ESTIMATION AND CONTROL OF A MANIPULATING UNMANNED AERIAL VEHICLE

by

Hossein Bonyan Khamseh

MSc, Aerospace Engineering, Shahid Beheshti University, Iran, 2010

BSc, Aerospace Engineering, Amirkabir University of Technology, Iran, 2008

A dissertation
presented to Ryerson University

in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy
in the program of
Mechanical and Industrial Engineering

Toronto, Ontario, Canada, 2018

AUTHOR'S DECLARATION

I hereby declare that I am the sole author of this dissertation. This is a true copy of the dissertation, including any required final revisions, as accepted by my examiners.

I authorize Ryerson University to lend this dissertation to other institutions or individuals for the purpose of scholarly research.

I further authorize Ryerson University to reproduce this dissertation by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

I understand that my dissertation may be made electronically available to the public.

This is a manuscript-based dissertation on the basis of following publications:

1) H. Bonyan Khamseh and F. Janabi–Sharifi, "UKF–based LQR control of a Manipulating Unmanned Aerial Vehicle," *Unmanned Systems*, vol. 5, no. 3, pp. 131–139, 2017.
2) H. Bonyan Khamseh and F. Janabi–Sharifi, "Modeling and Control of a Manipulating Unmanned Aerial Vehicle," In *Proc. Canadian Society for Mechanical Engineering (CSME) International Congress*, June 2016, Kelowna, BC, Canada.
3) H. Bonyan Khamseh, F. Janabi–Sharifi, and Abdelkader Abdessameud "Aerial manipulation – a literature survey," *Journal of Robotics and Auton. Sys*., vol. 107, pp. 221–235, 2018. (Bonyan Khamseh was the main author of the paper. Abdessameud contributed to the sections on sensory configurations, dynamic modelling, and overall structure of the paper)
4) H. Bonyan Khamseh and F. Janabi–Sharifi, "Unscented Kalman Filter State Estimation for Manipulating Unmanned Aerial Vehicles," Submitted to *Journal of Aerospace Science and Technology.*
5) H. Bonyan Khamseh, A. Assa and F. Janabi–Sharifi, "Adaptive Extended and Unscented Kalman Filters for Quadcopter State Estimation," Submitted to *Robotica*. (Bonyan Khamseh was the main author of the paper. Assa provided guidance and consultation on adaptive unscented Kalman filtering)

ESTIMATION AND CONTROL OF A MANIPULATING UNMANNED AERIAL VEHICLE

Hossein Bonyan Khamseh

Doctor of Philosoph*y, 2018*

Department of Mechanical and Industrial Engineering, Ryerson University

Abstract

A Manipulating Unmanned Aerial Vehicle (MUAV) is an aerial platform equipped with a mechanism to physically interact with its environment. The interaction is realized by means of robotic arm(s), vehicle body or suspension cable(s) and enables a wide range of novel applications including perching, grasping, pick–and–place, load transportation, etc. However, this is a challenging task as MUAVs are inherently unstable platforms with highly nonlinear and coupled dynamics, often associated with moving parts that result in complex modelling, estimation and control problems.

This thesis deals with the problem of estimation and control of MUAVs. First, a comprehensive literature survey covering various aspects of MUAVs such as those related to modelling, estimation and control is presented. In the first approach for MUAV state estimation and control, effects of robotic manipulator on Unmanned Aerial Vehicle (UAV) dynamic equations of motion is treated by adding process noise with unknown noise statistics to conventional UAV dynamic model. With that in mind, state estimating and control of a UAV by means of conventional Kalman filters and their adaptive counterparts are formulated. Having designed Linear Quadratic Regulator (LQR) laws, it is shown that adaptive Kalman filters provide accurate satisfactory estimation and overall control of a UAV, even with simultaneous uncertain process and measurement noise statistics.

Next, in order to improve the estimation and overall control performance of the previous approach, full nonlinear and coupled dynamic modelling of a MUAV based on Euler–Lagrange formulation is presented. Then, a General Unscented Kalman Filter (GUKF) is proposed to accomplish full state estimation of a MUAV, along with LQR control laws. Finally, in order to improve the execution time of GUKF, a computationally–efficient UKF known as Scaled

Spherical UKF (SSUKF) with estimation and overall control performance comparable to GUKF is formulated. It is shown that both UKF–based algorithms result in satisfactory estimation and setpoint/trajectory tracking of quadcopter UAV and its robotic manipulator, even in scenarios with increased noise level and a period of total outage of sensory data.

Acknowledgements

I would like to thank:

Dedication

This dissertation is dedicated to my beloved wife *Hedieh*, and my parents *Mahnaz* and *Hassan*, for their utter love and sincere support.

Table of contents

List of tables

List of figures

Nomenclature

$\boldsymbol{\alpha}^{(i)}$	α vector in SSUKF algorithm
α	Sigma points spread parameter
β	UKF distribution parameter
λ	UKF scaling parameter
φ, θ, ψ	roll, pitch and yaw angles
$\boldsymbol{\Omega}$	MUAV thrust mapping matrix
Ω_i	Rotation speed of the $i-th$ rotor
γ	Scaling factor in SSUKF algorithm
$\boldsymbol{\omega}$	Angular velocity vector
ϖ	Fading memory coefficient
$\boldsymbol{\xi}$	MUAV state vector
$\boldsymbol{\Phi}_b$	MUAV orientation vector
$\boldsymbol{\Phi}_C$	Rotation matrix of camera frame with respect to inertial frame
$\boldsymbol{\Phi}_E$	End effector orientation vector
$\boldsymbol{\Pi}, \boldsymbol{\Upsilon}$	Coefficient vectors in low pass filter
$\boldsymbol{\chi}^i_{k-1|k-1}$	Process sigma points
$\boldsymbol{\chi}^i_{k|k-1}$	Measurement sigma points
$\boldsymbol{a}$	State noise sample
$\boldsymbol{A}$	State matrix
b	Lift coefficient
$\boldsymbol{B}$	Input matrix
c	MUAV drag factor
$\boldsymbol{C}$	MUAV centrifugal and Coriolis matrix
d	Drag coefficient
$\boldsymbol{D}_q$	Quadcopter rotor–induced drag vector
$E[]$	Expected value operator
$\boldsymbol{f}$	Manipulator–induced force vector
$f_{b,i}$	Lift generated by the $i-th$ motor

$\mathcal{F}_E$	End effector coordinate frame
$\mathcal{F}_I$	Inertial coordinate frame
$\mathcal{F}_B$	Body coordinate frame
g	Gravity acceleration
$\boldsymbol{g}$	Gravitational vector
$\boldsymbol{H}$	Measurement matrix
$\boldsymbol{I}$	Identity matrix
$\boldsymbol{I}_q$	Quadcopter inertia matrix
I_{xx}, I_{yy}, I_{zz}	Moments of inertia
J	Optimal control cost function
$\boldsymbol{J}_P^{(l_i)}$	Position–related Jacobian of link i
$\boldsymbol{J}_O^{(l_i)}$	Angle–related Jacobian of link i
$\bar{\boldsymbol{J}}$	End effector Jacobian matrix
$\mathcal{K}$	MUAV kinetic energy
$\boldsymbol{K}$	Kalman filter gain
$\boldsymbol{K}_{LQR}$	Optimal control gain matrix
l	distance from a rotor's center to quadcopter geometric center
$\mathcal{L}$	MUAV Lagrangian function
$\boldsymbol{M}$	MUAV inertia matrix
$\boldsymbol{M}_q$	Quadcopter mass matrix
n	Number of system states
N	Adaptation window size
$\boldsymbol{N}$	MUAV input mapping matrix
$\boldsymbol{n}$	Manipulator–induced torque vector
n_m	Robotic manipulator degrees of freedom
$\boldsymbol{P}_{k|k-1}$	A priori error covariance matrix
$\boldsymbol{P}_{k|k}$	A posteriori error covariance matrix
$\boldsymbol{P}^y$	UKF covariance of predicted measurements matrix
$\boldsymbol{P}^{xy}$	UKF cross covariance matrix

$\boldsymbol{P}$	Algebraic Riccati equation solution
$\boldsymbol{p}_E$	End effector position vector
$\boldsymbol{p}_B$	MUAV position vector
$\boldsymbol{p}_{Bl_i}$	Position vector of the frame on link i with respect to $\mathcal{F}_B$
$\boldsymbol{p}_{l_i}$	Position vector of the frame on link i
$\boldsymbol{q}$	Manipulator joint angles vector
$\boldsymbol{Q}^\dagger$	Estimated process covariance matrix
$\boldsymbol{Q}^*$	Process covariance matrix
$\boldsymbol{Q}_{LQR}$	Optimal control state weight matrix
$\boldsymbol{Q}_q$	Quadcopter torque vector
$\boldsymbol{r}$	Measurement innovation vector
$\overline{\boldsymbol{R}}$	MUAV input coordinate transformation
$\boldsymbol{R}_B$	Rotation matrix from body to inertial frame
$\boldsymbol{R}^\dagger$	Estimated measurement covariance matrix
$\boldsymbol{R}^*$	Measurement covariance matrix
$\boldsymbol{R}_{LQR}$	Optimal control input weight matrix
$S(\cdot)$	Skew–symmetric matrix operator
$\boldsymbol{T}_q$	Quadcopter lift vector
$\mathcal{U}$	MUAV potential energy
$\boldsymbol{u}_{MUAV}$	MUAV generalized force vector
$\boldsymbol{v}$	Measurement noise
$\boldsymbol{w}$	Process noise
W_i	UKF state and covariance weight vector
W_i^m, W_i^c	SSUKF state and covariance weight vectors
$\widehat{\boldsymbol{x}}_{k|k-1}$	A priori state estimation vector
$\widehat{\boldsymbol{x}}_{k|k}$	a posteriori state estimation vector
$\boldsymbol{x}^*$	MUAV desired setpoint
$\boldsymbol{y}$	Measurement vector

Chapter 1 Introduction

1.1 Background and motivation

In recent years, a significant growth in Unmanned Aerial Vehicle (UAV) industry has been realized. As an example, in the United States only, there were approximately a million UAV or "drone" gifts for Christmas 2015 [1]. To this date, UAVs have been used in applications such as remote sensing of agricultural products [2], forest fire monitoring [3], search and rescue [4], border monitoring [5], transmission line inspection [6], and plant assets inspection [7]. Fully functional UAVs for plant inspection have appeared as recently as 2010 for UK onshore oil refineries [8]. In 2012, the supermajor oil and gas company, British Petroleum, established research teams to develop the necessary technologies to use UAVs for oil pipeline inspection in Prudhoe, Alaska [9] and over the course of only a few years, the technology has matured to become the standard practice for onshore and offshore platforms [7]. The above achievements have benefited various industries tremendously; however, an important common shortcoming in the mentioned applications is that the UAV is employed to merely sense, monitor and "see" the environment, but physical interaction with the environment is strictly avoided.

Motivated by this, researchers in the last few years have begun examining applications in which a UAV is required to perform perching, grasping, and manipulation [10]–[15]. This new area of research, usually known as aerial manipulation, encourages physical interaction of the UAV with its surrounding environment and enables UAVs to perform a whole new set of missions.

Aerial manipulation falls within a relatively well–studied broad research category known as mobile manipulation. However, most of the research carried out in mobile manipulation focuses on ground robots. The main distinct challenges in the aerial manipulation problem are:

1. Unlike ground robots, UAVs do not have a stable base and therefore forces and torques generated by the presence and movement of the manipulation mechanism and/or the payload directly affect the vehicle's position, attitude and even its stability;

2. Unlike ground robots, UAVs' propulsion system acts relatively poorly in close vicinity of the ground and/or walls;

3. UAVs are often underactuated platforms with highly nonlinear coupled dynamics, introducing further complications into their control design; and

4. UAVs usually have stringent payload weight constraints and therefore cannot accommodate industrial dexterous robotic manipulators.

The above challenges encourage the development of a new research theme for the aerial manipulation problem. Aerial manipulation is a relatively new field of research. Some of the pioneering works in this area appeared in the first years of the current decade [10], [11], [16]–[18] where the manipulation usually consisted of a gripper rigidly attached to a UAV body or was based on tethered configurations. Over the course of a few years, aerial manipulation has considerably evolved and more recent works, e.g. [14], [15], [19]–[22], address challenging problems such as valve turning and pick–and–place by several Degrees–of–Freedom (DoF) robotic manipulators.

In aerial manipulation, accurate and reliable state estimation is a fundamental challenge that has not been addressed in detail yet. In the majority of previous works, precise estimates of MUAV states is obtained from costly motion capture systems such as NaturalPoint OptiTrack™ or VICON™ systems [10], [12], [17], [20], [23], [24] or by extensive redundant sensor suites including GPS, Inertial Measurement Units (IMUs), barometers, and/or onboard cameras [23], [25]–[27]. The primary drawback of the mentioned approaches is that they either confine MUAVs to lab environment with precise sensing infrastructure or require extensive use of sensors onboard a MUAV. The main motivation of this thesis is to provide reliable and time–efficient state estimation for onboard MUAV implementation, solely based on onboard sensors. Therefore, the findings of this research enable autonomous operation of MUAVs in real–life scenarios. Our results show that the proposed algorithms perform well in various scenarios, including setpoint/trajectory tracking and increasing noise level, and can regain accurate MUAV state estimation and control even after periods of sensory data outage.

1.2 Research objectives

This manuscript–based dissertation consists of 5 main chapters, based on journal papers encompassing our research results. The main objective of this dissertation is to investigate state

estimation and control of a MUAV. The following steps serve as milestones to achieve the main objective:

Chapter 1 highlights the background and motivation behind the current research. Also, the main milestones and research objectives are described.

In Chapter 2, a comprehensive literature survey is presented. Various aspects of MUAVs including their physical subsystems, sensors, mission scenarios, modelling, state estimation and control are discussed in detail. This chapter is based on the following submitted paper:

H. Bonyan Khamseh, F. Janabi–Sharifi, and Abdelkader Abdessameud "Aerial manipulation – a literature survey," *Journal of Robotics and Auton. Sys.*, vol. 107, pp. 221–235, 2018. (Bonyan Khamseh was the main author of the paper. Abdessameud contributed to the sections on sensory configurations, dynamic modelling, and overall structure of the paper)

In Chapter 3, dynamic modelling of a quadcopter UAV, along with LQR control design, is discussed. Conventional and adaptive EKF and UKF algorithms are then developed for a quadcopter with known/unknown noise statistics to investigate manipulator effects on UAV estimation and control. This chapter is based on the following submitted paper:

H. Bonyan Khamseh, A. Assa and F. Janabi–Sharifi, "Adaptive Extended and Unscented Kalman Filters for Quadcopter State Estimation," Submitted to *Robotica*. (Bonyan Khamseh was the main author of the paper. Assa provided guidance and consultation on adaptive unscented Kalman filtering)

In Chapter 4, full dynamic modelling and LQR control design of a MUAV is undertaken. Then, general and computationally–efficient variants of UKF–based state estimation algorithms are developed. Estimation accuracy and overall control performance of UKF–based approaches with various initial conditions, noise scenarios, total sensory loss and trajectory tracking scenarios are investigated. This chapter is based on the following papers:

H. Bonyan Khamseh and F. Janabi–Sharifi, "UKF–based LQR control of a Manipulating Unmanned Aerial Vehicle," *Unmanned Systems*, vol. 5, no. 3, pp. 131–139, 2017.

H. Bonyan Khamseh and F. Janabi–Sharifi, "Unscented Kalman Filter State Estimation for Manipulating Unmanned Aerial Vehicles," Submitted to *Journal of Aerospace Science and Technology*.

Finally, Chapter 5 summarizes the conclusions, important findings and contributions of this research. Also, possible directions for future research are suggested.

Chapter 2 Literature survey

This chapter is based on the following submitted paper:

H. Bonyan Khamseh, F. Janabi–Sharifi, and Abdelkader Abdessameud "Aerial manipulation – a literature survey," *Journal of Robotics and Auton. Sys*., vol. 107, pp. 221–235, 2018.

* Bonyan Khamseh was the main author of the paper. Abdessameud contributed to the sections on sensory configurations, dynamic modelling, and overall structure of the paper

2.1 Introduction

In this section, a literature review is presented to discuss various aspects of MUAVs ranging from MUAV physical subsystems, sensory configurations, missions and operational scenarios, dynamic modelling, to estimation and control problems. Also, at the beginning of subsequent chapters, in–depth literature review pertaining to specific topic of the given chapter is presented.

2.2 Literature review

2.2.1 The physical subsystems of a MUAV

In general, an aerial manipulation system contains two main physical subsystems, a UAV platform and a manipulation mechanism, with the necessary sensors and control systems for its autonomous or semi–autonomous functionality. In this section, we describe the most common subsystems of a MUAV as well as the possible sensory configurations considered for various applications.

2.2.1.1 UAV platform

Structurally, Heavier–Than–Air (HTA) UAVs present the most common configurations and can be divided into fixed–wing and rotary–wing vehicles. Fixed–wing UAVs, in general, are required to maintain a minimum forward velocity (stall velocity) and therefore cannot hover with zero forward velocity. Not surprisingly, fixed–wing UAVs have not been employed in aerial manipulation problems and therefore are not discussed in this work. A number of rotary–wing vehicles have been used in MUAVs and are discussed in the following paragraphs. Airships, a class of Lighter–Than–Air (LTA) vehicles, can also be used as the UAV in aerial manipulation applications. As an example, in [28], [29], the authors proposed a hybrid UAV (quadcopter +

airship) equipped with 3 identical robotic arms to perform grasping tasks. However, despite their promising characteristics, airship systems are not frequently employed today, mainly due to their limited payload mass budget and also lack of proper infrastructure required for their operation. The authors do not expect to see widespread use of airships in aerial manipulation unless they first become popular in more conventional aerial missions such as monitoring and surveillance.

Among rotary–wing UAVs with hovering capability, octoquads [29], hexquads [30], [31], quadcopters [32]–[38], tri–rotors [39], conventional helicopters [20], [40]–[43], and ducted–fan vehicles [44]–[46] have been used in MUAV systems. From the available literature, it can be concluded that quadcopters are by far the most popular UAV platforms used for aerial manipulation, followed by small–size helicopters. This is mainly due to the simplicity of quadcopter mechanical design and hovering capability, complemented by the low–cost, agility and existing precise control schemes for these flying vehicles. Flight characteristics of the above mentioned types of UAVs are given in Table 2.1 (partially adopted from [47]).

Table 2.1 Flight characteristics of the common types of UAVs (1=Bad, 4=Very good)

	Fixed–wing	Multicopters	Conventional helicopters	Airships
Payload	4	3	2	1
Hovering	1	4	4	3
Low speed flight	1	4	4	4
Vertical take–off and landing	1	4	4	3
Indoor usage	1	4	3	2
Feasibility for MUAV applications	N/A*	4	3	2

* *Not applicable (to this date)*

The characteristics of some UAV platforms used in aerial manipulation are presented in Table 2.2 where the very limited payload weight budget of most of these platforms is clearly seen. In fact, except for the octoquad AMUSE [26], most multi–rotor UAVs used for research weigh less than 2 kg with a payload of a few hundreds of grams [30]. This constitutes a real problem in the design of MUAVs since most manipulation mechanisms to be attached to the UAV system impose a total allowable payload exceeding the capabilities of the majority of the available

UAVs (as given in Table 2.2). For instance, it was reported in [31], [32] that a total UAV payload of approximately 10 kg is generally required in most practical applications involving MUAVs equipped with fully actuated robotic arms. For example, in [28], a 6.5 kg robotic arm known as MK1 was exploited onboard a MUAV for hazardous material removal, perching, and security applications. As a matter of fact, to this date, an order–of–magnitude shortcoming in payload weight budget of current UAVs has been a key limitation in aerial manipulation. Although some progress in increasing payload capabilities of UAVs is observed in the recent years, the authors believe that this technological issue will still remain a key concern in this research area at least in the near future.

Table 2.2 Characteristics of multirotor UAVs used in aerial manipulation

Name	Developer	UAV type	UAV weight	Payload weight	size	Remarks
Smart Xcopter [14]	XCopter	quadcopter	900 gr	450 gr	50 cm, (shaft to shaft)	–
QARM1 [33]	University of Seville custom design	quadcopter	980 gr	More than 600 gr	–	200 gr manipulator payload
AscTec™ Pelican [34]	Ascending Technologies	quadcopter	1 kg	650 gr	65 cm, (shaft to shaft)	–
T–Rex 600 ESP [16]	Align	helicopter	4 kg	More than 1 kg	1.5 m rotor	Up to1 kg manipulator payload
AMUSE [26]	University of Seville custom design	octoquads	–	Up to 8 kg	82 cm (shaft to shaft)	4 pairs of co–axial rotors, 1.5 kg manipulator payload

2.2.1.2 Manipulation/interaction mechanism

The physical mechanisms interacting with the environment in aerial manipulation can be divided into the following four main categories (summarised in Table 2.3): (i) robotic manipulators, (ii) grippers, (iii) UAV body or a rigid tool, and (iv) tethers. Each of these mechanisms is discussed as follows.

Table 2.3 Characteristics of interaction mechanisms used in aerial manipulation

Manipulation type	Application	Remarks
Robotic manipulator	Perching, load transportation, force/torque exertion, pick–and–place, structure inspection, drawer opening, valve turning, peg–in–hole insertion tasks	1–DoF, 2–DoF and several–DoF manipulators, the most common type at the present, manipulator instantaneous workspace limited by the presence of UAV
Gripper directly attached to UAV body	Perching, Load transportation	Compliant and non–compliant grippers, common in early MUAV applications
UAV body or rigid tool attached to it	Force/torque exertion, structure inspection, aerial writing	Rigid tools attached to tilt–rotor UAVs efficient for large force exertion
Tether	Load transportation	Confined to applications requiring tensile forces only

The most popular category consists of an actuated robotic manipulator attached (usually underneath) to the body of an aerial platform. Quadcopters, as most common aerial platforms in MUAV applications, are underactuated systems as they have 6 DoF with only 4 control inputs. Complementing the underactuated UAV with a robotic arm with additional DoFs can result in 6–DoF control of the aerial manipulator end–effector. From this perspective, diverse configurations ranging from 1–DoF [35] to 2–DoF [14], [22], [36] and several–DoF [26], [37] manipulators were considered. While some of these mechanisms have been adapted from other robotic contexts, some have been specifically designed for aerial manipulation applications, e.g., [31], [33], [38]–[42]. In particular, incorporating redundant and hyper–redundant manipulators in MUAVs has been investigated in [25], [26], [38], [39], [43]–[45] to achieve higher reliability, optimization of a given secondary task, and granting access to hard–to–reach locations. A lightweight 5–DoF arm was developed in [38] to accomplish manipulation tasks onboard a MUAV. In that work, the main features of the arm are its compact design, reduced variation in MUAV center of gravity and self–folding during landing manoeuvres. In [26], a dextrous 7–DoF robotic arm with maximum payload of 1.5 kg was assembled underneath an octocopter. Through experiments, it was demonstrated that several joints of the arm can be actuated simultaneously while the MUAV maintains its attitude reasonably well in an outdoor

environment. In [43], a lightweight 6–DoF robotic arm was developed and integrated under an AscTec™ Pelican quadcopter. The robotic arm redundancy in [43] enabled the realization of primary mission of end–effector trajectory tracking, along with fulfilling a number of complementary tasks, such as MUAV center of gravity and arm joints control, with minimized joints velocities. In a similar context, a hierarchical–task formulation was proposed in [25] to prioritize a number of subtasks including control of gripper position and orientation, camera field of view, MUAV center of gravity and joint–limits avoidance. In [39], [44], a hyper–redundant manipulator was considered and three different approaches, namely, pseudo–inverse Jacobian, weighted pseudo–inverse Jacobian and a heuristic approach, were exploited to determine the inverse kinematics of the manipulator. The redundant DoFs were then employed to move the links such that the destabilizing effects on the manipulator base were minimized. Moreover, the manipulator redundancy resulted in a highly reachable workspace where the end–effector can be accurately controlled to track desired position and orientation inputs. It is important to mention here that most of the manipulator joints used in MUAVs are servo–driven revolute joints [15], [33]–[36] and only few works have used prismatic joints onboard MUAVs [46][47]. As an example, a servo–driven robotic arm for MUAV applications developed at Robotics, Mechatronics and Manufacturing Automation Laboratory (RMAL) of Ryerson University is shown in Fig. 2.1.
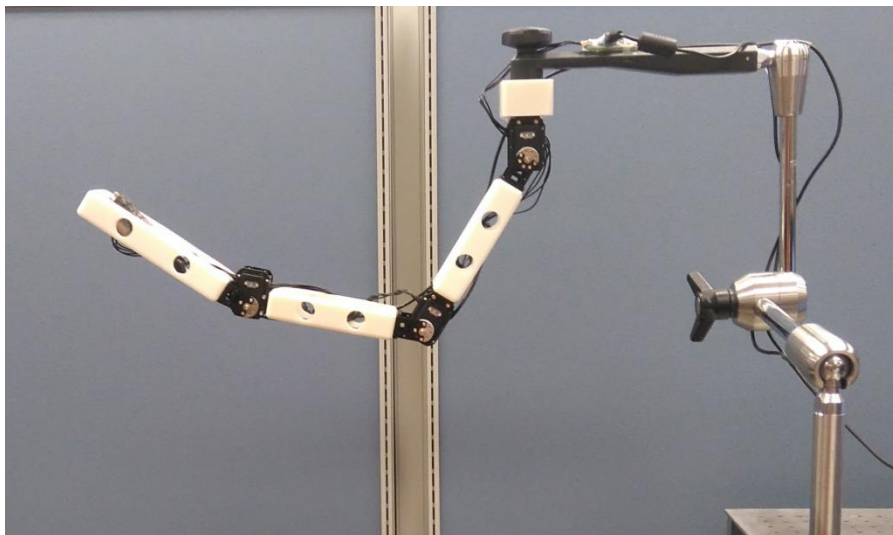
Fig. 2.1 RMAL robotic arm for MUAV applications

Grippers, directly attached under a UAV, are also popular mechanisms in aerial manipulation. Some previous works, e.g. [11], [16], [48], take advantage of simple grippers to grasp a given object and transport it. In the current literature, two main types of grippers are identified. The first type, i.e., non–compliant grippers such as those in [11], [49], resembles solid claws that grasp the object. The second type, i.e. compliant grippers presented in [16], [48], [50], can usually grasp the object, despite some level of position and orientation (pose) uncertainty. From manipulation perspective, non–compliant grippers are mainly for fine manipulation of objects and hence are not of primary importance in aerial manipulation. A chronological study of the literature indicates that the use of versatile actuated manipulator arms is becoming more common, as opposed to simple grippers used in early years of MUAVs.

A third category of interaction mechanisms in aerial manipulation is the UAV body itself [51]–[55] or a rigid tool attached to the UAV [13], [56]–[59]. In this context, some interesting applications have been presented in the literature such as [51] where the authors developed a ducted–fan VTOL vehicle capable of trajectory tracking during direct contact with vertical surfaces (walls). Similarly, the authors of [52], [53] used the propeller protection structure to establish contact and obtain a line (or surface) contact, as opposed to a point contact. Another interesting application was shown in [55] where the UAV body was employed to open a door. A more common subcategory here is to employ a rigid tool attached to the UAV to interact with the environment. As an example, the authors in [13] developed a quadcopter platform endowed with a rigid tool to exert 3D force to the environment. The latter work suggested that a counterweight (or simply an identical copy of the tool) can be mounted on the opposite side of the vehicle to balance the weight of the one actually in use. While this approach minimizes the non–symmetrical configuration of the MUAV (and thus minimizing vehicle products of inertia), it has the drawback of consuming important payload budget of the MUAV. Another interesting application was examined in [60] where a marker, rigidly attached to a quadcopter, was used to perform aerial writing tasks. The mentioned experiment verified successful physical interaction of the MUAV with the environment and may pave the way for more complicated missions involving inspection through contact. In scenarios where the tool is required to exert a large amount of force to the environment, e.g. to physically move objects obstructing vehicle's path, thrust–vectoring has been shown to be an effective approach [57]. Finally, for quadcopter UAVs,

an important constraint was analytically revealed in [59] stating that tool tip must be strictly above vehicle center of mass to maintain stability, which was taken into account in other works such as [13].

Use of tethers (or analogously cables) is also a popular scheme when interaction with the environment only requires tensile forces (as opposed to compressive forces) [10], [18]. This method is extremely popular in load transportation missions. However, it becomes ineffective in other missions that include force exertion and/or pick–and–place missions.

2.2.1.3 Sensory configuration of MUAVs

To this date, due to considerable challenges in MUAV state estimation, various sensor configurations have been examined. In most cases, UAV platforms use Inertial Measurement Units (IMUs) consisting of a 3–axis accelerometer, a 3–axis gyroscope, and a 3–axis magnetometer [23], [34]. This popular and efficient set of sensors provides the necessary set of inputs for various estimation algorithms (usually Extended Kalman Filter or Unscented Kalman Filter) which provide attitude state estimation of the UAV [61]–[64]. Also, GPS sensors [34], [65], [66], altimeters [67] and ultrasonic distance sensors [68] are used to provide further sensory input for state estimation algorithms.

In indoor environments, visual sensors constitute an efficient alternative to GPS in different applications [20], [23], [26], [43]. As an example, a motion capture system consisting of 18 V100:R2 OptiTrack™ cameras was used in [23] to provide position and velocity measurements of a quadcopter in a MUAV platform. Similarly, in [20], NaturalPoint® OptiTrack™ motion capture system was utilized to provide precise and reliable state measurements of a quadcopter–based MUAV. In the latter paper, a lightweight PointGrey™ Firefly camera was also installed onboard the MUAV to enable recognition and tracking of the object to be manipulated. Using a different approach, the authors in [48] used a monocular camera to enable a vision–based Simultaneous Localization And Mapping (SLAM) algorithm for navigation purposes. A secondary onboard camera was also used to detect infrared light sources embedded in the object to be manipulated. In [26], a camera mounted on the MUAV end–effector was used to extract relative position of an object to be manipulated and a secondary camera was attached to the UAV

body to improve relative positioning and hovering control of the MUAV. In [69], an eye–in–hand fisheye camera was used in the palm of a 3–DoF manipulator to provide omni–directional view of the hemisphere in front of the camera. In the next step, a polynomial camera calibration tool was used to compensate the distortion of the image obtained from the fisheye camera [70]. Then, contour of the target object was derived and used in an Image–Based Visual Servoing (IBVS) velocity control scheme. Another interesting application of visual sensors in MUAVs was demonstrated in [21] where the authors used circle (and ellipse) detection algorithms such that a MUAV could detect valves by means of its on–board camera. The authors believe that the currently available visual sensors in UAV applications will be extended and widely adopted in the coming years for such applications.

2.2.2 Missions and operational scenarios

In this section, the missions and operational scenarios considered for MUAVs are discussed.

2.2.2.1 Aerial manipulation missions

A wide variety of MUAV platforms are mainly used for load transportation [10], [11], [18], [71], [72]. Recently, more challenging milestones such as automatic barrel transportation, autonomous retrieval and transportation of ground robots, and cooperative load transportation by ground robots and MUAVs have been achieved [73]–[76]. To this date, three main approaches to load transportation are considered in the literature. In the first method, the load is hung below the UAV by tether. This is indeed an extension of a rich amount of literature under the general umbrella of slung load transportation [77], [78]. In the second and third approaches, the load is actually picked up by either a gripper or a manipulator, respectively. While all these load transportation methods are constrained by the limited low payload capabilities of MUAVs, the use of tethers alleviates the need of an additional mechanism attached to the aircraft and hence increases the payload of the MUAV. However, the first approach prevents pick up process automation and the stability of the UAV may be jeopardized due to load swings resulting from the system motion and environmental effects.

Tasks requiring force/torque exertion to the environment are a second important class of aerial manipulation missions. In this class, a wide range of applications such as infrastructure inspection, valve turning, door opening and drawer operation have been realized [54], [55], [79]–[81]. The authors in [80] introduced a MUAV that includes a 3–DoF robotic manipulator attached to the top of an octoquad, contrary to the majority of the literature where the arm is attached underneath the vehicle, and experimental results in bridge structure inspection have been presented. Another interesting application was investigated in [81] where the authors proposed an aerial manipulator capable of opening a drawer with unknown mechanical properties. The authors of [55] claimed that closed doors are a major obstacle for effective operation of aerial robots in indoor environment and proposed a force exertion scheme to overcome this problem. In their design, a quadcopter first approaches a door, changes its attitude and perches on the door by means of suction cups. Then, it uses a soft–bag actuator to twist the door knob and the thrust force of quadcopter is used to open the door. Finally, valve turning is another example of force/torque exertion by aerial manipulators. In [21], [22], the authors presented an aerial manipulator consisting of a quadcopter with dual 2–DoF robotic manipulators, where the two manipulators are used for grasping the valve and the UAV's yaw motion is employed to accomplish the valve turning task.

A third category of aerial manipulation missions consists of assembly and structural construction. Flight Assembled Architecture (FSA), a tower with the height of 6 m, is one of the first structures built exclusively by a team of quadcopters [82]. Also, in [17], the authors developed a team of quadcopters to build simplified truss–like structures consisting of nodes and members (rectangular prism) equipped with magnets. This work was further extended in [83] where a team of quadcopters was used to build a 3D truss structure, frequently used in scaffolds, tower cranes, and power transmission towers. Using reinforcement learning, the construction of 3D cubic truss–like structures with a quadcopter was realized in [84]. In [85], the authors employed quadcopters with a set of simple construction elements such as ropes, cables, and wires to build lightweight tensile structures. In [86], a framework was presented to enable quadcopters to perform one of the main challenges in building tensile structures, i.e. knot–tying, and an iterative learning algorithm was discussed to improve the quality of the resulting knots.

Furthermore, a number of other aerial manipulation applications that do not strictly fall into above categories have also been achieved. For instance, some researchers have proposed aerial missions for autonomous water sampling [67], forest canopy sampling [87], and sprinkling dispersants over areas affected by oil spills [88]. Also, aerial manipulation has been used for in–situ oil–spill cleanup operation, due to its cost–effectiveness an efficiency compared to alternative approaches such as mechanical cleanup [89]. Perching, as described in [12], [90], is another application that can help MUAVs save and/or restore energy and quickly place sensors in a given environment. Another interesting mission was realized in [91] in which a team of quadcopters was employed to transport a hose, where the hose was considered as a deformable linear object. More specifically, catenary curves were used in dynamic modeling of the hose which result in fast computation of the forces exerted to the quadcopters and, in turn, make real–time implementation possible. Finally, the work in [92] presented a force compliance methodology to insert a hose into a given pump. The authors of the latter work claim that the same methodology can be applied to accomplish valve turning and door opening missions as well.

2.2.2.2 Aerial manipulation scenarios

To this date, two main operational scenarios are observed in aerial manipulation. The popular scenario is a sequential one during which a MUAV first flies toward an object (to be manipulated) while locking the manipulation mechanism [11], [15], [22], [33]. In this phase, the manipulator is usually maintained in a configuration that minimizes its disturbance effects on the UAV motion. Once the MUAV is in close vicinity of the object, the manipulator (or simple gripper) is actuated to grasp the object and manipulates it accordingly. For MUAVs with simple grippers, the aerial platform flies over the object and once the object is within its reach, the gripper is actuated to grasp the object [11], [93], [94]. For MUAVs with manipulators, the MUAV tries to maintain its hovering condition while the manipulator is actuated to reach and grasp the object [15], [33]. Locking the manipulator during the first phase of the mission simplifies motion control problem of the system at the cost of extended operation time.

In the second scenario, the concurrent one, a UAV and its manipulator operate simultaneously to reduce the required time to carry out a given aerial manipulation task [20], [35], [95]. In [20], it

was experimentally shown that this method significantly improve the time required to perform a pick and place task. In addition, the manipulator and/or tether dynamics can be exploited in this approach to improve hovering and tracking performance of a UAV and to significantly extend a UAV flight envelope [46]. For instance, the tension of the cable linking a small–scale helicopter to the ground was employed in [96] to improve hovering performance under several environmental/disturbance scenarios. It is worth mentioning that the cable can also serve as unlimited power supply and/or wideband data transmission in such contexts. In conclusion, as compared to the sequential method, concurrent operation of the MUAV subsystems presents several advantages and might be adopted more widely in the future with improved dynamic modeling methods and control algorithms.

2.2.3 Modeling of a MUAV

In this section, kinematic and dynamic modeling of a MUAV and its interaction with the environment are discussed. For sake of simplicity, a MUAV consisting of a quadcopter and a manipulator with n_m DoF is considered. The MUAV model presented in this section is similar to that in [19] and can be used for other aerial platforms (such as tri–rotors or ducted–fan vehicles) with minor modification. Consider the schematic view of a MUAV in Fig. 2.2, where $\mathcal{F}_I$ is an inertial coordinate frame, a frame $\mathcal{F}_B$ is attached to the quadcopter center of mass, and $\mathcal{F}_E$ is considered as the frame attached to manipulator end–effector.
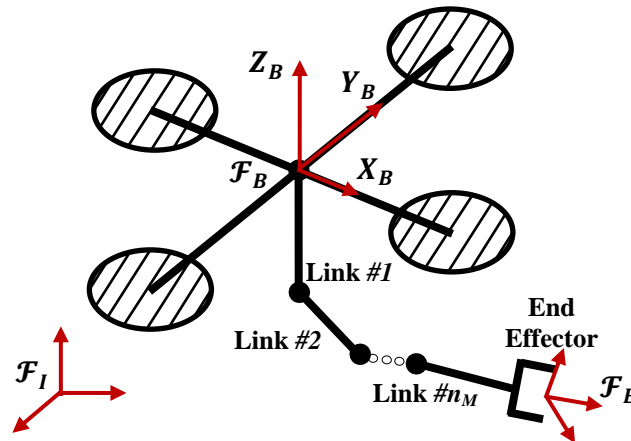
Fig. 2.2 Schematic view of a MUAV and coordinate frames

The position of the origin of $\mathcal{F}_B$ expressed in $\mathcal{F}_I$ is denoted by $\boldsymbol{p}_B \in \mathbb{R}^3$. The roll, pitch and yaw angles representing the orientation of $\mathcal{F}_B$ with resepct to $\mathcal{F}_I$ are, respectively, the components of the vector $\boldsymbol{\Phi}_B = [\varphi \quad \theta \quad \psi]^T$. Similarly, the position and orientation of $\mathcal{F}_E$ with respect to $\mathcal{F}_I$ are denoted by $\boldsymbol{p}_E \in \mathbb{R}^3$ and $\boldsymbol{\Phi}_E = [\varphi_E \quad \theta_E \quad \psi_E]^T$, respectively. With these definitions, one can write:

$$\dot{\boldsymbol{x}}_E = \bar{\boldsymbol{J}}(\boldsymbol{\Phi}_B, \boldsymbol{\Phi}_E, \boldsymbol{q})\dot{\boldsymbol{\xi}}, \tag{2.1}$$

where $\boldsymbol{x}_E = [\boldsymbol{p}_E^T \quad \boldsymbol{\Phi}_E^T]^T$, $\boldsymbol{\xi} = [\boldsymbol{x}_B^T \quad \boldsymbol{q}^T]^T$, $\boldsymbol{q} \in \mathbb{R}^{n_m}$ denotes the manipulator generalized coordinates, and $\boldsymbol{x}_B = [\boldsymbol{p}_B^T \quad \boldsymbol{\Phi}_B^T]^T$, see [19] for the Jacobian $\bar{\boldsymbol{J}}$. A well–known drawback of the above kinematics is the inevitable singularity inherent to Euler–angle representation of the attitude. However, the above formulation is acceptable for the majority of aerial manipulation applications, where aggressive maneuvers close to singularity conditions are generally avoided. Otherwise, singularity–free unit quaternion representation of the vehicle attitude can always be adopted, see [97], [98]. A mathematical extension of the unit quaternion representation, namely dual–quaternion representation, has been recently formulated to address kinematic and dynamic modeling of a MUAV [99]. An important advantage of dual–quaternion modeling of a MUAV is that it results in relatively simple transformations for consecutive rotations and translations of reference frames. However, in general, dual–quaternion representation is still not well explored and therefore its application to MUAV modeling is not expected to grow within the foreseeable future.

Regarding the dynamics of MUAVs, the equations of motion of such systems can be derived using either the Newton–Euler or Euler–Lagrange formulations [100], [101]. While both methods lead to identical equations of motion, it is well known that the former is more convenient for actual implementation whereas the latter is better suited for the study of dynamic properties and analytical investigation. In fact, the compact analytical form obtained from Euler–Lagrange formulation explicitly provides the inertia matrix, centrifugal and Coriolis forces matrix, and the gravitational forces vector and therefore is advantageous for control design purposes. On the other hand, solving the inverse dynamics problem can be done more efficiently

if the equations of motion are obtained using Newton–Euler formulation. Each of these methods is discussed separately in the following subsections.

2.2.3.1 Newton–Euler method

In Newton–Euler formulation, quadcopter and manipulator are considered as two distinct subsystems, interacting at the manipulator base frame. The dynamic equations of motion of a quadcopter UAV equipped with a manipulator are given by:

$$\begin{bmatrix} \boldsymbol{M}_q & \boldsymbol{O}_3 \\ \boldsymbol{O}_3 & \boldsymbol{I}_q \end{bmatrix} \begin{bmatrix} \ddot{\boldsymbol{p}}_B \\ \dot{\boldsymbol{\omega}}_B \end{bmatrix} + \begin{bmatrix} \boldsymbol{O}_3 \\ \boldsymbol{\omega}_B \times (\boldsymbol{I}_q \boldsymbol{\omega}_B) \end{bmatrix} = \begin{bmatrix} \boldsymbol{T}_q - g\boldsymbol{e}_3^T \\ \boldsymbol{Q}_q + \boldsymbol{D}_q \end{bmatrix} + \begin{bmatrix} \boldsymbol{f} \\ \boldsymbol{n} \end{bmatrix}, \tag{2.2}$$

where $\boldsymbol{M}_q$ and $\boldsymbol{I}_q$ are the mass and inertia matrices of the quadcopter and $\boldsymbol{\omega}_B$ is the angular velocity of the quadcopter. Also, $\boldsymbol{T}_q$ is the thrust (lift) force generated by 4 rotors, $\boldsymbol{Q}_q$ is the vector of input torques and $\boldsymbol{D}_q$ is the rotor–induced drag. Additionally, g is the gravity acceleration and $\boldsymbol{e}_3 = [0 \quad 0 \quad 1]^T$. Finally, $\boldsymbol{f}$ and $\boldsymbol{n}$ are the forces and torques exerted by the manipulator on the quadcopter. It is clear that Eq. 2.2 presents some major differences as compared to the conventional quadcopter dynamics found in the literature [102], [103]. In fact, Eq. 2.2 takes the static and dynamic effects of the manipulator into account, as well as the fact that the thrust force in the MUAV generates undesirable torques as the center of mass of the combined system moves away from that of the quadcopter.

2.2.3.2 Euler–Lagrange method

In this formulation, the quadcopter and its manipulator are treated as a single system [104]. The equations of motion of the system satisfy:

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{\xi}_i}\right) - \frac{\partial \mathcal{L}}{\partial \xi_i} = u_{MUAV,i}, \qquad i = 1, 2, \dots, 6 + n_m \tag{2.3}$$

17

where $\mathcal{L} = \mathcal{K} - \mathcal{U}$, is the Lagrangian, $\mathcal{K}$ and $\mathcal{U}$ are , respectively, the system total kinetic energy and potential energy, and $\boldsymbol{u}_{MUAV}$ is the generalized force vector. Having computed the total kinetic and potential energy, the combined system dynamics can be written as:

$$\boldsymbol{M}(\xi)\ddot{\xi} + \boldsymbol{C}(\xi, \dot{\xi})\dot{\xi} + \boldsymbol{g}(\xi) = \boldsymbol{u}_{MUAV} \tag{2.4}$$

with a generic element of $\boldsymbol{C}$ being given by Christoffel symbols of the first type [100]:

$$C_{ij} = \sum_{k=1}^{6+n_m} \frac{1}{2}\left(\frac{\partial M_{ij}}{\partial \xi_k} + \frac{\partial M_{ik}}{\partial \xi_j} - \frac{\partial M_{jk}}{\partial \xi_i} \right)\dot{\xi}_k, \tag{2.5}$$

and $\boldsymbol{g}(\xi) = \left(\frac{\partial \mathcal{U}}{\partial \xi} \right)^T$. The above derivation presents a unified dynamic model of MUAV and can be conveniently used for control design purposes.

Unlike the dynamic model of a UAV with no moving arm, a distinct characteristic of MUAV model is that its elements are configuration–dependent and hence time–varying. Also, the above model is subject to various uncertainties due to, for instance, imperfect knowledge of mass and inertia matrices and aerodynamics modeling. Furthermore, the majority of manipulators used in MUAVs are lightweight and custom–made, resulting in further uncertainties due to modeling errors/simplifications.

2.2.3.3 MUAV interaction with the environment

It is also important to note that modeling the MUAV interaction with the environment is an ongoing research area. In [50], bogie suspension approximation was used to study a helicopter and its compliant gripper mechanism grasping a fixed object. The bogie suspension approximation was further simplified later to a single–joint prismatic link subject to torsional spring forces at its both ends. This way, a relatively simple model of the contact between the MUAV and the object was derived and used for MUAV stability analysis during the contact

phase. In [79], the environment was considered as a compliant surface in the inertial frame and the dynamics of the MUAV end–effector interaction with the environment was described by Hunt–Crossley interaction model. In this model, the environment applies a force to the end–effector, directed along the opposite direction of surface penetration where the magnitude of the applied force depends on penetration depth and penetration velocity. The interaction of a MUAV tool–tip with the environment was studied in [13] using the so–called hard finger contact model, which is usually used when dealing with significant contact friction over a relatively small contact patch. In this model, the environment exerts a 3D force on the tool–tip but interaction torque is discarded. A last remark is that while the contact model used in MUAV applications needs to be accurate, it must be computationally–efficient as well. In this manner, the contact model can be accommodated within the MUAV onboard computational resources, to be used in practical applications.

2.2.4 Estimation and Control of MUAV

2.2.4.1 State and parameter estimation

Accurate and efficient estimation of the state and parameters of MUAVs is an important element of autonomous aerial manipulation missions. However, dynamic modeling of a MUAV results in tedious, coupled and highly nonlinear equations of motion and therefore conventional estimation algorithms are generally difficult to apply for MUAV applications. Among the limited works in this context, a least square problem was formulated in [11], using control input and acceleration measurements, to estimate the mass, center of mass and inertia parameters of a load grasped by a MUAV. A similar approach was used in [15] to estimate the MUAV center of mass and associated gravity torque. The performance of the proposed algorithm in [15] was shown to be adequate for static scenarios, however, dynamic changes due to arm movement degraded the performance of the overall system. Carrying an unknown mass was recently addressed in [105] as well, where the equations of motion were parameterized with respect to the unknown load mass. An important advantage of the method in [105] is that the load does not need to be hanging from underneath the MUAV and therefore it can be applied to missions where the MUAV is required to extend its robotic arm and grasp an object from its vicinity. From state estimation perspective, while EKF is the most popular and widely–used state estimation algorithm in UAV

applications [106]–[108], its linearization step along with its poor performance for highly nonlinear systems limit its application for MUAVs. Alternatively, Unscented Kalman Filter (UKF) has been formulated in [109], [110] to remedy the limitations of EKF in such applications. However, the improved performance of UKF–based state estimation methods comes, in general, at the cost of high computational complexity [110], [111]. In [110], scaled spherical UKF was formulated for MUAVs applications, and was shown to be computationally more efficient than conventional UKF, but the latter outperforms in terms of estimation accuracy.

Visual state estimation has also been examined in MUAV applications in recent years. For instance, in [26], a downward pointing camera was used to obtain relative position and orientation of a MUAV with respect to a known target. These estimates were then fused with centimeter–level position estimates obtained from a differential GPS suite, enabling precise control of the MUAV in various maneuvers. Accurate velocity estimation can be obtained using optical flow sensors; see [112], [113] for similar problems involving UAVs. The above works mostly require accurate a priori knowledge of camera intrinsic parameters. In order to relax this requirement, the authors in [114] addressed the problem of imperfect knowledge of onboard camera focal length due to poor camera initialization or unpredicted zoom changes. The camera focal length was iteratively estimated in [114] by solving a least–square problem and the proposed approach was shown to be robust to noise. Despite the interesting results mentioned above, most current MUAVs applications require precise state measurements/estimates provided by expensive sensors confined in indoor environments only, such as motion capture systems. As a matter of fact, the state and parameter estimation problem of MUAVs is yet largely unsolved, and a lot has to be done in this area to enable efficient autonomous functionality in indoor and outdoor MUAV scenarios.

2.2.4.2 MUAV control

Control of MUAVs can be studied from different perspectives discussed in this section.

Position Control

The primary control objective for MUAVs is to drive the attitude and/or position of the UAV and the manipulator end–effector to a desired position/trajectory such that a given mission is accomplished. Position control design of such nonlinear systems can be categorised as decoupled or coupled, depending on the dynamic models discussed earlier. In the first category, separate controllers for the UAV and the manipulation mechanism are developed [36], [101], [115], [116], whereas in the second, the overall system dynamics is considered [20], [35], see Table 2.4.

Table 2.4 Decoupled and coupled control strategies in MUAV applications

MUAV Control	Control algorithms	Advantages	Disadvantages
Decoupled	*UAV control algorithm*: various simple and Gain–scheduling Proportional–Derivative (PD) and PID control [15], [23], [36], feedback linearization [20], backstepping [26] and adaptive control [15], [117] *Manipulator control algorithm*: Independent joint control (PID, PID + gravity compensation) [36], [101]	Complex model of MUAV is not required. A wealth of literature is available for UAV control (similarly for manipulator control). It is computationally less expensive.	It usually requires that UAV and manipulator are not actuated simultaneously. Relatively less accurate control
Coupled	Feedback linearization [46], impedance control [104], nonlinear backstepping [95], LQR [35], adaptive control [69], [118] and model–predictive control [20], [60]	UAV and manipulator are actuated together and therefore less time is required for a given scenario. More accurate control can be achieved.	Complex model of MUAV is required. It is computationally expensive.

Decoupled control design of the UAV and manipulator

In this approach, the UAV and the manipulator are treated as separate subsystems [33], [119]. A general schematic diagram of this control approach is depicted in Fig. 2.3 where Ref_{UAV} and Ref_M are the reference (desired) states of the UAV and manipulator, respectively [33].
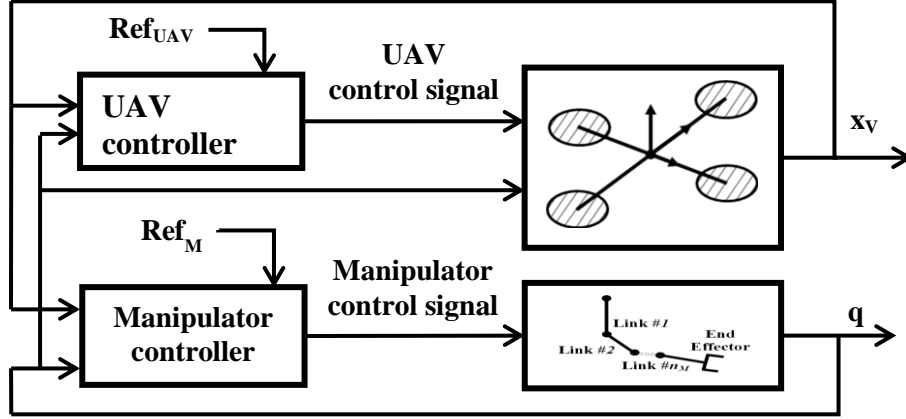
Fig. 2.3 Schematic diagram of decoupled UAV and manipulator control approach

In the simplest implementation, the effects of manipulator on the UAV are treated as disturbances and therefore are not explicitly taken into account in controller design. In [36], for instance, simple PD/PID control laws were separately designed for a UAV and each joint of the attached manipulator. As expected, it was concluded in [36] that such control design might lead to satisfactory behavior in simple scenarios but its performance considerably degrades in real and more realistic applications with variable inertia and center of mass of the system. The latter shortcoming was addressed in [11] by taking into account the MUAV center of mass offset from its geometric center in the control design to improve the performance of the system as compared to simple PID controllers. In [33], a Variable Parameter Integral Backstepping (VPIB) approach was considered to design a position controller with better performance, as compared to the PID controller, by taking (partially) into account the mutual effects of a quadcopter and a manipulator. Specifically, the moment of inertia and center of mass of the quadcopter were explicitly computed as functions of manipulator joint angles, and the quadcopter motion was compensated in the manipulator control. A further improvement was introduced in [26] where the authors included full MUAV dynamic model with the VPIB controller initially proposed in [33]. In this manner, the authors were able to incorporate dynamic effects of manipulator movements in the controller design and proposed a system capable of outdoor operations, outperforming conventional PID controllers.

Adaptive control is also another relevant control approach for MUAV position control that allows to systematically deal with time–varying parameters of the system, such as center of mass

22

and moment of inertia. In [101], a Lyapunov–based method was used to design a decoupled adaptive control taking into account variable moments of inertia and the center of mass of a MUAV (with a 2–DoF manipulator). The main theme in [101] is that disturbances due to manipulator and its possible payload can be effectively handled by an appropriate adaptation law in the outer position control loop of a quadcopter. Specifically, it was shown that decoupled PID control of the manipulator joints, PD control of the inner attitude control loop, and adaptive control of the outer position control loop can successfully achieve precise hovering flight and trajectory tracking. In [15], [117], PID control laws were augmented with a model reference adaptive control and two gain–scheduling schemes to control a multi–arm MUAV. More precisely, a switching automaton was used to trigger the adaptation phase depending on changes of the center of mass and moments of inertia of the aerial manipulator. While a simple non–adaptive PID controller can lead to instability of MUAVs in manipulation tasks, the use of the mentioned adaptive techniques guarantees stable flight during free–flight, manipulator motion, and also picking up small objects. From the above results in decoupled position control of MUAVs, further improvements in this direction can be attained using the various control techniques from UAV and robot manipulator control literature.

Coupled control of the UAV and manipulator

In the coupled approach, the UAV and manipulator are simultaneously controlled as a unified system. Fig. 2.4 illustrates a basic schematic of the control system in this case with Ref_{MUAV} being the reference (desired) states of the combined system.
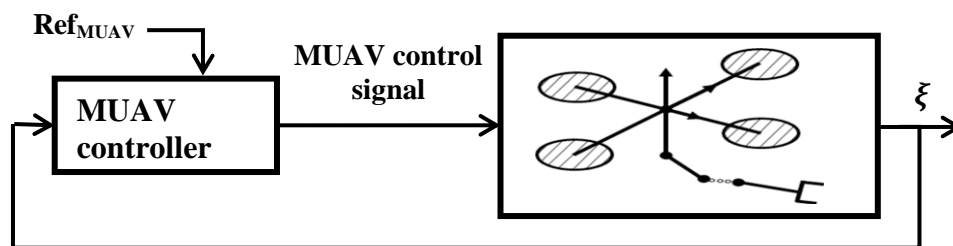
Fig. 2.4 Schematic diagram of coupled MUAV control approach

It should be noted that the equations of motion of the combined dynamic model of a MUAV take the nonlinear and coupled dynamics of the UAV and its manipulator into account. Therefore, unified model–based controller design for MUAVs is theoretically difficult and its actual implementation depends on the onboard computational resources of the system. Despite this fact, a number of control design techniques for MUAVs, such as feedback linearization [46], Cartesian impedance control [104], nonlinear backstepping [95], LQR [35], adaptive control [69] and Model–Predictive Control (MPC) [20], [60] can be found in the current literature. For example, the authors in [46] considered the planar motion of a quadcopter with a 2–DoF manipulator and designed tracking control laws based on feedback linearization. The approach in [46] takes advantage of the dynamics of the manipulator to enhance the maneuverability and tracking performance of the quadcopter. The robustness of the closed–loop system with respect to measurement errors, uncertainties and disturbances was also addressed in that work using additional integral terms in the proposed control schemes.

Backstepping is another technique implemented for coupled control in some aerial manipulation applications. In [95], the authors proposed control laws for simultaneous trajectory control of a hexquad (also known as hexrotor) and a 2–DoF manipulator. With a particular choice of the manipulator orientation with respect to the aerial vehicle, the proposed MUAV was able to perform maneuvers with the manipulator reaching beyond the perimeter of the UAV.

In a different approach, using the coupled dynamics of an unmanned helicopter and its attached 1–DoF manipulator, the authors of [35] applied a standard LQR design method to the linearized equations of motion of the system and feedback control laws were designed based on the solution of an algebraic Riccati equation. The result in [35] shows that standard design methods, such as LQR, can be used to stabilize the MUAV position in close vicinity of the trim point. However, not surprisingly, the performance degrades drastically or the system becomes unstable as the initial conditions deviate from the trim point. A similar LQR controller was proposed in [120], for a MUAV consisting of a quadcopter and a 2–DoF manipulator, where additional integral action was considered to eliminate the steady–state errors in the manipulator's position.

Adaptive control has also been applied to control a coupled system of UAV and its manipulator in [69], [118]. The authors in [118] introduced adaptive control laws, based on the inner–outer loop control of quadcopters, at the position control level to cope with modeling uncertainties.

The proposed approach in the latter paper takes perturbations due practical implementation into account such that simultaneous 3D trajectory tracking of the UAV and its manipulator was achieved. In [69], the authors proposed a sliding–mode adaptive controller for MUAV that is robust to environmental disturbances and can be applied to both position and velocity control, waypoint tracking, and visual servoing.

In [20], MPC control laws were designed such that pick–and–place missions were accomplished by a quadcopter equipped with a 2–DOF manipulator capable of reaching more than twice the radius of UAV. As a coupled control scheme, the proposed control algorithm in [20] was shown to improve the required time of a given task against a decoupled control approach. In fact, in one experiment, the required time to retrieve an object was improved from $15\,s$ in a decoupled control approach to $5\,s$ in a MPC–based coupled control approach. A drawback of MPC, however, is that it requires substantial computational resources, not always available onboard small–scale UAVs. Due to this limitation, the authors in [20] considered low–level controllers (based on feedback linearization) to follow the high–level trajectory generated by MPC control. Another application of MPC was discussed in [60] where a number of missions regarding MUAV interaction with the environment, including aerial writing, were discussed. In that work, stable interaction with the environment and trajectory control of a MUAV, while in contact with a surface, were achieved. Furthermore, by considering polyhedra enclosing the known obstacles and incorporating polyhedric constraints in the MPC cost function, obstacle avoidance was also realized. As a result, the authors were able to experimentally verify their approach in scenarios where the MUAV performed aerial writing and infrastructure inspection, while also avoiding obstacles.

In conclusion, from the abovementioned results on coupled and decoupled position control design approaches for MUAVs, it is expected that the coupled control design will receive a greater attention in the near future with a substantial development in control system design for such vehicles. This is also supported by the recent advances in UAV industry that alleviate the limited computational resources of small aerial vehicles in general.

Force/Impedance Control

In force control, it is generally desired to regulate the contact force against a given surface [56]. Decomposition of the system dynamics along and perpendicular to the surface is a popular approach to address hybrid force/position control. However, original formulation of this approach, e.g. [121], cannot be directly applied to aerial manipulation as it assumes that contact with a surface is maintained at all times. To overcome this issue, the authors of [56] used passive decomposition to decompose tool–tip dynamics to two subsystems, namely the so–called locked subsystem and shape subsystem. In this approach, the locked subsystem accounts for tangential motion of tool–tip with respect to the contact surface whereas the shape subsystem represents the motion perpendicular to the contact surface. In the next step, free–flight (before or after contact) motion controller for shape subsystem is defined such that it does not depend on contact–related parameters. Once contact is established, the shape subsystem control switches to force regulation control, making it possible to maintain the contact and a desired contact force. Control of the motion tangential to the surface is in turn possible by controlling the locked subsystem, even if the assumption of tool–tip contact with the surface is not met [56].

Impedance control is another relevant approach that is widely adopted in fixed–base robotic manipulation applications [122], [123]. In the context of aerial manipulation, the authors in [47] used impedance control to simultaneously achieve hovering flight and also maintain manipulator contact with a vertical surface. In [104], the authors employed Cartesian impedance control to perform autonomous hovering of a quadcopter equipped with a 3–DoF manipulator, subject to contact forces and external disturbances. The latter approach was extended in [124] using a redundant 3–DoF manipulator onboard a quadcopter UAV allowing to achieve additional subtasks along with the MUAV control. In particular, in [124], the authors studied scenarios where maintaining quadcopter initial position and optimizing a manipulability measure of the robotic arm were formulated as subtasks. It was also shown, by simulation, that hovering flight of a MUAV along with regulating manipulator contact force and performing secondary subtasks is achievable, even under environmental disturbances.

Vision–based Control

Use of motion capture systems and onboard vision sensors are also promising in MUAV control [12], [20], [26], [34], [45], [48], [69], [125]. In the current literature, stationary motion capture systems such as VICON™ and OptiTrack™ are usually employed to provide highly precise position and orientation information [20], [126]. However, as mentioned earlier, these sensors are primarily used in indoor structured environment and therefore significantly restrict MUAV application. Onboard visual sensors have been used to either provide extra input for MUAV state estimation [26] or to detect and/or track the target object [20], [48]. For example, in [48] the authors embedded an IR LED within the object to be manipulated. Using an onboard camera, the MUAV was able to detect the position of the object and successfully grasp it from the above. In [20], an onboard camera was used to detect several LED markers, corresponding to known point features, and determine the relative position between the MUAV and a given object, to be grasped by the manipulator. A more challenging task was accomplished in [127], where the authors used stereo cameras to generate a point cloud from which candidate objects for manipulation were extracted. The performance of object classification and candidate object selection algorithm were experimentally verified in several laboratory and outdoor environments but actual aerial grasping was not reported. That work was further improved in [128] where the UAV was held in a secure place by a support structure, mimicking its hovering scenario and grasping objects based on its visual feedback, however fully autonomous operation was not reported.

Visual servoing of MUAVs has been recently addressed in few works. In [12], based on visual features of a cylinder, an IBVS framework with a single monocular camera was proposed such that a MUAV can fly towards and grasp a cylindrical object of known radius. In contrast to most visual servoing approaches, the IBVS presented [12] is applicable to second–order and underactuated systems and therefore is better suited for MUAV applications. However, a shortcoming of [12] is that the cylinder axis was assumed to be perpendicular to gravity. In addition, an external motion capture system was used to control the MUAV motion along the cylinder axis. In [90], the authors mapped MUAV dynamics to image features and proposed an IBVS framework to maneuver the vehicle relative to cylinders with unknown orientations, using only a monocular camera. Finally, in [69], IBVS of a hexquad with a 3–DoF manipulator equipped with a fisheye camera was discussed. Specifically, an eye–in–hand fisheye camera was

used to extract the contour of a target object and then, using image moments, desired camera velocities were computed and fed to the IBVS control law. Another interesting approach was reported in [125], [129], where an IBVS scheme was formulated to bring an assembly part carried by a MUAV into a target position. The main difference of such configuration, also known as onboard eye–to–hand, with the standard IBVS formulation is that image features in both the assembly part and the target position are used in to define appropriate errors in the image space. The authors showed that IBVS of a MUAV can be ensured with effective joint–limits avoidance. The results in [129] were improved in [130] by employing image moments such that the assembly part approaches the known target with a predetermined constraint, e.g. perpendicular to the target plane. Adopting the image moments as visual features feedback, velocity–level control laws were proposed to nullify the image moments errors while the area error was set to be the last element converging to zero, i.e. the assembly part (almost) perpendicularly approaching the target.

Teleoperation

Teleoperation of MUAVs is a promising area of research that can further extend possible application of such platforms. However, practical teleoperation of MUAVs and its associated challenges such as imperfect communication, time–varying delay and information losses have not been studied to this date. Moreover, interaction with the environment and possible motion of the manipulator may, in general, generate destabilizing effects that are very difficult for a human operator to cope with. In [94], the authors affirmed that manual control of MUAVs is extremely difficult, even for experienced UAV pilots, and therefore concluded that even simple manipulation tasks are unfeasible without computer–aided control. A similar finding was reported in [22] where the authors introduced a Human–Machine–Interface (HMI) to address the mentioned problem. In that work, low–level control was carried out by an onboard autopilot, while a human operator provided the input such that MUAV hovers above a valve and grabs it. In a different application, in [13], a human operator was tasked to pilot a MUAV to establish contact between the MUAV tool–tip and a given surface. Having established the contact, the operator then used a joystick to command a trapezoidal–profile pushing force via the tool–tip to the surface. A successful experiment involving partial manual operation of MUAVs was also

reported in [94] where the MUAV was stabilized to hover over a rectangular block. Then, a human operator manually commanded 3 identical arms of the MUAV to extend and, using visual contact with the MUAV, grasp the block. The symmetrical configuration and motion of the arms with respect to UAV body played a key role in the successful human operation of the mentioned MUAV. Also, intuitively, it is easier for human operators to control the arm movement if the MUAV hovers above a given location and the operator is only tasked with controlling 1D arm extension.

First–Person–View (FPV) flight is also another plausible solution that may be adapted in MUAV applications. FPV flight, as opposed to Line–of–Sight (LoS) flight, is gaining increasing attention among UAV applications [131], [132], and especially among the hobbyists. However, to the best of authors' knowledge, FPV flight has not been used in MUAVs to this date. Motivated by the fact that operation of a MUAV is too demanding for a single operator, it is possible to design systems where position control of a UAV is carried out autonomously and a human pilot controls the arm and manipulation process. This way, the pilot can manually compensate for the UAV position error and accomplish the manipulation task. It is also important to note that while MUAV operation by a single operator is very challenging, the effect of manipulator movement on quadcopter control reduces as the mass ratio of the two subsystems decreases. Therefore, for larger quadcopters with smaller manipulators, human–based operation of MUAVs is expected to be achievable. However, the authors are unaware of any published work that determines such allowable ratio of manipulator to UAV mass.

Human–MUAV interaction

In general, a rich amount of literature exists on human interaction with fixed and mobile robots and, in particular, with humanoids [133]. However, there has been very limited research on physical human interaction with aerial robots. An example of such works is [134] where the authors used an admittance–based force control approach to enable a human user to physically interact with a quadcopter, through a foam brick underbelly of the UAV, and guide the quadcopter to a desired position. Such applications are necessary if aerial robots are to be used in close proximity of humans; e.g. in construction sites [135].

2.2.5 Summary

The area of aerial manipulation was studied in this chapter. The findings are summarized as follows:

- Various types of UAV platforms have been used in aerial manipulation applications. Strong hovering capability is the main characteristic that such vehicles should possess and therefore fixed–wing UAVs are not appropriate platforms for aerial manipulation applications. Among vehicles with hovering capability, quadcopters are the most popular ones, followed by helicopters. Other rotary–wing and LTA vehicles such as airships have also been used in some cases.

- At present, there is approximately an order–of–magnitude shortcoming in UAV payload budget in research community to incorporate industrial dextrous robotic manipulators onboard MUAV platforms. This shortcoming, along with limited computational resources of such vehicles, tends to be the main limiting factors in the foreseeable future.

- Based on the current literature, four main categories for physical interaction with the environment were identified. While tether–based configuration is the dominant mode for load transportation, its application to other MUAV missions is very limited. The configuration of a UAV equipped with actuated robotic manipulators is expected to grow more popular in the coming years.

- A broad range of MUAV missions have been realized to this date. Load transportation, infrastructure inspection, and pick–and–place are among the missions that have significant potential for further growth. A number of more radical applications such as door opening, hose transportation, peg–in–hole missions, and structure assembly have also been accomplished.

- To this date, mission scenarios of MUAVs have been defined in a conservative manner. More specifically, mission scenarios usually require either the UAV or its manipulator to be actuated at a given time. The authors believe that is going to change, yet requires substantial developments in modeling and control design techniques for MUAVs. Also, continuous improvement in computational resources of UAVs is an enabling development in the context of aerial manipulation.

- Two main methods for modeling of the dynamics of MUAVs, i.e. Newton–Euler and Euler–Lagrange approaches were discussed in this chapter. The former method is more appropriate for actual implementation whereas the latter is more suitable in analytical study of dynamic properties.

- Manual control of MUAVs is an open problem with very little documented work in the available literature. While complete control of a MUAV by human operator(s) is impractical, few successful efforts in partial manual–autonomous control of MUAVs have been reported in the literature.

- Tele–operation of MUAVs is a promising area of research that can considerably extend the horizon of MUAV applications. However, several theoretical and technical challenges such as imperfect communication, time–varying delay and information loss need to be addressed to enable practical tele–operation of such platforms.

- External motion capture systems and onboard visual sensors have been utilized in some works to enable precise estimation of MUAV states, detecting and tracking target objects and relative pose estimation. However, actual visual servoing of MUAVs is still in its infancy and is expected to receive considerable attention in the coming years.

- Different aspects of MUAV control such as decoupled vs. coupled control were studied. Although decoupled approach is theoretically simpler, it generally results in relatively more time–consuming and less accurate control of a MUAV. On the contrary, the coupled control approach is theoretically more complex but results in more efficient and accurate control of the MUAV. Various control algorithms in both of the approaches, along with their relative advantages and shortcomings, were discussed throughout the chapter.

Chapter 3 Adaptive Extended and Unscented Kalman Filters for Quadcopter State Estimation

This chapter is based on the following submitted paper:

H. Bonyan Khamseh, A. Assa, and F. Janabi–Sharifi, "Adaptive Extended and Unscented Kalman Filters for Quadcopter State Estimation", Submitted to *Robotica*.

* Bonyan Khamseh was the main author of the paper. Assa provided guidance and consultation on adaptive unscented Kalman filtering.

3.1 Introduction

Unmanned Aerial Vehicles (UAVs) are becoming increasingly popular for various applications such as target tracking [136], remote sensing of agricultural products [2], forest fire monitoring [3], search and rescue [4] and transmission line inspection [6]. Among different types of UAVs, quadcopters have become very popular due to their hovering capability, maneuverability and low cost [137]–[140]. Successful realization of various autonomous quadcopter missions heavily depends on robust and accurate state estimation algorithms. Regarding the state and parameter estimation problem, Extended Kalman Filter (EKF) is a very popular algorithm due to its advantages in noise filtering, prediction of future steps and relatively simple implementation. To this date, EKF has been used in various applications such as pose estimation [141], [142], visual servoing [143]–[145] and autonomous navigation [146]. Also, EKF is a widely used algorithm in the emerging field of UAVs [147]–[149]. As an example, in [147], the authors employed EKF to perform attitude estimation of a quadcopter and used a Proportional–Derivative (PD) controller for its stabilization. What is more, in [148], an EKF–based method was introduced to estimate wind speed and direction of a small fixed–wing UAV and to calibrate its airspeed. However, in [149]–[151], it was shown that EKF performance considerably degrades if the dynamics of the system and/or measurement model are highly nonlinear. Additionally, as reported in [150], EKF performance is seriously affected when large deviations between the estimated and actual trajectories exist.

In order to improve the performance of EKF, especially for highly nonlinear process and/or measurement models, Unscented Kalman Filter (UKF) can be employed [63], [152]–[154]. The main idea behind UKF is that a Gaussian distribution can be approximated relatively easily whereas approximating an arbitrary nonlinear function is more difficult [152]. Contrary to EKF,

derivation of Jacobian matrices is not necessary in UKF. This can be beneficial in cases where the derivation of Jacobian matrices is difficult and noisy, and therefore possible implementation difficulties are avoided [153]. To that end, instead of directly linearizing the nonlinear process and/or measurement models, UKF uses a deterministic sampling method to generate a minimal number of sigma points that capture mean and covariance estimates. The sample points are then propagated through the nonlinear process/measurement models and the mean and covariance of the estimate are numerically recovered.

It can be theoretically shown that UKF accuracy is comparable to third–order Taylor series expansion, for any nonlinearity [152]. As an example, for the problem of spacecraft localization, performance of EKF was compared to UKF in [154]. For two sample missions of Earth–Moon transfer and geostationary orbit rising, it was observed that UKF offers better performance in terms of localization accuracy and consistency of the estimates. Also, in [63], three–axis attitude determination of a UAV was considered. In that work, the authors found that UKF robustness to noise is much better than EKF and concluded that it is a better alternative, compared to conventional EKF. In summary, UKF performance is superior to EKF as (i) it better copes with nonlinearities in process and measurement models and (ii) it avoids use of Jacobian matrices, which can be noisy and difficult to derive/implement.

It is worthwhile to mention that the performance of both EKF and UKF heavily depends on accurate a priori knowledge of noise statistics i.e., process and measurement covariance matrices. It has been reported that uncertain a priori knowledge of noise statistics may result in very poor performance of EKF and UKF filters and in extreme cases, the system may become unstable [155]–[158]. In such cases, adaptive techniques are effective solutions to improve performance of Kalman filters [159], EKF [160]–[162] and UKF [163]–[165]. Some of the areas in which adaptive Kalman filtering has been investigated include mobile robot localization [160] and parameter estimation [165], pose estimation in visual servoing [141], [142], [155], [162] and aerospace vehicle attitude estimation [63], [156], [157]. Regarding the adaptation law, several schemes such as maximum likelihood, and correlation and covariance matching techniques have been proposed [155]. Among the mentioned schemes, covariance matching is reported to require less computational resources [142] and therefore is more appropriate for UAV online implementation.

Here, formulation of EKF– and UKF–based state estimation with known noise statistics, along with Linear Quadratic Regulator (LQR) laws for quadcopter flight are first presented. Two metrics will be considered to assess the estimation accuracy and overall control performance of each algorithm. Then, Adaptive Extended Kalman Filter (AEKF) and Adaptive Unscented Kalman Filter (AUKF) are formulated to achieve autonomous flight of a quadcopter UAV in presence of uncertain process and measurement noise statistics. To the best of authors' knowledge, this is the first work to employ covariance matching AEKF and AUKF for autonomous quadcopter flight. Contrary to previous works with simplified linear models, e.g. [141], [145], full nonlinear dynamic model of the quadcopter is considered in the present section. A second advantage of this work, compared to [63], [156], is that the obtained results are not used only for state estimation and it is shown that they are accurate enough to be used in feedback control laws. This, in turn, provides the solution for autonomous flight control of a quadcopter UAV, even in presence of uncertain noise statistics.

The rest of this chapter is as follows. Mathematical modeling of a quadcopter and design of LQR controller are presented in Section 3.2 and Section 3.3, respectively. Then, algorithms for EKF, UKF, and their adaptive counterparts, i.e. AEKF and AUKF are discussed in Section 3.4. A case study is developed in Section 3.5 to compare the performance of EKF to UKF with known noise statistics and various initial conditions through simulations. Then, a scenario is developed to compare the performance of conventional EKF and UKF to their adaptive counterparts in presence of unknown noise statistics. Summaries are given in Section 3.6.

3.2 Mathematical model of a quadcopter

A quadcopter UAV is equipped with two pairs of rotors, turning in opposite directions, see Fig. 3.1 [166].
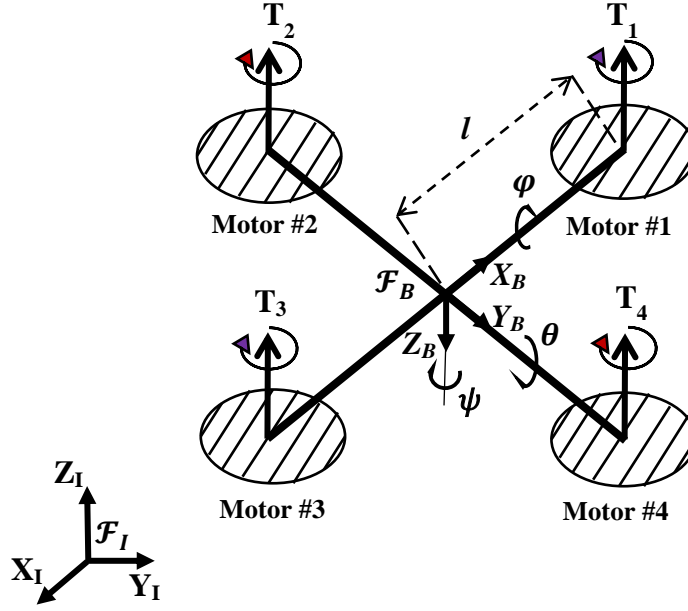
Fig. 3.1 Schematic view of a quadcopter

Here, let $\mathcal{F}_I = \{X_I, Y_I, Z_I\}$ and $\mathcal{F}_B = \{X_B, Y_B, Z_B\}$ be the inertial frame and body frame, respectively. What is more, $\boldsymbol{p}_B = [X \quad Y \quad Z]^T$ and $\boldsymbol{\Phi}_B = [\varphi \quad \theta \quad \psi]^T$ denote position of the body frame $\mathcal{F}_B$ in the inertial frame and quadcopter attitude (roll, pitch and yaw angles), respectively. A quadcopter can control its position and attitude by varying its rotors' speed. Vertical motion of a quadcopter is realized by simultaneously changing the speed (and thus lift force) of all four rotors. Translational motion in X_B −direction is realized by variation of the speed of rotor #1 and rotor #3. Similarly, an increase in the speed of rotor #2 with associated decrease in speed of rotor #4 will result in translation in Y_B −direction. The roll and pitch control is obtained by lift imbalance in rotor pairs # (2, 4) and (1, 3), respectively. Finally, the yaw motion is also controlled by an equal change in the speed of rotor pairs # (1, 3) and # (2, 4). Here, a number of assumptions are made to develop the nonlinear dynamic model of a quadcopter.

Assumption 1: It is assumed that the center of gravity of the quadcopter coincides with the origin of its body frame.

Assumption 2: The UAV is assumed to be symmetric such that its inertial matrix is diagonal.

Assumption 3: Lift and drag forces are assumed to be proportional to the square of the speed of the rotors [167], [168].

Assumption 4: The axes of the rotors are assumed to be fully aligned with the body $z-$axis.

For a quadcopter (rigid–body), kinematic equations of motion are given by:

$$\begin{cases} \dot{X} = v_X, \\ \dot{Y} = v_Y, \\ \dot{Z} = v_Z, \end{cases} \tag{3.1}$$

$$\begin{bmatrix} \dot{\varphi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & s_\varphi t_\theta & c_\varphi t_\theta \\ 0 & c_\varphi & -s_\varphi \\ 0 & s_\varphi/c_\theta & c_\varphi/c_\theta \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix}, \tag{3.2}$$

where φ, θ and ψ are the roll, pitch and yaw angles and $s_{(\alpha)}$, $c_{(\alpha)}$ and $t_{(\alpha)}$ denote trigonometric sine, cosine and tangent functions of the variable α. Also, p, q and r denote the three components of the angular velocity vector, where $\boldsymbol{\omega}_B = \begin{bmatrix} \omega_x, \omega_y, \omega_z \end{bmatrix}^T = [p, q, r]^T$ denotes quadcopter angular velocity expressed in the body frame. The corresponding rotation matrix from the body frame to the inertia frame is given by:

$$\boldsymbol{R}_B = \begin{bmatrix} c_\psi c_\theta & -s_\psi c_\varphi + c_\psi s_\theta s_\varphi & s_\psi s_\varphi + c_\psi s_\theta c_\varphi \\ s_\psi c_\theta & c_\psi c_\varphi + s_\psi s_\theta s_\varphi & -c_\psi s_\varphi + s_\psi s_\theta c_\varphi \\ -s_\theta & c_\theta s_\varphi & c_\theta c_\varphi \end{bmatrix}, \tag{3.3}$$

Also, from Newton's second law, the translational dynamics equations of motion in the inertial frame are given by [157]:

$$\begin{cases} m\ddot{X} = \left(s_\psi s_\varphi + c_\psi s_\theta c_\varphi \right) \sum_{i=1}^{4} f_{b,i}, \\ m\ddot{Y} = \left(-c_\psi s_\varphi + s_\psi s_\theta c_\varphi \right) \sum_{i=1}^{4} f_{b,i}, \\ m\ddot{Z} = mg - \left(c_\theta c_\varphi \right) \sum_{i=1}^{4} f_{b,i}, \end{cases} \tag{3.4}$$

in which m is the mass of the quadcopter and $g = 9.81 \frac{m}{s^2}$. In Eq. 3.4, $f_{b,i}$ is the thrust generated by the $i-th$ motor, expressed in body frame. It is assumed that $f_{b,i} = b\Omega_i^2$, where Ω_i is the rotation speed of the $i-th$ rotor and b is a proportionality coefficient. The quadcopter rotational dynamics is governed by Euler's second law and is given by [166]:

$$
\begin{cases}
I_{xx}\dot{\omega}_x = \omega_y\omega_z(I_{yy} - I_{zz}) + bl(\Omega_4^2 - \Omega_2^2), \\
I_{yy}\dot{\omega}_y = \omega_x\omega_z(I_{zz} - I_{xx}) + bl(\Omega_3^2 - \Omega_1^2), \\
I_{zz}\dot{\omega}_z = \omega_x\omega_y(I_{xx} - I_{yy}) + d(-\Omega_1^2 - \Omega_3^2 + \Omega_2^2 + \Omega_4^2),
\end{cases}
\tag{3.5}
$$

where I_{xx}, I_{yy} and I_{zz} are the quadcopter moments of inertia in the body frame. Also, in Eq. 3.5, l and d are the distance from a rotor's center to quadcopter geometric center and the drag coefficient, respectively. It can be already seen that the kinematic rotational equations of motion and all dynamic equations of motion of the quadcopter are nonlinear. Also, it can be observed that the translational dynamics equations of motion are coupled with rotational dynamics. One can linearize the above model about a given hovering point to obtain:

$$
\dot{x} = Ax + Bu,
\tag{3.6}
$$

where x is the state vector, i.e. $x = [X \quad \dot{X} \quad Y \quad \dot{Y} \quad Z \quad \dot{Z} \quad \varphi \quad \omega_x \quad \theta \quad \omega_y \quad \psi \quad \omega_z]^T$ and $u = [\Omega_1 \quad \Omega_2 \quad \Omega_3 \quad \Omega_4]^T$, i.e., the rotor speeds, is the input vector. Also, in Eq. 3.6, A and B are given by:

$$
A = \frac{\partial F}{\partial X}\Big|_{\substack{x^* \\ u^*}},
\tag{3.7}
$$

$$
B = \frac{\partial F}{\partial u}\Big|_{\substack{x^* \\ u^*}},
\tag{3.8}
$$

where x^* and u^* denote the hovering trim point and its associated input, respectively. Also, here F represents quadcopter equations of motion obtained in Eqs. 3.1, 3.2, 3.4 and 3.5 in compact form. For a hovering mission, the output to be tracked consists of four variables, namely the

position vector $\{X, Y, Z\}$ in the inertial frame and the yaw angle ψ. In the next subsection, LQR control laws for autonomous flight of a quadcopter UAV will be designed.

3.3 Linear quadratic regulator for the hovering mission

It can be verified that the obtained (A, B) pair in Eq. 3.7 and Eq. 3.8 is controllable [169]. Therefore, one can proceed to design a LQR controller by considering a possible cost function over infinite horizon given by [170] :

$$J(u) = \int_0^\infty \left(x^T Q_{LQR} x + u^T R_{LQR} u \right) dt, \qquad (3.9)$$

in which $Q_{LQR} \in \mathbb{R}^{12 \times 12}$ and $R_{LQR} \in \mathbb{R}^{4 \times 4}$ are positive definite weight matrices to penalize the deviation from the reference setpoint and the magnitude of the control signal, respectively. The above cost function can be minimized by state feedback law of the form:

$$u = -K_{LQR}(x - x^*) + u^*, \qquad (3.10)$$

Thus, the optimal gain K_{LQR} is obtained from:

$$K_{LQR} = R_{LQR}^{-1} B^T P, \qquad (3.11)$$

and P in turn is obtained from the following Algebraic Riccati Equation (ARE) [170]:

$$A^T P + PA - PBR_{LQR}^{-1} B^T P + Q_{LQR} = 0. \qquad (3.12)$$

The positive definite matrix Q_{LQR} can be chosen to be diagonal. For fixed R_{LQR}, if one increases the entries of Q_{LQR}, the deviation from the reference trajectory will be heavily penalized and the states rapidly converge to the desired values, i.e. high control gains [169]. The positive definite matrix R_{LQR} can also be chosen to be diagonal. In that case, for fixed Q_{LQR}, if one increases the entries of R_{LQR}, the magnitude of the control signal will be heavily penalized and the control

gains will be small. Thus, large values in the entries of $\boldsymbol{R}_{LQR}$ will result in slower system response and control laws with relatively small magnitude [169]. Proper selection of $\boldsymbol{Q}_{LQR}$ and $\boldsymbol{R}_{LQR}$ matrices will result in timely convergence of states to their desired value, without violating actuator saturation limits in terms of excessively large control inputs.

3.4 Kalman filtering
3.4.1 EKF state estimation

The original formulation of Kalman filter is only applicable to systems with linear process and measurement model. EKF is an ad hoc extension of the original Kalman filter that linearizes the process and measurement model about the available best estimate of the system states. In summary, quadcopter equations of motion were given in Eqs. 3.1, 3.2, 3.4 and 3.5 or, equivalently, by $\dot{\boldsymbol{x}} = \boldsymbol{F}(\boldsymbol{x}, \boldsymbol{u})$ in compact form. Adding the inherent uncertainty in the dynamic equations of motion, the discretized process and measurement models are given by:

$$\boldsymbol{x}_k = \boldsymbol{f}(\boldsymbol{x}_{k-1}, \boldsymbol{u}_k) + \boldsymbol{w}_k, \tag{3.13}$$

$$\boldsymbol{y}_k = \boldsymbol{H}\boldsymbol{x}_k + \boldsymbol{v}_k, \tag{3.14}$$

where $\boldsymbol{f}$ denotes discretized equations of motion control input and $\boldsymbol{w}_k$ and $\boldsymbol{v}_k$ are the process and measurement noise with zero mean Gaussian distribution and covariance $\boldsymbol{Q}_k^*$ and $\boldsymbol{R}_k^*$, respectively. In this formulation, the unmodelled dynamics and sensor noise are represented by covariance matrices $\boldsymbol{Q}_k^*$ and $\boldsymbol{R}_k^*$, respectively [171], [172]. It is worthwhile to note that in the current problem, the measurement model i.e. $\{X, Y, Z, \varphi, \theta, \psi\}$ is linear and therefore $\boldsymbol{H} \in \mathbb{R}^{6 \times 12}$ is a constant matrix, given as follows:

$$\boldsymbol{H} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}.$$

Time–update equations of EKF are then given by:

39

$$\hat{x}_{k|k-1} = f(\hat{x}_{k-1|k-1}, u_k), \tag{3.15}$$

$$P_{k|k-1} = A_{k-1} P_{k-1|k-1} A_{k-1}^T + Q_k^*, \tag{3.16}$$

where $\hat{x}_{k|k-1}$ and $\hat{x}_{k-1|k-1}$ are a priori estimate at step k and a posteriori estimate at step $k-1$, respectively. Also, $P_{k|k-1}$ denotes $\hat{x}_{k|k-1}$ error covariance and A_{k-1} is the Jacobian matrix given by:

$$A_{k-1} = \frac{\partial f}{\partial x}|_{\hat{x}_{k-1|k-1}, \ u_{k-1}}. \tag{3.17}$$

Once the measurements are available, the update equations are given by:

$$K_k = P_{k|k-1} H^T (H P_{k|k-1} H^T + R_k^*)^{-1}, \tag{3.18}$$

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k(y_k - H\hat{x}_{k|k-1}), \tag{3.19}$$

$$P_{k|k} = (I - K_k H) P_{k|k-1}, \tag{3.20}$$

where K_k is the Kalman gain and updated state estimate and updated estimation error covariance are obtained from Eq. 3.19 and Eq. 3.20, respectively. It is important to note that EKF is a suboptimal filter for nonlinear systems and in cases where the linearization error is large, its performance may become unsatisfactory.

3.4.2 UKF state estimation

UKF is a relatively new technique to improve EKF performance by avoiding the linearization step. In UKF, a deterministic sampling technique i.e. unscented transform is utilized to generate a minimal set of sample points (called sigma points hereafter) from the state a priori mean and covariance. Then, these sample points are propagated through the nonlinear process model and a

40

posteriori mean and covariance are numerically recovered [153], [173]. In UKF algorithm, in the first step, $2n + 1$ weighted sigma points are generated [63], [174]:

$$
\begin{cases}
\boldsymbol{\chi}^0_{k-1|k-1} = \widehat{\boldsymbol{x}}_{k-1|k-1}, \\
\boldsymbol{\chi}^i_{k-1|k-1} = \widehat{\boldsymbol{x}}_{k-1|k-1} + \left(\sqrt{(n + \lambda)\boldsymbol{P}_{k-1|k-1}} \right)_i, \\
\boldsymbol{\chi}^{i+n}_{k-1|k-1} = \widehat{\boldsymbol{x}}_{k-1|k-1} - \left(\sqrt{(n + \lambda)\boldsymbol{P}_{k-1|k-1}} \right)_i,
\end{cases}
\tag{3.21}
$$

where $i = 1, \dots, n$, $\lambda \in \mathbb{R}$ is a scaling parameter, $\left(\sqrt{(n + \lambda)\boldsymbol{P}_{k-1|k-1}} \right)_i$ is the $i-$th column (or row) of matrix square root of $(n + \lambda)\boldsymbol{P}_{k-1|k-1}$ and n is the number of system states. Then, each sigma point is propagated through the process model:

$$
\widehat{\boldsymbol{x}}^i_{k|k-1} = \boldsymbol{f}(\boldsymbol{\chi}^i_{k-1|k-1}, \boldsymbol{u}_{k-1}),
\tag{3.22}
$$

The predicted state estimate and the estimation error covariance are then given by:

$$
\widehat{\boldsymbol{x}}_{k|k-1} = \sum_{i=0}^{2n} W_i \widehat{\boldsymbol{x}}^i_{k|k-1},
\tag{3.23}
$$

$$
\boldsymbol{P}_{k|k-1} = \sum_{i=0}^{2n} W_i \left(\widehat{\boldsymbol{x}}^i_{k|k-1} - \widehat{\boldsymbol{x}}_{k|k-1} \right) \left(\widehat{\boldsymbol{x}}^i_{k|k-1} - \widehat{\boldsymbol{x}}_{k|k-1} \right)^T + \boldsymbol{Q}^*_k,
\tag{3.24}
$$

where W_i is given by:

$$
\begin{cases}
W_0 = \dfrac{\lambda}{n + \lambda}, \\
W_i = \dfrac{1}{2(n + \lambda)},
\end{cases}
\tag{3.25}
$$

and λ is a scaling parameter in unscented transform [175]. A similar procedure is adopted to

calculate the measurement and its error covariance:

$$
\begin{cases}
\chi_{k|k-1}^0 = \hat{x}_{k|k-1}, \\
\chi_{k|k-1}^i = \hat{x}_{k|k-1} + \left(\sqrt{(n+\lambda)P_{k|k-1}} \right)_i, \\
\chi_{k|k-1}^{i+n} = \hat{x}_{k|k-1} - \left(\sqrt{(n+\lambda)P_{k|k-1}} \right)_i,
\end{cases}
\tag{3.26}
$$

$$
\hat{y}_k^i = H\chi_{k|k-1}^i,
\tag{3.27}
$$

$$
\hat{y}_k = \sum_{i=0}^{2n} W_i \hat{y}_k^i,
\tag{3.28}
$$

$$
P_k^y = \sum_{i=0}^{2n} W_i \left(\hat{y}_k^i - \hat{y}_k \right) \left(\hat{y}_k^i - \hat{y}_k \right)^T + R_k^*,
\tag{3.29}
$$

$$
P_k^{xy} = \sum_{i=0}^{2n} W_i \left(\hat{x}_{k|k-1}^i - \hat{x}_{k|k-1} \right) \left(\hat{y}_k^i - \hat{y}_k \right)^T.
\tag{3.30}
$$

Once the measurements are available, the update equations are given by:

$$
K_k = P_k^{xy} \left(P_k^y \right)^{-1},
\tag{3.31}
$$

$$
\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k (y_k - \hat{y}_k),
\tag{3.32}
$$

$$
P_{k|k} = P_{k|k-1} - K_k P_k^y K_k^T.
\tag{3.33}
$$

3.4.3 AEKF state estimation

As it was discussed earlier, performance of EKF and UKF heavily relies on accurate knowledge of noise statistics i.e. $\boldsymbol{Q}_k^*$ and $\boldsymbol{R}_k^*$. However, in particular for $\boldsymbol{Q}_k^*$, accurate a priori knowledge cannot be guaranteed [176]–[178]. To cope with such scenarios, a covariance matching scheme is formulated. The main idea behind this scheme is to make the theoretical covariance conform to the observed statistics of the system over a given window N [155]. In general, small window size can better accommodate fast dynamics of the system but may result in reduced accuracy and, in extreme cases, lead to instability. On the other hand, a large window size usually avoids system instability, at the cost of reduced flexibility to system fast dynamics [142]. For adaptive filtering, measurement innovation is defined as:

$$r_k = y_k - H\widehat{x}_{k|k-1}. \tag{3.34}$$

An unbiased estimator for $\boldsymbol{r}_k$ can written as:

$$\bar{r} = \frac{1}{N}\sum_{j=1}^{N} r_j \tag{3.35}$$

where the covariance is:

$$cov_r = \frac{1}{N-1}\sum_{j=1}^{N}(r_j - \bar{r})(r_j - \bar{r})^T, \tag{3.36}$$

The expected value of the above covariance is   :

$$E[cov_r] = \frac{1}{N}\sum_{j=1}^{N}\boldsymbol{\Gamma}_k + \boldsymbol{R}_k^* \tag{3.37}$$

where:

$$\mathbf{\Gamma}_k = \mathbf{H} \mathbf{P}_k^- \mathbf{H}^T, \tag{3.38}$$

Therefore, from the above equations, one can write:

$$\mathbf{R}_k^\dagger = \frac{1}{N-1} \sum_{j=1}^{N} \left\{ (\mathbf{r}_j - \bar{\mathbf{r}})(\mathbf{r}_j - \bar{\mathbf{r}})^T - \frac{N-1}{N} \mathbf{\Gamma}_k \right\} \tag{3.39}$$

Similarly, for state noise sample, one can write:

$$\mathbf{a}_k = \widehat{\mathbf{x}}_{k|k} - \widehat{\mathbf{x}}_{k|k-1}, \tag{3.40}$$

And an unbiased estimator for $\mathbf{a}_k$ can be formed as:

$$\bar{\mathbf{a}} = \frac{1}{N} \sum_{j=1}^{N} \mathbf{a}_j \tag{3.41}$$

Noting that the covariance of $\mathbf{a}$ and its expected value are given by [179]:

$$cov_a = \frac{1}{N-1} \sum_{j=1}^{N} (\mathbf{a}_j - \bar{\mathbf{a}})(\mathbf{a}_j - \bar{\mathbf{a}})^T, \tag{3.42}$$

$$E[cov_a] = \frac{1}{N} \sum_{j=1}^{N} \mathbf{\Delta}_k + \mathbf{Q}_k^*, \tag{3.43}$$

One can write:

$$Q_k^\dagger = \frac{1}{N-1}\sum_{j=1}^{N}\left\{(a_j - \bar{a})(a_j - \bar{a})^T - \frac{N-1}{N}\Delta_k\right\} \tag{3.44}$$

where:

$$\Delta_k = A_k P_{k-1}^+ A_k^T - P_k^+, \tag{3.45}$$

It can be readily seen from the above equations that the computational requirement of adaptive scheme depends on the adaptation window size N. These computationally–expensive calculations are a potential drawback, especially for onboard implementation with limited computational resources. Therefore, a computationally–efficient recursive formulation is written as [141], [145]:

$$Q_{k+1}^\dagger = Q_k^\dagger + \frac{\varpi_k}{N-1}\left\{(a_k - \bar{a}_k)(a_k - \bar{a}_k)^T - (a_{k-N} - \bar{a}_k)(a_{k-N} - \bar{a}_k)^T \right.$$
$$\left. + \frac{1}{N}(a_k - a_{k-N})(a_k - a_{k-N})^T + \frac{N-1}{N}(\Delta_{k-N} - \Delta_k)\right\}, \tag{3.46}$$

$$R_{k+1}^\dagger = R_k^\dagger + \frac{\varpi_k}{N-1}\left\{(r_k - \bar{r}_k)(r_k - \bar{r}_k)^T - (r_{k-N} - \bar{r}_k)(r_{k-N} - \bar{r}_k)^T \right.$$
$$\left. + \frac{1}{N}(r_k - r_{k-N})(r_k - r_{k-N})^T + \frac{N-1}{N}(\Gamma_{k-N} - \Gamma_k)\right\}. \tag{3.47}$$

What is more, in the above equations, ϖ_k is a fading factor such that higher weights are given to the most recent samples whereas initial samples are given less significance, see [179] for more details. The original formulation of ϖ_k in [179] was given as:

$$\varpi_k = (k-1)(k-2)\cdots(k-\varrho)/k^\varrho, \tag{3.48}$$

where $\lim_{k\to\infty}\varpi_k = 1$ and also use of initial ϱ noise samples is delayed to offer better stability. However, the above definition results in excessively large numerator/denominator terms for large values of k and/or ϱ. As a result, one is confined to choose ϱ such that numerical problems are

prevented. In order to avoid the mentioned mathematical difficulty, here we define a different fading factor:

$$\varpi_k = \tanh(ck), \qquad \text{if } k \geq N, \tag{3.49}$$

where $c > 0$ is free parameter to adjust the fading factor values ($0 < c < 10^{-2}$ results in fading factors similar to those given in Eq. 3.48). One may note that the fading factors in Eq. 3.49 tend to offer the same characteristics of the ones in Eq. 3.48 whereas use of excessively large numbers is avoided and therefore Eq. 3.49 is applicable to large values of k and/or N.

3.4.4 AUKF state estimation

A relatively similar recursive approach can be adopted to adjust the noise statistics in AUKF. Yet, one must notice that the Jacobian matrices are not available in this approach and therefore the modified adaptation laws for AUKF are given by:

$$
\begin{aligned}
\boldsymbol{Q}_{k+1}^{\dagger} = \boldsymbol{Q}_k^{\dagger} + \frac{\varpi_k}{N-1} \Big\{ & (\boldsymbol{a}_k - \bar{\boldsymbol{a}}_k)(\boldsymbol{a}_k - \bar{\boldsymbol{a}}_k)^T - (\boldsymbol{a}_{k-N} - \bar{\boldsymbol{a}}_k)(\boldsymbol{a}_{k-N} - \bar{\boldsymbol{a}}_k)^T \\
& + (\boldsymbol{a}_k - \boldsymbol{a}_{k-N})(\boldsymbol{a}_k - \boldsymbol{a}_{k-N})^T \\
& + \frac{N-1}{N} \big(\boldsymbol{P}_{k|k} - \boldsymbol{P}_{k|k-1} + \boldsymbol{Q}_k^{\dagger} + \boldsymbol{P}_{k-N|k-N-1} - \boldsymbol{Q}_{k-N}^{\dagger} \\
& - \boldsymbol{P}_{k-N|k-N} \big) \Big\},
\end{aligned}
\tag{3.50}
$$

$$
\begin{aligned}
\boldsymbol{R}_{k+1}^{\dagger} = \boldsymbol{R}_k^{\dagger} + \frac{\varpi_k}{N-1} \Big\{ & (\boldsymbol{r}_k - \bar{\boldsymbol{r}}_k)(\boldsymbol{r}_k - \bar{\boldsymbol{r}}_k)^T - (\boldsymbol{r}_{k-N} - \bar{\boldsymbol{r}}_k)(\boldsymbol{r}_{k-N} - \bar{\boldsymbol{r}}_k)^T \\
& + (\boldsymbol{r}_k - \boldsymbol{r}_{k-N})(\boldsymbol{r}_k - \boldsymbol{r}_{k-N})^T - \frac{N-1}{N} \big(\boldsymbol{P}_k^y - \boldsymbol{P}_{k-N}^y \big) \Big\},
\end{aligned}
\tag{3.51}
$$

where $\boldsymbol{P}_k^y$ was given in Eq. 3.29 and ϖ_k, similar to previous section, is the weight accounting for the fading memory approach. In the next section, a case will be developed to evaluate the performance of EKF, UKF and their adaptive counterparts in different scenarios.

3.5 Case study, simulation results and discussion
3.5.1 EKF and UKF with certain noise statistics

In this section, a case study is developed to examine the feasibility of the approaches described in previous sections. For a quadcopter with the physical parameters shown in Table 3.1, the LQR design process was implemented in MATLAB. In the following simulations, Q_{LQR} was initialized as $I_{12} \in \mathbb{R}^{12 \times 12}$, i.e. identity matrix, and the following entries were adjusted by trial and error to improve control performance.

$$\begin{cases} Q_{LQR}(i,j) = 20 & \text{for } i = j = 1,3, \\ Q_{LQR}(i,j) = 50 & \text{for } i = j = 5, \\ Q_{LQR}(i,j) = 50 & \text{for } i = j = 11. \end{cases}$$

What is more, the weight matrix R_{LQR} was initialized as $0.01 \times I_4$. Having obtained the P matrix from the solution of the ARE equation, the optimal gain matrix K_{LQR} (and thus the optimal control law) was obtained.

Table 3.1 Physical parameters of case study quadcopter

	Numerical value	Unit
m	0.589	kg
I_{xx}	6.532×10^{-3}	kg.m^2
I_{yy}	6.532×10^{-3}	kg.m^2
I_{zz}	12.742×10^{-3}	kg.m^2
l	0.232	m

In the first scenario, it is assumed that the process and measurement noise statistics are a priori known. The process noise covariance is assumed to be a constant diagonal matrix and is given by:

$$\begin{cases} Q^*(i,j) = 10^{-5} & \text{for } i = j = 2,4,6, \\ Q^*(i,j) = 10^{-6} & \text{for } i = j = 8,10,12, \\ 0 & \text{elsewhere,} \end{cases}$$

where $Q^*(i,j)$ is the i −th row and j −th column of $\boldsymbol{Q}^*$ and the values are set in accordance with the available literature [180]–[182]. Also, the a priori known measurement noise covariance is given by:

$$\begin{cases} R^*(i,j) = 10^{-3} & \text{for } i = j = 1,2, \\ R^*(i,j) = 2 \times 10^{-3} & \text{for } i = j = 3, \\ R^*(i,j) = 10^{-3} & \text{for } i = j = 4,5,6. \end{cases}$$

Regarding the measurement noise covariance, the values for translational elements comply with position accuracy of a few centimeters. This is a conservative assumption as RTK–GPS is reported to offer 1–2 cm accuracy and 2–4 cm accuracy in horizontal and vertical measurements, respectively [183]–[185]. What is more, the rotational element covariance is consistent with angle measurements with a few degrees accuracy. Again, this is a conservative assumption as many attitude and heading reference systems provide accuracies up to 1–degree and even better than a degree in all three attitude angle measurements [186]–[188]. For the first case study, initial condition of the quadcopter is assumed to be $\boldsymbol{x}(0) = [0.5 \quad 0 \quad -0.5 \quad 0 \quad 1.5 \quad 0 \quad 10 \quad 0 \quad 10 \quad 0 \quad 10 \quad 0]^T$ and the command signal is given by $\boldsymbol{x}^*(0) = [0_{1\times4} \quad 2 \quad 0_{1\times7}]^T$. For the initial conditions and setpoint signal, the linear position and velocity are given in m and m/s while the angular position and velocity are given in deg and deg/s. For the described case study, simulations were carried out and obtained results for EKF and UKF are shown in Figs. 3.2–3.5. Also, estimation error mean and standard deviation and RMSE metric are given in Table 3.2 and Table 3.3, respectively.
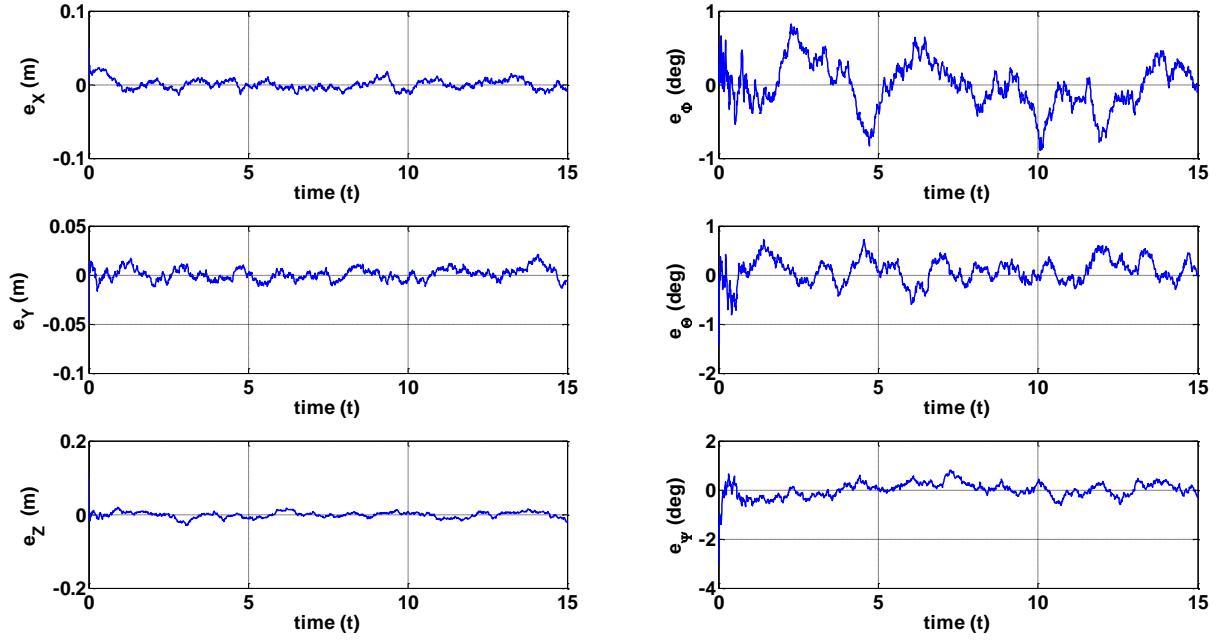
Fig. 3.2 EKF–based estimation error with known noise statistics
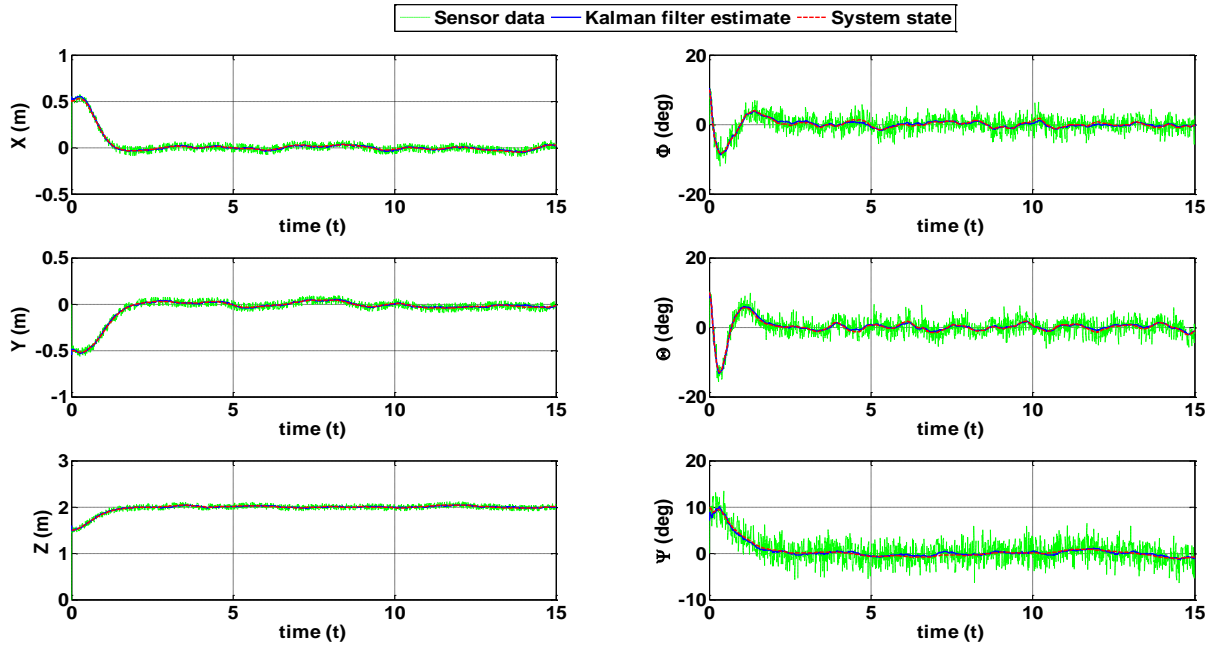
Fig. 3.3 EKF–based LQR control with known noise statistics
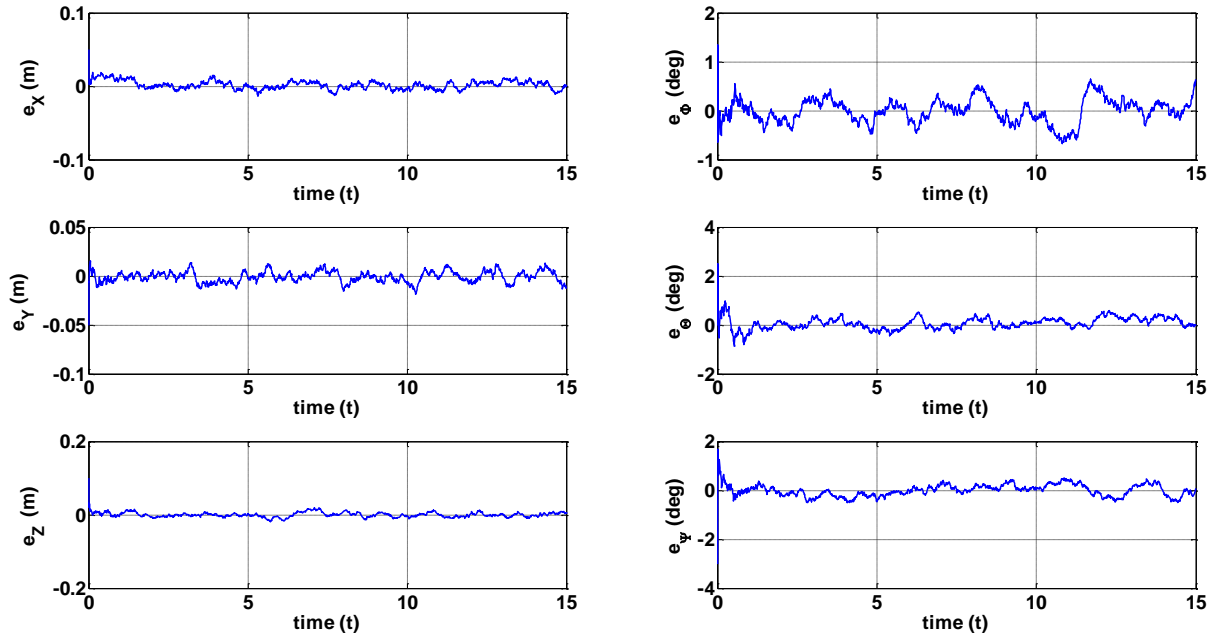
Fig. 3.4 UKF–based estimation error with known noise statistics
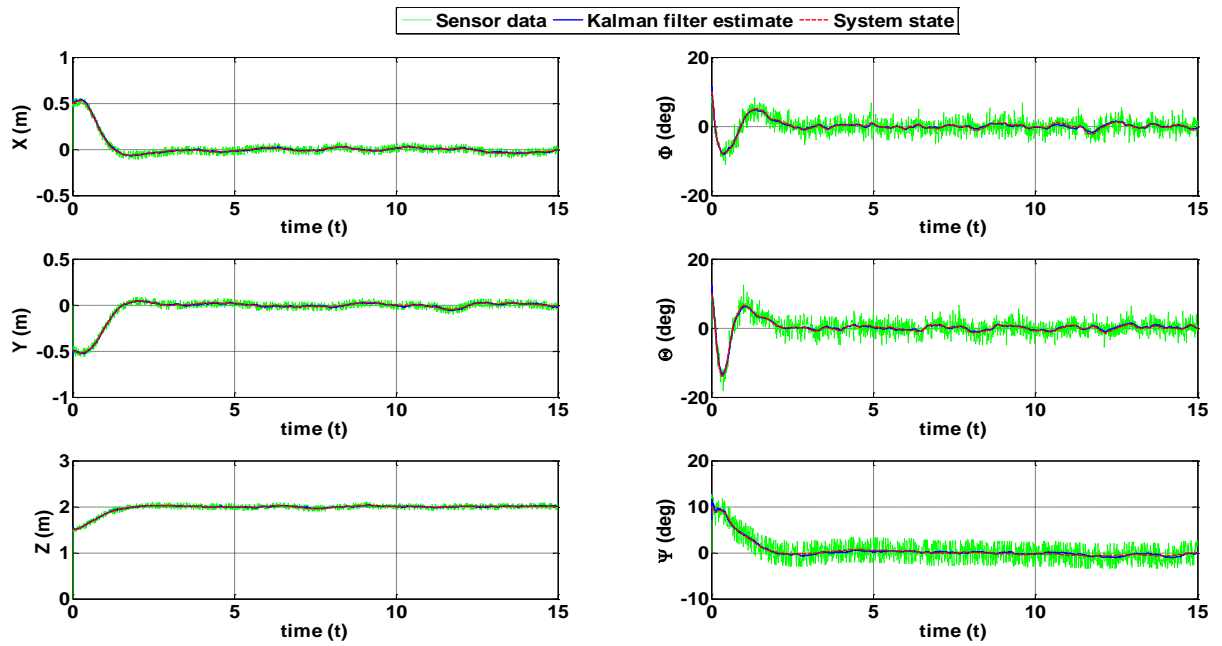
Fig. 3.5 UKF–based LQR control with known noise statistics

Table 3.2 estimation error mean and standard deviation with known noise statistics

		e_X (cm)	e_Y (cm)	e_Z (cm)	e_φ (deg)	e_θ (deg)	e_ψ (deg)
EKF	Mean	0.6	0.5	0.6	0.3	0.2	0.2
	STD	0.7	0.6	0.8	0.3	0.2	0.3
UKF	Mean	0.5	0.5	0.5	0.2	0.2	0.2
	STD	0.6	0.6	0.7	0.2	0.2	0.3

Table 3.3 RMSE metric with known noise statistics

	$RMSE_X$ (cm)	$RMSE_Y$ (cm)	$RMSE_Z$ (cm)	$RMSE_\varphi$ (deg)	$RMSE_\theta$ (deg)	$RMSE_\psi$ (deg)
EKF	11.6	12.8	9.7	1.7	2.2	2.0
UKF	11.3	12.2	9.7	1.7	2.2	2.0

In the above figures, where applicable, "Kalman filter estimate" is the best estimate of a state obtained from extended/unscented Kalman algorithms. In real–life scenarios, Kalman filter estimates are available for control design purposes and, as a result, these are the values used in the LQR feedback control law. The second element, "System state" obtained from Eq. 3.13, represents the actual value of a state. Due to the fact that w_k is not known in real–life scenarios, system states are not actually known and therefore cannot be used in feedback control law(s). In the mentioned figures, system states merely serve as the reference for comparison purposes and are not used as feedback in LQR control. Finally, "Sensor data" represents data obtained from sensor(s) for a given state. From Fig. 3.3 and Fig. 3.5, it can be readily seen that sensory data obtained from Eq. 3.14 take account of the measurement noise present in real–life scenarios.

In Fig. 3.2 and Fig. 3.4, e_ξ is the estimation error of the state ξ where $\xi \in \{X, Y, Z, \varphi, \theta, \psi\}$. Analysis of the results presented in Table 3.2 indicates that subcentimeter–level mean state estimation error has been achieved in all translational degrees of freedom, for both EKF and UKF. It is worthwhile to mention that noisy data with a few centimeters accuracy, representing sensor output, were fed to both EKF and UKF in translational channels and both algorithms have resulted in noticeably smoother and several times more accurate state estimates, in comparison with raw sensor output.

51

From Table 3.2, it can be seen that UKF mean estimation error is similar to EKF in Y channel but the former outperforms by approximately 0.1 cm in X and Z channels. Similarly, regarding the attitude angles, subdegree–level mean estimation error has been achieved in all rotational channels. In particular, UKF outperforms EKF in φ channel by 0.1 degree whereas mean estimation errors of both algorithms in θ and ψ channels are similar. In accordance with Fig. 3.3 and Fig. 3.5, simulation results indicate that UKF superior pose estimation results in better RMSE in X and Y translational degrees of freedom of the quadcopter in its hovering flight. More specifically, while $\mathrm{RMSE_Z}$ in UKF is similar to that of EKF, it is seen that the $\mathrm{RMSE_X}$ and $\mathrm{RMSE_Y}$ of UKF–based algorithm are better than EKF–based results by $0.3\ cm$ and $0.6\ cm$, respectively. For the rotational degrees of freedom, UKF–based and EKF–based results offer similar RMSE metric in all channels.

It is also important to examine the performance of EKF–based and UKF–based algorithms with various initial conditions. With that in mind, 10 scenarios with different initial conditions are defined. In the first 5 scenarios, the attitude angles increase in 10–degree steps such that in the 5th scenario the initial conditions are given by $\boldsymbol{x}(0) = [0.5\quad 0\quad -0.5\quad 0\quad 1.5\quad 0\quad 60\quad 0\quad 60\quad 0\quad 60\quad 0]^T$. Then, in the next 5 scenarios, the initial displacement of X and Y coordinates from the origin increase in 0.5 m steps such that the initial conditions for the last scenario is given by $\boldsymbol{x}(0) = [3\quad 0\quad -3\quad 0\quad 1.5\quad 0\quad 10\quad 0\quad 10\quad 0\quad 10\quad 0]^T$. For the described scenarios, simulations were carried out and the obtained results are shown in Table 3.4–3.7.

Table 3.4 estimation error mean and standard deviation in attitude–angle initial conditions scenarios

			e_X (cm)	e_Y (cm)	e_Z (cm)	e_φ (deg)	e_θ (deg)	e_ψ (deg)
Scenario # 1	EKF	Mean	0.6	0.5	0.6	0.3	0.2	0.2
		STD	0.7	0.6	0.8	0.3	0.2	0.3
	UKF	Mean	0.5	0.5	0.5	0.2	0.2	0.2
		STD	0.6	0.6	0.7	0.2	0.2	0.3
Scenario # 2	EKF	Mean	0.6	0.5	0.6	0.3	0.2	0.2
		STD	0.7	0.6	0.8	0.3	0.2	0.3
	UKF	Mean	0.5	0.5	0.5	0.2	0.2	0.2
		STD	0.6	0.6	0.7	0.2	0.2	0.3
Scenario # 3	EKF	Mean	0.6	0.5	0.6	0.3	0.2	0.2
		STD	0.7	0.6	0.8	0.3	0.3	0.3
	UKF	Mean	0.5	0.5	0.5	0.2	0.2	0.2
		STD	0.6	0.6	0.7	0.2	0.2	0.3
Scenario # 4	EKF	Mean	0.6	0.5	0.6	0.3	0.3	0.2
		STD	0.7	0.6	0.8	0.3	0.3	0.3
	UKF	Mean	0.5	0.5	0.5	0.2	0.2	0.2
		STD	0.6	0.6	0.7	0.2	0.2	0.3
Scenario # 5	EKF	Mean	0.6	0.5	0.6	0.3	0.3	0.3
		STD	0.7	0.6	0.8	0.3	0.3	0.3
	UKF	Mean	0.5	0.5	0.5	0.2	0.2	0.2
		STD	0.6	0.6	0.7	0.2	0.2	0.3

Table 3.5 RMSE metric in attitude–angle initial conditions scenarios

		$RMSE_X$ (cm)	$RMSE_Y$ (cm)	$RMSE_Z$ (cm)	$RMSE_\varphi$ (deg)	$RMSE_\theta$ (deg)	$RMSE_\psi$ (deg)
Scenario # 1	EKF	13.4	14.7	10.1	2.5	3.1	3.4
	UKF	13.0	14.1	10.2	2.5	3.1	3.4
Scenario # 2	EKF	15.3	16.4	10.9	3.3	4.2	4.3
	UKF	15.0	15.7	11.0	3.3	4.2	4.2
Scenario # 3	EKF	17.7	17.8	12.1	4.1	5.5	4.7
	UKF	17.3	17.0	12.1	4.0	5.4	4.6
Scenario # 4	EKF	20.5	18.3	13.5	4.8	6.7	4.8
	UKF	20.0	17.5	13.5	4.5	6.6	4.6
Scenario # 5	EKF	25.6	18.4	16.1	5.8	8.2	5.0
	UKF	25.1	17.7	16.2	5.7	8.0	5.0

Table 3.6 estimation error mean and standard deviation in position vector initial conditions scenarios

			e_X (cm)	e_Y (cm)	e_Z (cm)	e_φ (deg)	e_θ (deg)	e_ψ (deg)
Scenario # 6	EKF	Mean	0.6	0.5	0.6	0.3	0.2	0.2
		STD	0.7	0.6	0.8	0.3	0.2	0.3
	UKF	Mean	0.5	0.5	0.5	0.2	0.2	0.2
		STD	0.6	0.6	0.7	0.2	0.2	0.3
Scenario # 7	EKF	Mean	0.6	0.5	0.6	0.3	0.2	0.2
		STD	0.7	0.6	0.8	0.3	0.2	0.3
	UKF	Mean	0.5	0.5	0.5	0.2	0.2	0.2
		STD	0.6	0.6	0.7	0.2	0.2	0.3
Scenario # 8	EKF	Mean	0.6	0.5	0.6	0.3	0.2	0.2
		STD	0.7	0.6	0.8	0.3	0.2	0.3
	UKF	Mean	0.5	0.5	0.5	0.2	0.2	0.2
		STD	0.6	0.6	0.7	0.2	0.2	0.3
Scenario # 9	EKF	Mean	0.6	0.5	0.6	0.3	0.2	0.2
		STD	0.7	0.6	0.8	0.4	0.2	0.3
	UKF	Mean	0.5	0.5	0.5	0.2	0.2	0.2
		STD	0.6	0.6	0.7	0.2	0.2	0.3
Scenario # 10	EKF	Mean	0.6	0.5	0.6	0.3	0.3	0.2
		STD	0.7	0.6	0.8	0.4	0.3	0.3
	UKF	Mean	0.5	0.5	0.5	0.2	0.2	0.2
		STD	0.6	0.6	0.7	0.2	0.2	0.3

Table 3.7 RMSE metric in position vector initial conditions scenarios

		$RMSE_X$ (cm)	$RMSE_Y$ (cm)	$RMSE_Z$ (cm)	$RMSE_\varphi$ (deg)	$RMSE_\theta$ (deg)	$RMSE_\psi$ (deg)
Scenario # 6	EKF	21.5	23.7	10.2	2.8	3.5	2.0
	UKF	21.1	23.0	10.2	2.8	3.5	2.0
Scenario # 7	EKF	31.6	34.3	11.3	4.0	5.0	2.2
	UKF	30.9	33.8	11.3	4.0	4.9	2.2
Scenario # 8	EKF	41.0	45.6	12.9	5.1	6.2	2.5
	UKF	40.4	44.8	13.0	5.1	6.2	2.5
Scenario # 9	EKF	51.1	57.0	15.1	6.2	7.8	3.3
	UKF	50.6	56.4	15.1	6.1	7.5	3.2
Scenario # 10	EKF	61.4	68.8	17.8	7.1	8.9	4.1
	UKF	60.9	68.3	17.8	7.1	8.8	4.0

From the above tables, it can be readily seen that both EKF–based and UKF–based algorithms maintain accurate state estimation over a wide range of initial conditions. In particular, for attitude–angle initial conditions scenarios, both algorithms result in subcentimeter and subdegree accuracy in estimation of translational and rotational channels in all scenarios, respectively. A marginal increase in estimation error of EKF–based method was observed in θ channel where the estimation error increased to 0.3 degrees in Scenario #4 and Scenario #5 and also in ψ channel where estimation error increased to 0.3 degrees in Scenario #5. This is due to the fact that EKF formulation is based on linearized dynamics of the vehicle and therefore its performance marginally degrades in large attitude angles. However, for a hovering mission, both algorithms rapidly converge to the vicinity of hovering conditions and therefore estimation accuracy is preserved over a wide range of attitude–angle initial conditions. Regarding the scenarios with different initial position vectors, it can also be seen that both algorithms maintain subcentimeter accuracy in estimation of translational channels and subdegree estimation accuracy in rotational channels. For the EKF–based method, a slight 0.1 degree increase in estimation error of θ channel was observed in the 10th scenario. This is due to the fact that the quadcopter carries out larger angle maneuvers to compensate the larger initial displacement error and therefore EKF–based method slightly degrades before it converges to the vicinity of the hovering point. Regarding the overall control performance, it can be seen from Table 3.5 and 3.7 that the RMSE metrics increase as the initial conditions grow further from the hovering point. This is attributed to the fact that the controller requires more time to compensate for larger deviations from the setpoint. Also, for LQR being a linear control strategy, the controller performance degrades as it starts from larger initial conditions, with respect to the hovering point and therefore larger RMSE metrics are obtained. However, simulation results show that the controller can successfully accomplish the hovering mission with both EKF–based and UKF–based algorithms, over a wide range of initial conditions.

It is also important to mention that UKF improved accuracy in state estimation and overall control of the quadcopter is usually associated with higher execution time. In this work, an Intel® Core™–i5 3.2 GHz processor based computer with 8.00 GB RAM was used to carry out the MATLAB–based simulations. Average execution time of EKF and UKF algorithms is shown in Table 3.8 for comparison.

55

Table 3.8 RMSE metric with known noise statistics

	Average execution time per iteration (ms)
EKF	0.13
UKF	1.23

From the above table, it is observed that EKF average execution time is approximately an order of magnitude shorter than UKF. More specifically, while UKF execution on average requires 1.23 ms in each time step, EKF only requires 0.13 ms for execution in each time step. From the computational resources perspective, this is a significant advantage for EKF as it provides relatively accurate results, yet at much lower computational cost. This conforms to the observation that EKF is a popular choice for state estimation, especially for onboard implementation with limited computational resources. It is concluded that for a quadcopter UAV with known noise statistics, the ultimate selection between EKF and UKF for state estimation is a compromise between available computational resources and estimation and overall control performance of the vehicle based on its required mission specifications.

At this point, it is important to recall that performance of both EKF and UKF heavily depends on accurate a priori knowledge of process and/or measurement noise statistics. If such information is not available, poor performance of conventional Kalman filters is expected and therefore adaptive schemes should be considered. This is the main theme of the next simulation scenarios.

3.5.2 Adaptive EKF and UKF with uncertain noise statistics

In this case study, it is assumed that the actual process and measurement noise statistics, i.e. Q^* and R^* are identical to those of the first scenario, but are not a priori known. In general, while it is relatively easier to obtain R^* from experimental sensory data and/or sensor specifications, a priori knowledge of Q^* may not be accurate or even not available for design purposes at all [142], [145]. Here, we first examine the performance of AEKF and AUKF in a worst–case scenario where no a priori knowledge of Q^* is available. In such cases, one can use the fact that Q^*, as a covariance matrix, is positive semi–definite and therefore it can be initialized as zero matrix. Also, it is assumed that a priori measurement noise statistics are inaccurate such that the best estimate of measurement noise covariance is an order of magnitude smaller than its actual

value. For such a scenario with unknown noise statistics, simulations were carried out and obtained results are shown in Figs. 3.6–3.13, Table 3.9 and Table 3.10.

From the obtained results, it can be readily observed that the performance of both EKF and UKF have degraded severely and both algorithms fail to follow the command signal properly in uncertain noise statistics scenario. More specifically, as it can be seen from Fig. 3.6 and Fig. 3.8, both algorithms have resulted in excessively increasing mean estimation errors which will eventually lead to total loss of quadcopter control. This, in turn, has led to very large RMSE metrics, verifying that overall control performance of non–adaptive Kalman methods heavily depends on accurate a priori knowledge of noise statistics. Also, it is important to note that non–adaptive UKF performance is in general better than non–adaptive EKF with unknown noise statistics, though neither of the two is satisfactory for meaningful estimation and control purposes. Contrary to non–adaptive Kalman filters, it can be seen from Figs. 3.10–3.13 that AEKF and AUKF have resulted in accurate estimation and also control of the quadcopter. From Table 3.9, it can be seen that AEKF and AUKF have both resulted in subcentimeter mean estimation error in all translational channels and subdegree mean estimation error in all rotational channels. In general, it is observed that AUKF mean estimation is to some extent superior to AEKF, which has resulted in similar trends in overall control performance metrics. In that regard, from Table 3.10, it can be seen that AUKF overall control RMSE metrics are better than AEKF in in X and Y channels by 1.3 cm and 0.6 cm, respectively. In the Z channel, AUKF again slightly outperforms AEKF by 0.1 cm. In rotational channels, while both algorithms have resulted in similar RMSE for ψ channel, AUKF performance is better than AEKF in φ and θ channels by 0.1 degree and 0.3 degree, respectively. In general, it is concluded that AUKF is superior to AEKF in scenarios with unknown noise statistics in both mean estimation error and overall control performance. However, given the superiority of EKF–based algorithm in terms of execution time, the ultimate selection of estimation algorithm is a trade–off between the available computational resources and the estimation and control performance requirements of a given mission.
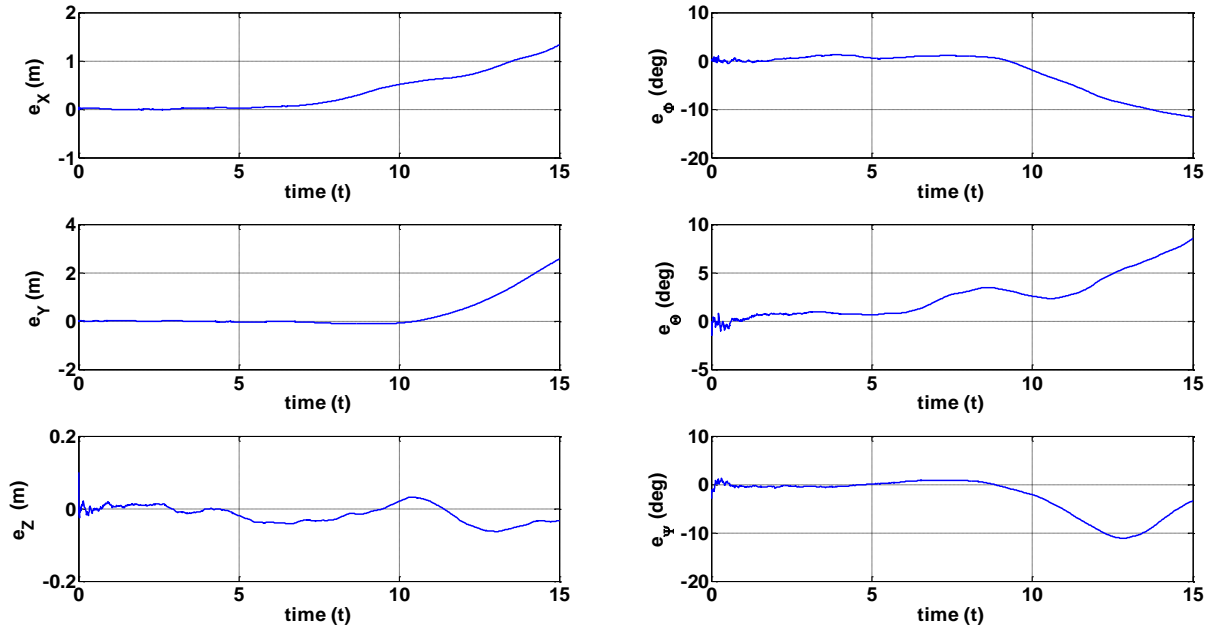
Fig. 3.6 EKF–based estimation error with unknown noise statistics
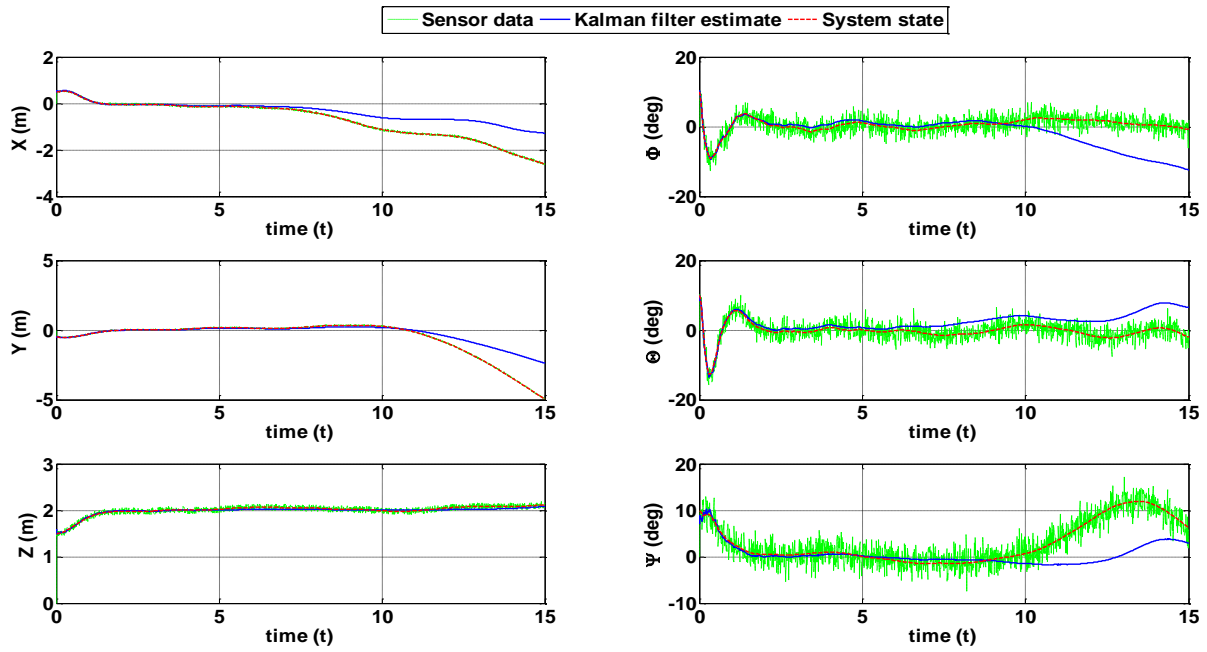
Fig. 3.7 EKF–based LQR control with unknown noise statistics
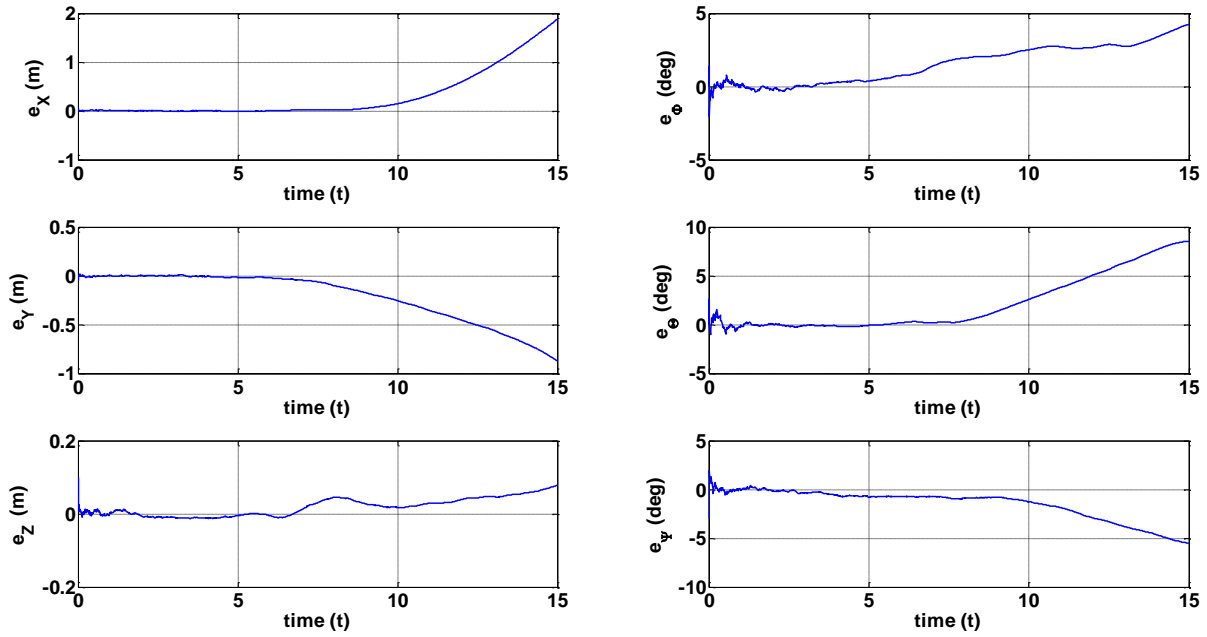
Fig. 3.8 UKF–based estimation error with unknown noise statistics
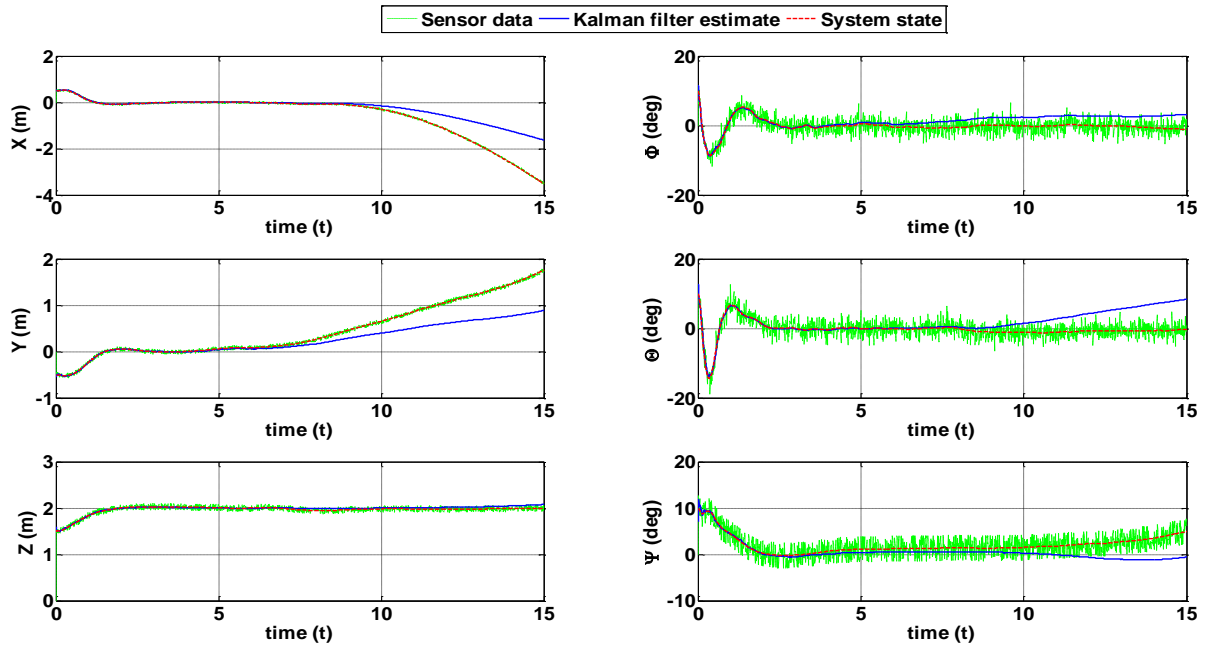
Fig. 3.9 UKF–based LQR control with unknown noise statistics

59
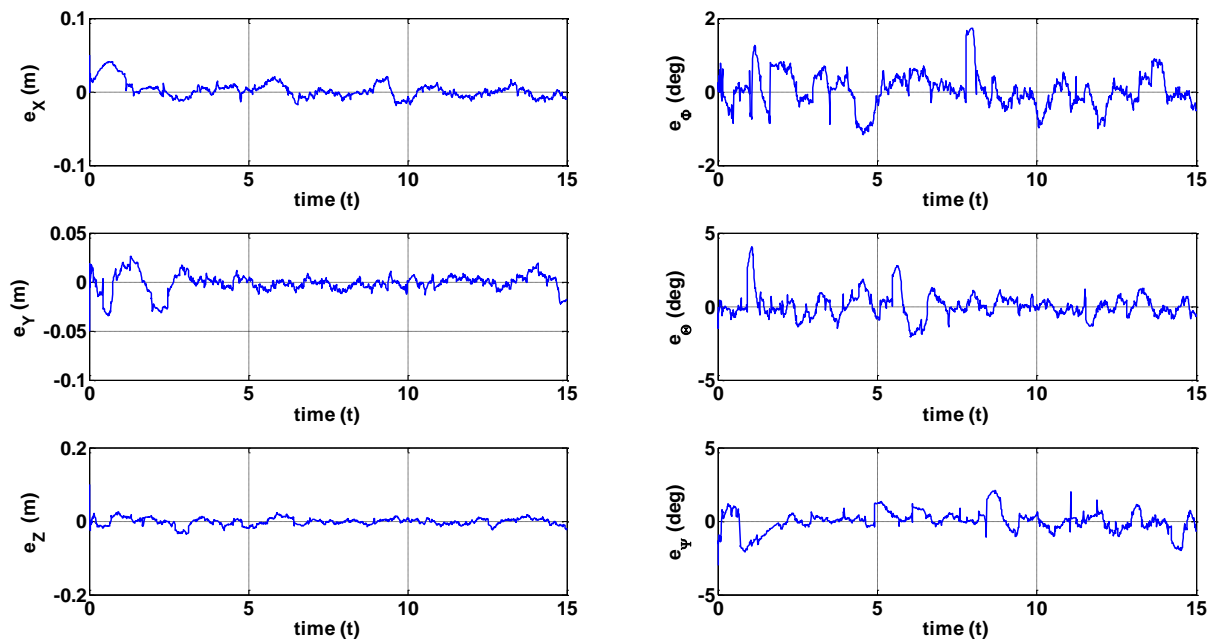
Fig. 3.10 AEKF–based estimation error with unknown noise statistics
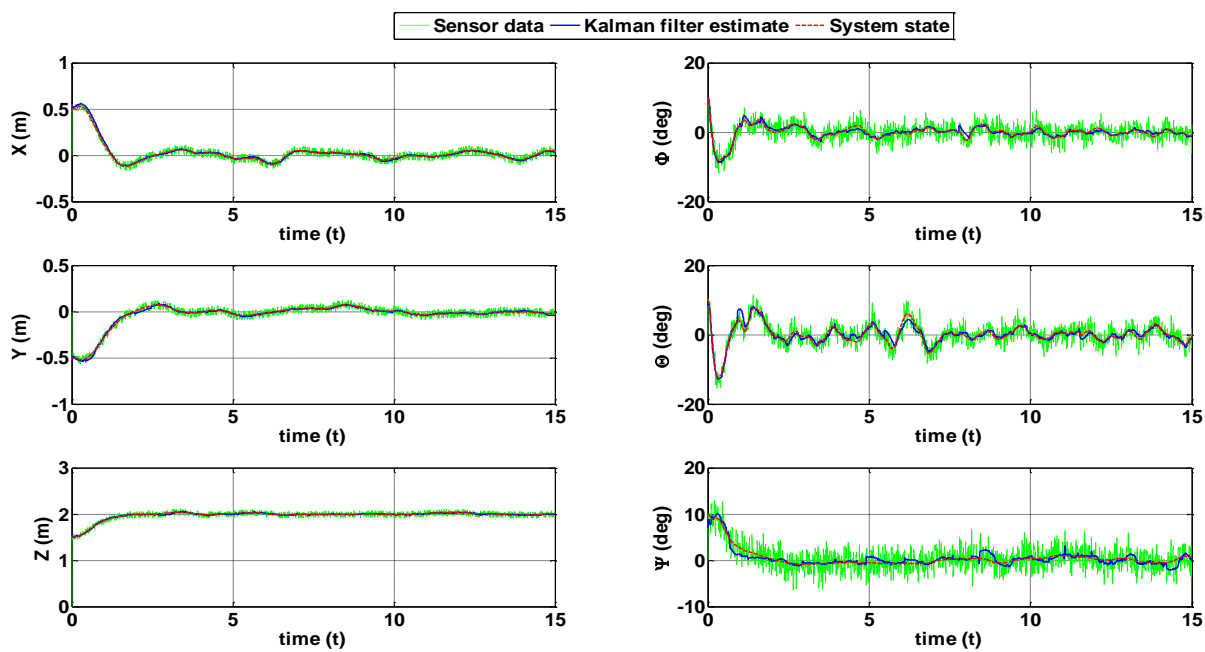
Fig. 3.11 AEKF–based LQR control with unknown noise statistics
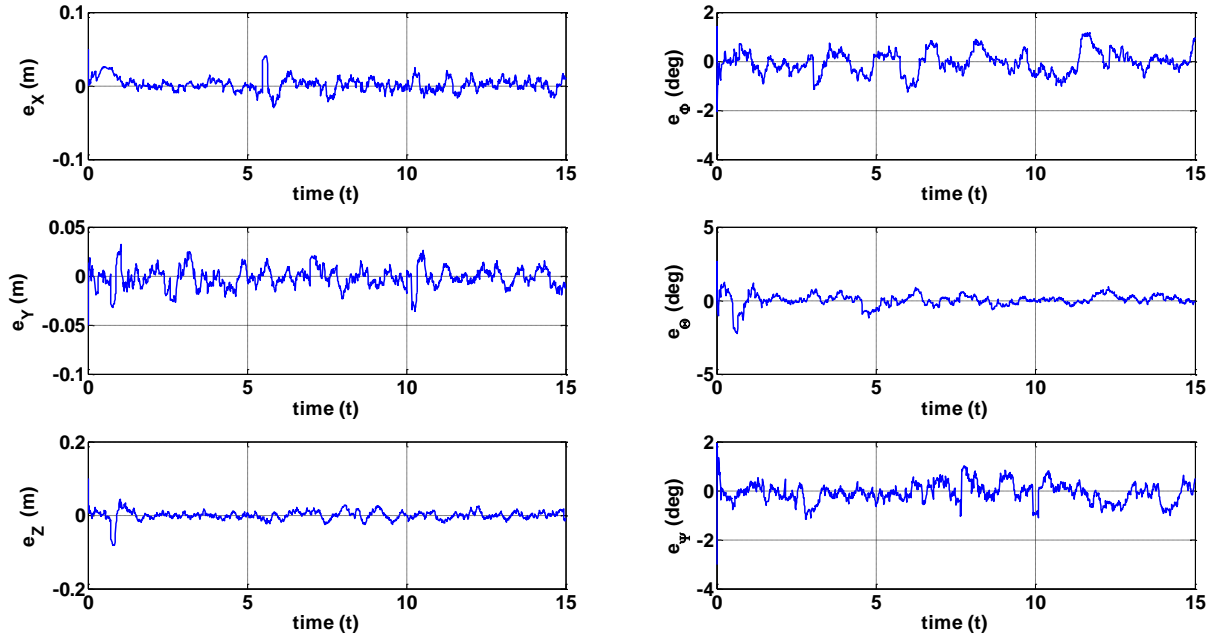
Fig. 3.12 AUKF–based LQR control with unknown noise statistics
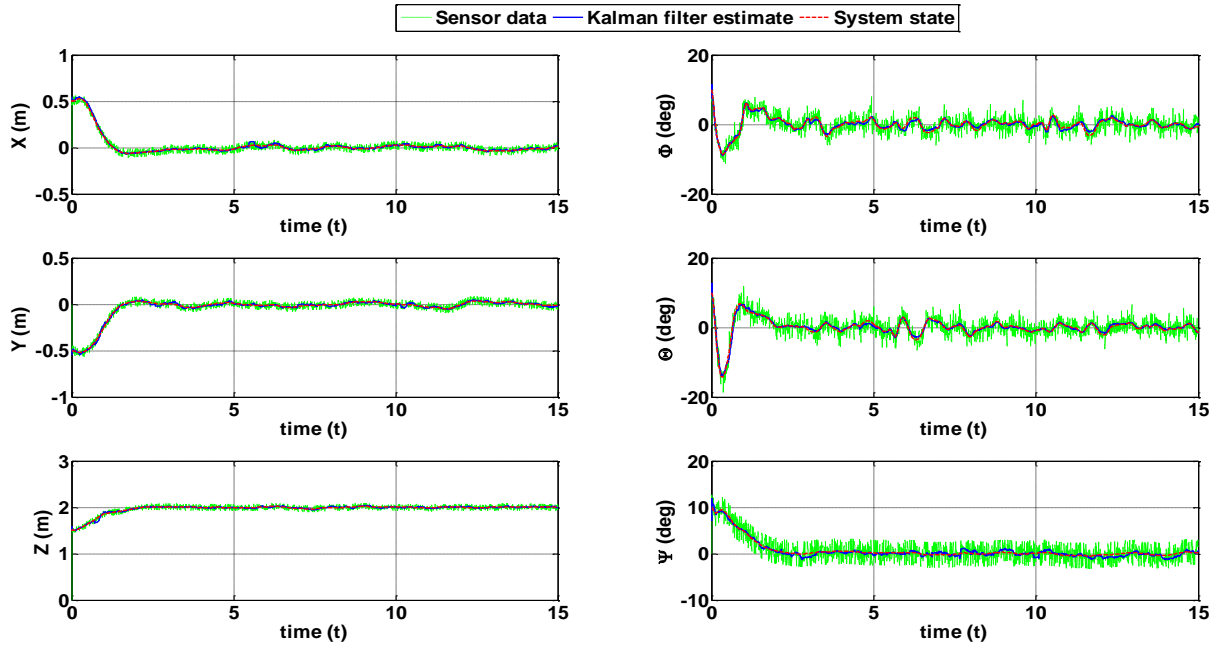
Fig. 3.13 AUKF–based LQR control with unknown noise statistics

Table 3.9 estimation error mean and standard deviation with unknown noise statistics

		e_X (cm)	e_Y (cm)	e_Z (cm)	e_φ (deg)	e_θ (deg)	e_ψ (deg)
EKF	Mean	34.4	34.0	2.5	3.0	2.6	2.7
	STD	38.8	65.2	2.6	4.2	2.7	3.8
UKF	Mean	29.7	20.7	2.3	1.6	2.1	1.5
	STD	39.2	25.1	2.4	1.7	2.7	1.6
AEKF	Mean	0.9	0.7	0.8	0.4	0.4	0.5
	STD	1.3	1.0	1.1	0.4	0.7	0.7
AUKF	Mean	0.6	0.6	0.8	0.3	0.3	0.4
	STD	0.7	0.8	1.2	0.3	0.4	0.6

Table 3.10 RMSE metric with unknown noise statistics

	$RMSE_X$ (cm)	$RMSE_Y$ (cm)	$RMSE_Z$ (cm)	$RMSE_\varphi$ (deg)	$RMSE_\theta$ (deg)	$RMSE_\psi$ (deg)
EKF	53.8	67.9	10.0	4.6	3.8	2.3
UKF	52.9	41.5	9.9	2.5	3.8	2.2
AEKF	12.7	12.9	9.9	1.9	2.7	2.0
AUKF	11.4	12.3	9.8	1.8	2.4	2.0

Additionally, given that the above simulations assume a worst–case scenario where no a priori knowledge of $\boldsymbol{Q}^*$ is available, it is important to study gradual degradation of EKF–based and UKF–based algorithms in scenarios with some degree of confidence in $\boldsymbol{Q}^*$. For that purpose, here we develop 4 more scenarios where in the first scenario the best estimate of $\boldsymbol{Q}^*$, i.e. $\boldsymbol{Q}^\dagger$, is an order of magnitude smaller than the actual value and in each subsequent scenario the mismatch increases by an order of magnitude such that the best estimate of $\boldsymbol{Q}^*$ in the 4th scenario is 4 orders of magnitude smaller than the actual value. For the described case studies, simulations were carried out and the obtained results are shown in Table 3.11 and 3.12.

Table 3.11 EKF/UKF estimation error mean and standard deviation in $Q^\dagger$ mismatch scenarios

			e_X (cm)	e_Y (cm)	e_Z (cm)	e_φ (deg)	e_θ (deg)	e_ψ (deg)
Scenario # 1	EKF	Mean	0.7	0.6	0.8	0.3	0.3	0.3
		STD	0.9	0.8	1.0	0.3	0.3	0.4
	UKF	Mean	0.5	0.6	0.6	0.3	0.2	0.3
		STD	0.6	0.8	0.8	0.3	0.3	0.3
Scenario # 2	EKF	Mean	0.7	0.8	1.0	0.5	0.4	0.4
		STD	1.0	1.0	1.2	0.6	0.5	0.5
	UKF	Mean	0.6	0.6	0.7	0.4	0.3	0.3
		STD	0.8	0.7	0.9	0.5	0.5	0.4
Scenario # 3	EKF	Mean	1.6	1.9	1.4	0.8	0.8	0.6
		STD	1.9	2.3	1.8	1.0	0.9	0.8
	UKF	Mean	1.1	0.8	0.8	0.4	0.4	0.3
		STD	1.1	0.8	1.0	0.4	0.5	0.6
Scenario # 4	EKF	Mean	3.8	5.6	2.0	1.2	1.1	0.8
		STD	4.0	7.3	2.1	1.6	1.1	0.9
	UKF	Mean	3.0	2.0	1.0	0.5	0.7	0.7
		STD	3.2	2.1	1.3	0.5	0.8	0.9

Table 3.12 EKF/UKF RMSE metric in $Q^\dagger$ mismatch scenarios

		$RMSE_X$ (cm)	$RMSE_Y$ (cm)	$RMSE_Z$ (cm)	$RMSE_\varphi$ (deg)	$RMSE_\theta$ (deg)	$RMSE_\psi$ (deg)
Scenario # 1	EKF	11.8	13.0	9.8	1.8	2.3	2.0
	UKF	11.2	12.1	9.7	1.8	2.3	2.0
Scenario # 2	EKF	12.2	13.8	9.9	1.9	2.4	2.0
	UKF	11.5	12.3	9.8	1.8	2.3	2.0
Scenario # 3	EKF	15.1	18.1	9.9	2.1	2.5	2.1
	UKF	12.7	12.9	9.8	1.9	2.3	2.1
Scenario # 4	EKF	18.5	30.0	9.9	2.7	2.7	2.2
	UKF	17.4	15.1	9.9	2.0	2.5	2.2

It can be seen from the above tables that incremental increase of mismatch in initial estimation of process noise leads to meaningful degradation of EKF and UKF algorithm estimation accuracy and hence overall control performance. More specifically, for the first mismatch scenario where the best estimate of Q^* is on order of magnitude smaller than the actual value, the estimation accuracy of EKF–based algorithm worsens by 0.1 cm in X and Y channels and by 0.2 cm in Z channel, compared to known noise statistics scenario. Similarly, for θ and ψ channels, the estimation accuracy decreases by 0.1 degree. The UKF–based algorithm also degrades in this scenario as its estimation accuracy decreases by 0.1 cm in Y and Z channels. In the rotational $, \varphi$ and ψ channels, the estimation accuracy of non–adaptive UKF algorithm also decreases by 0.1 degree. As the mismatch of best estimate of Q^* increases to two orders of magnitude in the second scenario, the estimation accuracy of both EKF and UKF algorithms further decrease in almost all rotational and translational channels. This, in turn, leads to higher overall control RMSE metrics as less accurate state estimates are used in the feedback–based LQR control. As the mismatch further increases to three orders of magnitude, the estimation accuracy in translational and rotational channels decreases to more than 1 cm and close to 1 degree, respectively. This is also accompanied by noticeable increase in RMSE metrics of both EKF and UKF algorithms which make the performance questionable. Further increase of the mismatch in best estimate of Q^* worsens estimation results of EKF and UKF algorithms, accompanied by increasing RMSE metric. While in general, UKF outperforms EKF in the mentioned scenarios, both algorithms fail to maintain meaningful estimation and control of the quadcopter in hovering flight as the mismatch in best estimate of Q^* increases to more than two orders of magnitude. Therefore, it is concluded that for small mismatch in noise statistics, conventional filters degrade to some extent but maintain reasonable estimation and control performance. Yet, as the mismatch increases, the performance of conventional non–adaptive EKF and UKF algorithms degrade considerably which results in inaccurate estimation and control and this eventually leads total loss of the vehicle control.

What is more, it is also important to examine how the adaptive algorithms perform in case of increasing noise statistics mismatch. In general, it is known that the performance of adaptive Kalman filter algorithms degrade as simultaneous mismatch in process and measurement noise statistics increase and, in severe cases, the algorithm convergence may be jeopardized [155],

[189]. For example, for a linear system case study, it was shown that an adaptive Kalman filter degraded considerably for cases with simultaneous two orders of magnitude mismatch in process and measurement noise statistics [189]. Here, in order to show the sensitivity of AEKF and AUKF to various covariance mismatch levels, a number of scenarios are defined. In the first scenario, the mismatch in measurement noise statistics is assumed to be an order of magnitude smaller than the actual value, i.e. $R^\dagger = 0.1R^*$ and process covariance mismatch is increased by order of magnitude steps until the adaptation algorithm fails. Then, the mismatch in measurement noise statistics is increased to two orders of magnitude and various processes noise mismatches are examined until the adaption fails. Following the same procedure, mismatch in measurement noise is further increased and various incremental mismatches in process noise are examined to obtain mismatch scenarios where AEKF and AUKF fail to compensate for simultaneous mismatch in noise statistics. For the described case studies, simulations were carried out and obtained results are presented in Tables 3.13–3.18.

Table 3.13 AEKF/AUKF estimation error mean and standard deviation with $\boldsymbol{R}^{\dagger} = 0.1\boldsymbol{R}^{*}$

			e_X (cm)	e_Y (cm)	e_Z (cm)	e_φ (deg)	e_θ (deg)	e_ψ (deg)
$\boldsymbol{Q}^{\dagger} = 0.1\boldsymbol{Q}^{*}$	EKF	Mean	0.7	0.6	0.7	0.3	0.3	0.3
		STD	0.8	0.8	0.8	0.3	0.3	0.4
	UKF	Mean	0.5	0.6	0.7	0.2	0.3	0.3
		STD	0.7	0.7	0.7	0.2	0.3	0.3
$\boldsymbol{Q}^{\dagger} = 0.01\boldsymbol{Q}^{*}$	EKF	Mean	0.7	0.6	0.7	0.4	0.4	0.3
		STD	0.8	0.7	0.8	0.4	0.4	0.4
	UKF	Mean	0.6	0.6	0.7	0.3	0.3	0.3
		STD	0.7	0.7	0.8	0.3	0.3	0.3
$\boldsymbol{Q}^{\dagger} = 0.001\boldsymbol{Q}^{*}$	EKF	Mean	0.8	0.6	0.7	0.4	0.4	0.3
		STD	0.9	0.8	0.8	0.4	0.5	0.5
	UKF	Mean	0.6	0.6	0.7	0.3	0.3	0.3
		STD	0.7	0.8	0.8	0.3	0.3	0.4
$\boldsymbol{Q}^{\dagger} = 0.0001\boldsymbol{Q}^{*}$	EKF	Mean	0.9	0.6	0.8	0.4	0.4	0.5
		STD	1.0	0.8	1.0	0.4	0.7	0.5
	UKF	Mean	0.6	0.6	0.8	0.3	0.3	0.4
		STD	0.7	0.8	0.9	0.3	0.4	0.6
$\boldsymbol{Q}^{\dagger} = 0$	EKF	Mean	0.9	0.7	0.8	0.4	0.4	0.5
		STD	1.3	1.0	1.1	0.4	0.7	0.7
	UKF	Mean	0.6	0.6	0.8	0.3	0.3	0.4
		STD	0.7	0.8	1.2	0.3	0.4	0.6

Table 3.14 AEKF/AUKF RMSE metric with $\boldsymbol{R}^\dagger = 0.1\boldsymbol{R}^*$

		RMSE_X (cm)	RMSE_Y (cm)	RMSE_Z (cm)	RMSE_φ (deg)	RMSE_θ (deg)	RMSE_ψ (deg)
$\boldsymbol{Q}^\dagger = 0.1\boldsymbol{Q}^*$	EKF	11.8	12.2	9.7	1.7	2.3	2.0
	UKF	11.3	12.2	9.7	1.7	2.2	2.0
$\boldsymbol{Q}^\dagger = 0.01\boldsymbol{Q}^*$	EKF	11.8	12.5	9.7	1.8	2.3	2.0
	UKF	11.3	12.3	9.7	1.7	2.2	2.0
$\boldsymbol{Q}^\dagger = 0.001\boldsymbol{Q}^*$	EKF	12.3	12.8	9.7	1.8	2.5	2.1
	UKF	11.4	12.3	9.7	1.8	2.3	2.0
$\boldsymbol{Q}^\dagger = 0.0001\boldsymbol{Q}^*$	EKF	12.5	12.8	9.8	1.8	2.4	2.0
	UKF	11.4	12.3	9.8	1.8	2.4	2.0
$\boldsymbol{Q}^\dagger = 0$	EKF	12.7	12.9	9.9	1.9	2.7	2.0
	UKF	11.4	12.3	9.8	1.8	2.4	2.0

In general, it can also be seen that the adaptive algorithms perform well for scenarios with small mismatch in measurement statistics, even with large mismatch in process noise statistics. From the obtained results, it can be observed that AUKF outperforms AEKF in most scenarios in most translational and rotational channels. In particular, for small mismatch in process noise, the adaptive algorithms results in estimation accuracy and overall control performance comparable to scenarios with known noise statistics. Even for scenarios with large mismatch in process noise, i.e. $\boldsymbol{Q}^\dagger = 0.0001\boldsymbol{Q}^*$ or in worst–case scenario of $\boldsymbol{Q}^\dagger = 0$, the adaptive algorithms results in subcentimeter and subdegree accuracy in translational and rotational channels, respectively. Also, similar trend can be observed in control RMSE metrics where adaptive algorithms have resulted in successful achievement of hovering flight. This, in turn, is due to the fact that measurements with relatively low mismatch level serve as reliable input for adaptation algorithms to compensate for even large mismatch in process noise statistics. Yet, estimation accuracy and overall control performance degrade more noticeably in scenarios with larger mismatch in measurement noise statistics, as shown in Table 3.15 and Table 3.16.

Table 3.15 AEKF/AUKF estimation error mean and standard deviation with $R^{\dagger} = 0.01R^*$

			e_X (cm)	e_Y (cm)	e_Z (cm)	e_φ (deg)	e_θ (deg)	e_ψ (deg)
$Q^{\dagger} = 0.1Q^*$	EKF	Mean	0.8	0.8	0.8	0.5	0.5	0.6
		STD	1.0	0.9	1.0	0.6	0.6	0.6
	UKF	Mean	0.6	0.8	0.8	0.5	0.5	0.5
		STD	0.8	1.0	1.0	0.6	0.6	0.6
$Q^{\dagger} = 0.01Q^*$	EKF	Mean	0.8	0.9	1.2	0.7	0.7	0.7
		STD	1.1	1.1	1.2	0.9	0.9	0.8
	UKF	Mean	0.8	0.8	1.1	0.7	0.6	0.7
		STD	1.0	1.0	1.2	0.9	0.8	0.8
$Q^{\dagger} = 0.001Q^*$	EKF	Mean	11.9	15.3	17.5	0.7	0.7	0.7
		STD	15.0	19.1	22.4	1.0	0.9	0.9
	UKF	Mean	10.2	12.6	14.4	0.7	0.7	0.8
		STD	13.6	16.1	18.7	0.9	0.9	1.0
$Q^{\dagger} = 0.0001Q^*$	N/A							

Table 3.16 AEKF/AUKF RMSE metric with $R^{\dagger} = 0.01R^*$

		$RMSE_X$ (cm)	$RMSE_Y$ (cm)	$RMSE_Z$ (cm)	$RMSE_\varphi$ (deg)	$RMSE_\theta$ (deg)	$RMSE_\psi$ (deg)
$Q^{\dagger} = 0.1Q^*$	EKF	12.0	13.5	10.2	2.0	2.5	2.0
	UKF	11.6	12.4	10.0	2.1	2.5	2.0
$Q^{\dagger} = 0.01Q^*$	EKF	12.8	13.8	10.4	2.4	2.7	2.4
	UKF	11.6	12.5	10.3	2.2	2.7	2.4
$Q^{\dagger} = 0.001Q^*$	EKF	13.6	22.6	14.8	4.5	2.8	2.6
	UKF	11.8	16.2	12.0	3.1	2.8	2.7
$Q^{\dagger} = 0.0001Q^*$	N/A						

From Table 3.15 and Table 3.16, it is observed that estimation errors in both translational and rotational channels are close to 1 cm and 1 degree even for small mismatch in process noise statistics, e.g. $Q^{\dagger} = 0.1Q^*$ and $Q^{\dagger} = 0.01Q^*$. Also, relatively large RSME metrics for such

scenarios indicate that both AEKF and AUKF have degraded considerably, but are still able to accomplish quadcopter hovering task with simultaneous two orders of magnitude mismatch in process and measurement noise statistics. Further increasing the mismatch in process noise statistics, i.e. $\boldsymbol{Q}^{\dagger} = 0.001\boldsymbol{Q}^{*}$, results in excessively large estimation errors for both AEKF and AUKF where very large RMSE metrics indicate that meaningful accomplishment of hovering mission has not been maintained. If one increases process noise mismatch to $\boldsymbol{Q}^{\dagger} = 0.0001\boldsymbol{Q}^{*}$ and larger, both AEKF and AUKF drastically fail. Similar to our results, in the relevant literature, it has been reported that simultaneous adaptation of process and measurement noise statistics with large mismatch is prone to poor performance as neither of the two provide reliable information for adaptation algorithms [155], [189]. Next, one may further increase the measurement noise statistics mismatch to three orders of magnitude, i.e. $\boldsymbol{R}^{\dagger} = 0.001\boldsymbol{R}^{*}$ to examine the performance of adaptation algorithms. The obtained results for such scenario are given in Table 3.17 and 3.18.

Table 3.17 AEKF/AUKF estimation error mean and standard deviation with $\boldsymbol{R}^{\dagger} = 0.001\boldsymbol{R}^{*}$

			e_X (cm)	e_Y (cm)	e_Z (cm)	e_φ (deg)	e_θ (deg)	e_ψ (deg)
$\boldsymbol{Q}^{\dagger} = 0.1\boldsymbol{Q}^{*}$	EKF	Mean	0.8	0.8	0.8	0.5	0.5	0.6
		STD	1.0	0.9	1.0	0.6	0.6	0.6
	UKF	Mean	0.6	0.8	0.8	0.5	0.5	0.5
		STD	0.8	1.0	1.0	0.6	0.6	0.6
$\boldsymbol{Q}^{\dagger} = 0.01\boldsymbol{Q}^{*}$	N/A							

Table 3.18 AEKF/AUKF RMSE metric with $\boldsymbol{R}^{\dagger} = 0.001\boldsymbol{R}^{*}$

		$RMSE_X$ (cm)	$RMSE_Y$ (cm)	$RMSE_Z$ (cm)	$RMSE_\varphi$ (deg)	$RMSE_\theta$ (deg)	$RMSE_\psi$ (deg)
$\boldsymbol{Q}^{\dagger} = 0.1\boldsymbol{Q}^{*}$	EKF	12.0	13.5	10.2	2.0	2.5	2.0
	UKF	11.6	12.4	10.0	2.1	2.5	2.0
$\boldsymbol{Q}^{\dagger} = 0.01\boldsymbol{Q}^{*}$	N/A						

In this extreme case with $R^\dagger = 0.001R^*$, for $Q^\dagger = 0.1Q^*$, it is observed that both AEKF and AUKF have successfully accomplished hovering mission of the quadcopter. More specifically, both algorithms have resulted in subcentimeter and subdegree estimation accuracy in translational and rotational channels. In general, it can be observed that AUKF outperforms AEKF in most channels both in terms of estimation accuracy and overall control performance. However, if one increases the mismatch in process noise statistics to two orders of magnitude, i.e. $Q^\dagger = 0.01Q^*$, both algorithms fail abruptly. This is due the fact that measurement information in this scenario are extremely unreliable ($R^\dagger = 0.001R^*$) and therefore the adaptive algorithms fail to compensate for two (or more) orders of magnitude mismatch in process noise. It is concluded that if large mismatch in process noise statistics are expected, it is important to assure relatively precise knowledge of measurement noise statistics. Based on our results, if simultaneous large mismatch in both process and measurement noise statistics are present, performance of adaptive algorithms is questionable and this, in extreme cases, results in total loss of vehicle control.

Finally, the effect of adaptation window size on the performance of AEKF and AUKF algorithms is studied. For that purpose, we considered the average mean estimation error in translational X, Y, Z channels and rotational φ, θ, ψ channels, denoted by e_{XYZ} and $e_{\varphi\theta\psi}$, respectively. Based on these average mean estimation metrics, adaption window size of $N = 15$ was found to provide satisfactory results in our simulations. In order to provide insight into the effects of adaptation window size on those metrics in AEKF and AUKF, N is varied from 5 to 50 with a step of 5 and simulation results are presented in Table 3.19.

Table 3.19 Average mean estimation error with different adaptation window size

N	Adaptive filter	e_{XYZ} (cm)	$e_{\varphi\theta\psi}$ (deg)
5	AEKF	1.1	0.7
	AUKF	0.9	0.6
10	AEKF	0.9	0.5
	AUKF	0.8	0.5
15	AEKF	0.8	0.4
	AUKF	0.7	0.3
20	AEKF	0.8	0.5
	AUKF	0.7	0.4
25	AEKF	0.9	0.6
	AUKF	0.8	0.4
30	AEKF	0.9	0.5
	AUKF	0.8	0.5
35	AEKF	0.9	0.6
	AUKF	0.7	0.4
40	AEKF	0.9	0.5
	AUKF	0.8	0.5
45	AEKF	0.8	0.6
	AUKF	0.8	0.5
50	AEKF	0.9	0.6
	AUKF	0.8	0.5

In general, it has been reported in pervious works that adaptation window size does not significantly affect the performance of adaptive Kalman filters [142], [159]. Our simulation results also show that both Kalman–based algorithms perform well for a wide range of the adaptation window size, without excessive variation in mean estimation errors and therefore the algorithm can be used robustly regarding the selection of N. To be more specific, from Table 3.19, it can be seen that both AEKF and AUKF result in relatively larger estimation errors for small window size of $N = 5$. This is expected as very small window size is not representative of

noise statistics and therefore the covariance matching technique cannot properly compensate for the difference between the estimated and actual noise statistics. As the window size increases to $N = 10$, both error metrics e_{XYZ} and $e_{\varphi\theta\psi}$ slightly improve in AEKF and AUKF and similar improvement continues as one increases the window size to $N = 15$. As one further increases the adaptation window size to $N = 20$, no further improvement in e_{XYZ} and $e_{\varphi\theta\psi}$ is observed. From Eq. 3.35 and Eq. 3.41, it is readily seen that larger adaptation window size is associated with higher computational time and therefore $N = 15$ was selected for the unknown noise statistics scenario in this subsection. As one further increases the adaptation window size, the two metrics begin to slightly worsen as the system is less flexible to incorporate fast dynamic of the quadcopter. In general, it is concluded that very small adaptation window size results in less accurate estimations, whereas very large adaptation window size results in reduced flexibility to system dynamics.

3.6 Summary

In this chapter, extended and unscented Kalman filters along with their adaptive counterparts for estimation and control of a quadcopter were formulated. Based on covariance matching approach, adaptive filters taking account of uncertainties in both process and measurement noise statistics with a new fading memory formulation were presented. Simulation results show that for known noise statistics, UKF slightly outperforms EKF both in terms of estimation and overall control performance, however with approximately an order–of–magnitude higher execution time. Simulation results also show that both algorithms successfully achieve hovering flight of the quadcopter starting from a wide range of initial conditions. For unknown noise statistics, it was shown that both EKF and UKF result in very poor estimation and control performance, eventually leading to total loss of quadcopter control. In such cases, AEKF and AUKF resulted in satisfactory estimation and overall control of the quadcopter, even with uncertainty in both process and measurement noise statistics. For relatively accurate measurement noise statistics, it was shown that adaptive algorithms successfully compensated for very large mismatch in process noise statistics. For increasing mismatch in measurement noise statistics, performance of adaptive filters degraded in scenarios with increasing uncertainties in process noise statistics to an extent where eventually loss of hovering mission was encountered.

Chapter 4 Unscented Kalman Filter State Estimation for Manipulating Unmanned Aerial Vehicles

This chapter is based on the following published paper:

H. Bonyan Khamseh and F. Janabi–Sharifi, "UKF–Based LQR Control of a Manipulating Unmanned Aerial Vehicle", *Unmanned Systems*, vol. 5, no. 3, pp. 131–139, 2017.

and the following submitted paper:

H. Bonyan Khamseh and F. Janabi–Sharifi, "Unscented Kalman Filter State Estimation for Manipulating Unmanned Aerial Vehicles", Submitted to *Journal of Aerospace Science and Technology.*

4.1 Introduction

Traditionally, Unmanned Aerial Vehicles (UAVs) have served applications such as forest fire monitoring [3], search and rescue [4], transmission line inspection [6], and border monitoring [5], to name a few. An important limitation of the above applications is that the UAV tends to fly and obtain remote–sensing data from its surrounding, but physical interaction with the environment is strictly avoided. However, in recent years, there has been increasing interest in applications where the UAV is required to perform perching, grasping, and manipulation [10], [11], [15], [190]. In order to enable such applications, a new area of research, i.e. aerial manipulation, studies varying configuration of UAVs and their physical interaction with the surrounding environment.

Accurate and efficient state estimation of an aerial manipulation system, viz. Manipulating Unmanned Aerial Vehicle (MUAV) hereafter, is an important element to enable autonomous aerial manipulation missions. This is a fundamental challenge that has been left largely intact to this date and is the main theme of the present work. Indeed, current MUAV systems either obtain precise state estimates from costly motion capture systems such as NaturalPoint OptiTrack™ or VICON™ systems [10], [12], [17], [20], [23] or employ redundant sensory data from GPS, Inertial Measurement Units (IMUs), barometers, and/or onboard cameras [23], [25]–[27]. This is further complicated as the most common state estimation algorithm for UAV applications, namely Extended Kalman Filter (EKF) [106]–[108], cannot be used for MUAVs with complex, coupled and highly nonlinear dynamics. In such cases, Unscented Kalman Filter (UKF) has been shown to

73

be a feasible solution as it effectively deals with nonlinearities and dynamic couplings [63], [111], [142], [154]. For instance, performance of EKF– and UKF–based state estimation algorithms for spacecraft localization problem was discussed in [154]. For earth–moon transfer and geostationary orbit rising scenarios, it was shown that UKF outperforms EKF, in terms of localization accuracy and estimation consistency. In another work [63], 3 Degrees–of–Freedom (DoF) orientation estimation of a fixed–wing UAV was investigated. The results obtained in that work showed that UKF outperforms EKF as it is more robust to noise and bias in process and measurement models. However, in general, a certain disadvantage of UKF–based state estimation is that its improved performance usually comes at the cost of high computational complexity, even for system with simple dynamics e.g. [111]. In order to remedy that shortcoming, in this chapter, Spherical unscented transform is formulated to improve the computational complexity of General UKF state estimation in MUAV applications. Also, a scaling method with improved distribution of sigma points will be integrated to improve the state estimation accuracy, especially for attitude estimation applications.

In order to assess the feasibility of the proposed state estimation schemes, dynamic modelling and control of a MUAV are first addressed. Here, Euler–Lagrange approach is adopted to obtain coupled nonlinear dynamic model of a MUAV because, in general, this approach is suitable for controller design and analytical investigation. From the control perspective, we adopt a Linear Quadratic Regulator (LQR) to simultaneously control a quadcopter UAV and its 2–DoF manipulator. LQR is a popular control scheme where the main advantages are that it provides optimal performance for given weight matrices, along with its easy implementation [35], [191], [192]. LQR has also been previously used in aerial manipulation applications [35], [109]. In [35], control of a small helicopter and its 1–DoF revolute–joint robotic arm was achieved using LQR scheme. In that work, it was discussed that LQR offers stability and acceptable performance of the MUAV near its trim point. Here, similar to [109], we extend LQR controller augmented with integral action such that simultaneous control a quadcopter and its 2–DoF manipulator is achieved.

The main contributions of this chapter are as follows. To the best of authors' knowledge, our proposed algorithm in [109] is the first work where an onboard UKF–based state estimation

74

algorithm for a MUAV with robotic arm was presented. The main advantages of the proposed algorithms in this work, common with those in [109], are summarized in the following.

- Avoiding the use of costly motion capture systems and not being confined to indoor environment;

- Degrading gracefully in case of increasing noise levels and/or loss of sensory data, preventing abrupt loss of MUAV control.

This, in turn, will enable autonomous operation of MUAVs for various real–life applications with solely onboard sensory information. Another novel aspect of the current work is that, in view of limited computational resources onboard MUAVs, a computationally–efficient UKF algorithm is formulated for MUAV state estimation for the first time. Also, in scenarios with increasing noise level and total loss of sensory data, sensitivity of UKF–based algorithms for MUAV state estimation is studied for the first time.

The rest of this chapter is organized as follows. Using Euler–Lagrange formulation, the coupled nonlinear dynamic model of a MUAV is presented in Section 4.2. LQR control design for a MUAV is presented in Section 4.3. In Section 4.4, two variants of UKF–based state estimation algorithm, based on General and Spherical unscented transforms, are discussed. Finally, simulation results and summaries are presented in Sections 4.5 and 4.6, respectively.

4.2 MUAV dynamic modelling

In the following, kinematic equations of a generic MUAV, schematically shown in Fig. 4.1, are presented.
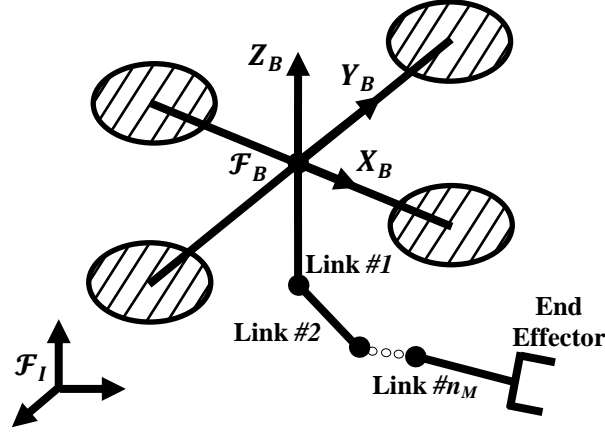
Fig. 4.1 Schematic view of a MUAV

In Fig. 4.1, $\mathcal{F}_I = \{X_I, Y_I, Z_I\}$ with the origin O_I and $\mathcal{F}_B = \{X_B, Y_B, Z_B\}$ with the origin O_B denote the inertial frame and body frame, respectively. The number of manipulator DoF is denoted by n_M. Also, position of the body frame $\mathcal{F}_B$ in the inertial frame is given by $\boldsymbol{p}_B = [X \quad Y \quad Z]^T$. Finally, $\boldsymbol{\Phi}_B = [\varphi \quad \theta \quad \psi]^T$ represents the UAV attitude, where φ, θ and ψ are the roll, pitch and yaw angles. Consequently:

$$\begin{cases} \dot{X} = v_X, \\ \dot{Y} = v_Y, \\ \dot{Z} = v_Z, \end{cases} \tag{4.1}$$

$$\boldsymbol{\omega}_B^I = \boldsymbol{T}\dot{\boldsymbol{\Phi}}_B \tag{4.2}$$

Here, $\boldsymbol{\omega}_B^I$ is the angular velocity of $\mathcal{F}_B$ with respect to $\mathcal{F}_I$. In body frame, (2) can be written:

$$\boldsymbol{\omega}_B = \boldsymbol{R}_B^T \boldsymbol{\omega}_B^I = \boldsymbol{R}_B^T \boldsymbol{T}\dot{\boldsymbol{\Phi}}_b = \boldsymbol{Q}\dot{\boldsymbol{\Phi}}_b. \tag{4.3}$$

where $\boldsymbol{R}_B$ is the rotation matrix from the body frame to the inertial frame. In order to obtain the kinematic equations of the end–effector, it is helpful to consider a frame at the center of mass of

76

each link, where the frame coordinates coincide with link principal axes of inertia. Denoting the position of the frame on link i by $\boldsymbol{p}_{l_i}$, one can write:

$$\boldsymbol{p}_{l_i} = \boldsymbol{p}_B + \boldsymbol{R}_B \boldsymbol{p}_{Bl_i}, \tag{4.4}$$

where $\boldsymbol{p}_{Bl_i}$ denotes the position vector of the frame on link i with respect to $\mathcal{F}_B$, expressed in $\mathcal{F}_B$. $\dot{\boldsymbol{p}}_{Bl_i}$ is given by [100]:

$$\dot{\boldsymbol{p}}_{Bl_i} = \boldsymbol{J}_{P1}^{(l_i)} \dot{q}_1 + \cdots + \boldsymbol{J}_{Pi}^{(l_i)} \dot{q}_i = \boldsymbol{J}_P^{(l_i)} \dot{\boldsymbol{q}}, \tag{4.5}$$

where $\boldsymbol{J}_P^{(l_i)}$ represents contribution of the manipulator Jacobian to linear velocity up to link i. Also, the robotic manipulator joint variables are denoted by $\boldsymbol{q} = [q_1 \quad \cdots \quad q_{n_M}]^T$. What is more, the angular velocity of the frame on link i with respect to the body frame is obtained as:

$$\boldsymbol{\omega}_{Bl_i} = \boldsymbol{J}_{O1}^{(l_i)} \dot{q}_1 + \cdots + \boldsymbol{J}_{Oi}^{(l_i)} \dot{q}_i = \boldsymbol{J}_O^{(l_i)} \dot{\boldsymbol{q}}. \tag{4.6}$$

Similarly, $\boldsymbol{J}_O^{(l_i)}$ represents contribution of the manipulator Jacobian to angular velocity up to link i. Therefore, one can write:

$$\dot{\boldsymbol{p}}_{l_i} = \dot{\boldsymbol{p}}_B - S(\boldsymbol{R}_B \boldsymbol{p}_{Bl_i}) \boldsymbol{\omega}_B^I + \boldsymbol{R}_B \boldsymbol{J}_P^{(l_i)} \dot{\boldsymbol{q}}, \tag{4.7}$$

where $S(\cdot)$ is the skew–symmetric matrix operator. Similarly, the kinematic angular velocity equation is obtained as:

$$\boldsymbol{\omega}_{l_i} = \boldsymbol{\omega}_B^I + \boldsymbol{R}_B \boldsymbol{J}_O^{(l_i)} \dot{\boldsymbol{q}}. \tag{4.8}$$

Regarding dynamic modeling of MUAVs, there are two main methods in the current literature, namely Newton–Euler and Euler–Lagrange approach [101], [104]. While both methods result in valid equations of motion, the latter is more suitable for controller design and analytical investigations and therefore is adopted here. In the Euler–Lagrange approach, the UAV and its manipulator are treated as a unified system. To that end, the system Lagrangian is $\mathcal{L} = \mathcal{K} - \mathcal{U}$, where $\mathcal{K}$ and $\mathcal{U}$ represent the system kinetic and potential energy, respectively. Consequently:

$$\frac{d}{dt}\frac{\partial \mathcal{L}}{\partial \dot{\xi}_i} - \frac{\partial \mathcal{L}}{\partial \xi_i} = u_{MUAV,i}, \tag{4.9}$$

where ξ_i is the $i-th$ element of the generalized vector $\boldsymbol{\xi} = [\boldsymbol{p}_B^T \quad \boldsymbol{\Phi}_B^T \quad \boldsymbol{q}^T]^T \in \mathbb{R}^{n_\xi \times 1}$, $n_\xi = 6 + n_M$, and $u_{MUAV,i}$ is the $i-th$ input. The total kinetic energy of the MUAV is given by:

$$\mathcal{K} = \mathcal{K}_B + \sum_{i=1}^{n_M} \mathcal{K}_{l_i}, \tag{4.10}$$

where $\mathcal{K}_B$ and $\mathcal{K}_{l_i}$ denote the kinetic energy of the quadcopter and link i, respectively. Kinetic energy of the quadcopter, in turn, is given by:

$$\mathcal{K}_b = \frac{1}{2} m_B \dot{\boldsymbol{p}}_B^T \dot{\boldsymbol{p}}_B + \frac{1}{2} \dot{\boldsymbol{\Phi}}_B^T \boldsymbol{Q}^T \boldsymbol{H}_B \boldsymbol{Q} \dot{\boldsymbol{\Phi}}_B, \tag{4.11}$$

where m_B and $\boldsymbol{H}_B$ are the mass and inertia matrix of the UAV, respectively. Also, kinetic energy of link i is given by:

$$\mathcal{K}_{l_i} = \frac{1}{2} m_{l_i} \dot{\boldsymbol{p}}_{l_i}^T \dot{\boldsymbol{p}}_{l_i} + \frac{1}{2} \boldsymbol{\omega}_{l_i}^T \boldsymbol{R}_B \boldsymbol{R}_{l_i}^B \boldsymbol{H}_{l_i} \boldsymbol{R}_B^{l_i} \boldsymbol{R}_B^T \boldsymbol{\omega}_{l_i}, \tag{4.12}$$

78

where $\mathbf{R}_{l_i}^B$ is the rotation matrix from the frame on link i to the body frame. Also, m_{l_i} and $\mathbf{H}_{l_i}$ are the mass and inertia matrix of link i, respectively. As a result, the total kinetic energy can be expressed in compact form as:

$$\mathcal{K} = \frac{1}{2} \boldsymbol{\dot{\xi}}^T \mathbf{M} \boldsymbol{\dot{\xi}},$$

(4.13)

where $\mathbf{M} \in \mathbb{R}^{(6+n_M) \times (6+n_M)}$ is the symmetric and positive definite inertia matrix whose elements are given by [104]:

$$\mathbf{M}_{11} = \left(m_B + \sum_{i=1}^{n_M} m_{l_i} \right) \mathbf{I_3},$$

(4.14)

$$\mathbf{M}_{33} = \sum_{i=1}^{n_M} \left(m_{l_i} \left(\mathbf{J}_P^{(l_i)} \right)^T \mathbf{J}_P^{(l_i)} + \left(\mathbf{J}_O^{(l_i)} \right)^T \mathbf{R}_{l_i}^B \mathbf{H}_{l_i} \mathbf{R}_B^{l_i} \mathbf{J}_O^{(l_i)} \right),$$

(4.15)

$$\mathbf{M}_{12} = \mathbf{M}_{21}^T = - \sum_{i=1}^{n_M} \left(m_{l_i} \left(\mathbf{R}_B \mathbf{p}_{Bl_i}^B \right) \mathbf{T} \right),$$

(4.16)

$$\mathbf{M}_{13} = \mathbf{M}_{31}^T = \sum_{i=1}^{n_M} \left(m_{l_i} \mathbf{R}_B \mathbf{J}_P^{(l_i)} \right),$$

(4.17)

$$\mathbf{M}_{23} = \mathbf{M}_{32}^T = \sum_{i=1}^{n_M} \left(\mathbf{Q}^T \mathbf{R}_{l_i}^B \mathbf{H}_{l_i} \mathbf{R}_B^{l_i} \mathbf{J}_O^{(l_i)} - m_{l_i} \mathbf{T}^T S \left(\mathbf{R}_B \mathbf{p}_{Bl_i}^B \right)^T \mathbf{R}_B \mathbf{J}_P^{(l_i)} \right),$$

(4.18)

where $\mathbf{I}_\alpha$ denotes the $(\alpha \times \alpha)$ identity matrix. The potential energy of the system can be written as:

$$\mathcal{U} = \mathcal{U}_B + \sum_{i=1}^{n_M} \mathcal{U}_{l_i}, \tag{4.19}$$

where $\mathcal{U}_B$ and $\mathcal{U}_{l_i}$ are the potential energy of the quadcopter and that of link i, respectively. The quadcopter potential energy is given by:

$$\mathcal{U}_b = -m_b g \boldsymbol{e}_3^T \boldsymbol{p}_b, \tag{4.20}$$

in which $\boldsymbol{e}_3 = [0 \quad 0 \quad 1]^T$ if the gravity acts along the opposite direction of Z_B. The potential energy of link i is given by:

$$\mathcal{U}_{l_i} = -m_{l_i} g \boldsymbol{e}_3^T \left(\boldsymbol{p}_B + \boldsymbol{R}_B \boldsymbol{p}_{Bl_i}^B\right). \tag{4.21}$$

Therefore, the total potential energy of the MUAV is given by:

$$\mathcal{U} = -m_B g \boldsymbol{e}_3^T \boldsymbol{p}_B - g \sum_{i=1}^{n_M} \left[m_{l_i} \boldsymbol{e}_3^T \left(\boldsymbol{p}_B + \boldsymbol{R}_B \boldsymbol{p}_{Bl_i}^B\right)\right]. \tag{4.22}$$

Having obtained the MUAV total kinetic and potential energy, one can use (9) to obtain:

$$\boldsymbol{M}(\xi)\ddot{\xi} + \boldsymbol{C}\left(\xi, \dot{\xi}\right)\dot{\xi} + \boldsymbol{g}(\xi) = \boldsymbol{u}_{MUAV}. \tag{4.23}$$

where $M \in \mathbb{R}^{n_\xi \times n_\xi}$ is the symmetric and positive definite inertia matrix whose elements are given in [104]. Also, the input vector u_{MUAV} is given by:

$$u_{MUAV} = \bar{R}Nu, \tag{4.24}$$

in which u is a vector consisting of thrust forces from quadcopter rotors and torques from manipulator joints. What is more, $N = diag(\Omega, I_{n_M})$ and Ω is given by [104]:

$$\Omega = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 0 & -l & 0 & l \\ -l & 0 & l & 0 \\ -c & c & -c & c \end{bmatrix}, \tag{4.25}$$

where l and c are the distance from a rotor to UAV geometric center and drag factor, respectively. What is more, $\bar{R} = diag(R_B, Q^{-1}, I_{n_M})$, $C(\xi, \dot{\xi})$ is the centrifugal and Coriolis matrix and g is the gravitational vector, obtained from the following equations:

$$c_{ij} = \sum_{i=1}^{n_\xi} \frac{1}{2} \left\{ \frac{\partial M_{kj}}{\partial \xi_i} + \frac{\partial M_{ki}}{\partial \xi_j} - \frac{\partial M_{ij}}{\partial \xi_k} \right\} \dot{\xi}_i, \tag{4.26}$$

$$g(\xi) = \frac{\partial \mathcal{U}}{\partial \xi}, \tag{4.27}$$

where M_{ij} is the element on the $i - th$ row and $j - th$ column of M. In the next section, LQR scheme is employed to design the input to simultaneously control the UAV and its robotic manipulator.

4.3 LQR control of a MUAV

For the LQR controller design, in the first step, an infinite horizon quadratic cost function is considered [170]:

$$J(\boldsymbol{u}) = \int_0^\infty \left(\boldsymbol{x}^T \boldsymbol{Q}_{LQR} \boldsymbol{x} + \boldsymbol{u}^T \boldsymbol{R}_{LQR} \boldsymbol{u} \right) dt, \tag{4.28}$$

where $\boldsymbol{x} = [\boldsymbol{\xi}^T \quad \dot{\boldsymbol{\xi}}^T]^T$, $\boldsymbol{Q}_{LQR} \in \mathbb{R}^{(12+2*n_M)\times(12+2*n_M)}$ and $\boldsymbol{R}_{LQR} \in \mathbb{R}^{(4+n_M)\times(4+n_M)}$ are positive definite weight matrices to penalize the deviation from the reference trajectory and the magnitude of the control signal, respectively. In order to minimize the above cost function, one can select a state feedback law of the form:

$$\boldsymbol{u} = -\boldsymbol{K}_{LQR}(\boldsymbol{x} - \boldsymbol{x}^*) + \boldsymbol{u}^*, \tag{4.29}$$

in which $\boldsymbol{x}^*$ is the desired setpoint. In that manner, the optimal gain $\boldsymbol{K}_{LQR}$ is obtained from:

$$\boldsymbol{K}_{LQR} = \boldsymbol{R}_{LQR}^{-1} \boldsymbol{B}^T \boldsymbol{P}, \tag{4.30}$$

and $\boldsymbol{P}$ in turn is obtained from the following algebraic Riccati equation [170]:

$$\boldsymbol{A}^T \boldsymbol{P} + \boldsymbol{P}\boldsymbol{A} - \boldsymbol{P}\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T \boldsymbol{P} + \boldsymbol{Q}_{LQR} = \boldsymbol{0}. \tag{4.31}$$

where $\boldsymbol{A}$ and $\boldsymbol{B}$ are linear state and input matrices, respectively. The positive definite matrices $\boldsymbol{Q}_{LQR}$ and $\boldsymbol{R}_{LQR}$ can be chosen to be diagonal. In that case, increasing the entries of $\boldsymbol{Q}_{LQR}$ results in heavy penalizing of the deviation of the states from the reference trajectory. Therefore, the states tend to rapidly converge to their desired values, albeit with high control input [170]. Similarly, if one increases the entries of $\boldsymbol{R}_{LQR}$, the magnitude of the control signal will be heavily

penalized and the control gains will be small. Given control input with relatively small magnitude, the convergence of states to their desired setpoint occurs in a longer time interval which will result in slower system response. Therefore, proper tuning of the weight matrices should result in timely convergence of the system states to their desired values, while not requiring excessively large control input. Finally, finite (non–zero) steady state error is an expected shortcoming in the performance of the proposed approach. In order to remedy that drawback, one may add an integral action to the LQR scheme. In [109], it was shown that LQR scheme along with an integral action outperforms the above scheme in that it can effectively control a UAV and its endowed manipulator and also successfully remove the steady state error. Consequently, a LQR control scheme with an integral action is adopted in this work. Finally, it is important to note a shortcoming of LQR controller in that it performs well when the MUAV initially begin from conditions close to its operational trim point. However, if the MUAV is expected to start from arbitrarily large deviations from its trim point, or if the MUAV is to compensate for real–world conditions e.g. windy environment, one should consider nonlinear MUAV control design which can better accommodate such scenarios. In the next section, UKF–based MUAV state estimation is discussed.

4.4 UKF state estimation of a MUAV

Kalman filter is probably the most popular scheme in state estimation [153]. Although the original formulation of Kalman filter was developed for linear systems, it is straightforward to extend it to nonlinear systems. The extended Kalman filter can be applied to nonlinear systems as it linearizes the dynamic model (also the measurement model) about the best estimate of the current mean and covariance. It has been shown that EKF performs considerably well in many cases but its performance degrades for highly nonlinear and coupled systems. Unscented Kalman Filter is a relatively new technique to address the shortcomings of EKF. In UKF, a deterministic sampling technique i.e. unscented transform is utilized to generate a minimal set of sample points from the state priori mean and covariance. These sampled points, also known as sigma points, are then propagated through the nonlinear process model and posterior mean and covariance are numerically recovered [153], [173]. To that end, discretized dynamic equations of a MUAV and the measurement model can be written in compact form as:

$$x_k = f(x_{k-1}, u_{k-1}) + w_k, \tag{4.32}$$

$$y_k = Hx_k + v_k, \tag{4.33}$$

where y is the measurement vector, and w_k and v_k are process and measurement noise with zero mean Gaussian distribution and covariance Q_k^* and R_k^*, respectively. It is worthy to note that in our problem, the measurement vector consists of p_B, Φ_B, and q and therefore $H \in \mathbb{R}^{(6+n_M) \times (12+2*n_M)}$ is a constant matrix. However, in the general case, the measurement model can be nonlinear as well. In the following, the General UKF–based state estimation algorithm along with a computationally–efficient variant, known as Spherical UKF, is presented.

4.4.1 MUAV general UKF state estimation

The general unscented Kalman filter for a quadcopter UAV state estimation was discussed in detail in Section 3.4.2. A similar procedure was carried out for GUKF–based MUAV state estimation, summarized in the following algorithm.

GUKF Algorithm

1. Generate the sigma points $\chi_{k-1|k-1}^i$ and propagate them through $\hat{x}_{k|k-1}^i = f(\chi_{k-1|k-1}^i, u_{k-1})$.

2. Predict state estimate $\hat{x}_{k|k-1}$ and the estimation error covariance $P_{k|k-1}$

3. Generate new sigma points $\chi_{k|k-1}^i$, $\hat{y}_k^i$ and $\hat{y}_k$

4. Propagate covariance matrices P_k^y and P_k^{xy} and Kalman gain matrix K_k

5. Update state measurement $\hat{x}_{k|k}$ and covariance matrix $P_{k|k}$

While the GUKF–based state estimation along with the LQR scheme can successfully control a MUAV, its computation complexity can potentially become a bottleneck in actual onboard implementation for autonomous MUAV flight. Therefore, in the next subsection, a computationally efficient variant, Spherical UKF, is presented.

4.4.2 MUAV scaled spherical UKF state estimation

In the current literature, there are two main approaches to improve computational requirements of the GUKF, namely the Simplex UKF (SiUKF) and Spherical UKF (SUKF) [111], [171]. The underlying principle of these methods is that a reduction in number of sigma points reduces computational load of UKF algorithm. To that end, the minimum number of sigma points that provide estimation accuracy in the order of GUKF is found to be $n + 2$ [193], [194]. However, the particular weight allocation procedure in SiUKF can rapidly lead to numerical instability as dimension of the states increases. For MUAV applications, in particular, SiUKF was implemented and it was observed that the algorithm fails for state estimation purposes. The SUKF offers a possible solution to the above shortcoming as it rearranges the sigma points and introduces a much more even weight allocation procedure that provides improved numerical stability [194], [195]. The Spherical sigma points are found using the following algorithm.

SSUKF Algorithm 1

1. Choose the weight $W_0 \in [0, 1)$. The first weight only affects the sigma points fourth and higher–order moments [152], [193].

2. The remaining weights are computed from:

$$W_i = \frac{1 - W_0}{n + 1}, \qquad i = 1, \cdots, n + 1 \tag{4.34}$$

It is important to note that the above weight allocation procedure results in identical weights, except for W_0, and therefore effectively avoids numerical instability of SiUKF algorithm.

3. Initialize the following α vectors:

$$\begin{cases} \alpha_1^{(0)} = 0, \\ \alpha_1^{(1)} = \dfrac{-1}{\sqrt{2W_1}}, \\ \alpha_1^{(2)} = \dfrac{1}{\sqrt{2W_1}}. \end{cases} \tag{4.35}$$

4.   Expand each $\alpha^{(i)}$ vector, $i = 0, \cdots, n+1$, recursively with the following steps for $j = 2, \cdots, n$:

$$\alpha_j^{(i)} = \begin{cases} \begin{bmatrix} \alpha_{j-1}^{(0)} \\ 0 \end{bmatrix}, & i = 0 \\[2em] \begin{bmatrix} \alpha_{j-1}^{(i)} \\ \dfrac{-1}{\sqrt{j(j+1)W_1}} \end{bmatrix}, & i = 1, \cdots, j \\[2em] \begin{bmatrix} \mathbf{0}_{j-1} \\ \dfrac{j}{\sqrt{j(j+1)W_1}} \end{bmatrix}, & i = j+1 \end{cases} \tag{4.36}$$

where in the above equation $\mathbf{0}_{j-1} \in \mathbb{R}^{(j-1)\times 1}$ is a vector containing $j-1$ zeros.

5. Having obtained $\alpha^{(i)}$ vectors recursively, the sigma points are determined by:

$$\chi_{k-1|k-1}^i = \widehat{x}_{k-1|k-1} + \left(\sqrt{P_{k-1|k-1}} \right)_i \cdot \alpha^{(i)} \quad i = 0, \cdots, n+1. \tag{4.37}$$

The sigma points obtained from the above algorithm are propagated through the process model given in (20). Also, a similar procedure is repeated to obtain $\chi_{k|k-1}^i$ which, in turn, will be used in

86

the measurement model given in (24). It is also important to note that the particular choice of $W_0 = 0$ in *Algorithm 1* can reduce the number of sigma points to $n + 1$, while only affecting fourth and higher–order moments of the estimation.

The above algorithm has been shown to provide accurate state estimation for systems with relatively simple dynamics, e.g. induction motors [111]. However, for systems with attitude angle estimation its performance is questionable. Due to the periodic nature, attitude angles do not belong to vector space and this, in turn, leads to problems in sigma point generation. To overcome this drawback, one can employ Scaled Spherical UKF (SSUKF) by scaling the sigma points using the following algorithm.

SSUKF Algorithm 2

1.   Scale the sigma points by modifying step 5 of *Algorithm 1*:

$$\chi^i_{k-1|k-1} = \widehat{x}_{k-1|k-1} + \gamma \left(\sqrt{P_{k-1|k-1}} \right)_i . \alpha^{(i)}, \tag{4.38}$$

where $10^{-4} \leq \gamma \leq 1$ is the scaling factor. Note that for $\gamma = 1$, SSUKF simplifies to SUKF given in *SSUKF Algorithm 1*.

2.   Update the weights W_i for calculating the mean:

$$W_i^m = \begin{cases} \dfrac{W_0 - 1}{\gamma^2} + 1 & i = 0 \\ \dfrac{W_i}{\gamma^2} & i \neq 0 \end{cases} \tag{4.39}$$

3.   Update the weights W_i for calculating the covariance:

$$W_i^c = \begin{cases} \dfrac{W_0 - 1}{\gamma^2} + 2 + \beta - \gamma^2 & i = 0 \\ \dfrac{W_i}{\gamma^2} & i \neq 0 \end{cases} \qquad (4.40)$$

where β is a parameter to reduce higher order effects and $\beta = 2$ is optimal for Gaussian distribution considered in this work. A final note is that the function given in *Algorithm 3* should be used in computing angular distance in mean and covariance computation [196].

SSUKF Algorithm 3 (Computation of angular distance)

Considering two angles $-\pi \leq \theta_i, \theta_j \leq \pi$, the angular distance d_{ij} from θ_i to θ_j is computed as:

1. $d_{ij} = \theta_j - \theta_i$.

2. If $d_{ij} > \pi$, then modify step 1 as $d_{ij} = \theta_j - \theta_i - 2\pi$.

3. If $d_{ij} < -\pi$, then modify step 1 as $d_{ij} = \theta_j - \theta_i + 2\pi$.

With the above modification, the GUKF and SSUKF algorithms can be applied to state estimation problem in MUAV applications.

4.4.3 MUAV low–pass filter state estimation

In order to provide a baseline for comparison purposes, a Low Pass Filter (LPF) is designed to provide MUAV state estimation based on camera pose estimation and encoder data. Here, an Infinite Impulse Response (IIR) LPF of the following form is adopted:

$$\hat{\bar{\xi}}_k = \sum_{k=1}^{M} \Pi_k \hat{\bar{\xi}}_{n-k} + \sum_{k=0}^{N} \Upsilon_k \pmb{y}_{n-k} \qquad (4.41)$$

where the weight elements are shown as $\pmb{\Pi} = \Pi_1, \Pi_2, \cdots, \Pi_M$ and $\pmb{\Upsilon} = \Upsilon_0, \Upsilon_1, \cdots, \Upsilon_N$ in compact form. Also, $\hat{\bar{\xi}}_k \in \mathbb{R}^{n_\xi \times 1}$ consists of the first eight elements of $\hat{\pmb{x}} = \begin{bmatrix} \hat{\bar{\xi}}^T & \hat{\dot{\bar{\xi}}}^T \end{bmatrix}^T$ in the $k - th$ step and $\pmb{y}$ denotes the actual measurements. It is trivial to see that $\pmb{\Pi}$ in the first term of Eq. 4.41 signifies the weight of best estimations (in M previous steps) whereas $\pmb{\Upsilon}$ signifies the weight of actual measurements (in the current and N previous steps). Finally, $\hat{\dot{\bar{\xi}}}$ is obtained from discrete–time difference of $\hat{\bar{\xi}}$ in consecutive time steps.

In the next section, a case study will be developed to evaluate the performance of LPF–based, GUKF–based and SSUKF–based state estimation algorithms along with optimal control scheme to control a MUAV.

4.5 Case study, simulation results and discussion

In this section, a MUAV consisting of a quadcopter equipped with a 2–DoF Revolute–Revolute robotic manipulator, confined to $X_B - Z_B$ plane of $\mathcal{F}_B$, with the characteristics given in Table 4.1, is considered. The LQR design process was carried out in MATLAB where $\pmb{Q}_{LQR}$ was initialized as $0.05\pmb{I}_{12} \in \mathbb{R}^{16 \times 16}$ and the following entries were tuned for satisfactory control performance.

$$Q_{LQR}(i,j) = 5 \qquad \text{for } i = j = 1,2.$$

Also, $\pmb{R}_{LQR}$ weight matrix was initialized as $0.01 \times \pmb{I}_4$ and the entries associated with joint angles were tuned by trial and error:

$$R_{LQR}(i,j) = 1 \qquad \text{for } i = j = 5,6.$$

Having obtained the $\pmb{P}$ matrix from the solution of the ARE equation, the optimal gain matrix $\pmb{K}_{LQR}$ (and thus the optimal control law) was obtained.

Table 4.1 Characteristics of MUAV case study (partially from [42])

Parameter	Value	Unit
UAV mass	4.34	kg
UAV moments of inertia	$\text{diag}\begin{pmatrix} 0.0820, \\ 0.0845 \\ 0.1377 \end{pmatrix}$	Kg.m^2
Distance from a motor to UAV geom. center	0.33	m
Link(s) mass	0.20	kg
Link(s) length	0.20	m

Here, the links are assumed to be identical and servo–driven, see Fig. 2.1 as an example of a robotic manipulator developed for aerial manipulation applications at Robotics, Mechatronics and Automation Laboratory (RMAL) of Ryerson University. What is more, the process noise covariance is assumed to be a constant diagonal matrix given by:

$$\begin{cases} Q^*(i,j) = 10^{-5} & \text{for } i = j = 9,10,11; \\ Q^*(i,j) = 10^{-6} & \text{for } i = j = 12,13,14,15,16; \\ 0 & \text{elsewhere;} \end{cases}$$

where $Q^*(i,j)$ is the $i-th$ and $j-th$ column of $\boldsymbol{Q}^*$. Also, the measurement noise covariance is given by:

$$\begin{cases} R^*(i,j) = 10^{-3} & \text{for } i = j = 1,2, \\ R^*(i,j) = 2 \times 10^{-3} & \text{for } i = j = 3, \\ R^*(i,j) = 10^{-3} & \text{for } i = j = 4,5,6 \\ R^*(i,j) = 10^{-4} & \text{for } i = j = 7,8, \\ 0 & \text{elsewhere.} \end{cases}$$

Regarding the measurement noise covariance, similar to the previous chapter, the values for translational elements are conservatively associated with RTK–GPS accuracies of 1–2 cm and 2–4 cm in horizontal and vertical measurements, respectively [183]–[185]. Also, measurement noise

covariance of rotational element conservatively assumes angle measurements with a few degrees accuracy, reasonably complying with commercial attitude and heading reference systems [186]–[188]. Similarly, measurement of robotic manipulator joint angles conservatively complies with subdegree–level measurement accuracy. Finally, MUAV initial conditions and the reference signal were considered as $[0.5\,\text{m} \quad -0.5\,\text{m} \quad 1\,\text{m} \quad 0\,\text{deg} \quad 0\,\text{deg} \quad 5\,\text{deg} \quad 0 \quad 55\,\text{deg} \quad 0_{1\times8}]^T$ and $[0_{1\times2} \quad 2\,\text{m} \quad 0_{1\times4} \quad 90\,\text{deg} \quad 0_{1\times8}]^T$, respectively. For the described case study, simulations were carried out and results are shown in Figs. 4.2–4.7 and Table 4.2 and Table 4.3.
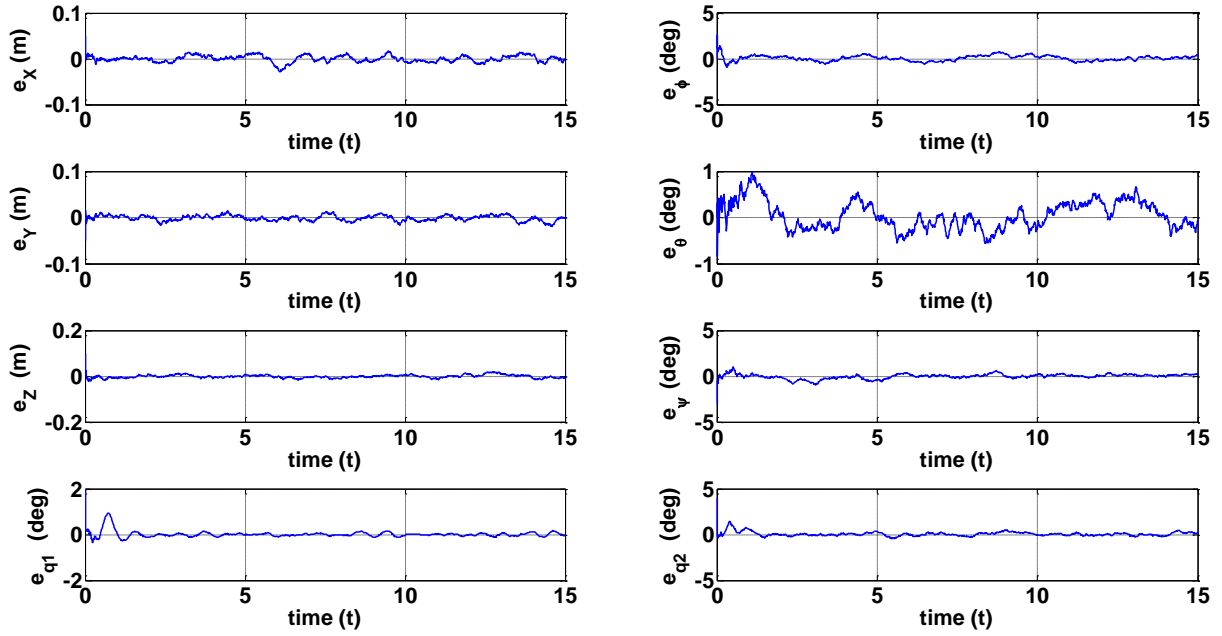
Fig. 4.2 GUKF–based state estimation error of a case study MUAV
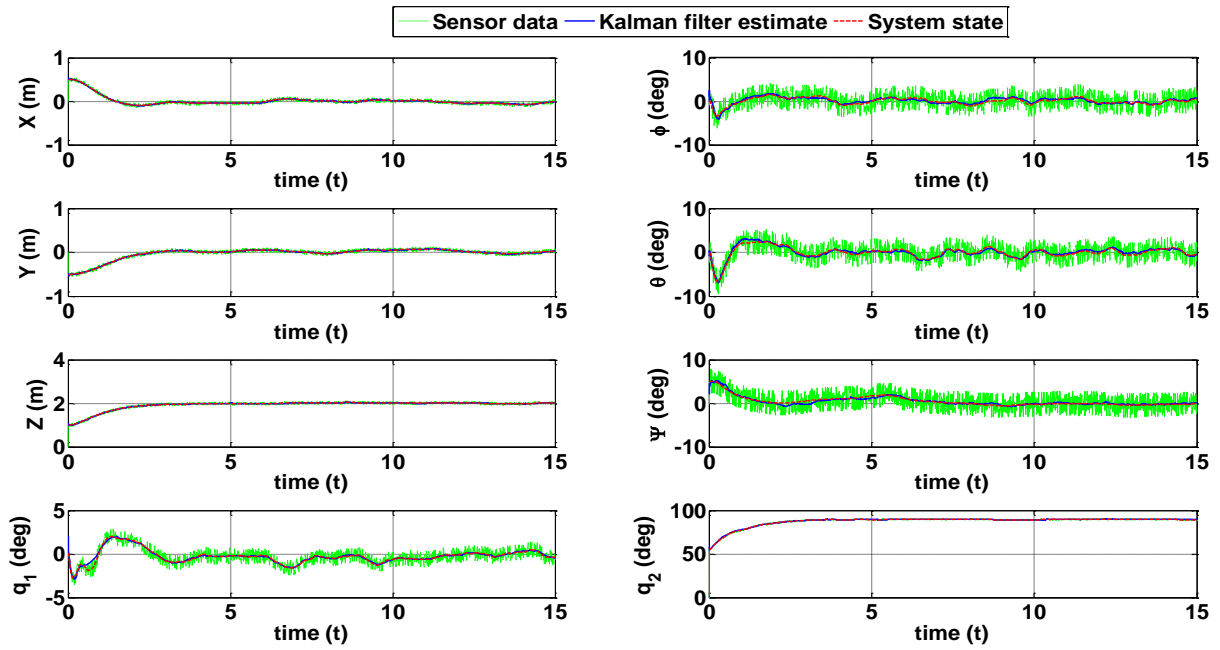
Fig. 4.3 GUKF–based control of a case study MUAV
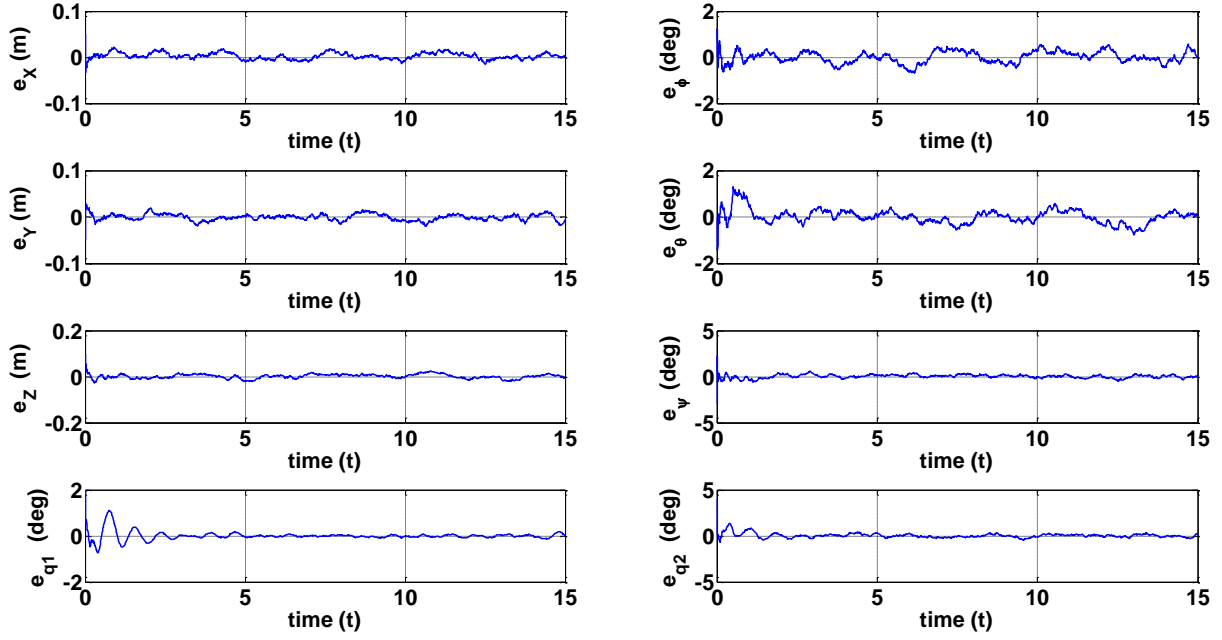
Fig. 4.4 SSUKF–based state estimation error of a case study MUAV
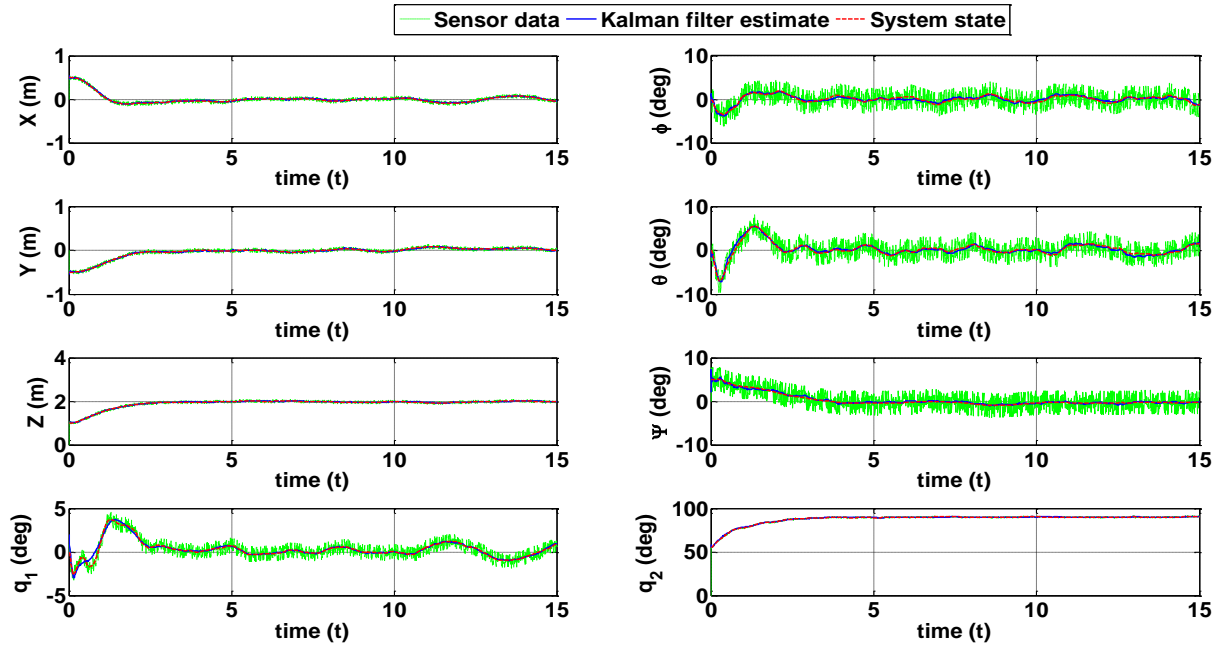
Fig. 4.5 SSUKF–based control of a case study MUAV
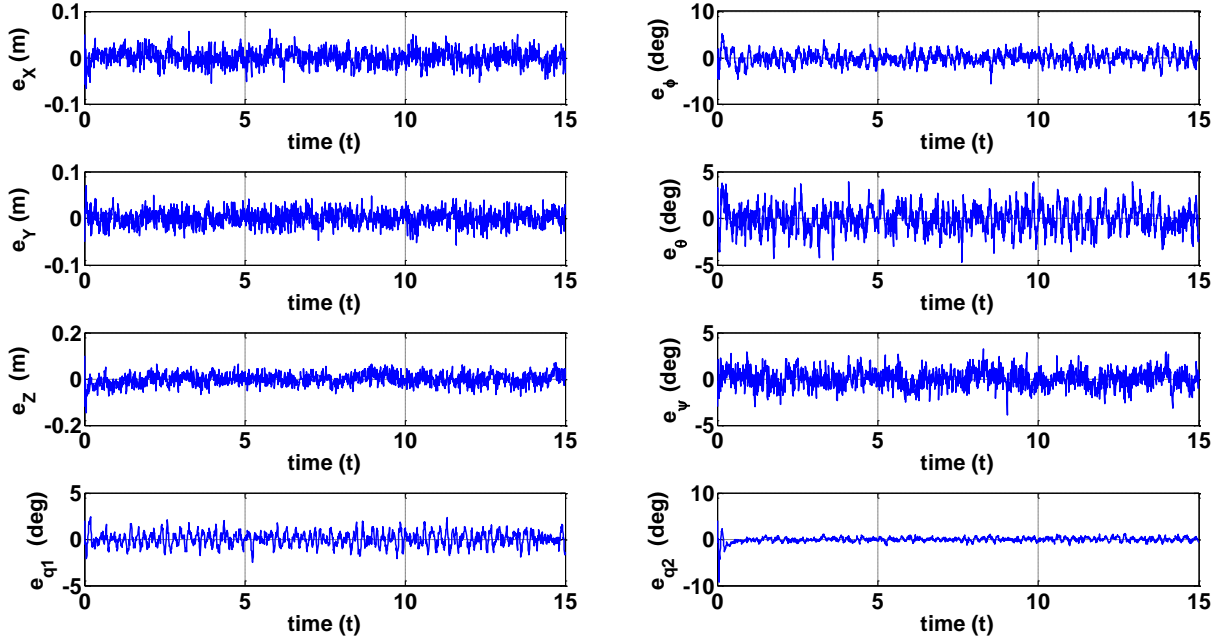
Fig. 4.6 LPF–based state estimation error of a case study MUAV
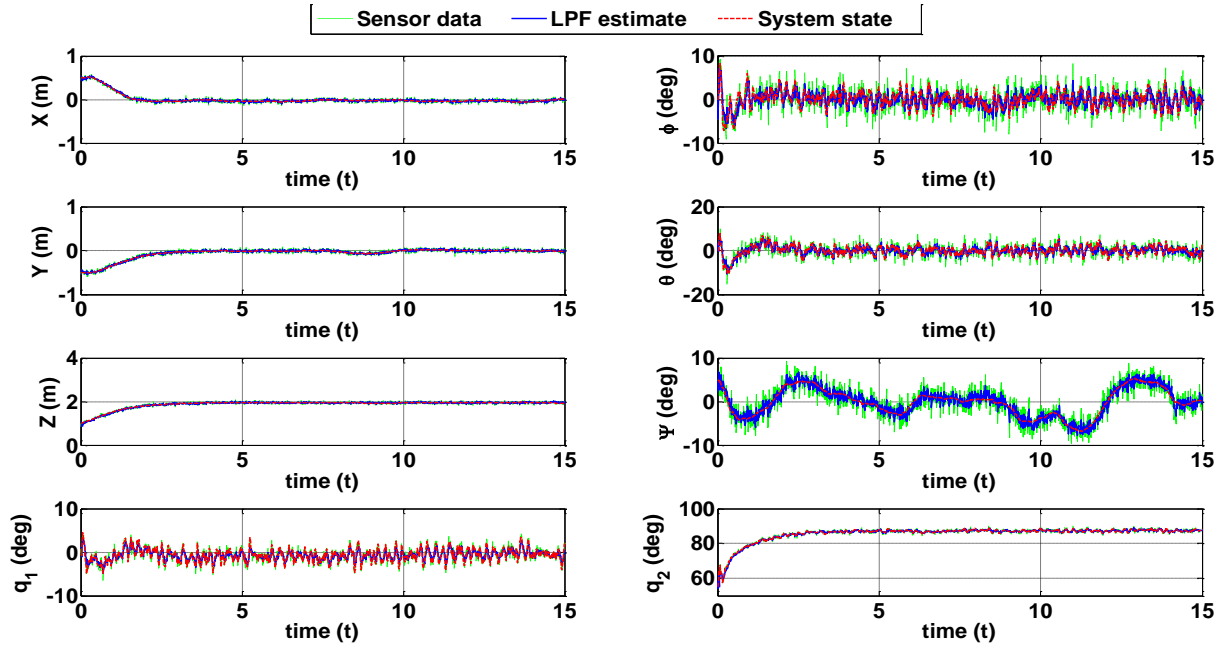
Fig. 4.7 LPF–based control of a case study MUAV

Table 4.2 Estimation error mean and standard deviation

		e_X (cm)	e_Y (cm)	e_Z (cm)	e_φ (deg)	e_θ (deg)	e_ψ (deg)	e_{q1} (deg)	e_{q2} (deg)
GUKF	*Mean*	0.5	0.5	0.6	0.2	0.2	0.2	0.1	0.2
	STD	0.7	0.6	0.7	0.3	0.3	0.3	0.1	0.2
SSUKF	*Mean*	0.6	0.6	0.8	0.2	0.2	0.2	0.1	0.2
	STD	0.7	0.7	1.0	0.3	0.3	0.2	0.2	0.3
LPF	*Mean*	1.3	1.3	1.9	1.1	1.1	0.8	0.6	0.4
	STD	1.7	1.7	2.5	1.3	1.4	1.0	0.7	0.6

Table 4.3 RMSE of translational and rotational degrees of freedom

	$RMSE_X$ (cm)	$RMSE_Y$ (cm)	$RMSE_Z$ (cm)	$RMSE_\varphi$ (deg)	$RMSE_\theta$ (deg)	$RMSE_\psi$ (deg)	$RMSE_{q1}$ (deg)	$RMSE_{q2}$ (deg)
GUKF	11.0	13.0	22.6	0.8	1.3	1.2	0.8	6.1
SSUKF	11.0	13.1	22.6	0.9	1.6	1.4	1.0	6.1
LPF	12.2	13.7	22.9	1.6	2.0	3.2	1.3	6.4

From Fig. 4.2–4.7, it can be readily seen that GUKF–based, SSUKF–based and LPF–based approaches along with LQR control successfully accomplish simultaneous control of the quadcopter and its robotic manipulator. In general, from Table 4.2 and Table 4.3, it can be seen that GUKF–based approach marginally outperforms SSUKF–based approach in estimation accuracy and overall control performance and LPF–based approach is the least accurate in both estimation and control performance. More specifically, estimation of MUAV position vector i.e. X, Y, Z has been achieved with sub–centimeter level accuracy in both GUKF and SSUKF whereas LPF results in 2–3 times less accurate estimation. Similarly, for the attitude angles, GUKF and SSUKF result in estimation accuracy of 0.2 degree whereas LPF offers accuracies close to 1 degree. Similar pattern is observed in manipulator joints angles where UKF–based algorithms result in 0.1 degree and 0.2 degree mean estimation error for q_1 and q_2 whereas LPF results in a few times less accurate estimations. From the overall control performance perspective, in translational X, Y, Z channels, GUKF and SSUKF performance are comparable. For LPF, while overall performance in Z channel is slightly worse than UKF–based approaches, meaningful degradation in X and Y channels control is observed. In particular, the largest

degradation occurs in X channel as it is most negatively affected by cumulative effect of imperfect control of the UAV and also less–accurate robotic manipulator control, as presented in Table 4.3. For the attitude angles, GUKF performance is slightly better than that of SSUKF, while LPF is again the least accurate. In particular, from Fig. 4.7, it can be seen that LPF performance is questionable in ψ channel where RMSE metric is more than 3 degrees. This is mainly attributed to the estimation accuracy of 0.8 degree, approximately 4 times worse than UKF–based approaches. For the manipulator control, GUKF outperforms SSUKF, and the latter outperforms LPF, in terms of overall control performance. It is concluded that for scenarios with low noise level, GUKF and SSUKF result in accurate and comparable estimation and overall control performance metrics, whereas LPF results in stable, yet less accurate metrics.

Next, performance of UKF–based approaches starting with a number of initial conditions is investigated. For that purpose, we define 10 initial conditions scenarios. In the first 5 scenarios the attitude MUAV angles grow further from the hovering point whereas in the next five scenarios manipulator joint angles vary. More specifically, in the first 5 scenarios, attitude angles vary with 5–degree steps such that the initial conditions in the 5th scenario are given by $[0.5\,\text{m} \quad -0.5\,\text{m} \quad 1\,\text{m} \quad 25\,\text{deg} \quad 25\,\text{deg} \quad 30\,\text{deg} \quad 0 \quad 55\,\text{deg} \quad 0_{1\times8}]^T$. Then, in the next 5 scenarios, manipulator joint angles grow further from the setpoint in 5–degree steps such that in the 10th scenario the initial conditions are given by $[0.5\,\text{m} \quad -0.5\,\text{m} \quad 1\,\text{m} \quad 0\,\text{deg} \quad 0\,\text{deg} \quad 5\,\text{deg} \quad -25\,\text{deg} \quad 30\,\text{deg} \quad 0_{1\times8}]^T$. For the described case studies, simulations were carried out and the obtained results are presented in Table 4.4–4.7.

Table 4.4 Estimation error mean and standard deviation in attitude–angle initial conditions scenarios

			e_X (cm)	e_Y (cm)	e_Z (cm)	e_φ (deg)	e_θ (deg)	e_ψ (deg)	e_{q1} (deg)	e_{q2} (deg)
Scenario #1	GUKF	*Mean*	0.5	0.5	0.6	0.2	0.2	0.2	0.1	0.2
		STD	0.7	0.6	0.7	0.3	0.3	0.3	0.1	0.2
	SSUKF	*Mean*	0.6	0.6	0.8	0.2	0.2	0.2	0.1	0.2
		STD	0.7	0.7	1.0	0.3	0.3	0.2	0.2	0.3
	LPF	*Mean*	1.3	1.3	1.9	1.1	1.1	0.8	0.6	0.4
		STD	1.7	1.7	2.5	1.3	1.4	1.0	0.7	0.6
Scenario #2	GUKF	*Mean*	0.5	0.5	0.6	0.2	0.2	0.2	0.1	0.2
		STD	0.7	0.6	0.7	0.3	0.3	0.3	0.1	0.2
	SSUKF	*Mean*	0.6	0.6	0.8	0.2	0.2	0.2	0.1	0.2
		STD	0.7	0.7	1.0	0.3	0.3	0.2	0.2	0.3
	LPF	*Mean*	1.3	1.3	1.9	1.1	1.1	0.8	0.6	0.4
		STD	1.7	1.7	2.5	1.3	1.4	1.0	0.7	0.6
Scenario #3	GUKF	*Mean*	0.5	0.5	0.6	0.2	0.2	0.2	0.1	0.2
		STD	0.7	0.6	0.7	0.3	0.3	0.3	0.2	0.2
	SSUKF	*Mean*	0.6	0.6	0.8	0.2	0.2	0.2	0.1	0.2
		STD	0.7	0.7	1.0	0.3	0.3	0.2	0.2	0.3
	LPF	*Mean*	1.4	1.3	1.9	1.1	1.1	0.8	0.6	0.4
		STD	1.7	1.7	2.5	1.3	1.4	1.0	0.7	0.6
Scenario #4	GUKF	*Mean*	0.5	0.5	0.6	0.2	0.2	0.2	0.1	0.2
		STD	0.7	0.6	0.7	0.3	0.3	0.3	0.2	0.2
	SSUKF	*Mean*	0.6	0.6	0.8	0.2	0.2	0.2	0.1	0.2
		STD	0.7	0.7	1.0	0.3	0.3	0.3	0.2	0.3
	LPF	*Mean*	1.4	1.3	1.9	1.1	1.1	0.8	0.6	0.4
		STD	1.7	1.7	2.5	1.3	1.4	1.0	0.7	0.6
Scenario #5	GUKF	*Mean*	0.6	0.5	0.6	0.2	0.3	0.2	0.1	0.2
		STD	0.8	0.6	0.8	0.3	0.3	0.3	0.2	0.2
	SSUKF	*Mean*	0.6	0.6	0.8	0.2	0.3	0.2	0.1	0.2
		STD	0.8	0.7	1.0	0.3	0.4	0.3	0.2	0.3
	LPF	*Mean*	1.4	1.3	1.9	1.1	1.1	0.8	0.6	0.4
		STD	1.7	1.7	2.5	1.4	1.4	1.0	0.7	0.6

Table 4.5 RMSE metric in attitude–angle initial conditions scenarios

		RMSE$_X$ (cm)	RMSE$_Y$ (cm)	RMSE$_Z$ (cm)	RMSE$_\varphi$ (deg)	RMSE$_\theta$ (deg)	RMSE$_\psi$ (deg)	RMSE$_{q1}$ (deg)	RMSE$_{q2}$ (deg)
Scenario #1	GUKF	11.9	14.6	22.6	1.1	1.5	2.1	0.9	6.5
	SSUKF	11.9	14.7	22.7	1.2	1.8	2.6	1.1	6.5
	LPF	13.2	15.1	23.2	1.8	2.3	3.3	1.4	6.6
Scenario #2	GUKF	12.9	16.5	22.8	1.5	1.9	3.2	1.0	6.9
	SSUKF	13.1	16.4	22.9	1.7	2.3	3.8	1.4	6.9
	LPF	14.4	16.5	23.6	2.1	2.6	3.8	1.4	6.9
Scenario #3	GUKF	13.8	19.2	23.0	2.1	2.5	4.4	1.2	7.4
	SSUKF	14.6	19.0	23.2	2.3	2.9	5.0	1.7	7.5
	LPF	15.5	19.0	24.2	2.4	3.0	4.5	1.8	7.8
Scenario #4	GUKF	14.9	23.5	23.4	2.8	3.1	5.8	1.5	7.8
	SSUKF	14.9	23.8	23.6	2.8	3.4	6.0	1.7	8.0
	LPF	16.5	24.9	24.8	2.8	3.5	6.8	1.8	8.2
Scenario #5	GUKF	16.2	30.0	23.9	3.8	3.9	7.5	1.8	8.3
	SSUKF	16.5	30.2	24.2	3.8	3.9	7.8	1.8	8.4
	LPF	17.3	32.0	25.5	4.1	4.0	8.5	1.9	8.8

Table 4.6 Estimation error mean and standard deviation in joint–angle initial conditions scenarios

			e_X (cm)	e_Y (cm)	e_Z (cm)	e_φ (deg)	e_θ (deg)	e_ψ (deg)	e_{q1} (deg)	e_{q2} (deg)
Scenario #6	GUKF	*Mean*	0.5	0.5	0.6	0.2	0.2	0.2	0.1	0.2
		STD	0.7	0.6	0.6	0.3	0.3	0.3	0.1	0.2
	SSUKF	*Mean*	0.6	0.6	0.8	0.2	0.2	0.2	0.1	0.2
		STD	0.7	0.7	1.0	0.3	0.3	0.2	0.2	0.3
	LPF	*Mean*	1.3	1.3	1.9	1.1	1.1	0.8	0.6	0.4
		STD	1.7	1.7	2.5	1.3	1.4	1.0	0.7	0.6
Scenario #7	GUKF	*Mean*	0.5	0.5	0.6	0.2	0.2	0.2	0.1	0.2
		STD	0.7	0.6	0.7	0.3	0.3	0.3	0.1	0.2
	SSUKF	*Mean*	0.6	0.6	0.8	0.2	0.2	0.2	0.2	0.2
		STD	0.7	0.7	1.0	0.3	0.3	0.2	0.3	0.3
	LPF	*Mean*	1.3	1.3	2.0	1.1	1.1	0.8	0.6	0.4
		STD	1.7	1.7	2.8	1.3	1.4	1.0	0.7	0.6
Scenario #8	GUKF	*Mean*	0.5	0.5	0.6	0.2	0.2	0.2	0.2	0.2
		STD	0.7	0.6	0.7	0.3	0.3	0.3	0.3	0.2
	SSUKF	*Mean*	0.6	0.6	0.8	0.2	0.2	0.2	0.2	0.2
		STD	0.7	0.7	1.0	0.3	0.3	0.2	0.3	0.3
	LPF	*Mean*	1.4	1.3	2.0	1.1	1.1	0.8	0.6	0.4
		STD	1.7	1.7	2.8	1.3	1.4	1.0	0.8	0.6
Scenario #9	GUKF	*Mean*	0.5	0.5	0.6	0.2	0.3	0.2	0.2	0.2
		STD	0.7	0.6	0.7	0.3	0.3	0.3	0.3	0.3
	SSUKF	*Mean*	0.6	0.6	0.9	0.3	0.3	0.2	0.2	0.2
		STD	0.7	0.7	1.1	0.3	0.3	0.3	0.3	0.3
	LPF	*Mean*	1.4	1.3	2.0	1.1	1.1	0.8	0.6	0.4
		STD	1.7	1.7	2.7	1.3	1.4	1.0	0.8	0.6
Scenario #10	GUKF	*Mean*	0.5	0.5	0.6	0.3	0.3	0.2	0.2	0.3
		STD	0.7	0.6	0.7	0.3	0.3	0.3	0.3	0.4
	SSUKF	*Mean*	0.6	0.6	0.9	0.3	0.3	0.3	0.2	0.3
		STD	0.7	0.7	1.1	0.3	0.3	0.3	0.3	0.4
	LPF	*Mean*	1.4	1.3	2.0	1.1	1.1	0.8	0.6	0.4
		STD	1.7	1.7	2.7	1.3	1.4	1.0	0.8	0.6

Table 4.7 RMSE metric in joint–angle initial conditions scenarios

		RMSE$_X$ (cm)	RMSE$_Y$ (cm)	RMSE$_Z$ (cm)	RMSE$_\varphi$ (deg)	RMSE$_\theta$ (deg)	RMSE$_\psi$ (deg)	RMSE$_{q1}$ (deg)	RMSE$_{q2}$ (deg)
Scenario #6	GUKF	11.0	13.1	22.6	0.8	1.4	1.2	1.0	6.9
	SSUKF	11.0	13.1	22.6	0.9	1.6	1.4	1.2	6.9
	LPF	12.1	13.8	22.9	1.6	2.0	3.2	1.3	7.0
Scenario #7	GUKF	11.2	13.2	22.6	0.8	1.4	1.2	1.4	7.6
	SSUKF	11.2	13.4	22.6	0.9	1.6	1.4	1.6	7.5
	LPF	12.2	13.8	22.9	1.6	2.0	3.2	1.6	7.7
Scenario #8	GUKF	11.4	13.4	22.6	0.8	1.4	1.2	1.9	8.2
	SSUKF	11.5	13.7	22.6	0.9	1.7	1.5	2.0	8.0
	LPF	13.1	13.9	22.9	1.6	2.0	3.3	2.1	8.3
Scenario #9	GUKF	12.0	13.8	22.7	0.8	1.6	1.2	2.3	8.5
	SSUKF	12.2	14.3	22.6	0.9	1.8	1.8	2.4	8.5
	LPF	13.4	14.4	22.9	1.7	2.0	3.4	2.5	8.7
Scenario #10	GUKF	12.2	14.0	22.7	0.8	1.6	1.2	2.8	8.8
	SSUKF	12.4	14.7	22.6	1.0	1.8	1.9	3.0	9.2
	LPF	13.8	14.7	22.9	1.7	2.0	3.5	2.5	9.7

For a wide range of initial conditions, it can be seen that both UKF–based algorithms and also the LPF–based algorithm provide accurate state estimation. More specifically, for varying initial attitude angles, subcentimeter and subdegree accuracy in translational and rotational channels is maintained over all 5 scenarios. Similarly, estimation accuracy in translational and rotational channels is well maintained for scenarios with variation in joint angles. Yet, as the difference between joint angles initial conditions and setpoint increase, it is observed that estimation accuracy worsens to some extent. For instance, in the last scenario (Scenario #10), estimation accuracy of GUKF and SSUKF algorithm has decreased by 0.1 degree in both joint angle channels. It is interesting to note that estimation accuracy of LPF–based algorithm has remained almost constant in all scenarios as the LPF–based approach does not explicitly rely on MUAV dynamics for state estimation and therefore is least disturbed by variations in initial conditions. From the control perspective, for scenarios with variation in attitude angle initial conditions, it is

observed that all translational, rotational, and also joint angle channels RMSE metric worsen consistently as the initial deviation from the setpoint increases. For scenarios with variation in joint angles initial conditions, translational and rotational channels experience moderate variation whereas more noticeable degradation in RMSE metric of rotational channels is observed. It is concluded that UKF–based algorithm and the LPF–based approach maintain their performance with marginal degradation for scenarios with various initial conditions of attitude angles and manipulator joint angles. Also, LQR–based controller was able to accomplish the setpoint tracking mission, albeit with increasing RMSE metric.

It is also worthy to mention that algorithm execution time is an important element, especially for sample–based approaches such as UKF. The MATLAB–based simulations in this work were carried out on an Intel® Core™–i5 3.2 GHz processor based computer with 8.00 GB RAM. From Table 4.8, it can be readily seen that the average execution time for GUKF estimation is approximately 4.7 ms whereas SSUKF offers approximately 35% improvement, resulting in 3.1 ms for state estimation in each time step. This considerable improvement in execution time is directly attributed to the reduced number of sigma points required in SSUKF, compared to GUKF. It is also important to note that both UKF–based algorithms result in approximately 2 orders of magnitude longer execution time, compared to simple LPF algorithm. The noticeable advantage in execution time of LPF–based approach is related to the fact that it first obtains position, orientation and joint variables based on the sensory data and best estimates in current and previous steps and then uses first–order differentiation to obtain the associated derivatives. As such, the LPF–based approach completely discards the complex and coupled MUAV dynamic model, does not generate/propagate a finite number of sigma points, and also does not require computationally–expensive matrix operations (e.g. computation of matrix square root, covariance matrices, Kalman gain) in UKF–based approaches. Yet, the advantageous execution time of this approach, compared to UKF–based approaches, comes at the cost of less accurate state estimation and control, and sensitivity to increasing noise levels.

Table 4.8 Execution time comparison of GUKF and SUKF schemes

	Average execution time per iteration (ms)
GUKF	4.7
SSUKF	3.1
LPF	0.1

Sensitivity to increasing noise levels is also important consideration for real–life applications. In the next subsection, the effect of increasing noise level on UKF algorithms estimation and overall control performance is examined.

4.5.1 Noise level scenarios

Here, two noise level scenarios are defined to examine sensitivity of GUKF and SSUKF to increasing noise levels. In the first scenario, measurement noise covariance matrix is considered to be 2 times larger than the scenario in previous subsection whereas measurement noise covariance matrix in the second scenario is 3 times larger than its original value. For the mentioned scenarios, simulations were carried out and obtained results for the first noise level scenario are shown in Figs. 4.8–4.11 and Table 4.9 and Table 4.10.
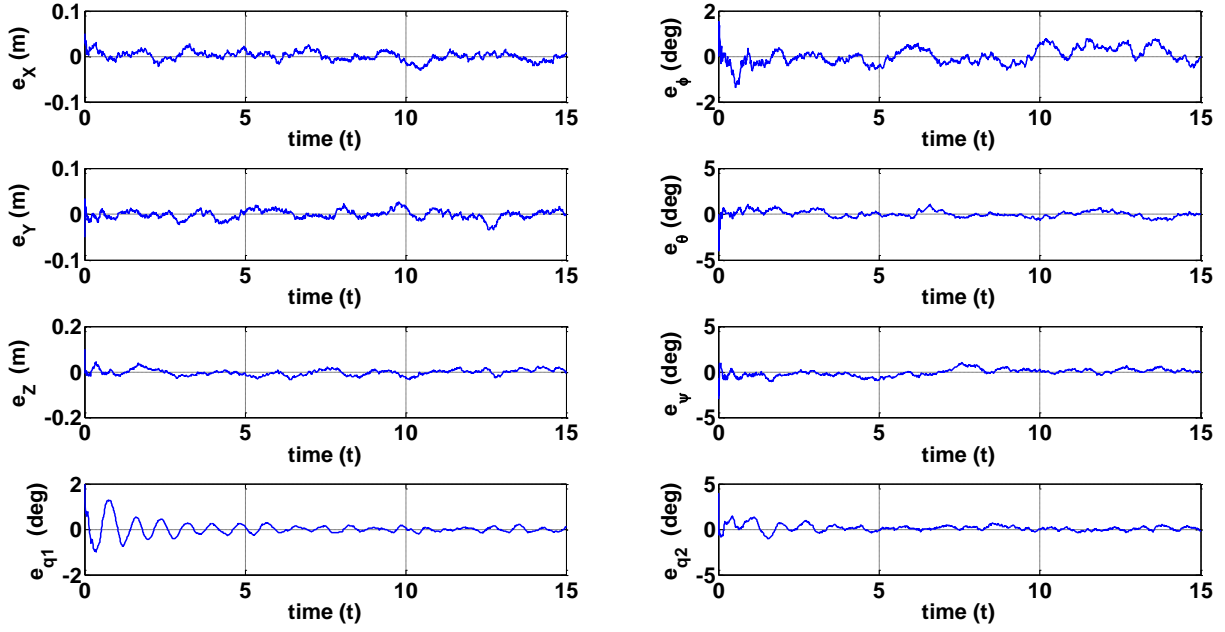
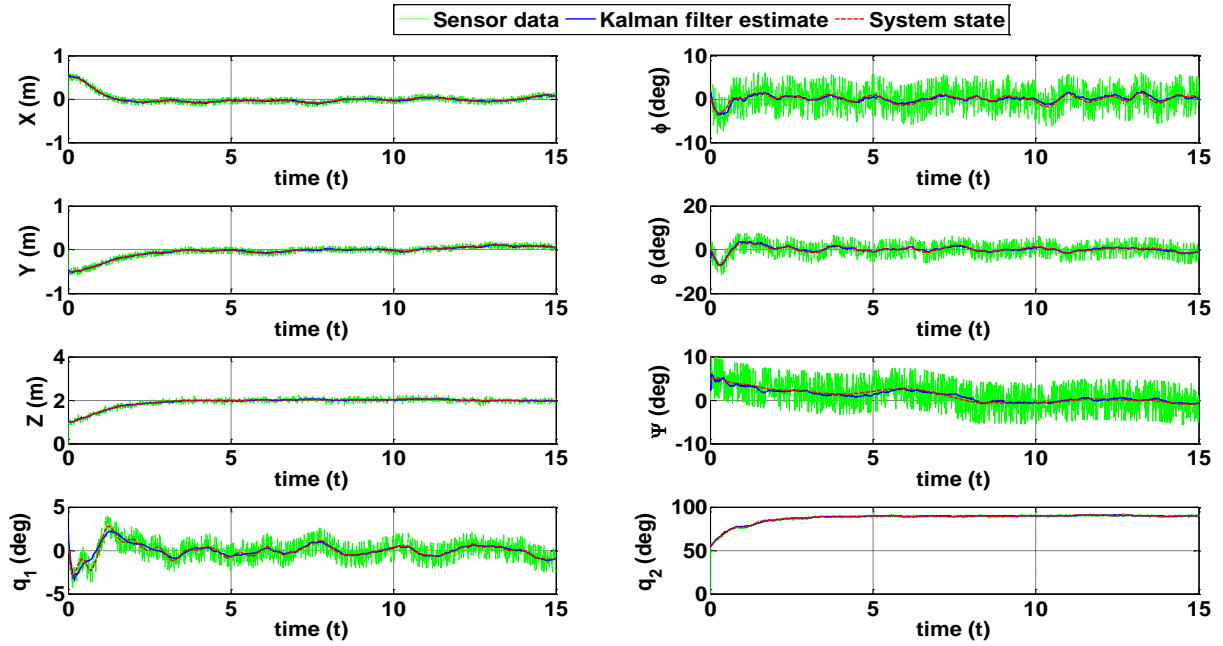Fig. 4.8 GUKF–based state estimation error of a case study MUAV in noise level scenario #1

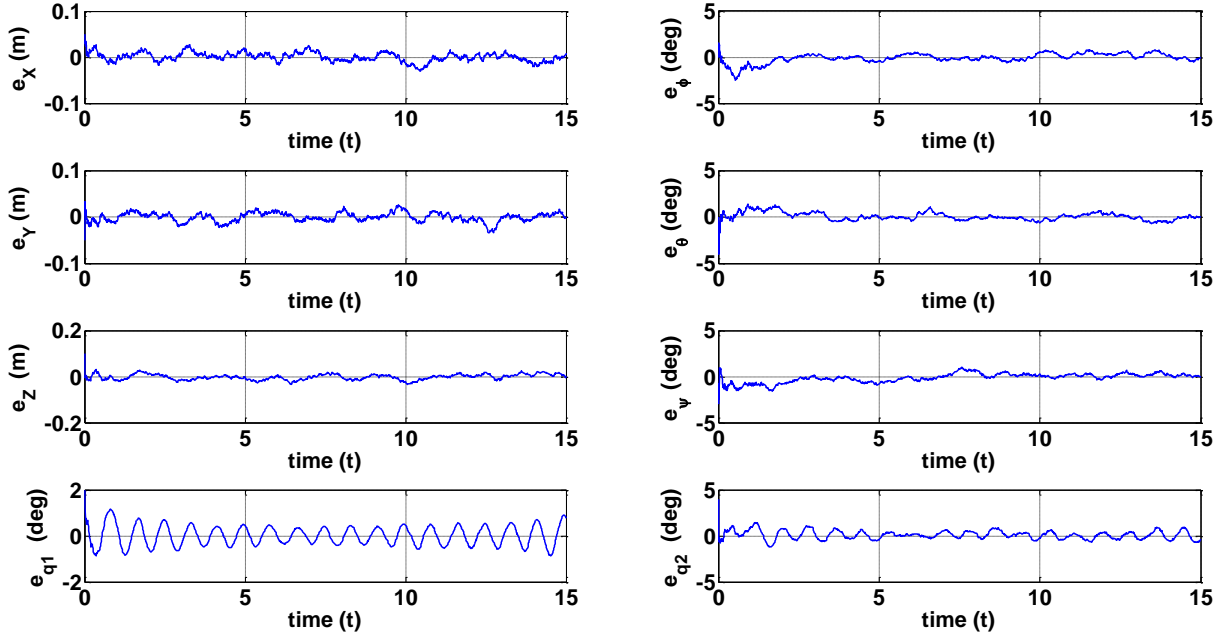Fig. 4.9 GUKF–based control of a case study MUAV in noise level scenario #1

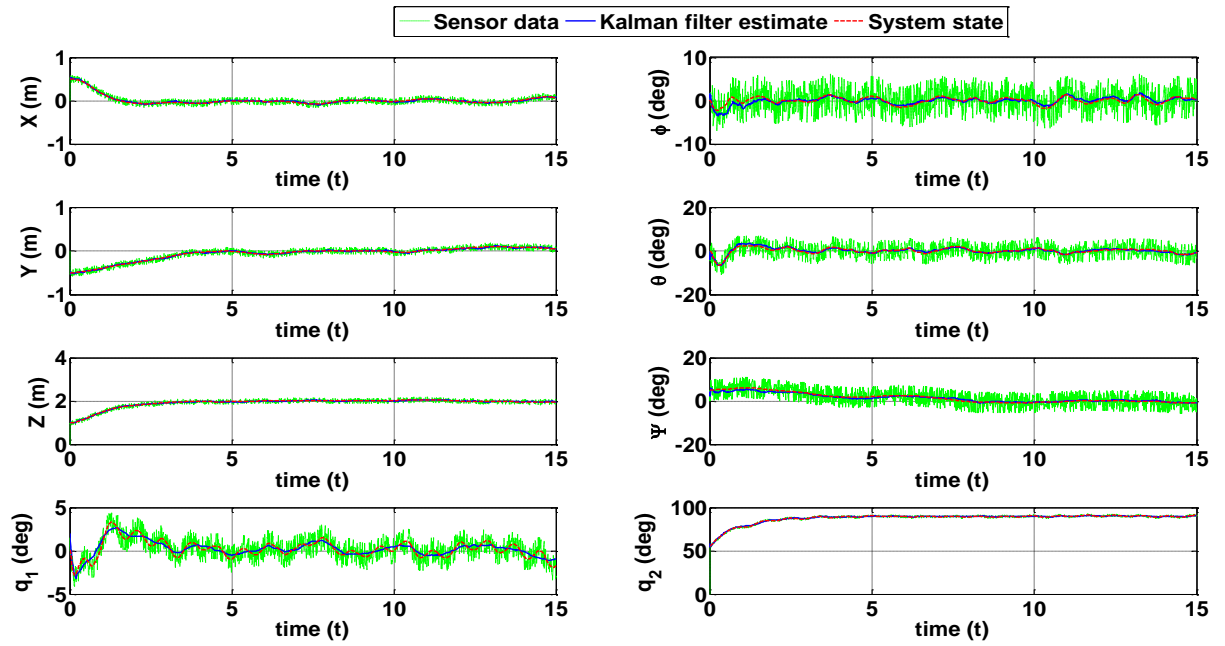Fig. 4.10 SSUKF–based state estimation error of a case study MUAV in noise level scenario #1

Fig. 4.11 SSUKF–based control of a case study MUAV in noise level scenario #1

Table 4.9 Estimation error mean and standard deviation in noise level scenario #1

		e_X (cm)	e_Y (cm)	e_Z (cm)	e_φ (deg)	e_θ (deg)	e_ψ (deg)	e_{q1} (deg)	e_{q2} (deg)
GUKF	*Mean*	0.8	0.8	1.1	0.3	0.3	0.3	0.2	0.3
	STD	1.0	1.0	1.4	0.4	0.4	0.4	0.3	0.4
SSUKF	*Mean*	0.8	0.8	1.0	0.4	0.3	0.4	0.4	0.4
	STD	1.0	1.0	1.4	0.5	0.4	0.5	0.5	0.5
LPF		N/A							

Table 4.10 RMSE metric in noise level scenario #1

	$RMSE_X$ (cm)	$RMSE_Y$ (cm)	$RMSE_Z$ (cm)	$RMSE_\varphi$ (deg)	$RMSE_\theta$ (deg)	$RMSE_\psi$ (deg)	$RMSE_{q1}$ (deg)	$RMSE_{q2}$ (deg)
GUKF	11.4	14.0	23.1	0.8	1.4	1.6	0.8	6.1
SSUKF	11.3	16.5	23.1	0.8	1.4	2.3	0.9	6.1
LPF	N/A							

It is important to mention that LPF was found to be particularly sensitive to increased noise level and simulation results show that LPF fails to achieve meaningful control of the MUAV in this scenario. This complies with the observation that most MUAVs without efficient state estimation algorithms in the current literature either heavily depend on highly–precise motion capture system or employ sensor suites with various redundant sensors. Regarding our proposed UKF–based algorithms, in accordance with Fig. 4.8 and Fig. 4.9, GUKF approach can still successfully control the MUAV, in spite of increased noise level. However, comparing metrics of Table 4.9 and Table 4.10 to those in the previous subsection, it can be readily seen that the estimation errors and overall control performance have degraded to some extent. More specifically, estimation errors of translation channels i.e. X, Y, Z have worsened to be close to 1 cm, along with minor degradation in attitude angles estimation. Similarly, estimation of the manipulator angles has worsened, leading to minor degradation of overall control performance in manipulator control. Based on the obtained data, it is concluded that GUKF–based approach has degraded gracefully in increased noise level Scenario #1. Compared to GUKF, simulation results show that estimation and control degradation is more important in the SSUKF–based approach. In

particular, while the estimation errors in translational channels are similar to those in GUKF approach, the overall control performance in Y channel has degraded approximately 18%. Also, for the attitude angle ψ, both estimation and overall control performance are noticeably degraded, compared to GUKF. Most importantly, from Fig. 4.10, it can be observed that estimation of manipulator first angle q_1 has led to a recurring behaviour where periodic estimation error persists over simulation time. While the amplitude of this periodic behaviour is relatively small, as reflected in marginal degradation in overall control performance in q_1 channel, it indicates that SSUKF is more severely affected by increasing noise levels. This will be more prominent in the next scenario where even larger noise levels, 5 times greater than the original values, are considered. For such a scenario, simulation results are presented in Fig. 4.12 and Fig. 4.13, along with Table 4.11 and Table 4.12.
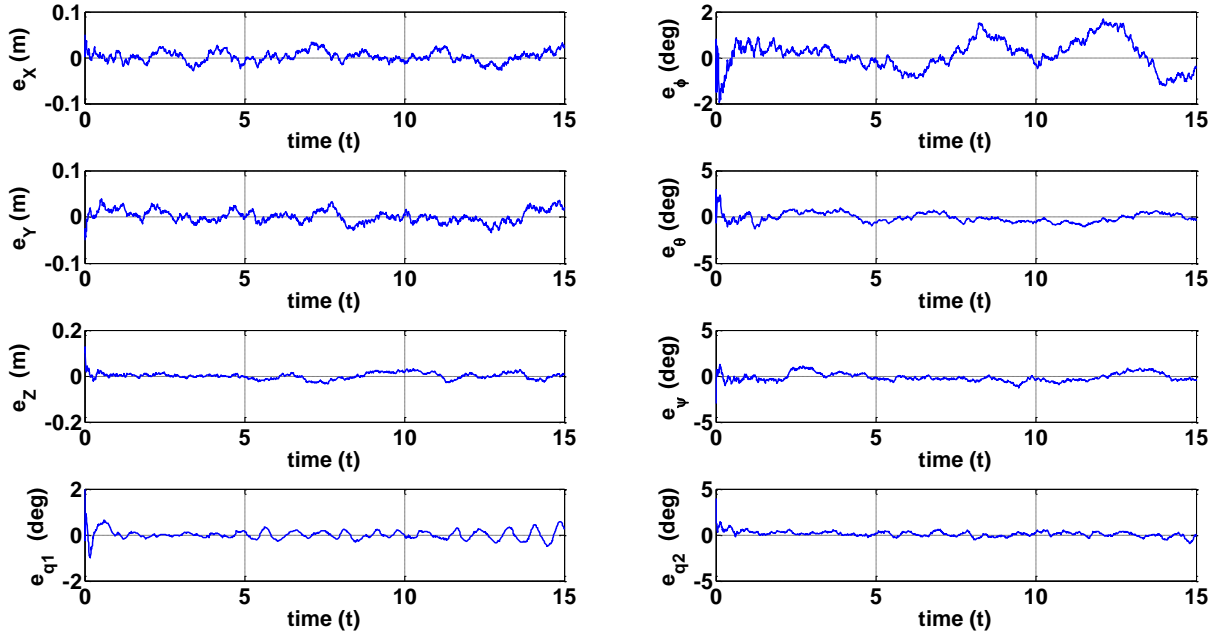
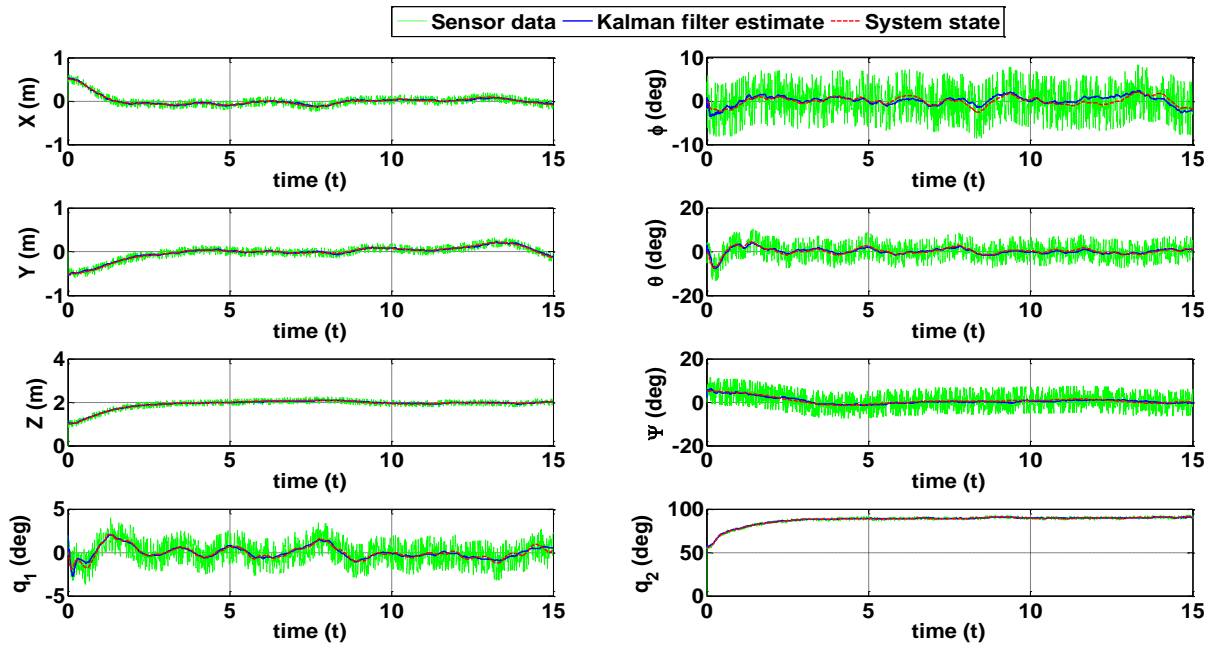Fig. 4.12 GUKF–based state estimation error of a case study MUAV in noise level scenario #2

Fig. 4.13 GUKF–based control of a case study MUAV in noise level scenario #2

Table 4.11 Estimation error mean and standard deviation in noise level scenario #2

		e_X (cm)	e_Y (cm)	e_Z (cm)	e_φ (deg)	e_θ (deg)	e_ψ (deg)	e_{q1} (deg)	e_{q2} (deg)
GUKF	*Mean*	1.0	1.1	0.9	0.5	0.4	0.4	0.2	0.2
	STD	1.2	1.3	1.2	0.7	0.5	0.4	0.2	0.3
SSUKF	N/A								
LPF	N/A								

Table 4.12 RMSE metric in noise level scenario #2

	$RMSE_X$ (cm)	$RMSE_Y$ (cm)	$RMSE_Z$ (cm)	$RMSE_\varphi$ (deg)	$RMSE_\theta$ (deg)	$RMSE_\psi$ (deg)	$RMSE_{q1}$ (deg)	$RMSE_{q2}$ (deg)
GUKF	11.4	15.1	23.2	1.1	1.5	1.7	0.7	6.2
SSUKF	N/A							
LPF	N/A							

Similar to noise Scenario #1, LPF fails to properly control the MUAV in Scenario #2 as well. This was expected as increased noise levels lead to inaccurate results from LPF which will further worsen with excessive noise levels. What is more, SSUKF also fail to achieve stable control of the MUAV in noise Scenario #2. This was also expected as poor estimation in attitude angles and also recurring behaviour in state estimation of manipulator angles was already observed in the previous scenario with lower noise level. In this scenario, increasing the noise level has resulted in even less accurate state estimation which has eventually led to lack of MUAV stable control. Unlike SSUKF, it is observed that GUKF–based approach undergoes graceful degradation in this scenario as well. In particular, the estimation of the translational channels X, Y, Z has worsened to about 1 cm wheras attitude angle estimations are worsened to about 1 degree. Combined with relatively accurate manipulator angles of 0.2 degrees, the overall control performance has degraded gracefully as well, yet preventing complete loss of MUAV control. It is important to note that relatively small oscillation have begun to occur in manipulator angle estimation, suggesting that further increase of noise level will result in total loss of MUAV control in GUKF as well. It is concluded that LPF is the most sensitive to increasing noise levels, whereas SSUKF performs well to some extent and results in control loss

for larger noise levels. GUKF–based approach was found to be the least sensitive to noise levels as its performance degrades gracefully even for noise levels several times larger than the original values. Excessively increasing the noise levels will eventually degrade GUKF performance to the extent where inevitable control loss is experienced.

In the next subsection, another important scenario for MUAV real–life applications, i.e., total sensory loss, is studied.

4.5.2 Total loss of sensory data

Here, a scenario where all sensory data are lost over a period of time is introduced. In order to study the performance of UKF–based algorithms in terms of estimation and overall control performance, it is assumed that no sensory data is available for a 2.5 s period. For that scenario, simulations were carried out and obtained results are shown in Fig. 4.14–4.17, Table 4.13 and Table 4.14.

Fig. 4.14 GUKF–based state estimation error of a case study MUAV in loss of sensory data scenario

Fig. 4.15 GUKF–based control of a case study MUAV in loss of sensory data scenario

110

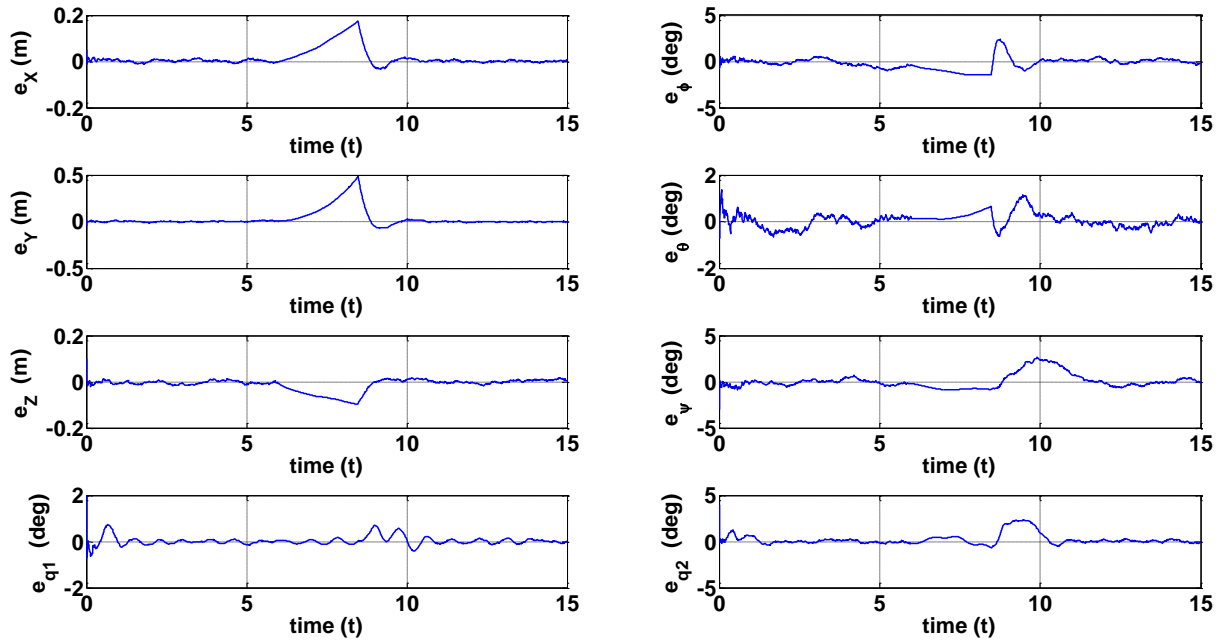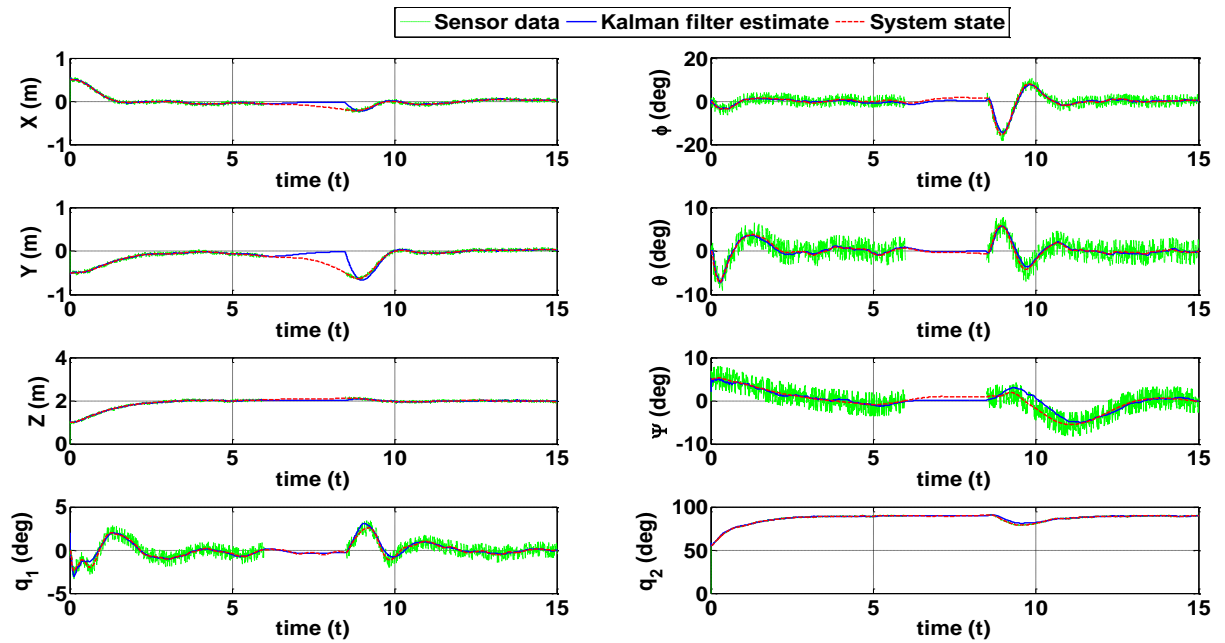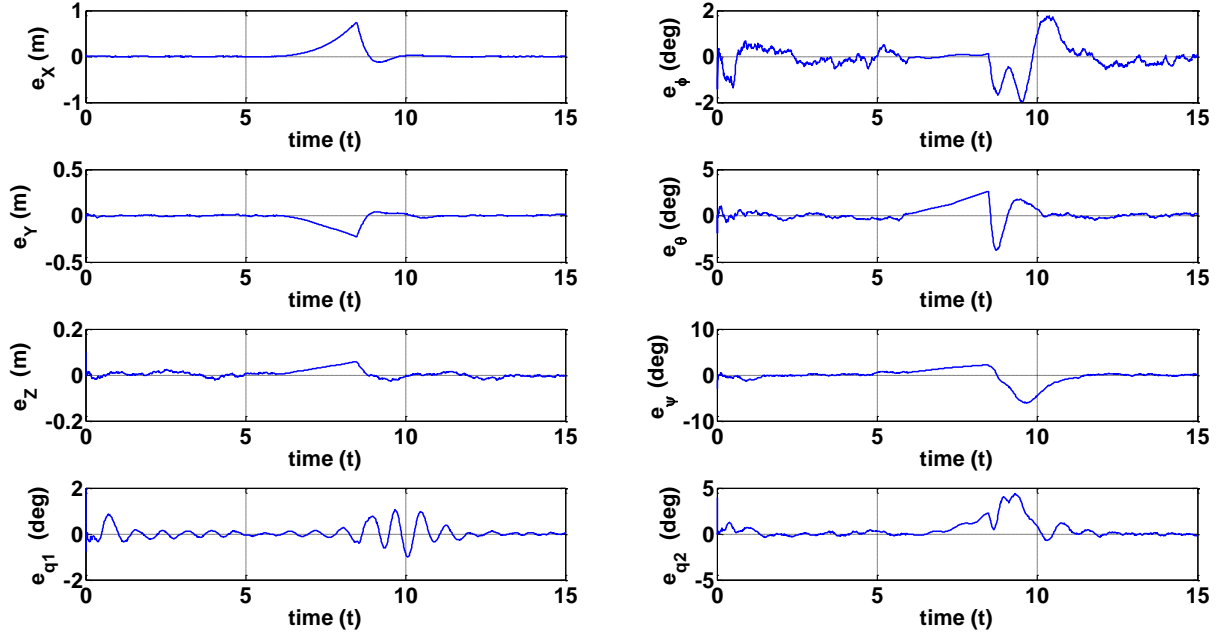Fig. 4.16 SSUKF–based state estimation error of a case study MUAV in loss of sensory data scenario
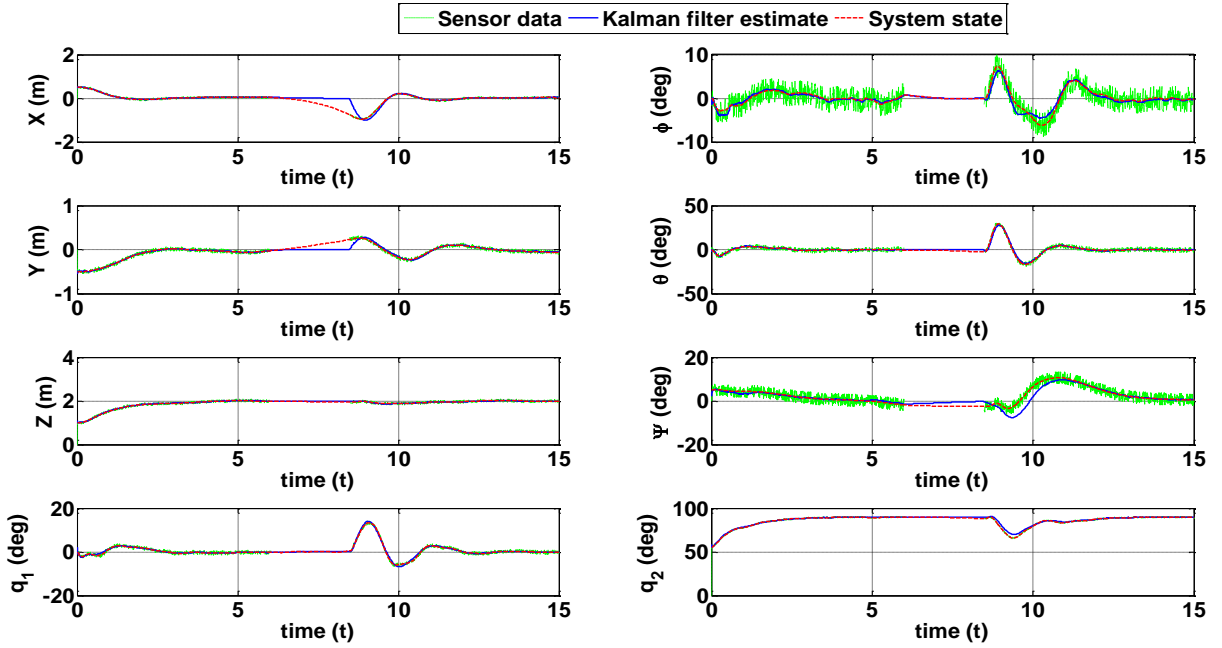

Fig. 4.17 SSUKF–based control of a case study MUAV in loss of sensory data scenario

111

Table 4.13 Estimation error mean and standard deviation in loss of sensory data scenario

		e_X (cm)	e_Y (cm)	e_Z (cm)	e_φ (deg)	e_θ (deg)	e_ψ (deg)	e_{q1} (deg)	e_{q2} (deg)
GUKF	*Mean*	1.8	3.8	1.7	0.5	0.2	0.5	0.1	0.4
	STD	3.6	9.0	2.6	0.6	0.3	0.8	0.2	0.6
SSUKF	*Mean*	5.6	2.4	1.2	0.4	0.5	0.9	0.2	0.6
	STD	13.8	4.9	1.6	0.6	0.9	1.6	0.3	1.0
LPF		N/A							

Table 4.14 RMSE metric in loss of sensory data scenario

	$RMSE_X$ (cm)	$RMSE_Y$ (cm)	$RMSE_Z$ (cm)	$RMSE_\varphi$ (deg)	$RMSE_\theta$ (deg)	$RMSE_\psi$ (deg)	$RMSE_{q1}$ (deg)	$RMSE_{q2}$ (deg)
GUKF	11.8	19.7	23.1	2.8	1.7	2.2	0.9	6.5
SSUKF	22.8	16.0	22.8	1.9	5.4	3.8	3.0	7.5
LPF	N/A							

From Figs. 4.14–4.17 it can be readily seen that no sensory data are available to remedy state estimation for a period of 2.5 s, beginning at time 6 s. During the mentioned period, state estimates begin to excessively diverge from the actual system states, resulting in increasing estimation errors observed in Fig. 4.14 and Fig. 4.16. The excessive errors in state estimation degrade the overall control performance. After the sensor outage period, sensory data becomes available and the objective of this simulation is to verify whether LPF and UKF–based approaches can regain reliable state estimation and control of the MUAV. Our simulation results show that basic LPF approach leads to complete loss of MUAV control in this scenario which conforms to the observation that current MUAV systems are equipped with redundant sensor suites to avoid MUAV control loss in case of sensory data outage. On the other hand, both UKF–based approaches successfully retrieve proper estimation and control of the MUAV, albeit with different performance metrics. While GUKF generally outperforms SSUKF in most channels, its performance is strongly superior to SSUKF in X, θ and q_1 channels. From Table 4.13, in X channel, GUKF state estimation error is better than 2 cm

whereas SSUKF results in more than 5 cm estimation error, with maximum estimation error reaching more than 70 cm. This can also be observed in overall control performance where RMSE of SSUKF is almost two times worse than GUKF. Similarly, in θ channel, SSUKF estimation error and overall control performance metrics are more than 2 times worse than GUKF. This is best shown in Fig. 4.17 where SSUKF results in large–amplitude oscillation before retrieving proper control of θ channel. Finally for q_1 channel, GUKF and SSUKF result in mean estimation errors of 0.1 and 0.2 degrees, respectively. The increase in estimation error is proportionally extended to overall control performance where SSUKF results in relatively large oscillations reaching approximately 15 degrees after sensory data becomes available.

In general, it is concluded that unlike LPF, UKF–based approaches are much less sensitive to sensory data loss and can retrieve MUAV control even in scenarios with full sensor outage for time periods as long as 2.5 s. Also, simulation results show that GUKF outperforms SSUKF in most channels, both in terms of estimation accuracy and overall control performance. Finally, in the next subsection, the performance of our proposed approaches in a trajectory tracking scenario is investigated.

4.5.3 Trajectory tracking scenario

In this section, a helix trajectory tracking scenario is developed. For that purpose, MUAV initial conditions are assumed to be identical to those in the previous subsections, i.e. $[0.5 \text{ m} \quad -0.5 \text{ m} \quad 1 \text{ m} \quad 0 \text{ deg} \quad 0 \text{ deg} \quad 5 \text{ deg} \quad 0 \quad 55 \text{ deg} \quad 0_{1\times8}]^T$ and the reference helical trajectory is defined by $X_H = \cos(0.3\pi t), \;\; Y_H = \sin(0.3\pi t), \;\; Z_H = 2 + 0.1t$. For the described case study, simulations were carried out and results are shown in Figs. 4.18–4.26 and Table 4.15 and Table 4.16.
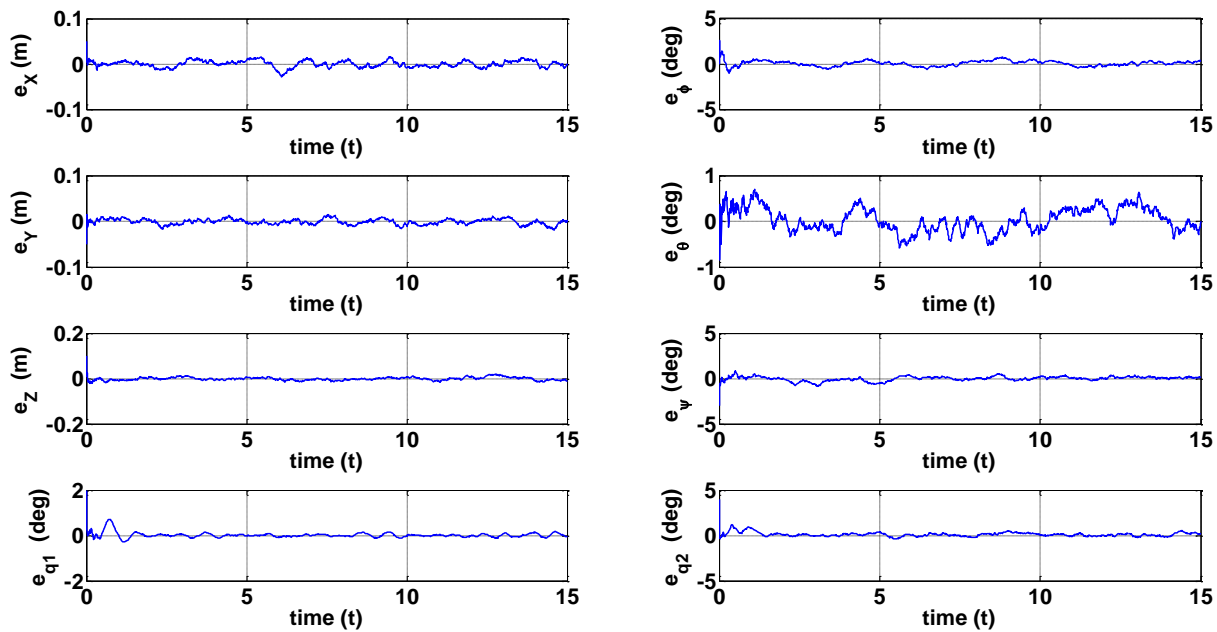
Fig. 4.18 GUKF–based state estimation error in helical trajectory tracking scenario
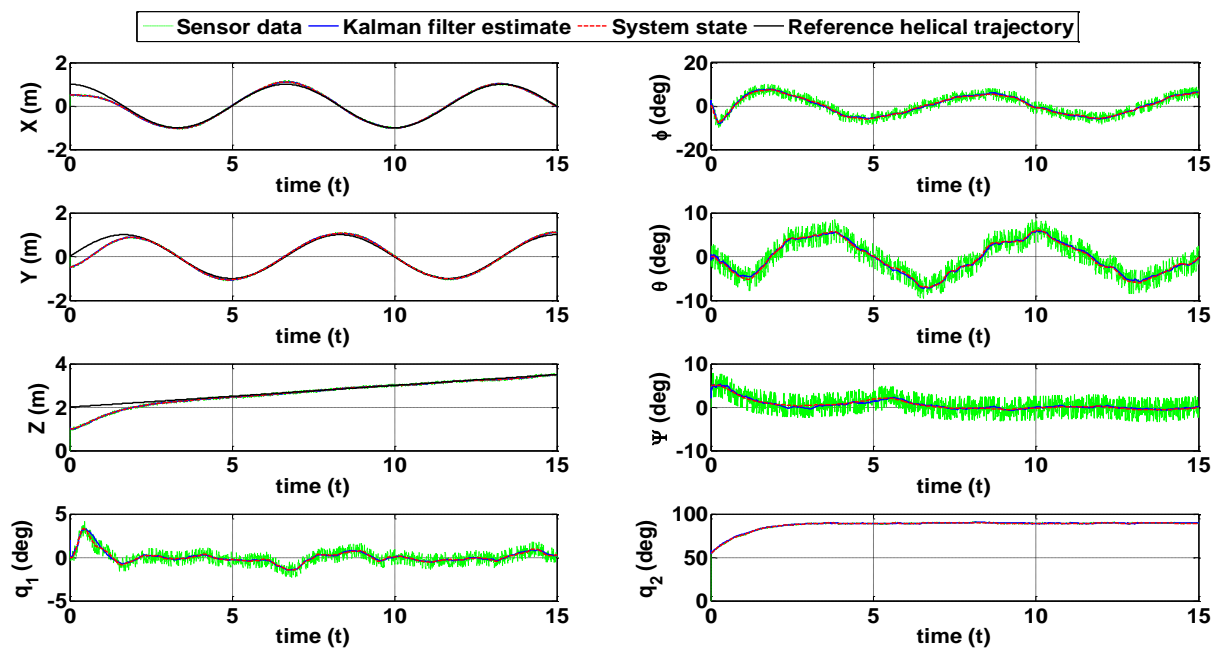
Fig. 4.19 GUKF–based MUAV control in helical trajectory tracking scenario
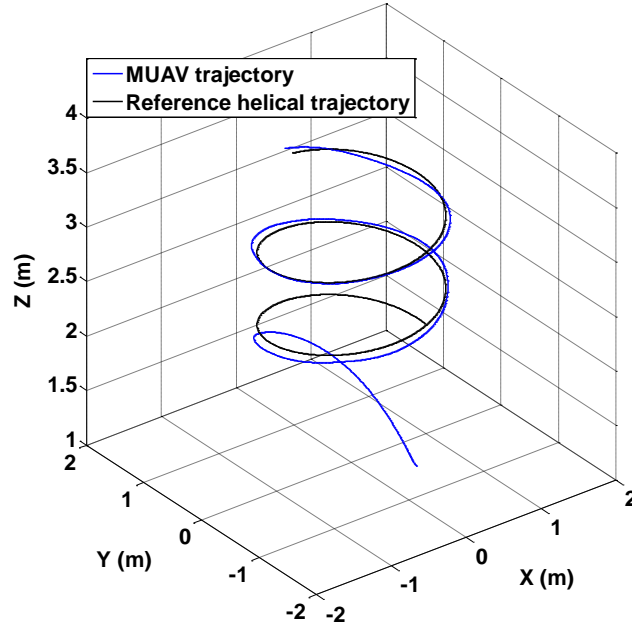
Fig. 4.20 GUKF–based MUAV helical trajectory tracking scenario
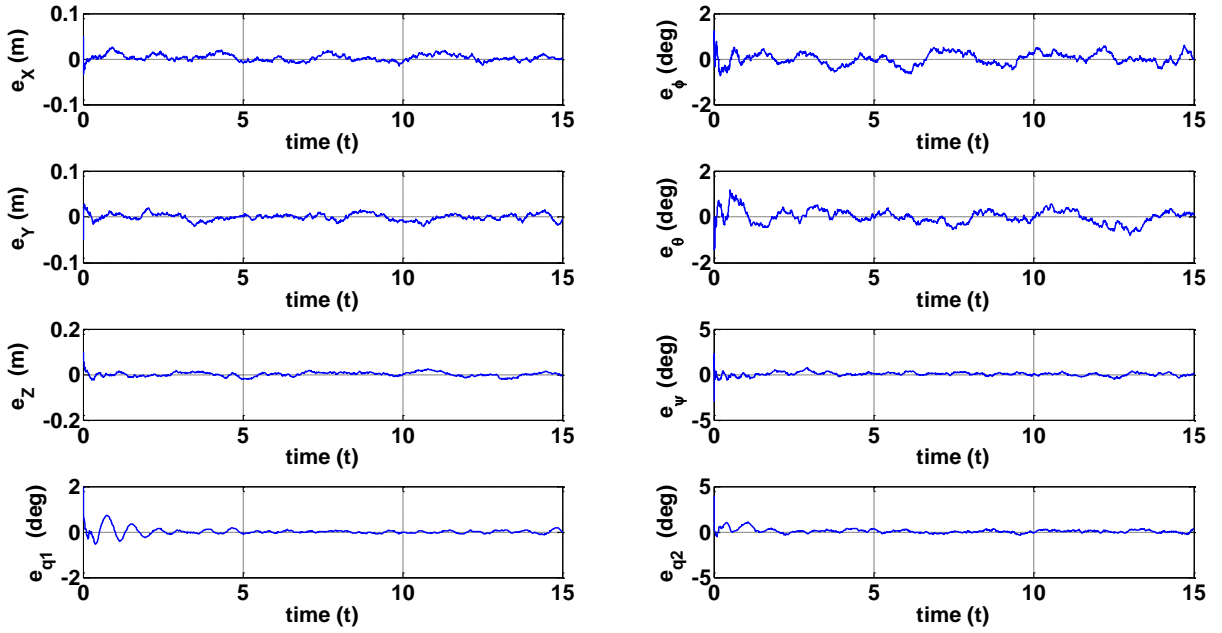
Fig. 4.21 SSUKF–based state estimation error in helical trajectory tracking scenario
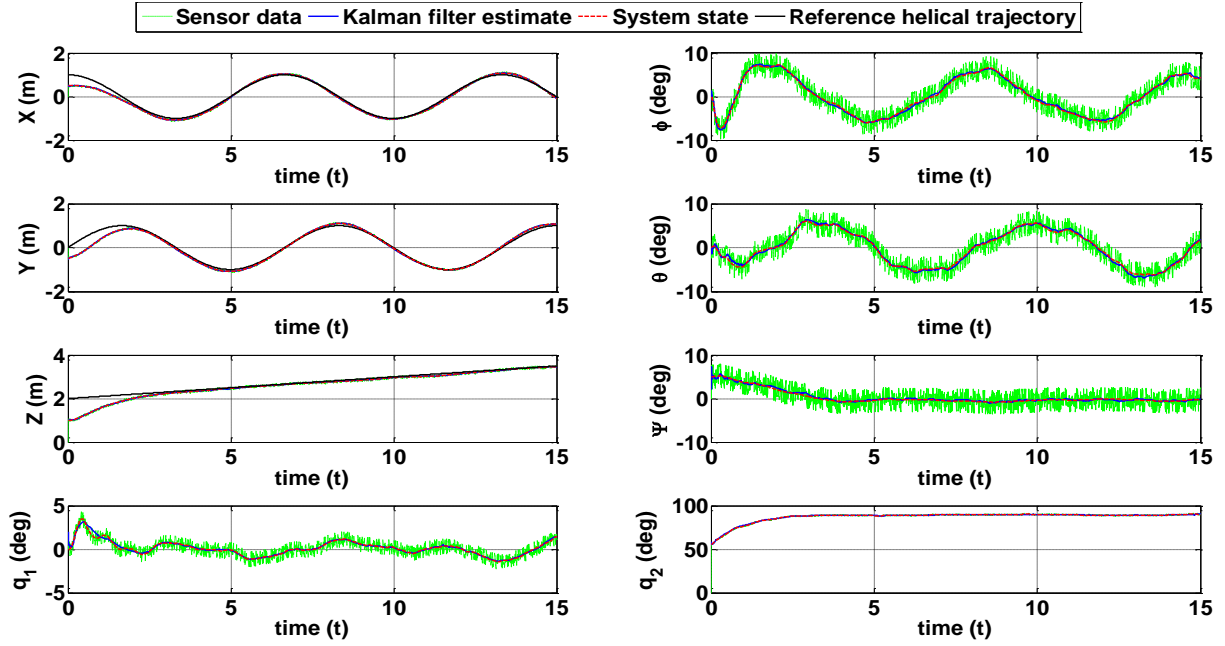
Fig. 4.22 SSUKF–based MUAV control in helical trajectory tracking scenario
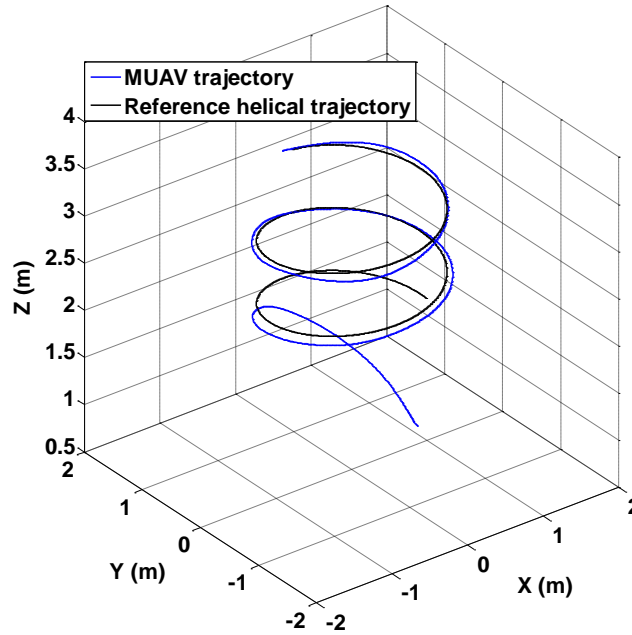
Fig. 4.23 SSUKF–based MUAV helical trajectory tracking scenario
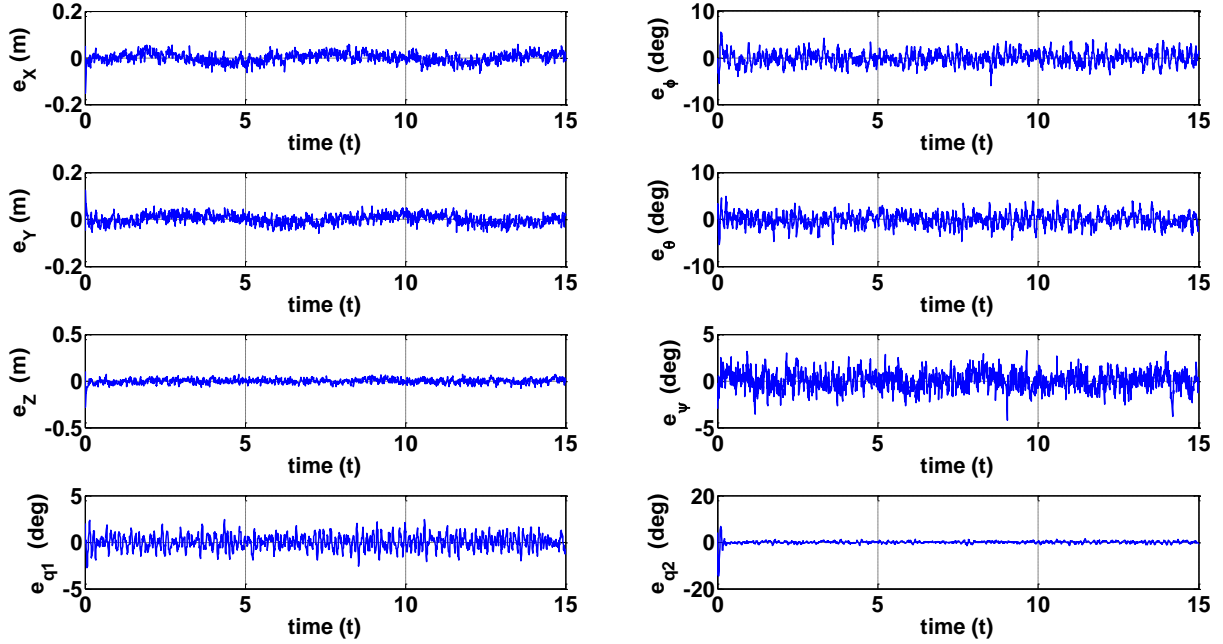
Fig. 4.24 LPF–based state estimation error in helical trajectory tracking scenario
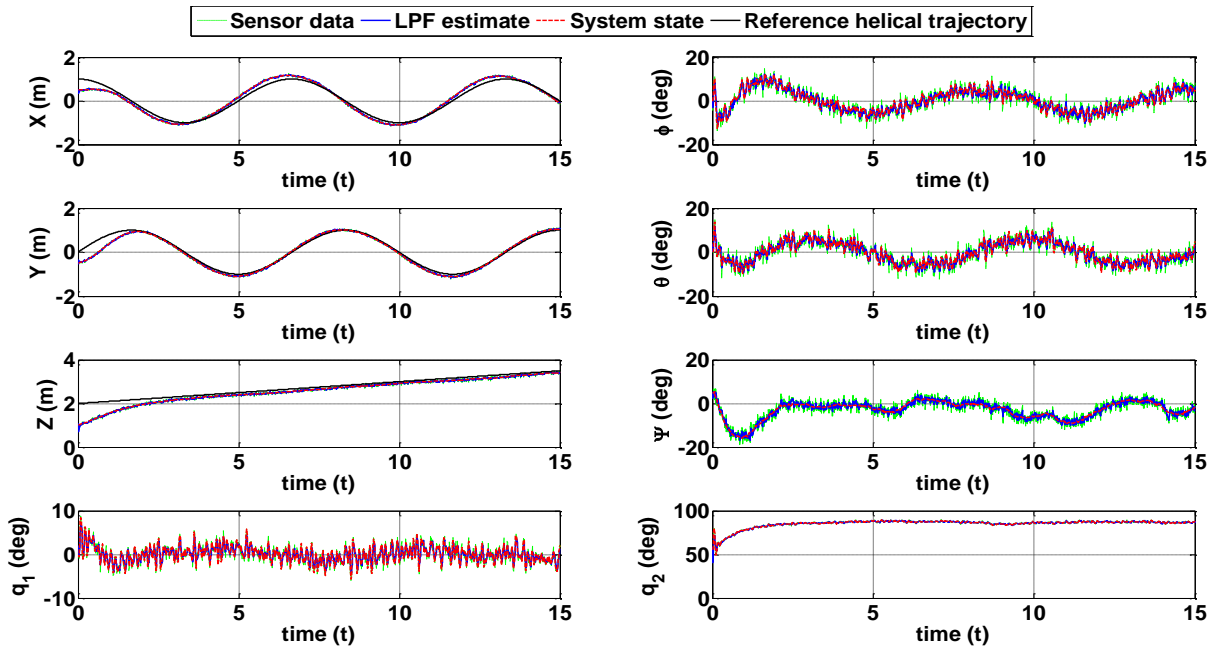
Fig. 4.25 LPF–based MUAV control in helical trajectory tracking scenario
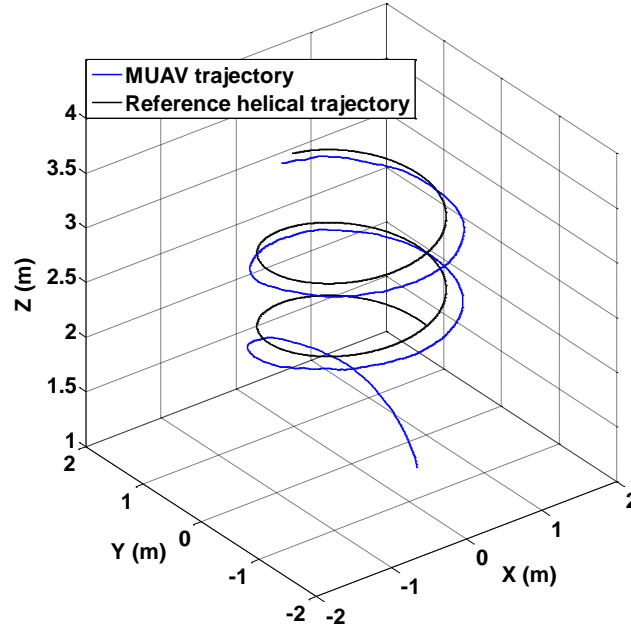
Fig. 4.26 LPF–based MUAV helical trajectory tracking scenario

Table 4.15 Estimation error mean and standard deviation in helical trajectory tracking scenario

		e_X (cm)	e_Y (cm)	e_Z (cm)	e_φ (deg)	e_θ (deg)	e_ψ (deg)	e_{q1} (deg)	e_{q2} (deg)
GUKF	*Mean*	0.6	0.5	0.6	0.2	0.2	0.2	0.1	0.2
	STD	0.7	0.6	0.7	0.3	0.3	0.3	0.1	0.2
SSUKF	*Mean*	0.6	0.6	0.8	0.2	0.2	0.2	0.1	0.2
	STD	0.7	0.7	1.0	0.3	0.3	0.2	0.2	0.3
LPF	*Mean*	1.6	1.6	2.0	1.1	1.1	0.8	0.6	0.4
	STD	2.0	2.0	2.6	1.3	1.4	1.0	0.8	0.7

Table 4.16 RMSE metric in helical trajectory tracking scenario

	$RMSE_X$ (cm)	$RMSE_Y$ (cm)	$RMSE_Z$ (cm)	$RMSE_\varphi$ (deg)	$RMSE_\theta$ (deg)	$RMSE_\psi$ (deg)	$RMSE_{q1}$ (deg)	$RMSE_{q2}$ (deg)
GUKF	11.6	16.5	23.7	4.2	3.7	1.3	0.8	6.5
SSUKF	12.8	16.9	23.7	4.2	3.8	1.5	0.9	6.5
LPF	15.5	17.0	25.3	4.7	4.4	5.1	1.6	7.0

The obtained results show that both GUKF and SSUKF well maintain their estimation accuracy in the helical trajectory tracking scenario. It can also be observed that LPF estimation accuracy degrades more noticeably, in particular in X and Y channels. Regarding the overall control performance, while both UKF–based algorithms and LPF–based approach have successfully tracked the helical trajectory, their performances vary. First of all, it can be observed that non–zero error between MUAV trajectory, blue line in Fig. 4.20, 4.23 and 4.26, and the reference helix, black line in the aforementioned figures, persists in trajectory tracking of both UKF–based and LPF–based approaches. This is a well–known drawback of LQR and similar linear control approaches, e.g. PID control, and has been reported in the existing literature [192], [197], [198]. Yet, the overall RMSE metrics reported in Table 4.16 also depend on state estimation accuracy provided to feedback–based LQR control laws. In particular, accurate state estimation of GUKF algorithm has resulted in more accurate trajectory tracking, compared to SSUKF, as reported in Table 4.16. Less accurate state estimation of LPF–based approach has in turn resulted in large RMSE metrics, compared to UKF–based algorithms, in translational, rotational and also joint angle channels. It is concluded that our proposed UKF–based approaches maintain accurate state estimation and, along with LQR control laws, can successfully accomplish a trajectory tracking scenario.

4.6 Summary

Two state estimation schemes, i.e. General Unscented Kalman Filter and Scaled Spherical Unscented Kalman Filter, were formulated for manipulating unmanned aerial vehicles. A basic LPF approach was also formulated to serve as a basis for comparison purposes. It was shown that both UKF–based algorithms and LPF–based approach, along with LQR controller, can successfully accomplish a setpoint tracking scenario starting from various initial conditions. What is more, in scenarios with low noise level, obtained results show that SSUKF performance is comparable to GUKF while LPF provides mediocre estimation and overall control performance. Increasing the noise level, LPF soon fails to properly control the MUAV while UKF–based approaches result in degraded estimation and control performance, without abrupt control loss. Compared to SSUKF, GUKF was found to perform better for moderate increase in noise level and also experiences graceful degradation in scenarios with large noise level, where SSUKF fails to maintain stable MUAV control. Finally, for scenarios with total sensory data loss, LPF fails to

maintain MUAV control where GUKF again outperforms SSUKF in terms of estimation and overall control performance. For a helical trajectory tracking scenario, both UKF–based algorithms outperformed LPF–based approach in terms of estimation and overall control performance where, in general, GUKF resulted in more accurate results compared to SSUKF–based algorithm. However, given that SSUKF outperformed GUKF in terms of execution time, the ultimate choice for MUAV state estimation algorithm should be based on required estimation and overall control performance, while also recognizing the computational requirements of each algorithm.

Chapter 5 Summary and Conclusions

5.1 Summary

This dissertation was mainly focused on state estimation and control of manipulating unmanned aerial vehicles.

In Chapter 2, a comprehensive literature survey on MUAVs including physical subsystems, sensory configurations, missions and operational scenarios, dynamic modelling, estimation and control problems was presented.

In Chapter 3, a framework was presented where effects of robotic manipulator on UAV dynamics was modelled as additive process noise with unknown noise statistics. For that purpose, dynamic equations of motion of a quadcopter were presented and LQR control laws were designed to achieve stable flight. Formulating extended and unscented Kalman filters (EKF and UKF, respectively), it was shown that both algorithms result in accurate estimation and overall control of a quadcopter starting from various initial conditions. Given poor performance of EKF and UKF algorithms for systems with uncertain process/measurement noise statistics, adaptive extended and unscented Kalman filters (AEKF and AUKF, respectively) were formulated for UAV state estimation. Based on covariance–matching recursive AEKF and AUKF formulation along with LQR control laws, UAV state estimation and control with uncertain process and measurement noise statistics in various covariance mismatch scenarios was achieved.

In order to improve the performance of previous approach and achieve simultaneous control of a UAV and its robotic manipulator, full dynamic model of a MUAV was presented in Chapter 4. For a MUAV consisting of a quadcopter and a 2–DoF robotic manipulator, Euler–Lagrange formulation was used to obtain MUAV dynamic equations of motion. Then, a general unscented Kalman filter (GUKF) was formulated to achieve MUAV state estimation along with LQR control laws, starting from various initial conditions. Given computational complexity of GUKF, a scaled spherical transform was considered to improve execution time based on reduced number of sigma points, compared to GUKF. To that end, a scaled spherical unscented Kalman filter (SSUKF) was formulated to achieve mean estimation error and overall control performance

comparable to GUKF, albeit with reduced execution time. Sensitivity of both algorithms in scenarios with increased noise levels and periods of total loss of sensory data, and also a trajectory tracking scenario were studied.

5.2 Conclusions

The main conclusions of this dissertation are:

1. It was shown that EKF and UKF both result in accurate state estimation of a quadcopter UAV with subcentimeter mean estimation error in translational channels and subdegree mean estimation error in rotational channels. In general, it was found that UKF outperforms EKF in mean estimation error and overall control performance. Yet, EKF offers an order–of–magnitude improvement in execution time, compared to UKF. Therefore, the ultimate selection of state estimation algorithm for a quadcopter UAV is a compromise between the required estimation and control performance metrics and available onboard computational resources.

2. For a quadcopter with uncertain noise statistics, it was shown that EKF and UKF result in excessively large estimation errors and very poor control performance. Covariance–matching adaptive filtering was used to improve conventional Kalman filters for a UAV with uncertain process and measurement noise statistics. It was shown AEKF and AUKF result in accurate estimation and overall control of a quadcopter with uncertain process and measurement noise statistics. Both adaptive filters failed as simultaneous mismatch in process and measurement noise increased to several orders of magnitude. In general, AUKF performance was found to be better than AEKF both in estimation and overall control.

3. General UKF was found to be a practical solution for MUAV state estimation. It was shown that subcentimeter estimation of translational channels and subdegree estimation of rotational channels of a MUAV with noisy sensory data was achievable. A modified UKF algorithm based on scaled transform was formulated to achieve performance comparable with GUKF, yet at approximately 35%

improvement in execution time. In cases of increasing noise level, GUKF was found to be less sensitive as it degraded gracefully in various noise scenarios. Both algorithms were found to be able to retrieve MUAV control after a period of total sensory data loss. Again, GUKF was found to outperform SSUKF in such scenarios, in terms of estimation error and overall control performance. Finally, GUKF was found to outperform SSUKF in a trajectory tracking scenario as well.

5.3 Contributions

The novel contributions of this dissertation can be summarized as:

1. Extended and Unscented Kalman filters were investigated for state estimation and control of a quadcopter autonomous flight. It was shown that UKF–based approach, in general, outperforms EKF in estimation and control of a quadcopter. For scenarios with uncertain noise statistics, it was shown that performance of conventional Kalman filters degrades severely, or in extreme cases, abrupt loss of UAV control occurs. For such scenarios, recursive covariance–matching adaptive Extended and Unscented Kalman filters were formulated for the first time for state estimation and control of a quadcopter autonomous flight. It was shown that AEKF and AUKF can achieve accurate UAV state estimation and control in scenarios with uncertain process and measurement noise statistics.

2. For a MUAV consisting of a quadcopter and a 2–DoF robotic manipulator, it was shown for the first time that UKF–based algorithms can effectively be used to provide accurate state estimation of a MUAV. LQR control based on the estimated states was found to result in satisfactory control of the MUAV starting from various initial conditions.

3. For the described MUAV case study, it was shown that GUKF–based state estimation and its computationally–efficient counterpart, i.e. SSUKF, can be carried out in real–time in MATLAB with an Intel® Core™–i5 3.2 GHz processor based computer with 8.00 GB RAM. SSUKF was found to outperform GUKF by approximately 35% in terms of execution time.

4. Sensitivity of UKF–based approaches (GUKF and SSUKF) to various noise levels and total loss of sensory data for MUAV state estimation and control was investigated for the first time. It was shown that GUKF degraded gracefully in such scenarios whereas SSUKF was more sensitive to noise levels and sensory data outage. Also, it as shown that both GUKF and SSUKF state estimation, along with LQR, can be used for trajectory tracking scenarios, where GUKF was found to outperform SSUKF–based results.

5.4 Recommendations for Future work

The research presented in this dissertation has the potential to be extended in the following aspects:

I. The UKF–based state estimation algorithms in Chapter 3 rely on GPS and attitude heading and reference system for sensory information. One possible improvement is to equip the quadcopter with an onboard camera which, in turn, will provide position and orientation angles sensory information and can be used as the only required onboard sensor for autonomous quadcopter missions.

II. The adaptive state estimation algorithm in Chapter 3 can be readily extended to MUAVs with state estimation and control schemes described in Chapter 4. This will enable autonomous operation of a MUAV even with uncertainties in process and/or measurement noise statistics, where even state of the art non–adaptive UKF–based algorithms lead to abrupt loss of MUAV control.

III. The MUAV dynamic model in Chapter 4 does not take vibration/noise caused by propellers and manipulator servos into account. Explicitly formulating the mentioned effects in MUAV dynamic modelling will lead to more accurate representation of the systems.

IV. The low–level state estimation and control algorithms in Chapter 4 can be implemented along with a high–level path planning scheme to provide autonomous operation of a MUAV. This will enable more sophisticated missions where a MUAV is required to navigate through known/unknown environment to localize objects of interest and then interact with them.

V. The proposed state estimation algorithms in Chapter 4 can be readily augmented with other control techniques such as MPC, robust and/or adaptive control. This is promising because MUAV mission can be optimized to meet certain mission requirements, provide robustness to unmodelled dynamics, and adapt to certain changes in the dynamics and/or environment.

VI. Wind effect in outdoor environment is expected to be an important source of disturbance to MUAVs. Two possible solutions are suggested. First, it is possible to measure wind speed by means of onboard sensors and augment it with MUAV

dynamic model. Second, it is also possible to formulate a parameter estimation problem to obtain an online estimation of wind speed based on available sensory information.

VII. Inclusion of force feedback can open the door to investigate 6D force estimation techniques and hybrid force–position control for realizing compliant MUAV motions.

VIII. Teleoperation issues have not been well studied in the context of MUAVs. Issues in teleoperation (e.g., imperfect communication, time–varying delay and information losses) and design of robust controllers in such scenarios could be another interesting extension for this research.

IX. This research has provided the necessary elements for autonomous operation of a MUAV in real–life scenarios, including uncertainties in MUAV dynamics and sensory measurements. Experimental validation of the proposed algorithms is a promising extension of this research as it will result in a MUAV that can operate solely based on its onboard sensory data.

References

[1]   "FAA Nightmare: A Million Christmas Drones," *2016*. [Online]. Available: http://aviationweek.com/commercial-aviation/faa-nightmare-million-christmas-drones. [Accessed: 04-Apr-2018].

[2]   F. G. Costa, J. Ueyama, T. Braun, G. Pessin, F. S. Osório, and P. A. Vargas, "The use of unmanned aerial vehicles and wireless sensor network in agricultural applications," in *Proc. IEEE International Geoscience and Remote Sensing Symposium (IGARSS),* Munich, Germany, 2012, pp. 5045–5048.

[3]   D. W. Casbeer, D. B. Kingston, R. W. Beard, and T. W. McLain, "Cooperative forest fire surveillance using a team of small unmanned air vehicles," *International Journal of Systems Science*, vol. 37, no. 6, pp. 351–360, 2006.

[4]   H. A. F. Almurib, P. T. Nathan, and T. N. Kumar, "Control and path planning of quadrotor aerial vehicles for search and rescue," in *Proc. SICE Annual Conference,* Tokyo, Japan, 2011, pp. 700–705.

[5]   R. W. Beard, T. W. McLain, D. B. Nelson, D. Kingston, and D. Johanson, "Decentralized cooperative aerial surveillance using fixed-wing miniature UAVs," *Proceedings of the IEEE*, vol. 94, no. 7, pp. 1306–1324, 2006.

[6]   H. Li, B. Wang, L. Liu, G. Tian, T. Zheng, and J. Zhang, "The design and application of SmartCopter: An unmanned helicopter based robot for transmission line inspection," in *Proc. Chinese Automation Congress (CAC),* Changsha, China, 2013, pp. 697–702.

[7]   M. T. Connolly, "The Use of Multi Rotor Remotely Operated Aerial Vehicles (ROAVs) as a Method of Close Visually Inspecting (CVI) Live and Difficult to Access Assets on Offshore Platforms," in *Abu Dhabi International Petroleum Exhibition and Conference,* Abu Dhabi, UAE, 2014.

[8]   A. Shukla and H. Karki, "Application of robotics in onshore oil and gas industry—A review Part I," *Robotics and Autonomous Systems*, vol. 75, pp. 490–507, 2016.

[9]   "BP Alaska: Unmanned Aerial Vehicle (UAV) Pilot Testing," 2012. [Online]. Available: https://www.youtube.com/watch?v=UO0rgiS3wgw. [Accessed: 04-Apr-2018].

[10]   N. Michael, J. Fink, and V. Kumar, "Cooperative manipulation and transportation with aerial robots," *Autonomous Robots*, vol. 30, no. 1, pp. 73–86, 2011.

[11]   D. Mellinger, Q. Lindsey, M. Shomin, and V. Kumar, "Design, modeling, estimation and

control for aerial grasping and manipulation," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS),* San Francisco, CA, 2011, pp. 2668–2673.

[12]   J. Thomas, G. Loianno, K. Sreenath, and V. Kumar, "Toward image based visual servoing for aerial grasping and perching," in *Proc. IEEE International Conference on Robotics and Automation (ICRA),* Hong Kong, 2014, pp. 2113–2118.

[13]   G. Gioioso, M. Ryll, D. Prattichizzo, H. H. Bülthoff, and A. Franchi, "Turning a near-hovering controlled quadrotor into a 3D force effector," in *Proc. IEEE International Conference on Robotics and Automation (ICRA),* Hong Kong, 2014, pp. 6278–6284.

[14]   S. Kim, S. Choi, and H. J. Kim, "Aerial manipulation using a quadrotor with a two dof robotic arm," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS),* Tokyo, Japan, 2013, pp. 4990–4995.

[15]   M. Orsag, C. M. Korpela, S. Bogdan, and P. Y. Oh, "Hybrid adaptive control for aerial manipulation," *Journal of intelligent & robotic systems*, vol. 73, no. 1–4, p. 693, 2014.

[16]   P. E. I. Pounds, D. R. Bersak, and A. M. Dollar, "Grasping from the air: Hovering capture and load stability," in *Proc. IEEE International Conference on Robotics and Automation (ICRA),* Shanghai, China, 2011, pp. 2491–2498.

[17]   Q. Lindsey, D. Mellinger, and V. Kumar, "Construction of cubic structures with quadrotor teams," *Proc. Robotics: Science & Systems VII*, 2011.

[18]   J. Fink, N. Michael, S. Kim, and V. Kumar, "Planning and control for cooperative manipulation and transportation with aerial robots," *The International Journal of Robotics Research*, vol. 30, no. 3, pp. 324–334, 2011.

[19]   K. Baizid *et al.*, "Experiments on behavioral coordinated control of an unmanned aerial vehicle manipulator system," in *Proc. IEEE International Conference on Robotics and Automation (ICRA),* Seattle, WA, 2015, pp. 4680–4685.

[20]   G. Garimella and M. Kobilarov, "Towards model-predictive control for aerial pick-and-place," in *Proc. IEEE International Conference on Robotics and Automation (ICRA),* Seattle, WA, 2015, pp. 4692–4697.

[21]   C. Korpela, M. Orsag, and P. Oh, "Towards valve turning using a dual-arm aerial manipulator," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS),* Chicago, IL, 2014, pp. 3411–3416.

[22] M. Orsag, C. Korpela, S. Bogdan, and P. Oh, "Valve turning using a dual-arm aerial manipulator," in *Proc. International Conference on Unmanned Aircraft Systems (ICUAS)*, Orlando, FL, 2014, pp. 836–841.

[23] M. Orsag, C. Korpela, M. Pekala, and P. Oh, "Stability control in aerial manipulation," in *Proc. American Control Conference (ACC)*, Washington, DC, 2013, pp. 5581–5586.

[24] D. I. Montufar, F. Munoz, E. S. Espinoza, O. Garcia, and S. Salazar, "Multi-UAV testbed for aerial manipulation applications," in *Proc. International Conference on Unmanned Aircraft Systems (ICUAS)*, Orlando, FL, 2014, pp. 830–835.

[25] V. Lippiello *et al.*, "Hybrid visual servoing with hierarchical task composition for aerial manipulation," *IEEE Robotics and Automation Letters*, vol. 1, no. 1, pp. 259–266, 2016.

[26] G. Heredia *et al.*, "Control of a multirotor outdoor aerial manipulator," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Chicago, IL, 2014, pp. 3417–3422.

[27] J. R. Kutia, K. A. Stol, and W. Xu, "Initial flight experiments of a canopy sampling aerial manipulator," in *Proc. International Conference on Unmanned Aircraft Systems (ICUAS)*, Arlington, VA, 2016, pp. 1359–1365.

[28] C. M. Korpela, T. W. Danko, and P. Y. Oh, "Designing a system for mobile manipulation from an unmanned aerial vehicle," in *Proc. IEEE Conference on Technologies for Practical Robot Applications (TePRA)*, Woburn, MA, 2011, pp. 109–114.

[29] C. Korpela *et al.*, "Flight stability in aerial redundant manipulators," in *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, Saint Paul, MN, 2012, pp. 3529–3530.

[30] P. Pounds and R. Mahony, "Design principles of large quadrotors for practical applications," in *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, Kobe, Japan, 2009, pp. 3265–3270.

[31] F. Huber *et al.*, "First analysis and experiments in aerial manipulation using fully actuated redundant robot arm," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Tokyo, Japan, 2013, pp. 3452–3457.

[32] M. Laiacker, F. Huber, and K. Kondak, "High accuracy visual servoing for aerial manipulation using a 7 degrees of freedom industrial manipulator," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Daejeon, South Korea, 2016, pp. 1631–1636.

[33] A. E. Jimenez-Cano, J. Martin, G. Heredia, A. Ollero, and R. Cano, "Control of an aerial robot with multi-link arm for assembly tasks," in *Proc. IEEE International Conference on Robotics and Automation (ICRA),* Karlsruhe, Germany, 2013, pp. 4916–4921.

[34] G. Arleo, F. Caccavale, G. Muscio, and F. Pierri, "Control of quadrotor aerial vehicles equipped with a robotic arm," in *Proc. 21st Mediterranean Conference on Control & Automation,* Chania, Greece, 2013, pp. 1174–1180.

[35] B. Yang, Y. He, J. Han, and G. Liu, "Rotor-flying manipulator: modeling, analysis, and control," *Mathematical Problems in Engineering*, vol. 2014, 2014.

[36] S. Kannan, M. A. Olivares-Mendez, and H. Voos, "Modeling and control of aerial manipulation vehicle with visual sensor," *IFAC Proceedings Volumes*, vol. 46, no. 30, pp. 303–309, 2013.

[37] R. Cano, C. Pérez, F. Pruano, A. Ollero, and G. Heredia, "Mechanical design of a 6-DOF aerial manipulator for assembling bar structures using UAVs," in *Proc. 2nd RED-UAS Workshop on Research, Education and Development of Unmanned Aerial Systems,* Compiegne, France, 2013.

[38] C. D. Bellicoso, L. R. Buonocore, V. Lippiello, and B. Siciliano, "Design, modeling and control of a 5-DoF light-weight robot arm for aerial manipulation," in *23rd Mediterranean Conference on Control and Automation,* Torremolinos, Spain, 2015, pp. 853–858.

[39] T. W. Danko and P. Y. Oh, "A hyper-redundant manipulator for mobile manipulating unmanned aerial vehicles," in *Proc. International Conference on Unmanned Aircraft Systems (ICUAS),* Atlanta, GA, 2013, pp. 974–981.

[40] A. Suarez, G. Heredia, and A. Ollero, "Lightweight compliant arm for aerial manipulation," in *proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS),* Hamburg, Germany, 2015, pp. 1627–1632.

[41] S. B. Backus, L. U. Odhner, and A. M. Dollar, "Design of hands for aerial manipulation: actuator number and routing for grasping and perching," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS),* Chicago, IL, 2014, pp. 34–40.

[42] T. W. Danko, K. P. Chaney, and P. Y. Oh, "A parallel manipulator for mobile manipulating UAVs," in *Proc. IEEE International Conference on Technologies for Practical Robot Applications (TePRA),* Woburn, MA, 2015, pp. 1–6.

[43] R. Rossi, A. Santamaria-Navarro, J. Andrade-Cetto, and P. Rocco, "Trajectory Generation for Unmanned Aerial Manipulators Through Quadratic Programming," *IEEE Robotics*

and Automation Letters, vol. 2, no. 2, pp. 389–396, 2017.

[44] T. W. Danko and P. Y. Oh, "Design and control of a hyper-redundant manipulator for mobile manipulating unmanned aerial vehicles," *Journal of Intelligent & Robotic Systems*, vol. 73, no. 1–4, p. 709, 2014.

[45] R. Naldi, P. Pounds, S. De Marco, and L. Marconi, "Output tracking for quadrotor-based aerial manipulators," in *Proc. American Control Conference (ACC),* Chicago, IL, 2015, pp. 1855–1860.

[46] A. Y. Mersha, S. Stramigioli, and R. Carloni, "Exploiting the dynamics of a robotic manipulator for control of UAVs," in *Proc. IEEE International Conference on Robotics and Automation (ICRA),* Hong Kong, 2014, pp. 1741–1746.

[47] F. Forte, R. Naldi, A. Macchelli, and L. Marconi, "Impedance control of an aerial manipulator," in *Proc. American Control Conference (ACC),* Montreal, QC, 2012, pp. 3839–3844.

[48] V. Ghadiok, J. Goldin, and W. Ren, "Autonomous indoor aerial gripping using a quadrotor," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS),* San Francisco, CA, 2011, pp. 4645–4651.

[49] D. Mellinger, M. Shomin, and V. Kumar, "Control of quadrotors for robust perching and landing," in *Proc. International Powered Lift Conference,* Philadelphia, PA, 2010, pp. 205–225.

[50] P. Pounds and A. Dollar, "Hovering stability of helicopters with elastic constraints," in *Proc. ASME Dynamic systems and control conference,* Cambridge, Massachusetts, 2010, pp. 781–788.

[51] L. Gentili, R. Naldi, and L. Marconi, "Modeling and control of VTOL UAVs interacting with the environment," in *Proc. 47th IEEE Conference on Decision and Control (CDC),* Cancun, Mexico, 2008, pp. 1231–1236.

[52] S. Bellens, J. De Schutter, and H. Bruyninckx, "A hybrid pose/wrench control framework for quadrotor helicopters," in *Proc. IEEE International Conference on Robotics and Automation (ICRA),* Saint Paul, MN, 2012, pp. 2269–2274.

[53] A. Y. Mersha, S. Stramigioli, and R. Carloni, "Variable impedance control for aerial interaction," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS),* Chicago, IL, 2014, pp. 3435–3440.

[54] K. Alexis, G. Darivianakis, M. Burri, and R. Siegwart, "Aerial robotic contact-based inspection: planning and control," *Autonomous Robots*, vol. 40, no. 4, pp. 631–655, 2016.

[55] H. Tsukagoshi, M. Watanabe, T. Hamada, D. Ashlih, and R. Iizuka, "Aerial manipulator with perching and door-opening capability," in *Proc. IEEE International Conference on Robotics and Automation (ICRA),* Seattle, WA, 2015, pp. 4663–4668.

[56] H.-N. Nguyen and D. Lee, "Hybrid force/motion control and internal dynamics of quadrotors for tool operation," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS),* Tokyo, Japan, 2013, pp. 3458–3464.

[57] C. Papachristos, K. Alexis, and A. Tzes, "Efficient force exertion for aerial robotic manipulation: Exploiting the thrust-vectoring authority of a tri-tiltrotor uav," in *Proc. IEEE International Conference on Robotics and Automation (ICRA),* Hong Kong, 2014, pp. 4500–4505.

[58] M. Fumagalli *et al.*, "Developing an aerial manipulator prototype: Physical interaction with the environment," *IEEE robotics & automation magazine*, vol. 21, no. 3, pp. 41–50, 2014.

[59] H.-N. Nguyen, C. Ha, and D. Lee, "Mechanics, control and internal dynamics of quadrotor tool operation," *Automatica*, vol. 61, pp. 289–301, 2015.

[60] G. Darivianakis, K. Alexis, M. Burri, and R. Siegwart, "Hybrid predictive control for aerial robotic physical interaction towards inspection operations," in *proc. IEEE International Conference on Robotics and Automation (ICRA),* Hong Kong, 2014, pp. 53–58.

[61] R. Johnson, J. Sasiadek, and J. Zalewski, "Kalman filter enhancement for UAV navigation," *Simulation Series*, vol. 35, no. 1, pp. 267–272, 2003.

[62] N. Sunderhauf, S. Lange, and P. Protzel, "Using the unscented kalman filter in mono-SLAM with inverse depth parametrization for autonomous airship control," in *proc. IEEE International Workshop on Safety, Security and Rescue Robotics,* Rome, Italy, 2007, pp. 1–6.

[63] H. G. De Marina, F. J. Pereda, J. M. Giron-Sierra, and F. Espinosa, "UAV attitude estimation using unscented Kalman filter and TRIAD," *IEEE Transactions on Industrial Electronics*, vol. 59, no. 11, pp. 4465–4474, 2012.

[64] C. Luo, S. I. McClean, G. Parr, L. Teacy, and R. De Nardi, "UAV position estimation and collision avoidance using the extended Kalman filter," *IEEE Transactions on Vehicular*

Technology, vol. 62, no. 6, pp. 2749–2762, 2013.

[65]   K. Kondak *et al.*, "Closed-loop behavior of an autonomous helicopter equipped with a robotic arm for aerial manipulation tasks," *International Journal of Advanced Robotic Systems*, vol. 10, no. 2, p. 145, 2013.

[66]   J. Zhang and H. Yuan, "Analysis of unmanned aerial vehicle navigation and height control system based on GPS," *Journal of Systems Engineering and Electronics*, vol. 21, no. 4, pp. 643–649, 2010.

[67]   J. Ore, S. Elbaum, A. Burgin, and C. Detweiler, "Autonomous aerial water sampling," *Journal of Field Robotics*, vol. 32, no. 8, pp. 1095–1113, 2015.

[68]   A. Albers, S. Trautmann, T. Howard, T. A. Nguyen, M. Frietsch, and C. Sauter, "Semi-autonomous flying robot for physical interaction with environment," in *Proc. IEEE Conference on Robotics Automation and Mechatronics (RAM),* Singapore, 2010, pp. 441–446.

[69]   S. Kim, H. Seo, S. Choi, and H. J. Kim, "Vision-guided aerial manipulation using a multirotor with a robotic arm," *IEEE/ASME Transactions on Mechatronics*, vol. 21, no. 4, pp. 1912–1923, 2016.

[70]   D. Scaramuzza, A. Martinelli, and R. Siegwart, "A toolbox for easily calibrating omnidirectional cameras," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Beijing, China,* 2006, pp. 5695–5701.

[71]   Q. Jiang and V. Kumar, "The inverse kinematics of cooperative transport with multiple aerial robots," *IEEE Transactions on Robotics*, vol. 29, no. 1, pp. 136–145, 2013.

[72]   I. Palunko, P. Cruz, and R. Fierro, "Agile load transportation: Safe and efficient load manipulation with aerial robots," *IEEE Robotics & Automation Magazine*, vol. 19, no. 3, pp. 69–79, 2012.

[73]   C. Wu, J. Qi, D. Song, X. Qi, T. Lin, and J. Han, "Development of an unmanned helicopter automatic barrels transportation system," in *Proc. IEEE International Conference on Robotics and Automation (ICRA),* Seattle, WA, 2015, pp. 4686–4691.

[74]   S. Zhao *et al.*, "A robust real-time vision system for autonomous cargo transfer by an unmanned helicopter," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 2, pp. 1210–1219, 2015.

133

[75] M. Laiacker, M. Schwarzbach, and K. Kondak, "Automatic aerial retrieval of a mobile robot using optical target tracking and localization," in *Proc. IEEE Aerospace Conference,* Big Sky, MT, 2015, pp. 1–7.

[76] N. Staub, M. Mohammadi, D. Bicego, D. Prattichizzo, and A. Franchi, "Towards Robotic MAGMaS: Multiple Aerial-Ground Manipulator Systems," in *Proc. IEEE International Conference on Robotics and Automation (ICRA),* Singapore, 2017, pp. 1307–1312.

[77] M. Bernard and K. Kondak, "Generic slung load transportation system using small size helicopters," in *Proc. IEEE International Conference on Robotics and Automation (ICRA),* Kobe, Japan, 2009, pp. 3258–3264.

[78] M. Bernard, K. Kondak, I. Maza, and A. Ollero, "Autonomous transportation and deployment with aerial robots for search and rescue missions," *Journal of Field Robotics*, vol. 28, no. 6, pp. 914–931, 2011.

[79] M. Fumagalli, R. Naldi, A. Macchelli, R. Carloni, S. Stramigioli, and L. Marconi, "Modeling and control of a flying robot for contact inspection," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS),* Vilamoura, Portugal, 2012, pp. 3532–3537.

[80] A. E. Jimenez-Cano, J. Braga, G. Heredia, and A. Ollero, "Aerial manipulator for structure inspection by contact from the underside," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS),* Hamburg, Germany, 2015, pp. 1879–1884.

[81] S. Kim, H. Seo, and H. J. Kim, "Operating an unknown drawer using an aerial manipulator," in *Proc. IEEE International Conference on Robotics and Automation (ICRA),* Seattle, WA, 2015, pp. 5503–5508.

[82] F. Augugliaro *et al.*, "The flight assembled architecture installation: Cooperative construction with flying machines," *IEEE Control Systems*, vol. 34, no. 4, pp. 46–64, 2014.

[83] Q. Lindsey and V. Kumar, "Distributed construction of truss structures," in *Algorithmic Foundations of Robotics X*, Springer, 2013, pp. 209–225.

[84] S. R. B. dos Santos, S. N. Givigi, and C. L. Nascimento, "Autonomous construction of structures in a dynamic environment using reinforcement learning," in *Proc. IEEE International Systems Conference (SysCon),* Orlando, FL, 2013, pp. 452–459.

[85] F. Augugliaro, A. Mirjan, F. Gramazio, M. Kohler, and R. D'Andrea, "Building tensile

structures with flying machines," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS),* Tokyo, Japan, 2013, pp. 3487–3492.

[86]   F. Augugliaro, E. Zarfati, A. Mirjan, and R. D'Andrea, "Knot-tying with flying machines for aerial construction," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS),* Hamburg, Germany, 2015, pp. 5917–5922.

[87]   J. R. Kutia, K. A. Stol, and W. Xu, "Canopy sampling using an aerial manipulator: A preliminary study," in *Proc. International Conference on Unmanned Aircraft Systems (ICUAS),* Denver, CO, 2015, pp. 477–484.

[88]   P. Ranque, D. Freeman, K. Kernstine, D. Lim, E. Garcia, and D. Mavris, "Stochastic agent-based analysis of UAV mission effectiveness," in *Proc. 11th AIAA ATIO Conference,* Virginia Beach, VA, 2011, vol. 20011.

[89]   M. Fingas, *The basics of oil spill cleanup*. CRC press, 2012.

[90]   J. Thomas, G. Loianno, K. Daniilidis, and V. Kumar, "Visual servoing of quadrotors for perching by hanging from cylindrical objects," *IEEE Robotics and Automation Letters*, vol. 1, no. 1, pp. 57–64, 2016.

[91]   J. Estévez, J. M. Lopez-Guede, and M. Graña, "Quasi-stationary state transportation of a hose with quadrotors," *Robotics and Autonomous Systems*, vol. 63, pp. 187–194, 2015.

[92]   C. Korpela, P. Brahmbhatt, M. Orsag, and P. Oh, "Towards the realization of mobile manipulating unmanned aerial vehicles (MM-UAV): Peg-in-hole insertion tasks," in *Proc. IEEE International Conference on Technologies for Practical Robot Applications (TePRA),* Woburn, MA, 2013, pp. 1–6.

[93]   C. M. Korpela, T. W. Danko, and P. Y. Oh, "MM-UAV: Mobile manipulating unmanned aerial vehicle," *Journal of Intelligent & Robotic Systems*, vol. 65, no. 1, pp. 93–101, 2012.

[94]   M. Orsag, C. Korpela, and P. Oh, "Modeling and control of MM-UAV: Mobile manipulating unmanned aerial vehicle," *Journal of Intelligent & Robotic Systems*, pp. 1–14, 2013.

[95]   M. Kobilarov, "Nonlinear trajectory control of multi-body aerial manipulators," *Journal of Intelligent & Robotic Systems*, vol. 73, no. 1–4, p. 679, 2014.

[96]   L. A. Sandino, D. Santamaria, M. Bejar, K. Kondak, A. Viguria, and A. Ollero, "First experimental results on enhancing hovering performance of unmanned helicopters by

structures with flying machines," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS),* Tokyo, Japan, 2013, pp. 3487–3492.

[86]   F. Augugliaro, E. Zarfati, A. Mirjan, and R. D'Andrea, "Knot-tying with flying machines for aerial construction," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS),* Hamburg, Germany, 2015, pp. 5917–5922.

[87]   J. R. Kutia, K. A. Stol, and W. Xu, "Canopy sampling using an aerial manipulator: A preliminary study," in *Proc. International Conference on Unmanned Aircraft Systems (ICUAS),* Denver, CO, 2015, pp. 477–484.

[88]   P. Ranque, D. Freeman, K. Kernstine, D. Lim, E. Garcia, and D. Mavris, "Stochastic agent-based analysis of UAV mission effectiveness," in *Proc. 11th AIAA ATIO Conference,* Virginia Beach, VA, 2011, vol. 20011.

[89]   M. Fingas, *The basics of oil spill cleanup*. CRC press, 2012.

[90]   J. Thomas, G. Loianno, K. Daniilidis, and V. Kumar, "Visual servoing of quadrotors for perching by hanging from cylindrical objects," *IEEE Robotics and Automation Letters*, vol. 1, no. 1, pp. 57–64, 2016.

[91]   J. Estévez, J. M. Lopez-Guede, and M. Graña, "Quasi-stationary state transportation of a hose with quadrotors," *Robotics and Autonomous Systems*, vol. 63, pp. 187–194, 2015.

[92]   C. Korpela, P. Brahmbhatt, M. Orsag, and P. Oh, "Towards the realization of mobile manipulating unmanned aerial vehicles (MM-UAV): Peg-in-hole insertion tasks," in *Proc. IEEE International Conference on Technologies for Practical Robot Applications (TePRA),* Woburn, MA, 2013, pp. 1–6.

[93]   C. M. Korpela, T. W. Danko, and P. Y. Oh, "MM-UAV: Mobile manipulating unmanned aerial vehicle," *Journal of Intelligent & Robotic Systems*, vol. 65, no. 1, pp. 93–101, 2012.

[94]   M. Orsag, C. Korpela, and P. Oh, "Modeling and control of MM-UAV: Mobile manipulating unmanned aerial vehicle," *Journal of Intelligent & Robotic Systems*, pp. 1–14, 2013.

[95]   M. Kobilarov, "Nonlinear trajectory control of multi-body aerial manipulators," *Journal of Intelligent & Robotic Systems*, vol. 73, no. 1–4, p. 679, 2014.

[96]   L. A. Sandino, D. Santamaria, M. Bejar, K. Kondak, A. Viguria, and A. Ollero, "First experimental results on enhancing hovering performance of unmanned helicopters by

using a tethered setup," *Robotics and Autonomous Systems*, vol. 79, pp. 147–155, 2016.

[97] S. Di Lucia, G. D. Tipaldi, and W. Burgard, "Attitude stabilization control of an aerial manipulator using a quaternion-based backstepping approach," in *Proc. European Conference on Mobile Robots (ECMR),* Lincoln, UK, 2015, pp. 1–6.

[98] A. Chovancová, T. Fico, P. Hubinský, and F. Duchoň, "Comparison of various quaternion-based control methods applied to quadrotor with disturbance observer and position estimator," *Robotics and Autonomous Systems*, vol. 79, pp. 87–98, 2016.

[99] H. Abaunza, P. Castillo, A. Victorino, and R. Lozano, "Dual Quaternion Modeling and Control of a Quad-rotor Aerial Manipulator," *Journal of Intelligent & Robotic Systems*, pp. 1–17, 2017.

[100] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo, "Robotics: modelling, planning and control, ser. Advanced Textbooks in Control and Signal Processing," *Springer*, vol. 26, p. 29, 2009.

[101] S. Kannan, M. Alma, M. A. Olivares-Mendez, and H. Voos, "Adaptive control of aerial manipulation vehicle," in *Proc. IEEE International Conference on Control System, Computing and Engineering (ICCSCE),* Batu Ferringhi, Malaysia, 2014, pp. 273–278.

[102] S. Bouabdallah, P. Murrieri, and R. Siegwart, "Design and control of an indoor micro quadrotor," in *Proc. IEEE International Conference on Robotics and Automation (ICRA),* New Orleans, LA, 2004, vol. 5, pp. 4393–4398.

[103] S. Bouabdallah and R. Siegwart, "Full control of a quadrotor," in *Proc. IEEE/RSJ international conference on Intelligent robots and systems (IROS),* San Diego, CA, 2007, pp. 153–158.

[104] V. Lippiello and F. Ruggiero, "Cartesian impedance control of a UAV with a robotic arm," *IFAC Proceedings Volumes*, vol. 45, no. 22, pp. 704–709, 2012.

[105] H. Lee and H. J. Kim, "Estimation, Control, and Planning for Autonomous Aerial Transportation," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 4, pp. 3369–3379, 2017.

[106] A. Moutinho, M. Figueirôa, and J. R. Azinheira, "Attitude estimation in SO (3): a comparative UAV case study," *Journal of Intelligent & Robotic Systems*, vol. 80, no. 3–4, pp. 375–384, 2015.

[107] T. Zhang and Y. Liao, "Attitude measure system based on extended Kalman filter for multi-rotors," *Computers and Electronics in Agriculture*, vol. 134, pp. 19–26, 2017.

[108] F. A. Goodarzi and T. Lee, "Extended Kalman filter on SE (3) for geometric control of a quadrotor UAV," in *Proc. International Conference on Unmanned Aircraft Systems (ICUAS),* Arlington, VA, 2016, pp. 1371–1380.

[109] H. Bonyan Khamseh and F. Janabi-Sharifi, "UKF–Based LQR Control of a Manipulating Unmanned Aerial Vehicle," *Unmanned Systems*, vol. 5, no. 3, pp. 131–139, 2017.

[110] H. Bonyan Khamseh and F. Janabi–Sharifi, "Unscented Kalman filter state estimation for manipulating unmanned aerial vehicles," *Journal of Aerospace Science and Technology (Under Review).*

[111] S. Jafarzadeh, C. Lascu, and M. S. Fadali, "State estimation of induction motor drives using the unscented Kalman filter," *IEEE Transactions on Industrial Electronics*, vol. 59, no. 11, pp. 4207–4216, 2012.

[112] V. Grabe, H. H. Bülthoff, and P. R. Giordano, "On-board velocity estimation and closed-loop control of a quadrotor UAV based on optical flow," in *Proc. IEEE International Conference on Robotics and Automation (ICRA),* Saint Paul, MN, 2012, pp. 491–497.

[113] V. Lippiello and B. Siciliano, "Wall inspection control of a VTOL unmanned aerial vehicle based on a stereo optical flow," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS),* Vilamoura, Portugal, 2012, pp. 4296–4302.

[114] A. Santamaria-Navarro, P. Grosch, V. Lippiello, J. Sola, and J. Andrade-Cetto, "Uncalibrated Visual Servo for Unmanned Aerial Manipulation," *IEEE/ASME Transactions on Mechatronics*, vol. 22, no. 4, pp. 1610–1621, 2017.

[115] A. Khalifa, M. Fanni, A. Ramadan, and A. Abo-Ismail, "Modeling and control of a new quadrotor manipulation system," in *Proc. First International Conference on Innovative Engineering Systems (ICIES),* Alexandria, Egypt, 2012, pp. 109–114.

[116] S. Kannan, S. Quintanar-Guzman, J. Dentler, M. A. Olivares-Mendez, and H. Voos, "Control of aerial manipulation vehicle in operational space," in *Proc. 8th International Conference on Electronics, Computers and Artificial Intelligence (ECAI),* Ploiesti, Romania, 2016, pp. 1–4.

[117] M. Orsag, C. Korpela, S. Bogdan, and P. Oh, "Lyapunov based model reference adaptive control for aerial manipulation," in *Proc. International Conference on Unmanned Aircraft Systems (ICUAS),* Atlanta, GA, 2013, pp. 966–973.

[118] F. Caccavale, G. Giglio, G. Muscio, and F. Pierri, "Adaptive control for UAVs equipped with a robotic arm," *IFAC Proceedings Volumes*, vol. 47, no. 3, pp. 11049–11054, 2014.

[119] C. Korpela, M. Orsag, M. Pekala, and P. Oh, "Dynamic stability of a mobile manipulating unmanned aerial vehicle," in *Proc. IEEE International Conference on Robotics and Automation (ICRA),* Karlsruhe, Germany, 2013, pp. 4922–4927.

[120] H. Bonyan Khamseh and F. Janabi–Sharifi, "Modeling and Control of a Manipulating Unmanned Aerial Vehicle," in *Proc. Canadian Society for Mechanical Engineering International Congress (CSME),* Kelowna, BC, 2016.

[121] R. M. Murray, Z. Li, S. S. Sastry, and S. S. Sastry, *A mathematical introduction to robotic manipulation*. CRC press, 1994.

[122] N. Hogan, "Impedance control: An approach to manipulation," in *Proc. American Control Conference,* San Diego, CA, 1984, pp. 304–313.

[123] S. I. Part, "Impedance control: An approach to manipulation," *Journal of dynamic systems, measurement, and control*, vol. 107, p. 17, 1985.

[124] V. Lippiello and F. Ruggiero, "Exploiting redundancy in Cartesian impedance control of UAVs equipped with a robotic arm," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS),* Vilamoura, Portugal, 2012, pp. 3768–3773.

[125] R. Mebarki and V. Lippiello, "Image-Based Control for Aerial Manipulation," *Asian Journal of Control*, vol. 16, no. 3, pp. 646–656, 2014.

[126] J. Thomas, J. Polin, K. Sreenath, and V. Kumar, "Avian-inspired grasping for quadrotor micro UAVs," in *Proc. ASME International Design Engineering Technical Conference (IDETC),* Portland, Oregon, 2013.

[127] P. Ramon Soria, R. Bevec, B. C. Arrue, A. Ude, and A. Ollero, "Extracting Objects for Aerial Manipulation on UAVs Using Low Cost Stereo Sensors," *Sensors*, vol. 16, no. 5, p. 700, 2016.

[128] P. Ramon Soria, B. C. Arrue, and A. Ollero, "Detection, Location and Grasping Objects Using a Stereo Sensor on UAV in Outdoor Environments," *Sensors*, vol. 17, no. 1, p. 103, 2017.

[129] R. Mebarki, V. Lippiello, and B. Siciliano, "Image-based control for dynamically cross-coupled aerial manipulation," in *Proc. IEEE/RSJ International Conference on Intelligent*

Robots and Systems (IROS), Chicago, IL, 2014, pp. 4827–4833.

[130] R. Mebarki, V. Lippiello, and B. Siciliano, "Exploiting image moments for aerial manipulation control," in *Proc. ASME Dynamic Systems and Control Conference,* Palo Alto, CA, 2013.

[131] N. K. Ure, S. Omidshafiei, B. T. Lopez, A. Agha-Mohammadi, J. P. How, and J. Vian, "Online heterogeneous multiagent learning under limited communication with applications to forest fire management," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS),* Hamburg, Germany, 2015, pp. 5181–5188.

[132] P. Moonumca, Y. Yamamoto, and N. Depaiwa, "Adaptive PID for controlling a quadrotor in a virtual outdoor scenario: Simulation study," in *Proc. IEEE International Conference on Mechatronics and Automation (ICMA),* Takamatsu, Japan, 2013, pp. 1080–1086.

[133] S.-H. Hyon, J. G. Hale, and G. Cheng, "Full-body compliant human–humanoid interaction: balancing in the presence of unknown external forces," *IEEE Transactions on Robotics*, vol. 23, no. 5, pp. 884–898, 2007.

[134] F. Augugliaro and R. D'Andrea, "Admittance control for physical human-quadrocopter interaction," in *Proc. European Control Conference (ECC),* Zurich, Switzerland, 2013, pp. 1805–1810.

[135] "Human–UAV Interaction," 2012. [Online]. Available: https://www.youtube.com/watch?v=R4hdHwbCDac&feature=youtu.be. [Accessed: 04-Apr-2018].

[136] M. E. Campbell and W. W. Whitacre, "Cooperative tracking using vision measurements on seascan UAVs," *IEEE Transactions on Control Systems Technology*, vol. 15, no. 4, pp. 613–626, 2007.

[137] S. Grzonka, G. Grisetti, and W. Burgard, "A fully autonomous indoor quadrotor," *IEEE Transactions on Robotics*, vol. 28, no. 1, pp. 90–100, 2012.

[138] R. Mahony, V. Kumar, and P. Corke, "Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor," *IEEE Robotics & Automation Magazine*, vol. 19, no. 3, pp. 20–32, 2012.

[139] Y. Bi and H. Duan, "Implementation of autonomous visual tracking and landing for a low-cost quadrotor," *Optik-International Journal for Light and Electron Optics*, vol. 124, no. 18, pp. 3296–3300, 2013.

[140] C. Hajiyev and H. E. Soken, "Robust adaptive Kalman filter for estimation of UAV dynamics in the presence of sensor/actuator faults," *Aerospace Science and Technology*, vol. 28, no. 1, pp. 376–383, 2013.

[141] F. Janabi-Sharifi and M. Marey, "A kalman-filter-based method for pose estimation in visual servoing," *IEEE Transactions on Robotics*, vol. 26, no. 5, pp. 939–947, 2010.

[142] A. Assa and F. Janabi-Sharifi, "A Kalman filter-based framework for enhanced sensor fusion," *IEEE Sensors Journal*, vol. 15, no. 6, pp. 3281–3292, 2015.

[143] V. Lippiello, B. Siciliano, and L. Villani, "Adaptive extended Kalman filtering for visual motion estimation of 3D objects," *Control Engineering Practice*, vol. 15, no. 1, pp. 123–134, 2007.

[144] A. Shademan and F. Janabi-Sharifi, "Sensitivity analysis of EKF and iterated EKF pose estimation for position-based visual servoing," in *Proc. IEEE Conference on Control Applications (CCA),* Toronto, Ont., 2005, pp. 755–760.

[145] M. Ficocelli and F. Janabi-Sharifi, "Adaptive filtering for pose estimation in visual servoing," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS),* Maui, HI, 2001, vol. 1, pp. 19–24.

[146] R. Toledo-Moreo, M. A. Zamora-Izquierdo, B. Ubeda-Minarro, and A. F. Gómez-Skarmeta, "High-integrity IMM-EKF-based road vehicle navigation with low-cost GPS/SBAS/INS," *IEEE Transactions on Intelligent Transportation Systems*, vol. 8, no. 3, pp. 491–511, 2007.

[147] M. Tarhan and E. Altuğ, "EKF based attitude estimation and stabilization of a quadrotor UAV using vanishing points in catadioptric images," *Journal of Intelligent & Robotic Systems*, vol. 62, no. 3–4, pp. 587–607, 2011.

[148] A. Cho, J. Kim, S. Lee, and C. Kee, "Wind estimation and airspeed calibration using a UAV with a single-antenna GPS receiver and pitot tube," *IEEE transactions on aerospace and electronic systems*, vol. 47, no. 1, pp. 109–117, 2011.

[149] K. D. Sebesta and N. Boizot, "A real-time adaptive high-gain EKF, applied to a quadcopter inertial navigation system," *IEEE Transactions on Industrial Electronics*, vol. 61, no. 1, pp. 495–503, 2014.

[150] A. Nemra and N. Aouf, "Robust INS/GPS sensor fusion for UAV localization using SDRE nonlinear filtering," *IEEE Sensors Journal*, vol. 10, no. 4, pp. 789–798, 2010.

[151] G. G. Rigatos, "Nonlinear Kalman filters and particle filters for integrated navigation of unmanned aerial vehicles," *Robotics and Autonomous Systems*, vol. 60, no. 7, pp. 978–995, 2012.

[152] S. Julier, J. Uhlmann, and H. F. Durrant-Whyte, "A new method for the nonlinear transformation of means and covariances in filters and estimators," *IEEE Transactions on automatic control*, vol. 45, no. 3, pp. 477–482, 2000.

[153] S. J. Julier and J. K. Uhlmann, "A new extension of the Kalman filter to nonlinear systems," in *Proc. International Symposium on Aerospace/Defense Sensing, Simulation and Control,* Orlando, FL, 1997, vol. 3, no. 26, pp. 182–193.

[154] A. Giannitrapani, N. Ceccarelli, F. Scortecci, and A. Garulli, "Comparison of EKF and UKF for spacecraft localization via angle measurements," *IEEE Transactions on aerospace and electronic systems*, vol. 47, no. 1, pp. 75–84, 2011.

[155] R. Mehra, "Approaches to adaptive filtering," *IEEE Transactions on automatic control*, vol. 17, no. 5, pp. 693–698, 1972.

[156] C. Hajiyev and H. E. Soken, "Robust adaptive unscented Kalman filter for attitude estimation of pico satellites," *International Journal of Adaptive Control and Signal Processing*, vol. 28, no. 2, pp. 107–120, 2014.

[157] H. E. Soken and S. Sakai, "Adaptive tuning of the unscented Kalman filter for satellite attitude estimation," *Journal of Aerospace Engineering*, vol. 28, no. 3, p. 4014088, 2014.

[158] G. Chang and M. Liu, "An adaptive fading Kalman filter based on Mahalanobis distance," *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, vol. 229, no. 6, pp. 1114–1123, 2015.

[159] A. Almagbile, J. Wang, and W. Ding, "Evaluating the performances of adaptive Kalman filter methods in GPS/INS integration," *Journal of Global Positioning Systems*, vol. 9, no. 1, pp. 33–40, 2010.

[160] L. Jetto, S. Longhi, and G. Venturini, "Development and experimental validation of an adaptive extended Kalman filter for the localization of mobile robots," *IEEE Transactions on Robotics and Automation*, vol. 15, no. 2, pp. 219–229, 1999.

[161] P. S. Maybeck, R. L. Jensen, and D. A. Harnly, "An adaptive extended Kalman filter for target image tracking," *IEEE Transactions on Aerospace and Electronic Systems*, no. 2, pp. 173–180, 1981.

[162] V. Lippiello, B. Siciliano, and L. Villani, "Position-based visual servoing in industrial multirobot cells using a hybrid camera configuration," *IEEE Transactions on Robotics*, vol. 23, no. 1, pp. 73–86, 2007.

[163] Z. Jiang, Q. Song, Y. He, and J. Han, "A novel adaptive unscented Kalman filter for nonlinear estimation," in *Proc. 46th IEEE Conference on Decision and Control,* New Orleans, LA, 2007, pp. 4293–4298.

[164] Q. Song and Y. He, "Adaptive unscented Kalman filter for estimation of modelling errors for helicopter," in *Proc. IEEE International Conference on Robotics and Biomimetics (ROBIO),* Guilin, China, 2009, pp. 2463–2467.

[165] S. Qi and H. Jian-Da, "An adaptive UKF algorithm for the state and parameter estimations of a mobile robot," *Acta Automatica Sinica*, vol. 34, no. 1, pp. 72–79, 2008.

[166] S. Bouabdallah and R. Siegwart, "Design and control of a miniature quadrotor," in *Advances in unmanned aerial vehicles*, Springer, 2007, pp. 171–210.

[167] A. L. Salih, M. Moghavvemi, H. A. F. Mohamed, and K. S. Gaeid, "Flight PID controller design for a UAV quadrotor," *Scientific research and essays*, vol. 5, no. 23, pp. 3660–3667, 2010.

[168] A. Chovancová, T. Fico, Ľ. Chovanec, and P. Hubinsk, "Mathematical modelling and parameter identification of quadrotor (a survey)," *Procedia Engineering*, vol. 96, pp. 172–181, 2014.

[169] L. D. Minh and C. Ha, "Modeling and control of quadrotor MAV using vision-based measurement," in *Proc. International Forum on Strategic Technology (IFOST),* Ulsan, South Korea, 2010, pp. 70–75.

[170] D. E. Kirk, *Optimal control theory: an introduction*. Courier Corporation, 2012.

[171] D. Simon, *Optimal state estimation: Kalman, H infinity, and nonlinear approaches*. John Wiley & Sons, 2006.

[172] A. Peirce and F. Rochinha, "An integrated extended Kalman filter–implicit level set algorithm for monitoring planar hydraulic fractures," *Inverse Problems*, vol. 28, no. 1, pp. 15009–15021, 2011.

[173] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with applications to tracking and navigation: theory algorithms and software*. John Wiley & Sons, 2004.

[174] E. A. Wan and R. Van Der Merwe, "The unscented Kalman filter for nonlinear estimation," in *Proc. IEEE Symposium on Adaptive Systems for Signal Processing, Communications, and Control* ,Lake Louise, Alberta, 2000, pp. 153–158.

[175] S. Kolås, B. A. Foss, and T. S. Schei, "Constrained nonlinear state estimation based on the UKF approach," *Computers & Chemical Engineering*, vol. 33, no. 8, pp. 1386–1401, 2009.

[176] A. De Ruiter, "A simple suboptimal Kalman filter implementation for a gyro-corrected satellite attitude determination system," *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, vol. 224, no. 7, pp. 787–802, 2010.

[177] C. Hajiyev, "Adaptive filtration algorithm with the filter gain correction applied to integrated INS/radar altimeter," *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, vol. 221, no. 5, pp. 847–855, 2007.

[178] H. E. Soken and C. Hajiyev, "Pico satellite attitude estimation via robust unscented Kalman filter in the presence of measurement faults," *ISA transactions*, vol. 49, no. 3, pp. 249–256, 2010.

[179] K. Myers and B. D. Tapley, "Adaptive sequential estimation with unknown noise statistics," *IEEE Transactions on Automatic Control*, vol. 21, no. 4, pp. 520–523, 1976.

[180] R. Sanz, L. Rodenas, P. Garcia, and P. Castillo, "Improving attitude estimation using inertial sensors for quadrotor control systems," in *Proc. International Conference on Unmanned Aircraft Systems (ICUAS),* Orlando, FL, 2014, pp. 895–901.

[181] M. H. Amoozgar, A. Chamseddine, and Y. Zhang, "Experimental test of a two-stage Kalman filter for actuator fault detection and diagnosis of an unmanned quadrotor helicopter," *Journal of Intelligent & Robotic Systems*, vol. 70, no. 1–4, pp. 107–117, 2013.

[182] R. Mebarki and V. Lippiello, "Image moments-based velocity estimation of UAVs in GPS denied environments," in *Proc. IEEE International Symposium on Safety, Security, and Rescue Robotics,* Hokkaido, Japan, 2014, pp. 1–6.

[183] Y. Tamura, M. Matsui, L.-C. Pagnini, R. Ishibashi, and A. Yoshida, "Measurement of wind-induced response of buildings using RTK-GPS," *Journal of Wind Engineering and Industrial Aerodynamics*, vol. 90, no. 12–15, pp. 1783–1793, 2002.

[184] T. Takasu and A. Yasuda, "Development of the low-cost RTK-GPS receiver with an open source program package RTKLIB," in *Proc. International symposium on GPS/GNSS,* Jeju, South Korea, 2009, pp. 4–6.

[185] R. Lenain, B. Thuilot, C. Cariou, and P. Martinet, "Adaptive control for car like vehicles guidance relying on RTK GPS: Rejection of sliding effects in agricultural applications," in *Proc. IEEE International Conference on Robotics and Automation (ICRA),* Taipei, Taiwan, 2003, pp. 115–120.

[186] E. White and J. A. Rios, "FAA certification of a MEMS attitude and heading reference system," in *Proc. Institute of Navigation National Technical Meeting,* San Diego, CA, 2002, pp. 158–169.

[187] A. Brown and Y. Lu, "Performance test results of an integrated GPS/MEMS inertial navigation package," in *Proc. Institute of Navigation GNSS Meeting,* Long Beach, California, 2004, pp. 825–832.

[188] D. Hazry and M. R. Mohd Sofian, "Study of inertial measurement unit sensor," *Proc. International Conference on Man-Machine Systems,* Batu Ferringhi, Malaysia, p. 5A8-1-5A8-4, 2009.

[189] S. Akhlaghi, N. Zhou, and Z. Huang, "Adaptive adjustment of noise covariance in Kalman filter for dynamic state estimation," in *Proc. IEEE Power & Energy Society General Meeting,* Chicago, IL, 2017, pp. 1–5.

[190] H. Yang and D. Lee, "Dynamics and control of quadrotor with robotic manipulator," in *Proc. IEEE International Conference on Robotics and Automation (ICRA),* Hong Kong, 2014, pp. 5544–5549.

[191] I. D. Cowling, O. A. Yakimenko, J. F. Whidborne, and A. K. Cooke, "A prototype of an autonomous controller for a quadrotor UAV," in *Proc. European Control Conference (ECC),* Kos, Greece, 2007, pp. 4001–4008.

[192] E. Reyes-Valeria, R. Enriquez-Caldera, S. Camacho-Lara, and J. Guichard, "LQR control for a quadrotor using unit quaternions: Modeling and simulation," in *Proc. International Conference on Electronics, Communications and Computing (CONIELECOMP),* Cholula, Mexico, 2013, pp. 172–178.

[193] S. J. Julier and J. K. Uhlmann, "Reduced sigma point filters for the propagation of means and covariances through nonlinear transformations," in *Proc. American Control Conference (ACC),* Anchorage, AK, 2002, vol. 2, pp. 887–892.

[194] S. J. Julier and J. K. Uhlmann, "Unscented filtering and nonlinear estimation," *Proceedings of the IEEE*, vol. 92, no. 3, pp. 401–422, 2004.

[195] S. J. Julier, "The spherical simplex unscented transformation," in *Proc. American Control*

Conference (ACC), Denver, CO, 2003, vol. 3, pp. 2430–2434.

[196] E.-H. Shin and N. El-Sheimy, "An unscented Kalman filter for in-motion alignment of low-cost IMUs," in *Proc. Position Location and Navigation Symposium,* Monterey, CA, 2004, pp. 273–279.

[197] A. T. Nugraha and T. Agustinah, "Quadcopter Path Following Control Design Using Output Feedback with Command Generator Tracker LOS Based At Square Path," *Journal of Physics: Conference Series*, vol. 947, no. 1, pp. 12074–12081, 2018.

[198] D. Mellinger, M. Shomin, N. Michael, and V. Kumar, "Cooperative grasping and transport using multiple quadrotors," in *Distributed autonomous robotic systems*, Springer, 2013, pp. 545–558.