UNIVERSITY OF TECHNOLOGY SYDNEY
Faculty of Engineering and Information Technology

CONTROL ARCHITECTURE AND PATH PLANNING FOR QUADCOPTERS IN FORMATION

by

Van Truong Hoang

A Thesis Submitted
in Partial Fulfillment of the
Requirements for the Degree

Doctor of Philosophy

Sydney, Australia

2019

Certificate of Original Authorship

I, Van Truong Hoang declare that this thesis, is submitted in fulfilment of the requirements for the award of Doctor of Philosophy degree, in the School of Electrical and Data Engineering, Faculty of Engineering and Information Technology at the University of Technology Sydney.

This thesis is wholly my own work unless otherwise reference or acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis.

This document has not been submitted for qualifications at any other academic institution.

This research is supported by the Australian Government Research Training Program.

Signature: Production Note:
Signature removed prior to publication.

Date: 28 June 2019

ABSTRACT

CONTROL ARCHITECTURE AND PATH PLANNING FOR QUADCOPTERS IN FORMATION

by

Van Truong Hoang

Unmanned aerial vehicles (UAVs) have found many areas of operation with numerous studies available in the literature. However, increasing demands in applications and the rapid development of technologies have transcended the use of a single UAV to the formation and their coordination. In the literature, UAVs' low-level control, path planning, and formation maintenance have been addressed mainly in separation. This research proposes a control architecture that integrates those three subsystems with a task assessment unit and communication links to accommodate a variety of applications.

At the low level, robustness of the UAV control systems is important for applications which require accurate attitudes, also for safety maintenance and configuration preservation when flying in formation. In operations, UAVs are often subject to nonlinearity, external disturbances, parametric uncertainties and strong coupling, which may downgrade their control performance. Therefore, the first focus is to design robust control schemes to track desired attitudes under various conditions. Accordingly, robust low-level controllers for UAVs are developed, namely the adaptive quasi-continuous and adaptive twisting sliding mode control. They offer a novel technique to adaptively change the control parameters of the so-called sliding modes for the sake of performance improvements.

To deploy multiple-UAV systems, the proposed control architecture includes robust control, path planning, and formation maintenance to create a real-time system that can be used for many engineering purposes. The system coordinates multiple

UAVs in a specific formation to collect data of the inspected objects. The hardware extension on the basis of 3DR Solo drones includes the Internet of Things (IoT) and environmental sensors. Communication links are implemented by employing IoT boards for components of the control architecture to equip them with network and data processing capabilities.

For UAV formation control, a novel multi-objective angle-encoded particle swarm optimisation algorithm is proposed to generate formation trajectories. Here, the algorithm is developed to minimise a cost function incorporating multiple objectives subject to formation constraints that include inspection task completion, shortest paths and safe operation of the drones.

To handle difficulties arising from various inspection surfaces, avoid possible dynamic collisions, and maintain safe motion of the whole UAV formation, the path planning algorithm is incorporated with a reconfigurable capability developed to be integrated to the control architecture. This integration allows for flexible changing of the formation to accommodate additional constraints on collision avoidance, flight altitude, communication range, and visual inspection requirements.

Throughout the dissertation, analytical work developed is validated by extensive simulation, comparisons and experiments to evaluate the proposed approach and confirm its feasibility and effectiveness. Discussions on theoretical aspects and implementation details are included together with some recommendations.

Acknowledgements

Firstly, I would like to express my deepest appreciation to my principal supervisor, Associate Professor Quang Ha for the opportunity to work on this topic as well as his academic supervision of my candidature here at the University of Technology, Sydney. His direction, motivation, expertise and criticism allowed me to reach this significant milestone in my life.

I also would like to express my gratitude to my co-supervisor, Dr. Manh Duong (Danny) Phung, for his enthusiastic supervision and support of my academic research. It was an immense help, and he contributed significantly to the completion of this work. He is more than a brother with valuable suggestions and assistance in all other aspects of life during my time in Sydney.

I wish to thank my another co-supervisor, Dr. Ngoc Ha Pham for his academic advice to improve the research.

My gratitude also goes to my friend and colleague, Mr. Ansu Man Singh for his warm advice and encouragement as well as his time to support me when I faced the crisis time of my Ph.D. candidate. Thanks also go to my colleagues, Mr. Dinh Tran Hiep, Dr. Santanu Metia, Mr. Qiuchen Zhu for their assistance to overcome difficulties during the candidature. I also wish to thank all of the other Ph.D. candidates in my group as well as friends at UTS who have helped me and made my life more enjoyable.

I would like to take this opportunity to gratefully acknowledge the Vietnam International Education Development (VIED) under the Ministry of Education and Training (MoET) and the Graduate Research School (GRS), University of Technology, Sydney (UTS), for the financial support of my research under the UTS-VIED Ph.D. Student Collaborative Scholarship.

Most importantly, I would like to dedicate this dissertation to my family. I am forever in debt to my parents, Thi Phuong Nguyen and Ngoc Hap Hoang. Without the sacrifices they have made on my behalf, I would not be where I am today. My wife, Thi Hue Bui, my beloved children Trong Huynh and Huyen Trang have given me unbounded love and support throughout my life. I owe my wife for the extremely hard time she has had to take care of all family members by herself so that I have been able to concentrate only on my study. This thesis could not have been completed without her silent sacrifices.

List of publications

The content of this thesis has been published in the following papers:

Journal papers

J-1. **V. T. Hoang**, M. D. Phung, T. H. Dinh, and Q. P. Ha, "System architecture for real-time surface inspection using multiple UAVs," IEEE Systems Journal, accepted.

J-2. **Van Truong Hoang**, Quang Hieu Pham, "Super-twisting second-order sliding mode for attitude control of quadcopter UAVs," Special Issue On Measurement, Control and Automation, Vietnam Journal of Automation Today, vol. 20, pp. 52-59, 2017.

Conference papers

C-1. **V. T. Hoang**, M. D. Phung, T. H. Dinh, Q. Zhu, and Q. P. Ha, "Reconfigurable multi-UAV formation using angle-encoded PSO," in The 2019 IEEE 15th International Conference on Automation Science and Engineering (CASE 2019), Vancouver, Canada, 22-26 AUG 2019, accepted.

C-2. Qiuchen Zhu, Tran Hiep Dinh, **Van Truong Hoang**, Manh Duong Phung, and Quang Phuc Ha, "Crack detection using enhanced thresholding on UAV based collected images," in Australasian Conference on Robotics and Automation 2018 (ACRA 2018), Canterbury, New Zealand, 4-6 DEC 2018.

C-3. **V. T. Hoang**, M. D. Phung, T. H. Dinh, and Q. P. Ha, "Angle-encoded swarm optimization for UAV formation path planning," in The 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2018), Madrid, Spain, 1-5 OCT 2018. IEEE, pp. 5239-5244.

C-4. **V. T. Hoang**, M. D. Phung, and Q. P. Ha, "Adaptive twisting sliding mode control for quadrotor unmanned aerial vehicles," in Control Conference (ASCC), The 11th Asian, Gold Coast, Australia, 17 - 20 DEC 2017. IEEE, pp. 671-676.

C-5. Manh Duong Phung, **Van Truong Hoang**, Tran Hiep Dinh, and Quang Ha, "Automatic crack detection in built infrastructure using unmanned aerial vehicles," in Proceedings of the 2017 International Symposium on Automation and Robotics in Construction (ISARC 2017), Taipei, Taiwan. 28 JUN-1 JUL 2017, pp. 823-829.

C-6. **Van Truong Hoang**, Ansu Man Singh, Manh Duong Phung, and Quang Ha, "Adaptive second-order sliding mode control of UAVs for civil applications," in Proceedings of the 2017 International Symposium on Automation and Robotics in Construction (ISARC 2017), Taipei, Taiwan. 28 JUN-1 JUL 2017, pp. 816-822.

C-7. Ansu Man Singh, **Van Truong Hoang**, and Quang Ha, "Fast terminal sliding mode control for gantry cranes," in Proceedings of the 2016 International Symposium on Automation and Robotics in Construction (ISARC 2016), Auburn, USA, 18-21 JUL 2016, pp. 437-443.

Contents

List of figures

Abbreviation

1-SMC	First-order sliding mode control
2-SMC	Second-order sliding mode control
2-D	Two-dimensional
3-D	Three-dimensional
ANN	Artificial neural network
APF	Artificial potential field
AdTSM	Adaptive twisting sliding mode
AQCSM	Adaptive quasi-continuous sliding mode
ATSM	Accelerated twisting sliding mode
CGLA	Geometric learning algorithm
COG	Center of gravity
IP	Internet protocol
IWP	Intermediate waypoint
GA	Genetic algorithm
GCS	Ground control stations
GPS	Global positioning system
GST	Google satellite map
HALE	High altitude long endurance
HOSM	Higher-order sliding modes
IoT	Internet of things
LIDAR	Light detection and ranging
L-F	Leader-follower
LQG	Linear quadratic gaussian

LQR	Linear quadratic regulator
MALE	Medium altitude long endurance
MAV	Micro UAV
MPC	Model predictive control
MUAV	Miniature UAV
NAV	Nano UAV
PD	Proportional-derivative
PID	Proportional-integral-derivative
PRM	Probabilistic roadmap method
PSO	Particle swarm optimisation
QCSM	Quasi-continuous sliding mode
RCU	Remote computational units
RRT	Rapidly exploring random tree
RTK	Real-time kinematic
RTP	Real-time transport protocol
SOSM	Second-order sliding modes
STSM	Super-twisting sliding mode
SUAV	Small UAV
TCP	Transmission control protocol
TUAV	Tactical UAV
UAV	Unmanned aerial vehicle
UDP	User datagram protocol
UGV	Unmanned ground vehicle
VRB	Virtual rigid body
VS	Virtual structure
VTOL	Vertical take-off and landing
θ-PSO	Angle-encoded particle swarm optimisation

Nomenclature and Notation

	General Formatting Style
$[\dot{\cdot}]$	The time derivative of a variable
$[\cdot]^T$	Transpose of a vector or a matrix
$\|\cdot\|$	The norm of a vector
$\text{sign}(\cdot)$	The signum function
	Specific Symbol Usage
$\{E\}$	Earth fixed inertial coordinate frame
$\{B\}$	A quadcopter's body coordinate frame
$\{F\}$	A formation coordinate frame
$\{O\}$	A inertia frame of the formation
c	Force-to-torque coefficient of a quadcopter
c_1, c_2	PSO gain coefficients
C_j	Coordinates of the jth IWP
C_k	Centers coordinate of the kth obstacle
c_x	$cos(x)$
d	Disturbance vector
d_{com}	Communication range
d_i	Distance between the UAV_i and the formation centroid
d_k^S	Safe distance of the kth obstacle
d_θ	Pitch disturbance
d_ϕ	Roll disturbance
d_ψ	Yaw disturbance
e	A vector containing control error

F	Sum of aerodynamic forces affecting the quadcopter
F_i	Lift force generated by the ith motor/propeller
g	Gravitational acceleration
g_{ij}	A PSO global-best position
H	Transformation matrix
I	Inertia matrix of a quadcopter
J	The overall cost function
J_1	A cost component corresponding to a path length
J_2	A cost component corresponding to a violation
J_3	A cost component corresponding to a flying altitude
J_F	A cost function for path planning algorithm
J_R	A cost component corresponding to an intermediate waypoint
K	Number of obstacles
L_i	The total number of segments of the path T_{Fi}
M_j	Number of intermediate waypoints
p	Roll angular velocity of a quadcopter
P_n^*	Flying path for the UAVn
P_F	Position of the formation's centroid
p_{ij}	A PSO local-best positions
P_F^j	A set of waypoints of the formation at the jth intermediate waypoint
P_n^j	A set of waypoints of the UAVn at the jth intermediate waypoint
q	Pitch angular velocity of a quadcopter
R	Rotation matrix
r	Yaw angular velocity of a quadcopter
r_1, r_2	PSO pseudorandom scalars
r_F	Radius of the formation
r_k	Radius of the kth obstacle

r_Q	Radius of a quadcopter including propellers
$r_{l,k}^S$	Safe radius from the lth path segment to the kth obstacle
r_n^S	Safe radius of the nth UAV
$S(\omega)$	A skew-symmetric matrix
s_x	$sin(x)$
T_F^*	Reference trajectory of the formation centroid
T_{Fi}	The formation path at ith iteration
T_n	Reference trajectory for the nth UAV
u	A vector of control input
u_D	Discontinuous control component
u_{eq}	Equivalent control component
u_T	A twisting sliding controller
u_z	Altitude control input
u_θ	Pitch control input
u_ϕ	Roll control input
u_ψ	Yaw control input
V_0	Lyapunov function of the sliding surface
v_{ij}	Velocity of the ith particle in dimension j
V_l	Violation cost of the lth segment
V_n	Velocity profile for the nth UAV
$V_{\sigma,\alpha}$	Global Lyapunov function
X	State of a dynamic system
x_{ij}	Position of the ith particle in dimension j
x_{max}	Upper restriction of the search space
x_{min}	Lower restriction of the search space
z_k^{max}	Maximum altitude of the kth obstacle
α	A vector containing adaptive gains

α_i	An adaptive gain of the state i
$\alpha_{M,i}$	A maximum possible value of the adaptive gain
$\alpha_{m,i}$	A threshold of the adaptive gain
$\Delta\theta_{ij}$	phase angle increment of the ith particle in dimension j
Θ	Orientation vector of a quadcopter
θ	Pitch angle
θ_{ij}	Phase angle of the ith particle in dimension j
Λ_g	θ-PSO global best positions
Λ_i	θ-PSO personal best positions
σ	A vector containing the sliding surface components
τ	A torque component caused by thrust forces
τ_a	A torque component caused by the aerodynamic friction
τ_b	A torque component caused by body gyroscopic effects
τ_p	A torque component caused by propeller gyroscopic effects
τ_θ	Pitch torque
τ_ϕ	Roll torque
τ_ψ	Yaw torque
ϕ	Roll angle
ψ	Yaw angle
ω	A vector containing the three angular rates
Ω_r	Overall residual propeller angular speed

Chapter 1

Introduction

1.1 Unmanned aerial vehicles

Unmanned aerial vehicles (UAVs) are aircraft without a human pilot on board, and sometimes they can be referred to as drones, remotely piloted vehicles, remotely piloted aircraft, remotely operated aircraft or unmanned aerial combat vehicles. UAVs can be remotely controlled by human operators or autonomously governed by computer programs and systems. Under the development of higher levels of control algorithms such as path planning and simultaneous localisation and mapping (SLAM), and advanced technologies like precise maps, artificial intelligence, and computing, new generations of UAVs are expected to execute their flight missions in higher degrees of autonomy and with less involvement of human beings.

1.1.1 Unmanned aerial vehicle categories

As being widely used, there is a high degree of differentiation with respect to a UAVs size, objective, capabilities, and cost. Their types are then classified on different basis and criterion. Typically, they are divided into two functional categories, i.e., configuration, and size and endurance of UAVs.

Configuration based UAVs

Based on their configuration, UAVs are basically divided into three types: Fixed-wing, Rotary-wing and Hybrid.

- *Fixed-wing UAVs*

These UAVs have wings that are fixed at a designed position on the UAVs' shell. These wings produce lift force by employing horizontal velocity of the UAVs. The movement of UAVs also depend on the shape of the wings and/or their attached fins. The movement in take-off and landing phase of this UAV type requires alternatively a runway, launching mechanisms and/or landing parachute.

The fixed-wing UAVs are able to work in long duration, large travel distance with high velocity. Besides that, they are the simplest designs. However, the support systems are more complicated as there is the requirement of runways or launching devices. Other disadvantage of these UAVs is the ease of damage during take-off and landing processes.

- *Rotary-wing UAVs*

They are often known as Vertical Take-off and Landing (VTOL) aircrafts. Instead of using wings to generate lift force, these UAVs employ motors/propellers to provide floating force as well as vertical and horizontal movements. Runways or special launching devices are not required for these UAVs as they are able to take-off and land vertically.

In comparison with fixed-wings, rotary-wing UAVs have some advantages, i.e., less complex system, hovering ability, and flexible manoeuvrability. Their systems are simpler as they do not need runways or launching devices. The rotary wings give them the ability to hover at a location. Particularly, they are flexible with high manoeuvrability with the capability of changing direction in any time and at any place.

Advantages of rotary-wings enable them to carry out missions in more complex and limited workspaces. The hovering ability makes them the most appropriate flying vehicle to execute an assigned task, for instance, monitoring

a specific position, tracking unpredicted velocity and movement direction of objects, taking photographs, and surveillance. They are advantageous for quick preparation by their simple systems without a special working condition required.

Rotary-wing UAVs, however, are not able to operate at a long distance and endurance due to their high power consumption. They are also limited in speed and travel altitude.

- *Hybrid UAVs*

  Hybrid UAVs or convertible UAVs, flapping wings [153] with the idea of combining advantageous and reducing disadvantageous properties of the above two main types of UAVs. A hybrid UAV design is based on the new concept for long endurance and distance flights, high speed, capability of hovering and take-off and landing vertically. Hybrid UAVs usually include both wings and rotors. They are mechanically and aerodynamically very complex due to tilting mechanisms, thus, they have a high cost for design and maintenance.

Size and endurance based UAVs

On a size and endurance basis, UAVs are classified into High altitude long endurance (HALE), Medium altitude long endurance (MALE), Medium Range or Tactical UAV (TUAV), Close-Range UAV, miniature or small UAV (MUAV of SUAV), micro UAV (MAV) and nano UAV (NAV). As shown in Figure 1.1, the first four types are large UAVs and are often used for the armed forces involving UAVs, the major standards based on cruising altitude, time and distance of the UAVs. The remaining three of them are designed to be man-portable. These UAVs are potential to operate in both military and civilian situations, but they are widely used for for civilian purposes thanks to some advantages such as low-cost, easier to operate, simpler system in comparison with the large UAVs.

- *HALE*

  These UAVs are capable of flying at 15,000 metres height and have more than 24 hour endurance. They are often used in exceedingly long travel distance to operate reconnaissance and surveillance missions. They may be used to carry weapons so that most of them are operated by air forces.

- *MALE*

  The MALE UAVs altitude range is from 5,000 to 15,000 metres and their endurance is upto 24 hours. Their travel distance is minimum 500 kilometres, but shorter than the HALE with their missions being almost identical. MALE UAVs are suitable for being used by air forces.

- *TUAV*

  These air vehicles are often operated at a range from 100 to 300 kilometres. TUAVs systems are simpler and smaller the above HALE and MALE. They are operated also by armies and navies.

- *Close-Range UAV*

  These UAVs are operating in the distance smaller than TUAV, upto 100 kilometres. They are serving in both military and civilian purposes. Their missions include reconnaissance, monitoring, surveillance, inspection, and crop-spraying.

- *Miniature UAV (MUAV) or Small UAV (SUAV)*

  These UAVs are not heavier than 20-25 kilograms, but not smaller than the Micro UAVs. Their operating ranges are not in excess of 30 kilometres. They are capable of being hand-launched thanks to their small mass.

- *Micro UAV or MAV*

MAV is originally known as a UAV with wing-span no larger than 150 millimetres, maximum length is 15 centimetres, and its weight is not heavier than 100 grams. The MAV is preferred to operate in urban or complex environments with flight endurance from 20 to 60 minutes. Therefore, its cruising speed is adequate to be able to cover a specific location.

- *NAV Nano Air Vehicles*

  NAVs are classified as special UAVs with extremely small size, and are ultra lightweight, i.e., less than 15 centimetres and 20 grams, respectively. They are capable of performing their tasks in both military and civilian situations. Recently, they have been used in swarms for some special missions such as radar confusion or ultra-short range surveillance.

1.1.2 Quadcopters: an introduction

Quadcopters are categorised as rotary-wing and vertical take-off and landing (VTOL) UAVs. Typically, a quadcopter has four motors and four corresponding propellers attached to its body. These four propellers provide thrust force for the quadcopter to fly. To achieve fly data and to control a quadcopter, a range of sensors is equipped to the onboard control system, including IMU, GPS, and altitude measuring devices. In addition, for different desired application, specific computational units and sensors may be externally attached on a quadcopter such as compatible computers, communicational and imaging devices.

Some advantages of using the quadcopter compared to other types of UAV are hovering capability, manoeuvrable flexibility, small size and simple mechanical design. Thanks to their ability to hover, quadcopters are suitable for missions that require consistent monitoring of a special area. This capability allows the drone to vertically take-off and land, thus minimising the launching and landing region as

Figure 1.1 : Categories of size and endurance based UAVs

(Source: Nonami *et al.* [144])

well as needing fewer supporting devices. Besides that, quadrotors are highly agile as they can make abrupt changes in direction, so that they are able to track their reference trajectories sharply. Furthermore, the small size and mechanically simple design allow the UAV to be easily dissembled, deployed and transported.

On the other hand, quadcopters also have their disadvantages, such as low endurance and payload limitation. A quadrotor has to carry its weight via the thrust produced by the motors, so more power is required to stay aloft. The current capacity in battery technologies nowadays also limits their time in operation. Additionally,

because of the limited size and the need for more accessories, a quadcopter usually has a small payload carrying capability.

It is commonly believed that quadcopters can execute missions which are dull, dirty or dangerous for human pilots. They are becoming an indispensable vehicle for military applications, such as aerial intelligence, surveillance, and reconnaissance [136, 92]. Quadcopters also have the capability to assist ground soldiers in counter improvised explosive devices activities [194, 171] and combat deployment [194]. Recent reports have illustrated that quadcopters also have a great potential for civil applications, such as research and environmentally critical missions, for example:

- Infrastructure: automatic 3D reconstruction for building condition assessment [29], securing superstructures of high-rise buildings [34], or monitoring and inspecting of civil infrastructure [71, 155, 154] (Figure 1.2).

- Images and video: taking aerial imageries and videos [129], geological and archaeological surveying [128], taking aerial photography for mapping [180], incident assessment and control [113, 213].

- Security: pipeline security and powerline inspection [105], disaster control and monitoring [11, 184], searching for missing persons, traffic and coastline monitoring [68], border surveillance [64].

- Environmental situations: forestry fire detection [1], pollution control and monitoring [177, 114], sampling and analysing of atmosphere for forecasting [193], earthquake reconnaissance [60].

- Scientific research: obtaining data from hazardous or remote locations [219,86].

- Agriculture: monitoring and spraying [53, 143].

Figure 1.2 : Quadcopter operation in monitoring of high-rise buildings

Over the last decade, the quadcopter UAVs have received much research attention. The development of quadcopters covers many areas, including mechatronics and robotics research, control and planning, data engineering and communication, see. e.g., [44, 28, 29, 155, 20].

1.1.3 UAVs research areas

Low-level Control

Whereas UAVs has been developing many different categories, various control algorithms are specially required for them. As can be expected, each control design has its own advantages and disadvantages. As a result, a large number of researches to improve the control algorithms for UAVs have been reported together with advanced solutions in the literature.

While applications of quadcopters continue to grow, a great deal of effort is being devoted to better handle the control problem of quadcopters to cope with the complexity of their dynamics, system parameter variations and particularly, large external disturbances. A quadrotor drone has generally six degrees of freedom but only four independent inputs, i.e., the four rotor speeds, thus making it an under-actuated system. Apart from the coupling condition of rotational and translational motion, UAVs are also subject to highly nonlinear dynamics and aerodynamic effects, which cause microscopic frictions acting on the quadcopter, leading to the need to generate compensative forces to maintain proper movements at the steady state.

In outdoor applications, it is critical to maintain robustness and resilience of the control system to cope with the highly non-linear dynamics of UAVs and system uncertainties, sensor noise and coupling effects between the rotational and translational motions, or disturbances from aerodynamics and other external factors. Besides that, the challenges of a UAV control problem are caused by the complexity of its dynamics due to the complex structure as well as the design issue. In the case of quadcopters, their dynamic models are under-actuated because its equations present four input forces with six output states, including three translational and three rotation movements.

Therefore, designing robust control algorithms for quadcopters is an interesting topic. In addition, robust state feedback controllers are very demanding in these cases.

Path planning

The unmanned aerial vehicle (UAV) has been widely applied to military and civil fields as an effective platform. The overall control algorithm for UAVs must be upgraded to high-level control that has the functionality necessary to calculate the paths required by the UAVs to navigate their assigned tasks. Thus, the UAV path planning has become an important and interesting research focus in recent years.

Unmanned aerial vehicle (UAV) path planning is a process in which the UAV finds a path from the starting point to the destination. In this process, based on the task requirements and certain restraints, the UAV should assess the condition and information of the planning space and move along a collision-free path. Depending on the requirements of the algorithms, and the UAVs, these paths can have added to them some requirements such as short and smooth paths, under different types of constraints like limitation in altitude, and threats/prohibited flight zones avoidance. Once a path is generated, the cost of the assignment can be calculated and used in the vehicle assignment process and, if selected, the path can be used by the vehicle to perform the task.

Multi-UAV formation

The expanding demands in applications of quadcopters together with the rapid development of technologies, especially computing, sensors, and communications have transcended the use of a single UAV to the formation and coordination of a group of them. Therein, using UAVs in cooperation has advantages over an individual UAV in that they complete a common task faster with high accuracy and use

fewer sensors [142, 112]. The use of multiple UAVs has several advantages, such as cost effectiveness, enhanced strength, higher performance and efficiency [186,69,108].

Therefore, the formation problem of UAVs has become a very large field of interesting research in the last several years. Generally, a multi-UAV formation can be described as a set of two or more aerial vehicles traveling together, whose dynamic states are coupled through a common control scheme [186,142]. The ability to control UAVs working cooperatively and preserving the predefined formation configuration are primary problems. On the other hand, research in the field of control of multi-UAVs formation is still facing some other challenges, such as evaluating the control architecture and communication network limitations. As such, the formation control covers some extended problems to be discussed, including communication loss and delay, simultaneous localisation and mapping, and collision and obstacle avoidance.

1.2 Motivation

Inspection of built infrastructures to their unavoidable aging and deterioration process is essential to supervise their safe and serviceable condition. The main reason is without this work being done, unpredicted problems may occur to structures, e.g., failure, collapse, even the risk of fatality. Figure 1.3 shows some examples of collapsed structures around the world.

There has been an expanding interest in the study of infrastructure inspection. Conventional methods can be listed as foot patrol, ground vehicles and robots. These methods can be highly accurate but only for defects that can be well observed from the ground. Additionally, visual inspection is time consuming, monotonous and subjective, resulting in even larger defects possibly being missed [133, 87].

Recentlty, application of UAVs has received an increasing interest in this field thanks to their flexibility, easy manoeuvering and cost-effectiveness, compared to the

above methods. The use of UAVs is becoming an expanding innovation in inspection applications [25, 187].

As using single UAVs has proved its applicability in inspection, recent works have started to study the coordination of a group of UAVs to achieve higher time efficiency, superior performance and durable resilience [52, 109, 13]. The work in [42] recommended a multi-platform UAV system for power line inspection tasks. The experiments show that the time to inspect 70 km of power line can be reduced to 3 hours compared to a week as with the traditional inspection method. In [109], a power network damage assessment system using multiple UAVs was presented allowing the minimisation of operating time and cost.

While there has been introduced in the literature a considerable volume of researches on the above subtasks individually, they mainly focus on aspects of low-level control, path planning and formation problems in separation. To the best knowledge of the candidate, no existing work has provided an overall system to connect them, particularly in the field of infrastructure inspection.

Inspired by the aforementioned problems and recent growing research interests, this dissertation proposes an architecture for a multiple-UAV swarm in infrastructure inspection, namely Control architecture and path planning for quadcopters in formation.

1.3 Thesis objectives

Within the given time frame and provided facilities, the following specific objectives are expected to be achieved:

- Construct a system architecture of multi-UAV cooperation control to accomplish a common assigned task in infrastructure inspection. The system is a combination of low-level control for UAVs and high-level control including path

Figure 1.3 : Worldwide bridges collapse, a sign of things to come if infrastructure maintenance is ignored,

(a) The West Gate Bridge in Melbourne, 1970,

(b) A pedestrian bridge in Miami, 2018, and

(c and d) The Genoa bridge in Italy, 2018

(Source: ABC News, https://www.abc.net.au/news/2018-08-17/genoa-bridge-collapse-road-safety-ponte-morandi-west-gate/10131098, access February 2019)

planning and formation problems.

- **Low-level control:** Based on a quadcopter UAV model and some advanced second-order sliding mode controllers, control algorithms are developed with adaptive gains for the quadcopter. The objective here is for the robust attitude control of the UAVs against harsh conditions when they execute their missions

outdoors.

- **High-level control:** Based on the reconstructed environments by employment of available satellite maps, develop a new algorithm for path planning to generate a flyable and feasible trajectory for the multiple-UAV triangular formation in a given workspace. The algorithm should provide an optimised path in terms of distance while considering collision avoidance and inspection problems. The reconfigurable formation algorithms are then enhanced to accommodate the complication of the working environment and safety of the UAVs.

In order to complete the dissertation, those main objectives are split partially as the followings:

i. To develop dynamic models of a quadcopter UAV in the presence of system disturbances and uncertainties.

ii. To formulate and construct adaptive sliding mode controllers for UAV low-level control, with robustness property to deal with uncertain dynamics and external disturbances.

iii. To construct an overall system architecture containing both low-level and high-level controls.

iv. To develop the system's hardware and communication network, which are required for preserving the formation configuration, safe-keeping of UAVs, and inspected signal processing.

v. To formulate the path planning problem of multiple UAVs navigating in a desired shape for infrastructure inspection tasks.

vi. To generate optimised, feasible and obstacle-free paths for the whole formation by minimising a cost function that incorporates multiple constraints for shortest paths and safe operation of the formation.

vii. To formulate a reconfigurable formation control problem of multi-UAV to accommodate with inspection tasks of some complex structures.

vii. To provide a new algorithm for the above reconfigurable formation problem taking into account the constraints for visual inspection and selection of intermediate waypoints.

ix. To test the proposed algorithms practically in real UAVs and environmental scenarios for validation purposes.

1.4 Thesis organisation

This dissertation consists of six main chapters and is organised as follows:

- *Chapter 2*: Literature survey

  The second chapter presents the relevant literature survey of the three main parts to construct the thesis: UAV control, path planning, and formation control. The first part shortly describes generic forms of sliding mode controllers with a focus on the first-order and high-order SMCs. Specifically, brief descriptions of the two proposed SMCs to be developed, that they are quasi-continuous and twisting SMCs, are also presented there. Simultaneously, existing control approaches using SMC, as well as other control techniques for UAVs are reviewed. The second part provides a literature survey of path generation methods employed to produce desired trajectories for UAVs. The chapter ends with reviews of proposed control approaches for multiple-UAV formation.

- *Chapter 3*: Adaptive robust control design for quadcopters

  This chapter presents the development of sliding mode controllers for quadcopter UAVs including three parts: modelling, control designs, and results. The first part describes the modelling of quadcopter UAVs. The second part includes the selection of the sliding manifolds, followed by the development of the two sliding mode controllers with their adaptive gains. The chapter ends with simulations and comparisons with experimental data to illustrate the performance of the proposed control developments.

- *Chapter 4*: Multi-UAV system architecture

  This chapter provides background materials required for the design and analysis of a multi-UAV system. It starts with a short introduction of the overall system architecture. It then introduces the theoretical background for UAV path generation and formation. The chapter ends with descriptions of a specific experimental UAV and its hardware development for infrastructures inspection.

- *Chapter 5*: Angle-encoded swarm optimisation for UAV formation path planning

  A path planning algorithm for a formation of UAVs conducting inspection tasks of built infrastructure is proposed in this chapter. Here, the angle-encoded particle swarm optimisation is developed to generate the desired trajectory for multi-UAV formation represented by its centroid. Some new constraints are introduced for the path planning algorithms to improve collision-free capability and task efficiency.

- *Chapter 6*: Reconfigurable multi-UAV formation

  In this chapter, an algorithm for the reconfigurable formation of multiple UAVs is proposed. A cost function that includes various constraints on flight safety

and visual inspection is first developed for the θ-PSO path planning algorithm. A reconfigurable process is then added. The planned path and formation are then combined to derive the trajectory and velocity profiles for the UAVs.

- *Chapter 7*: Conclusion

  A brief summary of the thesis contents and its contributions is given in the final chapter, followed by recommendations for future works.

Chapter 2

Literature survey

In this chapter, reviewing of low-level control for unmanned aerial vehicles, their path generation techniques and formation control approaches, is presented.

2.1 Low-level control for unmanned aerial vehicles

2.1.1 Robust and adaptive controllers

Robust approaches

Robust control algorithms are capable of dealing with disturbances or unmodeled system dynamics. This feature is crucial for the UAVs' control system, particularly when they are utilised to operate outdoors. Consequently, the development of robustness characteristics for the UAVs control system has been taken into a particular consideration of the research world [134, 30, 189].

As the major limitation of robust controllers is poor tracking capability [220], a combination of various control methods have been recommended. In [172], a robust optimal attitude control system was created based on linear control and robust compensation. The paper [166] classified UAVs to complex aerodynamic forces then applied the backstepping-based controller that induces integral sliding modes to gain robustness of the system over smoothly bounded disturbances. The combination of the sliding-mode technique and the backstepping control technique also can be found in [17, 138, 30]. These approaches guarantee fast convergence rapidity thanks to the backstepping technique and robust characteristics of the SMC. However, the global stability of these systems has not yet been guaranteed.

The robust control concept is also achieved by a combination of distinctive approaches related to the field of control, in which, the fault-tolerant control together with robust schemes are popular for fault tolerance and robustness against disturbances, respectively. A number of successful efforts have been claimed in development of a robust control algorithms for UAVs, e.g., [110, 84, 111]. Performances of these controllers are satisfying in terms of attitude or position trajectory tracking capability when some particular circumstances are considered. The signum of the error technique had also used in [214], which presented a tracking controller in combination with an immersion and invariance ($I\&I$)- based adaptive control methodology. This approach gave asymptotic tracking of the time-varying reference trajectory. However, accuracy position had not been guaranteed. On the other hand, high chattering amplitude and frequency can be observed from the results. In [99], a robust attitude tracking controller for quadrotors is based on the nonlinear disturbance observer and a constructive design procedure to guarantee a robust transient performance. This control achieves a good result; however, a number of assumptions must be satisfied, i.e., the parameter uncertainty and maximum allowable pitch angle, which may restrict the flexibility of the drone.

Adaptive approaches

In real-time operation, quadcopters are subjected to disturbances, especially wind gusts and parametric uncertainties, thus adaptive control laws are applied to cope with these changes in the system. A continuous time-varying adaptive scheme was introduced in [45] where uncertainties in mass, moments of inertia and aerodynamic damping coefficients are known in advance. A feedback linearisation adaptive controller is provided in [148] in case of some dynamic variations in the drone center of gravity. The L_1 adaptive laws are proposed in [188, 81] to deal with wind and mass variations and in [41, 55] to compensate for model uncertainties and bounded

disturbances within a particular frequency range. The main drawback of these approaches is the trade-off between the control performance and the robustness. The paper [37] introduced a control method for quadcopters using nonlinear adaptive control based on robust fixed point transformation phenomena. Results show the reduction of tracking errors thanks to the susceptibility of the controller to parameter uncertainties or external disturbances.

For the target of solving the tradeoff between control performance and robustness, robust control designs for UAVs have been incorporated with some adaptive schemes. An adaptive robust control is utilised to compensate for the parametric uncertainty in [134]. This research is limited in handling the drone's altitude with uncertainty coming from the total mass of the structure. The work in [84] introduced a combination of a PID-like with an adaptive scheme for both translational and rotational controllers of the UAV. This approach achieved robustness and good performances of the control method, but high values of adaptation gains may downgrade the system stability. The work in [152] focuses on the robust attitude control for UAV by mixing the backstepping sliding mode controller with adaptive radial basis function neural network. This control scheme enhanced robustness and control performance; however, the complexity of the control design may restricts its implementation in a real practical UAV.

2.1.2 Sliding mode controllers

Introduction of sliding mode control

Sliding mode control (SMC) belongs to the variable structure control (VSC) systems [191]. SMC is identified as an efficient mechanism to design robust controllers for complex high-order nonlinear dynamic systems operating under uncertain conditions. Despite having a long history of more than 40 years, SMC has received a continued and frequent consideration from the worldwide researchers.

Sliding mode control is widely used due to its salient robustness against the influence of modelling errors and external disturbances [203, 16, 44, 140]. In [203], the system was separated into the full-actuated and underactuated systems and SMC was applied into the latter one. The quadcopter attitude is controlled by using a fractional-order differintegral SMC, as presented in [85]. A good stability and a robustness of the system were achieved with injected noise. In SMC, the chattering effect occurring in the steady state usually excites unmodeled frequencies of the system dynamics. To reduce the chattering effect, high-order sliding modes (HOSM) [102, 103] have been offered as a most likely preferable solution [192].

The significant benefit of SMC is low sensitivity to system variations, perturbations, and disturbances that rejects the requirement of accurate modeling. SMC allows the decoupling of the global system motion into separate partial elements of lower dimension and, as a result, decreases the complication of feedback design. In SMC, control actions are discontinuous state functions so that they can easily be implemented using power converters with relay mode as the unique possible operation method. As a consequence, the research intensity is maintained at a high level, and SMC has been verified to apply to a wide range of problems in controls and robotics.

First-order SMC (1-SMC)

This part presents a brief description of the sliding concept and the first-order sliding mode control (1-SMC). The sliding mode phenomenon may appear in dynamic systems controlled by ordinary differential equations with discontinuous state

functions, in which the control input in the second order system is represented as:

$$\ddot{x} + a_2\dot{x} + a_1 x = u, \tag{2.1}$$

$$u = M\text{sign}(\sigma), \tag{2.2}$$

$$\sigma = cx + \dot{x}, \quad c > 0, \tag{2.3}$$

where x is the state vector, u is the control input, and σ is the sliding variable, respectively, a_1, a_2, M, and c are some constants, and sign$(\cdot)$ represents the signum function.

This system generates two discontinuous values M and $-M$ on the line $\sigma = 0$ in the state plane $(x, \dot{x})$, which leads to the appearance of a sliding mode along the line. The equation $\dot{x} + cx = 0$ can be represented as the sliding mode equation.

To understand the conventional 1-SMC, a first order uncertain system is first represented as

$$\dot{x}(t) = ax(t) + bu(t) + d(x, t), \tag{2.4}$$

where $x(t) \in \mathbb{R}, u(t) \in \mathbb{R}$, and a, b are constants, $d(x, t) \in \mathbb{R}$ represents the uncertainty with the bounds being known in a priori. To stabilise the system (2.4), if initial value of $x(t)$ is positive then $\dot{x}(t)$ should be negative and vice versa. Accordingly, control law should be inverse with the sign of $x(t)$ to guarantee the stabilisation of $x(t)$.

Consider a control law:

$$u(t) = -b^{-1}(ax(t) + Q\,\text{sign}(x)), \tag{2.5}$$

where $Q > 0$ is chosen such that

$$\left|d(x, t)\right| \leq Q. \tag{2.6}$$

With control law (2.5), system (2.4) becomes

$$\dot{x}(t) = -Q\,\text{sign}(x(t)) + d(x, t). \tag{2.7}$$

The analysis of the above system can be explained as follows.

Case 1: Initial value of $x(0) > 0$, the condition (2.6) implies $\dot{x}(t) < 0$, thus, $x(t)$ is decreasing and moving towards $x(t) = 0$.

Case 2: Initial value of $x(0) < 0$, the condition (2.6) implies $\dot{x}(t) > 0$, thus, $x(t)$ is increasing and also moving towards $x(t) = 0$.

Of the above two cases, it is clear that regardless of the initial condition, the control (2.5) will enforce the system state $x(t)$ to approach $x(t) = 0$. This control law also guarantees a minimum decreasing (or increasing) rate of $x(t)$, so $x(t) = 0$ will be reached in finite time. The trajectory of approaching $x(t) = 0$ is known as the reaching phase.

Case 3: Initial value of $x(t) = 0$, the discontinuous part of the control law has not yet been determined, but at the time the trajectory crosses $x(t) = 0$, again it is forced back on $x(t) = 0$. Therefore, discontinuity of the control law occurs at $x(t) = 0$ at very high frequency of switching. As a result, $x(t) = 0$ can be robustly maintained and this phenomenon is called the chattering effect.

The trajectory in case 3 is the sliding phase (sliding mode). During this phase, the discontinuous part of the control law will preserve $x(t) = 0$ even in existence of consistent perturbations.

Some advantages of the 1-SMC can be listed as: theoretically accurate compensation concerning bounded matched uncertainties, reduced order of sliding equations, and finite time convergence to the sliding surface. However, the chattering phenomenon is a drawback of sliding modes. Besides that, the sliding variables converge in finite time but the state variables only converge asymptotically, and the sliding surface design is restricted to have relative degree one with respect to the control.

To reduce the disadvantages mentioned above, particularly the chattering effect, high-order sliding modes (HOSM) have been introduced [102, 159, 170, 192].

High-order SMC (HOSM)

HOSM is a movement on a discontinous set of a dynamic system. The sliding order designates the dynamics smoothness degree in the neighbourhood of the mode. If the task is to provide for keeping a constraint given by equality of a smooth function σ to zero, the sliding order is a number of continuous derivatives of σ:

$$\sigma = \dot{\sigma} = \ddot{\sigma} = \ldots = \sigma^{(r-1)} = 0, \tag{2.8}$$

creating an r-dimensional condition on the state of the dynamic system. HOSM preserves the main advantages of the 1-SMC, but is able to eliminate the chattering influence and produce higher accuracy as its realisation can produce for up to the rth order of sliding accuracy [103].

Consider a dynamic system of the form

$$\dot{x} = f(t, x) + g(t, x)u, \qquad \sigma = \sigma(t, x). \tag{2.9}$$

Here, $x \in \mathbb{R}^n, f, g$ and $\sigma : \mathbb{R}^{n+1} \to \mathbb{R}$ are unknown smooth functions, $u \in \mathbb{R}^n$ is also uncertain. The task is to provide in finite-time for exact keeping of $\sigma \equiv 0$.

The relative degree r of the system is assumed to be constant and known. The rth time derivative of σ is

$$\sigma^{(r)} = a(t, x) + b(t, x)u, \tag{2.10}$$

where $a(t, x) = \sigma^{(r)}|_{u=0}, b(t, x) = (\partial/\partial u)\sigma^{(r)} \neq 0$. It is supposed that for some $K_m, K_M, C > 0$

$$0 < K_m \leq \frac{\partial}{\partial u}\sigma^{(r)} \leq K_M \qquad \sigma^{(r)}|_{u=0} \leq C, \tag{2.11}$$

which is always true at least locally.

The Eqs. (2.9) and (2.11) imply the following differential inclusion:

$$\sigma^{(r)} \in [-C, C] + [K_m, K_M]u. \tag{2.12}$$

The major issue in implementation of HOSMs is a large amount of information required, i.e., any r-sliding controller to hold $\sigma = 0$ needs $\sigma, \dot{\sigma}, \ddot{\sigma}, \ldots, \sigma^{(r-1)}$ to be available.

Popular HOSM are super-twisting, quasi-continuous and twisting. The super-twisting sliding mode (STSM) algorithm is a combination of a second order SMC algorithm and a unique continuous sliding mode algorithm and was applied into a UAV in [44, 119, 10]. Quasi-continuous (QC) sliding mode can be made arbitrarily smooth outside of the HOSM manifold [46]. Twisting SMC have been developed for UAVs and have been introduced in liturature in [165]. The performance of the abovementioned SMCs nevertheless depends on the knowledge of the uncertainties with bounded gradients. In a practical scenario, the drones are affected by disturbances, particularly, from wind gusts while operating in outdoor environments and parameter variations, e.g., changes in mass and inertia. The overestimating of the disturbance boundary yields to the larger than necessary control gains and this is a drawback which needs further researches.

Next subsections will introduce the two HOSMs in systems with relative degree two to be developed in this project for quadcopters' low-level control systems. They are the quasi-continuous SMC (QCSM) and the twisting SMC (TSMC).

Quasi-continuous SMC

For the quasi-continuous SMC [101], it is stated that:

Provided $\beta_1, \beta_2, \ldots, \beta_{r-1}, \alpha > 0$ are chosen sufficiently large in the list order, the

controller

$$u = -\alpha\Psi_{r-1,r}\left(\sigma, \dot{\sigma}, \ddot{\sigma}, \ldots, \sigma^{(r-1)}\right) \tag{2.13}$$

is r-sliding homogeneous and provides for the finite-time stability of (2.11), (2.12). The finite-time stable r-sliding mode $r \equiv 0$ is established in the system (2.9), (2.12).

Each choice of parameters $\beta_1, \ldots, \beta_{r-1}$ determines a controller family applicable to all systems (2.9) of the relative degree r. The parameter α is chosen specifically for any fixed C, K_m, K_M, most conveniently by computer simulation, avoiding redundantly large estimations of C, K_m, K_M. Obviously, α is to be negative with $(\partial/\partial u)\sigma^{(r)} < 0$. Four quasi-continuous sliding mode controllers with $r \leq 4$ are represented as below:

1) First order ($r = 1$),

$$u = -\alpha \mathrm{sign}\sigma. \tag{2.14}$$

2) Second order ($r = 2$),

$$u = -\alpha \frac{\dot{\sigma} + |\sigma|^{1/2}\,\mathrm{sign}\sigma}{|\dot{\sigma}| + |\sigma|^{1/2}}. \tag{2.15}$$

3) Third order ($r = 3$),

$$u = -\alpha \frac{\ddot{\sigma} + 2\left(|\dot{\sigma}| + |\sigma|^{2/3}\right)^{-1/2}\left(\dot{\sigma} + |\sigma|^{2/3}\,\mathrm{sign}\sigma\right)}{|\ddot{\sigma}| + 2\left(|\dot{\sigma}| + |\sigma|^{2/3}\right)^{1/2}}. \tag{2.16}$$

4) Fourth order ($r = 4$),

$$u = -\alpha \frac{\varphi_{3,4}}{N_{3,4}}, \tag{2.17}$$

where

$$\varphi_{3,4} = \ddot{\sigma} + 3\left[\ddot{\sigma} + \left(|\dot{\sigma}| + 0.5|\sigma|^{3/4}\right)^{-1/3}\left(\dot{\sigma} + 0.5|\sigma|^{3/4}\,\mathrm{sign}\sigma\right)\right] \left[|\ddot{\sigma}| + \left(|\dot{\sigma}| + 0.5|\sigma|^{3/4}\right)^{2/3}\right]^{-1/2},$$

and

$$N_{3,4} = |\ddot{\sigma}| + 3\left[|\ddot{\sigma}| + \left(|\dot{\sigma}| + 0.5|\sigma|^{3/4}\right)^{2/3}\right]^{1/2}.$$

The control is a continuous function of time everywhere except the r-sliding set (2.8). Those control schemes characterise considerably less chattering, but their complication is a concern for them to be implemented.

Twisting SMC controller

Consider an uncertain dynamic system with relative degree two in the following differential equation:

$$\ddot{\sigma}(t) = a(t,x) + b(t,x)u, \tag{2.18}$$

where $\left|a(t,x)\right| \leq C, 0 \leq K_m \leq b(t,x) \leq K_M$ are unknown, $u \in \mathbb{R}$ is the control.

The twisting controller can be applied to 2-sliding mode $\sigma = \dot{\sigma} = 0$ (2.18) and can attract the trajectories in finite time. The controller is defined by [102]:

$$u = \begin{cases} -\mu\alpha \text{sign}(\sigma) & \text{if } \sigma\dot{\sigma} \leq 0 \\ -\alpha \text{sign}(\sigma) & \text{if } \sigma\dot{\sigma} > 0, \end{cases} \tag{2.19}$$

where $\mu < 1$ is a fixed positive number and $\alpha > 0$ is the control gain.

2.1.3 Other control strategies

Proportional, Integral and Derivative (PID) controllers and their developments are preferred on UAVs because of their simple structure, good performance and the possibility of tuning characteristics. PID is designed by taking the attitude errors, calculated as the difference between a state and its corresponding referent values. Enhancements of PID controllers for quadcopters [221, 21, 118] attempt to minimise the error by adjusting the process control inputs with fast response and small overshoot, but their robustness is not yet satisfied [8].

Optimal Control Algorithms are chosen for the quadcopter to get the best value for a performance index and minimise a selected variable. Some exemplary

optimal controllers are a combination of H_∞, L_1, and Linear Quadratic Regulator (LQR) with Kalman filter. H_∞ control for quadcopter attitude was reported in [137, 163, 4]. The control methodology was designed to provide good performance and robustness against the uncertainties while improving smoothness. The proposed strategy is able to deal with external disturbances and parametric uncertainties applied to the rotational motion robustly. However, overshoot and transient time problems are hardly solved by using this control scheme. A design and implementation of an L_1-optimal control for a quadcopter are introduced in [172]. The controller is selected to minimise the L_∞-gain. A robust nonlinear controller is achieved with the rejection of small persistent disturbances.

Linear Quadratic Regulator (LQR) and Linear Quadratic Gaussian (LQG) can be used for the quadcopter dynamic system to minimise a predetermined cost function. In [145, 135], LQR and LQG for quadrotors show their good performance. A simple path-following LQR in [36] reveals that accurate path following was achieved in simulation using optimal real-time trajectories, under wind gusts and disturbances. However, the controller lost track after each obstacle avoidance task. A disadvantage of those methods is their complicated computation in solving the matrix algebra.

Backstepping control method has its advantage in less computational resources and well disturbance rejection. This approach has been suggested for quadrotor control in [124, 80, 57]. The paper [124] proposed a backstepping controller for quadcopter in which the model is divided into underactuated, fully actuated and propeller subsystems. In [80], this approach was applied in quadrotor attitude stabilisation. The integrator backstepping control was developed in [57] to enhance the system robustness. Good tracking was achieved by using the above-mentioned control scheme with small steady-state errors, time response and overshoot. However, the control

performance in dealing with parametric uncertainty had not been examined.

Model predictive control (MPC) algorithms are designed for the trajectory tracking problem of quadcopters. In [23], a learning-based MPC is proposed for quadcopters by piecewise affine equations, where attitude and planar control are separated and MPC is combined with robust control. The authors in [139] built a trajectory generator to be used as a diminishing horizon model predictive controller. This method extends the state of the art approach by calculating state interception trajectories in real time, using sophisticated optimisation techniques to explicitly include input constraints in the trajectory generation problem. Generalised Predictive Control [130] is proposed for constrained trajectory tracking using a combination of geometric control and constrained generalised predictive control.

Intelligent control algorithms are developed to deal with uncertainties, and mathematical complexities of the systems, in which fuzzy logic and neural networks are typical. The fuzzy controller was applied in [127, 39, 61] to manage the quadcopter position and orientation with good response and robustness. Artificial neural networks algorithms were developed for a quadrotor in [56, 24] to stabilise the drone taking into account modeling errors, measurement noises, and environmental disturbances. The method showed control performance improvements while reducing the computational time. A neural network with an adaptive scheme was applied in [22] for stabilisation of the quadrotor under a sinusoidal disturbance. Simulation results showed good control performance without having any weights drift.

Hybrid control Algorithms For the reason that any single control scheme has its limitations that others can be used to compensate for, there are some combination of one or more philosophies have been designed for UAVs. A combination of backstepping and neural networks was presented in [123], in which the use of

backstepping gains good tracking performance as well as maintaining stability, and neural networks are employed to compensate for the dynamic uncertainties. Thus, even though a precise dynamic model is not required, greater adaptability and robustness can be achieved from this proposed approach. The work in [147] designs a control scheme using a Gain scheduling and Lyapunov based model reference adaptive control. Results showed this proposed scheme is capable of stabilising the oscillations produced from the unstable PID attitude control, and a high-level control system based on a switching automaton is proposed to ensure the safety of the aerial manipulation. In [66, 100], hybrid controllers were developed to deal with complex aerobatic maneuvers, and high angle transitions between flight modes. The control schemes are robust to switching conditions as each flight mode has almost global stability features, making a design of a complex maneuver. These approaches, however, still have some difficulties with complicated reachability and are restricted by the drone dynamic uncertainties.

2.2 Path generation techniques for UAVs

Over the last decade, there has been extensive research in path planning for UAVs, typically A^*, D^*, Rapidly Exploring Random Tree (RRT), Roadmap and Probabilistic Methods, optimisation, and potential field approaches. Recently, artificial intelligence and machine learning also involve the path creation problem. The above-listed methods for UAV path planning are reviewed and provided in the following details.

Node Based Optimal Algorithms A^* and D^* [131, 52, 122, 212] are typical representatives of the node-based optimal as well as heuristic-based algorithms, which focus on dynamic problems. The strength of these algorithms lies in the ability to judge or evaluate the best path from point to point so they can provide flyable paths on a square or cubic grid of occupied or unoccupied cells. To develop

the A^*-based algorithm in tri-dimensional workspaces, the θ^* and then ϕ^* were proposed [38, 141, 58]. These algorithms are capable of handling orographic obstacles and operating in urban environments [40]. The strength of the A^* algorithm lies in the ability to judge or evaluate the best path from point to point so it can provide flyable paths for UAV. This strength is also the weakness of the star-based path planning as it is not able to find the real shortest path between the nodes so that their solutions are suboptimal [40]. Also, the generated paths are not smooth. Additionally, it performs inaccurately in harsh conditions, and hence, is ineffective for some critical path planning problems [131, 41].

A generic cost function for A^* algorithm is the following:

$$f(i) = s(i) + h(i), \tag{2.20}$$

where $s(i)$ is the known cost function of the path from the initial point to the ith point, $h(i)$ is the heuristic estimation of cost from the ith point to the target. Here, $s(i)$ implements a priority of the points to be passed through so that the value of f will be lower, resulting in the shorter length of the overall path.

Rapidly Exploring Random Tree and its developments employ a random exploration from the centre of the given workspace and expanding a tree through a stochastic sampling of the configuration area. This approach has proven its feasibility in the probabilistic scope; as a result, it is able to generate a feasible path considering the robot's dynamic constraints [97, 98, 168, 202, 201, 20]. However, these methods project real data into a polygonal environment and plan in a geometric space, thus making the method sensitive to sensor noise [146]. In addition, the rate of convergence or optimality has not been reported so far [67].

Roadmap Methods employ a designed graph (roadmap) to fit into the workspace. Based on the roadmap, a path will be generated to follow predetermined topographical properties. The roadmap method are categorised into Visibility Graph [3],

Voronoi Roadmap [2, 125], Freeway (Highway) Method [82], and Silhouette Method [35]. Originally, they were used to produce an exact solution to the point vehicle problem completely and optimally in only two-dimensional space. Since the minimum-length path comes arbitrarily close to obstacles many times in a typical path, these approaches offer no safety buffer to prevent collisions in the case of systems with uncertainty in their position [67]. Voronoi Roadmap was introduced to solve the path following problem, the minimum distance path considering obstacle avoidance [151]. The algorithm is feasible but not optimal in real-time [67]

Probabilistic Approaches include two main sub-methods, i.e., stochastic search and probabilistic roadmap (PRM) methods. The stochastic search approaches aim to escape from local minima that are found in a potential field [26, 185]. These approaches are able to find paths, but their feasibility is not easy to confirm [67]. The PRM [204, 93, 199] is the most persuasive because it allows a solution to problems of stochastic complexity and dimension, which is convergent in the probabilistic perspective. Drawbacks of PRM include low convergence rate, insufficient optimal and smooth paths [67, 121].

Potential Field Methods (PFM) are one of the most preferable in path planning applications. These methods employ assigning value computed via a potential function to the free workspace, and simulating the vehicle's reaction as it manoeuvres towards the lowest potential. So far, there has been extensive research in potential-based path planning for UAVs such as [181, 72, 27, 116, 115, 31, 126]. Advantages of the potential field methods are low computational complexity [67] but may be infeasiable because of the local minima trap. Among them, artificial potential field-based (APF) approaches have proved to generate feasible paths in complex environments thanks to their advantagous cost at time consuming [126]. Recently, APF has been developed for UAV's path planning to cope with problems of local minima, narrow passage, and moving obstacles [202, 212]. However, they are com-

putationally infeasible when the number of obstacles increase, leading to collision avoidance becoming difficult to guarantee [126].

A generic potential function for path planning problems potential at any space in the environment is represented by the resultant force P as follow [83, 162]:

$$P = P_0 + P_g, \tag{2.21}$$

where P_0 is the maximum of the potentials due to individual obstacles, P_g is the attractive force generated by the goal.

Optimal path planning algorithms also contribute to the development of the topic [156]. The paper [78] proposed a feasible path with smaller travel distance which is generated from an optimal control problem. In [32], the optimal control theory is used to support APF while the pathfinding problem is reconstructed into the constrained optimisation. On the other hand, an optimal path planning for UAVs using Genetic Algorithm (GA) to travel all control points in an optimal way has been found to be more productive while avoiding electronic detectors [182]. The GA also can support the A^* algorithm in real-time optimal path planning for UAV/UGV systems [106]. Together with their significant improvements, PSO-based path planning has been employed widely to find safe, feasible and flyable paths for UAVs to operate in different threat environments [59, 156, 62, 63]. Since advantages of these approaches have been well proven in the literature and simulations, the experimental results have not been reported.

Recently, the rapid development of Artificial Neural Network (ANN) and machine learning has been employed to provide a better design for trajectories creation, such as [76, 65, 6, 50, 211]. The Geometric Learning Algorithm (CGLA) is proposed to produce optimal paths for a single UAV [211] and multi-UAVs's [210], considering collision avoidance and information exchange [210]. The paper [6] introduced a deep neural network approximation for a risk-aware resampling technique for quad-

copters. Even though these advanced methods are feasible to generate precise paths for UAVs in a large search space, those approaches however require large training data as the input [33, 215].

2.3 Formation control approaches

In the last decade, solving robotic and UAV formation problems, in particular, have grown to be the leading motivation for researchers. The reason behind this is that a UAV team working cooperatively will achieve greater performance and better effectiveness in some challenging missions. From the control mechanism perspective, formation problems have been studied and can be categorised as consensus-based, artificial potential function-based, leader-follower (L-F), behaviour-based and virtual structure (VS). From the control structure perspective, the formation control strategies are classified as centralised, decentralised and distributed control.

Consensus-based approaches The consensus problem is to gain an agreement regarding a particular quantity of interest that depends on the state of a multi-agent system (MAS) [94, 205, 48]. The solution aims to achieve a common state of all members in the MAS. This problem can be divided into unconstrained and constrained consensus problems. The state of all agents in the constrained problem has to converge to a common objective function asymptotically, but the state of all vehicles asymptotically become the same without any objective function in the latter problem. A simple goal for each agent in the unconstrained consensus is to minimise a cost function, presented as:

$$U_i(x) = \sum_{j \in N_i} \left\| x_j - x_i - d_{ij} \right\|^2, \tag{2.22}$$

where x_i represents the position of agent i, d_{ij} is a desired inter-agent relative position vector, and agent j is the neighbour of agent i.

Consensus has become an attractive problem in the study of swarms of multiple flying vehicles. The paper [94] proposed a cooperative formation control of a UAVs group considering the collision avoidance problem. A consensus-based protocol and a leader-follower structure are used to maintain the formation shape while the collision avoidance problem is solved by a potential artificial scheme. The paper [48] presented a control analysis and design for time-varying UAV formation. The formation algorithms are introduced with variation for each agent velocity to achieve the configuration. Consensus-based approaches are then applied to cope with the reconfiguration control problem. Another second-order consensus control for the formation of UAVs is introduced in [205], where the position and velocity variables of the adjacent neighbor pairs are used to create control commands. The guidance and control schemes are coupled to preserve a defined shape, while respectively handling the position and attitude. The inter-vehicle collisions are the main drawback of the consensus strategies [108].

Leader-follower approaches Having the advantage of simplicity, since the trajectory of the swarm is apparently assigned to the leader(s), the formation of L-F configuration is expressly studied and improved [132, 79, 175, 174, 150, 208]. Within the L-F configuration, one vehicle is nominated as the leader, while the others are determined as followers. The states of the leader establish the entire group variables; the other agents will operate accordingly to the leader states. The objective of the approach is to guarantee that:

$$\begin{aligned} &\lim_{t\to\infty}(p_i(t) - p_r(t)) = \delta_i, \\ &\lim_{t\to\infty}(v_i(t) - \dot{p}_r(t)) = 0, \quad i = 0, 1, \ldots, N, \end{aligned} \tag{2.23}$$

where p_r is the global reference position of the formation, p_i and v_i are the position and velocity of the centre of gravity of the ith UAV, N is the total number of the followers (0 is assigned to the leader), δ_i is a position offset.

Despite the advantage of the L-F formation, the whole flock may be damaged by the failure of the leader [15, 49]. Furthermore, it is challenging for the leader-follower strategy to reconfigure the formation [108].

Virtual structure approaches In the virtual leader or virtual structure approaches [95,14,107], the whole formation is managed as a unique virtual rigid body structure. In contrast to the L-F where the explicit leader is appointed in advance, each vehicle on the swarm will follow a moving point, which can be considered as the virtual head/leader. The guidance method for VS is a simpler approach compared to others because each member will operate as an individual agent. The following three steps derive the control law of each agent:

Step 1. Defining the dynamics of the virtual structure,

Step 2. Converting the structure motion into the desired trajectory of each individual UAV,

Step 3. Determining a suitable control law for each individual UAV based on its desired path.

However, the VS based formation is only able to present the synchronised movements, which are not fully distributed due to the requirement of each UAV to track its own trajectory. Besides, it would be a challenge to solve the collision avoidance tasks. A massive amount of computation and communication of the central controller is required. Moreover, the formation may be destroyed under some uncertainties, or the short processing time required for the tracking mission [108].

Behaviour-based approaches [12, 176, 90, 195] The behavior-based strategies inherit the concept of the formation behaviors of animals in nature, by fusing the data from the equipped sensors to maximise or minimise a specific cost function in

order to reach some common targets. This approach is a combination of the outputs of various controllers through suitable weight coefficients that set the relative priority among them. Some behaviors are designed for each agent in the swarm, including trajectory tracking, goal seeking, formation keeping, and obstacle avoidance. These approaches are suitable for uncertain environments and/or multiple objectives of the UAVs. However, the behaviors may destructively interfere with each other, and it is too complicated in a mathematical theoretic analysis [108].

Artificial potential function-based approaches [47, 89] The problem is specially designed for multiple vehicles in a flock with limited sensing ranges. This approach applies the negative gradient of a mixture of attractive and repulsive potential functions as control inputs to satisfy the convergence while maintaining the collision-free properties, respectively. A typical potential function is defined as [47]:

$$\phi = \sum_{i=1}^{N} (\gamma_i + 0.5\beta_i), \tag{2.24}$$

where γ_i and β_i are the goal and related collision avoidance functions.

The major weakness of this strategy is the appearance of equilibrium, where the summary vector field vanishes and the agents may be ambushed at wrong equilibrium locations [108].

Centralised control strategies [88,149] employ a single controller to manage the entire swarm according to the collected data from all members. Thanks to its simple, easy achieving and efficient approach, the centralised control can be a useful strategy for a team of a small number of agents. However, for large-scale practical systems, which require a greater computational capacity and communication bandwidth, this method would be vulnerable to any fault of the central processor [108, 7].

Decentralised control strategies [89, 206, 169] are recommended to cope with the robustness problem, the difficulty and complexity of the study of the entire system. The approaches are likewise the most exciting research problems compared to the others. The main idea is that each agent creates a decision based on the data collected from the other agents as well as from the equipped sensors. Hence, the overall formation is not dependent on a single controller but on several controllers, which contain decentralised controllers executed on each module.

Distributed control strategies are the extended concept of the decentralised control by sharing local data. The control problems are usually related to the decentralised control design structure with communication constraints. Significant efforts have been dedicated to solve the distributed control problems over the last decade, such as [108, 14, 208, 117].

Despite existing approaches for multiple quadcopters to perform a desired formation having been well discussed, problems dealing with uncertainties, collision avoidance, reliable communication are still the tasks to be solved. The main contribution of this research is to provide a cooperative formation scheme based on the virtual head method. The formation centroid will be used to track a pre-planned trajectory while carrying out some harsh missions in a limited time, taking into account parametric uncertainties and external disturbances.

2.4 Summary

This chapter has comprehensively reviewed up-to-date approaches that are being proposed in recent publications. There are three different parts to form the whole control architecture for multiple-UAVs formation flight, including the individual on-board controller (low level control) of UAVs, path planning and formation.

Chapter 3

Adaptive robust control design for quadcopters

3.1 Introduction

This chapter presents the developments of sliding mode controllers for quadcopter UAV. It includes three parts, which are the background of sliding mode controllers, control designs for quadcopters, and results of the proposed control schemes.

The first part introduces the dynamic modelling of a quadrotor UAV in the presence of nonlinearity, external disturbances, parametric uncertainties, and coupling effect.

The main part introduces two robust schemes named the adaptive second-order quasi-continuous SMC (AQCSM) and the adaptive twisting SMC (AdTSM). The control design includes the selection of the sliding manifolds, followed by the development of the two sliding mode controllers with an adaptive gain. Stability of the control systems is analysed by using a global Lyapunov functions for convergence of both the sliding dynamics and adaptation scheme.

The chapter ends with extensive simulation studies and comparisons with experimental data to illustrate the performance of the proposed control developments.

3.2 Quadcopter model

To represent the position and orientation of rigid-body systems, two coordinate frames are used: the inertial frame, and the body frame. For this, an earth-fixed North, East, Down (NED) coordinate frame $\{E\} = (x_E, y_E, z_E)$ is assumed to be an inertial frame given the typical distance and travel time of quadcopters.

The model of the quadcopter used in this work is illustrated in Figure 3.1. The body frame of a quadcopter (x_B, y_B, z_B) is specified by the orientation of the quadcopter with the rotor axes pointing in the positive z_B direction and the arms pointing in x_B and y_B directions.

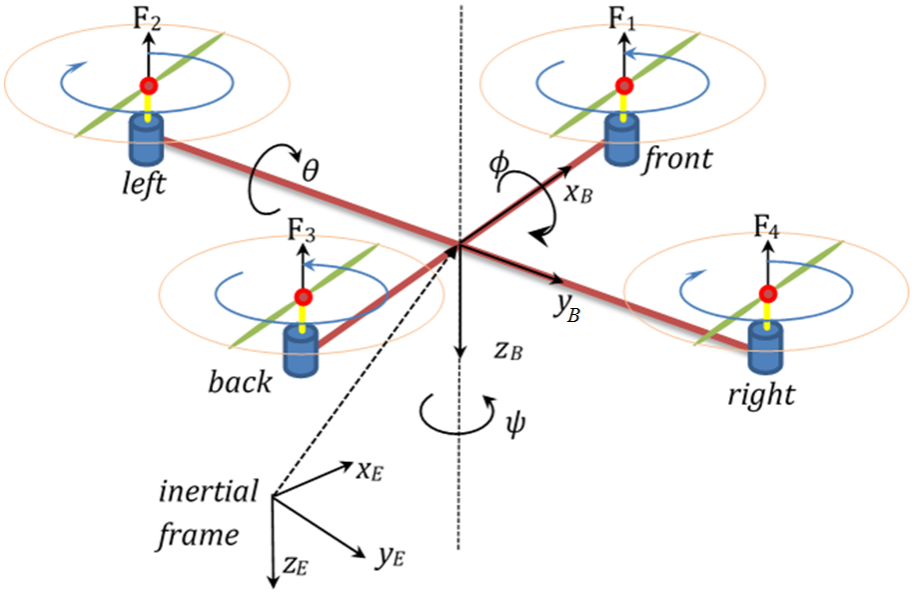


Figure 3.1 : A schematic diagram of quadcopter

The quadcopter is assumed to be symmetric around the z_B axis with four motors $M_i, i \in 1, 2, 3, 4$ located at the same distance from the quadcopter's centre of mass. Rotational velocities of the propellers are denoted by Ω_i. It is assumed that M_1 and M_3 are rotating counter-clockwise and M_2 and M_4 are rotating clockwise.

Without loss of generality, the quadcopter is assumed to be a rigid body structure, it has its own body frame $\{B\}$. The position of the origin of $\{B\}$ with respect to $\{E\}$ is denoted by $P = (x, y, z)^T$, the velocity of $\{B\}$ with respect to $\{E\}$ by $\nu = (U, V, W)^T$.

The orientation of a rigid body is described by the roll, pitch, and yaw angles corresponding to its rotations around the x_B, y_B and z_B axes. Denoting those angles

as $\Theta = (\phi, \theta, \psi)^T$, their rates are then given by $\dot{\Theta} = (\dot{\phi}, \dot{\theta}, \dot{\psi})^T$. The rates relate with angular velocities, $\omega = [p, q, r]^T$, by the following transformation:

$$\omega = H\dot{\Theta}, \tag{3.1}$$

where H is given by:

$$H = \begin{bmatrix} 1 & 0 & -s_\theta \\ 0 & c_\phi & c_\theta s_\phi \\ 0 & -s_\phi & c_\theta c_\phi \end{bmatrix}, \tag{3.2}$$

in which $s_x = sin(x)$ and $c_x = cos(x)$. The following rotational matrix $R \in SO(3)$ transform a vector in $\{B\}$ to $\{E\}$ is described by:

$$R = \begin{bmatrix} c_\psi c_\theta & c_\psi s_\theta s_\phi - s_\psi c_\phi & c_\psi s_\theta c_\phi + s_\psi s_\phi \\ s_\psi c_\theta & s_\psi s_\theta s_\phi + c_\psi c_\phi & s_\psi s_\theta c_\phi - c_\psi s_\phi \\ -s_\theta & c_\theta s_\phi & c_\theta c_\phi \end{bmatrix}. \tag{3.3}$$

Since the focus is on the attitude control so only torque components that cause changes in the orientation are considered. They include torques caused by thrust forces τ, body gyroscopic effects τ_b, propeller gyroscopic effects τ_p, and aerodynamic friction τ_a. The torque τ consists of three components corresponding the roll, pitch and yaw rotations, $\tau = [\tau_\phi \ \ \tau_\theta \ \ \tau_\psi]^T$. They are given by:

$$\tau_\phi = l(F_2 - F_4), \tag{3.4}$$

$$\tau_\theta = l(-F_1 + F_3), \tag{3.5}$$

$$\tau_\psi = b(-F_1 + F_2 - F_3 + F_4), \tag{3.6}$$

where l is the distance from the motor to the UAV centre of mass and b is the drag factor. The body gyroscopic torque τ_b is given by:

$$\tau_b = -S(\omega)I\omega, \tag{3.7}$$

where $S(\omega)$ is a skew-symmetric matrix,

$$S(\omega) = \begin{bmatrix} 0 & -r & q \\ r & 0 & -p \\ -q & p & 0 \end{bmatrix}. \tag{3.8}$$

The propeller gyroscopic torque τ_p is determined as:

$$\tau_p = \begin{bmatrix} I_r \Omega_r q \\ -I_r \Omega_r p \\ 0 \end{bmatrix},$$

where I_r is the inertial moment of rotor, $\Omega_r = -\Omega_1 + \Omega_2 - \Omega_3 + \Omega_4$ is the residual angular velocity of rotor in which Ω_k denotes the angular velocity of the propeller k (k=1,2,3,4). Finally, the aerodynamic friction torque τ_a is given by:

$$\tau_a = k_a \omega^2, \tag{3.9}$$

where k_a depends on aerodynamic friction factors, $k_a = \text{diag}[k_{ax}, k_{ay}, k_{az}]$. Given those torque components, the attitude dynamic model of the quadcopter is described as:

$$I\ddot{\Theta} = \tau_b + \tau + \tau_p - \tau_a, \tag{3.10}$$

where $I = \text{diag}[I_{xx}, I_{yy}, I_{zz}]$ is the inertia matrix when the quadrotor is assumed to be symmetrical.

In this system, the gyroscopic and aerodynamic torques are considered as external disturbances. Thus, the control inputs mainly depend on torque τ and from (3.4), (3.5) and (3.6), they can be represented as:

$$\begin{bmatrix} u_\phi \\ u_\theta \\ u_\psi \\ u_z \end{bmatrix} = \begin{bmatrix} \tau_\phi \\ \tau_\theta \\ \tau_\psi \\ F \end{bmatrix} = \begin{bmatrix} 0 & l & 0 & -l \\ -l & 0 & l & 0 \\ -c & c & -c & c \\ 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \end{bmatrix}, \tag{3.11}$$

where $F = F_1 + F_2 + F_3 + F_4$ is the UAV lift, u_z represents the total thrust acting on the four propellers and u_ϕ, u_θ and u_ψ respectively represent the roll, pitch and yaw torques, c is a force-to-torque scaling coefficient. As only the attitude of quadcopter will be controlled, u_z is assumed to balance with the gravity. Consequently, the dynamics of quadcopters can be represented in the following form for attitude control:

$$\ddot{\Theta} = I^{-1}\left(-S(\omega)I\omega + u + d\right), \tag{3.12}$$

where $u = [u_\phi, u_\theta, u_\psi]^T$ is the input vector and $d = [d_\phi, d_\theta, d_\psi]^T$ is the disturbance vector. For small rotation angles of the quadrotor, ω is approximate to $\dot{\Theta}$, so the second-order nonlinear dynamic equations of the three Euler angles are derived as:

$$\ddot{\phi} = \frac{1}{I_{xx}}\left[(I_{yy} - I_{zz})qr + u_\phi + d_\phi\right], \tag{3.13}$$

$$\ddot{\theta} = \frac{1}{I_{yy}}\left[(I_{zz} - I_{xx})pr + u_\theta + d_\theta\right], \tag{3.14}$$

$$\ddot{\psi} = \frac{1}{I_{zz}}\left[(I_{xx} - I_{yy})pq + u_\psi + d_\psi\right]. \tag{3.15}$$

The quadcopter dynamics can be then represented as follows:

$$\begin{cases} \dot{X}_1 = X_2 \\ \dot{X}_2 = I^{-1}\left[f(X) + u + d\right], \end{cases} \tag{3.16}$$

where $X_1 = \Theta$, $X_2 = \dot{\Theta}$, $X = [X_1, X_2]^T$ is the state vector, $f(X)$ is the matrix represented as

$$f(X) = \begin{pmatrix} (I_{yy} - I_{zz})qr \\ (I_{zz} - I_{xx})pr \\ (I_{xx} - I_{yy})pq \end{pmatrix}. \tag{3.17}$$

In this system, the following assumptions are made:

A.1 The quadcopter structure is rigid and symmetric. The propellers are rigid.

A.2 The signals Θ and $\dot{\Theta}$ can be measured by on-board sensors.

A.3 The reference trajectories and their first and second time derivatives are bounded.

A.4 The velocity and the acceleration of the quadcopter are bounded.

A.5 The orientation angles are limited to $\phi \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$, $\theta \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$ and $\psi \in [-\pi, \pi]$.

A.6 The rotational speeds of rotors are bounded.

3.3 Sliding manifold

A design of a sliding manifold for sliding mode controllers of quadcopters will be specified in this section. The mathematical model of the quadcopters was represented in Eq. (3.12). The sliding function determining the system's equivalent dynamics is presented as:

$$\sigma = \dot{\mathbf{e}} + \Lambda \mathbf{e}, \tag{3.18}$$

where $\Lambda = \text{diag}(\lambda_\phi, \lambda_\theta, \lambda_\psi)$ is a positive definite matrix to be designed, and $\mathbf{e}$ is the control error:

$$\mathbf{e} = \Theta_d - \Theta, \tag{3.19}$$

where Θ and Θ_d are respectively the actual and the desired values of quadcopters' attitude.

Taking the derivative of σ, one has:

$$\dot{\sigma} = \ddot{\Theta} - \ddot{\Theta}_d + \Lambda \dot{\mathbf{e}}. \tag{3.20}$$

For small angular rotations of the quadcopter, ω can be approximated to $\dot{\Theta}$ [217]. Substituting $\ddot{\Theta}$ (3.12) into (3.20) yields:

$$\dot{\sigma} = -\ddot{\Theta}_d + \Lambda \dot{\mathbf{e}} + I^{-1}[-S(\omega) I \omega + u + d]. \tag{3.21}$$

In the next two sections, the control signals u_ϕ, u_θ and u_ψ in (3.12) are used to control the three angles $\{\phi, \theta, \psi\}$ to reach the reference value $\Theta_d = \{\phi_d, \theta_d, \psi_d\}^T$.

3.4 Adaptive quasi-continuous sliding mode control

In the HOSM control, the quasi-continuous (QC) SMC [46] introduces the capability of maintaining the properties of the first order SMC while creating smooth responses. Its performance however depends on the knowledge of disturbance boundaries which are not always available. In practice, the quadcopter may be subject to various disturbances and uncertainties such as wind gusts and modelling errors that may downgrade the control performance. To address this concern, the second-order sliding mode (SOSM) controller with an adaptive gain has been applied to drive the sliding variable and its derivative to zero in the presence of bounded disturbances [179].

An adaptive quasi-continuous second-order sliding mode scheme is proposed to control the attitude of quadcopters subject to nonlinear dynamics, strong coupling, high uncertainties and disturbances with unknown boundaries. Here, the quasi-continuous SMC retains the advantage of robustness while attenuating the control chattering and facilitating the implementation.

3.4.1 QCSM control design

The conventional second-order QCSM in Eq. (2.15) is used in this design under the following form:

$$u = -\alpha \frac{\dot{\sigma} + |\sigma|^{1/2} \operatorname{sign}(\sigma)}{|\dot{\sigma}| + |\sigma|^{1/2}}, \tag{3.22}$$

where α is the control gain to be adjusted. The control is continuous everywhere apart from the origin where $\sigma = \dot{\sigma} = 0$.

Since I is symmetric and positive definite, the following Lyapunov function is

chosen to avoid the inversion of the inertia matrix:

$$V_0 = \frac{1}{2}\sigma^T I\sigma. \tag{3.23}$$

Taking the time derivative of V gives

$$\dot{V}_0 = \frac{1}{2}\left(\dot{\sigma}^T I\sigma + \sigma^T I\dot{\sigma}\right) + \frac{1}{2}\sigma^T \dot{I}\sigma = \sigma^T\left(I\dot{\sigma} + \frac{1}{2}\dot{I}\sigma\right). \tag{3.24}$$

By substituting $\dot{\sigma}$ from (3.24) to (3.21), one has

$$\dot{V}_0 = \sigma^T\left(-I\ddot{\Theta}_d + I\Lambda\dot{e} - S(\omega)I\omega + u + d + \frac{1}{2}\dot{I}\sigma\right). \tag{3.25}$$

Let $I = I_0 + \Delta I$, where I_0 and ΔI represent the nominal and uncertain parts of the inertia matrix. According to A1, one has $\dot{I} = 0$. Equation (3.25) becomes

$$\begin{aligned}\dot{V}_0 &= \sigma^T\{ - S(\omega)\Delta I\omega - \Delta I\ddot{\Theta}_d + \Delta I\Lambda\dot{e} + d + \frac{1}{2}\dot{I}\sigma + u - S(\omega)I_0\omega - I_0\ddot{\Theta}_d + I_0\Lambda\dot{e} \\ &= \sigma^T\{\Delta P + u + P\},\end{aligned} \tag{3.26}$$

where

$$\Delta P = -S(\omega)\Delta I\omega - \Delta I\ddot{\Theta}_d + \Delta I\Lambda\dot{e} + d, \tag{3.27}$$

$$P = -S(\omega)I_0\omega - I_0\ddot{\Theta}_d + I_0\Lambda\dot{e}. \tag{3.28}$$

Let $\Xi = [\Xi_1, \Xi_2, \Xi_3]^T$ denote the sum of ΔP and P. Since the disturbance d and uncertain parameter ΔI are bounded, from (3.27) and (3.28), it can be seen that Ξ is also bounded, i.e., $|\Xi_i| \leq \Xi_{M,i},\ i = 1, 2, 3$. Consider system (3.12) with the sliding variable $\sigma(\omega, t)$ as in (3.18). From assumptions A1-A6 in section 3.2, the sliding motion on the manifold is achieved by the controller (3.22) if the gain α_i can be selected such that [161]:

$$\alpha_i \geq \Xi_{M,i}. \tag{3.29}$$

However, the bound $\Xi_{M,i}$ is not easy to evaluate in practice. In addition, the chattering will increase if a high value of α_i is chosen. Thus, the goal now is to

drive the sliding variable σ and its derivative $\dot{\sigma}$ to zero in finite time by means of quasi-continuous SMC without overestimation of the control gain.

3.4.2 Adaptive QCSM design

The proposed gain-adaptation law is supposed to minimise the chattering phenomenon while driving σ and $\dot{\sigma}$ to zero even in the presence of disturbances. For initial conditions $\omega_i(0), \sigma_i(0)$, and $\alpha_i(0) > 0$, the reaching and sliding on the manifold is globally achieved in finite time by controller (2.15) with the following adaptive gain [157]:

$$\dot{\alpha}_i = \begin{cases} \bar{\omega}_i \big|\sigma_i(\omega, t)\big| \operatorname{sign}(|\sigma_i(\omega, t)| - \epsilon_i) & \text{if } \alpha_i > \alpha_{m,i} \\ \eta_i & \text{if } \alpha_i \leq \alpha_{m,i} \end{cases}, \tag{3.30}$$

where $\bar{\omega}_i$, ϵ_i, η_i are small positive constants, and $\alpha_{m,i}$ is a threshold of the adaptation.

To analyse the stability of the proposed controller, let us first define a global Lyapunov function candidate for σ and α as:

$$V(\sigma, \alpha) = V_0 + \sum_{i=1}^{3} \frac{1}{2\gamma_i}(\alpha_i - \alpha_{M,i})^2, \tag{3.31}$$

where V_0 has been defined in Eq. (3.23), γ_i is a positive constant and $\alpha_{M,i}$ is the maximum possible value of the adaptive gain α_i. The derivative of the Lyapunov function (3.31) is obtained as

$$\dot{V}(\sigma, \alpha) = \dot{V}_0 + \sum_{i=1}^{3} \frac{1}{\gamma_i}(\alpha_i - \alpha_{M,i})\dot{\alpha}_i. \tag{3.32}$$

Taking $\dot{V}_0$ from (3.26) and $\dot{\alpha}_i$ from (3.30), equation (3.32) under the control law (2.15) becomes

$$\begin{aligned} \dot{V}(\sigma, \alpha) = \sum_{i=1}^{3} \sigma_i \left[\Xi_i - \alpha_i \left(\frac{\dot{\sigma}_i + |\sigma_i|^{1/2} \operatorname{sign}(\sigma_i)}{|\dot{\sigma}_i| + |\sigma_i|^{1/2}} \right) \right] + \\ + \sum_{i=1}^{3} \frac{1}{\gamma_i}(\alpha_i - \alpha_{M,i})\bar{\omega}_i |\sigma_i| \operatorname{sign}(|\sigma_i| - \epsilon_i). \end{aligned} \tag{3.33}$$

When σ_i is slowly time-varying, $\dot{\sigma}_i(t)$ is very small and can be negligible. Equation (3.33) then becomes

$$\dot{V}(\sigma,\alpha) = \sum_{i=1}^{3} \sigma_i \left[\Xi_i - \alpha_i \text{sign}(\sigma_i)\right] + \sum_{i=1}^{3} \frac{1}{\gamma_i}(\alpha_i - \alpha_{M,i})\bar{\omega}_i|\sigma_i|\,\text{sign}(|\sigma_i| - \epsilon_i). \quad (3.34)$$

It can be seen that $\dot{V} \leq 0$ given (23) and $\alpha_i \leq \alpha_{M,i}$ [157].

3.5 Adaptive twisting sliding mode control

In the HOSM control framework, most popular are twisting controllers [159] and their modified versions like super-twisting [160, 70], adaptive super-twisting [179], and accelerated twisting [54]. Owing to their advantages, these HOSM techniques have been applied to UAV control [217,164]. However, these control laws are indeed complicated and would require some simplification. To this end, the one-stage algorithm of the accelerated twisting sliding mode (ATSM), where the control gain is modified to be always greater than an exponential function of the sliding function magnitude, appears not too complicated but can guarantee accelerated finite-time, or at least, fixed-time convergence [54]. Motivated by the work therein, an adaptive scheme is developed to be able to adjust the control gain of the twisting control law and apply it to control the attitude of quadcopters in harsh conditions with nonlinearity, external disturbances, uncertain dynamics and strong coupling.

Given the desired angle reference $X_{1d} = \{\phi_d, \theta_d, \psi_d\}^T$, the overall control law is proposed as:

$$u(t) = u_{eq}(t) + u_D(t), \quad (3.35)$$

where $u_{eq}(t) = (u_{eq,i})^T$ and $u_D(t) = (u_{D,i})^T$, $i = 1, 2, 3$, are respectively the equivalent control and the discontinuous part containing switching elements. In this system, the sliding surface equation is chosen as in Eq. (3.18) and the control error is $\mathbf{e} = X_1 - X_{1d}$.

3.5.1 Equivalent control design

The equation (3.18) can be rewritten for the attitude sliding surface as:

$$\sigma = (\dot{X}_1 - \dot{X}_{1d}) + \Lambda(X_1 - X_{1d}). \tag{3.36}$$

Taking the time derivative of σ gives:

$$\dot{\sigma} = (\ddot{X}_1 - \ddot{X}_{1d}) + \Lambda(\dot{X}_1 - \dot{X}_{1d}), \tag{3.37}$$

or

$$\dot{\sigma} = -\ddot{X}_{1d} + \dot{X}_2 + \Lambda\dot{e}. \tag{3.38}$$

Substituting $\ddot{X}$ from (3.16) to (3.38) yields:

$$\dot{\sigma} = -\ddot{X}_{1d} + I^{-1}\left[f(X) + u\right] + \Lambda\dot{e}. \tag{3.39}$$

When the sliding mode has been induced, u can be considered as the equivalent control u_{eq}. By driving the derivative of sliding surface to zero, the equivalent control rule can be obtained as follows:

$$u_{eq} = I\left(\ddot{X}_{1d} - \Lambda\dot{e}\right) - f(X). \tag{3.40}$$

3.5.2 Discontinuous control design

The discontinuous control is

$$u_D = u_T, \tag{3.41}$$

where the twisting controllers $u_{T,i}$, $i = 1, 2, 3$ for attitude angles, are modified from Eq. (2.19) as:

$$u_{T,i} = \begin{cases} -\mu_i\alpha_i \text{sign}(\sigma_i) & \text{if } \sigma_i\dot{\sigma}_i \leq 0 \\ -\alpha_i \text{sign}(\sigma_i) & \text{if } \sigma_i\dot{\sigma}_i > 0, \end{cases} \tag{3.42}$$

where $\mu_i < 1$ is a fixed positive number and $\alpha_i > 0$ is the control gain [102]. To improve the control transient and tracking performance, the gain α_i in (3.42) could be selected to satisfy the following condition for the one-stage accelerated twisting algorithm [54]:

$$\alpha_i = \max\{\alpha_{*,i}, \gamma_i |\sigma_i|^{\rho_i}\}, \tag{3.43}$$

where $\alpha_{*,i}$, γ_i and ρ_i are positive constants.

Given that fixed time stability is required over a large operational region of the UAV, and motivated by the simplicity of the one-stage accelerated twisting algorithm mentioned above, it is proposed to adjust the gain α_i in (3.42) adaptively as in [157, 75], to be constructed based on the following equation:

$$\dot{\alpha}_i = \begin{cases} \bar{\omega_i} \big|\sigma_i(\omega, t)\big| \operatorname{sign}(|\sigma_i(\omega, t)|^{\rho_i} - \epsilon_i) & \text{if } \alpha_i > \alpha_{m,i} \\ \eta_i & \text{if } \alpha_i \leq \alpha_{m,i}, \end{cases} \tag{3.44}$$

where $\bar{\omega_i}, \rho_i > 0$, ϵ_i and η_i are positive constants and $\alpha_{m,i}$ is an adaptation threshold, chosen to be greater than $\alpha_{*,i}$.

In trying to find a condition for the convergence of the proposed control and adaptation schemes, the below Lyapunov function candidate is selected:

$$V = \frac{1}{2}\sigma^T I \sigma + \sum_{i=1}^{3} \frac{1}{2\gamma_i}(\alpha_i - \alpha_{M,i})^2, \tag{3.45}$$

where I is the inertia matrix, γ_i is a positive constant, and $\alpha_{M,i}$ is the maximum value of the adaptive gain, i.e. $0 < \alpha_{m,i} < \alpha < \alpha_{M,i}$. According to the assumption A.1 in section (3.2), $\dot{I} = 0$. Thus, by taking the time derivative of V and substituting $\dot{\sigma}$ from (3.39), which yields:

$$\begin{aligned} \dot{V} =& \sigma^T I \dot{\sigma} + \sum_{i=1}^{3} \frac{1}{\gamma_i}(\alpha_i - \alpha_{M,i})\dot{\alpha}_i \\ =& \sigma^T \left(-I\ddot{X}_{1d} + I\Lambda\dot{e} - S(\omega)I\omega + u + d \right) + \sum_{i=1}^{3} \frac{1}{\gamma_i}(\alpha_i - \alpha_{M,i})\dot{\alpha}_i. \end{aligned} \tag{3.46}$$

Equation (3.24) can be rewritten as,

$$\begin{aligned}\dot{V} &= \sigma^T(d+u_T) + \sum_{i=1}^{3}\frac{1}{\gamma_i}(\alpha_i-\alpha_{M,i})\dot{\alpha}_i \\ &= \sum_{i=1}^{3}\left[\sigma_i(d_i+u_{T,i}) + \frac{1}{\gamma_i}(\alpha_i-\alpha_{M,i})\dot{\alpha}_i\right]. \end{aligned} \tag{3.47}$$

For the case $\sigma_i\dot{\sigma}_i \leq 0$, from the twisting control law, one has

$$\begin{aligned}\dot{V} &= \sum_{i=1}^{3}\sigma_i\left[d_i - \alpha_i\mu_i\text{sign}(\sigma_i)\right] + \sum_{i=1}^{3}\frac{1}{\gamma_i}(\alpha_i-\alpha_{M,i})\bar{\omega}_i|\sigma_i|\,\text{sign}(|\sigma_i|^{\rho_i}-\epsilon_i) \\ &= \sum_{i=1}^{3}|\sigma_i|\,\mu_i\Big[\frac{d_i\text{sign}(\sigma_i)}{\mu_i} - \alpha_i + \frac{\bar{\omega}_i}{\gamma_i\mu_i}(\alpha_i-\alpha_{M,i})\text{sign}(|\sigma_i|^{\rho_i}-\epsilon_i)\Big]. \end{aligned} \tag{3.48}$$

By assuming that the disturbance d is bounded, i.e., $|d_i| \leq \Xi_{M,i}$, and with sufficiently small ϵ_i such that $|\sigma_i|^{\rho_i} > \epsilon_i$ [157], $\dot{V} \leq 0$ if

$$\left|\frac{d_i\text{sign}(\sigma_i)}{\mu_i}\right| \leq \alpha_i \text{ or } \alpha_i \geq \frac{\Xi_{M,i}}{\mu_i}. \tag{3.49}$$

Noting that only the case $\alpha_i > \alpha_{m,i}$ is considered here as otherwise the last term in the right hand side of (3.48) becomes $\frac{1}{\gamma_i}(\alpha_i-\alpha_{M,i})\eta_i < 0$. For the case $\sigma_i\dot{\sigma}_i > 0$, from (3.42) the same result will be achieved as above if considering $\mu_i = 1$.

3.6 Results

3.6.1 Simulation setup

Extensive simulation and comparisons have been conducted to evaluate the performance of the proposed controller. The quadcopter model used in this study is the 3DR Solo drone. The model of the test quadcopter is shown in Figure 4.5 in which L_x, d_x, r_x and h_x are measured distances used to compute the system parameters, as listed in Table 3.1. The UAV was deployed to perform the tasks of infrastructure inspection, as shown in Figure 3.2.

Numerical simulations have been carried out in three different situations, i.e., nominal conditions, under a presence of disturbances and parametric variations.

Figure 3.2 : Insfrastructure inspection

Table 3.1 : Parameters of the quadcopter model

Parameter	Value	Unit
m	1.50	kg
l	0.205	m
g	9.81	m/s^2
I_{xx}	$8.85 \cdot 10^{-3}$	$kg.m^2$
I_{yy}	$15.5 \cdot 10^{-3}$	$kg.m^2$
I_{zz}	$23.09 \cdot 10^{-3}$	$kg.m^2$

The initial states of the quadrotor are assumed to be at its steady position, where all control angles and angular velocities are zero. The desired angles are adjusted in the simulation as $\phi = -10^0$, $\theta = 10^0$ and $\psi = 45^0$. In the disturbance scenario,

a torque of $0.5N.m$ is individually added in each axis of the drone. Particularly, to demonstrate the performance of the controllers in dynamic variation conditions, the simulation parameters are varied to counteract the modelling errors. In this case, the Solo's most capable payload of 0.8 kg, is added to the model and the inertial matrix is augmented with some deliberately fixed uncertainties as listed in Table 3.2.

Table 3.2 : Uncertainties added to the inertia matrix

ΔI	x	y	z
x	0.4825	0.0044	-0.0077
y	0.0044	0.2437	0.0115
z	-0.0077	0.0115	0.2437

The next subsections will present the results of extensive simulations and validations, which has been carried out to evaluate the performance of the proposed control algorithms.

3.6.2 AQCSM simulation and validation

Design parameters used for the controllers are given in Table 3.3.

Table 3.3 : Control design parameters

Variable	Value	Variable	Value
λ_1	4.68	λ_2	4.68
λ_3	3.84	$\epsilon_{1,2,3}$	0.7
α_0	1.24	$\bar{\omega}_{1,2,3}$	200
$\alpha_{m,1}$	0.01	$\alpha_{m,2}$	0.02
$\alpha_{m,3}$	0.03	$\eta_{1,2,3}$	0.01

Control performance in nominal conditions

The simulation results are shown in Figures 3.3 and 3.4, where the time scale in the latter is zoomed in to observe the abrupt change in the control torque and coupling effect. It can be seen that all controllers smoothly drive the angles to the desired values with relatively small overshoot within two seconds. According to (3.11), there are strong coupling relations between the control states. As a result, it can be seen that the AQCSM controller can handle this problem to control the attitude to reach the reference values and then track them without being perturbed.

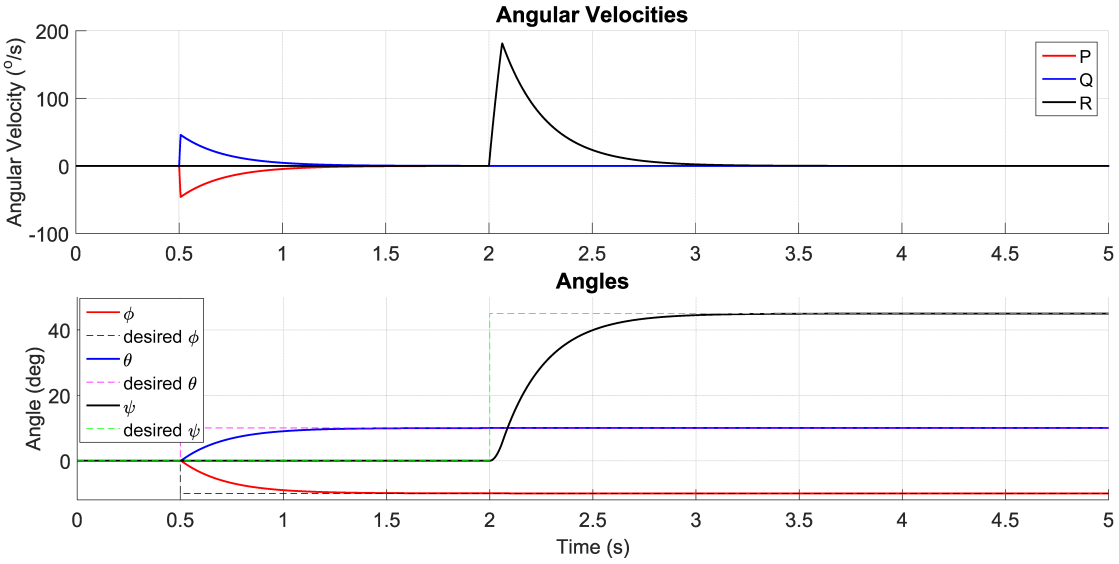


Figure 3.3 : Responses of the quadcopter in nominal conditions (P, Q and R- roll, pitch and yaw angular velocities)

Responses to disturbances

In this simulation, a torque disturbance with the amplitude of $0.5N.m$ is added to all three axes of the quadcopter. The reference values are chosen to be the same as in the previous simulation. The response is shown in Figure 3.5. As can be seen from the figures, the AQCSM controller can cope with disturbances to reach the references and maintain the drone stability.

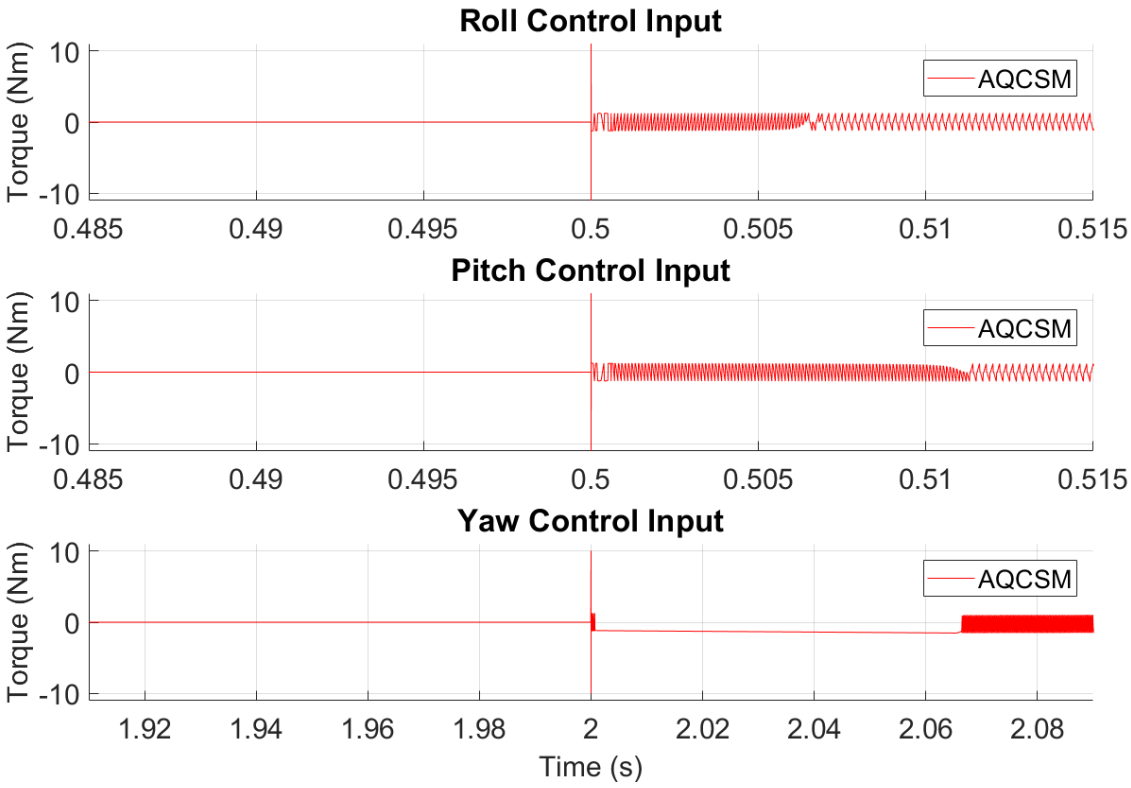


Figure 3.4 : Control torques

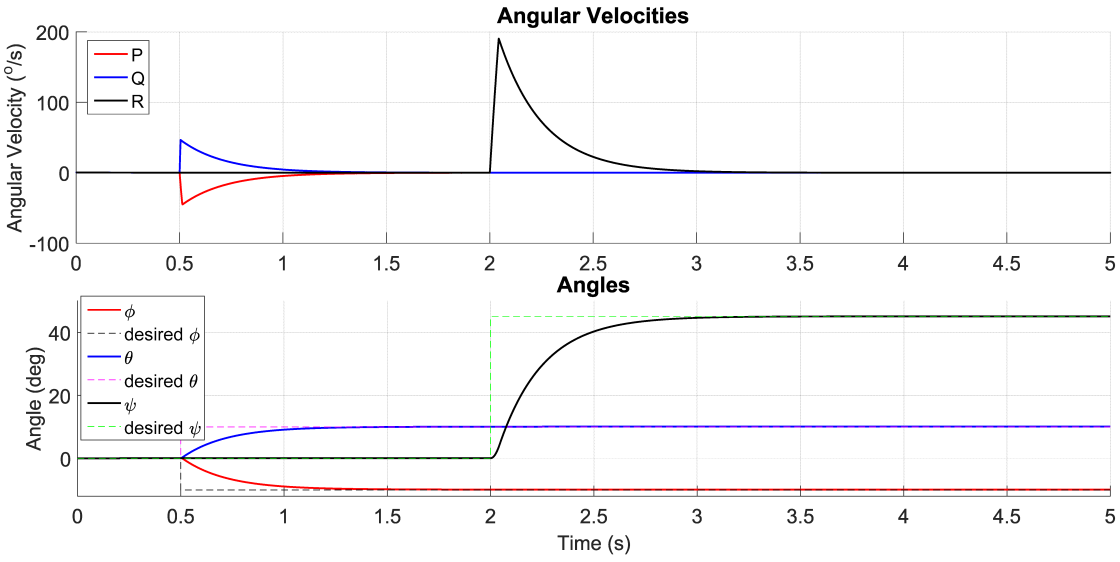


Figure 3.5 : Angular velocity and angle responses in the presence of disturbances

Responses to parametric variations

Figure 3.6 shows the results in comparison with the nominal conditions. The almost identical settling time and overshoot between responses corresponding to

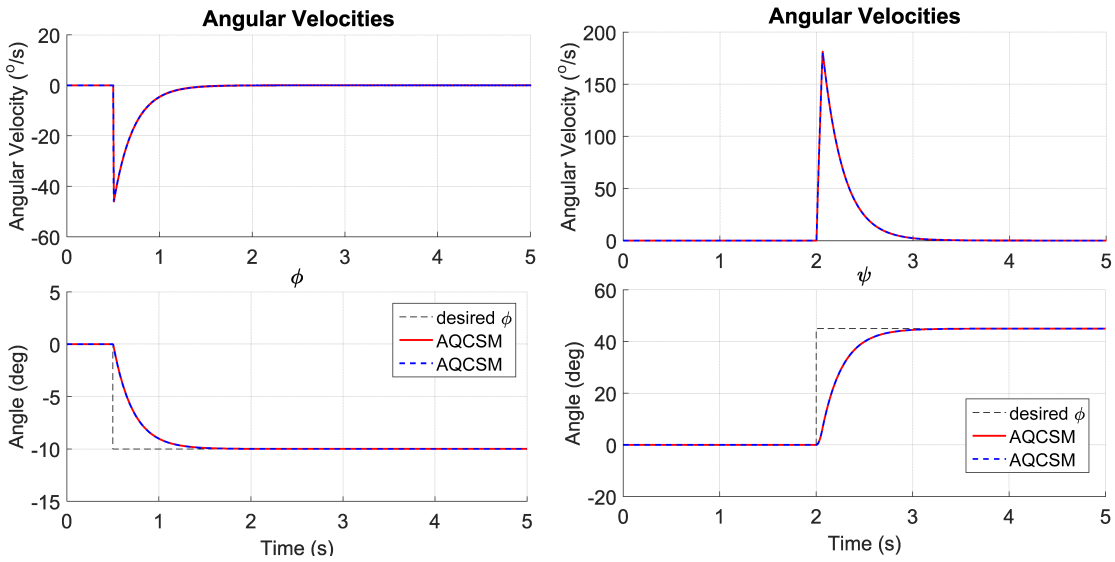


Figure 3.6 : Comparison of angle and angular velocity responses in the presence of parametric variations (red) and the nominal conditions (blue dash)

those scenarios indicate the robustness of the proposed AQCSM controller. The adaptive gain $\alpha_1(t)$ response versus time is shown in Figure 3.7. The higher gain magnitudes which are observed in the two bottom sub-figures imply more energy is required to stabilise the system in dealing with disturbances and uncertainties. This result also verifies the feasibility of the control scheme.

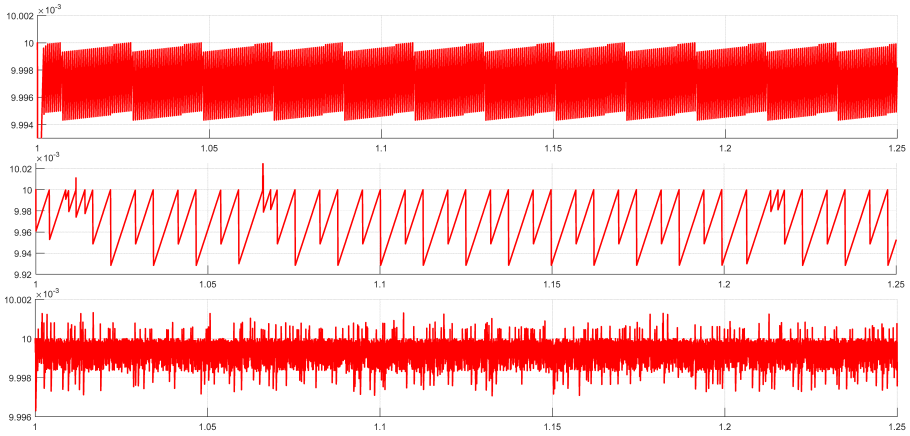


Figure 3.7 : The adaptation of gain $\alpha_1(t)$ in various scenarios

3.6.3 AdTSM simulation and validation

The control parameters used for this study are given in Table 3.4.

Table 3.4 : Control design parameters

Variable	Value	Variable	Value
λ_1, λ_2	4.68	$\bar{\omega}_1, \bar{\omega}_2, \bar{\omega}_3$	200
λ_3	3.84	$\alpha_{m,1}, \alpha_{m,2}, \alpha_{m,3}$	2.001
ρ_1, ρ_2, ρ_3	3.0	$\alpha_{M,1}, \alpha_{M,2}, \alpha_{M,3}$	2.12
μ_1, μ_2, μ_3	1/4	$\Xi_{M,1}, \Xi_{M,2}, \Xi_{M,3}$	0.5
$\epsilon_1, \epsilon_2, \epsilon_3$	0.6	η_1, η_2, η_3	0.01

Control performance in nominal conditions

Performance of the controller is first evaluated in nominal conditions. The system responses and controller outputs are shown in Figures 3.8 and 3.9, respectively, where the latter shows the zoomed-in time scale to observe the abrupt change in references and coupling effects. It can be seen that the proposed controller smoothly drives the angles to the reference values within one second and with a small overshoot despite strong coupling relations among control variables as described in (3.13-3.15). However, there still exists minor chattering from the numerical integration of the control system, as depicted in Figure 3.9. This can be interpreted as the trade-off to obtain a better control performance, which requires a larger gain α_i in (3.42). Nevertheless, this phenomenon can be mitigated by adaptively adjusting α_i to its threshold value in the steady state.

Responses to disturbances

In this simulation, the robustness of the controller is evaluated by adding the disturbances with the mean value of 0.5 Nm to the torques in all three body axes of the

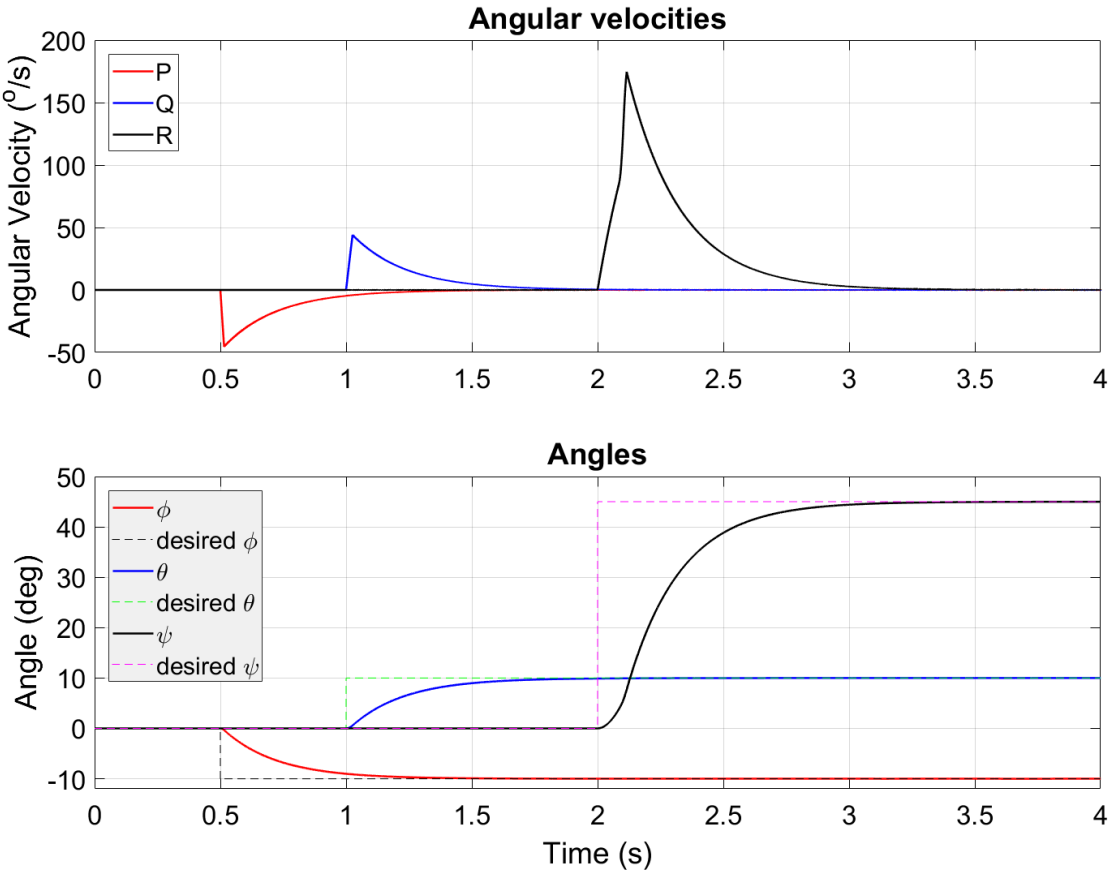


Figure 3.8 : Responses of the quadcopter (P, Q and R- roll, pitch and yaw angular velocities) in nominal conditions

quadcopter, corresponding to the angular acceleration disturbances in (3.13-3.15). Reference values were selected to be the same as in the previous simulation. Results are shown in Figure 3.10. It can be seen that the proposed controller effectively rejects external disturbances to drive the quadcopter to reach the expected attitude within a similar time period as in nominal conditions.

Responses to parametric variations

In this simulation, the quadrotor is subject to several sources of uncertainties including variations in loads and moments of inertia. Specifically, a load of 0.8 kg is added to the model together with the uncertainties in moments of inertia, listed

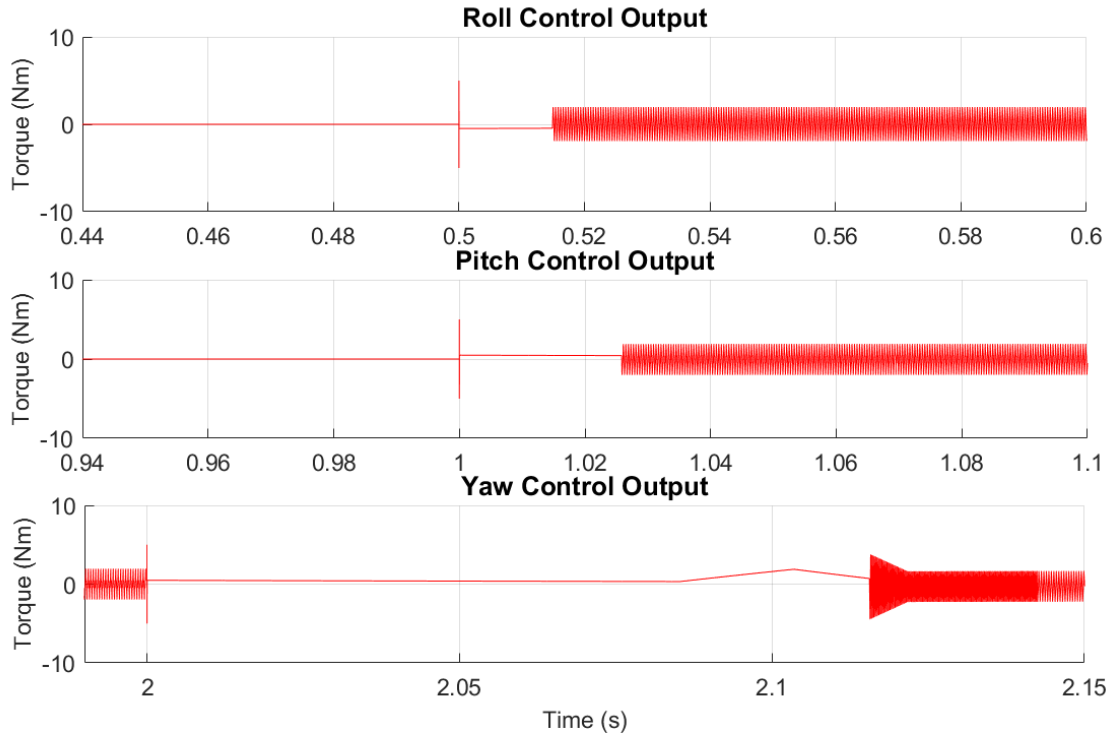


Figure 3.9 : Control torques

in table 3.2.

Figure 3.11 shows the results in comparison with the nominal conditions. The settling time and overshoot of the responses are almost identical, indicating the high robustness of the proposed controller. The variation of the adaptive gain observed in simulation is in the interval $2.001 \leq \alpha_i(t) \leq 2.12, i = 1, 2, 3$. Higher gain magnitudes imply more energy is required to stabilise the system to cope with the increase in disturbances and uncertainties owing to the effectiveness of the adaptation.

Comparison and validation with real-time data

For evaluation of the proposed control approach, simulation results were compared with real-time data obtained by using the built-in PID controller of the 3DR Solo drone to perform the mentioned attitude control. Figure 3.12 shows the flying path and data recorded, omitting position information, during the experiment. To

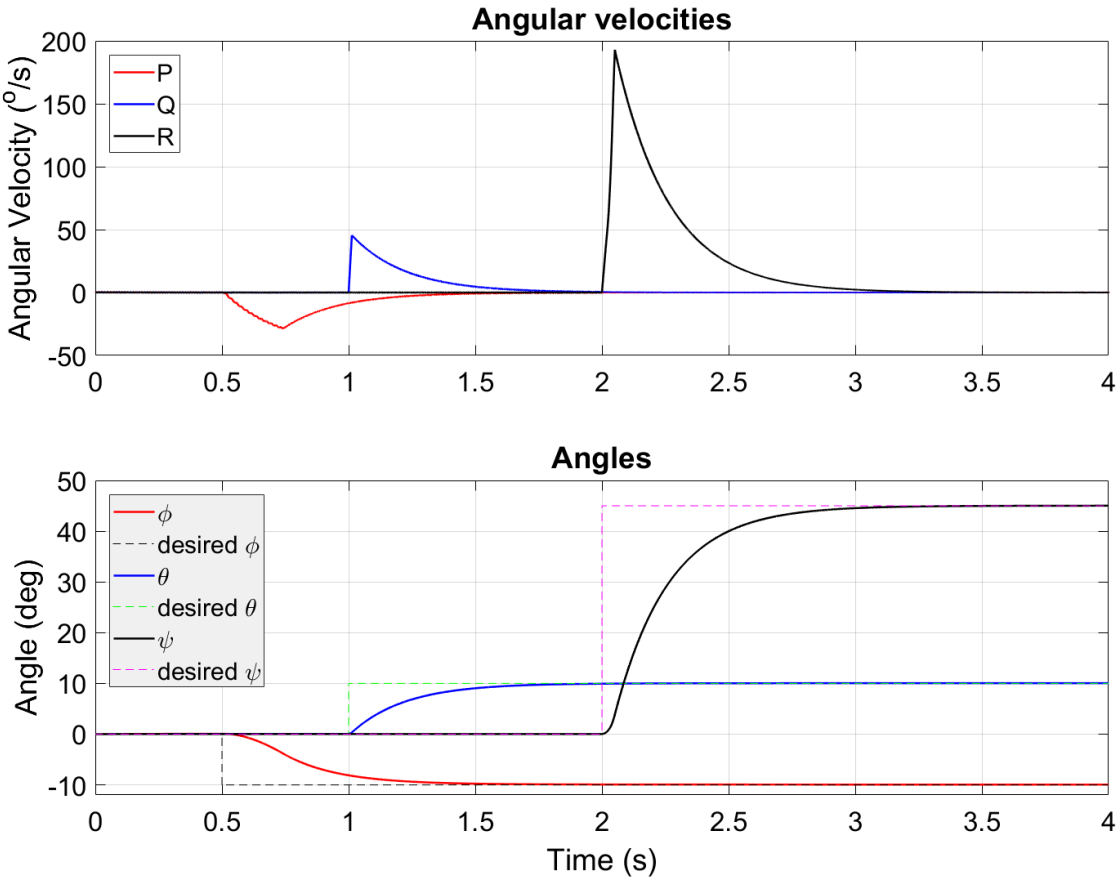


Figure 3.10 : Angular velocity and angle responses in the presence of disturbances.

compare the performance of the proposed controllers with other control techniques, the output responses were compared with those obtained by using the conventional SMC and the accelerated twisting sliding mode (ATSM) [54]. The comparison was conducted under the three different scenarios, as illustrated in Figures 3.14 - 3.16. Also, for the validation purpose, the simulation results were compared with real-time data obtained when the drone is performing similar attitude control tasks. These results are shown in Figure 3.13 for the three control torques, where the proposed controller results in better tracking performance with reduced chattering. Indeed, Figures 3.14 - 3.17 show the time responses of Euler angles and angular velocities wherein the yaw tracking errors in the steady state are zoomed in. It can be seen that all controllers show similar performance as in nominal conditions. In the pres-

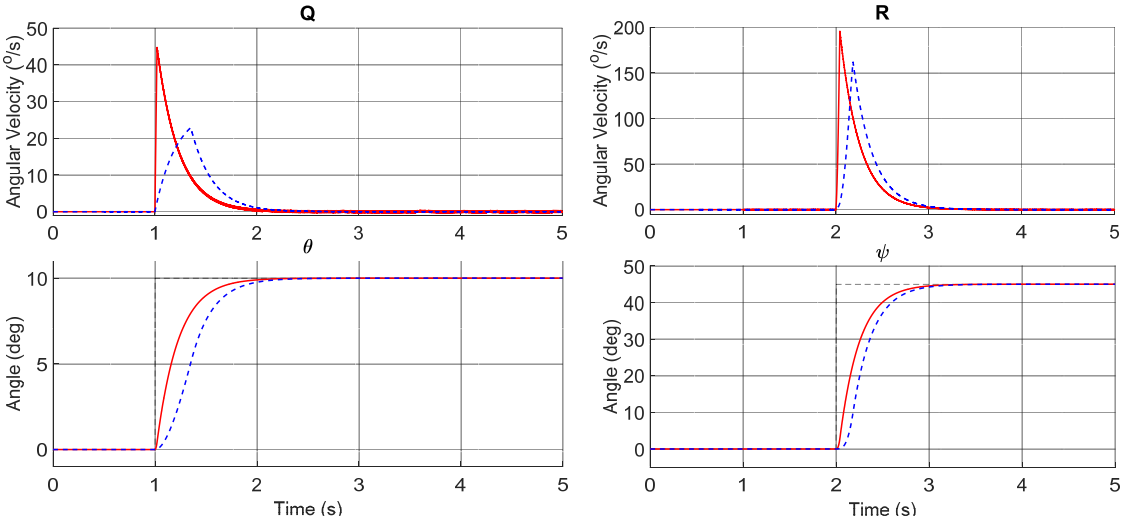


Figure 3.11 : Angle and angular velocity responses in the presence of parametric variations

Figure 3.12 : Experimental data acquisition

ence of disturbances, the proposed controllers can, however, exhibit the smallest tracking errors, indicating a better capability of dealing with disturbances. In the case of parametric variations, the proposed controllers can also provide the fastest

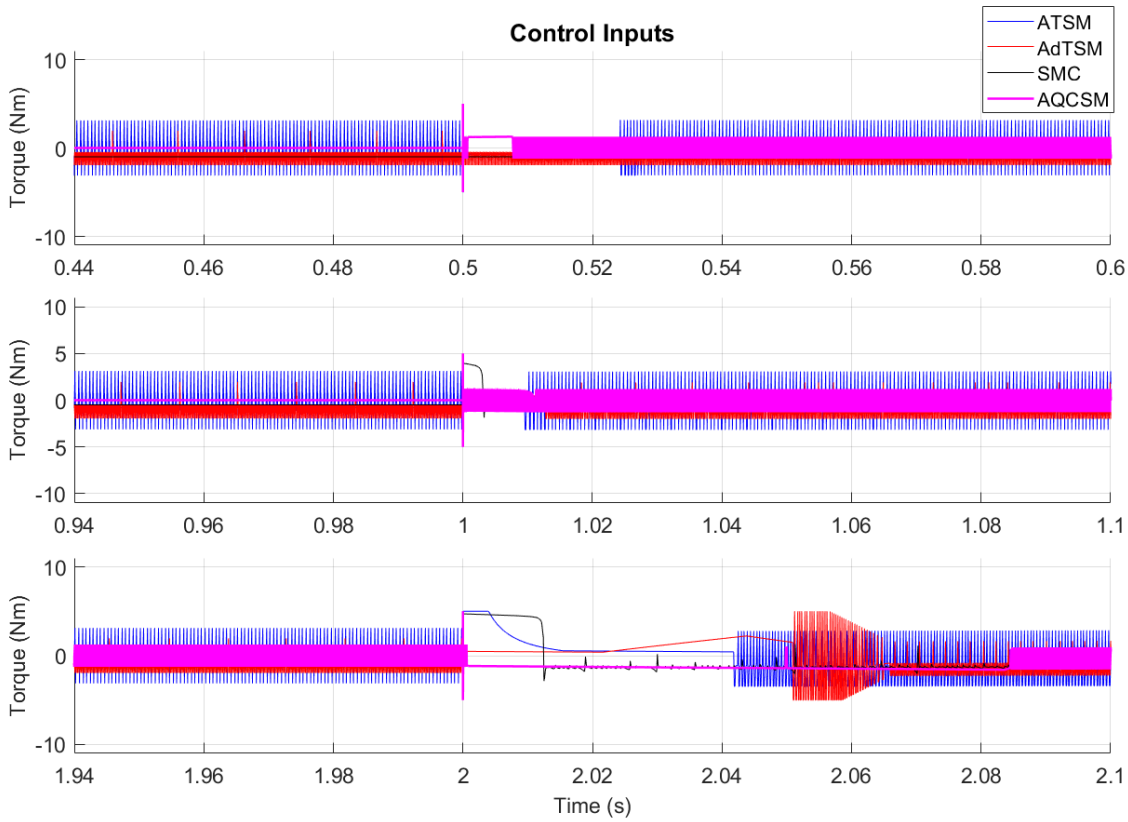


Figure 3.13 : Time responses of three control inputs

convergence thanks to the proposed adaptive scheme.

3.7 Summary

In this chapter, designs and developments of the low-level control system for a quadcopter UAV are presented. For each quadcopter, robust onboard control schemes are developed based on the adaptive second-order sliding mode control. The ultimate objective of the development is to robustly control the quadcopter attitude in the outdoor environment, which is subjected to external disturbances as well as system uncertainties for the surface inspection application. The control enhancement contains the selection of the sliding manifold and the development of the quasi-continuous and twisting second-order sliding mode controllers with adaptive schemes. A global Lyapunov candidature function is selected for stability analy-

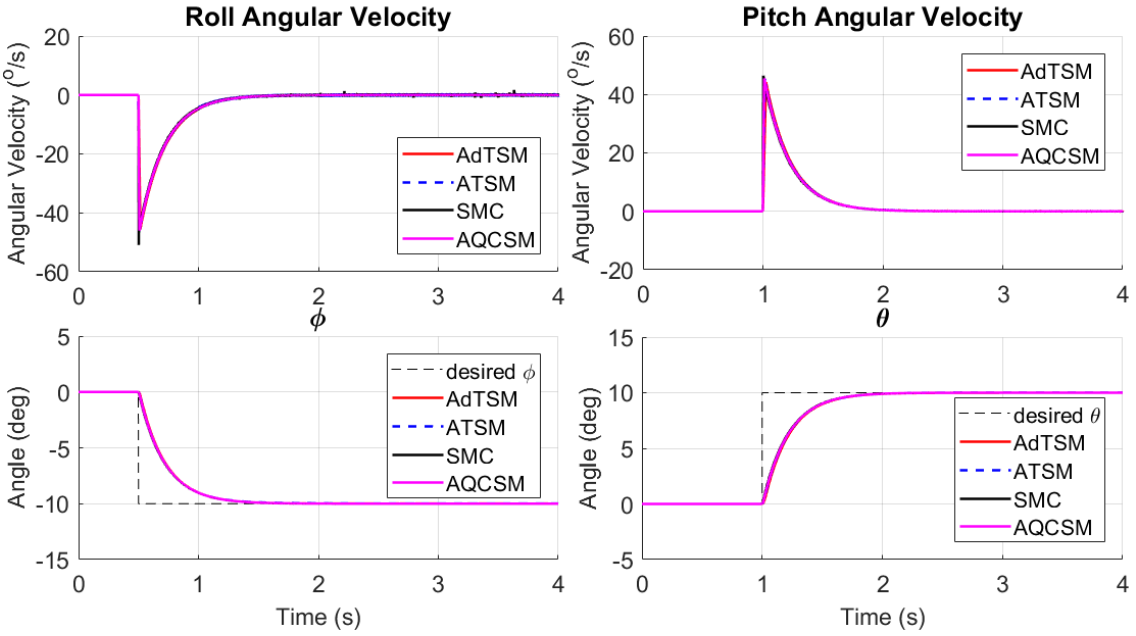


Figure 3.14 : The roll and pitch angle and angular velocity responses of controllers in nominal condition

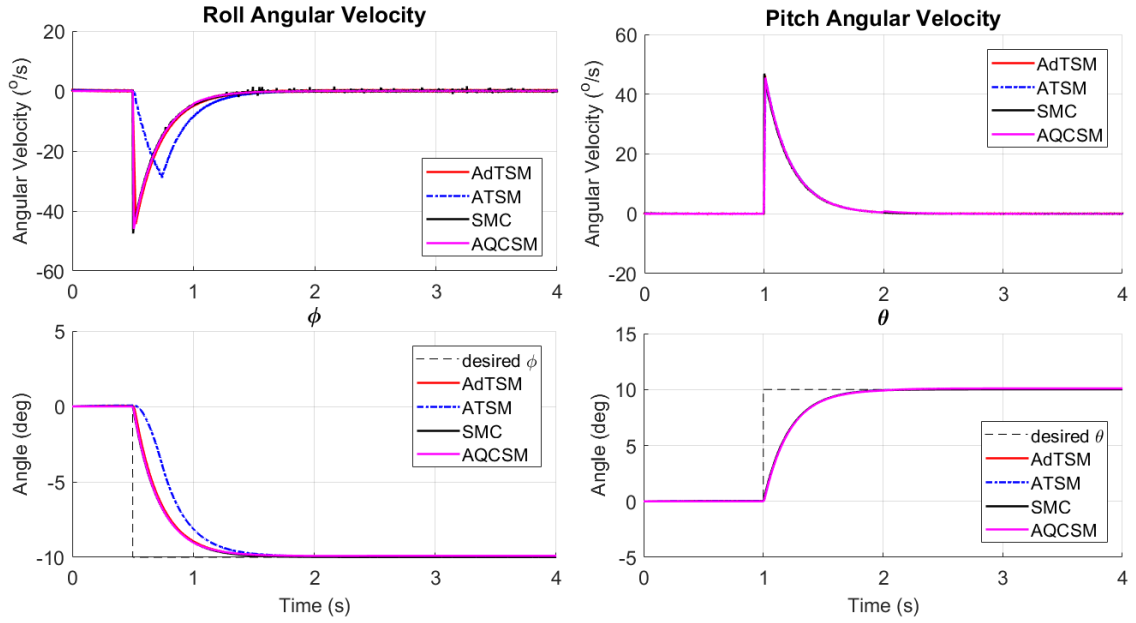


Figure 3.15 : The roll and pitch angle and angular velocity responses of controllers in occurrence of disturbances

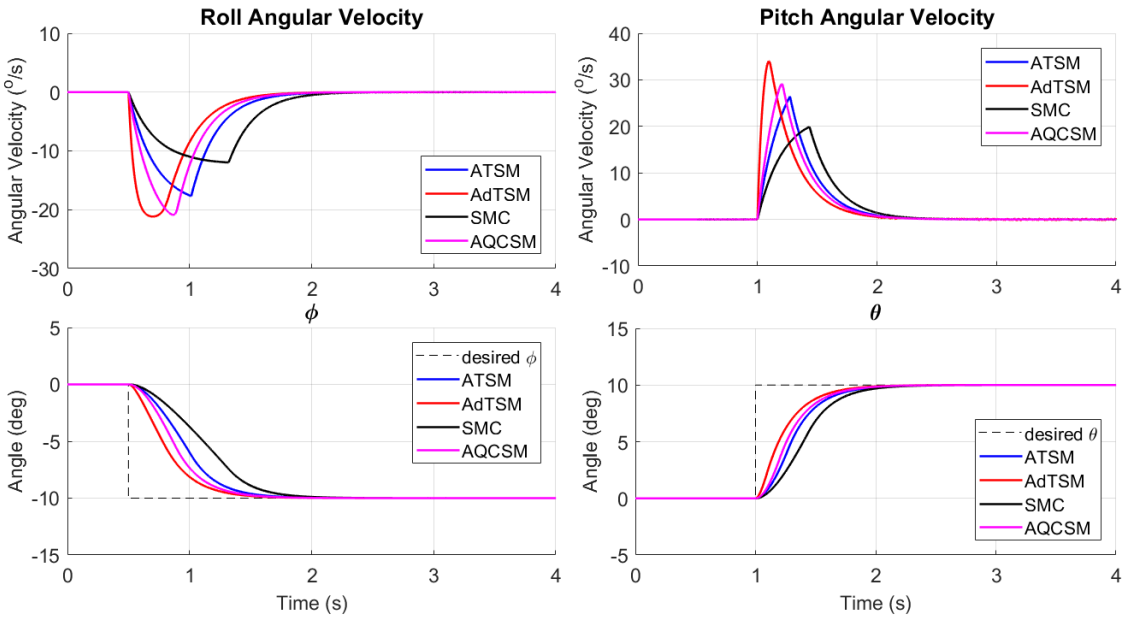


Figure 3.16 : The roll and pitch angle and angular velocity responses of controllers in occurrence of parametric variations

sis of the overall system to illustrate the convergence of both the sliding dynamics and adaptation schemes. The performance and advancement of the proposed approaches are evaluated by some extensive simulation and comparisons with real-time experiments. Results obtained demonstrate the feasibility of the proposed systems.

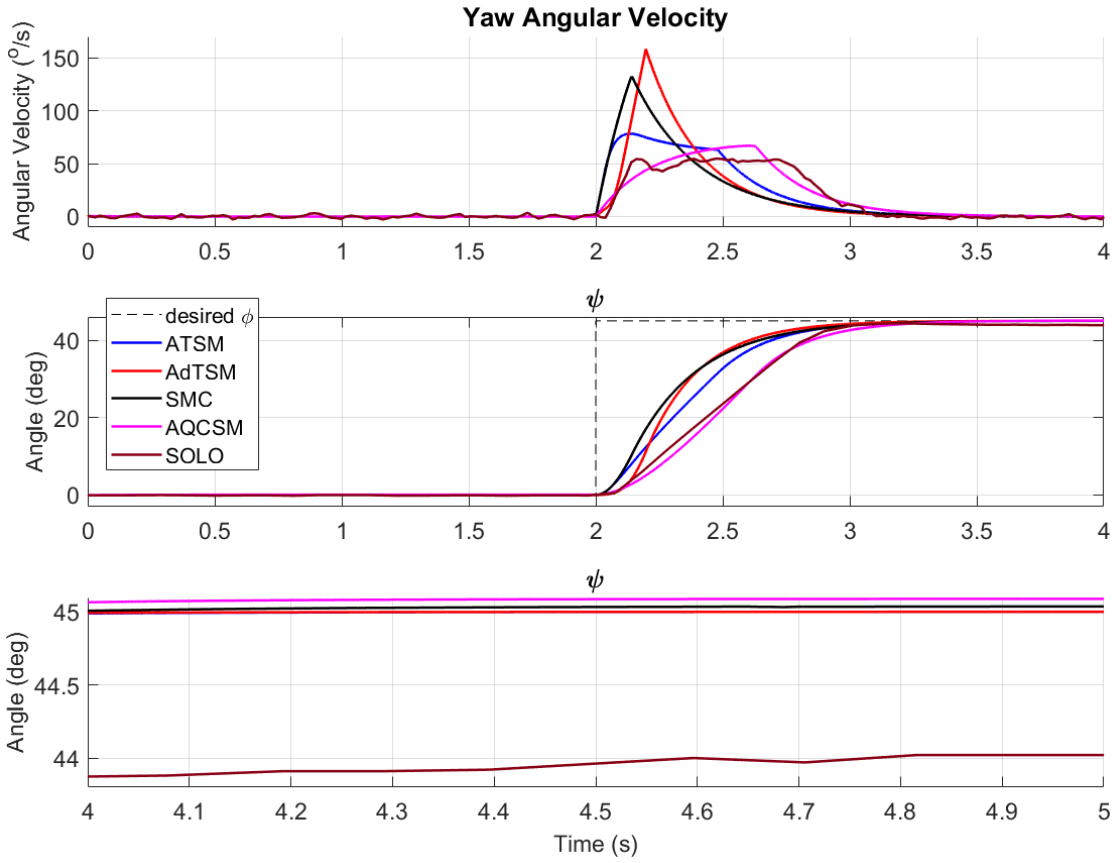


Figure 3.17 : Zoom-in tracking errors of controllers at the steady state

Chapter 4

Multi-UAV system architecture

A real-time control system for surface inspection using multiple unmanned aerial vehicles is presented in this chapter. The UAVs are coordinated in a specific formation to collect data of the inspecting objects. The formation is established via using the angle-encoded particle swarm optimisation to generate an inspecting path and redistribute it to each UAV where communication links are embedded with an IoT board for network and data processing capabilities. Other hardware extensions for specific 3DR Solo drones allows them to fly safe, to maintain the formation topology, and to improve their mission efficiency.

4.1 Introduction

This chapter provides materials required for the design and development of a system architecture of multi-UAVs in real-time operating. It starts with an overview of system architecture. Then, it provides brief theoretical backgrounds for path planning and multi-UAV formation problems. The chapter ends with descriptions of a specific UAV and its hardware development.

The first part begins with descriptions of the assigned mission assessment that is a basis to select a relevant configuration and control scheme for the multi-UAV system. Then, it is broken down into functional explanations of the subsystems.

The theoretical basis for analysis and evaluation of the systems is provided in the second part of the chapter. Since the low-level control designs have been provided in Chapter 3, backgrounds of path planning and multi-UAV formation problems are

covered in this part.

The chapter ends with details of the experimental quadcopters that will be used to verify the desired system developments further and also to analyse the performance of the whole system in real time experiments. The last part concludes by detailing of hardware development necessarily for UAVs and the formation to fulfil their assigned tasks.

4.2 System overview

Figure 4.1 illustrates a structure of a multi-UAV formation. There are three layers: the task assessment layer, the high-level control layer and the low-level control layer. They have a tight relationship with each other in order to create the entire system architecture of multi-UAVs conducting infrastructure and inspection tasks.

The top layer, named Task Assessment, is necessary for decision making, defect detection and sensor selection. It starts with the choice of the target monitored structure or environment. From this, the inspected surface(s) will be determined. Based on the selected defect detection technique, relevant sensors will be chosen and attached on UAVs. Then, parameters of the selected area of interest will be measured, specifically, its range of length, width, and height. These parameters are necessary to create the 3-D terrain topology for the path planning problem. Based on the 3-D terrain, the inspected starting point and target point are designated. The task assessment for visual-based infrastructure inspection will be discussed in Section 4.3.1.

The high-level control block contains two main modules, i.e., the Formation module, and Path Planning module.

Based on a general mission requirement and the working space conditions, the formation module will determine positions and mission allocations for each UAV.

Hierarchy of multiple UAVs system

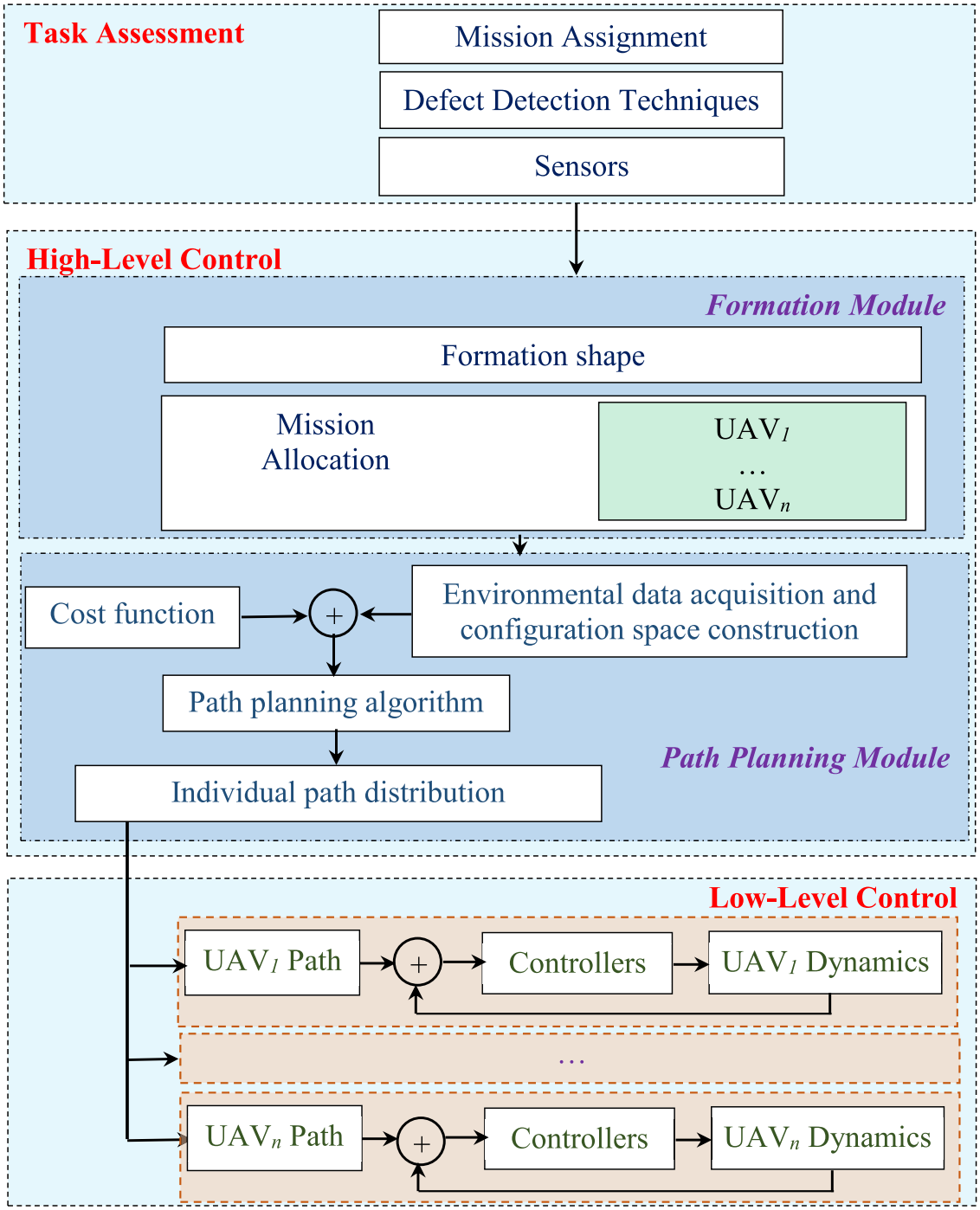


Figure 4.1 : System architecture

As a result, the formation shape is decided in order to cover the inspected structure and task criteria. The location for each UAV is also determined on the basis of its designated working areas and equipped sensors. The formation shape is then decided, followed by a suitable shape maintenance algorithm. In complex environments, a unique formation topology may not guarantee the safety as well as the mission performance. Therefore, reconfigurable topologies are required to allow the formation to work adaptively in some dynamic working spaces. Such an algorithm will be covered in Chapter 6.

By taking results from the module on the top of Figure 4.1, planned paths will be generated in the second module. Here, by adding details of the environmental data acquisition process, the second module will plan a safe, feasible and flyable route for the entire formation. The reference trajectory will be generated and distributed for the corresponding UAV. The planning algorithm will be investigated in Chapter 5.

The lowest layer is known as the task execution layer, that belongs to the low-level control for the UAVs. This layer is the vital one in real-time flight to preserve the formation configuration while conducting inspection tasks efficiently. Therefore, the designed controllers in Chapter 3 are proposed for tracking the generated path whilst mitigating control errors and rejecting disturbances and parametric uncertainties.

In the fly time, by using the attached sensors and communication networks, the safety of the whole formation is preserved. For visual-based inspection, the cameras on each UAV will continuously take pictures of the structure's surface. These pictures, which also carry the time and GPS's location of the place that they were taken, are then processed to detect any defects on the surface of the monitored object.

4.3 Theoretical background

4.3.1 Task assessment

Given the surface to be inspected, the task assessment layer computes a set of camera configurations that together cover the whole surfaces of interest given constraints on image quality. Each configuration c_i corresponds to a position and an orientation of the camera. The constraints are for image sticking and defect detection with the following requirements:

(i) photos are taken when the camera is perpendicular to the inspected surface;

(ii) their resolution is sufficiently high to distinguish the smallest feature, s_f; and

(iii) the photos are overlapped to a percentage, o_p, specified by the sticking algorithm.

The first requirement confines the camera orientation to the normal of the inspected surface. The resolution requirement suggests the computation of the field of view of the camera as:

$$a_{fov} = \frac{1}{2} r_c s_f, \tag{4.1}$$

where r_c is the camera resolution. Thus, the distance $d_{f,i}$ from the ith UAV to the inspecting surface can be found as:

$$d_{f,i} = \frac{a_{fov} f}{s_s}, \tag{4.2}$$

where f and s_s are respectively the focal length and sensor size of the camera. Let G be a finite set of geometric primitives g_i comprising the surfaces to be covered. Each geometric primitive g_i corresponds to a surface patch covered by a camera shot. Having taken the overlapping percentage into account, the geometric primitive g_i is computed as:

$$g_i = (1 - o_p) a_{fov}. \tag{4.3}$$

Equation 4.2 and 4.3 are sufficient to determine configurations c_i to cover the set of primitives G [156] which is then fed to the high-level control layer for path planning.

4.3.2 UAV path planning problem

This section introduces a fundamental path planning process to produce a real-time trajectory from a start location to a target location.

Given that a set of UAV's location (x, y, z) and attitude (θ, ψ) are the horizontal and vertical angles, denote the start and target waypoints as P_s and P_f. The path planning algorithms are basically applied to generate one or more flight paths $T^*(q)$ to connect P_s and P_f. Mathematically, path planning can be expressed as:

$$P_s \xrightarrow{T^*(q)} P_f, \tag{4.4}$$

where $T^*(q)$ is the resulting path, and q is defined as a path parameter. According to the desire mission, the path parameter can be represented by different attributes, e.g., waypoints, orientation angles, forces, velocities, or distances.

Path planning problem for a single UAV from the start pose $P(x_s, y_s, z_s, \theta_s, \psi_s)$ to the finish pose $P(x_f, y_f, z_f, \theta_f, \psi_f)$ is then defined as:

$$P(x_s, y_s, z_s, \theta_s, \psi_s) \xrightarrow{T^*(q)} P(x_f, y_f, z_f, \theta_f, \psi_f). \tag{4.5}$$

Path planning problem for a group of n UAVs, with the corresponding start pose P_{si} and finish pose P_{fi}, where both are linked by a path $T_i^*(q)$, gives:

$$P_{si}(x_{si}, y_{si}, z_{si}, \theta_{si}, \psi_{si}) \xrightarrow{T_i^*(q)} P_{fi}(x_{fi}, y_{fi}, z_{fi}, \theta_{fi}, \psi_{fi}), \quad i = 1, \cdots, n. \tag{4.6}$$

Path planning problem for a rigid body formation of UAVs, represented by a virtual point P_F, usually the formation's centre of mass, from the start pose P_{Fs} to the finish pose P_{Ff} can be represented as:

$$P_{Fs}(x_{Fs}, y_{Fs}, z_{Fs}, \theta_{Fs}, \psi_{Fs}) \xrightarrow{T_F^*(q)} P_{Ff}(x_{Ff}, y_{Ff}, z_{Ff}, \theta_{Ff}, \psi_{Ff}). \tag{4.7}$$

The above path planning problem is a basic problem to generate a connected route between two waypoints. Generally, a real-time trajectory also requires avoiding obstacles or collisions, but also optimising a given functional under a set of constraints. Depending on a specific mission, different requirements have been introduced, i.e., minimising path length, performances optimisation, collision avoidance, real-time planning or risk minimisation. In particular, the path length and collision avoidance are the major issues in any path planning algorithm.

In some specific tasks, there are various constraints involved in path planning to complete the tasks more efficiently. The adopted path must be flyable and safe to guide the UAV through complex environments and simultaneously achieve highly effective inspection tasks for a given area of interest. The UAVs should be capable of following any resulting path. In multi-UAV path planning, the algorithms must also allow for the deployment of several UAVs in a coordinated manner. In some cases, the path planning algorithms must be computationally efficient with less memory occupation to operate in real time.

If the paths contained a set of waypoints, but their connection is presented by straight-line segments, they may not be flyable because the UAV cannot turn instantaneously through each waypoint. For a flyable path, each segment must have a common tangent to produce a continuous path so that some segments must be curved and smooth.

Generally, path planning for UAVs has the following attributes [216, 215]:

- Flyable trajectory: Flyable paths meet kinematic or motion constraints and dictate the manoeuvrability of the UAVs.
- Physical feasibility: This refers to the physical limitations in the use of UAVs, which include the maximum path distance and the minimum path length.

- Safety: is achieved by avoiding obstacles, either fixed or moving, that intersect the path, both internal and external.

- Performance of the mission: This refers to whether a path can satisfy the requirements of a specified mission. To complete the mission, various requirements must be met when designing a path. These requirements usually include the maximal turning angle, the maximum climbing/diving angle, and the minimal flying height. Other constraints, such as maintaining communication range, minimising path length, maintaining working altitude, can be added into the system.

- Real-time implementation: This refers to the efficiency of path planning. The flight environments of UAVs are usually constantly changing. Therefore, the path-planning algorithm must be computationally efficient. Replanning ability is critical for adapting to unforeseen threats.

Hence the path planning problem for a multi-UAV formation in (4.7) can be extended by taking into account a number of constraints as the following form:

$$P_{Fs}(x_{Fs}, y_{Fs}, z_{Fs}, \theta_{Fs}, \psi_{Fs}) \xrightarrow{\coprod T_F^*(q)} P_{Ff}(x_{Ff}, y_{Ff}, z_{Ff}, \theta_{Ff}, \psi_{Ff}), \tag{4.8}$$

where $\coprod$ represents the set of constraints.

Offline path planning: Offline path planning is a traditional method to create an appropriate path for a UAV in a predetermined and static environment. All data in this environment has been defined in advance including some specific constraints. The set of constraints, which may affect the planning results, includes obstacles and/or threats, kinematics of vehicle, and optimised path or fuel consumption. Among them, minimising the path length together with collision avoidance are the most preferable.

Online path planning: Online path planning is a fundamental concern for

the implementation of UAV flight in real-time. Solving this problem will increase the effectiveness, accuracy, and adaptability to any real and practical environments. The path planning method is classified as a dynamic multi-objective optimisation problem. Online path planning can be divided into global and local planning problems. While the former one generates the path from the start to the target points for UAVs in the entire working space, the latter one uses sensory information as its input. Whenever a potential collision with an obstacle has been detected, the local path algorithm will be triggered to replan the path online. A key technology of online path planning for UAVs is the capability of quick response to the changes in dynamic environments so that accelerated path planning algorithms are required.

4.3.3 Formation concepts

This subsection presents a number of concepts related to the UAV formation. It begins with some definitions that would be used in this project before describing a virtual structure formation model.

Multi-UAV swarm definitions

Rigid Body [9]: A rigid body is composed of a system of point masses fixed by holonomic constraints such that: $r_i - r_j = d_{ij} = const$ where r_i, r_j are the positions of the particles.

To represent the position and orientation of rigid-body systems, two coordinate frames are used: the inertial frame, and the body frame. An earth-fixed North, East, Down (NED) coordinate frame $\{O\} = (x_O, y_O, z_O)$ is often assumed to be an inertial frame given the typical distance and travel time of the rigid-body.

Virtual Structure (VS) [104]. A virtual structure is a group of N UAVs, which maintain a (semi-) rigid geometric relationship to each other and to a frame of reference.

The particles can be considered as remaining immobile with respect to a specific reference frame that is also moving within space. If the geometric relations are not enforced by a physical system of constraints but instead by a human-made control system, then the structure is called a Virtual Structure. In a robotic system, the generation of a Virtual Structure is facilitated by the use of sensing, mobility and intelligent control.

Virtual Rigid Body (VRB): A VRB is a group of N UAVs and a global reference frame of the formation $\{O\}$, in which the local positions of the UAVs are specified by a set of potentially time-varying vectors $\{r_1(t), r_2(t), \ldots, r_N(t)\}$.

The VRB is extended from VS with a body frame rigidly attached among the group. Consider a group of N UAVs in a 3D environment with the two reference frames: the inertia frame $\{O\}$ and the body frame $\{B_i\}, i = 1, 2, \ldots, N$. Let $P_F(t) \in \mathbb{R}^3$ denote the position, and $R_{OF}(t) \in SO(3)$ the rotation matrix of the origin of the formation frame $\{F\}$ to the inertia frame $\{O\}$, where t is time. The relationship between $P_i(t)$ and $r_i(t)$ of the UAV i can be presented as:

$$P_i(t) = P_F(t) + R_{OF}(t) r_i(t) \quad \forall i \in \{1, 2, \ldots, N\}. \tag{4.9}$$

Formation [218]: A formation Π is a VRB with constant local positions $\{r_1(t)$, $r_2(t), \ldots, r_N(t)\}$ in the local reference frame $\{F\}$ for a group of N UAVs with a time duration $T_\Pi > 0$.

In a VRB formation, the position $P_i(t)$ of UAVi is constant, but its orientations will vary in the formation frame $\{F\}$ to preserve the relative positions in VRB throughout the path following period.

Transformation [218]: A transformation Φ is a VRB with time varying local positions $\{r_1(t), r_2(t), \ldots, r_N(t)\}$ in $\{F\}$ for a group of N UAVs with a time duration $T_\Phi > 0$.

During the transformation period, the position $P_i(t)$ of the UAVi is time-varying. Its initial position in the transformation Φ matches the origin formation shape, while its target position must be adjusted to meet the new formation shape. The transformation process, hence, operates as a bridge to reconfigure between two different formation topologies.

Virtual structure of multi-UAV formation

In the VS approach, positions of the UAVs are usually defined in a frame with respect to a reference point. A trajectory is generated for the reference point, and then a set of desired positions for each UAV can be constructed thanks to its relation to the VS as it in time.

This work uses a triangular UAV formation model with the inertial and formation frames represented in Figure 4.2. All measurements are referred to the inertial frame $\{O\}$ with axes x_O, y_O and z_O. Positions of UAV$_i$, $i = 1, 2, 3$, in the inertial frame are denoted as $P_i = \{x_i, y_i, z_i\}$. The formation frame, $\{x_F, y_F, z_F\}$, is defined such that the origin P_F is chosen to be coincident with the centroid of the triangle; the axis x_F is the direction from the centroid of the triangle to the UAV$_1$ position; the axis z_F is perpendicular to the plane containing three UAVs pointing downward; and the axis y_F is perpendicular to the plane formed by the x_F and z_F axes. This moving frame allows the determination of its relative orientation with respect to the fixed inertial frame.

The formation is a rigid body in which P_F, V_F, ψ_F and ω_F are the inertial position, velocity, heading angle, and angular velocity, respectively. Each UAV can be represented by its position, velocity, heading angle, and angular velocity P_i, V_i, ψ_i and ω_i with respect to the global inertial frame $\{O\}$, or by $P_{iF}, V_{iF}, \psi_{iF}$ and ω_{iF} with respect to the formation frame $\{F\}$.

Then the equations for the reference position and velocity profiles for each UAV

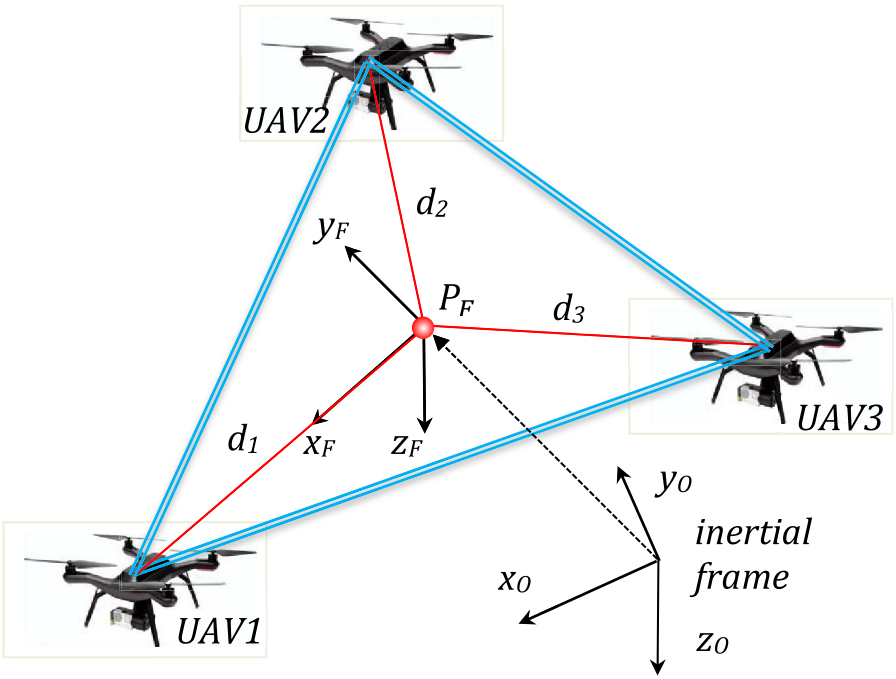


Figure 4.2 : Inertial and formation frames in UAV formation

in $\{O\}$ are:

$$\begin{aligned} P_i^d(t) &= P_F(t) + R_{OF}(t)P_{iF}^d(t) \\ V_i^d(t) &= V_F(t) + R_{OF}(t)V_{iF}^d(t) + \omega_F(t) \times (R_{OF}(t)P_{iF}^d(t)), \end{aligned} \tag{4.10}$$

where superscript ^{d} represents the reference values of the variables for the ith UAV.

Throughout this thesis, the centroid of the desired formation will be considered as the reference point for the whole group. Based on the desired positions P_i^d, and velocities V_i^d of the UAVi, it can be observed that the relative values for each UAV are generated with respect to the virtual centroid P_F of the formation. These values will be fed into the guidance system of the UAVs. The low-level controllers will operate to keep the UAVs on their desired trajectory in the formation shape. This process will be detailed in the next two chapters.

Formation initialisation

Given the desired path, the formation flight starts with an initialisation phase in which each UAV needs to reach its desired initial position from an arbitrary location without collision. While UAVs typically can take off automatically based on the built-in software, their movements to the desired initial positions require a number of checks as follows:

- Crossing check: verify whether or not the path of a UAV_i crosses the paths of other UAVs after taking into account marginal clearance ranges;

- Conflict check: verify whether or not a UAV_i enters the conflict region of other UAVs at the same time instant;

- Deadlock check: verify whether or not a UAV_i enters the assigned location of another UAV in the formation.

Those checks can be implemented by introducing a binary action variable c_i to each UAV. The value of $c_i = 0$ implies the UAV_i stops to avoid a collision whereas the value of $c_i = 1$ enables it to continue to fly. Denoting ΔT_i as the remaining flight time to the desired initial position of UAV_i, the initialisation process represented via the value of c_i that avoids the collision between every two UAVs is then presented as in Figure 4.3.

After the initialisation, the low-level control is applied to UAVs based on (5.18) or (6.9) to maintain the shape. During the flight, on-board computers calculate the inverse kinematics (the formation variables based on the positions of UAVs), and then compare it with their neighbours and the formation centroid to obtain position errors. Those errors are then eliminated by the tracking control action generated.

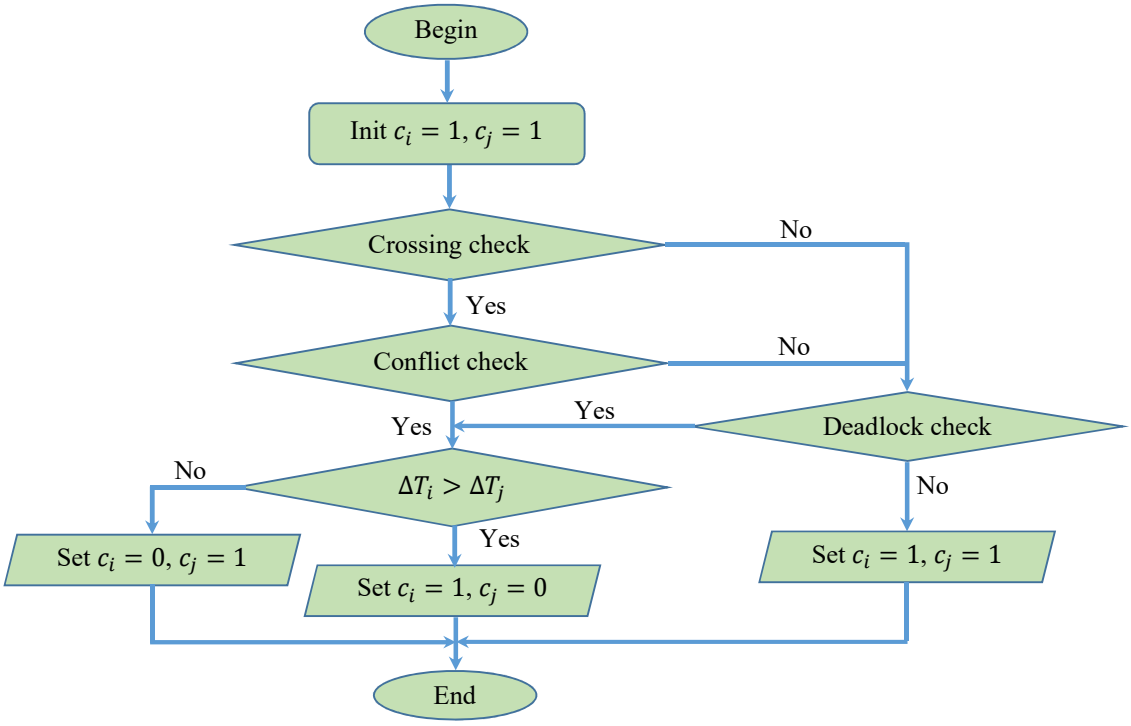


Figure 4.3 : Initialisation process for UAV formation

4.4 Experimental quadcopter and hardware development

4.4.1 Experimental quadcopter

The quadrotor platform used for the experiments presented in this thesis is the 3DR Solo. It weighs about 1.5 kg itself or 1.8 kg with GoPro camera and Solo Gimbal. The drone has a flight time of about 25 minutes or 20 minutes with the 420 g maximum payload. All sensor data from the drone can be wirelessly transmitted to a ground station PC either running Windows or Linux. The main processor on Solo is the iMX.6 running Yocto Linux connected to a Pixhawk autopilot.

The iMX.6 is a series of applications processors based on the ARM Cortex architecture. On Solo, iMX.6 controls the high-level operation of the drone: smart shots, camera and gimbal control, mobile app communication, and accessory interaction are all implemented in this layer. It is equipped with three processors, two are Cor-

tex M4 168 MHz running Pixhawk firmware for low-level control and the other is an ARM Cortex A9 running Arducopter flight operating system.

Pixhawk flight controller is a flexible autopilot hardware design based on the so-called project for the academic, hobby and developer communities. On Solo, Pixhawk autopilot is applied to control flight modes, drone stability and handles all attitude estimation, inertial navigation, and failsafe monitoring for Solo. It receives data from internal sensors, the external GPS module, the external compass module, and 3DR Solo Link to calculate Solo's in-flight dynamics. Pixhawk outputs telemetry data to the 3DR Link network and sends control commands to Solo's four motors via the electronic speed controllers. Pixhawk communicates with Solo onboard computer and ground control stations via the MAVLink telemetry protocol.

Figure 4.4 shows the communication protocol designed for 3DR Solo drone via Solo Link for Solo development. This module manages communication between Solo and the Controller on the 3DR Link secure Wi-Fi network. Solo Link receives all control inputs, outputs telemetry, and outputs video signals to communicate with the ground over the 3DR Link network. Solo Link also runs software processes that regulate advanced automated functions and data conversion. However, this protocol has not been assured so far, so that ground stations or companion computers must periodically check the status of the UAVs in order to confirm whether or not a command has been accomplished. This procedure is called the matching acknowledgement (ACK). Thus, there is a demand to improve the communication network.

The drone is equipped with a wide array of sensors included in the following details:

- Inertial measurement units (IMUs): the integrated 6-axis motion tracking device that combines a 3-axis gyroscope, 3-axis accelerometer, and a digital motion

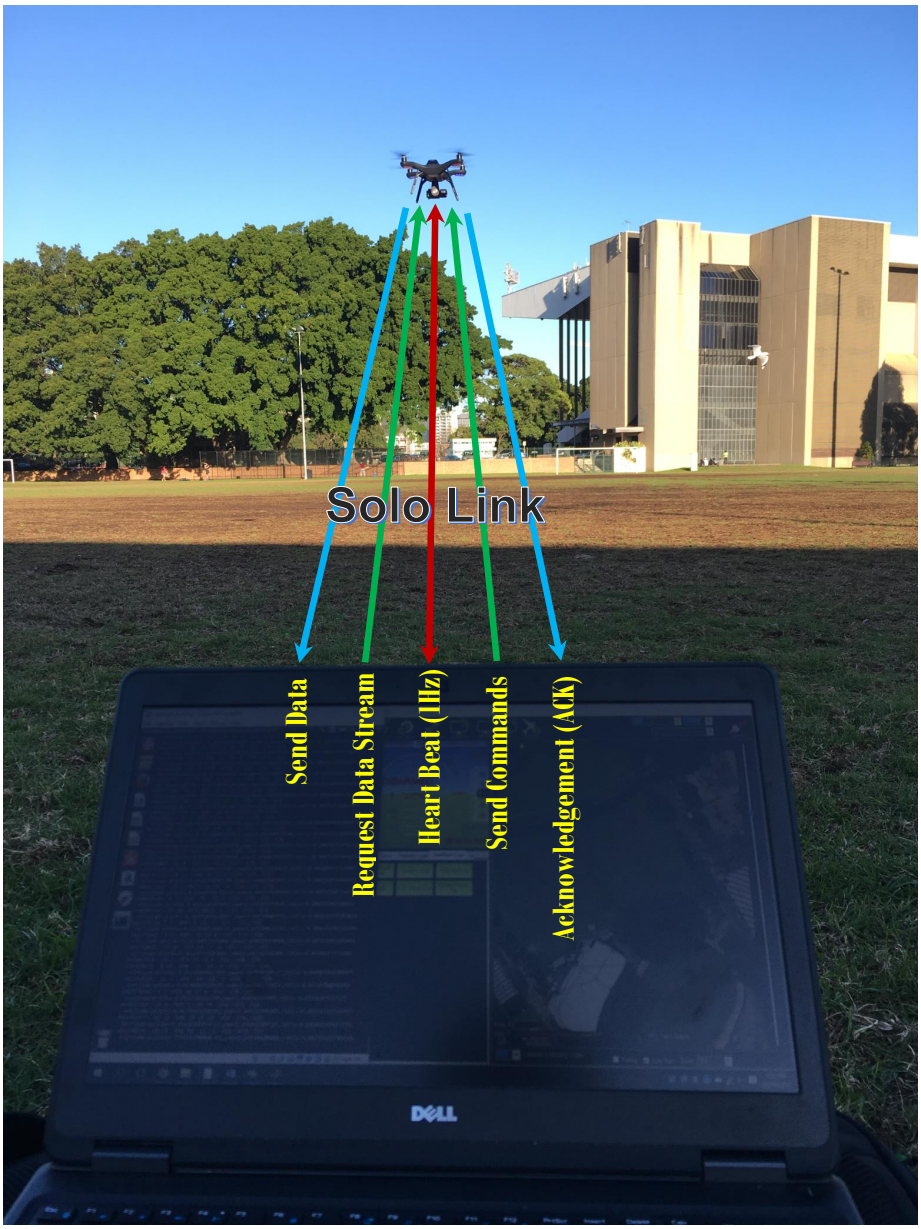


Figure 4.4 : Message flows between 3DR Solo and GCS via Solo Link

processor.

- Altitude meter: the barometric pressure sensor with an altitude resolution of 10 cm.

- Other devices including a compass module and a GPS Module for the drone navigation, four electronic speed controllers (ESCs), and a twin dipole antenna for

Solo wi-fi network (Solo Link).

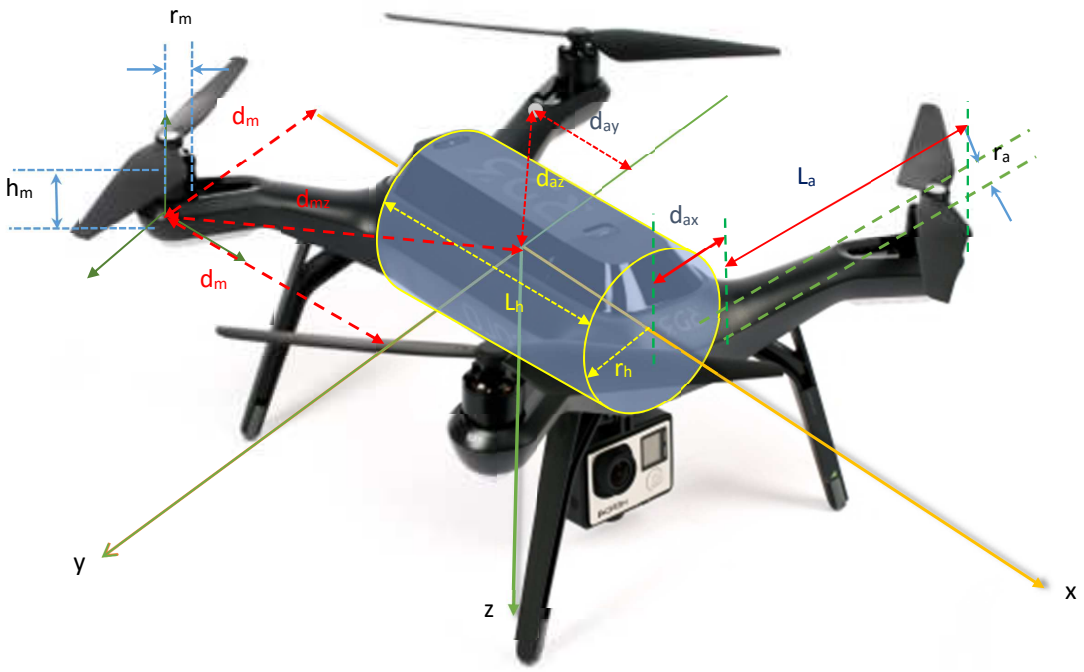


Figure 4.5 : The 3DR Solo drone with body coordinate frame

The programming is carried out through the ground control station called Mission Planner (MP) and uploaded to the UAV. MP is an application for the ArduPilot open source autopilot project. In this work, some main functions of MP can be listed as:

- Environmental data acquisition via a satellite map.

- Load the firmware into the autopilot board.

- Load autonomous missions, which are usually the resulting paths generated by the proposed path planning technique, into Solo autopilot.

- Download and analyse data logs in order to review them after each field test.

- Monitor flight status while in operation.

Through the literature, it is recognisable that UAV-based inspection systems require the synchronisation in control, path planning and data processing not only between layers within single UAVs but also among them. For real-time inspection, extra communication channels however are needed among UAVs, ground control stations (GCS), and other remote computational units (RCU) such as server computers or cloud computing systems to tackle the high demand of data fusion and processing. Those requirements pose the need for a homogeneous communication platform that allows all components of an inspection system to be integrated. The current use of radio transceiver modules with a pulse mode modulation 2.4GHz uplink and 2.4/5.8GHz downlink is insufficient as they can only form a private network among on-site devices [183]. Some studies suggested the deployment of additional UAVs as communication relay stations to extend the network [91,28]. Although this approach is suitable for tasks that cover a certain structure or area, it is ill-use for structures located in remote areas like bridges, power lines or wind turbines which require hundreds-of-kilometre communication range. Several studies proposed to use satellite communication to overcome this problem [42, 209]. This is, however, expensive, complex and not always feasible.

When operating outdoors, Solo may require up to five minutes to acquire a strong GPS lock. Solo also requires to select a clear view of the sky to improve GPS signal strength when the number of satellites is 6 and above. These conditions are sometimes not easy to satisfy. On the other hand, the horizontal position and speed accuracy are up to 5 metres and 1 m/s, respectively. These limitations are challenging for collision avoidance and inspection performance that requires more accurate distances to inspected objects.

Therefore, Solo is retrofitted to improve flight safety and its mission efficiency. The hardware extension together with communication development are presented in the following subsection.

4.4.2 Hardware extension and communication development

Figure 4.6 presents an overview of the proposed inspection system. Its core includes three UAVs equipped with cameras to collect data of the inspecting structure. The UAVs, via IoT boards, form a private network called SkyNet that based on it they can exchange data to fly cooperatively. The UAVs, at the same time, also share data with GCS via a wireless gateway router that is shown in Figure 4.8. The router relays the data via the Internet to RCU where the data will be processed to detect defects. The localisation of UAVs is conducted via GPS modules combined with IMUs that monitor internal states of UAVs. Depending on the inspection task, real-time kinematic (RTK) GPS modules will be used to improve the positioning accuracy. The operation of each UAV is monitored via its accompanied remote controller which will take over the control in emergency situations.

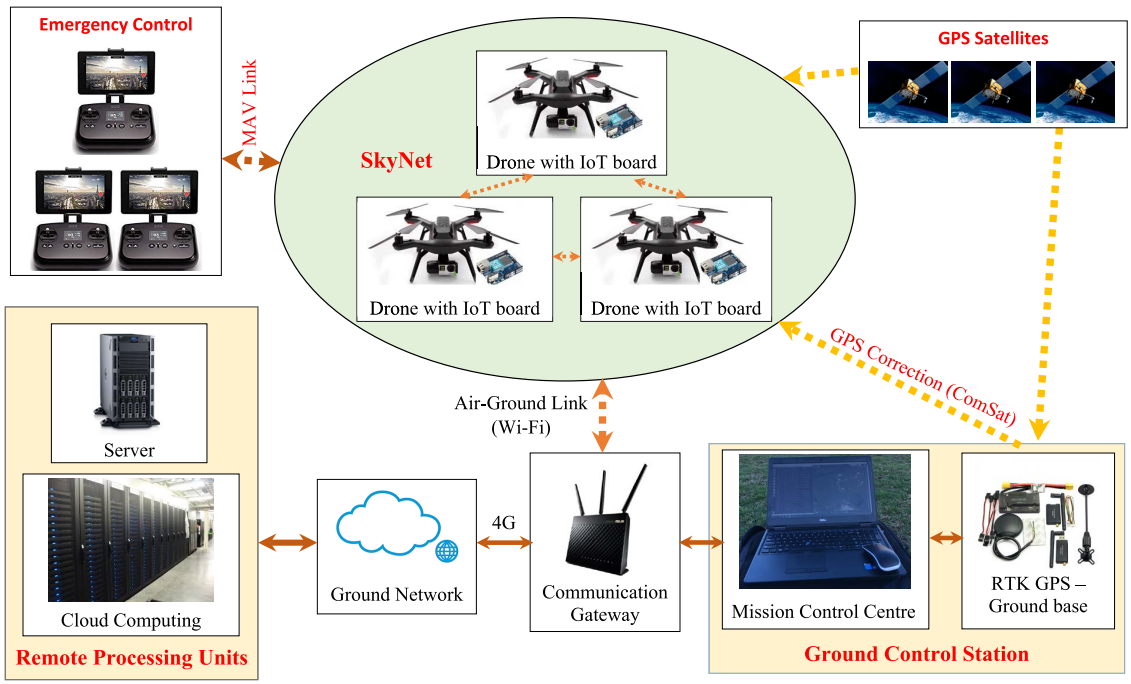


Figure 4.6 : Data communication structure

4.4.3 IoT Devices

As illustrated in Figure 4.6, the inspection system involves real-time cooperation of many components within the IoT framework. As not all components are pre-installed with networking and data processing capabilities, they are equipped with IoT boards, each including a processor and a microcontroller as shown in Figure 4.7. The processor provides devices capabilities to not only connect to the wifi network but also process the receiving data via Linux operating system installed. The microcontroller enables the devices to work with other low hardware interfaces such as voltage/current interface, AD/DA converter, PWM, etc. The board thus turns a normal device into a smart one that can be integrated into the Internet protocol (IP) based networks. Nevertheless, the built-in antennas of IoT boards are insufficient for outdoor communication as they are designed for indoor applications. High gain external antennas are therefore needed to extend the communication range as well as provide stable signals, especially for moving objects like UAVs. In this system, a 6 dBi detachable antenna is used for each IoT board.

For the Internet access, mobile broadband networks are utilised based on wireless gateway routers that have SIM card slots. As mobile broadband networks are present almost everywhere, this approach allows the inspection system to be deployed for any structure without the need for relay stations nor satellite communication.

4.4.4 Communication Protocols

Apart from the hardware, transport protocols also play an important role in ensuring the security and efficiency of the data exchanged. In IoT, the most popular transport protocols include the Transmission Control Protocol (TCP), User Datagram Protocol (UDP), and Real-Time Transport Protocol (RTP). TCP is a sophisticated protocol which was originally designed for the reliable transmission of static data such as emails and files over low-bandwidth, high-error-rate networks.

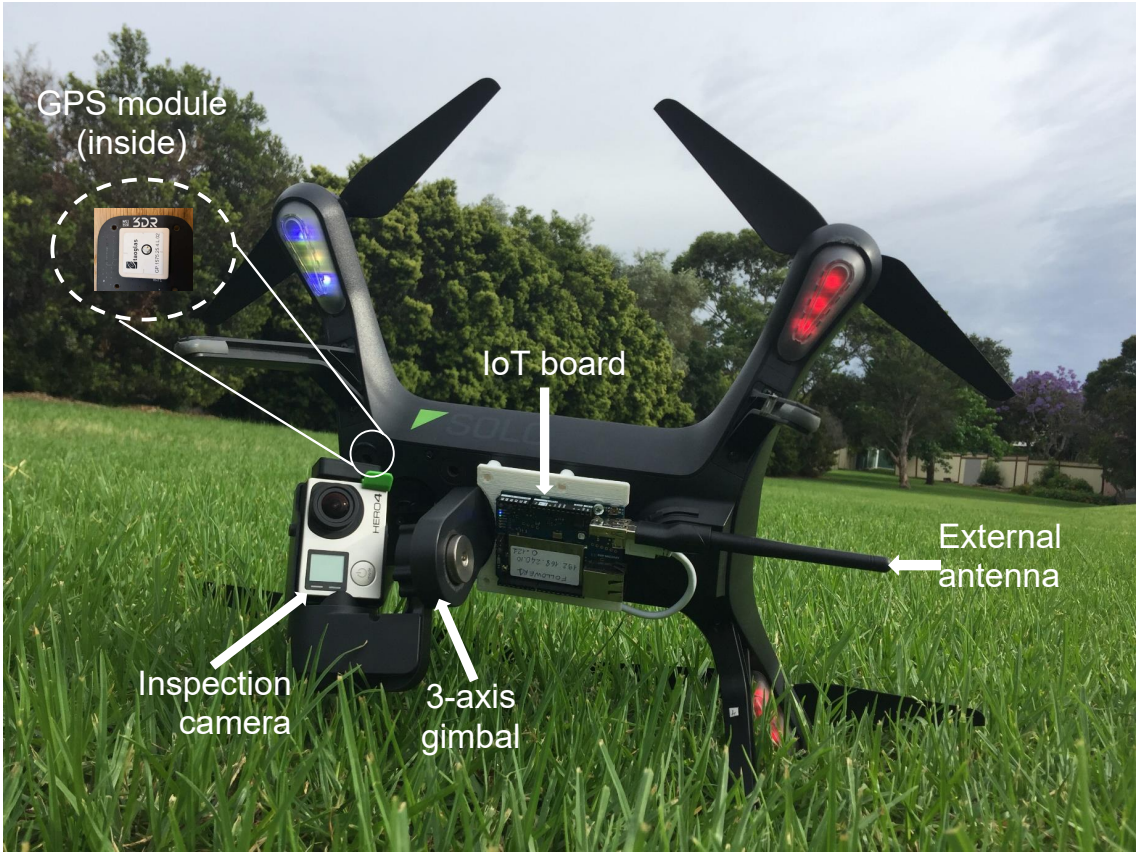


Figure 4.7 : The 3DR Solo drone with retrofitted components

UDP is based on the idea of sending a datagram from a device to another as fast as possible without considering the state of the network. UDP is thus able to minimise the transmission delay and delay jitter achieved under good network conditions. On top of UDP, a relatively new transport protocol called RTP was developed and has become the standard for delivering real-time multimedia data. RTP provides facility for jitter compensation and detection of out-of-sequence arrival in data. According to those protocol features, the system employs RTP for photo/video streaming and UDP for sensing data transmitting whereas TCP is chosen for delivering administrative data and control commands.

Figure 4.8 : Three Solo drones with the gateway router

4.4.5 Data Processing

One of the main benefits of using IoT is the capability of conducting computation and data processing at various levels of the system, for example, the device level by IoT boards, the control level by built-in computers of UAVs and the application level by networked server computers or cloud computing services. The determination of processing level depends on the amount of data to be processed and real-time requirement. In this system, IoT boards are used for processing communication data among UAVs such as position, velocity and other state information to minimise the latency. The control algorithm on the other hand is handled by the built-in computers of UAVs to enhance reliability. Other information and especially collected image data are processed by server computers to cope with the high demand in computation and energy consumption.

4.4.6 Other hardware retrofitting

Figure 4.7 also shows the hardware extension for a single 3DR Solo drone. Apart from the IoT board, other retrofitted devices include a Light Detection and Ranging (LIDAR) scanner, a RTK compatible GPS receiver, and a camera for environmental data acquisition.

The LIDAR sensor is fitted on top of the drone for flight safety and mapping. A 25D mm Metal Gearmotors is fitted coincidently to give the LIDAR a 360^o rotation. This design helps the LIDAR capability to generates two-dimensional shape and distance to surrounding environments. Its mission is to carry out continuous horizontal scan with high precision among a radius of 15m to build up a 3D map. Besides that, the obtained distances to obstacles will be processed for free crashing. To improve

Figure 4.9 : The 3DR Solo drone with LIDAR

accuracy of the drone' navigation capability, the onboard GPS receivers are updated by a centimeter precision GNSS devices, which is the 3DR Solo GPS u-blox NEO-M8P-2-10 incorporating a TW2712 Tallysman Wireless antenna. These quadcopters are also able to monitor wirelessly from the central processing computer via a radio network using Radio Telemetry designed by DroTex. It periodically updates the correction data from the base GPS station to improve the positioning precision.

The camera used is a Hero 4 Black with the focal length of 34.4 mm, resolution of 12 megapixels, and wireless network capability. It is attached to a three-axis gimbal with one degree of freedom for controlling its elevation (pitch) angle. The photos taken by this camera will be streamed to the RCU where the defect detection

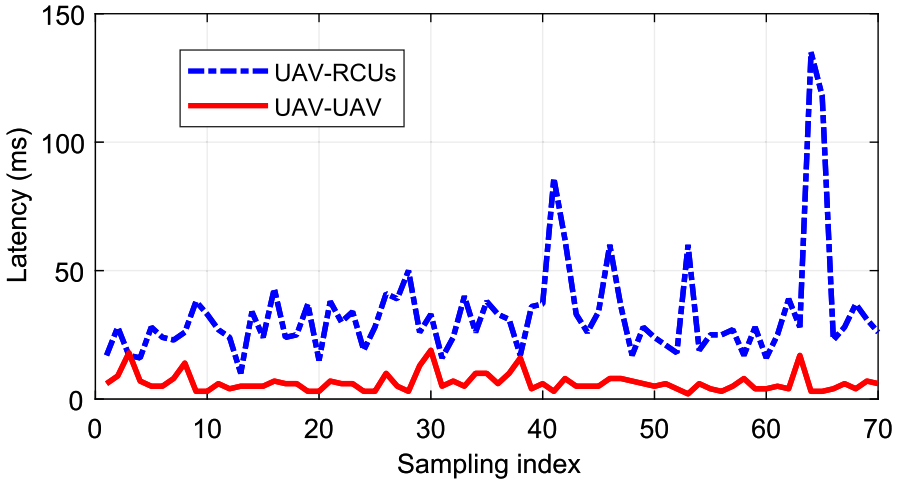


Figure 4.10 : Communication delay between UAVs (red solid line) and between UAVs and RCUs (blue dashed line)

algorithm is run.

To evaluate communication performance, the network latency recorded during experiments is presented in Figure 4.10. Here, there are two types of latency involved, one relates to the communication between UAVs within the on-site local network and the other includes the communication between UAVs and RCUs via 4G networks. It can be seen that the communication delay and jitter between UAVs are rather small at the average of 6.4 ms and 3.6 ms, respectively whereas those values for the communication between UAVs and RCUs are 32.6 ms and 20.2 ms. Those results are explainable as the 4G network obviously requires more intermediate relay stations during data transmission. More importantly, they show that the proposed IoT-based architecture, on one hand, preserves the fast communication between UAVs necessary for formation control while on the other hand, it allows additional resources to be included for real-time data processing which is essential for structural inspection.

4.5 Summary

This chapter presented the framework employed in this thesis. The multi-layer architecture has been introduced to allow various modules of a complicated system to be integrated to fulfil a common task. The system features a new communication platform based on the Internet of Things which can exploit vast processing capabilities of RCU and remove the burdens in communication distance. This chapter also detailed the experimental drones and the hardware development that would be used throughout the dissertation to evaluate the performance of all the designed approaches.

Chapter 5

Angle-encoded swarm optimisation for UAV formation path planning

5.1 Introduction

This chapter presents a path planning algorithm for the triangular formation model of UAVs, given in section 4.3.3, conducting inspection tasks of built infrastructure. The chapter covers two main parts; these are the proposed formation path planning algorithm, and experiment results.

The first part of the chapter presents a novel and feasible path planning technique for a group of unmanned aerial vehicles conducting surface inspection of infrastructure. The approach begins with a brief description of the angle-encoded PSO, followed by theoretical analysis and review to justify the selection of the θ-PSO. A new constraint function is developed to incorporate with θ-PSO to generate the desired trajectory for the centroid of the group. Finally, the generated path is translated into an individual track for each UAV based on its defined position in the formation.

Extensive simulation, comparison, and experiments are presented in the second part to illustrate the validity and effectiveness of the proposed path planning algorithm.

5.2 Formation path planning

When generating a path for the desired motion of multiple UAVs in a group, a number of constraints are required to be fulfilled for maintenance of the formation,

maneuverability of every single UAV, operating space, and obstacle avoidance. In this work, all of the constraints will be incorporated into a multi-objective function. The path planning problem can be then simplified to the creation of a feasible path for the centroid of the UAV formation. Since the major goal is to construct optimal paths for all UAVs in the group, it is essential to speed up the convergence of the optimisation process for the whole formation. Therefore, the angle-coded PSO (θ-PSO) will be used as follows.

5.2.1 Angle-encoded PSO or θ-PSO

The angle-encoded PSO [200] is described as:

$$\begin{cases} \Delta\theta_{ij}^{k+1} = w\Delta\theta_{ij}^{k} + c_1 r_{1i}^{k}(\lambda_{ij}^{k} - \theta_{ij}^{k}) + c_2 r_{1i}^{k}(\lambda_{gj}^{k} - \theta_{ij}^{k}) \\ \theta_{ij}^{k+1} = \theta_{ij}^{k} + \Delta\theta_{ij}^{k+1}, (i = 1, 2, ..., N; j = 1, 2, ..., S) \\ x_{ij}^{k} = \dfrac{1}{2}\left[(x_{max} - x_{min})sin\left(\theta_{ij}^{k}\right) + x_{max} + x_{min}\right], \end{cases} \tag{5.1}$$

where N is the swarm size, S is the dimension of the searching space, w is the inertial weight, r_1 and r_2 are two pseudorandom scalars, c_1 and c_2 are the gain coefficients, $\theta_{ij} \in [-\pi/2, \pi/2]$ and $\Delta\theta_{ij} \in [-\pi/2, \pi/2]$ are respectively the phase angle and phase angle increment of the ith particle in dimension j; $\Lambda_g = [\lambda_{g1}, \lambda_{g2}, ..., \lambda_{gS}]$ and $\Lambda_i = [\lambda_{i1}, \lambda_{i2}, ..., \lambda_{iS}]$ are respectively the ith particle's global and personal best positions; and x_{max} and x_{min} are the upper and lower restrictions of the search space.

For path planning of the centroid, each particle is associated with a specific path instance T_{Fi} and the phase angle-encoded population can be presented as:

$$\Theta = [\Theta_1, \Theta_2, ..., \Theta_N]^T. \tag{5.2}$$

Suppose each path T_{Fi} consists of $v+2$ waypoints, including the start and target ones. As those start and target waypoints are predetermined, they can be excluded from the particle. Thus each particle has the dimension of $3v$ and can be represented

by the following fixed-length phase angle-encoded vector:

$$\Theta_i = [\theta_{i1}, ..., \theta_{iv}, \theta_{i,v+1}, ..., \theta_{i,2v}, \theta_{i,2v+1}, ..., \theta_{i,3v}]. \tag{5.3}$$

Using the mapping $f : [-\pi/2, \pi/2] \rightarrow [x_{min}, x_{max}]$, the position of a particle is obtained as:

$$X_i = f(\Theta_i) = [x_{i1}, ..., x_{iv}, x_{i,v+1}, ..., x_{i,2v}, x_{i,2v+1}, ..., x_{i,3v}], \tag{5.4}$$

where $x_{ij} = f(\theta_{ij})$ is the jth dimension of the ith particle's position ($i = 1, ..., N$; $j = 1, ..., 3v$), $x_{i1}, ..., x_{iv}, x_{i,v+1}, ..., x_{i,2v}$ and $x_{i,2v+1}, ..., x_{i,3v}$ represent the x, y and z coordinates of the vth waypoint of path T_{Fi}, respectively.

5.2.2 Rationale

The θ-PSO is a variant of the well-known particle swarm optimisation, which is a population-based stochastic optimisation algorithm inspired by social behavior of bird flocking [178, 62]. PSO is applicable for optimisation problems in multi-dimensional search spaces [178, 158, 96, 59]. The algorithm provides a high-quality optimal solution in a short time interval. The convergence of PSO is faster in comparison with some stochastic approaches [158,51,156]. Importantly, this method is beneficial when it can be easily implemented [18].

In PSO, a set of particles is generated, each seeks for the optimum solution by moving in a way that is a compromise between its own experience and the social experience. Initially, each particle is assigned a random position, x_i, and velocity, v_i. The particle motion is then updated by the following equations:

$$\begin{aligned} v_{ij}^{k+1} &= w v_{ij}^{k} + c_1 r_{1i}^{k}(p_{ij}^{k} - x_{ij}^{k}) + c_2 r_{1i}^{k}(p_{gj}^{k} - x_{ij}^{k}) \\ x_{ij}^{k+1} &= x_{ij}^{k} + v_{ij}^{k+1}, (i = 1, 2, ..., N; j = 1, 2, ..., S), \end{aligned} \tag{5.5}$$

where p_{ij} and p_{gj} are the local-best and global-best positions of the particle i, and subscript k is the iteration index. The values of p_{ij} and p_{gj} are evaluated based on

a cost function to be defined in the next section. The rest of the variables in (5.5) are similar to those in (5.1).

In Eq. (5.5), it can be seen that a solution of a particle has these three alternative options: to track its private trajectory, to follow its best prior position, or to move toward the global best position. The correlation among them depends on three coefficients w, c_1 and c_2. In this case, w plays a crucial role in the convergence performance when it decides the impact of previous velocities on the current one [158]. This parameter adjusts the relationship between the local and global search capabilities of the entire swarm [62].

Besides that, a precise selection of the acceleration coefficients will increase the convergence speed and attenuate the local minima [50]. The values of c_1 and c_2 are selected to be equal to 2.0 or close values by most of the previous researchers. However, this selection may be unsafe to the optimisation process. An often-used solution is to keep the velocities in a boundary range. However, the bound of velocity also generates some other problems such as no justifiable law is determined. As a result, this velocity range may break down the particle's trajectory convergence [158].

The angle-encoded PSO [200,62] was developed to overcome the above problems. Compared to the conventional PSO, the following advantages can be achieved by θ-PSO: higher optimisation precision, faster convergence, fast optimisation time and rate and higher capability to deal with high-dimensional search spaces [77, 43].

For traditional path planning, the position of particles often represents the location of UAVs. This representation can give good results, but it also slows down the swarm convergence if the momenta of particles are not well-adjusted [96, 62]. Furthermore, when producing a path for the desired motion of multiple UAVs in a group, a number of constraints are required to be fulfilled for maintenance of the formation, manoeuvrability of every single UAV, operating space, and obsta-

cle avoidance. All of the constraints will be incorporated into a multi-objective function, that makes the planning problem more complicated. Since the goal is to construct optimal paths for all UAVs in the group, it is essential to speed up the convergence of the optimisation process for the whole formation. The advantageous properties of θ-PSO become essential for path planning problems, particularly in online applications. Therefore, the angle-coded PSO was selected.

5.2.3 Cost function

The selection of a proper cost function for the PSO is essential to achieve the globally optimal solution for the searching process. The cost function for any trajectory is often formed by two major evaluations, the length and violation cost of the path. The former helps to minimise the total travelling distance of the path whereas the latter is to avoid collisions of UAVs with each other and with obstacles. In a 3D site, other constraints should also be required, e.g., restrictions in flying altitude, heading angle and path curve. Quadcopters are used in this work thanks to their capabilities to carry out sharp and abrupt changes in angles and curves [74]. Thus, the constraints in angle and curve can be relaxed. The multi-objective constraint is now incorporated into the cost function in the following form:

$$J_F(T_{Fi}) = \sum_{m=1}^{3} \beta_m J_m(T_{Fi}), \tag{5.6}$$

where T_{Fi} is the formation path i to be evaluated; β_m is the weighting factor indicating the corresponding threat intensity; and $J_m(T_{Fi})$, $m = 1, 2, 3$, are the costs associated with the path length, collision violation and flying altitude, respectively. To determine $J_m(T_{Fi})$, the formation path T_{Fi} is split into L_i segments, where L_i is chosen to be sufficiently large so that each segment can be considered to be straight and represented by coordinates of ending nodes $P_{i,l} = \{x_{i,l}, y_{i,l}, z_{i,l}\}, l = 0, \cdots, L_i$. By denoting the length of the segment connecting nodes $P_{i,l}$ and $P_{i,l-1}$ as the Euclidean norm $\left\|P_{i,l} - P_{i,l-1}\right\|$, the cost J_1 corresponding to the path length is then

calculated for all segments:

$$J_1(T_{Fi}) = \sum_{l=1}^{L_i} \left\| P_{i,l} - P_{i,l-1} \right\| . \tag{5.7}$$

Let K be the total number of all obstacles for a given UAV within its operation space. Assume that each obstacle is prescribed in a cylinder with the center's coordinate $C_k = \{x_k, y_k\}$ and radius r_k, as shown in Figure 5.1. The surfaces of cylinders then can be used to form constraints for obstacle avoidance. Denoting $M_l = \{x_l^M, y_l^M, z_l^M\}$ as the midpoint of segment l with the altitude z_l^M equivalent to its z coordinate in the inertial frame. The safe distance $d_{l,k}^S$ to obstacle k is calculated from the cylinder center to its surface at the altitude z_l^M as:

$$d_{l,k}^S = \begin{cases} \sqrt{r_k^2 + (z_l^M - z_k)^2} & \text{if } z_l^M \leq z_k^{max} \\ \sqrt{r_k^2 + (z_k^{max} - z_k)^2} & \text{if } z_l^M > z_k^{max}, \end{cases} \tag{5.8}$$

where z_k^{max} is the maximum altitude of obstacle k. To compute the violation cost

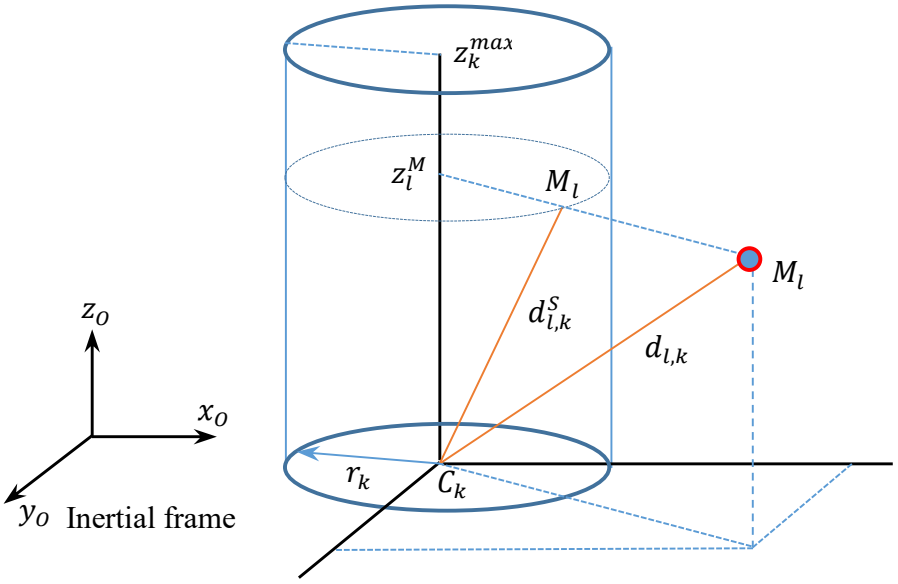


Figure 5.1 : Obstacle representation and safe distance calculation

between each generated path and obstacle centres, the formation is assumed to be

rigid and can be fit within a sphere with the radius

$$r_{l,k}^{S} = r_Q + r_F + d_{l,k}^{S}, \tag{5.9}$$

where r_Q is the radius of quadcopters including propellers, r_F is the radius of the formation,

$$r_F = \max(d_i), \tag{5.10}$$

in which d_i is the distance from quadcopter i to the formation centroid. The violation cost can now be derived as follows:

For the kth obstacle, compute the distance from its center C_k to M_l:

$$d_{l,k} = \sqrt{(x_l^M - x_k)^2 + (y_l^M - y_k)^2 + (z_l^M - z_k)^2}. \tag{5.11}$$

At a given altitude z_l^M, $d_{l,k}$ is then compared with the safe distance to the obstacle. The comparison results in the following violation function:

$$V_{l,k}(T_{Fi}) = \max(1 - \frac{d_{l,k}}{r_{l,k}^{S}}, 0). \tag{5.12}$$

This function ensures that the distance $d_{l,k}$ must be larger than the safe distance for obstacle avoidance. The violation cost is then computed for all obstacles as:

$$V_l(T_{Fi}) = \frac{1}{K} \sum_{k=1}^{K} V_{l,k}(T_{Fi}). \tag{5.13}$$

For all L_i segments, the final violation cost on average (with respect to the centroid) is represented as:

$$J_2(T_{Fi}) = \frac{1}{L_i} \sum_{l=1}^{L_i} V_l(T_{Fi}), \tag{5.14}$$

In terms of flying altitude, UAVs are often required to follow the terrain at a certain height to avoid crashing. Thus, the altitude of each UAV must be within a predefined interval between two given extrema, the minimum and maximum safe

clearances $z_{\min}$ and $z_{\max}$. Thus, the corresponding cost component can be expressed as

$$\begin{cases} J_3(T_{Fi}) = \sum_{l=1}^{L_i} \delta_l \\ \delta_l = \begin{cases} z_l^M - z_{\max}, & \text{if } z_l^M > z_{\max} \\ 0, & \text{if } z_{\min} \leq z_l^M \leq z_{\max} \\ z_{\min} - z_l^M, & \text{if } 0 < z_l^M < z_{\min} \\ \infty, & \text{if } z_l^M \leq 0. \end{cases} \end{cases} \tag{5.15}$$

This last condition is critical for safe operations as negative values of the altitude could be generated causing UAVs to crash to the ground.

5.2.4 Path planning implementation

The implementation starts with choosing the operation space of UAVs and the infrastructure to be inspected. This can be done by using a navigation map with satellite images. For example, here a monorail bridge as a testbed subject to inspection can be loaded on the Mission Planner, as shown in Figure 5.2. The obstacles as displayed in Figure 5.3 are also identified based on this map. Furthermore, range sensors such as lidars can be used to form a 3D map of the environment as in [156]. Based on those inputs, the cost function together with constraints can be defined as described in the previous section. The θ-PSO algorithm will then be run to obtain the desired path. The steps can be summarised as follows:

(1) Compute all parameters for UAVs and formation model, based on the desired task.

(2) Select appropriate parameters for PSO such as the population size, the number of iterations, gain coefficients and so on.

Figure 5.2 : Mission Planner incorporating Google Satellite Map to create initial information and an inspection plan

(3) Identify environmental information of the flying field and initialise parameters of obstacles.

(4) Initialise randomly the path of each particle from the start point to the target point.

(5) Evaluate each path based on the cost function (6.4).

(6) Compute each particle's personal best and the global best positions by running PSO repeatedly.

(7) The desired path T_F^* is chosen as the maximum number of iterations is reached.

5.2.5 Path generation for individual UAV

Given the optimal path, T_F^*, generated by the θ-PSO for the formation centroid, it is necessary to produce a specific path for each individual UAV so that the shape of the formation during the flight can be maintained. Those paths can be computed

based on the PSO's generated path and the desired relative distances among the UAVs. Let $P_{n,d} = [x_{n,d}, y_{n,d}, z_{n,d}]^T$ be the reference position for each UAV during the flight and $P_n = [x_n, y_n, z_n]^T$ be the actual position; P_n can be obtained from GPS data of the nth UAV. The relative position errors of the nth UAV during the flight in the inertial frame are determined as:

$$\begin{bmatrix} e_{n,x} \\ e_{n,y} \\ e_{n,z} \end{bmatrix} = \begin{bmatrix} x_{n,d} - x_n \\ y_{n,d} - y_n \\ z_{n,d} - z_n \end{bmatrix}, \tag{5.16}$$

Using the inverse rotation matrix, as $R_{FO}(t) = R_{OF}^{-1}(t)$, the errors in (5.16) can be converted into the errors in the formation frame as:

$$\begin{bmatrix} e_{n,x}^F \\ e_{n,y}^F \\ e_{n,z}^F \end{bmatrix} = R_{FO}(t) \begin{bmatrix} e_{n,x} \\ e_{n,y} \\ e_{n,z} \end{bmatrix}. \tag{5.17}$$

The customised path for each UAV can then be represented in terms of trajectory control command as:

$$T_n = T_F^* + \Delta T_n, \tag{5.18}$$

where T_F^* is the trajectory of the formation centroid, computed by θ-PSO, and ΔT_n is the amount added to direct the UAV away from the centroid. ΔT_n is calculated from the desired relative distances among the UAVs and the relative position errors in (5.17), $\Delta T_n = [e_{n,x}^F, e_{n,y}^F, e_{n,z}^F]^T$. The output T_n will be fed to the internal controller of UAV n for trajectory tracking.

5.2.6 Overall algorithm

Figure 6.2 present the pseudo code for the above path generation processes.

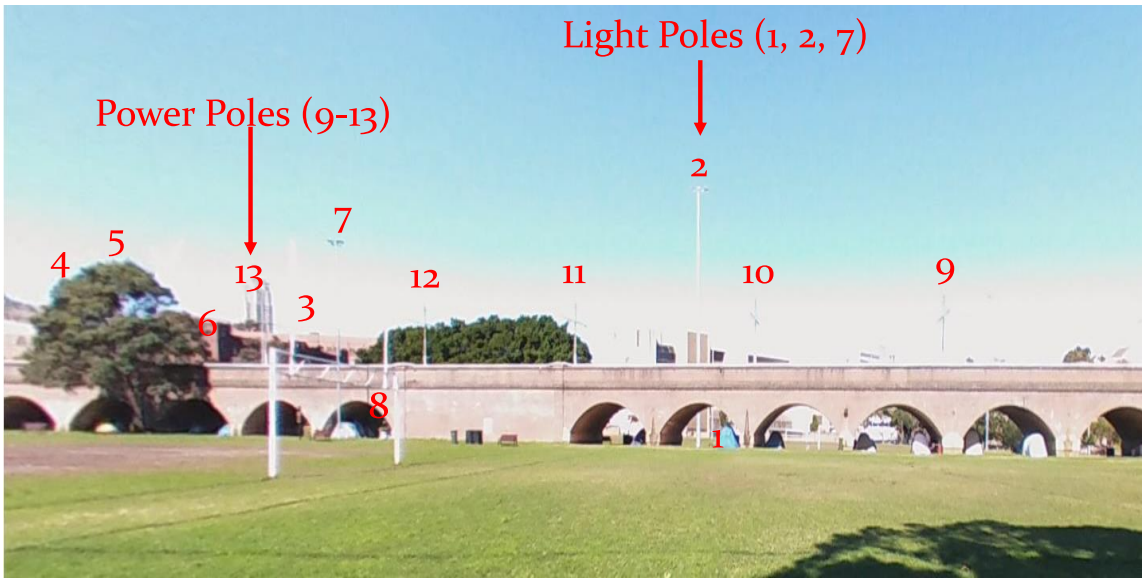


Figure 5.3 : The light rail bridge to be inspected with the obstacles identified

5.3 Experiments

A series of experiments have been conducted to evaluate the feasibility and efficiency of the proposed algorithm.

5.3.1 Experimental setup

The task assigned in experiments is to inspect simultaneously different surfaces of a bridge using three UAVs. As mentioned, the Mission Planner, a proprietary ground control station software, incorporating the Google Satellite Map (GST) is used to collect initial information about the structure and its surrounding environment. Here, the operation space is chosen with dimensions $\{141m \times 101m \times 40m\}$, equivalent to the GST coordinates $\{-33.87601, 151.191182, 0\}$ and $\{-33.875086, 151.192676, 40\}$, as illustrated in Figure 5.2. The starting and target points are set as $P_0 = \{40.0, 8.0, 30\}$ and $P_{v+1} = \{64, 108, 34\}$, respectively. Therein, ten obstacles are identified, each with a different radius.

The formation platform chosen in this work is three identical 3DR Solo drones as

```
/* Preparation:                                                          */
   Determine the inspection surface;
   Identify the upper and lower boundaries of the working space, start and target
   positions of the swarm;
   Identify obstacles in the working space, check and adjust obstacles' parameters if
   needed ;
   Group all the above data and save in a common file (init file);
/* Initialisation:                                                       */
   Initialise the working environment by loading the init file to global memory;
   Initialise the θ-PSO parameters, i.e., w, swarm_population, and swarm_iteration and
   generate a random path to connect the start and the target points;
   Set the range of constraints for each particle's phase angle and angular increment in
   [π/2, π/2];
/* θ-PSO:                                                               */
foreach i < (swarm_iteration) do
    foreach j < (swarm_population) do
        Calculate new phase angle increment value in the range of limitation;  /* using
         1st equation in (5.1) */
        Calculate new phase angle value in the range of limitation;        /* using 2nd
         equation in (5.1) */
        Calculate new position;                  /* using 3rd equation in (5.1) */
        Check Violation cost;                                  /* using (5.14) */
        Evaluate each path based on the Best_Costs and Violation cost;
        Update each particle_personal_best and the global_best positions;
    end
    Update global_best and Violation costs;
end
   Save global_best and Violation cost;
/* Path generation:                                                      */
   Final path is chosen as the maximum number of iterations is reached.;
   Generate individual paths for UAVs.                          /* using (5.18) */
```

Figure 5.4 : Pseudo code for path generation process.

shown in Figure 5.5, whereby the controllers at the low level for each UAV have been reported in Chapter 3. The communications among them were conducted by adding

Figure 5.5 : 3DR Solo drones used in experiments

an additional Internet-of-Things board to each drone and a base Wi-Fi router. The IoT boards together with the ground Wi-Fi station form a network that can connect to the internet to transmit the data to other processing station. Also through this network, the drones can exchange their position, velocity and status data during flight. By employing that information, the onboard computer calculates the inverse kinematics (the formation variables based on the positions of the robots), compares it with their neighbours and the formation centroid to obtain the position errors. Those errors are then eliminated by the tracking control action generated.

To conduct the path planning algorithm, the number of particles, waypoints, and iterations are respectively selected as 100, 10, and 300. Parameters of the three quadcopters with respect to the centroid of the formation are $\Delta T_1 = [0, 0, 2]$ m, $\Delta T_2 = [3, 0, -1]$ m and $\Delta T_3 = [-3, 0, -1]$ m. The minimum and maximum clearances between UAVs and the terrain are set to $z_{\max} = 32$ m and $z_{\min} = 28$ m, respectively.

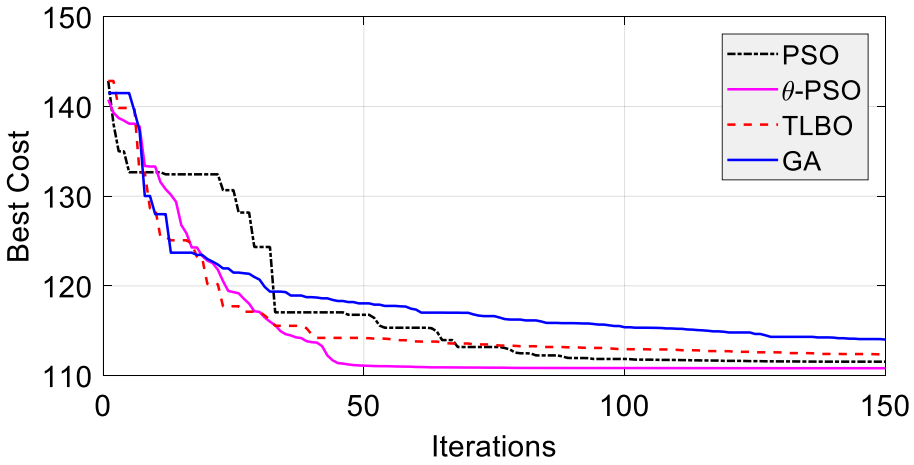


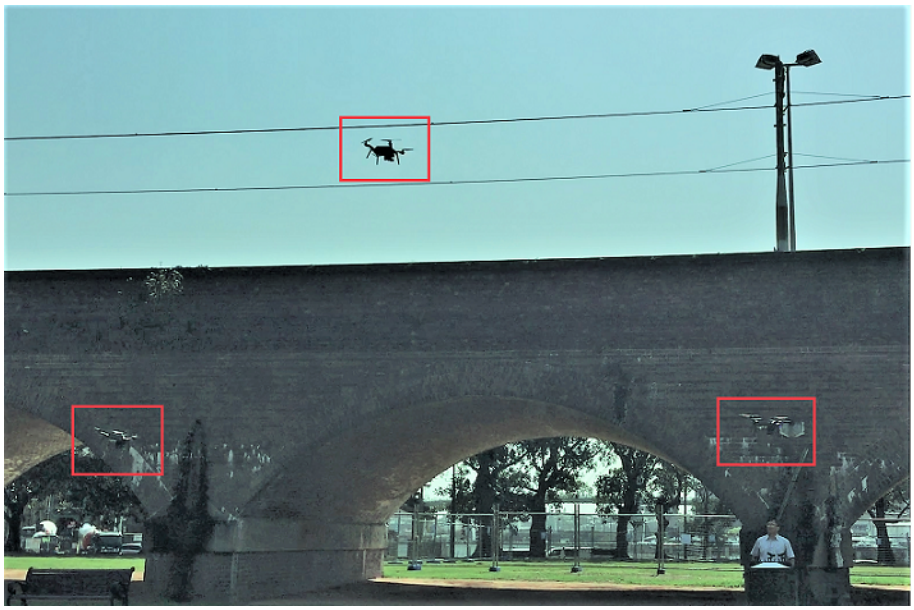
Figure 5.6 : Convergence comparison between PSO, GA, TLBO and θ-PSO

5.3.2 Results

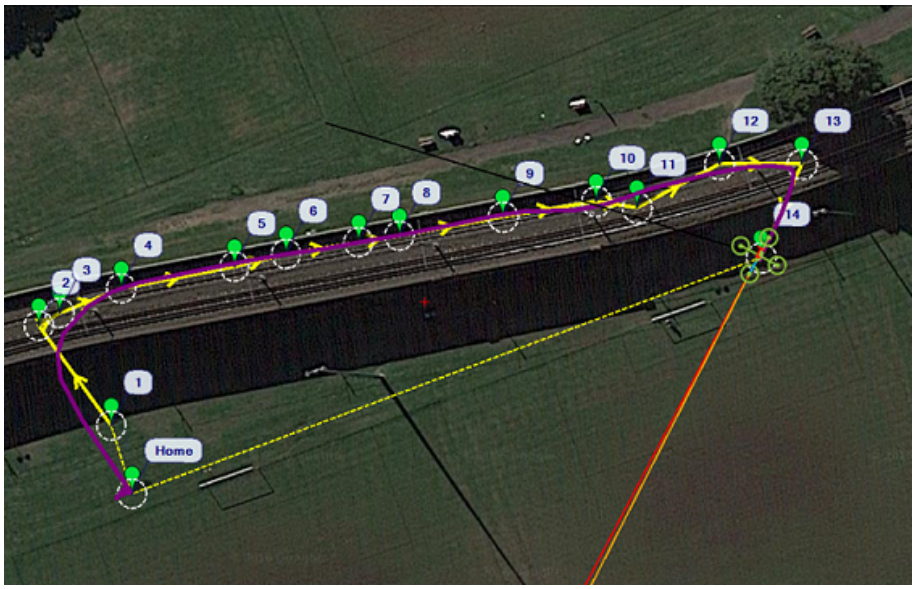
The path planning results are presented in this subsection, whereby it is expected that the designed method can generate collision-free paths for the three UAVs with sufficiently fast convergence using the proposed algorithm. For this, let us first compare the performance of the proposed θ-PSO with a conventional PSO algorithm and two other bio-inspired algorithms, Genetic Algorithm (GA) [151] and Teaching-Learning-Based Optimization (TLBO) [167]. Figure 5.6 shows the cost values over iterations. It can be seen that although both algorithms are convergent, the θ-PSO introduces a faster and more stable convergence. The results are confirmed as recorded in Table 5.1 which shows the average cost value and convergence iterations.

Table 5.1 : GA, TLBO, PSO and θ-PSO performance comparison

Algorithm	Min cost	Initial cost	Iterations
GA	114.17	142.91	147
PSO	113.80	142.84	84
PSO	112.43	143.0	102
θ-PSO	111.02	142.84	68



(a) Triangular UAV formation



(b) Planned path (yellow) and flown path (violet)

Figure 5.7 : Bridge inspection with UAV formation

Given the generated path, field-test experiments were conducted, in which the triangular formation automatically navigated along the inspected surface, as depicted in Figure 5.7. The 3D trajectories generated from that of the formation centroid path are shown in Figure 5.8, where it can be seen that the three drones can take off,

reach their individual altitude set-point, descend and finally arrive at their target position at almost the same interval, while maintaining the desired triangular shape. This result can be further verified via the altitude time responses of the three UAVs as recorded in Figure 5.9. It is clear that the quadcopters are capable of avoiding obstacles and preserving the desired formation configuration during the inspection task. For further evaluation, Figure 5.10 shows the error between the planned and flown paths computed by selecting the closest coordinates of the flight trajectories to the reference points. Those small errors, mainly caused by the positioning system of UAVs from the GPS signal received, imply feasibility and reasonable smoothness of the generated path for the deployment of the drone formation.

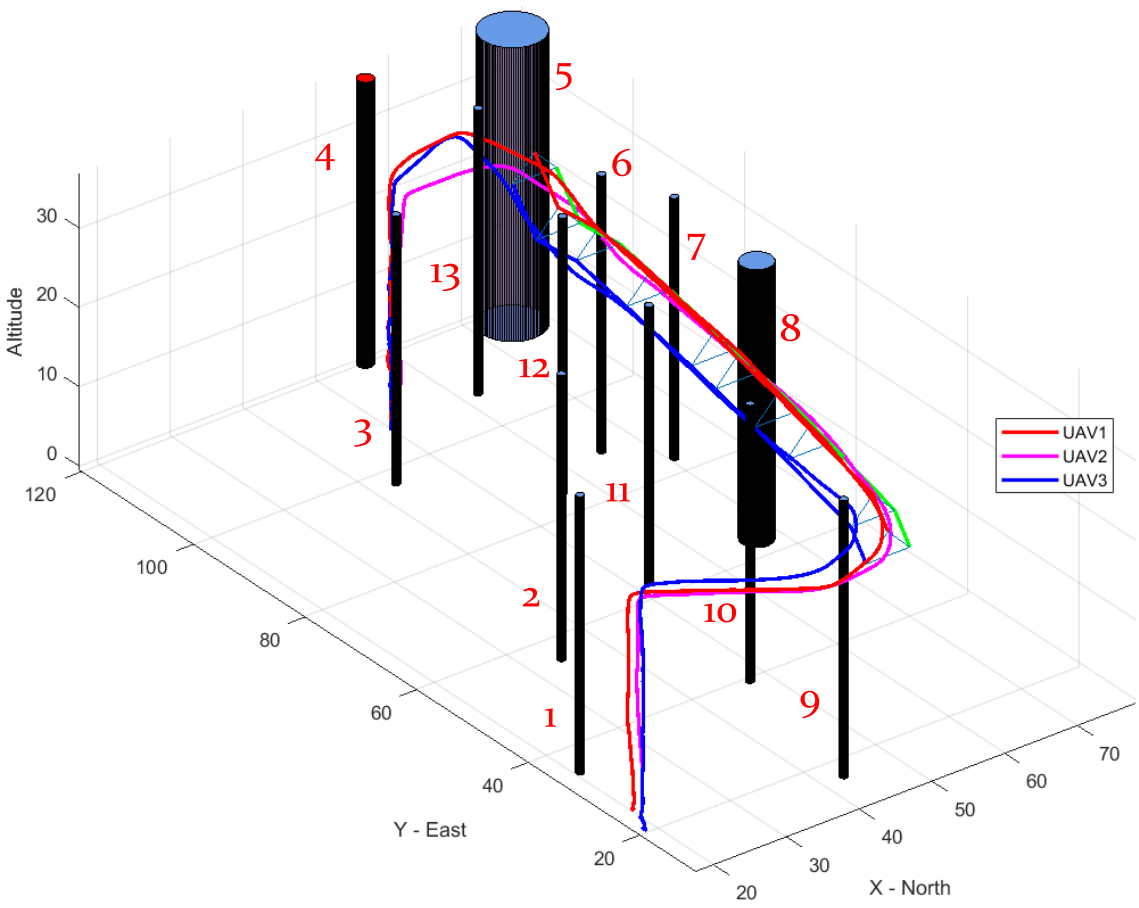


Figure 5.8 : Trajectories of three drones tracking the planned paths

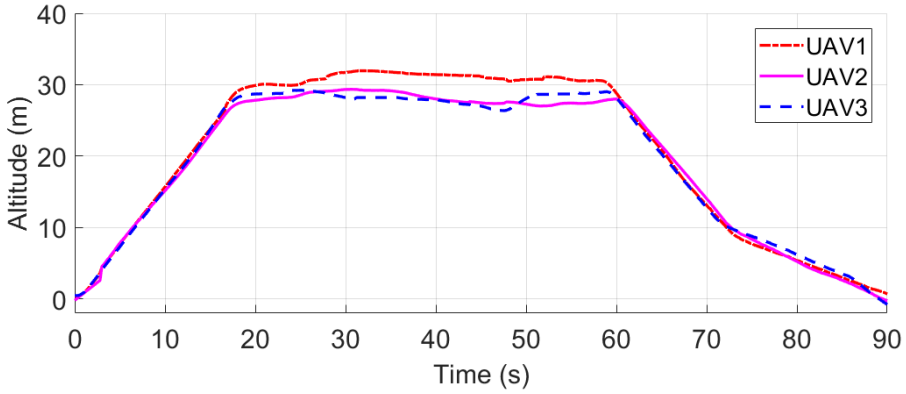


Figure 5.9 : Altitudes of the three drones in the formation test

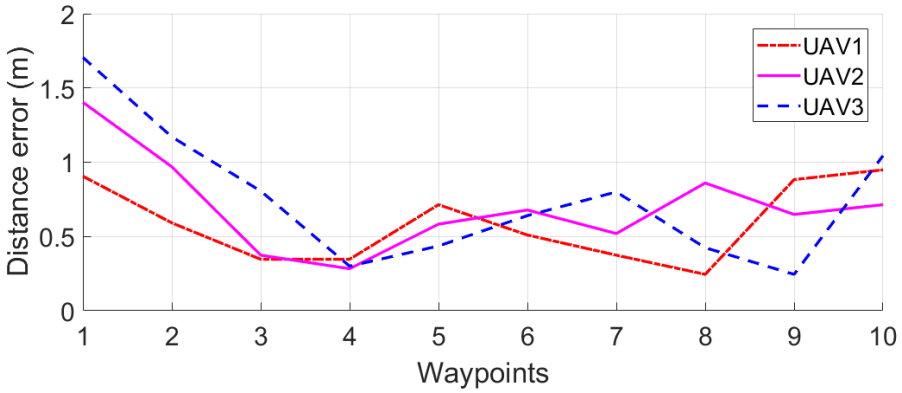


Figure 5.10 : Errors between the planned and flown paths

5.4 Conclusion

The chapter has presented a novel approach for the path planning problem of multiple UAVs navigating in a desired shape for infrastructure inspection tasks. Here the angle-coded PSO is proposed to find feasible and obstacle-free paths for the whole formation by minimising a cost function that incorporates multiple constraints for shortest paths and safe operation of the drones. From the centroid, customised paths are generated for each individual UAV to maintain the formation using a proprietary software while inter-UAV communication is achieved via the IoT boards. Implementation on a triangular formation is reported along with field tests on a

monorail bridge. The results confirmed the validity and feasibility of the proposed approach for UAV formation inspection of built infrastructure.

Chapter 6

Reconfigurable multi-UAV formation

6.1 Introduction

This chapter provides a proposed algorithm for the reconfigurable formation of multiple UAVs used in visual-based inspection of infrastructure, in which the path planning algorithm in Chapter 5 is inherited and developed to generate a feasible path for the overall formation configuration, taking into account the constraints for visual inspection. The rest of the chapter is organised as follows.

The methodology is a combination of the path planning result and a reconfigurable strategy in order to complete a safe trajectory for an individual UAV during its operation in formation. It begins with some potential transforming shapes, followed by the introduction of the intermediate waypoint and ends with the individual trajectory generation for each UAV and the overall algorithm.

The next section provides the experimental setup and results, which illustrate the validity and effectiveness of the proposed algorithm. The chapter ends with a conclusion.

6.2 Methodology

6.2.1 Introduction of UAV formation topologies

The triangular formation in Chapter 4 can be used to coordinate the UAVs for inspection tasks given that the formation is treated as a rigid body. Under that assumption, the path generated in Section 5.2 can be directly used as the reference

for the formation centroid. In practice, the rigid-body assumption is not always held as the UAVs may need to change the formation shape to adapt to the operating environment. For example, a narrow passage or an unwanted obstacle may require the UAVs to fly in a row or column instead of the triangular formation. As explained in Figure 6.1 the following reconfigurations are considered in this study:

- Alignment: The UAVs form a line. It is used for the scenarios of appearing narrow passages/obstacles where it is only possible for a single UAV to pass.
- Rotation: The UAVs rotate as a rigid body structure to preserve the formation shape. It allows the UAVs to quickly turn back to the previous formation configuration.
- Shrinkage: The UAVs fly toward the formation centroid while maintaining the formation shape. This configuration is used in the case of being required to maintain the overlap among photos taken.

6.2.2 Reconfiguration with intermediate waypoints

In order to reconfigure, the UAVs need to re-route their flying paths through adjacent space and thus require intermediate waypoints (IWPs). To identify those waypoints, additional constraints are required as follows:

1. The distance between UAVs must be within the communication range but not smaller than two times the UAV radius:

$$d_{\text{com}} \geq d(P_m, P_n) \geq 2r_Q, \tag{6.1}$$

where d_{com} is the communication range, and r_Q is the safe radius of a UAV, and $d(P_m, P_n)$ is the distance between UAV_m and UAV_n.

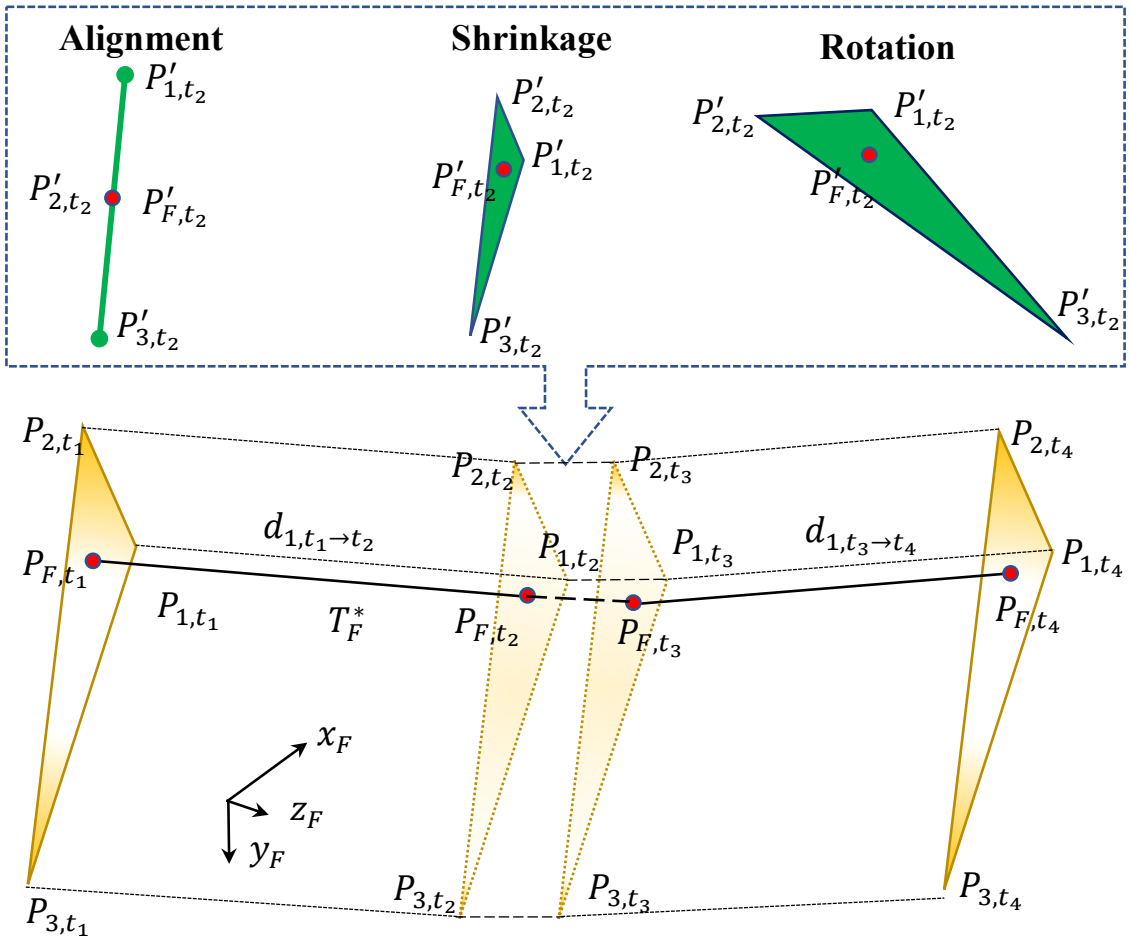


Figure 6.1 : Reconfigurable formation

2. The UAVs must fly within a certain distance to the surface being inspected:

$$d_n^s \in [d_{\min}^s, d_{\max}^s], \quad n \in \{1, 2, 3\}, \tag{6.2}$$

where d_n^s is the distance from UAV_n to the surface to be inspected, $d_{\min}^s$ and $d_{\max}^s$ are respectively the minimum and maximum distances from a UAV to the surface.

Assume that each obstacle in the working environment of UAVs is modelled as a cylinder with the center's coordinate C_k, radius r_k and height z_k. For any different obstacles p and $q, \forall p \neq q,\ p, q \in \{1, 2, \ldots, K\}$, one has their radii r_p, r_q, and centre coordinates $C_p(x_p, y_p)$, $C_q(x_q, y_q)$, respectively. Denoting P_p and P_q as intersection

points between the straight line created by C_p and C_q and the two circles (C_p, r_p) and (C_q, r_q), location of the jth IWP is then determined as the midpoint of P_pP_q:

$$C_j = \begin{cases} \frac{1}{2}(P_p + P_q), & \text{if } r_n^S \leq d_{p,q} < r_{l,k}^S \\ \varnothing & \text{otherwise} \end{cases}, \tag{6.3}$$

where $C_j = (x_j, y_j)$ is the coordinates of the jth IWP in the horizontal plane, $d_{p,q}$ is the smallest distance between the two adjacent obstacles, and r_n^S is the safe radius of UAVn. Finally, the cost function 6.4 in the last chapter needs to be updated to include the cost caused by the intermediate waypoints as follows:

$$J(T_{Fi}) = J_F(T_{Fi}) + J_R(T_{Fi}), \tag{6.4}$$

where $J_R(T_{Fi})$ represents the distance from intermediate waypoints to path segments:

$$J_R(T_{Fi}) = \frac{1}{L_i M_j} \sum_{l=1}^{L_i} \sum_{j=1}^{M_j} \sqrt{(x_l - x_j)^2 + (y_l - y_j)^2}, \tag{6.5}$$

where M_j is the total number of IWPs.

At each IWP, the formation is reconfigured by changing positions of the UAVs to a designated position in the new selected shape. The UAVs then come back to their original defined position after passing those IWPs. Hence, the changing shape can be divided into two phases, transformation and reconfiguration, conducted between time intervals $[t_1, t_2]$ and $[t_3, t_4]$ respectively as shown in Figure 6.1. The new shape is maintained between those phases, from t_2 to t_3, to keep the UAVs safe while travelling inside the narrow passage.

The next step is to find a set of positions, P_n for each UAV$_n$ such that the planned trajectory of the whole formation, represented by the formation centroid P_F, and the formation shape are preserved.

6.2.3 Reference trajectory generation

Given the optimal path, P_F^*, generated by the θ-PSO for the formation centroid in which the intermediate waypoints for reconfiguration have been included, specific paths for each UAV can be computed based on the formation model presented in Chapter 4.

For IWP$_j$, a set of new waypoints $P_F^j = [P_{F,t_1}, \ldots, P_{F,t_4}]$ for the formation is generated. Depending on the predefined position in the reconfiguration shape and P_F^j, a set of waypoints for UAV$_n$, $P_n^j = [P_{n,t_1}, \ldots, P_{n,t_4}]$, is also computed. Let ΔP_n be the set of desired difference in position between UAV$_n$ and the formation centroid at time t:

$$\Delta P_n = (P_n \cup P_n^j) - (P_F \cup P_F^j), \tag{6.6}$$

this difference is calculated in the inertial frame as:

$$\Delta P_n^O = R_{OF}^{-1}(t)\Delta P_n, \tag{6.7}$$

where R_{OF} is the rotation matrix. The flying path for each UAV is then given by:

$$P_n^* = P_F^* + \Delta P_n. \tag{6.8}$$

Finally, by combining the results of the path planning process and the reconfigurable algorithm, the completed set of trajectory commands for the nth UAV is determined as:

$$T_n = [P_n^*, V_n]^T, \tag{6.9}$$

where V_n is the velocity profile set for UAV$_n$. This command set will be fed to the internal controller of the UAV for trajectory tracking.

6.2.4 Algorithm implementation

The implementation of the reconfigurable formation algorithm can be described by the pseudo code in Figure 6.2. It starts with the initialisation of the inspection

surface, working space, obstacle positions, flight constraints and θ-PSO parameters. The θ-PSO is then executed based on the cost function (6.4) to generate an optimal path for the formation centroid. At each IWP, the chosen formation shape is the basis to compute the new set of positions for UAV$_n$ w.r.t their corresponding positions of the centroid. The distance error, $\Delta d_{n,t_1 \to t_2}$, is found by comparing between the travel distances, $d_{n,t_1 \to t_2}$ and $d'_{n,t_1 \to t_2}$, of the nominal and transformation shape, respectively. The ground velocity increment $\Delta V_{n,t_1 \to t_2}$ is found based on $\Delta d_{t_1 \to t_2}$ and the transformation time t_t computed from the planned path. A similar process is applied for the period of $[t_3, t_4]$.

Given the path planning, the reconfiguration process, the result paths will be loaded into the onboard controller of the UAVs. Here, the low-level control system on the UAV will execute to track the assigned trajectory. Finally, the entire system architecture of three layers: task assessment, high-level control and low-level control in Figure 4.1 can be represented in the flowchart in Figure 6.3.

It is noted that only static and known position obstacles are considered in this work, so the algorithm for reconfiguration is appropriate for offline applications. External dynamic obstacles can be avoided by using additional sensors such as ultrasonic sensors or Lidars with an extended path planning module. This topic is however beyond the scope of this work.

6.3 Experiments

A number of experiments have been conducted to evaluate the validity and efficiency of the proposed algorithm. The setup and results are presented below.

6.3.1 Experimental setup

The task assigned in experiments is to inspect surfaces of a light rail bridge using three UAVs. The UAVs used are the 3DR Solo drones retrofitted with inspection

```
   /* Preparation:                                                              */
 1   Determine the inspection surface(s);
 2   Identify boundaries of the working space;
 3   Identify obstacle set K;
 4   Group all the above data and save in a common file (init file);
   /* Initialisation:                                                          */
 5   Initialise the working environment by loading the init file to global memory;
 6   Initialise constraints, i.e., r_Q, d_com, d^s_min, d^s_max;
 7   Determine locations of IWPs;                    /* using (6.1), (6.2), and (6.3) */
 8   Initialise θ-PSO parameters;
 9   Generate a random path to connect the start and target waypoints;
10   Set θ_ij ∈ [−π/2, π/2] and Δθ_ij ∈ [−π/2, π/2];
   /* Path Planning:                                                           */
11 foreach i <(swarm_iteration) do
12     foreach j < (swarm_population) do
13         Compute new value of Δθ_ij;              /* using 1st equation in (5.1) */
14         Compute new value of θ_ij;               /* using 2nd equation in (5.1) */
15         Compute new position;                    /* using 3rd equation in (5.1) */
16         Check Violation cost;                                  /* using (5.14) */
17         Evaluate each path based on the Best_Costs and Violation cost;
18         Update each particle_personal_best and the global_best positions;
19     end
20     Update global_best and Violation costs;
21 end
22   Save global_best and Violation cost;
23   Final path is chosen as the maximum number of iterations is reached.
24   Generate the individual path for UAV_n.
   /* Path generation:                                                         */
25 foreach UAV_n do
26     foreach j = 1 to M_j do
27         Select a relevant formation shape;
28         Compute the new position set P^j_n;
29         Compute the mission time t_t and t_r;
30         Determine Δd_{n,t1→t2} and Δd_{n,t3→t4};
31         Compute ΔV_{n,t1→t2} and ΔV_{n,t3→t4};
32         P*_n ← P*_F, ΔP_n;                       /* using (6.6)-(6.8) */
33         V_n ← ΔV_{n,t1→t2}, ΔV_{n,t3→t4};
34     end
35     T_n ← P*_n, V_n.                                        /* using (6.9) */.
36 end
```

Figure 6.2 : Pseudo code for reconfigurable trajectory generation process

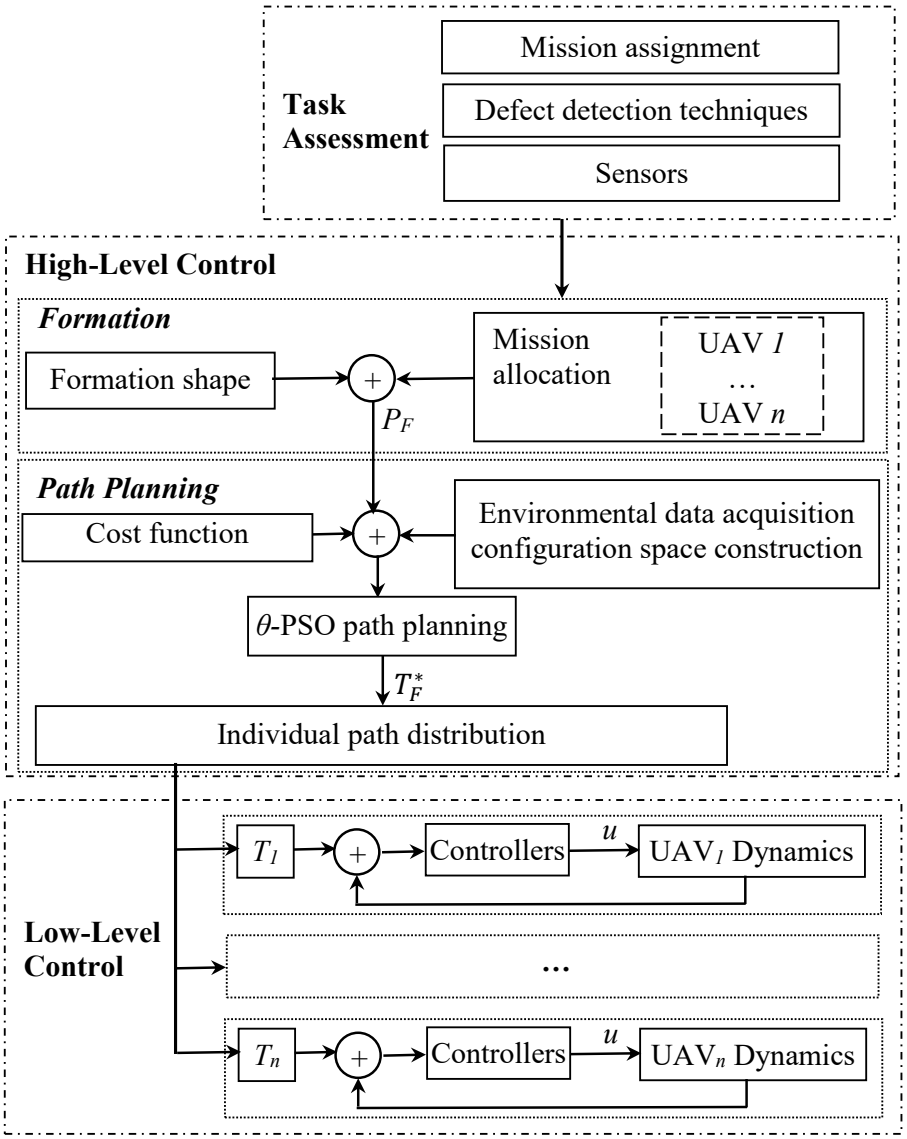


Figure 6.3 : Flowchart of the entire system framework

cameras and communication boards [73]. The operation space is chosen in the rectangular area with two opposite corners at GST coordinates of $\{-33.87601, 151.191182\}$ and $\{-33.875086, 151.192676\}$. Therein, actual obstacles are identified and the mission of UAVs is to inspect the surface represented by their poles numbered from (1) to (12). These obstacles include a pole (2), a light pole (4), bridge piers (1, 3, 5, 6, 7), power poles (8, 10), and a tree (9).

The initial configuration for inspection was a triangle formation with initial po-

sitions of the UAVs relative to the formation centroid to be $\Delta T_1 = [0, 2, 0]$ m, $\Delta T_2 = [-2, -1, 0]$ m and $\Delta T_3 = [2, -1, 0]$ m. The minimum and maximum clearances between UAVs and the terrain are $z_{\max} = 15$ m and $z_{\min} = 7$ m, respectively. The UAVs are required for flight within the relative distance to the inspected structure as $d_n^s \in [1, 5]$ m. The formation is set to flight at a constant ground velocity of 3 m/s. For θ-PSO, the number of particles, waypoints, and iterations are respectively selected as 100, 7, and 150.

6.3.2 Results

The evaluation is conducted in three reconfiguration shapes where the formation needs to change its configuration to keep safe while fulfilling the inspection task. The reference points are chosen to coincide with the centroid of the triangle created by the three UAVs. In experiments, it is planned that the designed formation shape starts to reconfigure at waypoint 11 and fully transforms to the new shape at about 1 m before waypoint 13. The new shape would be preserved until 1 m after the waypoint 13 and then the reconfiguration process would be completed at waypoint 14, which is illustrated in the right image of Figure 6.5. Figure 6.4 shows pictures that were captured from the field test of the formation transformation from the original horizontal triangle shape (Figure 6.4.a) to the alignment (Figure 6.4.b) and the vertical triangle (Figure 6.4.c) ones.

In the experiment with the alignment reconfiguration, Figure 6.6 shows the capability of the formation in traversing the narrow corridor in which space is just enough for a single UAV to pass through. It shows clearly in the figure that the reconfiguration is completed to allow the UAVs to go through the narrow passage between obstacles 4 and 5 without any contact.

In the rotation transformation, the UAV_1 kept following its planned path while the two others changed their flight heights to reach their new positions in the vertical

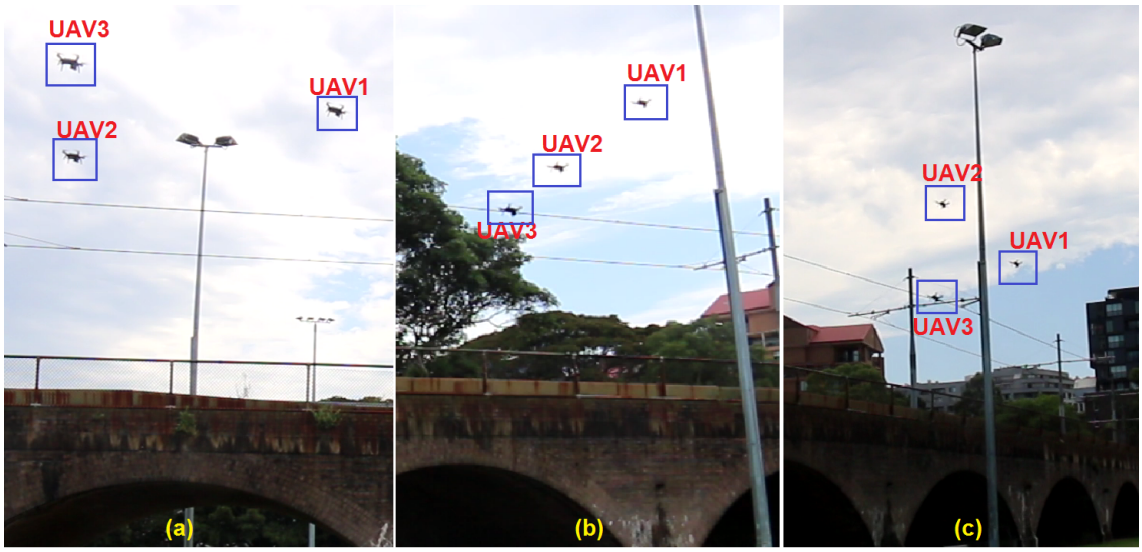


Figure 6.4 : Transformation of triangular formation (a) to alignment (b), and rotation (c) configurations

plane as shown in Figure 6.7. Specifically, changes in the altitude happened at time $t_1 = 20$ s to reconfigure the UAVs fully to the new shape at time $t_2 = 26$ s. The new shape is then preserved until $t_3 = 27$ s and finally converted back to its original at $t_4 = 35$ s. The velocity profiles of the UAVs shown in Figure 6.8 imply the relatively stable movement of UAVs during this experiment.

Figure 6.9 shows the result of the shrink reconfiguration. The UAVs start to change their altitudes when encountering obstacle 3 and shrink the triangular formation to pass through obstacle 9. This result proves the capability of the proposed algorithm in handling situations where the formation shape needs to be maintained, but its size needs adjusting to avoid collisions.

On the other hand, it is also noted that the reconfiguration in experiments was conducted by using offline satellite maps. While this approach is relevant for most static civil infrastructure, occasionally unexpected dynamic obstacles not included in the calculation may cause safety concerns. The problem can be overcome by incorporating real-time data acquired by sensors installed on UAVs.

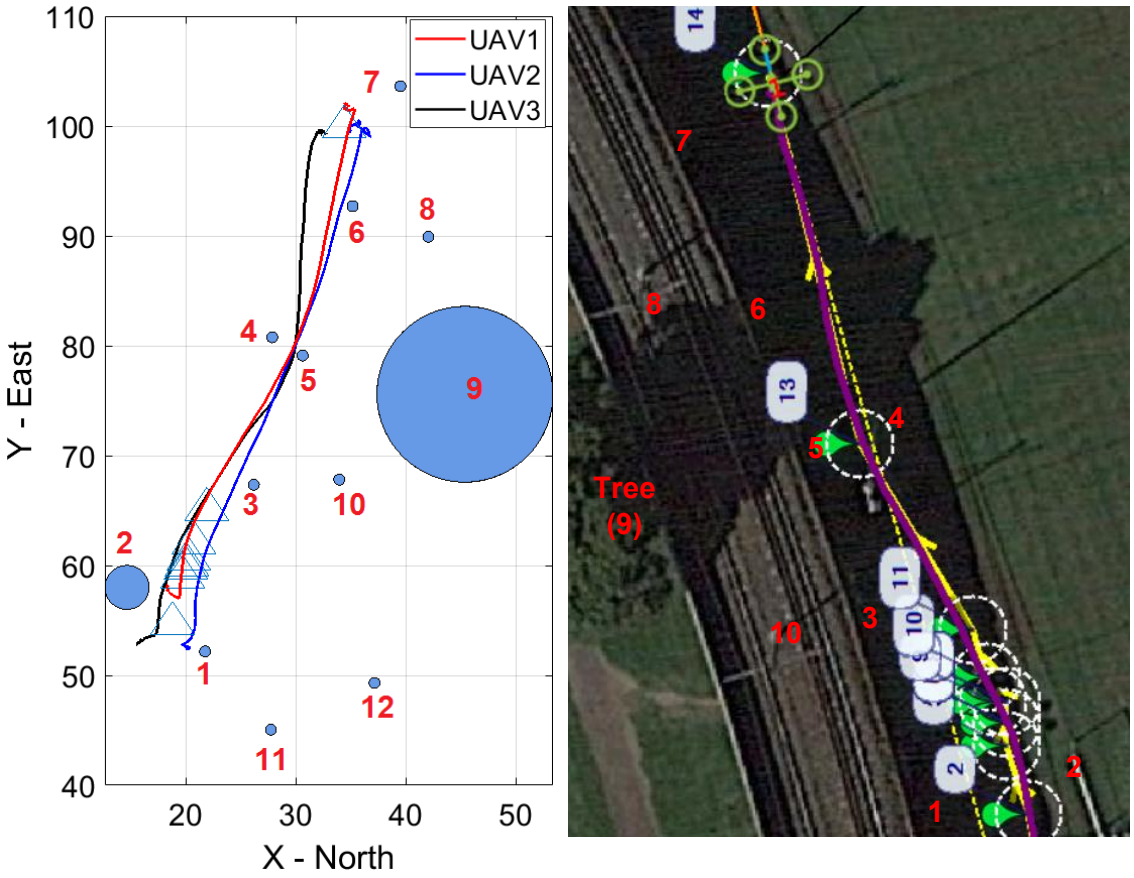


Figure 6.5 : UAVs' trajectories in horizontal plane in the alignment formation

6.4 Conclusion

A path planning algorithm for multi-UAV formation has been proposed in this chapter, in which its shape can vary in accordance with the operating environment. The core of this algorithm is the derivation of a cost function that takes into account the constraints on collision avoidance, flight altitude, communication range, and visual inspection requirements. Based on this cost function, the θ-PSO has been used to generate the path for the formation which is then used to determine trajectories for individual UAVs. Also in this work, the use of intermediate waypoints is proposed for reconfiguration which can be accomplished in the alignment, rotation, or shrink fashion. A number of experiments have been conducted to evaluate the performance

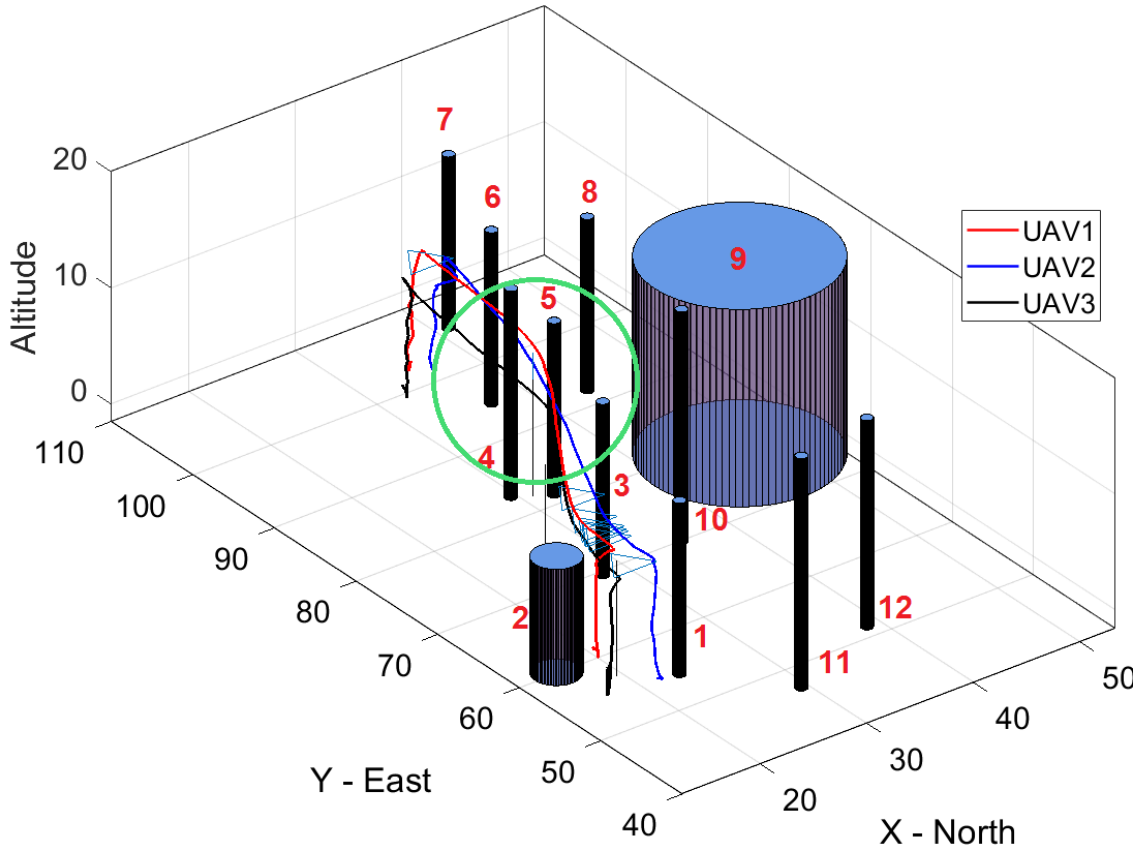


Figure 6.6 : 3D real-time plot for alignment formation

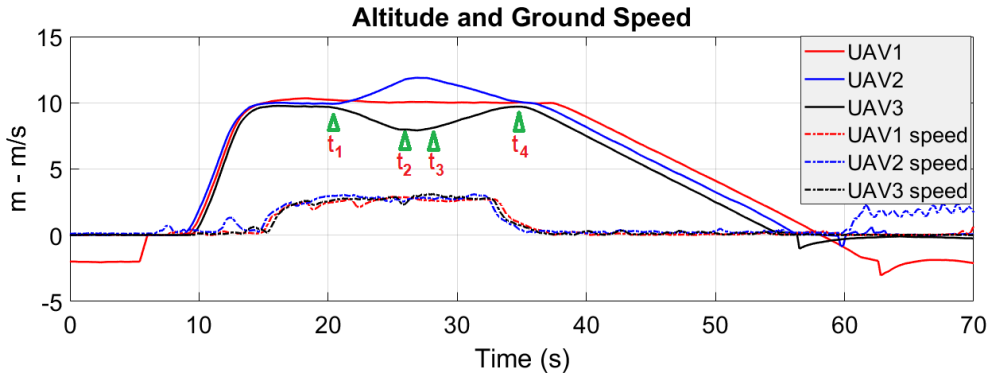


Figure 6.7 : Altitudes and ground speeds of UAVs during the experiment with rotating reconfiguration

of the proposed algorithm for inspection tasks.

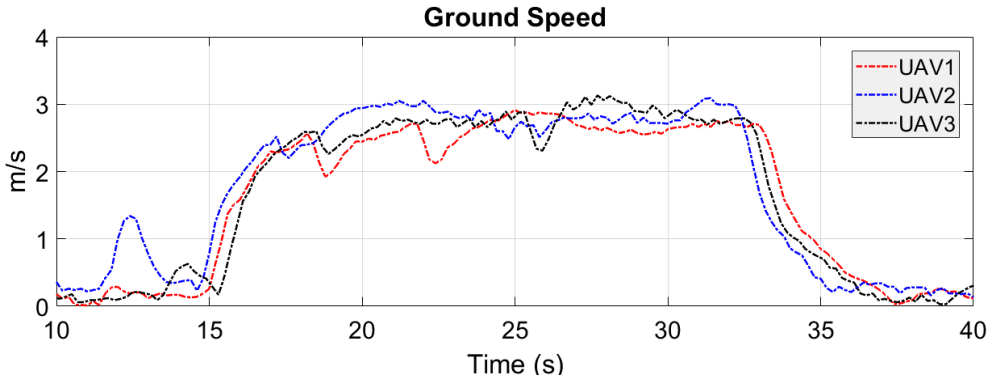


Figure 6.8 : Ground speed of UAVs in rotating reconfiguration

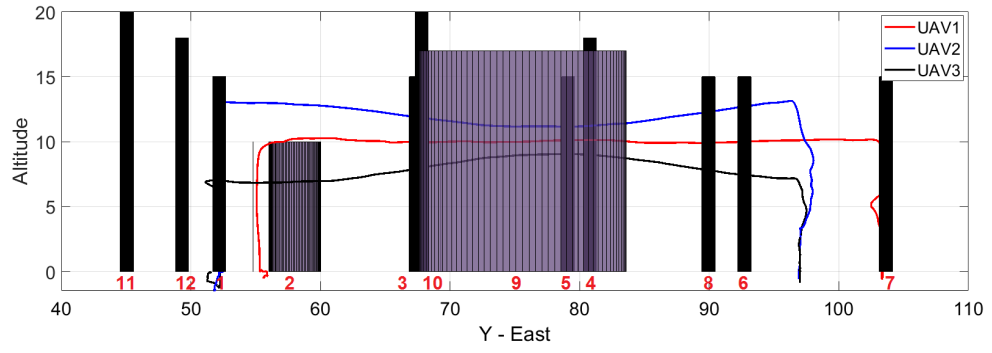


Figure 6.9 : Trajectories of UAVs in the experiment with shrink reconfiguration

Chapter 7

Conclusion

This chapter summarises the work of this dissertation and proposes potential development in the future. The chapter is organised as follows. Section 7.1 summarises the contents of the previous chapters. The contributions of the dissertation are reviewed in Section 7.2 and some future research problems are recommended in Section 7.3.

7.1 Thesis summary

Chapter 1 introduces the issues raised in Unmanned Aerial Vehicles research in general and multiple UAV formations in particular. The motivation here is the demand for the cooperation of multiple flying agents moving in desired shapes to perform the inspection of built infrastructures task that a single UAV is not able to accomplish in isolation. The chapter introduced the classification of UAVs. The objectives and organisation of this dissertation were also covered in the chapter.

Together with the literature review of the low-level control approaches for UAV, an extensive survey of robotic motion planning and control methods available for analysis and design of multi-UAV formation is given in Chapter 2.

Chapter 3 deals with the low-level control problems for quadcopter UAVs. Adaptive second-order sliding mode control (2-SMC) laws have been designed for the system in the presence of disturbances, variations, and coupling effects. Two sliding approaches have been utilised, named the quasi-continuous and twisting techniques. These 2-SMC laws are developed with their corresponding adaptive schemes to drive

the attitude state variables of the UAV towards the sliding surface in finite time. The control designs have been analysed by using global Lyapunov functions for convergence of both the sliding dynamics and adaptation scheme. Extensive simulation results are provided to illustrate the feasibility of the proposed design approach in some practical scenarios.

A multi-UAV framework which includes robust control, path planning and formation preservation is proposed in Chapter 4. It starts with an overview of the system architecture. The chapter then covers background materials required for the design and analysis of the entire system, including fundamental knowledge of path planning and formations involving multiple UAVs. At the end, the chapter provides brief descriptions of the quadcopter UAV and hardware development for experiments throughout this project.

A path planning algorithm for multi-UAV formation conducting inspection tasks of built infrastructure is proposed in Chapter 5. The approach begins with an explanation of a particle swarm optimisation algorithm using angle-coded PSO (θ-PSO). Theoretical analysis of the conventional PSO and the θ-PSO are reviewed to provide a rationale for the choice of θ-PSO in this work. The chapter then presents a function containing proper constraints to improve collision avoidance capability and task efficiency. The proposed path planning method ends with algorithms of path planning implementation for the formation centroid and path generation for each UAV. Extensive simulation, comparison, and experiments have been conducted for assessment. The results illustrate the validity and effectiveness of the proposed path planning algorithm not only for the centroid of the formation but also the path creation for UAVs.

Chapter 6 concerns multiple-UAV reconfigurable formation problems. In this chapter, the θ-PSO path planning, obtained by using the algorithm in Chapter

5, is proposed to combine with the reconfigurable formation. The chapter begins with the introduction of UAV formation topologies, followed by the use of a 3D model of the inspected surface and its surrounding environment to determine a set of intermediate waypoints. A formation shape is then defined at these WPs so that multiple UAVs are feasible to operate. New constraints are also introduced to improve collision avoidance capability and task efficiency. Based on that, an optimal path for the centroid of the formation is produced. The methodology is completed by the achievement of trajectories for UAVs by integrating the generated path with the selected reconfiguration shapes. Experiments have been conducted with the results demonstrating the validity and effectiveness of the proposed algorithm.

7.2 Thesis contributions

According to the literature survey, and the best knowledge of the candidate, the study on this thesis is relevant to current research topics and the trend in future studies on UAVs. This study also enables future improvements for various applications, such as, infrastructure inspection and health monitoring, aerial photography, and disaster control and monitoring. These listed applications are extracted from the literature survey for UAVs. More importantly, experimental results vigorously evaluate the advancement in this research since these real cases are scarce in the literature which is full of simulations. Towards building a versatile and feasible system architecture for multiple UAV formation in robotic inspection tasks, these contributions have been made in this thesis:

7.2.1 Robust low-level control of the quadcopter drive

As quadcopter drones are subject to nonlinearity, strong coupling, uncertain dynamics and external disturbances, a great deal of control effort has been placed on robust controllers for which the sliding mode control (SMC) methodology is most

popular [159,44,217,207]. This work focused on (i) alleviating chattering associated with SMC, (ii) adjusting the control gains to improve robustness, and (iii) addressing feasibility to support control implementation on real-time operations of the UAVs. For this, adaptive quasi-continuous and twisting sliding mode control laws have been developed in this thesis. The proposed adaptation schemes offer new algorithms for changing the control parameters for performance improvements while retaining simplicity of the chattering-free second-order sliding mode controllers.

7.2.2 A prototype of system architecture for multi-UAV formation control

Deploying a formation of multiple UAVs for surface inspection requires the synchronisation in control, path planning and data processing not only between the control levels within the system and between each UAV. Therefore, a suitable communication is essential. Approaches using a private network among on-site devices [183] or communication relay stations to extend the network [91, 28] often cover a small range but not large areas. The proposed system architecture features (i) integration of various modules to fulfil an inspection task using a new communication platform based on the Internet of Things, (ii) compatible hardware extension on each Solo 3DR drone to guarantee the safe flight and task performance of each UAV as well as the whole formation, and (iii) improvement in remote processing capabilities while reducing computational latency in real-time operations.

7.2.3 Multi-UAV path planning for formation control

While path planning algorithms such as A-star, Dijkstra, rapidly-exploring random tree and probabilistic roadmap are available (see. eg. [204, 52, 5, 20]), the requirements for generation of collision-free trajectories for the group of UAVs flying in a desired shape can be rendered as a multi-objective optimisation problem. To this end, particle swarm optimisation [156] or iterative viewpoint resampling [19]

are typically more efficient by considering also the coverage among waypoints. In this work, to speed up the convergence of the optimisation process for the whole formation, new angle-coded PSO algorithms have been proposed. The advantages include (i) realistic execution of the planning algorithm, (ii) the generation of optimised paths with collision avoidance simultaneously, (iii) feasibility of facilitating coordination with regards to an inspecting task, and (iv) capability of scanning in the formation flight.

7.2.4 Reconfigurable formation of multiple UAVs

UAV formation reconfiguration is required to accommodate a complicated inspection task or a difficult environment. The requirement involves defining a set of motional parameters and a relevant process to establish the new configuration [197,196]. For this, among available techniques such as semi-analytic approach [173], PSO [190] or hybrid PSO [197, 51], nonlinear programming [120] and hierarchical evolutionary [198] trajectory planning, this work aimed to augment the proposed optimisation algorithm for path planning with additional constraints based on intermediate waypoints to satisfy the requirements for safety and task efficiency given a number of formation types. The new reconfigurable formation path planning technique can therefore (i) guarantee safe flights of the UAV group, and (ii) improve the formation flexibility inspection as well as the inspection capacity.

7.3 Discussion and future research

7.3.1 Discussion

While the inspection system has been implemented successfully, it is worth discussing problems during the development for further investigation. The first issue relates to the communication between UAVs and the gateway router. As most current IoT boards are designed for indoor applications, their built-in antennas are

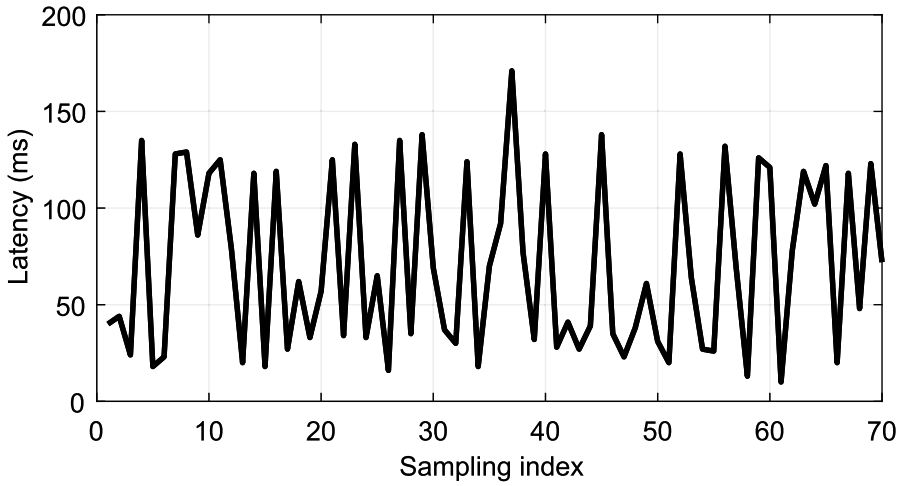


Figure 7.1 : Network delay during the inspection

insufficient for outdoor communication in inspection tasks. A high gain external antenna is therefore needed to extend the communication range as well as provide stable signals, especially for moving objects like UAVs. Nevertheless, the communication range between UAVs and the gateway router is still limited to approximately 100 m due to the transmitting power of the IoT board. While this range is sufficient for inspecting structures like buildings or wind turbines, it may be ill-use for long structures like bridges. A possible solution is to use more gateway routers as repeaters.

Another issue relates to the network latency. It is interesting that the average transmission delay in our experiments is 71 ms, which is even larger than the processing time to detect defects. Moreover, the network jitter is also quite large with the standard deviation of 45 ms as shown in Figure 7.1. In the worst case, the delay together with the processing time limit the real-time capability of the inspection system to 4 frames per second. The rationale is the use of mobile broadband networks which require wireless data transmission via multiple intermediate relay stations.

7.3.2 Future work

Potential future works for the Control architecture and path planning for quadcopters in formation topic can be explored in two main aspects, namely, the online path planning and reconfiguration, and a generic reconfigurable architecture for robotic formation control.

Online path planning and reconfiguration

The proposed framework for planning and control is for workspaces with known and stationary obstacles and their maps are assumed to be available. When the formation is deployed in an unknown or partially known workspace with moving obstacles, simultaneous localisation and mapping (SLAM), and online planing and replanning are essential.

On the other hand, it is also noted that the reconfiguration in experiments was conducted by using offline satellite maps. While this approach is relevant for most static civil infrastructure, occasionally unexpected dynamic obstacles not included in the calculation may cause safety concerns. The problem can be overcome by incorporating real-time data acquired by sensors installed on UAVs. In this case, each UAV needs to have capabilities to do online path planning as well as communicate with other UAVs to determine its role in the new formation. Another approach is to pass all the sensing information to the RCU for calculating new paths and re-upload them to the UAVs. In either approach, extra complication is added to the system which may pose further issues. The abovementioned problems are suggested as open issues for further research.

A generic reconfigurable architecture for robotic formation control

If UAVs and their systems had grown in the last decades, their path planning and formation problems have been expanding in both research and real-world appli-

cations. Their autonomous capabilities and technology readiness require them to be systematically and comprehensively incorporated in a unique system architecture.

The results obtained from this thesis not only confirm its validity in infrastructure inspection but also practically suggest a possibility of extending the work toward a generic reconfigurable architecture for robotic formation control in complex environments. Furthermore, developing the work in this dissertation towards a generic and practical framework remains a challenging topic for future investigation.

Bibliography

[1] M. F. Abdullah, I. Wijayanto, and A. Rusdinar, "Position estimation and fire detection based on digital video color space for autonomous quadcopter using odroid xu4," in Control, Electronics, Renewable Energy and Communications (ICCEREC), 2016 International Conference on. IEEE, 2016, pp. 169–173, doi:10.1109/ICCEREC.2016.7814976.

[2] E. U. Acar, H. Choset, and P. N. Atkar, "Complete sensor-based coverage with extended-range detectors: A hierarchical decomposition in terms of critical points and voronoi diagrams," in Intelligent Robots and Systems, 2001. Proceedings. 2001 IEEE/RSJ International Conference on, vol. 3. IEEE, 2001, pp. 1305–1311, doi:10.1109/IROS.2001.977163.

[3] Z. Ahmad, F. Ullah, C. Tran, and S. Lee, "Efficient energy flight path planning algorithm using 3-d visibility roadmap for small unmanned aerial vehicle," International Journal of Aerospace Engineering, vol. 2017, 2017, doi:10.1155/2017/2849745.

[4] G. Alizadeh and K. Ghasemi, "Control of quadrotor using sliding mode disturbance observer and nonlinear h_∞," International Journal of Robotics, vol. 4, no. 1, pp. 38–46, 2015.

[5] A. Ammar, H. Bennaceur, I. Châari, A. Koubâa, and M. Alajlan, "Relaxed dijkstra and a* with linear complexity for robot path planning problems in large-scale grid environments," Soft Computing, vol. 20, no. 10, pp. 4149–4171, Oct 2016, doi:10.1007/s00500-015-1750-1.

[6] O. Andersson, M. Wzorek, and P. Doherty, "Deep learning quadcopter control via risk-aware active learning," in AAAI, 2017, pp. 3812–3818.

[7] M. Aranda, G. López-Nicolás, C. Sagüés, and Y. Mezouar, "Formation control of mobile robots using multiple aerial cameras," IEEE Transactions on Robotics, vol. 31, no. 4, pp. 1064–1071, 2015, doi:10.1109/TRO.2015.2452777.

[8] L. M. Argentim, W. C. Rezende, P. E. Santos, and R. A. Aguiar, "Pid, lqr and lqr-pid on a quadcopter platform," in Informatics, Electronics & Vision (ICIEV), 2013 International Conference on. IEEE, 2013, pp. 1–6, doi:.

[9] V. I. Arnol'd, Mathematical methods of classical mechanics. Springer, 1989.

[10] A.-R. Babaei, M. Malekzadeh, and D. Madhkhan, "Adaptive super-twisting sliding mode control of 6-dof nonlinear and uncertain air vehicle," Aerospace Science and Technology, vol. 84, pp. 361 – 374, 2019, doi:10.1016/j.ast.2018.09.013.

[11] V. Baiocchi, D. Dominici, and M. Mormile, "Uav application in post-seismic environment," Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci., XL-1 W, vol. 2, pp. 21–25, 2013, doi:10.5194/isprsarchives-XL-1-W2-21-2013.

[12] T. Balch and R. C. Arkin, "Behavior-based formation control for multirobot teams," IEEE Transactions on robotics and automation, vol. 14, no. 6, pp. 926–939, 1998, doi:10.1109/70.736776.

[13] A. Banaszek, S. Banaszek, and A. Cellmer, "Possibilities of use of UAVS for technical inspection of buildings and constructions," IOP Conference Series: Earth and Environmental Science, vol. 95, p. 032001, dec 2017, doi:10.1088/1755-1315/95/3/032001.

[14] I. Bayezit and B. Fidan, "Distributed cohesive motion control of flight vehicle

formations," IEEE Transactions on Industrial Electronics, vol. 60, no. 12, pp. 5763–5772, 2013, doi:10.1109/TIE.2012.2235391.

[15] R. W. Beard, J. Lawton, and F. Y. Hadaegh, "A coordination architecture for spacecraft formation control," IEEE Transactions on control systems technology, vol. 9, no. 6, pp. 777–790, 2001, doi:10.1109/87.960341.

[16] L. Besnard, Y. B. Shtessel, and B. Landrum, "Quadrotor vehicle control via sliding mode controller driven by sliding mode disturbance observer," Journal of the Franklin Institute, vol. 349, no. 2, pp. 658–684, 2012, doi:10.1016/j.jfranklin.2011.06.031.

[17] B. J. Bialy, J. Klotz, K. Brink, and W. E. Dixon, "Lyapunov-based robust adaptive control of a quadrotor uav in the presence of modeling uncertainties," in 2013 American Control Conference, June 2013, pp. 13–18, doi:10.1109/ACC.2013.6579806.

[18] Z. Bingül and O. Karahan, "A fuzzy logic controller tuned with pso for 2 dof robot trajectory control," Expert Systems with Applications, vol. 38, no. 1, pp. 1017–1031, 2011, doi:10.1016/j.eswa.2010.07.131.

[19] A. Bircher, K. Alexis, M. Burri, P. Oettershagen, S. Omari, T. Mantel, and R. Siegwart, "Structural inspection path planning via iterative viewpoint resampling with application to aerial robotics," in Robotics and Automation (ICRA), 2015 IEEE International Conference on. IEEE, 2015, pp. 6423–6430, doi:10.1109/ICRA.2015.7140101.

[20] A. Bircher, K. Alexis, U. Schwesinger, S. Omari, M. Burri, and R. Siegwart, "An incremental sampling-based approach to inspection planning: the rapidly exploring random tree of trees," Robotica, vol. 35, no. 6, p. 13271340, 2017, doi:10.1017/S0263574716000084.

[21] H. Bolandi, M. Rezaei, R. Mohsenipour, H. Nemati, and S. M. Smailzadeh, "Attitude control of a quadrotor with optimized pid controller," Intelligent Control and Automation, vol. 4, no. 03, p. 335, 2013, doi:10.4236/ica.2013.43039.

[22] H. Boudjedir, F. Yacef, O. Bouhali, and N. Rizoug, "Adaptive neural network for a quadrotor unmanned aerial vehicle," International Journal in Foundations of Computer Science and Technology, vol. 2, no. 4, pp. 1–13, 2012.

[23] P. Bouffard, A. Aswani, and C. Tomlin, "Learning-based model predictive control on a quadrotor: Onboard implementation and experimental results," in Robotics and Automation (ICRA), 2012 IEEE International Conference on. IEEE, 2012, pp. 279–284, doi:10.1109/ICRA.2012.6225035.

[24] O. Bouhali and H. Boudjedir, "Neural network control with neuro-sliding mode observer applied to quadrotor helicopter," in Innovations in Intelligent Systems and Applications (INISTA), 2011 International Symposium on. IEEE, 2011, pp. 24–28, doi:10.1109/INISTA.2011.5946063.

[25] G. Cai, J. Dias, and L. Seneviratne, "A survey of small-scale unmanned aerial vehicles: Recent advances and future development trends," Unmanned Systems, vol. 2, no. 02, pp. 175–199, 2014, doi:10.1142/S2301385014300017.

[26] S. Caselli, M. Reggiani, and R. Rocchi, "Heuristic methods for randomized path planning in potential fields," in Computational Intelligence in Robotics and Automation, 2001. Proceedings 2001 IEEE International Symposium on. IEEE, 2001, pp. 426–431, doi:10.1109/CIRA.2001.1013238.

[27] O. Cetin, S. Kurnaz, O. Kaynak, and H. Temeltas, "Potential field-based navigation task for autonomous flight control of unmanned aerial vehicles,"

International Journal of Automation and Control, vol. 5, no. 1, pp. 1–21, 2010, doi:10.1504/IJAAC.2011.037377.

[28] O. Cetin and I. Zagli, "Continuous airborne communication relay approach using unmanned aerial vehicles," Journal of Intelligent & Robotic Systems, vol. 65, no. 1, pp. 549–562, Jan 2012, doi:10.1007/s10846-011-9556-6.

[29] B. Chen, L. Deng, Y. Duan, S. Huang, and J. Zhou, "Building change detection based on 3d reconstruction," in Image Processing (ICIP), 2015 IEEE International Conference on. IEEE, 2015, pp. 4126–4130, doi:10.1109/ICIP.2015.7351582.

[30] F. Chen, R. Jiang, K. Zhang, B. Jiang, and G. Tao, "Robust backstepping sliding-mode control and observer-based fault estimation for a quadrotor uav," IEEE Transactions on Industrial Electronics, vol. 63, no. 8, pp. 5044–5056, Aug 2016, doi:10.1109/TIE.2016.2552151.

[31] J.-H. Chen, M.-C. Su, R. Cao, S.-C. Hsu, and J.-C. Lu, "A self organizing map optimization based image recognition and processing model for bridge crack inspection," Automation in Construction, vol. 73, pp. 58–66, 2017, doi:10.1016/j.autcon.2016.08.033.

[32] Y.-b. Chen, G.-c. Luo, Y.-s. Mei, J.-q. Yu, and X.-l. Su, "Uav path planning using artificial potential field method updated by optimal control theory," International Journal of Systems Science, vol. 47, no. 6, pp. 1407–1420, 2016, doi:10.1080/00207721.2014.929191.

[33] Y. Chen, J. Yu, X. Su, and G. Luo, "Path planning for multi-uav formation," Journal of Intelligent & Robotic Systems, vol. 77, no. 1, p. 229, 2015, doi:10.1007/s10846-014-0077-y.

[34] S. S. Choi and E. K. Kim, "Design and implementation of vision-based structural safety inspection system using small unmanned aircraft," in Advanced Communication Technology (ICACT), 2015 17th International Conference on. IEEE, 2015, pp. 562–567, doi:10.1109/ICACT.2015.7224924.

[35] H. Choset and J. Burdick, "Sensor-based exploration: The hierarchical generalized voronoi graph," The International Journal of Robotics Research, vol. 19, no. 2, pp. 96–125, 2000, doi:10.1177/02783640022066770.

[36] I. D. Cowling, O. A. Yakimenko, J. F. Whidborne, and A. K. Cooke, "A prototype of an autonomous controller for a quadrotor uav," in Control Conference (ECC), 2007 European. IEEE, 2007, pp. 4001–4008, doi:10.23919/ECC.2007.7068316.

[37] B. G. Czakó and K. Kósi, "Novel method for quadcopter controlling using nonlinear adaptive control based on robust fixed point transformation phenomena," in Applied Machine Intelligence and Informatics (SAMI), 2017 IEEE 15th International Symposium on. IEEE, 2017, pp. 000 289–000 294, doi:10.1109/SAMI.2017.7880320.

[38] K. Daniel, A. Nash, S. Koenig, and A. Felner, "Theta*: Any-angle path planning on grids," Journal of Artificial Intelligence Research, vol. 39, pp. 533–579, 2010, doi:10.1613/jair.2994.

[39] A. Das, T. T. Leong, S. Suresh, and N. Sundararajan, "Meta-cognitive interval type-2 fuzzy controller for quadcopter flight control-an eeg based approach," in Fuzzy Systems (FUZZ-IEEE), 2016 IEEE International Conference on. IEEE, 2016, pp. 2501–2507, doi:10.1109/FUZZ-IEEE.2016.7738008.

[40] L. De Filippis, G. Guglieri, and F. Quagliotti, "Path planning strategies for uavs in 3d environments," Journal of Intelligent & Robotic Systems, vol. 65,

no. 1, pp. 247–264, 2012, doi:10.1007/s10846-011-9568-2.

[41] P. De Monte and B. Lohmann, "Position trajectory tracking of a quadrotor helicopter based on l1 adaptive control," in Control Conference (ECC), 2013 European. IEEE, 2013, pp. 3346–3353, doi:10.23919/ECC.2013.6669410.

[42] C. Deng, S. Wang, Z. Huang, Z. Tan, and J. Liu, "Unmanned aerial vehicles for power line inspection: A cooperative way in platforms and communications," Journal of Communications, vol. 9, no. 9, pp. 687–692, 2014, doi:10.12720/jcm.9.9.687-692.

[43] K. DeoBodha, V. Mukherjee, V. KumarYadav, K. Saurabh, and S. Anium, "A levy flight based voltage particle swarm optimization for multiple-objective mixed cost-effective emission dispatch," in 2018 8th International Conference on Cloud Computing, Data Science Engineering (Confluence), Jan 2018, pp. 1–6, doi:10.1109/CONFLUENCE.2018.8442919.

[44] L. Derafa, A. Benallegue, and L. Fridman, "Super twisting control algorithm for the attitude tracking of a four rotors uav," Journal of the Franklin Institute, vol. 349, no. 2, pp. 685–699, 2012, doi:10.1016/j.jfranklin.2011.10.011.

[45] C. Diao, B. Xian, Q. Yin, W. Zeng, H. Li, and Y. Yang, "A nonlinear adaptive control approach for quadrotor uavs," in Control Conference (ASCC), 2011 8th Asian. IEEE, 2011, pp. 223–228.

[46] S. Ding, A. Levant, and S. Li, "New families of high-order sliding-mode controllers," in 2015 54th IEEE Conference on Decision and Control (CDC). IEEE, 2015, pp. 4752–4757, doi:10.1109/CDC.2015.7402960.

[47] K. D. Do, "Bounded controllers for formation stabilization of mobile agents with limited sensing ranges," IEEE Transactions on Automatic Control, vol. 52, no. 3, pp. 569–576, 2007, doi:10.1109/TAC.2007.892382.

[48] X. Dong, B. Yu, Z. Shi, and Y. Zhong, "Time-varying formation control for unmanned aerial vehicles: Theories and applications," IEEE Transactions on Control Systems Technology, vol. 23, no. 1, pp. 340–348, 2015, doi:10.1109/TCST.2014.2314460.

[49] X. Dong, Y. Zhou, Z. Ren, and Y. Zhong, "Time-varying formation tracking for second-order multi-agent systems subjected to switching topologies with application to quadrotor formation flying," IEEE Transactions on Industrial Electronics, vol. 64, no. 6, pp. 5014–5024, 2017, doi:10.1109/TIE.2016.2593656.

[50] H. Duan, P. Li, Y. Shi, X. Zhang, and C. Sun, "Interactive learning environment for bio-inspired optimization algorithms for uav path planning," IEEE Transactions on Education, vol. 58, no. 4, pp. 276–281, 2015, doi:10.1109/TE.2015.2402196.

[51] H. Duan, Q. Luo, Y. Shi, and G. Ma, "Hybrid particle swarm optimization and genetic algorithm for multi-uav formation reconfiguration," IEEE Computational intelligence magazine, vol. 8, no. 3, pp. 16–27, 2013, doi:10.1109/MCI.2013.2264577.

[52] F. Duchoň, A. Babinec, M. Kajan, P. Beňo, M. Florek, T. Fico, and L. Jurišica, "Path planning with modified a star algorithm for a mobile robot," Procedia Engineering, vol. 96, pp. 59–69, 2014, doi:10.1016/j.proeng.2014.12.098.

[53] V. Duggal, M. Sukhwani, K. Bipin, G. S. Reddy, and K. M. Krishna, "Plantation monitoring and yield estimation using autonomous quadcopter for precision agriculture," in Robotics and Automation (ICRA), 2016 IEEE International Conference on. IEEE, 2016, pp. 5121–5127, doi:10.1109/ICRA.2016.7487716.

[54] Y. Dvir and A. Levant, "Accelerated twisting algorithm," IEEE Transactions on Automatic Control, vol. 60, no. 10, pp. 2803–2807, 2015, doi:10.1109/TAC.2015.2398880.

[55] Z. T. Dydek, A. M. Annaswamy, and E. Lavretsky, "Adaptive control of quadrotor uavs: A design trade study with flight evaluations," IEEE Transactions on Control Systems Technology, vol. 21, no. 4, pp. 1400–1406, July 2013, doi:10.1109/TCST.2012.2200104.

[56] M. Ö. Efe, "Neural network assisted computationally simple pi$^{\lambda}$d$^{\mu}$ control of a quadrotor uav," IEEE Transactions on Industrial Informatics, vol. 7, no. 2, pp. 354–361, May 2011, doi:10.1109/TII.2011.2123906.

[57] Z. Fang and W. Gao, "Adaptive integral backstepping control of a micro-quadrotor," in Intelligent Control and Information Processing (ICICIP), 2011 2nd International Conference on, vol. 2. IEEE, 2011, pp. 910–915, doi:10.1109/ICICIP.2011.6008382.

[58] M. Faria, I. Maza, and A. Viguria, "Applying frontier cells based exploration and lazy theta* path planning over single grid-based world representation for autonomous inspection of large 3d structures with an uas," Journal of Intelligent & Robotic Systems, pp. 1–21, 2018, doi:10.1007/s10846-018-0798-4.

[59] J. L. Foo, J. Knutzon, V. Kalivarapu, J. Oliver, and E. Winer, "Path planning of unmanned aerial vehicles using b-splines and particle swarm optimization," Journal of aerospace computing, Information, and communication, vol. 6, no. 4, pp. 271–290, 2009, doi:10.2514/1.36917.

[60] K. W. Franke, K. M. Rollins, C. Ledezma, J. D. Hedengren, D. Wolfe, S. Ruggles, C. Bender, and B. Reimschiissel, "Reconnaissance of two lique-

faction sites using small unmanned aerial vehicles and structure from motion computer vision following the april 1, 2014 chile earthquake," Journal of Geotechnical and Geoenvironmental Engineering, vol. 143, no. 5, p. 04016125, 2016, doi:10.1061/(ASCE)GT.1943-5606.0001647.

[61] C. Fu, A. Sarabakha, E. Kayacan, C. Wagner, R. John, and J. M. Garibaldi, "A comparative study on the control of quadcopter uavs by using singleton and non-singleton fuzzy logic controllers," in Fuzzy Systems (FUZZ-IEEE), 2016 IEEE International Conference on. IEEE, 2016, pp. 1023–1030, doi:10.1109/FUZZ-IEEE.2016.7737800.

[62] Y. Fu, M. Ding, and C. Zhou, "Phase angle-encoded and quantum-behaved particle swarm optimization applied to three-dimensional route planning for uav," IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans, vol. 42, no. 2, pp. 511–526, 2012, doi:10.1109/TSMCA.2011.2159586.

[63] Y. Fu, M. Ding, C. Zhou, and H. Hu, "Route planning for unmanned aerial vehicle (uav) on the sea using hybrid differential evolution and quantum-behaved particle swarm optimization," IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 43, no. 6, pp. 1451–1465, 2013, doi:10.1109/TSMC.2013.2248146.

[64] J. S. Gadda and R. D. Patil, "Quadcopter (uavs) for border security with gui system," International Journal of Engineering Research and Technology, vol. 2, no. 12, pp. 620–624, 2013.

[65] S. A. Gautam and N. Verma, "Path planning for unmanned aerial vehicle based on genetic algorithm & artificial neural network in 3d," in Data Mining and Intelligent Computing (ICDMIC), 2014 International Conference on. IEEE, 2014, pp. 1–5, doi:10.1109/ICDMIC.2014.6954257.

[66] J. H. Gillula, G. M. Hoffmann, H. Huang, M. P. Vitus, and C. J. Tomlin, "Applications of hybrid reachability analysis to robotic aerial vehicles," The International Journal of Robotics Research, vol. 30, no. 3, pp. 335–354, 2011, doi:10.1177/0278364910387173.

[67] C. Goerzen, Z. Kong, and B. Mettler, "A survey of motion planning algorithms from the perspective of autonomous uav guidance," Journal of Intelligent and Robotic Systems, vol. 57, no. 1-4, p. 65, 2010, doi:10.1007/s10846-009-9383-1.

[68] J. Gonçalves and R. Henriques, "Uav photogrammetry for topographic monitoring of coastal areas," ISPRS Journal of Photogrammetry and Remote Sensing, vol. 104, pp. 101–111, 2015, doi:10.1016/j.isprsjprs.2015.02.009.

[69] L. Gupta, R. Jain, and G. Vaszkun, "Survey of important issues in uav communication networks," IEEE Communications Surveys & Tutorials, vol. 18, no. 2, pp. 1123–1152, 2016, doi:10.1109/COMST.2015.2495297.

[70] Q. P. Ha, M. T. Nguyen, J. Li, and N. M. Kwok, "Smart structures with current-driven mr dampers: Modeling and second-order sliding mode control," IEEE/ASME Transactions on Mechatronics, vol. 18, no. 6, pp. 1702–1712, Dec 2013, doi:10.1109/TMECH.2013.2280282.

[71] Y. Ham, K. K. Han, J. J. Lin, and M. Golparvar-Fard, "Visual monitoring of civil infrastructure systems via camera-equipped unmanned aerial vehicles (uavs): a review of related works," Visualization in Engineering, vol. 4, no. 1, p. 1, 2016, doi:10.1186/s40327-015-0029-z.

[72] H. Helble and S. Cameron, "3-d path planning and target trajectory prediction for the oxford aerial tracking system," in Robotics and Automation, 2007 IEEE International Conference on. IEEE, 2007, pp. 1042–1048, doi:10.1109/ROBOT.2007.363122.

[73] V. T. Hoang, M. D. Phung, T. H. Dinh, and Q. P. Ha, "Angle-encoded swarm optimization for uav formation path planning," in 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Oct 2018, pp. 5239–5244, doi:10.1109/IROS.2018.8593930.

[74] V. T. Hoang, M. D. Phung, and Q. P. Ha, "Adaptive twisting sliding mode control for quadrotor unmanned aerial vehicles," in 2017 11th Asian Control Conference (ASCC), Dec 2017, pp. 671–676, doi:10.1109/ASCC.2017.8287250.

[75] V. T. Hoang, A. M. Singh, M. D. Phung, and Q. P. Ha, "Adaptive second-order sliding mode control of uavs for civil applications," in Automation and Robotics in Construction (ISARC), 2017 International Symposium on, 2017, pp. 816–822, doi:10.22260/ISARC2017/0114.

[76] J. F. Horn, E. M. Schmidt, B. R. Geiger, and M. P. DeAngelo, "Neural network-based trajectory optimization for unmanned aerial vehicles," Journal of Guidance, Control, and Dynamics, vol. 35, no. 2, pp. 548–562, 2012, doi:10.2514/1.53889.

[77] V. Hosseinnezhad and E. Babaei, "Economic load dispatch using θ-pso," International journal of electrical power & energy systems, vol. 49, pp. 160–169, 2013, doi:10.1016/j.ijepes.2013.01.002.

[78] S. Hota and D. Ghose, "Optimal path planning for an aerial vehicle in 3d space," in Decision and Control (CDC), 2010 49th IEEE Conference on. IEEE, 2010, pp. 4902–4907, doi:10.1109/CDC.2010.5717246.

[79] Z. Hou and I. Fantoni, "Leader-follower formation saturated control for multiple quadrotors with switching topology," in Research, Education and Development of Unmanned Aerial Systems (RED-UAS), 2015 Workshop on. IEEE, 2015, pp. 8–14, doi:10.1109/RED-UAS.2015.7440985.

[80] X. Huo, M. Huo, and H. R. Karimi, "Attitude stabilization control of a quadrotor uav by using backstepping approach," Mathematical Problems in Engineering, vol. 2014, 2014, doi:10.1155/2014/749803.

[81] M. Q. Huynh, W. Zhao, and L. Xie, "L1 adaptive control for quadcopter: Design and implementation," in Control Automation Robotics & Vision (ICARCV), 2014 13th International Conference on. IEEE, 2014, pp. 1496–1501, doi:10.1109/ICARCV.2014.7064537.

[82] J. Y. Hwang, J. S. Kim, S. S. Lim, and K. H. Park, "A fast path planning by path graph optimization," IEEE Transactions on systems, man, and cybernetics-part a: systems and humans, vol. 33, no. 1, pp. 121–129, 2003, doi:10.1109/TSMCA.2003.812599.

[83] Y. K. Hwang and N. Ahuja, "A potential field approach to path planning," IEEE Transactions on Robotics and Automation, vol. 8, no. 1, pp. 23–32, 1992, doi:10.1109/70.127236.

[84] S. Islam, P. X. Liu, and A. E. Saddik, "Robust control of four-rotor unmanned aerial vehicle with disturbance uncertainty," IEEE Transactions on Industrial Electronics, vol. 62, no. 3, pp. 1563–1571, March 2015, doi:10.1109/TIE.2014.2365441.

[85] C. Izaguirre-Espinosa, A. J. Muñoz-Vázquez, A. Sánchez-Orta, V. Parra-Vega, and P. Castillo, "Attitude control of quadrotors based on fractional sliding modes: theory and experiments," IET Control Theory & Applications, vol. 10, no. 7, pp. 825–832, 2016, doi:10.1049/iet-cta.2015.1048.

[86] A. S. Joshi, I. A. Shaikh, D. M. Paul, N. R. Patil, and D. Shedge, "Data acquisition system & visual surveillance at remote locations using quad copter,"

International Journal of Advances in Engineering & Technology, vol. 7, no. 2, p. 575, 2014.

[87] J. Katrasnik, F. Pernus, and B. Likar, "A survey of mobile robots for distribution power line inspection," IEEE Transactions on Power Delivery, vol. 25, no. 1, pp. 485–493, Jan 2010, doi:10.1109/TPWRD.2009.2035427.

[88] S. Keshmiri and S. Payandeh, "A centralized framework to multi-robots formation control: Theory and application," in Collaborative Agents-Research and Development. Springer, 2011, pp. 85–98, doi:978-3-642-22427-0_7.

[89] D. H. Kim, H. Wang, and S. Shin, "Decentralized control of autonomous swarm systems using artificial potential functions: Analytical design guidelines," Journal of Intelligent and Robotic Systems, vol. 45, no. 4, pp. 369–394, 2006, doi:10.1007/s10846-006-9050-8.

[90] S. Kim and Y. Kim, "Optimum design of three-dimensional behavioural decentralized controller for uav formation flight," Engineering Optimization, vol. 41, no. 3, pp. 199–224, 2009, doi:10.1080/03052150802406532.

[91] S. Kim, H. Oh, J. Suk, and A. Tsourdos, "Coordinated trajectory planning for efficient communication relay using multiple uavs," Control Engineering Practice, vol. 29, pp. 42 – 49, 2014, doi:10.1016/j.conengprac.2014.04.003.

[92] C. Korpela, P. Root, J. Kim, S. Wilkerson, and S. A. Gadsden, "A framework for autonomous and continuous aerial intelligence, surveillance, and reconnaissance operations," in SPIE Defense+ Security. International Society for Optics and Photonics, 2016, pp. 982 803–982 803, doi:10.1117/12.2223350.

[93] M. Kothari and I. Postlethwaite, "A probabilistically robust path planning algorithm for uavs using rapidly-exploring random trees," Journal of Intelligent & Robotic Systems, pp. 1–23, 2013, doi:10.1007/s10846-012-9776-4.

[94] Y. Kuriki and T. Namerikawa, "Consensus-based cooperative formation control with collision avoidance for a multi-uav system," in American Control Conference (ACC), 2014. IEEE, 2014, pp. 2077–2082, doi:10.1109/ACC.2014.6858777.

[95] A. Kushleyev, D. Mellinger, C. Powers, and V. Kumar, "Towards a swarm of agile micro quadrotors," Autonomous Robots, vol. 35, no. 4, pp. 287–300, 2013, doi:10.1007/s10514-013-9349-9.

[96] N. M. Kwok, Q. P. Ha, D. Liu, G. Fang, and K. C. Tan, "Efficient particle swarm optimization: a termination condition based on the decision-making approach," in Evolutionary Computation, 2007. CEC 2007. IEEE Congress on, 2007, pp. 3353–336, doi:10.1109/CEC.2007.4424905.

[97] S. M. LaValle, "Rapidly-exploring random trees: A new tool for path planning," 1998.

[98] S. M. LaValle, "From dynamic programming to rrts: Algorithmic design of feasible trajectories," in Control Problems in Robotics. Springer, 2003, pp. 19–37, doi:10.1007/3-540-36224-X_2.

[99] K. Lee, J. Back, and I. Choy, "Nonlinear disturbance observer based robust attitude tracking controller for quadrotor uavs," International Journal of Control, Automation and Systems, vol. 12, no. 6, pp. 1266–1275, Dec 2014, doi:10.1007/s12555-014-0145-x.

[100] T. Lee, M. Leok, and N. H. McClamroch, "Nonlinear robust tracking control of a quadrotor uav on se (3)," Asian Journal of Control, vol. 15, no. 2, pp. 391–408, 2013, doi:10.1002/asjc.567.

[101] A. Levant, "Quasi-continuous high-order sliding-mode controllers," IEEE Transactions on Automatic Control, vol. 50, no. 11, pp. 1812–1816, Nov 2005,

doi:10.1109/TAC.2005.858646.

[102] A. Levant, "Sliding order and sliding accuracy in sliding mode control," International journal of control, vol. 58, no. 6, pp. 1247–1263, 1993, 10.1080/00207179308923053.

[103] A. Levant, "Higher-order sliding modes, differentiation and output-feedback control," International journal of Control, vol. 76, no. 9-10, pp. 924–941, 2003, doi:10.1080/0020717031000099029.

[104] M. A. Lewis and K.-H. Tan, "High precision formation control of mobile robots using virtual structures," Autonomous robots, vol. 4, no. 4, pp. 387–403, 1997, doi:10.1023/A:100881470.

[105] 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 Chinese Automation Congress (CAC), 2013. IEEE, 2013, pp. 697–702, doi:10.1109/CAC.2013.6775824.

[106] J. Li, G. Deng, C. Luo, Q. Lin, Q. Yan, and Z. Ming, "A hybrid path planning method in unmanned air/ground vehicle (uav/ugv) cooperative systems," IEEE Transactions on Vehicular Technology, vol. 65, no. 12, pp. 9585–9596, 2016, doi:10.1109/TVT.2016.2623666.

[107] N. H. Li and H. H. Liu, "Formation uav flight control using virtual structure and motion synchronization," in American Control Conference, 2008. IEEE, 2008, pp. 1782–1787, doi:10.1109/ACC.2008.4586750.

[108] F. Liao, R. Teo, J. L. Wang, X. Dong, F. Lin, and K. Peng, "Distributed formation and reconfiguration control of vtol uavs," IEEE Transactions on Control Systems Technology, vol. 25, no. 1, pp. 270–277, Jan 2017, doi:10.1109/TCST.2016.2547952.

[109] G. J. Lim, S. Kim, J. Cho, Y. Gong, and A. Khodaei, "Multi-uav pre-positioning and routing for power network damage assessment," IEEE Transactions on Smart Grid, vol. 9, no. 4, pp. 3643–3651, 2016, doi:10.1109/TSG.2016.2637408.

[110] H. Liu, Y. Bai, G. Lu, Z. Shi, and Y. Zhong, "Robust tracking control of a quadrotor helicopter," Journal of Intelligent & Robotic Systems, vol. 75, no. 3, pp. 595–608, Sep 2014, doi:10.1007/s10846-013-9838-2.

[111] H. Liu, J. Xi, and Y. Zhong, "Robust motion control of quadrotors," Journal of the Franklin Institute, vol. 351, no. 12, pp. 5494–5510, 2014, doi:10.1016/j.jfranklin.2014.10.003.

[112] P. Liu, A. Y. Chen, Y.-N. Huang, J.-Y. Han, J.-S. Lai, S.-C. Kang, T. Wu, M.-C. Wen, M. Tsai et al., "A review of rotorcraft unmanned aerial vehicle (uav) developments and applications in civil engineering," Smart Struct. Syst, vol. 13, no. 6, pp. 1065–1094, 2014, doi:10.12989/sss.2014.13.6.1065.

[113] J. Long, Z. Gao, P. Orenstein, and H. Ren, "Control strategies for dispersing incident-based traffic jams in two-way grid networks," IEEE Transactions on Intelligent Transportation Systems, vol. 13, no. 2, pp. 469–481, 2012, doi:10.1109/TITS.2011.2171035.

[114] Y. Lu, D. Macias, Z. S. Dean, N. R. Kreger, and P. K. Wong, "A uav-mounted whole cell biosensor system for environmental monitoring applications," IEEE transactions on nanobioscience, vol. 14, no. 8, pp. 811–817, 2015, doi:10.1109/TNB.2015.2478481.

[115] G.-c. Luo, J.-q. Yu, Y.-s. Mei, and S.-y. Zhang, "Uav path planning in mixed-obstacle environment via artificial potential field method improved by additional control force," Asian Journal of Control, vol. 17, no. 5, pp. 1600–1610,

2015, doi:10.1002/asjc.960.

[116] J. Luo, W. Su, and D. Wang, "The improvement of the artificial potential field robot path planning based on 3-d space," in Automatic Control and Artificial Intelligence (ACAI 2012), International Conference on. IET, 2012, pp. 2128–2131, doi:10.1049/cp.2012.1419.

[117] Q. Luo and H. Duan, "Distributed uav flocking control based on homing pigeon hierarchical strategies," Aerospace Science and Technology, vol. 70, pp. 257–264, 2017, doi:10.1016/j.ast.2017.08.010.

[118] Y. Luo, M. J. Er, L. L. Yong, and C. J. Chien, "Intelligent control and navigation of an indoor quad-copter," in Control Automation Robotics & Vision (ICARCV), 2014 13th International Conference on. IEEE, 2014, pp. 1700–1705, doi:10.1109/ICARCV.2014.7064572.

[119] L. Luque-Vega, B. Castillo-Toledo, and A. G. Loukianov, "Robust block second order sliding mode control for a quadrotor," Journal of the Franklin Institute, vol. 349, no. 2, pp. 719–739, 2012, doi:10.1016/j.jfranklin.2011.10.017.

[120] G. Ma, H. Huang, and Y. Zhuang, "Time optimal trajectory planning for reconfiguration of satellite formation with collision avoidance," in Control and Automation (ICCA), 2010 8th IEEE International Conference on. IEEE, 2010, pp. 476–479, doi:10.1109/ICCA.2010.5524223.

[121] T. T. Mac, C. Copot, D. T. Tran, and R. De Keyser, "Heuristic approaches in robot path planning: A survey," Robotics and Autonomous Systems, vol. 86, pp. 13–28, 2016, doi:10.1016/j.robot.2016.08.001.

[122] B. MacAllister, J. Butzke, A. Kushleyev, H. Pandey, and M. Likhachev, "Path planning for non-circular micro aerial vehicles in constrained environments,"

in Robotics and Automation (ICRA), 2013 IEEE International Conference on. IEEE, 2013, pp. 3933–3940, doi:10.1109/ICRA.2013.6631131.

[123] T. Madani and A. Benallegue, "Adaptive control via backstepping technique and neural networks of a quadrotor helicopter," IFAC Proceedings Volumes, vol. 41, no. 2, pp. 6513–6518, 2008, doi:10.3182/20080706-5-KR-1001.01098.

[124] T. Madani and A. Benallegue, "Backstepping control for a quadrotor helicopter," in Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on. IEEE, 2006, pp. 3255–3260, doi:10.1109/IROS.2006.282433.

[125] E. Magid, R. Lavrenov, and I. Afanasyev, "Voronoi-based trajectory optimization for ugv path planning," in Mechanical, System and Control Engineering (ICMSC), 2017 International Conference on. IEEE, 2017, pp. 383–387, doi:10.1109/ICMSC.2017.7959506.

[126] N. Malone, H.-T. Chiang, K. Lesser, M. Oishi, and L. Tapia, "Hybrid dynamic moving obstacle avoidance using a stochastic reachable set-based potential field," IEEE Transactions on Robotics, vol. 33, no. 5, pp. 1124–1138, 2017, doi:10.1109/TRO.2017.2705034.

[127] M. Manceur, N. Essounbouli, and A. Hamzaoui, "Second-order sliding fuzzy interval type-2 control for an uncertain system with real application," Fuzzy Systems, IEEE Transactions on, vol. 20, no. 2, pp. 262–275, 2012, doi:10.1109/TFUZZ.2011.2172948.

[128] R. Mark and E. Billo, "Low altitude unmanned aerial photography to assist in rock art studies," SAA Archaeological Record, 2016.

[129] R. A. Martin, A. Hall, C. Brinton, K. Franke, and J. D. Hedengren, "Privacy aware mission planning and video masking for uav systems," Proceedings

of the AIAA Infotech@ Aerospace, San Diego, CA, USA, pp. 4–8, 2016, doi:10.2514/6.2016-0250.

[130] J. L. Mendoza-Soto and H. R. Cortés, "Generalized predictive control for trajectory tracking of a quadcopter vehicle," in Unmanned Aircraft Systems (ICUAS), 2017 International Conference on. IEEE, 2017, pp. 206–212, doi:10.1109/ICUAS.2017.7991368.

[131] B.-b. Meng and X. Gao, "Uav path planning based on bidirectional sparse a* search algorithm," in Intelligent Computation Technology and Automation (ICICTA), 2010 International Conference on, vol. 3. IEEE, 2010, pp. 1106–1109, doi:10.1109/ICICTA.2010.235.

[132] D. Mercado, R. Castro, and R. Lozano, "Quadrotors flight formation control using a leader-follower approach," in Control Conference (ECC), 2013 European. IEEE, 2013, pp. 3858–3863, doi:10.23919/ECC.2013.6669637.

[133] N. Metni and T. Hamel, "A uav for bridge inspection: Visual servoing control law with orientation limits," Automation in construction, vol. 17, no. 1, pp. 3–10, 2007, doi:10.1016/j.autcon.2006.12.010.

[134] B. Min, J. Hong, and E. T. Matson, "Adaptive robust control (arc) for an altitude control of a quadrotor type uav carrying an unknown payloads," in 2011 11th International Conference on Control, Automation and Systems, Oct 2011, pp. 1147–1151.

[135] L. D. Minh and C. Ha, "Modeling and control of quadrotor mav using vision-based measurement," in Strategic Technology (IFOST), 2010 International Forum on. IEEE, 2010, pp. 70–75, doi:10.1109/IFOST.2010.5668079.

[136] S. Mintchev and D. Floreano, "A pocket sized foldable quadcopter for situational awareness and reconnaissance," in Safety, Security, and Rescue Robotics

(SSRR), 2016 IEEE International Symposium on. Ieee, 2016, pp. 396–401, doi:10.1109/SSRR.2016.7784333.

[137] A. Mokhtari, A. Benallegue, and B. Daachi, "Robust feedback linearization and gh/sub/spl infin//controller for a quadrotor unmanned aerial vehicle," in Intelligent Robots and Systems, 2005.(IROS 2005). 2005 IEEE/RSJ International Conference on. IEEE, 2005, pp. 1198–1203, doi:10.1109/IROS.2005.1545112.

[138] M. R. Mokhtari and B. Cherki, "A new robust control for minirotorcraft unmanned aerial vehicles," ISA transactions, vol. 56, pp. 86–101, 2015, doi:10.1016/j.isatra.2014.12.003.

[139] M. W. Mueller and R. D'Andrea, "A model predictive controller for quadrocopter state interception," in Control Conference (ECC), 2013 European. IEEE, 2013, pp. 1383–1389, doi:10.23919/ECC.2013.6669415.

[140] F. Muñoz, I. González-Hernández, S. Salazar, E. S. Espinoza, and R. Lozano, "Second order sliding mode controllers for altitude control of a quadrotor uas: Real-time implementation in outdoor environments," Neurocomputing, vol. 233, pp. 61–71, 2017, doi:10.1016/j.neucom.2016.08.111.

[141] A. Nash and S. Koenig, "Any-angle path planning," AI Magazine, vol. 34, no. 4, pp. 85–107, 2013, doi:10.1609/aimag.v34i4.2512.

[142] P. T. Nathan, H. A. Almurib, and T. N. Kumar, "A review of autonomous multi-agent quad-rotor control techniques and applications," in Mechatronics (ICOM), 2011 4th International Conference On. IEEE, 2011, pp. 1–7, doi:10.1109/ICOM.2011.5937132.

[143] J. Navia, I. Mondragon, D. Patino, and J. Colorado, "Multispectral mapping in agriculture: terrain mosaic using an autonomous quadcopter uav,"

in Unmanned Aircraft Systems (ICUAS), 2016 International Conference on. IEEE, 2016, pp. 1351–1358, doi:10.1109/ICUAS.2016.7502606.

[144] K. Nonami, F. Kendoul, S. Suzuki, W. Wang, and D. Nakazawa, "Autonomous flying robots-unmanned aerial vehicles and micro aerial vehicles," 2010, doi:10.1007/978-4-431-53856-1.

[145] T. Nuchkrua and M. Parnichkun, "Identification and optimal control of quadrotor," Thammasat International Journal of Science and Technology, vol. 17, no. 4, pp. 36–53, 2012.

[146] S. Nuske, S. Choudhury, S. Jain, A. Chambers, L. Yoder, S. Scherer, L. Chamberlain, H. Cover, and S. Singh, "Autonomous exploration and motion planning for an unmanned aerial vehicle navigating rivers," Journal of Field Robotics, vol. 32, no. 8, pp. 1141–1162, 2015, doi:10.1002/rob.21596.

[147] 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, pp. 693–707, Jan 2014, doi:10.1007/s10846-013-9936-1.

[148] I. Palunko and R. Fierro, "Adaptive control of a quadrotor with dynamic changes in the center of gravity," IFAC Proceedings Volumes, vol. 44, no. 1, pp. 2626–2631, 2011, doi:10.3182/20110828-6-IT-1002.02564.

[149] D. Panagou and V. Kumar, "Maintaining visibility for leader-follower formations in obstacle environments," in Robotics and Automation (ICRA), 2012 IEEE International Conference on. IEEE, 2012, pp. 1811–1816, doi:10.1109/ICRA.2012.6224893.

[150] D. Panagou and V. Kumar, "Cooperative visibility maintenance for leader–follower formations in obstacle environments," IEEE Transactions on Robotics, vol. 30, no. 4, pp. 831–844, 2014, doi:10.1109/TRO.2014.2304774.

[151] Y. V. Pehlivanoglu, "A new vibrational genetic algorithm enhanced with a voronoi diagram for path planning of autonomous uav," Aerospace Science and Technology, vol. 16, no. 1, pp. 47–55, 2012, doi:10.1016/j.ast.2011.02.006.

[152] C. Peng, Y. Bai, X. Gong, Q. Gao, C. Zhao, and Y. Tian, "Modeling and robust backstepping sliding mode control with adaptive rbfnn for a novel coaxial eight-rotor uav," IEEE/CAA Journal of Automatica Sinica, vol. 2, no. 1, pp. 56–64, January 2015, doi:10.1109/JAS.2015.7032906.

[153] L. Petricca, P. Ohlckers, and C. Grinde, "Micro-and nano-air vehicles: State of the art," International journal of aerospace engineering, vol. 2011, 2011, doi:10.1155/2011/214549.

[154] M. D. Phung, T. H. Dinh, V. T. Hoang, and Q. P. Ha, "Automatic crack detection in built infrastructure using unmanned aerial vehicles," in Automation and Robotics in Construction (ISARC), 2017 International Symposium on, 2017, pp. 823–829, doi:10.22260/ISARC2017/0115.

[155] M. D. Phung, C. H. Quach, D. T. Chu, N. Q. Nguyen, T. H. Dinh, and Q. Ha, "Automatic interpretation of unordered point cloud data for uav navigation in construction," in Control, Automation, Robotics and Vision (ICARCV), 2016 14th International Conference on. IEEE, 2016, pp. 1–6, doi:10.1109/ICARCV.2016.7838683.

[156] M. D. Phung, C. H. Quach, T. H. Dinh, and Q. Ha, "Enhanced discrete particle swarm optimization path planning for uav vision-based surface inspection," Automation in Construction, vol. 81, pp. 25–33, 2017, doi:10.1016/j.autcon.2017.04.013.

[157] F. Plestan, Y. Shtessel, V. Bregeault, and A. Poznyak, "New methodologies for adaptive sliding mode control," International journal of control, vol. 83,

no. 9, pp. 1907–1919, 2010, doi:10.1080/00207179.2010.501385.

[158] R. Poli, J. Kennedy, and T. Blackwell, "Particle swarm optimization," Swarm intelligence, vol. 1, no. 1, pp. 33–57, 2007, doi:10.1007/s11721-007-0002-0.

[159] A. Polyakov and A. Poznyak, "Lyapunov function design for finite-time convergence analysis:twisting controller for second-order sliding mode realization," Automatica, vol. 45, no. 2, pp. 444–448, 2009, doi:10.1016/j.automatica.2008.07.013.

[160] A. Polyakov and A. Poznyak, "Reaching time estimation for super-twisting second order sliding mode controller via lyapunov function designing," IEEE Transactions on Automatic Control, vol. 54, no. 8, pp. 1951–1955, 2009, doi:10.1109/TAC.2009.2023781.

[161] C. Pukdeboon, A. S. Zinober, and M.-W. L. Thein, "Quasi-continuous higher order sliding-mode controllers for spacecraft-attitude-tracking maneuvers," IEEE Transactions on Industrial Electronics, vol. 57, no. 4, pp. 1436–1444, 2010.

[162] M. Radmanesh, M. Kumar, P. H. Guentert, and M. Sarim, "Overview of path-planning and obstacle avoidance algorithms for uavs: A comparative study," Unmanned Systems, pp. 1–24, 2018, doi:10.1142/S2301385018400022.

[163] G. V. Raffo, M. G. Ortega, and F. R. Rubio, "An integral predictive/nonlinear H_∞ control structure for a quadrotor helicopter," Automatica, vol. 46, no. 1, pp. 29–39, 2010, doi:10.1016/j.automatica.2009.10.018.

[164] S. Rajappa, C. Masone, H. H. Blthoff, and P. Stegagno, "Adaptive super twisting controller for a quadrotor uav," in 2016 IEEE International Conference on Robotics and Automation (ICRA), May 2016, pp. 2971–2977, doi:10.1109/ICRA.2016.7487462.

[165] S. M. Rakhtala and M. Ahmadi, "Twisting control algorithm for the yaw and pitch tracking of a twin rotor uav," International Journal of Automation and Control, vol. 11, no. 2, pp. 143–163, 2017, doi:10.1504/IJAAC.2017.083296.

[166] H. Ramirez-Rodriguez, V. Parra-Vega, A. Sanchez-Orta, and O. Garcia-Salazar, "Robust backstepping control based on integral sliding modes for tracking of quadrotors," Journal of Intelligent & Robotic Systems, vol. 73, no. 1, pp. 51–66, Jan 2014, doi:10.1007/s10846-013-9909-4.

[167] R. V. Rao, V. J. Savsani, and D. Vakharia, "Teaching–learning-based optimization: a novel method for constrained mechanical design optimization problems," Computer-Aided Design, vol. 43, no. 3, pp. 303–315, 2011, doi:10.1016/j.cad.2010.12.015.

[168] A. T. Rashid, A. A. Ali, M. Frasca, and L. Fortuna, "Path planning with obstacle avoidance based on visibility binary tree algorithm," Robotics and Autonomous Systems, vol. 61, no. 12, pp. 1440–1449, 2013, doi:10.1016/j.robot.2013.07.010.

[169] H. Rezaee and F. Abdollahi, "A decentralized cooperative control scheme with obstacle avoidance for a team of mobile robots," IEEE Transactions on Industrial Electronics, vol. 61, no. 1, pp. 347–354, 2014, doi:10.1109/TIE.2013.2245612.

[170] G. J. Rubio, J. M. Cañedo, V. I. Utkin, and A. G. Loukianov, "Second order sliding mode block control of single-phase induction motors," International Journal of Robust and Nonlinear Control, vol. 24, no. 4, pp. 682–698, 2014, doi:10.1002/rnc.2913.

[171] D. Sathyamoorthy, "A review of security threats of unmanned aerial vehicles and mitigation steps," The Journal of Defence and Security, vol. 6, no. 1,

p. 81, 2015.

[172] A. C. Satici, H. Poonawala, and M. W. Spong, "Robust optimal control of quadrotor uavs," IEEE Access, vol. 1, pp. 79–93, 2013, doi:10.1109/ACCESS.2013.2260794.

[173] L. Sauter and P. Palmer, "Onboard semianalytic approach to collision-free formation reconfiguration," IEEE Transactions on Aerospace and Electronic Systems, vol. 48, no. 3, pp. 2638–2652, 2012, doi:10.1177/0278364913515307.

[174] E. Semsar-Kazerooni and K. Khorasani, "Switching control of a modified leader–follower team of agents under the leader and network topological changes," IET control theory & applications, vol. 5, no. 12, pp. 1369–1377, 2011, doi:10.1049/iet-cta.2010.0396.

[175] E. Semsar-Kazerooni and K. Khorasani, "Optimal consensus algorithms for cooperative team of agents subject to partial information," Automatica, vol. 44, no. 11, pp. 2766–2777, 2008, doi:10.1016/j.automatica.2008.04.016.

[176] R. Sharma and D. Ghose, "Collision avoidance between uav clusters using swarm intelligence techniques," International Journal of Systems Science, vol. 40, no. 5, pp. 521–538, 2009, doi:10.1080/00207720902750003.

[177] Z. Shen, Z. He, S. Li, Q. Wang, and Z. Shao, "A multi-quadcopter cooperative cyber-physical system for timely air pollution localization," ACM Transactions on Embedded Computing Systems (TECS), vol. 16, no. 3, p. 70, 2017, doi:10.1145/3005716.

[178] Y. Shi and R. C. Eberhart, "Parameter selection in particle swarm optimization," in International conference on evolutionary programming. Springer, 1998, pp. 591–600, doi:10.1007/BFb0040810.

[179] Y. Shtessel, M. Taleb, and F. Plestan, "A novel adaptive-gain supertwisting sliding mode controller: methodology and application," Automatica, vol. 48, no. 5, pp. 759–769, 2012, doi:10.1016/j.automatica.2012.02.024.

[180] S. Siebert and J. Teizer, "Mobile 3d mapping for surveying earthwork projects using an unmanned aerial vehicle (uav) system," Automation in Construction, vol. 41, pp. 1–14, 2014, doi:10.1016/j.autcon.2014.01.004.

[181] K. Sigurd and J. How, "Uav trajectory design using total field collision avoidance," in AIAA Guidance, Navigation, and Control Conference and Exhibit, 2003, p. 5728, doi:10.2514/6.2003-5728.

[182] A. Sonmez, E. Kocyigit, and E. Kugu, "Optimal path planning for uavs using genetic algorithm," in Unmanned Aircraft Systems (ICUAS), 2015 International Conference on. IEEE, 2015, pp. 50–55, doi:10.1109/ICUAS.2015.7152274.

[183] E. Stingu and F. L. Lewis, "A hardware platform for research in helicopter uav control," Journal of Intelligent and Robotic Systems, vol. 54, no. 1, pp. 387–406, Mar 2009, doi:10.1007/s10846-008-9271-0.

[184] M. Sudarma, I. A. Swamardika, and A. M. Pratama, "Design of quadcopter robot as a disaster environment remote monitor," International Journal of Electrical and Computer Engineering, vol. 6, no. 1, p. 188, 2016.

[185] C. S. Tan, R. Sutton, and J. Chudley, "An incremental stochastic motion planning technique for autonomous underwater vehicles," in Proceedings of IFAC Control Applications in Marine Systems Conference, 2004, pp. 483–488, doi:10.1016/S1474-6670(17)31778-0.

[186] H. G. Tanner, G. J. Pappas, and V. Kumar, "Leader-to-formation stability," IEEE Transactions on Robotics and Automation, vol. 20, no. 3, pp. 443–455,

June 2004, doi:10.1109/TRA.2004.825275.

[187] M. C. Tatum and J. Liu, "Unmanned aircraft system applications in construction," Procedia Engineering, vol. 196, pp. 167–175, 2017, doi:10.1016/j.proeng.2017.07.187.

[188] K. M. Thu and A. Gavrilov, "Designing and modeling of quadcopter control system using l1 adaptive control," Procedia Computer Science, vol. 103, pp. 528–535, 2017, doi:10.1016/j.procs.2017.01.046.

[189] C. T. Ton and W. MacKunis, "Robust attitude tracking control of a quadrotor helicopter in the presence of uncertainty," in 2012 IEEE 51st IEEE Conference on Decision and Control (CDC), Dec 2012, pp. 937–942, doi:10.1109/CDC.2012.6426266.

[190] M. Turpin, N. Michael, and V. Kumar, "Capt: Concurrent assignment and planning of trajectories for multiple robots," The International Journal of Robotics Research, vol. 33, no. 1, pp. 98–112, 2014, doi:10.1177/0278364913515307.

[191] V. Utkin, "Variable structure systems with sliding modes," IEEE Transactions on Automatic Control, vol. 22, no. 2, pp. 212–222, April 1977, doi:10.1109/TAC.1977.1101446.

[192] V. Utkin, "Discussion aspects of high-order sliding mode control," IEEE Transactions on Automatic Control, vol. 61, no. 3, pp. 829–833, 2016, doi:10.1109/TAC.2015.2450571.

[193] T. F. Villa, F. Gonzalez, B. Miljievic, Z. D. Ristovski, and L. Morawska, "An overview of small unmanned aerial vehicles for air quality measurements: Present applications and future prospectives," Sensors, vol. 16, no. 7, p. 1072, 2016, doi:10.3390/s16071072.

[194] R. J. Wallace and J. M. Loffi, "Examining unmanned aerial system threats & defenses: A conceptual analysis," International Journal of Aviation, Aeronautics, and Aerospace, vol. 2, no. 4, p. 1, 2015, doi:10.15394/ijaaa.2015.1084.

[195] J. Wang and M. Xin, "Integrated optimal formation control of multiple unmanned aerial vehicles," IEEE Transactions on Control Systems Technology, vol. 21, no. 5, pp. 1731–1744, 2013, doi:10.1109/TCST.2012.2218815.

[196] K. Wang, "Minimum-fuel formation reconfiguration of multiple free-flying spacecraft," J. of Astronautical Sciences, vol. 47, no. 1, pp. 77–102, 1999.

[197] P. Wang and F. Hadaegh, "Optimal formation-reconfiguration for multiple spacecraft," in Guidance, Navigation, and Control Conference and Exhibit, 1998, p. 4226, doi:10.2514/6.1998-4226.

[198] S. Wang and C. Zheng, "A hierarchical evolutionary trajectory planner for spacecraft formation reconfiguration," IEEE Transactions on Aerospace and Electronic Systems, vol. 48, no. 1, pp. 279–289, 2012, doi:10.1109/TAES.2012.6129635.

[199] Z. Wang and J. Cai, "Probabilistic roadmap method for path-planning in radioactive environment of nuclear facilities," Progress in Nuclear Energy, vol. 109, pp. 113–120, 2018, doi:10.1016/j.pnucene.2018.08.006.

[200] Z. Wei-min, L. Shao-jun, and Q. Feng, "θ-pso: a new strategy of particle swarm optimization," Journal of Zhejiang University-SCIENCE A, vol. 9, no. 6, pp. 786–790, Jun 2008, doi:10.1631/jzus.A071278.

[201] N. Wen, X. Su, P. Ma, L. Zhao, and Y. Zhang, "Online uav path planning in uncertain and hostile environments," International Journal of Machine

Learning and Cybernetics, vol. 8, no. 2, pp. 469–487, 2017, doi:10.1007/s13042-015-0339-4.

[202] N. Wen, L. Zhao, X. Su, and P. Ma, "Uav online path planning algorithm in a low altitude dangerous environment," IEEE/CAA Journal of Automatica Sinica, vol. 2, no. 2, pp. 173–185, 2015, doi:10.1109/JAS.2015.7081657.

[203] R. Xu and Ü. Özgüner, "Sliding mode control of a quadrotor helicopter," in Decision and Control, 2006 45th IEEE Conference on. IEEE, 2006, pp. 4957–4962, doi:10.1109/CDC.2006.377588.

[204] F. Yan, Y.-S. Liu, and J.-Z. Xiao, "Path planning in complex 3d environments using a probabilistic roadmap method," International Journal of Automation and Computing, vol. 10, no. 6, pp. 525–533, Dec 2013, doi:10.1007/s11633-013-0750-9.

[205] M.-d. Yan, X. Zhu, X.-x. Zhang, and Y.-h. Qu, "Consensus-based three-dimensional multi-uav formation control strategy with high precision," Frontiers of Information Technology & Electronic Engineering, vol. 18, no. 7, pp. 968–977, Jul 2017, doi:10.1631/FITEE.1600004.

[206] A. Yang, W. Naeem, and M. Fei, "Decentralised formation control and stability analysis for multi-vehicle cooperative manoeuvre," IEEE/CAA Journal of Automatica Sinica, vol. 1, no. 1, pp. 92–100, 2014, doi:10.1109/JAS.2014.7004625.

[207] Y. Yang and Y. Yan, "Attitude regulation for unmanned quadrotors using adaptive fuzzy gain-scheduling sliding mode control," Aerospace Science and Technology, vol. 54, pp. 208 – 217, 2016, doi:10.1016/j.ast.2016.04.005.

[208] X. Yu and L. Liu, "Distributed formation control of nonholonomic vehicles subject to velocity constraints," IEEE Transactions on Industrial Electronics,

vol. 63, no. 2, pp. 1289–1298, 2016, doi:10.1109/TIE.2015.2504042.

[209] Y. Zeng, R. Zhang, and T. J. Lim, "Wireless communications with unmanned aerial vehicles: opportunities and challenges," IEEE Communications Magazine, vol. 54, no. 5, pp. 36–42, May 2016, doi:10.1109/MCOM.2016.7470933.

[210] B. Zhang, W. Liu, Z. Mao, J. Liu, and L. Shen, "Cooperative and geometric learning algorithm (cgla) for path planning of uavs with limited information," Automatica, vol. 50, no. 3, pp. 809 – 820, 2014, doi:10.1016/j.automatica.2013.12.035.

[211] B. Zhang, Z. Mao, W. Liu, and J. Liu, "Geometric reinforcement learning for path planning of uavs," Journal of Intelligent & Robotic Systems, vol. 77, no. 2, pp. 391–409, 2015, doi:10.1007/s10846-013-9901-z.

[212] G. Zhang and L.-T. Hsu, "A new path planning algorithm using a gnss localization error map for uavs in an urban area," Journal of Intelligent & Robotic Systems, pp. 1–17, 2018, doi:10.1007/s10846-018-0894-5.

[213] Q. Zhang, C. Zhou, N. Xiong, Y. Qin, X. Li, and S. Huang, "Multimodel-based incident prediction and risk assessment in dynamic cybersecurity protection for industrial control systems," IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 46, no. 10, pp. 1429–1444, 2016, doi:10.1109/TSMC.2015.2503399.

[214] B. Zhao, B. Xian, Y. Zhang, and X. Zhang, "Nonlinear robust adaptive tracking control of a quadrotor uav via immersion and invariance methodology," IEEE Transactions on Industrial Electronics, vol. 62, no. 5, pp. 2891–2902, May 2015, doi:10.1109/TIE.2014.2364982.

[215] Y. Zhao, Z. Zheng, and Y. Liu, "Survey on computational-intelligence-based uav path planning," Knowledge-Based Systems, 2018, doi:10.1016/j.knosys.2018.05.033.

[216] C. Zheng, L. Li, F. Xu, F. Sun, and M. Ding, "Evolutionary route planner for unmanned air vehicles," IEEE Transactions on robotics, vol. 21, no. 4, pp. 609–620, 2005, doi:10.1109/TRO.2005.844684.

[217] E.-H. Zheng, J.-J. Xiong, and J.-L. Luo, "Second order sliding mode control for a quadrotor uav," ISA transactions, vol. 53, no. 4, pp. 1350–1356, 2014, doi:10.1016/j.isatra.2014.03.010.

[218] D. Zhou, Z. Wang, and M. Schwager, "Agile coordination and assistive collision avoidance for quadrotor swarms using virtual structures," IEEE Transactions on Robotics, vol. 34, no. 4, pp. 916–923, Aug 2018, doi:10.1109/TRO.2018.2857477.

[219] J. T. Zou, Z. Y. Pan, D. L. Zhang, and R. F. Zheng, "Integration of the target position correction software with the high endurance quadcopter for search and rescue mission," in Applied Mechanics and Materials, vol. 764. Trans Tech Publ, 2015, pp. 713–717, doi:10.4028/www.scientific.net/AMM.764-765.713.

[220] A. Zulu and S. John, "A review of control algorithms for autonomous quadrotors," Open Journal of Applied Sciences, vol. 4, no. 14, p. 547, 2014, doi:10.4236/ojapps.2014.414053.

[221] Z. Zuo, "Trajectory tracking control design with command-filtered compensation for a quadrotor," IET Control Theory Appl., vol. 4, no. 11, pp. 2343–2355, 2010, doi:10.1049/iet-cta.2009.0336.