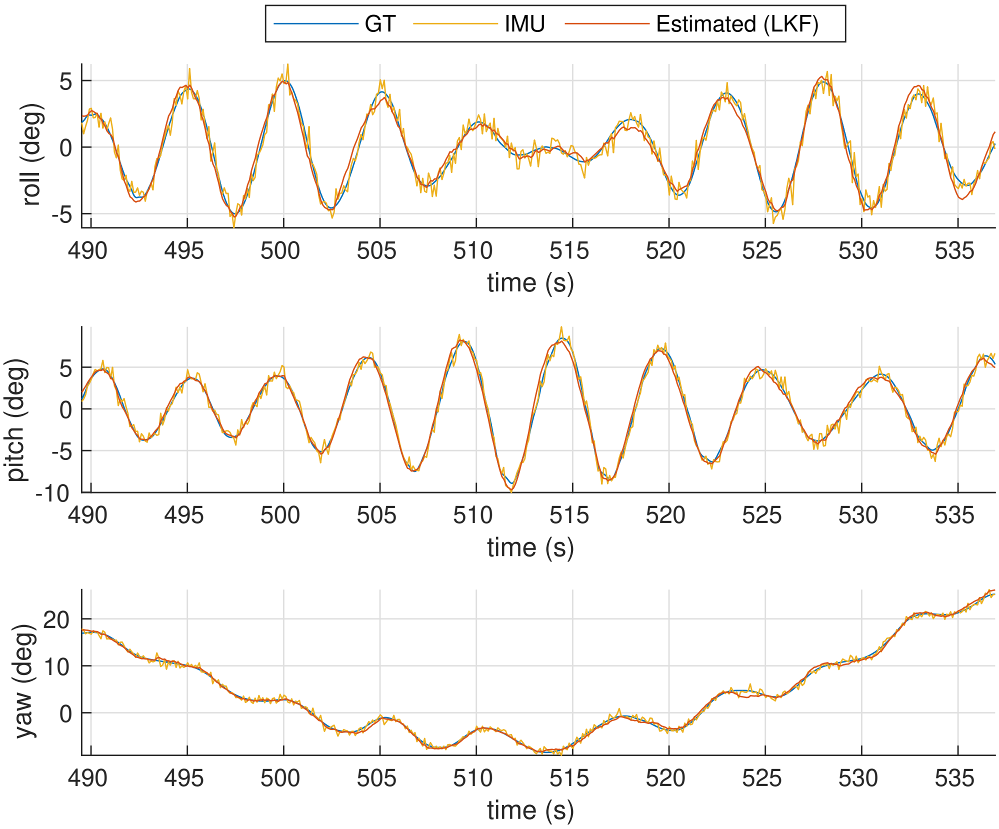
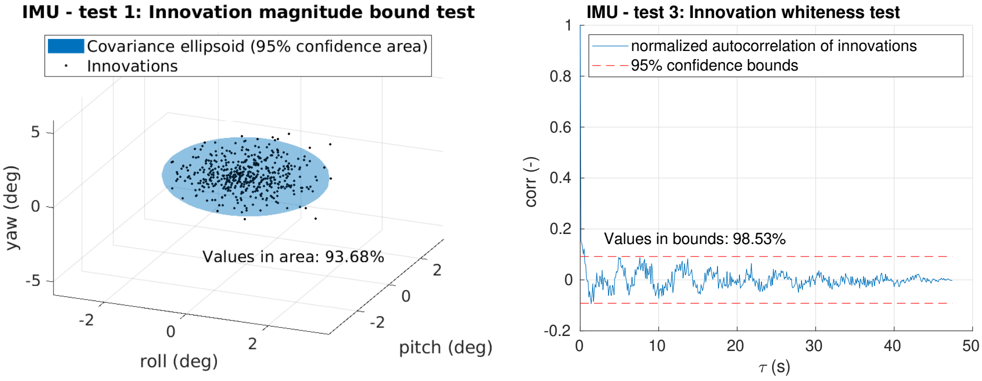


Figure A.2: Estimated states (ϕ, θ, ψ) by the Linear Kalman filter using the IMU measurements.



(a) Innovation test 1.

(b) Innovation test 3.

Figure A.3: Innovation test 1 and test 3 of the Linear Kalman filter for states (ϕ, θ, ψ) using the IMU measurements.

UVDAR – Linear Kalman filter verification – orientation

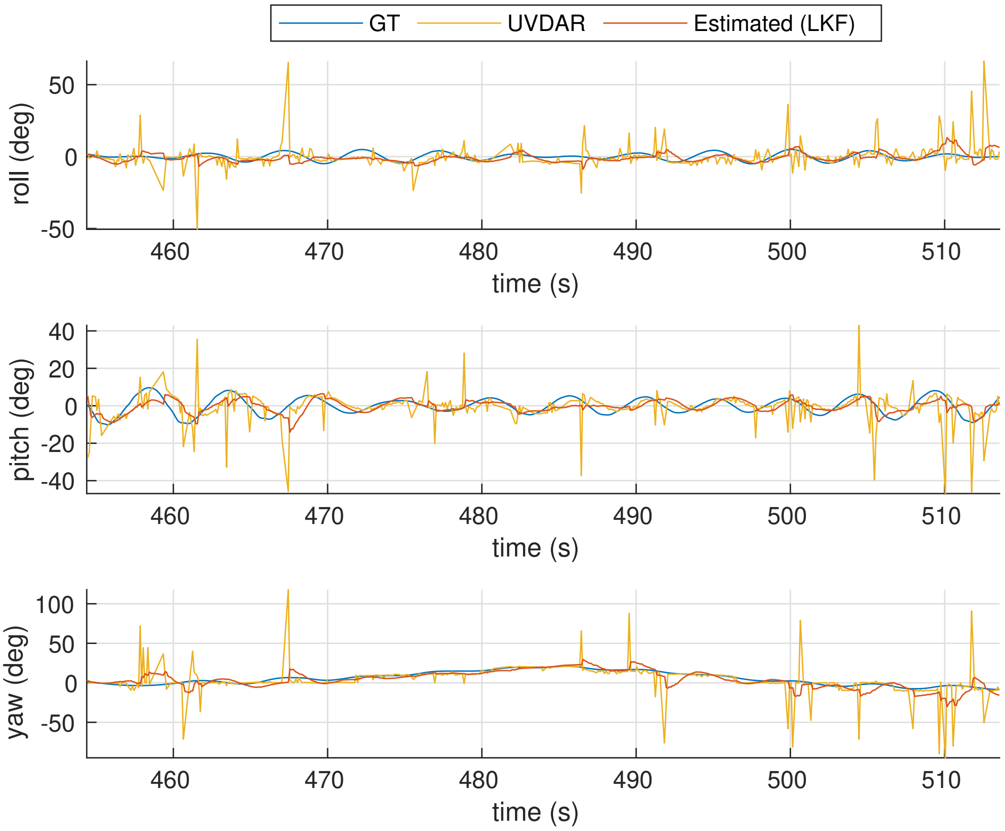
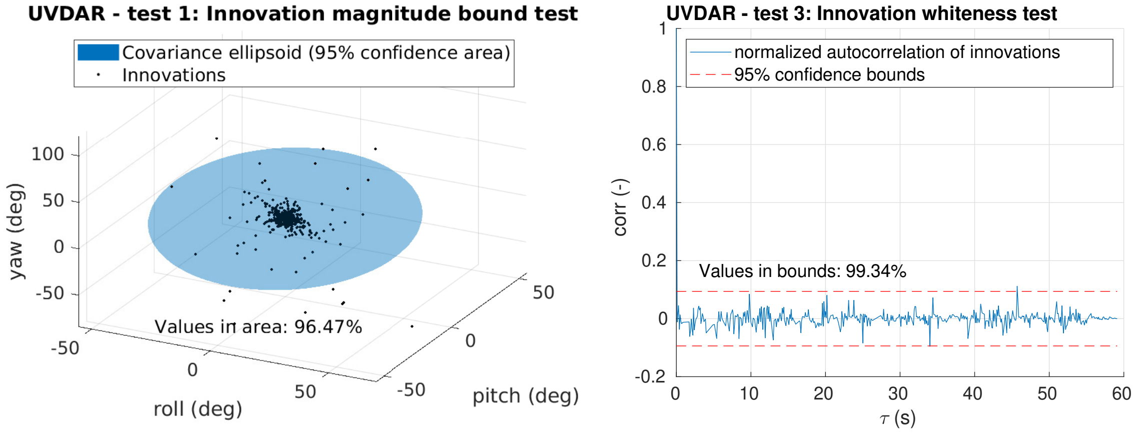


Figure A.4: Estimated states ($\phi,\ \theta,\ \psi$) by the Linear Kalman filter using the UVDAR measurements.



(a) Innovation test 1.

(b) Innovation test 3.

Figure A.5: Innovation test 1 and test 3 of the Linear Kalman filter for states ($\phi,\ \theta,\ \psi$) using the UVDAR measurements.

AprilTag – Linear Kalman filter verification – position

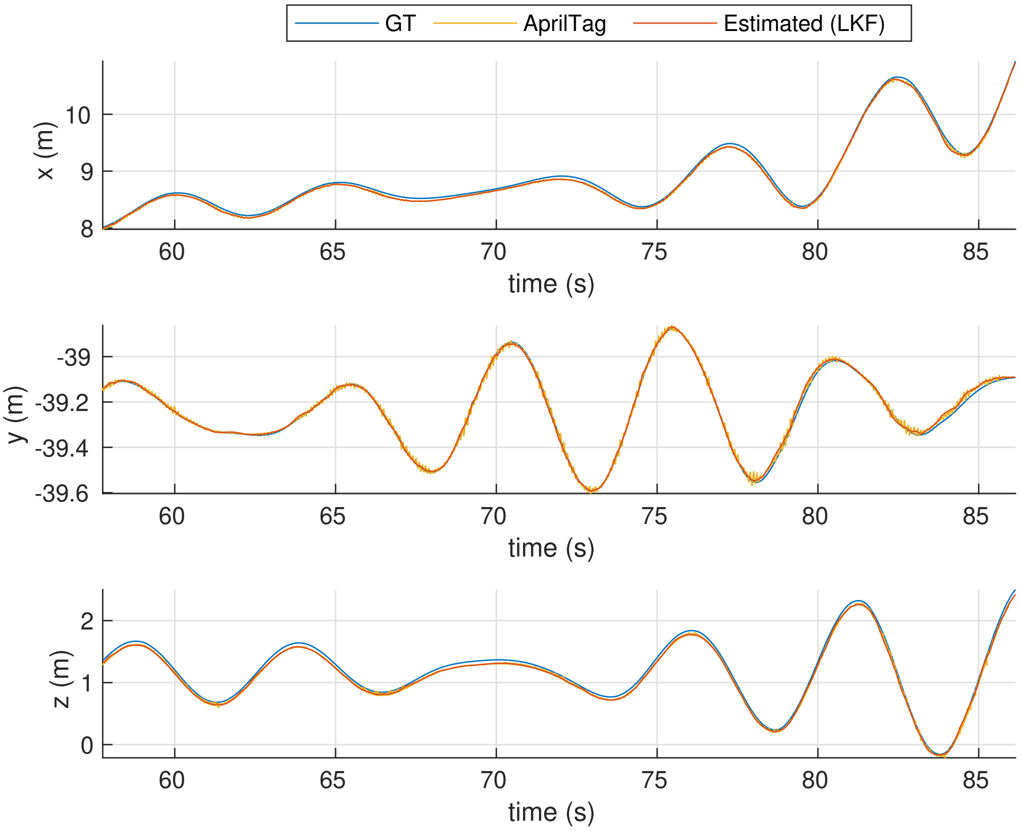
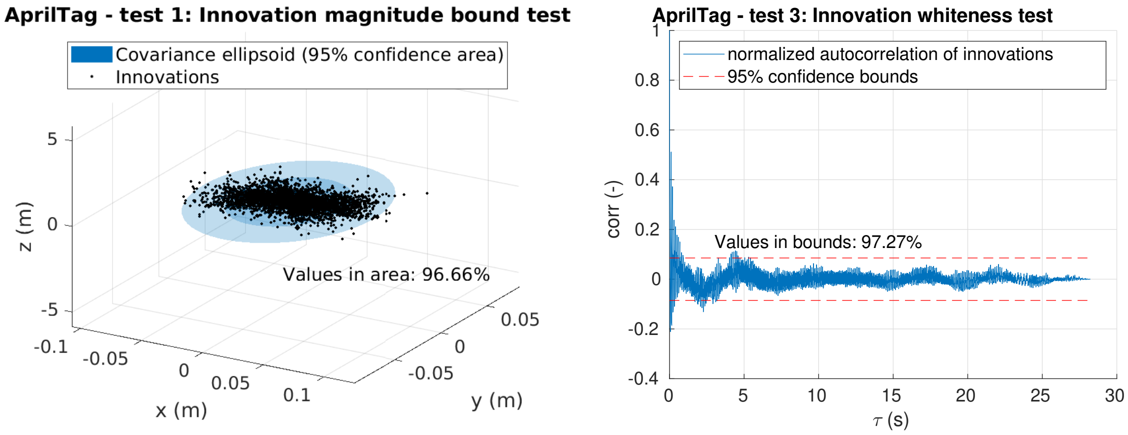


Figure A.6: Estimated states $(x,\ y,\ z)$ by the Linear Kalman filter using the AprilTag measurements.



(a) Innovation test 1.

(b) Innovation test 3.

Figure A.7: Innovation test 1 and test 3 of the Linear Kalman filter for states $(x,\ y,\ z)$ using the AprilTag measurements.

AprilTag – Linear Kalman filter verification – orientation

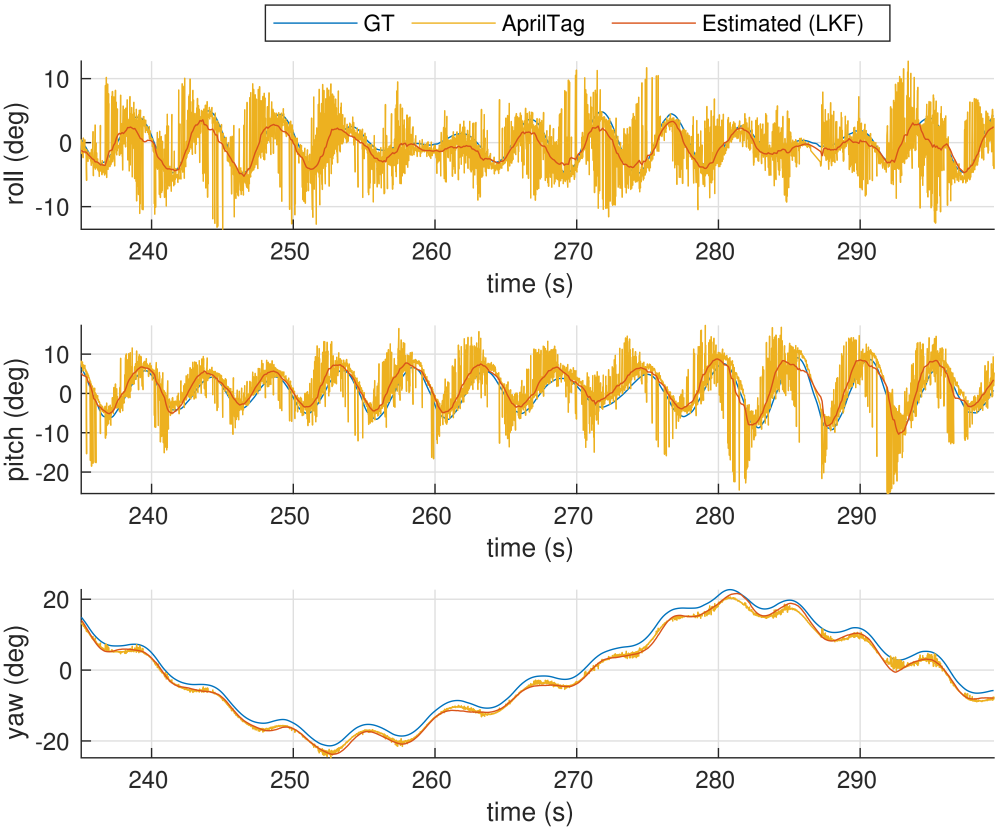
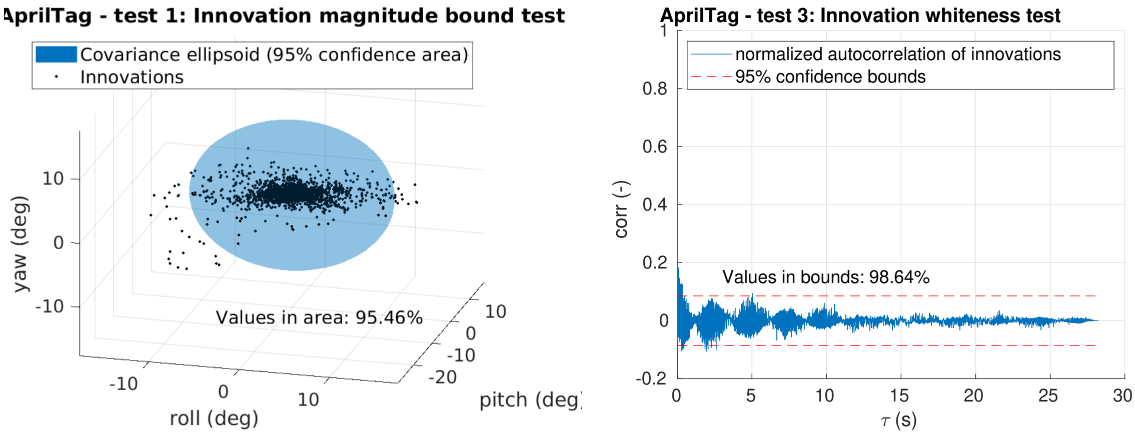


Figure A.8: Estimated states ($\phi,\ \theta,\ \psi$) by the Linear Kalman filter using the AprilTag measurements.



(a) Innovation test 1.

(b) Innovation test 3.

Figure A.9: Innovation test 1 and test 3 of the Linear Kalman filter for states ($\phi,\ \theta,\ \psi$) using the AprilTag measurements.

GPS – Unscented Kalman filter verification – position

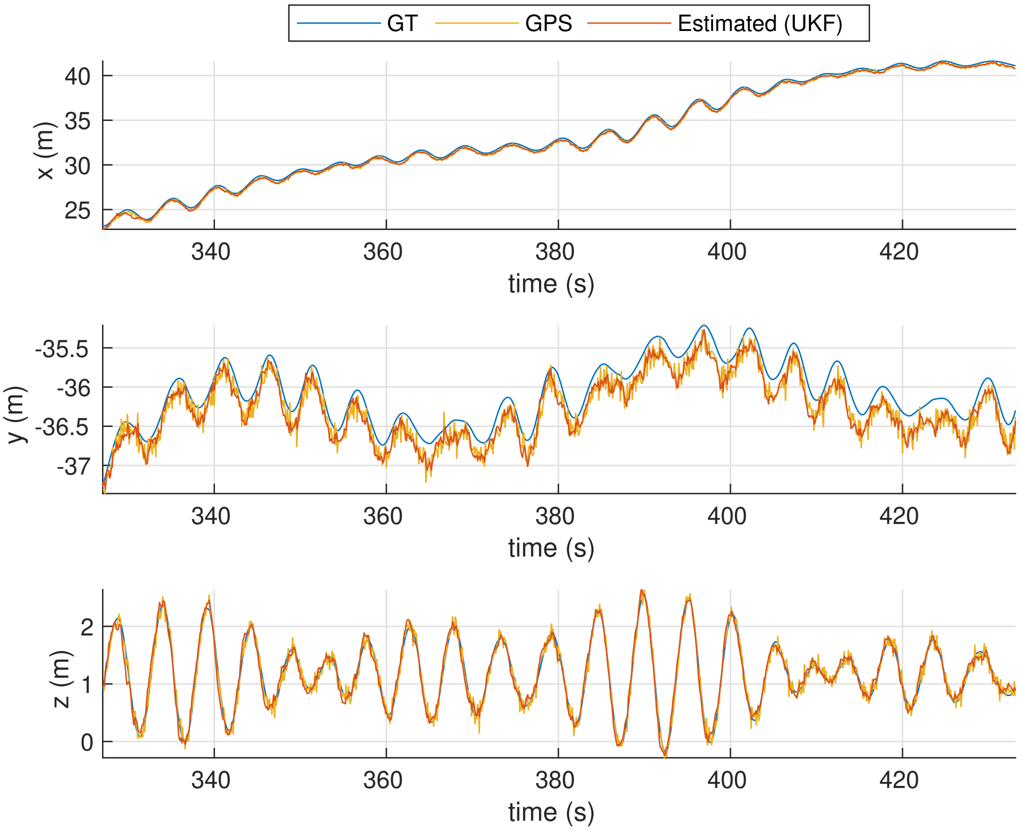


Figure A.10: Estimated states $(x,\ y,\ z)$ by the Unscented Kalman filter using the GPS measurements.

IMU – Unscented Kalman filter verification – orientation

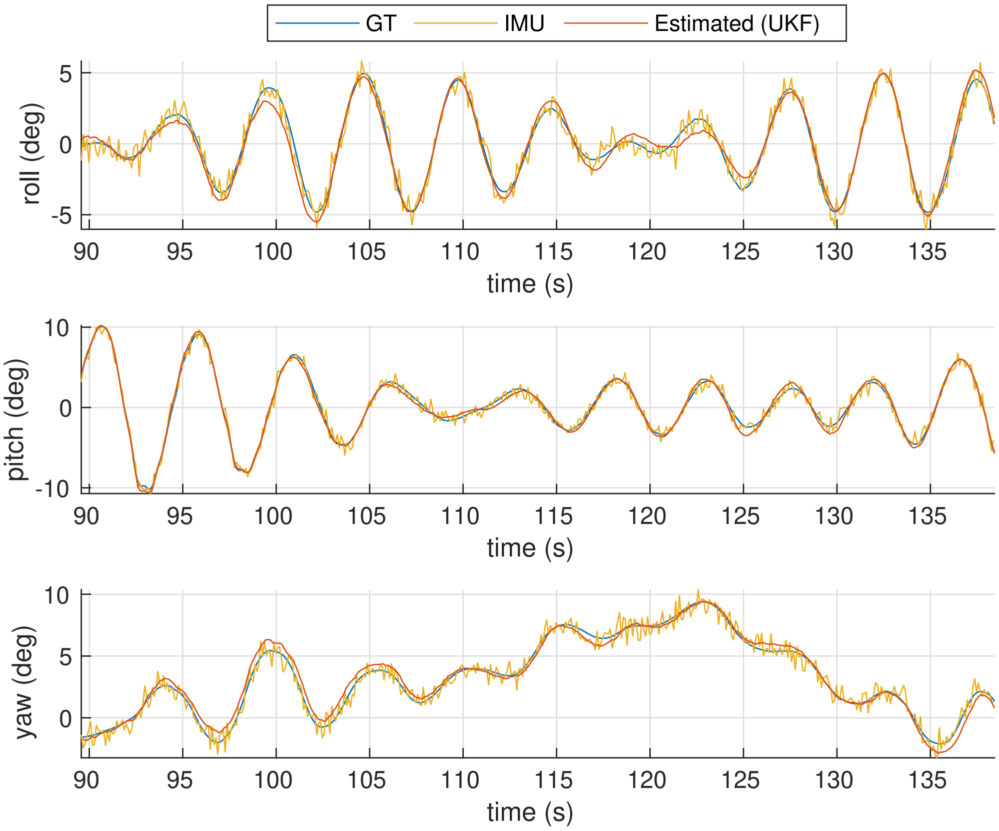
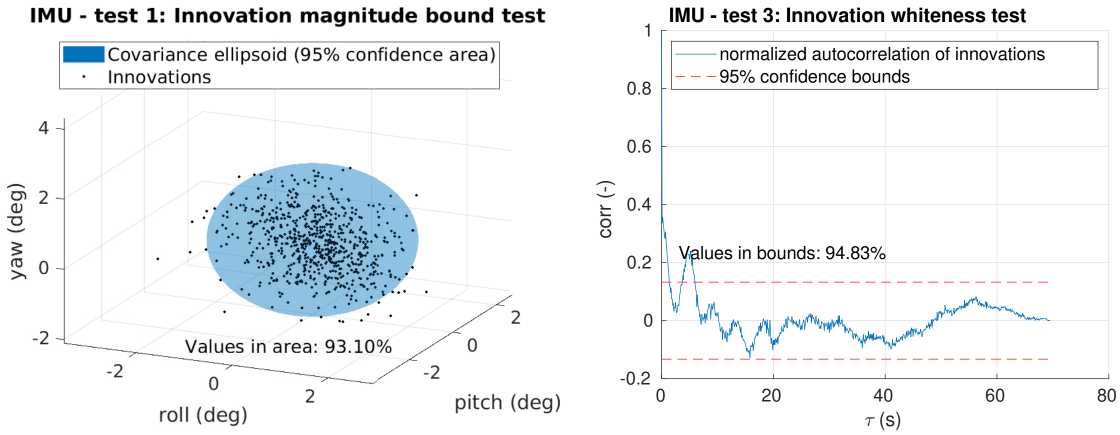


Figure A.11: Estimated states (ϕ, θ, ψ) by the Unscented Kalman filter using the IMU measurements.



(a) Innovation test 1.

(b) Innovation test 3.

Figure A.12: Innovation test 1 and test 3 of the Unscented Kalman filter for states (ϕ, θ, ψ) using the IMU measurements.

UVDAR – Unscented Kalman filter verification – position

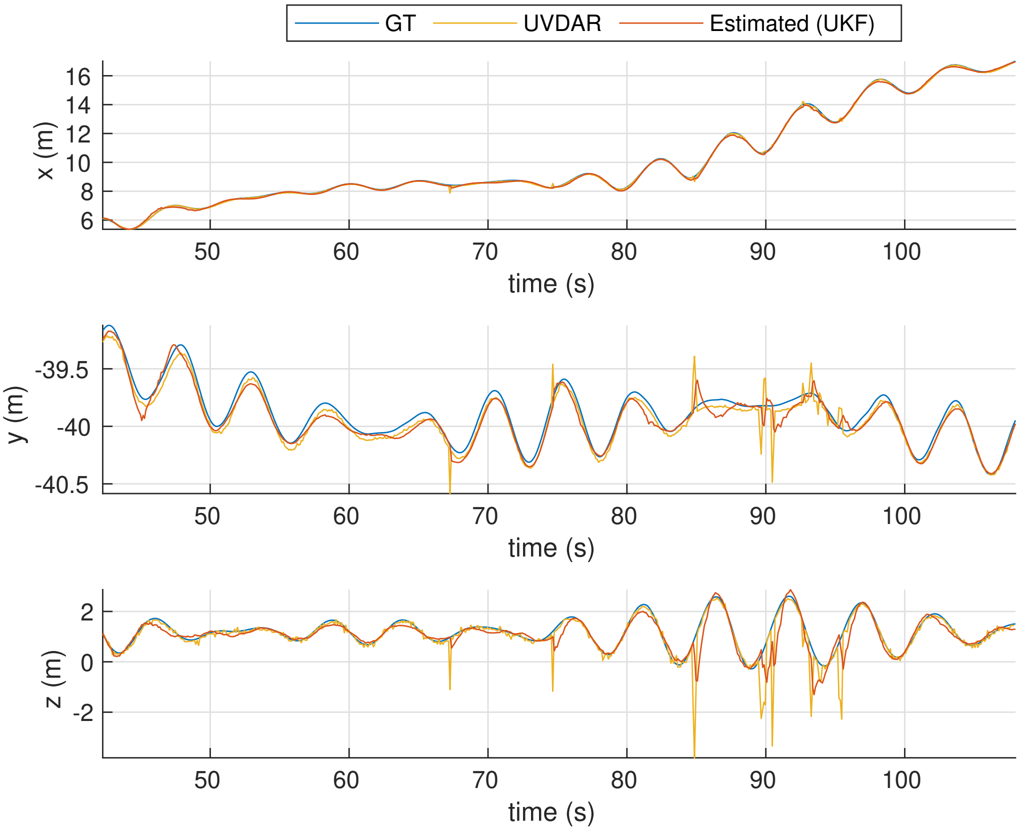
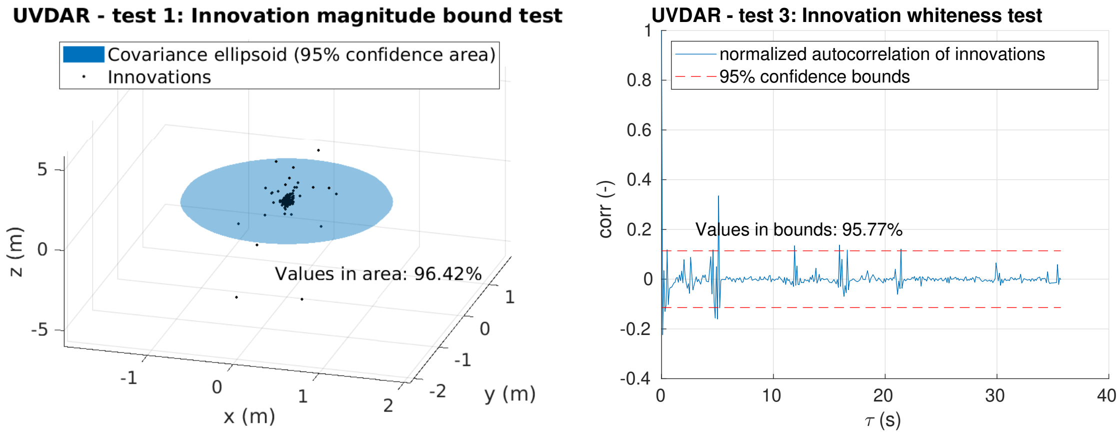


Figure A.13: Estimated states $(x,\ y,\ z)$ by the Unscented Kalman filter using the UVDAR measurements.



(a) Innovation test 1.

(b) Innovation test 3.

Figure A.14: Innovation test 1 and test 3 of the Unscented Kalman filter for states $(x,\ y,\ z)$ using the UVDAR measurements.

UVDAR – Unscented Kalman filter verification – orientation

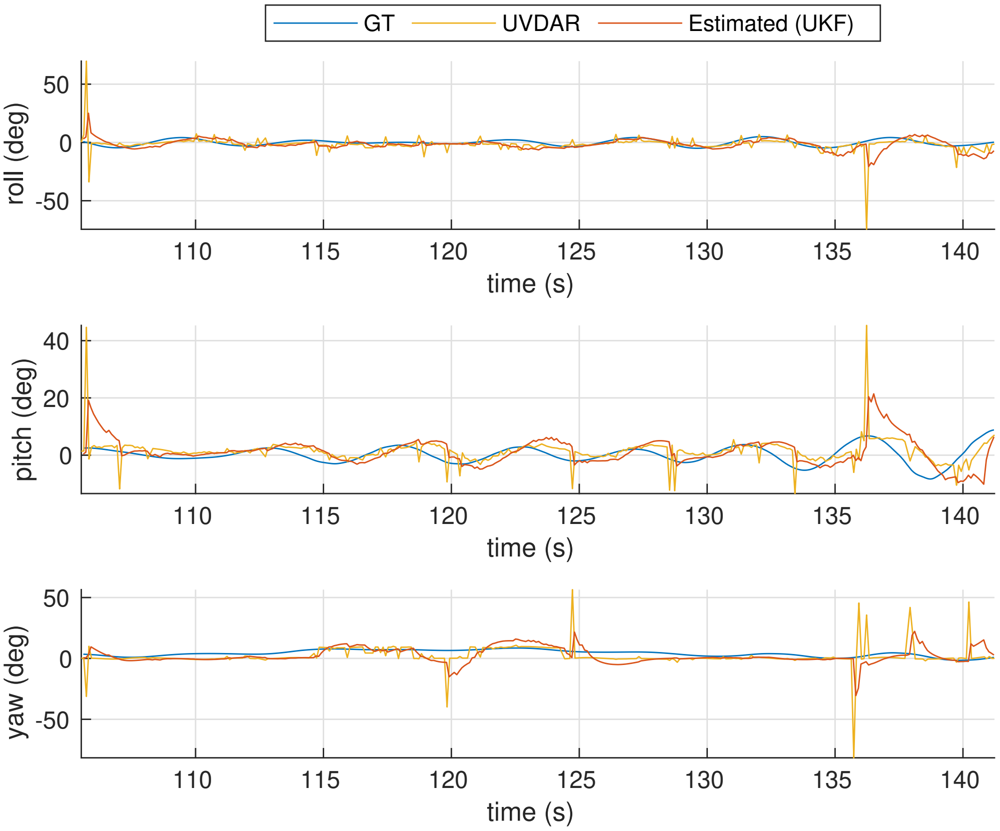
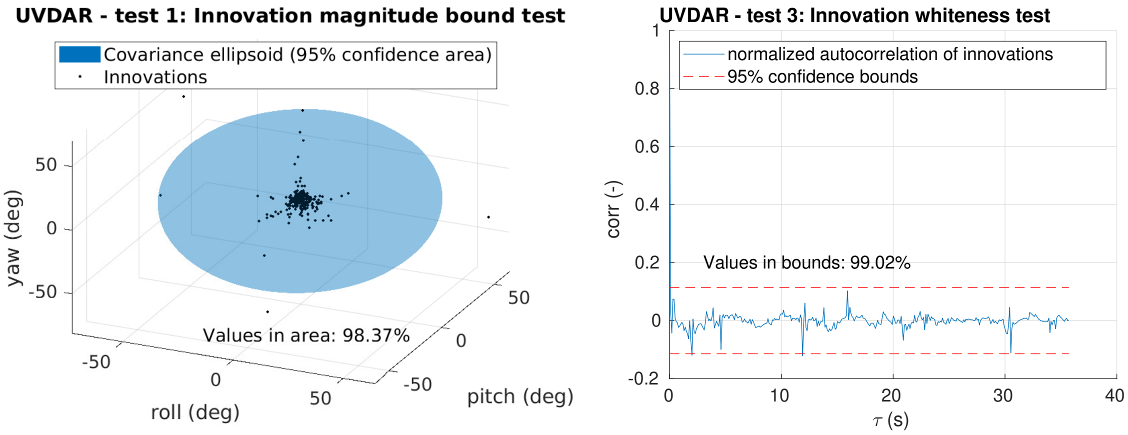


Figure A.15: Estimated states (ϕ, θ, ψ) by the Unscented Kalman filter using the UVDAR measurements.



(a) Innovation test 1.

(b) Innovation test 3.

Figure A.16: Innovation test 1 and test 3 of the Unscented Kalman filter for states (ϕ, θ, ψ) using the UVDAR measurements.

AprilTag – Unscented Kalman filter verification – position

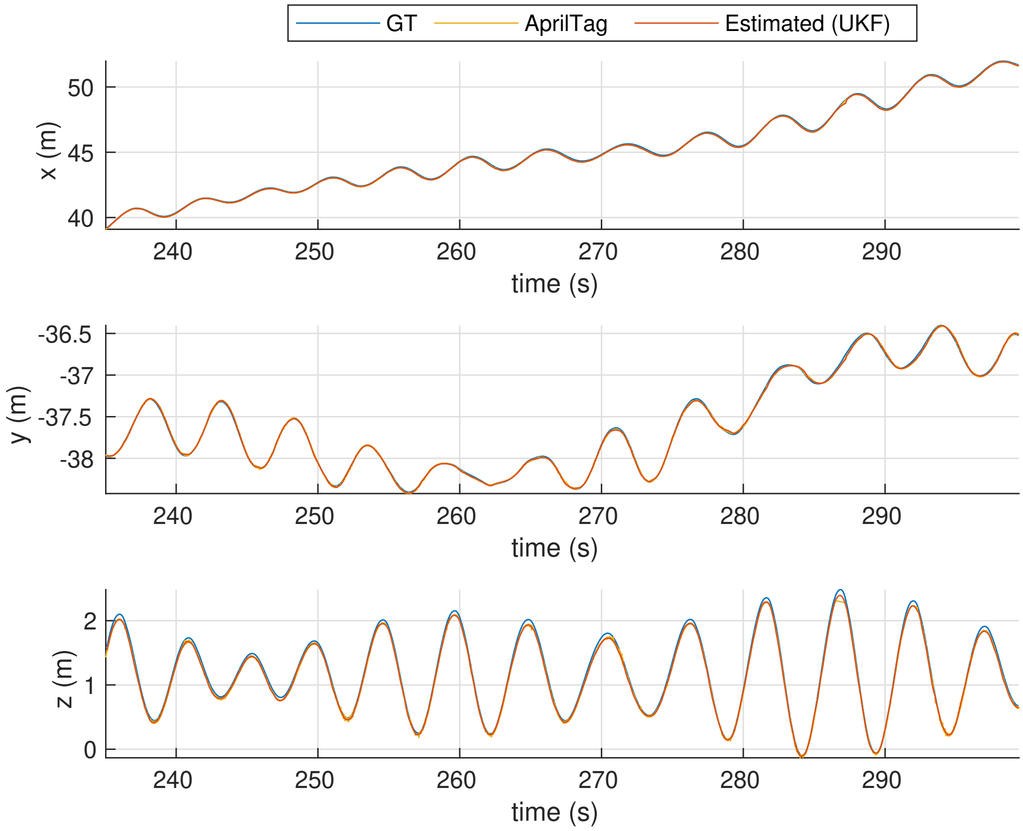
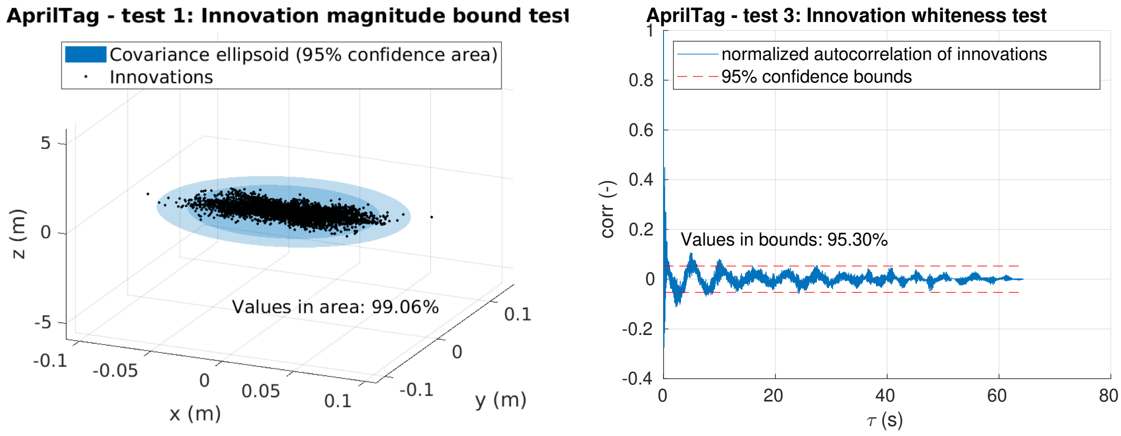


Figure A.17: Estimated states ($x,\ y,\ z$) by the Unscented Kalman filter using the AprilTag measurements.



(a) Innovation test 1.

(b) Innovation test 3.

Figure A.18: Innovation test 1 and test 3 of the Unscented Kalman filter for states ($x,\ y,\ z$) using the AprilTag measurements.

Appendix B

Real-world experiments

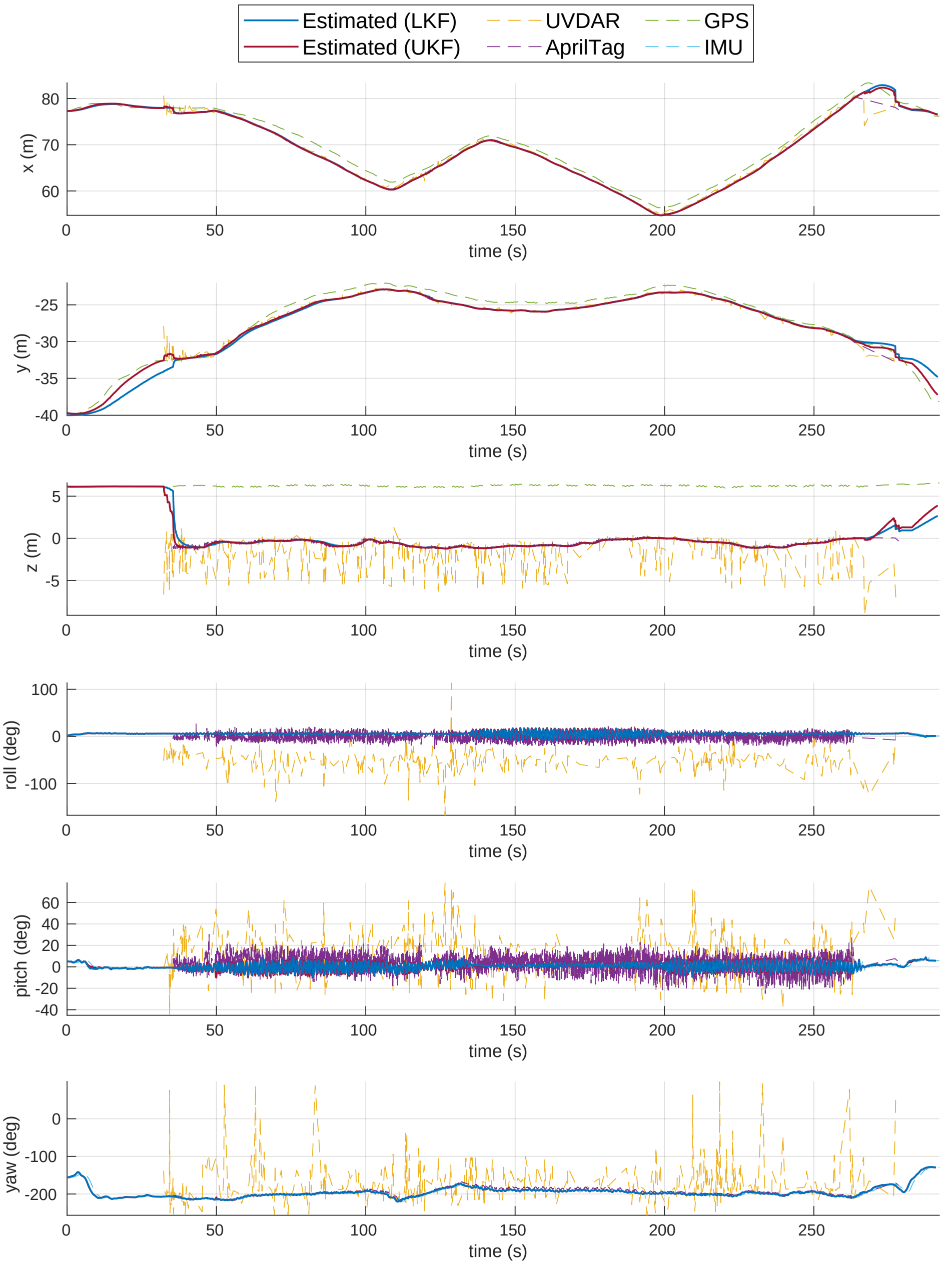


Figure B.1: Estimation of the USV states (x, y, z, ϕ, θ, ψ) using the Linear Kalman filter (LKF) and Unscented Kalman filter (UKF) while the UAV followed the USV.

Appendix C

CD Content

The names of all root directories on the attached CD are listed in Table C.1.

Directory name	Description
thesis.pdf	thesis in pdf format
sources/thesis	LaTeX source codes
sources/usves	software source codes
videos	videos from real-world experiments

Table C.1: CD Content

Appendix D

List of abbreviations

In Table D.1 are listed abbreviations used in this thesis.

Abbreviation	Meaning
CTU	Czech Technical University
DOF	Degree of Freedom
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
IMU	Inertial Measurement Unit
LTI	Linear time-invariant
MAV	Micro Aerial Vehicle
MPC	Model Predictive Control
MRS	Multi-robot Systems
ROS	Robot Operating System
RTK	Real-time Kinematics
UAV	Unmanned Aerial Vehicle
USV	Unmanned Surface Vehicle
RMSE	Root Mean Square Error
UVDAR	UltraViolet Direction And Ranging
UV	UltraViolet
VRX	Virtual RobotX
WAM-V	Wave Adaptive Modular Vessel
GT	Ground Truth
UTM	Universal Transverse Mercator

Table D.1: List of abbreviations