Vision Assisted Guidance Techniques for Quadrotor Landing

A THESIS

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE

DEGREE OF

Doctor of Philosophy

BY

ALVIKA GAUTAM

DEPARTMENT OF COMPUTER SCIENCE ENGINEERING

INDRAPRASTHA INSTITUTE OF INFORMATION TECHNOLOGY DELHI

NEW DELHI– 110020

MAY, 2019

Acknowledgements

This thesis is a combined effort of several people who have contributed towards its completion in different ways. The following lines are a humble effort to express my gratitude to all of them. First thanks goes to my advisor, Dr. P.B Sujit, who has been a combination of everything I expected out of my advisor. He has been a source of constant support (in very difficult times, which were quite a few during my PhD), motivation and encouragement and I cannot thank him enough for the belief he showed in me. This thesis and my growth in the past five years are credits to his technical prowess and constant engagement in my work. He has guided me in all technical matters, while forcing me at all the right times to chart my own course. I would also like to thank my co-advisor, Dr. Srikanth Saripalli, without whose guidance and encouragement I wouldn't have been able to complete this journey, especially the hardware experiments. The 6 months I spent at Texas A & M University working with him, were a big learning experience for me. One of the qualities that I observed in him and am trying very hard to imbibe in myself is to keep calm and smiling when hardware experiments fail (because they do, a lot) and to keep working sincerely and patiently to get the results. I would like to thank Dr. Ashwini Ratnoo, a collaboration with whom taught me a great deal about paper writing and how to look at your work from different perspectives. What I could learn from Dr. Ratnoo in a few months' collaboration, about having a zoomed in lens for the minute things in theoretical analysis and paper writing, I doubt I could have learnt anywhere else. I would also like to give a special mention of my labmate Mandeep Singh and thank him for working hard and enthusiastically with me on the field to complete the hardware experiments on time.

This thesis and my PhD itself would not have seen culmination if not for my family. My parents have given me the freedom to choose, the confidence to decide and a faith that no matter what, I have them watching my back. My gratitude to my family is incomplete without a very special mention for my father. He has been a friend, a confidante and the most enthusiastic listener of my PhD work. He has been there for me through all stressful times and also a part of all the joyous moments. I want to take this opportunity to express my heartfelt

gratitude to my family for making me what I am and being proud of it.

I believe no PhD journey can ever be successful without the support of some great friends and for me that support came from Anuprriya, Harsha and Ajay. Anuprriya has been the closest of my friends. She is my most cherished gift that IIIT has given me and has shared the ups and downs of my journey - happiness, sorrows-everything. She is also the one who has taught me how to be excellent at your work, while also enjoying your life to the fullest (I still have a long way to go before I master the art). Harsha is my buddy, critic and motivator. It is the endless cups of coffee and at least a lecture per day from him, that gave me a new perspective to look at everything from work to life. He has been the one who was there in everything - ranging from technical discussions and last-minute review of papers/presentations to being there for helping me with any and everything and partying (be it a joyous moment or to drown a sorrow) . Ajay is the one, who has always been there despite being miles away. He has made me believe that bond of friendship can develop even if you don't meet for years (something I never used to believe in). He has been there for everything ranging from hours long technical discussions (tutorials about anemometers) to my irrational stress filled moments and yet stood strong by my side through it all. His humility, calmness and ability to smile in even tough situations is something that I strive for in myself. My friends never lost belief in me even when I was full of self-doubt (which was a lot) and I can't thank them enough for being a part of this journey and my life.

My PhD journey and dive into the field of robotics would have been a lot more difficult had it not been for Parikshit. He has always given me valuable technical inputs, last minute reviews along with words of encouragement whenever I was in doubt (not to mention a long list of jokes). I would also like to thank my labmates in IIITD, and this is incomplete without a special mention Dheryta, Milan and my roommate Anupriya. They have been a source of constant support and motivation for me. Anupriya has become a dear friend in a short time and I have learnt a lot from them about working persistently and not giving up. I would also like to thank all my labmates at Texas A & M University for making my 6 months there full of fun and learning. I would also like to express my gratitude to Admin Staff, especially Priti and Sheetu. They are ever smiling and helpful people who have always helped in fast resolution of admin related matters.

Last but not the least, I would like to thank TCS research for selecting me as a recipient of TCS PhD research fellowship and providing financial support in the form of contingency funds and international travel support for conferences. I would also like to thank my reviewers for giving their valuable time to review this thesis.

Certificate

This is to certify that the thesis titled " Vision Assisted Guidance Techniques for Quadrotor Landing " being submitted by Alvika Gautam to INDRAPRASTHA INSTITUTE OF INFORMATION TECHNOLOGY DELHI, for the award of the DOCTOR OF PHILOSOPHY, is an original research work carried out under my supervision. In my opinion, the thesis has reached the standards fulfilling the requirements of the regulations relating to the degree.

The results contained in this thesis have not been submitted in part or full to any other university or institute for the award of any degree/diploma.

Dr. P.B Sujit and Dr. Srikanth Saripalli
DEPARTMENT OF COMPUTER SCIENCE ENGINEERING
INDRAPRASTHA INSTITUTE OF INFORMATION TECHNOLOGY DELHI

Abstract

Rotary vehicles like quadrotors, have become an essential part of many applications like mapping, object delivery, tracking, patrolling, and communication relay, etc. A key requirement for these applications is that the vehicle needs to perform these tasks autonomously from takeoff to landing. Landing is the most crucial task in the mission, which has a direct impact on the physical safety of the vehicle. Further, in mapping and object delivery applications, the vehicle landing area is constrained, and hence requires precision landing capability. Additionally, external disturbances like winds and visibility add another dimension to challenges in actual landing scenarios. This thesis focuses on the development of vision-assisted guidance techniques for landing of a quadrotor.

The primary concern of a guidance strategy is to enable persistent tracking of the landing pad (referred to as target) which could be either stationary or moving and accurately land on the target. Additionally, for a moving target, the velocity of the UAV and the target must be the same to achieve zero closing velocity at the time of landing. Besides this, it is desired that the convergence profile of the UAV velocity to the target velocity is slower in the beginning and faster towards the end for achieving time efficient landing. To this end, we have developed a novel log polynomial guidance for accurately landing the vehicle on the target, while meeting the required velocity characteristics. The target is persistently detected and the distance to the target is estimated using an onboard camera vision algorithm. Based on the estimated distance, the log polynomial guidance law drives the vehicle towards the target and ensures landing with zero closing velocity. We theoretically prove the convergence of the proposed guidance law. The guidance law is applicable to both stationary and moving targets. We have performed simulations and carried out field experiments to demonstrate the efficacy of the guidance law.

The proposed guidance law is analyzed in detail for a vertical landing scenario. The guidance law consists of a log polynomial function, which is dependent on the distance between the UAV and the target, and can be

controlled by two design parameters. We have carried out a detailed theoretical analysis of the impact of these design parameters on the guidance characteristics. We show that the decay characteristics of the log function, inherently guarantee zero closing velocity and acceleration at touchdown. Analytic conditions are deduced on the two design parameters for achieving, strictly decreasing, initially accelerating, and hovering velocity profiles, respectively. Closed-form expressions are derived for maximum velocity, and engagement time constraints as a function of design parameters. The results of the vertical landing scenario can be used to derive speed profiles of landing in different dimensions (x, y, z). Additionally, it provides a visual representation of the choice of velocity profile, maximum velocity constraint, and engagement time in two-dimensional design parameter space. These results help us in devising the landing behaviors for a robust landing and can give insights to the overall landing process (rather than just the terminal phase) to suit the needs of the application.

Any standard autonomous UAV mission is subject to hostile environmental conditions. Landing being an inherently difficult maneuver becomes more challenging due to environmental uncertainties like wind disturbance, uneven landing surface and constraints imposed by target localization errors, uncertain motion of the target and time critical nature of applications. Thus, it becomes imperative that the guidance strategy is robust and it degrades gracefully even in unforeseen conditions, and a decision to abort and/or reinitialize the landing process can be made accordingly. In order to understand these effects, we have carried out qualitative analysis of the log polynomial guidance law in simulations under varying environmental conditions and movement of the target. The simulations have been performed on a realistic quadrotor simulator, Microsoft Airsim with unreal engine. The robustness of the guidance law is further analyzed through, outdoor experiments. These studies show that the proposed guidance law is capable of guiding the vehicle to land accurately on any target, stationary and moving, in reasonably adverse environmental conditions.

In summary, this thesis focuses on developing a novel landing guidance law for a quadrotor using vision to land on stationary and moving targets under wind disturbance and uncertain target motion. We have shown convergence analysis and also experimentally validated the guidance principle through realistic simulations and outdoor hardware experiments.

List of publications

Journals

1. A. Gautam, A. Ratnoo, P.B. Sujit: Log polynomial velocity profile for vertical landing, AIAA Journal of Guidance, Control, and Dynamics, Vol. 41, No. 7, 2018, pp. 1617–1623.

2. A. Gautam, S. Saripalli and P.B. Sujit: Vision-assisted robust autonomous landing on moving targets, to be submitted to IEEE Transactions on Aerospace Electronic Systems.

3. A. Gautam, S. Saripalli and P.B. Sujit: Adaptive pure pursuit guidance based autonomous landing of quadrotor using vision, to be submitted to IEEE Transactions on Aerospace Electronic Systems.

Conferences

1. A. Gautam, P.B. Sujit and S. Saripalli: Vision based robust autonomous landing of a quadrotor on a moving target, International Symposium on Experimental Robotics, Buenos Aires, Argentina, 2018. Accepted

2. A. Gautam, S. Saripalli, P.B. Sujit and A. Ratnoo: Robust autonomous landing of a quadrotor on a moving target, 74th Annual AHS Forum, Phoenix, Arizona, 2018.

3. A. Gautam, P.B. Sujit and S. Saripalli: Autonomous quadrotor landing using vision and pursuit guidance, IFAC World Congress, Toulouse, France, 2017, 50 (1), 10501–10506.

4. A. Gautam, P.B. Sujit and S. Saripalli: Applicaton of guidance laws to quadrotor landing. International Conference on Unmanned Aircraft Systems, Denver, Colorado, 2015, pp. 372–379.

5. A. Gautam, P.B. Sujit and S. Saripalli: A survey of autonomous landing techniques for uavs, International Conference on Unmanned Aircraft Systems, Orlando, Florida, 2014, pp. 1210–1218.

Contents

List of Tables

List of Figures

Nomenclature

$\dot{\psi}$	Rate of change projection of LOS angle in xy plane
$\dot{\theta}$	Rate of change of LOS angle
a_z	Guidance command for the UAV in the vertical plane
χ_t	Target course angle in radians
χ_u	UAV course angle in radians
$\dot{R}_{xy}$	LOS rate in xy plane
$\dot{R}_z$	Rate of change of vertical distance
γ	UAV flight path angle in radians
ψ	Projection of LOS angle on xy plane in radians
ψ_f	Desired terminal impact LOS angle
LOS	Line of sight
θ	LOS angle in radians
a_{xy}	Guidance command for the UAV in the horizontal plane
K_r, K_g, K_b	Tuning parameters for RGB
R	LOS distance between UAV and Target in meters
R_{xy}^0	Initial LOS distance in xy plane (meters).
R_{xy}	LOS distance in xy plane (meters)
R_z	vertical distance between the UAV and the target
t	Target
u	UAV
V_t^0	Initial target speed in m/s
V_u^0	Initial UAV speed in m/s
V_t	Target speed in m/s
V_u	UAV speed in m/s

x_{t_t}, y_{t_t}	xy position of the target at time t
$x_{u_t}, y_{u_t}, z_{u_t}$	Position of the UAV at time T
xy	horizontal plane
z	vertical axis
EKF	Extended Kalman Filter
GPS	Global Positioning System
HAG	Height Above Ground
HSV	Hue-Saturation-Value
INS	Inertial Navigation System
KF	Kalman Filter
LQG	Linear Quadratic Gaussian
LQR	Linear Quadratic Regualtor
MAV	Micro aerial vehicle
PID	Proportional Integral Derivative
POSE	Position and Orientation of the Vehicle
PP	Pure pursuit
RGB	Red-Green-Blue
UAV	Unmanned Aerial Vehicle
VTOL	Vertical Takeoff and Landing Vehicle

Chapter 1

Autonomous UAV landing techniques

1.1 Introduction

Unmanned aerial vehicles (UAVs) are highly effective in remote operations. These vehicles have been used in several types of applications like surveillance[1], search[2], agriculture[3, 4] border patrol[5], scientific experiments [6], and mapping [7]. Communication, sensor and control techniques have evolved over the past few decades that has led to the development of a wide range of UAVs varying in shape, size, configuration, and characteristics. The common types of UAVs are fixed wing UAVs, quadrotors and helicopters at different scales (large UAVs or miniature vehicles or micro aerial vehicle). Fixed wing UAVs have a simple structure, fly at high speeds, and for a longer duration as compared to rotary wing UAVs. However, some of the fixed wing UAVs may require a runway for takeoff and landing, while those that can be either hand launched or through a catapult mechanism can be landed on their belly. On the other hand, rotary wing UAVs have an advantage of hovering, which is useful for monitoring some regions of interest. Rotary wing UAVs have agile maneuvering capability but at the same time they have high mechanical complexity, low speed and short flight range.

UAV flight consists of different phases, namely, take-off, climb, cruise, descent and finally landing. Most of the UAV autopilots have autonomous take-off (catapult and hand launched) and cruise but limited autonomous landing capabilities due to high risks and reliability issues. The accuracy of the landing must be high otherwise the aircraft may crash. Autonomous landing is one of the most challenging part of the flight. Landing must be done in a limited amount of time and space. Hence precise sensing techniques and accurate control is required during this maneuver.

There are two aspects to the design of a landing controller namely sensing and control. The choice and

design of these aspects must take into consideration a number of factors like: the type of landing (indoor or outdoor), visibility, type of terrain, wind disturbances, etc. The sensing techniques used can be either active or passive. Although radar, LIDAR, ultrasonic, and other time-of-flight active sensors can provide accurate range estimates, they are subject to several disadvantages. For example, surfaces and environmental factors can affect the ultrasonic range finder measurements. Most types of radars especially microwave radar which has a low frequency band does not provide sufficient angular resolution but it is insensitive to weather environment. On the other hand millimeter wave radar is smaller in size, provides fine angular resolution but it is limited to detection range and is sensitive to weather conditions. Also, by their nature, active sensors rely on self-emitted RF, laser or ultrasonic signals, which can be used to detect, or be used to localize by triangulation. Furthermore, active sensing modalities have relatively high rates of power consumption.

Visible-light or IR cameras, on the other hand are passive and rely solely on the reflected ambient light or heat, thereby lowering the probability of detection as well as reduced power consumption. Factors that make vision-in-the-loop difficult are: achieving the accuracy of active sensors without time-of-flight measurements; high bandwidth and computation subject to operation in real-time; and inherent photogrammetric ambiguities making some computations ill-conditioned. Therefore, selection on the type and the number of sensors to be used is a trade-off between accuracy, system complexity, power consumption and application scenario. These sensing techniques are used to estimate the POSE (position and orientation) of the UAV which is then used by the controllers. The type of controllers can range from simple linear control to complex techniques involving intelligent and hybrid control systems.

A general block diagram of the landing control system is shown in Figure 1.1. The system consists of four blocks namely sensors/navigation system, guidance controller, flight controller and type of the UAV. The sensors/navigation system determines the POSE of the vehicle. This information is used for the flight and guidance controllers. The guidance controller generates guidance commands like change in velocity, acceleration and rotation to follow a desired trajectory. The flight controller takes the guidance command to generate the appropriate actuation commands according to the type of the UAV (VTOL or fixed wing).

In this chapter, we review the existing work on landing technique for UAVs. We broadly classify the literature based on the type of sensors and controllers used for landing. These are vision-based, control-based, guidance-based and net recovery-based landing techniques and we also provide a detail comparison of these landing techniques.

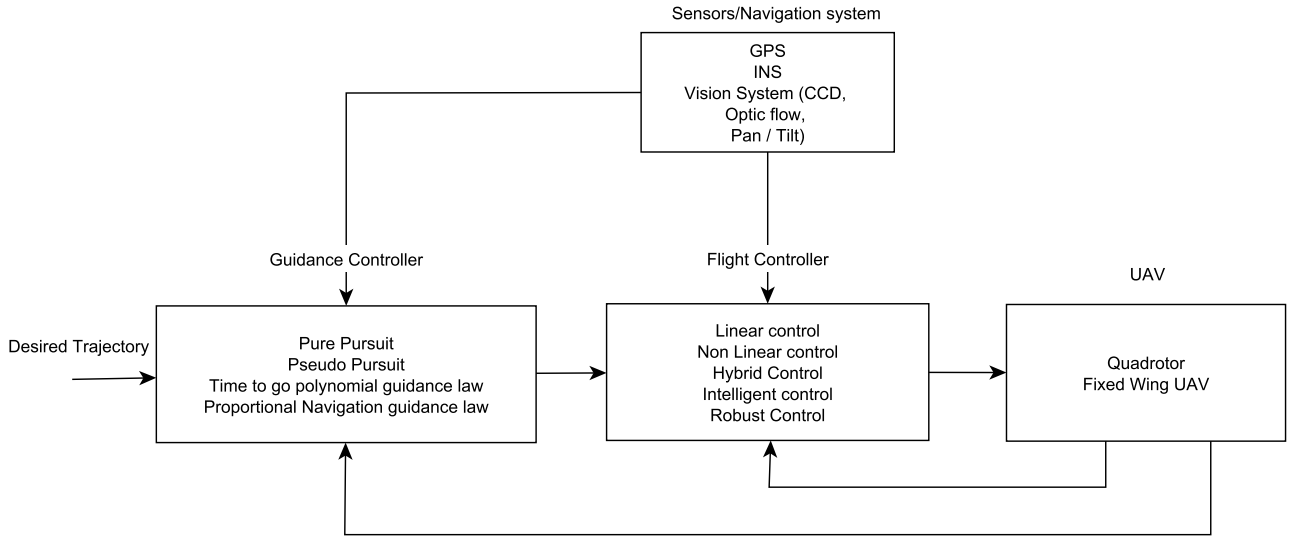


Figure 1.1: A generic landing control system consisting of Sensor/Navigation system, Guidance and Flight controllers along with the type of the UAV

1.2 Landing Site Selection

Emerging applications of unmanned aerial vehicles demand more autonomous capabilities to guarantee safe operation. Emergency landing situations due to power, actuation, or sensing failures may not include special navigation markers for landing. In such scenarios, it is essential to find a suitable landing site taking UAV flying characteristics, application scenario, and safety into account. Therefore, there is need to develop efficient algorithms that can detect flat zones [8, 9], and select a landing site quickly.

1.2.1 Site selection search

There are three steps in the site selection process: search, detection and determination of the best landing site. Park et al. [10] used a stereo vision system to detect primary landing sites and also store information about secondary candidate landing sites for a quadrotor. The system consisted of two cameras pointing downwards. Most appropriate landing site was decided by a decision map which was a weighted combination of three maps: evaluated depth map, flatness map and energy consumption map. Highest scoring sites were made as primary where as the others were stored as candidate landing sites. The weights of the parameters could be adjusted depending on the type and situation of landing to be performed. Three types of topographies were considered in the testing scenario: rough sinusoidal, mountain and urban. Brockers et al. [11] used a single camera for visual detection and reconstruction of the position of navigation targets but used a motion capture system for

scale information and closed loop control. Homography estimation was derived from matched feature points with the assumption of a planar surface which can be used for ego motion estimation. The navigation system operated in three sequential phases with three distinct behaviors, in order to first detect the target (detection phase), then determine an accurate estimate of the target location (refinement phase), and finally to perform an autonomous maneuver based on the final target estimate (descent and landing phase). In the detection phase, imagery data was collected for landing platform detection while the vehicle followed a predefined path. Once the first landing spot waypoint was generated, the control system transitioned to refinement phase where additional measurements were taken to increase the estimated waypoint's accuracy. This included collection of additional 3D waypoints in the vicinity of the current estimate. In the final phase the vehicle is first sent to a point at current altitude which was immediately above the landing waypoint and then it descended to land on the waypoint at an appropriate rate. Scherer et al. [12] studied the problem of evaluation of a landing site. The terrain evaluation was divided into two parts: coarse evaluation followed by a fine evaluation. The coarse evaluation found candidate areas applying a fast plane fit to measure the slope, roughness, and other statistics. The fine evaluation looked into skid contact and body clearance of a location and evaluated promising areas by fitting a 3D model of the helicopter and landing gear to a triangulated surface of the environment. Based on these evaluations, a weighted linear combination of factors determined the goodness of a landing site. These factors were: goodness from rough evaluation, area under the landing gears, volume under the helicopter, distance from the closest bad landing site and distance to the ground goal point. Depending on the values of the weights, different landing sites could be chosen as the best landing site.

1.2.2 Planning for site selection

A UAV needs to plan and perform a sequence of flight operations to select a landing site that minimizes the expected cost of flight, to the desired goal location. This requires planning under uncertain environments. Kushleyev et al. [13] proposed to solve the problem of computing this sequence using a probabilistic planning framework, called PPCP (Probabilistic planning with clear preferences) [14]. The PPCP algorithm decomposes into a series of deterministic graph searches that are fast-to-solve and require little memory. Theoretical guarantees on the optimality of the returned policies are also described. Griffith et al. [15] presented a system that plans nominal paths for a miniature aerial vehicle (MAV) through a terrain. A rapidly exploring random tree (RRT) algorithm was used to build a tree of traversable paths through an environment modeled using *apriori* data. Branches in the tree were then checked to ensure that they satisfy turn radius and climb rate constraints and are also collision free. While on course, the MAV avoided frontal threats via data from a forward-pointing laser, and used flow field information from three optic flow cameras to negotiate rough terrain. A hybrid path

planner capable of both single-mission path planning in known environments, as well as path re-planning in uncertain and dynamic environments was presented by Ping et al. [16]. Initially, the planner uses Dijkstra's algorithm to find a minimum cost path inside a 2-D grid representation of the environment. By using a grid map, sampling the C-space can be avoided, which in turn reduces the computational complexity. In the roadmap grid, UAV constraints, such as minimum flying altitude, as well as landing time were represented as cost functions that penalized the cost of the path to prevent the UAV from traversing an undesirable flight path.

1.2.3 Approach trajectory

The selection of a landing site and the trajectory followed to reach the site are application dependent. For example, in perch and stare application, the MAV should land close to the edge of the elevated surface, for emergency or fast landing tasks, a safer location is preferable [17]. For VTOL most of the work presented in the literature assumes that the selected landing site can be reached by using two waypoints. The first waypoint is a point directly above the landing zone and the second waypoint is the landing spot itself. For a fixed wing UAV, to accomplish a successful landing, there are three main factors which must be under control [18]. First of them is the lateral position of the UAV with reference to the runway since the goal is to touchdown on the lateral middle point of the runway. The second attribute is the vertical position, which is the AGL (above ground level) altitude of the UAV. It is a dynamic value for a fixed wing UAV, since it changes according to the distance to the runway. The last main attribute is the speed. The speed value is a static value and it depends on the aircraft characteristics. The main aim is to keep the desired speed value during the period of final approach.

1.3 Landing controllers

A typical landing system uses GPS (Global positioning system) and INS (inertial navigation system) along with range sensors like radar altimeter, or barometric pressure sensor, since the altitude measurement from the GPS in inaccurate. Using the fused information of the GPS, INS and the altitude, one can design efficient landing controllers for different types of aircrafts. We now present a review of landing controllers for quadrotors and fixed-wing aircrafts.

1.3.1 PID control

The proportional-integral-derivative (PID) controller is the most widely used feedback design. The control $u(t)$ is determines as

$$u(t) = K_p e(t) + K_I \int_0^t e(t)dt + K_D \frac{de(t)}{dt}, \tag{1.1}$$

where K_P is the proportional gain, K_I is the integral gain and K_D is the differential gain. Most of the aircrafts use the PID as a low-level control technique. The other control techniques including the vision-based control are high level techniques that provide signals to the PID controller. Erginer and Altug [19] presented a PD controller design for a quadrotor vehicle that has different PD control loops for controlling quad-rotor altitude, pitch angle, yaw and its motion. Quadrotor control using vision was also presented where the POSE was estimated using feature extraction technique. Lippiello et al. [20] proposed a PID based landing controller for a quadrotor for emergency landing. Ha and Kim [21] proposed an adaptive gain PID control technique using genetic algorithms for landing under harsh weather conditions.

1.3.2 Nonlinear control techniques

An aircraft model can be either a linearized or nonlinear aircraft model. In a linearized model, the longitudinal and lateral dynamics of the aircraft are decoupled which eases the design by developing separate controllers for longitudinal and lateral controllers. Nonlinear control techniques such as feedback linearization, sliding mode control and backstepping control designs are often used for nonlinear aircraft model [22].

1.3.2.1 Feedback linearization

Feedback linearization is a technique used for controlling nonlinear systems. It attempts to introduce auxiliary nonlinear feedback in such a way that the system can be treated as linear for the purpose of control design. Prasad and Pradeep [23] used this technique for landing control of a fighter aircraft. Voos and Nourghassemi [24] proposed a stabilized flight and landing strategy for quadrotor UAVs where they use the nonlinear feedback linearization technique to linearize and decouple three out of the six degrees of freedom. Burchett[25] applied feedback linearization to vehicle point mass dynamics to control the approach and landing of a reusable launch vehicle. This resulted in a linear system with inputs that were combinations of lift, drag and bank angle. By applying a simple aerodynamic model, lift and drag were mapped to negative z axis acceleration and speed-brake commands. Direct application of feedback linearization requires second and third order derivatives of

uncertain aerodynamic systems which does not guarantee stability. To overcome this, flight dynamics can be separated into slow and fast dynamics with sufficient timescale separation [26].

1.3.2.2 Sliding mode control

Sliding mode control technique is a nonlinear control technique that changes the nonlinear dynamics by application of a discontinuous control signal. In this technique, trajectories are forced to reach a sliding manifold in a finite amount of time and remain at that manifold for all future time. These trajectories in sliding mode control are defined as solutions to a set of sliding functions where the number of variables to track the trajectory should be equal to the number of available control inputs [27]. The main issue with sliding mode control is chattering and high control demand. Therefore, appropriate selection of sliding functions and reaching laws needs to be designed. Lee et al. [28] designed an adaptive sliding mode control to take into consideration the effect of sensor noise, uncertainty in the estimates and ground effect. Huang et al. [29] developed an integral sliding mode control to ensure the tracking is minimized to zero and the controller is robust to parameter uncertainties and external disturbances.

1.3.2.3 Backstepping control

Backstepping control is another nonlinear technique which can be used for designing a landing controller. The backstepping approach provides a recursive method for stabilizing the origin of a system in a strict-feedback form. In this system, the designer can start at a basic known stable system and "back out" new controllers that progressively stabilize each outer subsystem. For autonomous landing of a UAV the subsystems can be rotation and linear translation subsystem [30]. Using the backstepping approach one can synthesize the control law by forcing a system to follow a desired trajectory. Ahmed and Pota [31] presented an application of backstepping controller for landing of a quadrotor using a tether. This approach was extended in [32] where the backstepping-based controller takes advantage of the “decoupling" of the translation and rotation dynamics of the rigid body, resulting in a two-step procedure to obtain the vehicle control inputs. Lee and Kim [26] proposed a flight and landing control using backstepping along with neural networks where the backstepping controller tracked the angle of attack, side slip angle and roll commands assuming that aerodynamic model is completely known. Yoon et al. [33] proposed an adaptive backstepping controller design for aircraft landing with wind disturbance and actuator failures by the use of hedging techniques. Nonlinear six degree of freedom aircraft model was considered for the design of the backstepping controller that tracked a desired glide slope towards the runway. In order to estimate the modeling errors of aerodynamic coefficients in the nonlinear model, the

adaptive parameter estimation of the nonlinear function was adopted.

1.3.3 Intelligent control techniques

Intelligent control is a category of control technique that uses various artificial intelligence computing approaches like fuzzy logic, neural networks, and machine learning. Several landing controllers using these techniques have been proposed. However, the validation of these type of controllers on field experiments is limited.

1.3.3.1 Fuzzy logic

Fuzzy logic is a form of many valued logic. It handles the concept of partial truth, where, the truth value may range from completely true to completely false. Fuzzy control system is a system that makes use of fuzzy logic. It accepts analog continuous input values ranging from 0 to 1 instead of discrete values 0 and 1. The process of converting an input value to a fuzzy value is called “fuzzification" [34]. Fuzzy logic is used in the landing problem as it can accommodate nonlinearities due to aerodynamics, actuators, sensors, and environmental disturbances. Also, fuzzy logic controllers can be combined with conventional controllers like PID to model the system in a more realistic manner. Nho and aggarwal [35] developed a PD type of fuzzy logic controller and tested it on simulations using both linear and nonlinear models. Miguel et al. [36] presented a fuzzy logic based UAV landing using 3D position estimation.

1.3.3.2 Neural network-based control

Neural network-based control basically involves two steps: system identification and control. Neural networks have the ability to learn. Given a specific task to solve, and a class of functions $\mathfrak{F}$, learning means using a set of observations to find $f \in \mathfrak{F}$ which solves the task in some optimal sense. Malaek et al. [37] addressed the problem of designing an intelligent auto-landing controller in the presence of different wind patterns to expand the flight safety envelope. Four different types of controllers were designed namely, PID, neuro, hybrid neuro PID, and ANFIS-PID [38]. Neuro controller was designed to control the aircraft in glide and flare modes. Hybrid neuro controller was designed to handle the aircraft in very strong wind pattern. A controller based on vision and neural network was proposed in [39] where the pilot action was modeled using neural network. The constraint with most of the neural networks is that they are reliable when landing conditions fall within the range of trained data set, thus a significant amount of training data is required for sufficient reliability.

1.3.4 Hybrid control techniques

A system that exhibits both continuous and discrete behavior is a hybrid system. State of a hybrid control system is defined by a set of continuous variables and a discrete control mode. In order to perform an autonomous landing, a sequence of complex tasks must be performed, especially if there are obstacles on the runway. The sequence of actions represent discrete system while the actions during the sequence is a continuous system and hence landing problem can be treated as a hybrid system. Koo and Sastry [40] presented a hybrid control design for the landing problem by modeling the outer-inner loop of the vehicle as a hybrid system. A pack controller controlled the discrete state of the system based on the continuous state. The hybrid controller encoded the switching sequences for the phases in the landing scenario. Frazzoli et al. [41] proposed a hierarchical control architecture for maneuvering autonomous helicopters. In order to reduce the computational requirements of the control problem, a hybrid system framework was used. The hybrid controller was based on an automaton whose states represented feasible trajectory primitives. At the highest level, a strategic control layer was present, in which the current mission objectives were defined. Mission planning and way-point generation were typically included at this level. An intermediate layer was defined which included the guidance and trajectory planning. The best sequence of "maneuvers", or trajectory primitives was decided at this level. The lowest level, included the traditional control functions (stabilization, regulation, and command tracking).

1.3.5 Robust control techniques

Robust control methods explicitly deal with uncertainties in control design. A controller designed for a particular set of parameters and assumptions is said to be robust if it works well under a different set of assumptions or uncertainties. One of the robust control techniques used for UAV landing is the mixed H_2/H_∞ control technique [42]. H_2 control involves minimizing the H_2 norm of a system, while H_∞ method minimizes the effect of disturbance on the system performance. Liao et al. [43] proposed a robust fault tolerant controller that can handle external wind disturbances and control failures by employing H_2 control technique on a linearized aircraft model. Wang et al. [44] used this approach to address the landing problem of a flying wing UAV so that it tracks its landing trajectory even under the influence of uncertainties and disturbances. H_2 performance variables were formulated as an LQG problem to meet efficient dynamic responses and H_∞ was used to eliminate the disturbance due to ground effect and atmospheric disturbances. Lungu and Lungu [45] developed a longitudinal landing controller using H_2/H_∞ and dynamic inversion techniques. The dynamic inversion technique determines the guidance command while the H_2/H_∞ provides robustness against sensor noise and environmental disturbances.

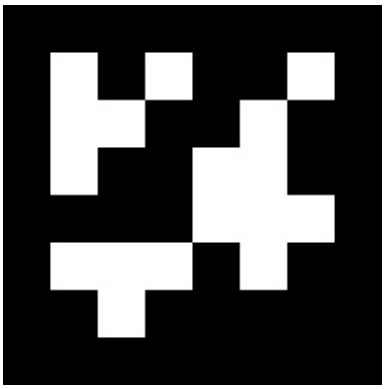


Figure 1.2: A sample ArUco marker

1.4 Vision-based Landing

The localization with GPS involves errors in meters (3-5 meters). Therefore, for precision landing in confined areas, it is essential to have additional information about the target position for accurate landing. One such sensor is camera, where we can use computer vision algorithms to acquire the target information which in turn can be for autonomous landing. There are a number of vision-based control techniques for heli-pad detection, tracking and landing (both indoor and outdoor). Classical vision-based target tracking and landing focuses on object recognition using edge detection techniques. Although vision provides a natural modality for object detection and landing, it can only sense the changes due to the applied forces not the forces themselves. Hence vision-based techniques are integrated with conventional control techniques for a robust landing design.

1.4.1 Indoor landing

Indoor landing involves landing in a controlled environment with less environmental disturbances as compared to outdoor landing. Wenzel et al. [46] use a Wii remote infrared camera for their visual tracking approach with the control algorithm running on an onboard microcontroller. They track a pattern of infrared spots by looking downwards with a fixed camera. The integrated circuit provides the pixel position of each spot at a high frequency. The estimated pose is used in various integrated PID control loops to control the vehicle motion. Sani et al. [47] uses ArUco markers as shown in Figure 1.2, along with inertial sensors to land on a stationary target. The markers enables the vehicle to localize accurately. Green et al. [48] use optical flow in an indoor environment to land vertically on a landing pad.

1.4.2 Outdoor landing

Outdoor Landing is a more challenging problem due to the presence of external disturbances such as wind, visibility, etc. The key component of any vision-based landing is to detect the heli-pad (also called as "target") using object detection or pattern recognition techniques. We will now describe some of the popular approaches to detect the target.

1.4.2.1 Image segmentation

It involves partitioning an image into sets of pixels and assign a label to every pixel in the image such that pixels with the same label share some visual characteristics. This can then be used to identify lines, object boundaries and curves in the image. Thresholding is one of the easiest methods of image segmentation that converts a gray scale image to a binary image.

1.4.2.2 Image moments

In image processing, image moment is the weighted average of the image pixels' intensities, or a function of such moments. These usually have some special characteristics and interpretation. They are generally used to describe objects after segmentation.

1.4.2.3 Monocular vision

It is a technique in which each eye is used separately unlike binocular vision thus increasing the field of view but limiting the depth perception. In this technique a single camera is used and hence suitable for aerial vehicles due to limited payload capability. Merz et al. [49] developed a navigation filter based on a Kalman Filter that was used to fuse highly accurate vision system estimates with inertial data provided by the onboard accelerometers and high rate gyros. This filtered out a large part of noise and outliers and also these filters made a smooth landing possible even when the vision system was blind. Daquan and Hongyue[50] proposed a vision-based navigation algorithm using Extended Kalman Filter (EKF) to estimate aircraft's POSE where the inputs to this were parameters like image gradients of centerline and threshold bar of runway lighting, longitudinal and lateral mean of the image coordinates of observed airport lights. Miller et al. [51] proposed a method to land using image registration. They used information about the terrain surrounding the runway from different scales and distances instead of the visual features of the runway itself. Geometric features are obtained which are

approximately linear indicators of the quantities to be measured. The course deviation of the UAV can be estimated from the camera model and registered image is used as an input to a linear feedback control loop.

1.4.2.4 Stereo vision

It is the process of extracting 3D information from images by using two cameras located along the same baseline. The information about a scene from two vantage points is compared, after which 3D information can be extracted by examining the relative positions of the objects. Pan et al. [52] used a combination of monocular and stereo vision to estimate approach angle and relative height of UAV with the use of Hough transform, RANSAC (Random Sample Consensus) [53] algorithm and vanishing line geometry. Sereewattana and Ruchanurucks[54] proposed depth estimation of markers for landing control using stereo vision. A single camera is used to capture two consecutive ground images for simulating a pair of stereo images. The markers consisted of four different colored circles. Their positions were extracted using Hough transform. The height between the UAV and the markers in world coordinates was then determined. Saripalli et al. [55] integrated vision with low level postural control to achieve precise autonomous landing of a helicopter. The vision algorithm consists of preprocessing, geometric invariant extraction, object recognition and estimation. The landing target is detected and extracted using thresholding and filtering technique. Invariant descriptors are calculated based on moments of inertia of the object. Height of the helicopter can be calculated using differential GPS and the pose of the heli-pad in helicopter body frame is calculated using x and y coordinates of the camera in image plane. The state estimates were sent to a behavior based hierarchical controller where the control problem was partitioned into a set of loosely coupled behaviors with each behavior being responsible for a task. This work was further extended in [56] where, the problem of tracking and landing on a moving target using an autonomous helicopter was addressed. The target was detected in a similar way as [55] and tracked using a Kalman filter by modeling the equations of the target as a linear system with the assumption that target moves only in one dimension. A variant of thresholding technique called adaptive thresholding was used by Lange et al. [57] to detect a landing pad and land a multirotor UAV using vision in GPS denied environments.

1.4.2.5 Egomotion

Egomotion is another technique that has been used for landing. Egomotion estimation involves estimating a camera's motion relative to a rigid fixed scene. Shakernia et al. [58] modeled the vision problem as a special case of egomotion estimation problem, where, the UAV position and velocity relative to the landing pad are estimated. Estimated motion and structure from vision are used in the control loop. In this case, the UAV was

commanded to track a fixed point at the desired configuration above the landing pad. Brockers et al. [11] also used egomotion to determine stabilization of the UAV and then for vertical landing.

1.4.2.6 IR imaging

Xu et al. [59] used infrared images to detect the heli-pad. The temperature difference between the target and background plays a key role in finding the object. To detect, a high emissivity black powder is spread on the object.

1.4.2.7 Optical flow

Optical flow is another mechanism that can provide information about the target. Barber et al. [60] addressed the autonomous landing problem for MAV (miniature/micro air vehicles). The height-above-ground (HAG) is determined using optic flow by relating the flow of features across an imaging array. The number of pixels that a given object moves in the imaging plane can be combined with MAVs, IMU and GPS data to determine HAG. The optic flow method was also used by Herisse et al. [61] to perform the landing of a VTOL on a moving platform.

1.5 Other sensing techniques for autonomous landing

Imagery can be used for target recognition and navigation from reference points when operating in unknown environments, as well as enhanced situational awareness. Also, many applications demand autonomous flight capability beyond the visual range; for example: information collection during natural disasters. Autonomous flight, landing, obstacle avoidance and self-charging ability might also be needed depending on the situation. Iwakura et al. [62] proposed the design of a movable range finding sensor system for precise landing of a quadrotor. The system consisted of one ultrasonic sensor and four IR sensors and the servo motors. The IR sensor was able to move within a 120 degree range using a servo motor for each sensor and it had a measuring range from 1m to 5.5m. These IR sensors were installed at each end of the vehicle legs. The ultrasonic sensor was attached on the bottom of the MAV, pointing downwards. The basic idea of position calculation in the absence of GPS using this design was to detect the edge of object. Each IR sensor, which was rotated by a servo motor, was used for edge detection.

Arora et al. [63] proposed a lidar based tracking system for landing on a moving ship deck in adverse lighting conditions where a camera might not be preferable. The authors argue that infrastructure free ship deck landing

in varying light conditions make lidar an obvious choice as a perception modality. This work proposed a particle filter based solution model. There has been work on radar based methods also for autonomous landing especially landing on the moving deck of a ship. These require heavy infrastructure on the ship and/or a data-link between the UAV and the ship deck.

1.6 Guidance-based landing

Guidance is a strategy that determines on how to follow the nominal path in the presence of off-nominal conditions, wind disturbances, and navigational uncertainties. Some of the guidance laws from missile literature have been borrowed for UAVs. Proportional guidance and pursuit guidance are two of the widely used laws for UAV guidance. Proportional guidance aims at maintaining a constant angle between the LOS and the target, whereas, pursuit guidance aims at generating commands to make the velocity vector of the UAV point towards the target.

The pursuit guidance can be further classified into pure pursuit and pseudo pursuit guidance laws. Pure pursuit guidance law leads the UAV towards a true target, while the pseudo pursuit guidance law generates a guidance command to track a virtual target. As the virtual target converges towards the true target, the UAV following the virtual target eventually converges to the actual target location [64]. Bang et al. [65] designed a guidance law for automatic landing of UAV using vision sensors for both fixed wing and rotary wing UAV. The method iteratively estimated a time-to-go until target intercept and modified the acceleration command based upon the revised time-to-go estimate. The time-to-go estimate depends on the position, velocity, and the acceleration of the vehicle and the target.

1.7 Recovery landing techniques for small UAVs

Traditional landing of a fixed wing aircraft undergoes phases like glide slope and flare which require a large area. To accurately determine landing coordinates high precision differential GPS and radar altimeter may be required which are costly compared to the cost of small UAVs. In addition, the small UAVs have limited payload capacity. Net recovery landing methods have been (in particular vision-based net recovery methods) proposed to land UAVs in restricted areas. Kim et al. [66] proposed a fully autonomous vision-based net recovery system for a fixed wing UAV. It consisted of a ground vision system whose aim was to detect a recovery net during landing and provide longitudinal and lateral bearing angle to the vehicle. These angles were calculated according to a guidance law based on the net position. Pursuit guidance law was used for longitudinal guidance

and a nonlinear pursuit guidance law for lateral guidance in order to land the UAV.

Another approach for net recovery is to divide the landing phase into spiral descent and final approach [67]. The aircraft aligns along the flight path angle towards the approach direction by the end of the spiral descent phase. The aircraft is guided from approaching waypoint to the recovery net using a pseudo pursuit guidance law. Approaching waypoints were generated using a cubic polynomial in the guidance law. Another landing technique in this category is "arrested landing" which involves catching a wire on board the ship with a deployed tail hook, bringing the aircraft to a quick stop [68]. Huh et al. [69] proposed a vision-based automatic landing system using a dome-shaped airbag for small UAVs. Color and shape based detection vision algorithms are applied for robust detection under varying lighting conditions due to dome's isotropic shape and distinctive color.

Airbag based recovery landing fares well than net based landing especially in the presence of cross winds. The isotropic shape of the dome allows the airplane to approach from any direction to avoid crosswind unlike net based landings [69].

1.8 Collision avoidance during autonomous landing

Some of the landing sites may have obstacles during the approach which the UAV needs to avoid. Therefore, given a selected landing site and coordinates of the goal, planning an optimal collision free trajectory to land at the site is equally essential. Gageik et al. [70] proposed a simple approach for obstacle detection and collision avoidance of a quadrotor using ultrasonic sensor and data fusion. The proposed solution consisted of two independent modules for obstacle detection and collision avoidance. Obstacle detection module used redundant ultrasonic sensors to increase detection resolution and reliability. Collision avoidance approach consisted of dividing the area around quadcopter into three zones: Far zone, close zone and dangerous zone. The collision avoidance module was implemented as a state machine having four states which were activated depending upon whether any obstacle was detected or not and to which of the above three mentioned zones it belonged to. In each of the state machine pitch roll angle and speed were controlled using a PID loop depending upon the distance and direction of the obstacle. Scherer et al. [71] presented first results on autonomous landing and obstacle avoidance for a full-scale helicopter. The main component system design consisted of a 3D LADAR scanning system operating in two modes: Forward direction for obstacle detection and downward direction for safe landing zone search from a high altitude. Obstacle avoidance approach used evidence grid representation [72] to register obstacles separately. The evidence grid representation turns wide angle range measurements from mobile robot-mounted sonar sensors into detailed spatial maps. It accumulates diffuse

evidence about the occupancy of a grid of small volumes of nearby space from individual sensor readings into increasingly confident and detailed maps of a robot's surroundings. The planned trajectory was compared to a C-Space expansion of the evidence grid and a new path was planned if the trajectory intersected with an obstacle. LIDT and D* lite algorithms [73] were used to create a path around obstacles, and then the resulting waypoint paths were forward-simulated to ensure that they did not intersect the obstacle. Other approaches for obstacle detection and avoidance use 3D camera systems like the Kinect camera from Microsoft or laser scanners [74]. The disadvantage of these is that any optical sensor is sensitive to light and such environment. Therefore smoke, steam and every gas which absorbs light will cause optical sensor systems to fail.

1.9 Comparison of landing strategies

In the above sections, several types of landing controllers using various sensors have been designed. We have captured the main characteristics of different landing techniques developed in the literature in Table 1.9. The table summarizes the characteristics in terms of the type of UAV, the employed landing technique and their features. Using the information provided in this table, one can select a landing technique for a given type of unmanned aircraft and environment in which they would like to use it.

Vision-based techniques are popular due to low payload weight and computationally cheaper algorithms to detect the landing pad. Due to their wide usage, we have provided a detailed comparison of different vision-based techniques in Table 1.2. The table provides information on the type of UAV, sensors used for landing, the technique developed along with assumptions and whether the evaluation was performed in simulation or with real-world experiments.

1.10 Contribution of this thesis

From the above literature review, it is evident that vertical landing is the preferred mode of landing maneuver in quadrotors. However, this approach is inefficient in terms of the time taken to land. Also, there is no study on how a user can modify the velocity of the quadrotor to obtain timely landing. These studies have been limited to designing vertical landing controllers in the presence of disturbances. Further, the number of landing experiments conducted to study robustness of the proposed system with respect to occlusion and wind disturbances are limited in the literature. This thesis focuses on addressing these issues for vision assisted quadrotor landing. The main contributions of this thesis are

Ref.	Controller Details	UAV type	Additional Characteristics	System Details
[19], [35]	PID[19], PD with fuzzy logic[35]	VTOL	None	• 6DOF simulations • Pitch, roll and altitude control
[23], [24], [25]	Nonlinear Feedback Linearization	Fixed Wing(F-16 Model)[23] [25], VTOL [24]	Actuator dynamics and aerodynamic drags	• 3DOF aircraft Model • Ground effect is studied
[27]	Sliding Mode Control	Fixed Wing(HARV Model)	None	• 6DOF Nonlinear Model • Simulations only • Lyapunov stability criteria • Comparison with PID controller
[26], [31], [32]	Backstepping and Neural Networks	Fixed Wing (F-16 model)[26], Rotary UAV [31] [32]	aerodynamic model uncertainties	• 6DOF Model • Backstepping control • Adaptive Neural Networks for aerodynamic modeling error under uncertainties • Flapping correction and servo dynamics
[40]	Hybrid Control	Simulated for a general UAV	Runway traffic	• Landing as a sequence of tasks • Switching strategy between controllers • Design Correctness analyzed by a reachability computation

Table 1.1: Comparison of control based landing techniques using GPS

- Development of a novel log polynomial velocity controller that ensures zero closing velocity at touch down. The velocity controller is integrated with a pursuit guidance law and onboard vision-based landing pad detection system. We have evaluated the proposed methodology using simulations and hardware

experiments on a 3DR IRIS quadrotor for landing on a stationary and moving target.

- We have theoretically analyzed a simplified version of the log polynomial velocity controller to obtain closed form solution for landing time. Also, we analyzed the effect of varying the velocity controller parameters – a and b, to obtain different velocity profiles for landing. This study can be used in selecting the desired velocity profile for landing.

- We developed a new compensated trajectory shaping guidance law to enable the quadrotor to land on the maneuvering target and on inclined surfaces. The guidance law is integrated with the log polynomial function based velocity controller and its performance is evaluated using simulations and experiments. Further, we performed several experiments to study the robustness of the landing system in the presence of different target maneuvers, wind conditions and landing surface inclination.

Ref	UAV Type	Equipment	Techniques	Assumptions	Testing
[55], [56]	VTOL	1. Downward CCD camera 2. ultrasonic sonar and INS	1. Invariant moments 2. Kalman filter based tracking	1. Helicopter is in Hover 2. Target tracking in one dimension	Flight Tested
[60], [61]	MAV in [60], VTOL in [61]	1. Optic flow sensor 2. IMU, GPS 3. Barometric pressure sensor	Optic flow sensor and barometric altimeter based HAG estimation	None mentioned	Tested in different wind conditions
[49]	VTOL	1. CCD camera mounted on a PTU 2. Accelerometer and angular rate gyros	1. Contour extraction 2. Kalman filter fusion	Special landing pad with reference Pattern	Flight tested with wind (up to 10m/s).
[46]	VTOL	1. Wii Remote IR camera 2. IMU, GPS and compass module, pressure sensor	Pattern analysis to retrieve pose from the camera and IMU	Indoor landing with accurate roll and pitch estimation	Tested with hardware .
[59]	Not specified	1. Black object on the landing target 2. Thermal imager, GPS and INS	1. Recognition from infrared images 2. Affine moment invariants	'T' model based landing target	Simulations
[52], [54], [50]	Not specified	Two camera sensors	1. Monocular, stereo, and machine vision 2. EKF, Hough transform, RANSAC algorithm	Markers used on heli-pad in [54]	Simulated results

Table 1.2: Comparison of vision-based landing techniques

Chapter 2

Quadrotor landing using vision and pursuit guidance

Vision assisted quadrotor landing at a desired location involves two steps: (a) detection of the target and (b) generating control commands to the vehicle such that it lands accurately on the target. In the vehicle, fused INS and GPS information is used for localization, however, the altitude measurement from GPS is inaccurate and hence additional sensors like radar altimeter, ultrasound or barometric pressure sensor are used for altitude measurement [75][76][77]. The GPS-based measurement has errors between 3-5 meters [78] and the vehicle may not land at the desired location with only GPS-based navigation. In this chapter, we use camera vision to provide information about the target location in the form of a landing pad with distinct color as shown in Figure 2.1. By estimating the location of the target, a guidance command is generated to the vehicle for landing as shown in Figure 2.2.

Landing controllers can be implemented in two ways. In the first approach, a controller is desired that directly determines the control signals for the vehicle motors. In the second approach, a successive loop closure method is employed, where the low level controller uses PID controller and the guidance command is generated at a higher level. The first approach is employed for high maneuvering trajectories. In this chapter, we use the second approach, as it is flexible and offers finer control on each module of the system without interfering with the working of inner PID control loops of the UAV. Since we are using guidance loop, we need to ensure that at touch down the closing velocity of the target to the vehicle is zero. Otherwise, the vehicle may have some physical damage due to impact.

Most of the landing techniques using vision that are proposed in the literature have been limited to vertical

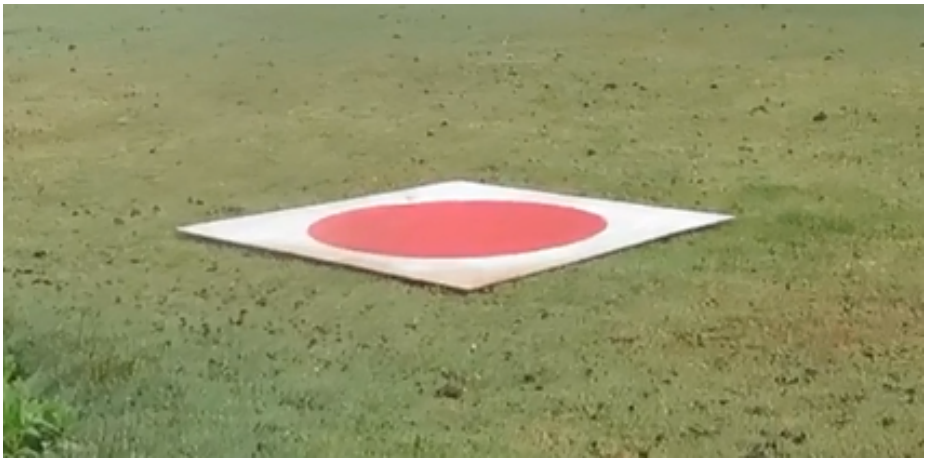


Figure 2.1: The target for landing is a red colored circle which is drawn on a wooden sheet.

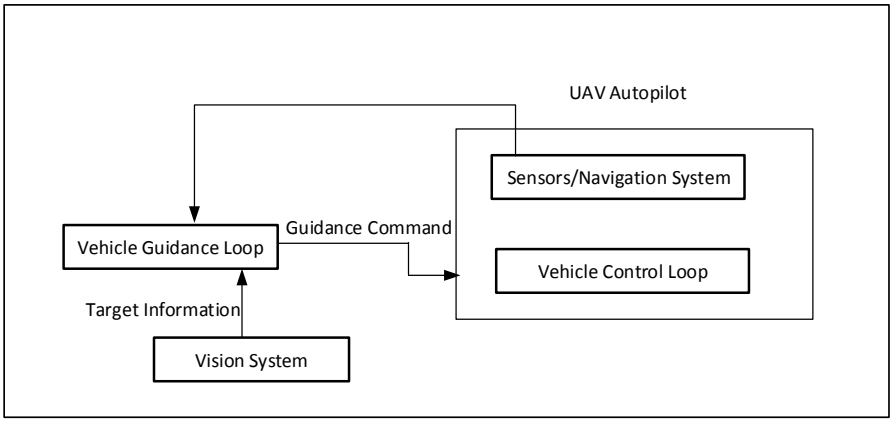


Figure 2.2: Block diagram of a guidance based landing system.

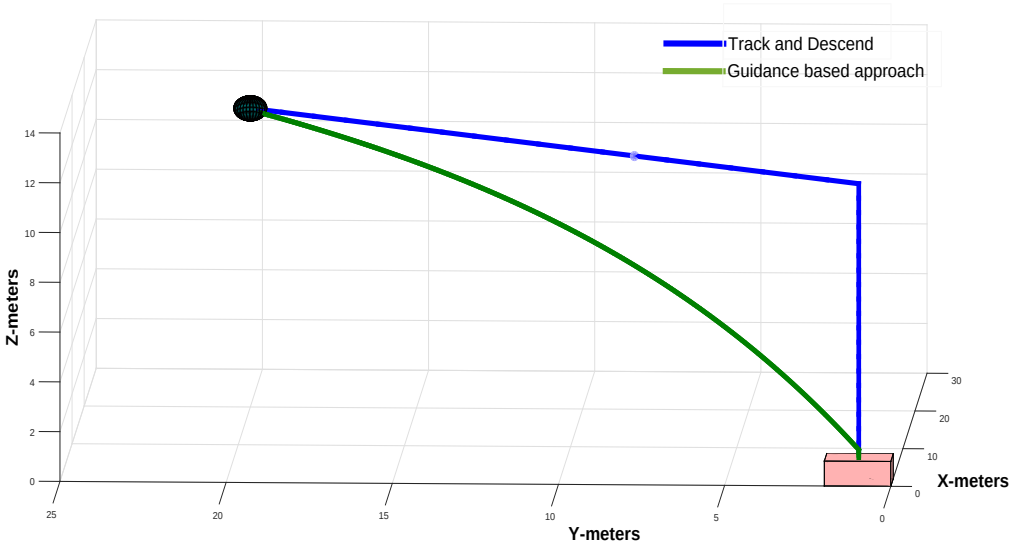


Figure 2.3: Trajectory of typical landing of a quadrotor and the proposed landing using pure pursuit guidance law.

landing approach only. In such approaches, quadrotor tracks the target while moving along a linear path until it is above the target and then descends vertically to land as shown in Figure 2.3. Although, the approach is simple, the resultant trajectory is inefficient in terms of increased flight time during the landing process.

In this chapter, we propose a pure pursuit based guidance law to land on a stationary as well as a moving target. Using the guidance commands, the quadcopter approaches the target while simultaneously descending in altitude, thus following a more efficient trajectory as compared to track and descend approach. In order to ensure zero closing velocity for landing, we develop a novel log polynomial function based velocity controller. We evaluate the efficacy of the guidance law extensively in simulation and validate the performance through outdoor field experiments on a 3DR IRIS quadrotor platform carrying a camera and an on-board computing unit. Since we are using several techniques for the landing, we will provide brief description of these approaches before using them. We will first describe basics of guidance laws, then we describe the target detection algorithm using the camera vision, followed by the novel log polynomial velocity controller and evaluation of the landing system in simulation and outdoor experiments.

2.1 Basics of Pure Pursuit (PP) guidance

In this section, we provide basics of 2D and 3D pure pursuit guidance laws for completeness and then use these guidance laws for landing in the subsequent sections.

2.1.1 2D Pure Pursuit (PP) Guidance

Consider a planar UAV-target engagement geometry as shown in Figure 2.4. The speed of the UAV and the target are V_u and V_t respectively. The notation χ_u, χ_t, ψ, and R_{xy} represent the UAV course angle, target course angle, line-of-sight (LOS) angle, and range separation respectively. The target is assumed to be non maneuvering, that is χ_t is constant. Using the engagement geometry in Figure 2.4, $\dot{R}_{xy}$ and $\dot{\psi}$ are calculated as,

$$\dot{R}_{xy} = V_t \cos(\chi_t - \psi) - V_u \cos(\chi_u - \psi), \tag{2.1}$$

$$\dot{\psi} = (V_t \sin(\chi_t - \psi) - V_u \sin(\chi_u - \psi))/R_{xy}. \tag{2.2}$$

The guidance command, denoted as a_{xy} is the lateral acceleration applied normal to the velocity vector. The

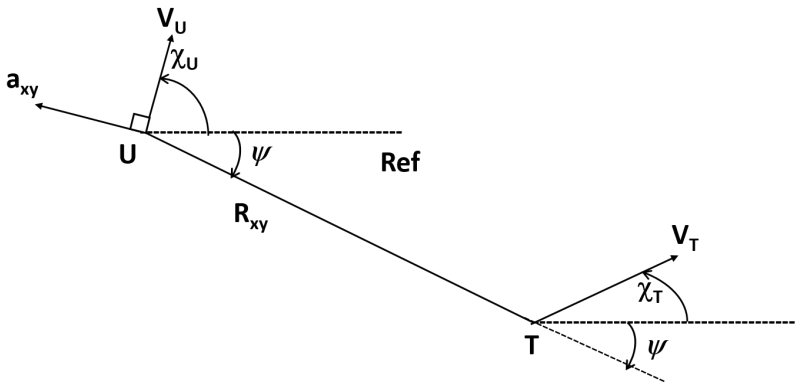


Figure 2.4: Planar engagement geometry of UAV-target

guidance command is expressed as,

$$\dot{\chi}_u \quad = a_{xy}/V_u. \tag{2.3}$$

Pure pursuit (PP) works on the principle, that if the tracking vehicle persistently points towards the target then it will ultimately intercept it [79]. In other words, PP aligns χ_u towards ψ. A practical implementation of the pure pursuit uses a feedback law as,

$$a_{xy} \quad = -K_a(\chi_u - \psi), K_a > 0, \tag{2.4}$$

where K_a is the gain. However, for PP to be effective, it is important to maintain $\dot{\chi}_u = \dot{\psi}$, so that the vehicle can point towards the target at all times. Combining equation (2.3) and (2.4), the guidance command for simple pure pursuit is given as,

$$a_{xy} \quad = V_u\dot{\psi} - K_a(\chi_u - \psi). \tag{2.5}$$

2.1.2 3D Pure Pursuit guidance

An actual landing process occurs in three dimensional space. We extend the 2D pursuit guidance principle to 3D scenario. We consider the target moving on the ground represented by XY horizontal plane. The UAV starts from some position (x_0, y_0, z_0) assuming that the target is visible from this initial location.

The parameters to be considered are the course and flight path angles of the UAV. Expressions for the

distance between UAV and target and these angles are given as,

$$R = \sqrt{(x_t - x_u)^2 + (y_t - y_u)^2 + (z_t - z_u)^2}, \tag{2.6}$$

$$R_{xy} = \sqrt{(x_t - x_u)^2 + (y_t - y_u)^2}, \tag{2.7}$$

$$R_z = z_t - z_u, \tag{2.8}$$

$$\psi = \tan^{-1}(y_t - y_u / x_t - x_u), \tag{2.9}$$

$$\theta = \tan^{-1}(z_t - z_u / R_{xy}). \tag{2.10}$$

Figure 2.5 shows the engagement geometry for a UAV landing on the target in a 3D scenario. Expressions of $\dot{R}_{xy}$, $\dot{R}_z$, $\dot{\psi}$ and $\dot{\theta}$ are derived similar to the 2D case.

$$\dot{R}_{xy} = V_t \cos(\chi_t - \psi) - V_u \cos(\theta) \cos(\chi_u - \psi), \tag{2.11}$$

$$\dot{R}_z = V_u \sin(\theta), \tag{2.12}$$

$$\dot{\psi} = (V_t \sin(\chi_t - \psi) - V_u \cos(\gamma) \sin(\chi_u - \psi))/R_{xy}, \tag{2.13}$$

$$\dot{\theta} = -(V_u \sin(\gamma) \cos(\theta) + \dot{R}_{xy} \sin(\theta))/R. \tag{2.14}$$

Guidance command in 3D is given by,

$$a_{xy} = V_u \cos(\chi_u)\dot{\psi} - K_a(\chi_u - \psi), \tag{2.15}$$

$$a_z = V_u\dot{\theta} - K_a(\gamma - \theta), \tag{2.16}$$

$$\dot{\chi}_u = a_{xy}/V_u \cos(\gamma), \tag{2.17}$$

$$\dot{\gamma} = a_z/V_u. \tag{2.18}$$

We assume the target is non-maneuvering with constant velocity and heading. The kinematic model for the target and UAV are given as

$$x_{t_{t+\Delta t}} = x_{t_t} + V_t \cos(\chi_t)\Delta t, \tag{2.19}$$

$$y_{t_{t+\Delta t}} = y_{t_t} + V_t \sin(\chi_t)\Delta t, \tag{2.20}$$

$$x_{u_{t+\Delta t}} = x_{u_t} + V_u \cos(\gamma) \cos(\chi_u)\Delta t, \tag{2.21}$$

$$y_{u_{t+\Delta t}} = y_{u_t} + V_u \cos(\gamma) \sin(\chi_u)\Delta t, \tag{2.22}$$

$$z_{u_{t+\Delta t}} = z_{u_t} + V_u \sin(\gamma)\Delta t. \tag{2.23}$$

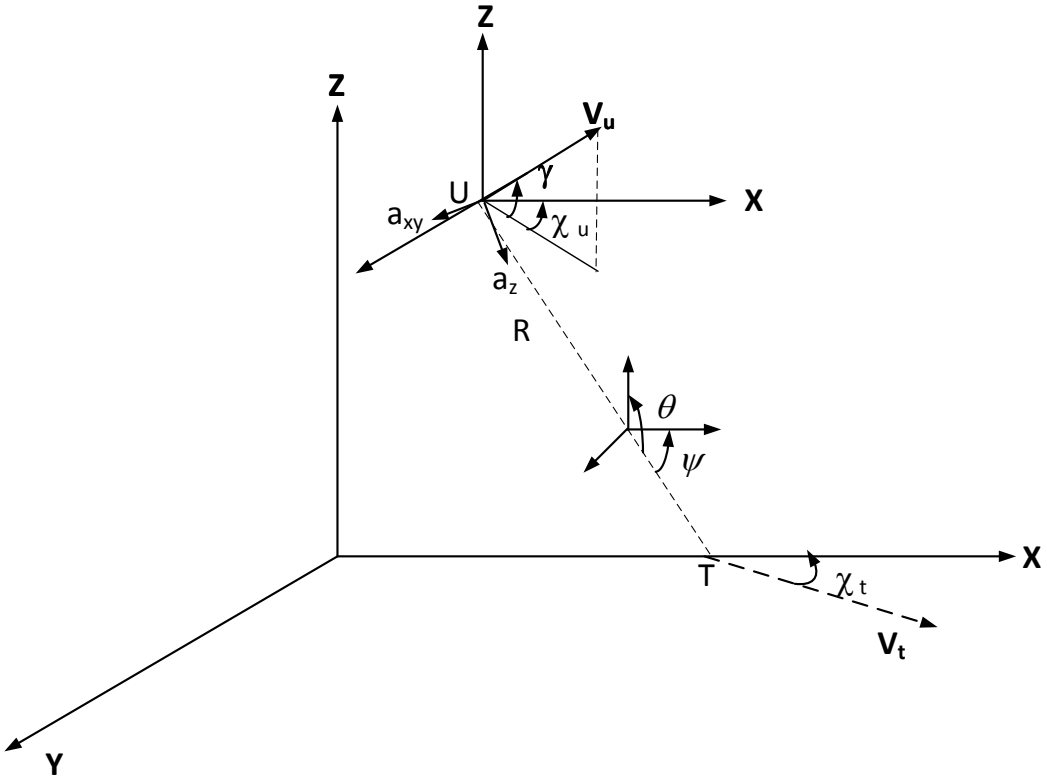


Figure 2.5: 3D engagement geometry for a UAV landing on a moving target

2.2 Landing pad detection using vision

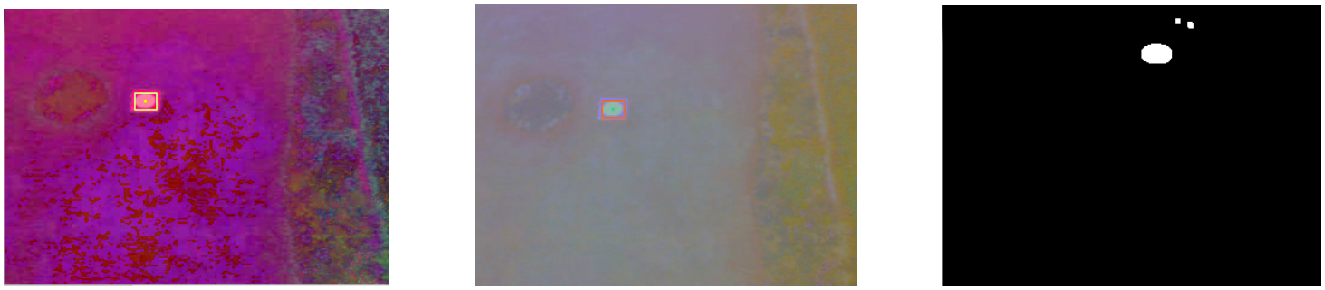
Our focus is to design an autonomous landing controller; hence we select a target in the form of a red colored circle with known diameter. The selected landing pad pattern can be detected using blob detecting algorithm which is computationally cheap.

The obtained RGB image was converted to HSV colour space due to its robustness to change in lighting conditions. Figure 2.6(a) shows the obtained image. Hue defines the dominant color of an area, saturation measures the colorfulness of an area in proportion to its brightness. The value is related to the color luminance. These parameters are calculated as follows,

$$H = \frac{\arccos(1/2((R-G)+(R-B))}{\sqrt{(R-G)^2+(R-B)(G-B)}}, \tag{2.24}$$

$$S = 1 - 3\frac{\min(R,G,B)}{R+G+B}, \tag{2.25}$$

$$V = \frac{R+G+B}{3}. \tag{2.26}$$



(a) (b) (c)

Figure 2.6: a) HSV color space mask b) YCrCb color space mask c) Combined mask after thresholding

Next, the original RGB image was converted to YCrCb color space. Figure 2.6(b) shows the obtained image. In this space, color is represented by *luma*, constructed as a weighted sum of RGB values, and two color difference values Cr and Cb that are formed by subtracting *luma* from RGB red and blue components. The conversion equations are given as,

$$Y = K_r R + K_g G + K_b B, \tag{2.27}$$

$$Cr = R - Y, \tag{2.28}$$

$$Cb = B - Y, \tag{2.29}$$

where K_r, K_g, and K_b are the tuning parameters associated with the RGB colors. The union of the two color space masks was taken and morphological operations were applied to remove the noise, while preserving the mask boundaries. Erosion was performed using a 8 x 8 rectangular kernel, followed by dilation using a 3 x 3 rectangular kernel. Figure 2.6(c) shows the obtained mask with multiple contours. The target is detected by extracting the largest contour in the scene.

2.2.1 Target state estimation using Kalman Filter

During the landing process, the target may be partially or fully occluded or may move out of the field of view. Hence, we use Kalman Filter (KF) to estimate the target parameters, which is robust to environmental changes.

The detection algorithm determines the x and y centroid pixel coordinates of the target, which is given as the input to the KF. The target is modeled in discrete time as,

$$X_t(n) = F_t X_t(n-1) + w(n), \tag{2.30}$$

where, $w(n)$ is the random process noise and the subscript n denotes the current time and $n-1$ denotes the

previous time instant. X_t is the state vector comprising of target characteristics: centroid pixel coordinates (x_t, y_t), rate of change of pixel coordinate positions ($\dot{x}_t$,$\dot{y}_t$), dimensions of the detected target contour ($width$ (w_c), $height$ (h_c)). F_t is the state-transition model that is applied to the previous state $X_t(n-1)$. We measure the position of the target using target detection (using vision). Measurement at time n is given by,

$$Z_t(n) = H_t X_t(n) + v(n), \tag{2.31}$$

where, H_t is the measurement model and $v(n)$ is the random measurement noise in landing pad parameter measurements. Both process and measurement noise are modeled as white, zero mean Gaussian noise with covariance matrices Q and R respectively. The formulation of the Kalman Filter is given as follows,

$$\begin{bmatrix} x_t(n) \\ y_t(n) \\ \dot{x}_t(n) \\ \dot{y}_t(n) \\ w_{c_n} \\ h_{c_n} \end{bmatrix} \begin{bmatrix} 1 & 0 & dt & 0 & 0 & 0 \\ 0 & 1 & 0 & dt & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_t(n-1) \\ y_t(n-1) \\ \dot{x}_t(n-1) \\ \dot{y}_t(n-1) \\ w_t c_{n-1} \\ h_t c_{n-1} \end{bmatrix}. \tag{2.32}$$

The prediction step of the Kalman Filter is given by,

$$\hat{X}_t^-(n) = F_t \hat{X}_t(n-1), \tag{2.33}$$

$$P_t^-(n) = F_t P_t(n-1) F_t' + Q, \tag{2.34}$$

where $\hat{X}_t^-(n)$ is the predicted state estimate. The update step of the Kalman Filter is given by,

$$K_t(n) = P_t^-(n) H_t' \left(H_t P_t^-(n) H' + R\right)^{-1}, \tag{2.35}$$

$$P_t(n) = \left(I - K_t(n) H_t\right) P_t^-(n), \tag{2.36}$$

$$\hat{X}_t(n) = \hat{X}_t^-(n) + K_t(n) \left(Z_t(n) - H_t \hat{X}_t^-(n)\right). \tag{2.37}$$

In the above equations, superscript $'$ denotes the matrix transpose. K_t is the Kalman gain, P_t is the predicted error covariance and Q and R are the covariance matrices for process and measurement noise respectively. The estimated parameters of the target are given as input to the guidance loop.

2.3 Pure pursuit based autonomous landing

The guidance objective is to land the vehicle smoothly on the target, while persistently tracking it. Pure pursuit (PP) aims at keeping the velocity vector of UAV directed towards the target which is also one of the goals for landing and moreover it is simple to implement.

The simple pure pursuit guidance law given in equation (2.5) does not constrain the UAV-target closing velocity. It is the velocity with which the tracking vehicle closes on to the target and is given as negative of range rate, that is, $-\dot{R}$. As the UAV approaches the target, closing velocity between UAV and the target should asymptotically decrease for a safe and precise landing. To achieve this, it must satisfy the following condition

$$\lim_{R \to 0} -\dot{R} \to 0. \tag{2.38}$$

The relative velocity between the UAV and the target must be zero at the time of landing. Additionally, velocity of UAV must be relatively high for significant part of the flight to approach the target in a time efficient manner. To achieve both these properties, we propose a novel log of polynomial [80] function to control the vehicle velocity. The log of polynomial function is given as,

$$f(x) = N^{log_n(1+ax+bx^2+cx^3)}, \tag{2.39}$$

$$x = R_{xy}/R^0_{xy}. \tag{2.40}$$

subject to,

$$1 + a + b + c = n. \tag{2.41}$$

Theorem 1. *Let f(x) be a log of polynomial function of the ratio of current to initial line of sight distance (R_{xy}/R^0_{xy}). Then, the closing velocity ($-\dot{R}_{xy}$) exponentially decreases to zero as the distance goes to zero if $V_u = f(x)V_t$.*

Proof. In PP, the aim is to continuously point towards the target and the guidance command is aimed at maintaining χ_u=ψ.
Therefore equation (2.1) reduces to

$$\dot{R}_{xy} = V_t \cos(\chi_t - \psi) - V_u. \tag{2.42}$$

As, the UAV approaches the target, $R_{xy} \to 0$ and $\chi_t \to \psi$, reducing equation (2.42) to

$$\lim_{R_{xy}\to 0} \dot{R}_{xy} = V_t - V_u. \tag{2.43}$$

The variation in velocity is given as,

$$V_u = V_t f(x), \tag{2.44}$$

where,

$$N = V_u^0/V_t^0, \tag{2.45}$$

$$x = R_{xy}/R_{xy}^0. \tag{2.46}$$

Initially, R_{xy}=R_{xy}^0, hence x=1. Therefore, from equations (2.40) and (2.44):

$$\log_n(1 + ax + bx^2 + cx^3) = 1, \tag{2.47}$$

$$V_u = NV_t \;\; if \;\; R_{xy} = R_{xy}^0. \tag{2.48}$$

As UAV approaches the target and $R_{xy} \to 0$, $x \to 0$ and $log_n(1 + ax + bx^2 + cx^3) \to 0$. As a result,

$$\lim_{R_{xy}\to 0} V_u \to V_t, \tag{2.49}$$

$$\lim_{R_{xy}\to 0} \dot{R}_{xy} \to 0. \tag{2.50}$$

□

The velocity of the UAV is a log polynomial function of the distance between the vehicle and the target. This variation of the UAV velocity based on distance provides us a knob which can be modified during different phases of the flight. Besides achieving the desired closing velocity conditions, log of polynomial function has other desirable properties of the UAV flight as,

1. It has a slower decay for most of the flight which makes sure that velocity of the UAV is significantly higher than the target.

2. Faster decay towards the end which quickly drives the closing velocity to minimum as UAV approaches the target.

In 3D, $-\dot{R}_{xy}$, ψ reflect the velocity with which the UAV approaches the target and its alignment with the target. $-\dot{R}_z$ reflects the velocity with which the UAV descends. Theorem 1 can be extended to a 3D scenario to drive both $\dot{R}_{xy}$ and $\dot{R}_z$ to minimum.

Lemma 2. *Let f(x) be a log of polynomial function of the ratio of current to initial line of sight distance (R^0/R). Then, the closing velocity in xy plane and z direction ($-\dot{R}_{xy}$ and $-\dot{R}_z$) exponentially decreases to zero as the distance goes to zero if $V_u = f(x)V_t$.*

Proof. At the time of landing, as $R \to 0$, θ becomes very small. Taking a small angle approximation, equation (2.10) reduces to,

$$\dot{R}_{xy} = V_t \cos(\chi_t - \psi) - V_u \cos(\chi_u - \psi). \tag{2.51}$$

Also, as UAV approaches the target $\chi_t \to \psi$ and $\sin(\theta) \to 0$. Therefore, as $R \to 0$,

$$\dot{R}_z \to 0. \tag{2.52}$$

$$\dot{R}_{xy} \to V_t - V_u. \tag{2.53}$$

As $R_{xy} \to 0$, $\dot{R}_{xy}$ given by equation (2.53) approaches to 0 as explained in theorem 1. □

2.4 Simulation results

Consider a scenario as shown in Figure 2.7, where the UAV lands on a target using two approaches – our proposed method and the track-and-land method. For the simulation, the initial UAV velocity is 2 m/s, the target velocity is 1 m/s, R^0_{xy} is 40 meters and the simulation is terminated when $R_{xy} \leq 0.1$ meters. From the simulations, the vehicle takes 45 seconds to land on the target using our proposed method, while the track-and-land approach takes 72 seconds. This shows the superior performance of the proposed guidance-based landing in terms of time to land for a simple scenario. It is evident that using the proposed method UAV lands quickly on the target.

In the subsequent subsections, we present the results of guidance based landing method for a 3D scenario in MATLAB, followed by the results obtained in ROS using a realistic quadrotor simulator which takes quadrotor dynamics into account. Then, the experimental results are presented.

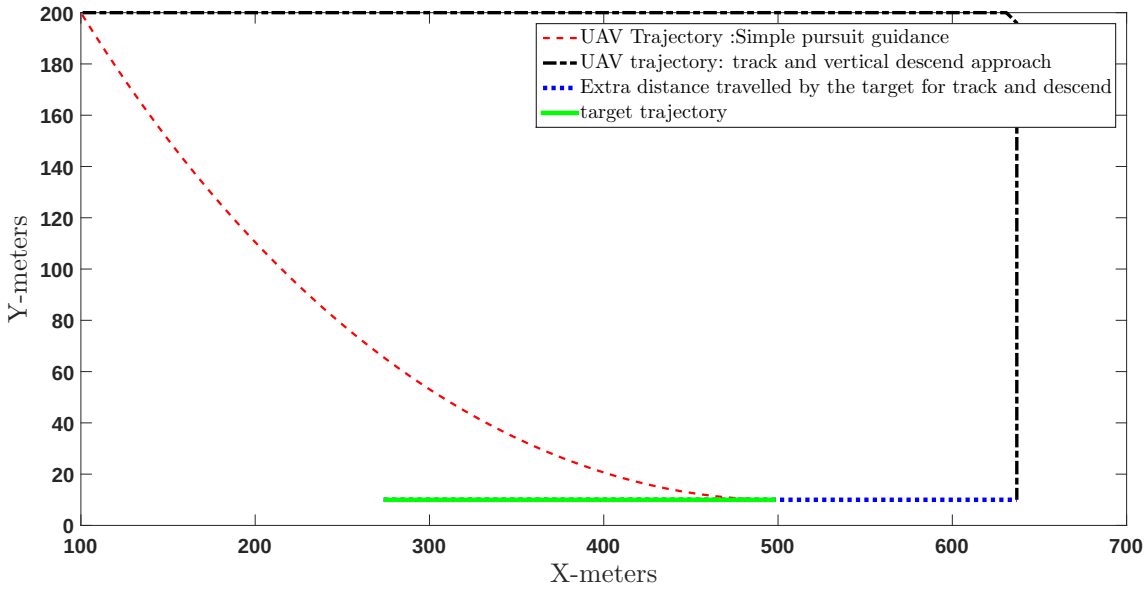


Figure 2.7: Landing trajectories comparison between simple pure pursuit and track descend approach

2.4.1 Landing using 3D guidance in simulation

We evaluate the developed guidance law for a moving target in straight line and in the presence of wind disturbances.

2.4.1.1 Straight line target motion

We consider a scenario, where the UAV is at an initial R_{xy} distance of 40 meters to target and at an altitude of 60 meters above the ground. The target heading is $\psi = 0$ deg, the initial UAV heading is $\chi_u = \psi$ and initial flight path angle $\gamma = \theta$. The simulations are terminated when $R \leq 0.1$ meters and $|\dot{R}_{xy}| \leq 0.02$ m/s. The initial UAV velocity is 2 m/s and the target velocity is 1 m/s.

The landing of the quadrotor on to the target is shown in Figure 2.8(a). The vehicle monotonically approaches the target and lands with zero closing velocity as shown in Figure 2.9(a) and 2.9(d). In these figures, we can see that $\dot{R}_{xy}$ and $\dot{R}_z$ approach 0 as $R \to 0$ thereby achieving the desired terminal conditions. Figures 2.10(a) and 2.10(d) show the acceleration profiles of the vehicle. From the figure, we can see that the acceleration reaches zero as the vehicle is approaching to land on the target.

In order to assess the performance of the guidance-based landing for varying initial target heading angles, we conducted two simulations with initial target heading angles as 30 deg and 60 deg . Figures 2.8(b) and

2.8(c) show the trajectories of the vehicles for landing. The closing velocity for both the scenarios are very similar to that the of the zero degree heading angle of the target as shown in Figures 2.9(b) and 2.9(e) for the scenario with $30\,\mathrm{deg}$ target heading angle, and Figures 2.9(c) and 2.9(f) for $60\,\mathrm{deg}$ target heading angle. Even the acceleration profiles are similar for the $30\,\mathrm{deg}$ and $60\,\mathrm{deg}$ target heading angle scenarios as shown in Figures 2.10(b), 2.10(e), 2.10(c) and 2.10(f).

2.4.1.2 Effect of wind

Quadrotor motion is affected by wind disturbances. In the presence of wind, velocity components of the quadrotor, V_{ux} and V_{uy} along X and Y direction are given by,

$$V_{ux} = V_u \cos(\gamma)\cos(\chi_u) + V_w cos(\psi_w), \tag{2.54}$$

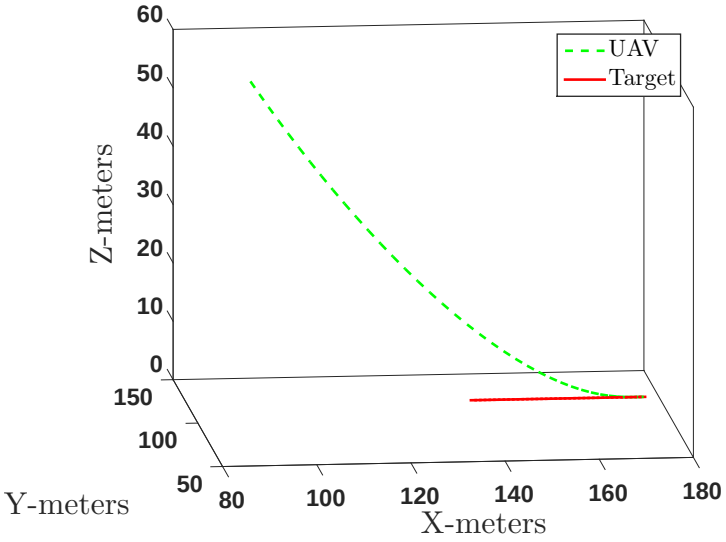
$$V_{uy} = V_u \cos(\gamma)\sin(\chi_u) + V_w sin(\psi_w). \tag{2.55}$$

We assume that the wind disturbance is along the horizontal plane only. We first consider two cases of wind disturbances with speeds of V_w = $0.2V_u$ and $0.5V_u$ in the direction ψ_w = $60\,\mathrm{deg}$. The simulation parameters for the UAV and target were same as the no wind case for a target heading of $0\,\mathrm{deg}$. Trajectories in the presence of wind disturbances are shown in Figure 2.11(a). Lateral acceleration and closing velocity profiles are shown in figures 2.11(b), 2.11(c), 2.12(a) and 2.12(b). We also performed simulations with varying direction of wind. The speed of wind was V_w = $0.3V_u$ with directions varying w.r.t course angle i.e χ_u+$\pi/4$ and χ_u-$\pi/4$. Trajectories in the presence of wind disturbance of varying direction are plotted in Figure 2.13(a). Lateral acceleration and closing velocity profiles are shown in figures 2.13(b), 2.13(c) and 2.14. It can be seen that a successful landing with minimum closing velocity is achieved even in the presence of wind disturbances.

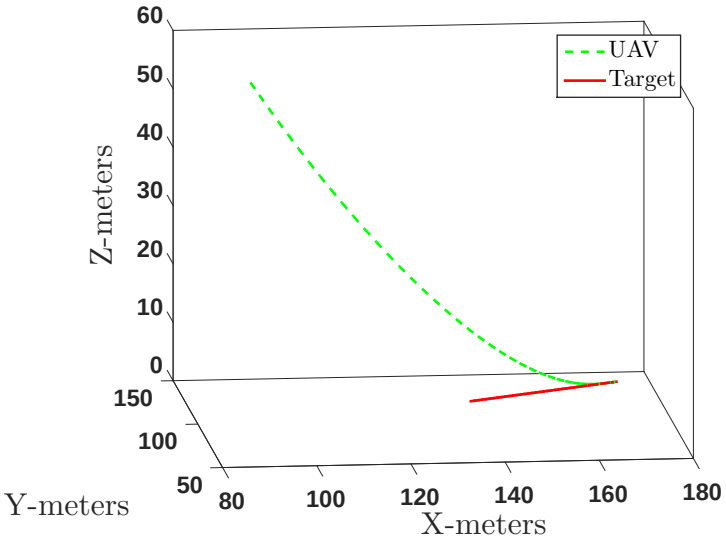
2.4.2 Landing with noise in target information

In the previous case, it was assumed that accurate information about the target is available to the UAV. In the field, the received target information is often noisy. Therefore, to analyze the robustness of the landing controller simulations are performed with a noisy target model. State vector $X_t(n)$

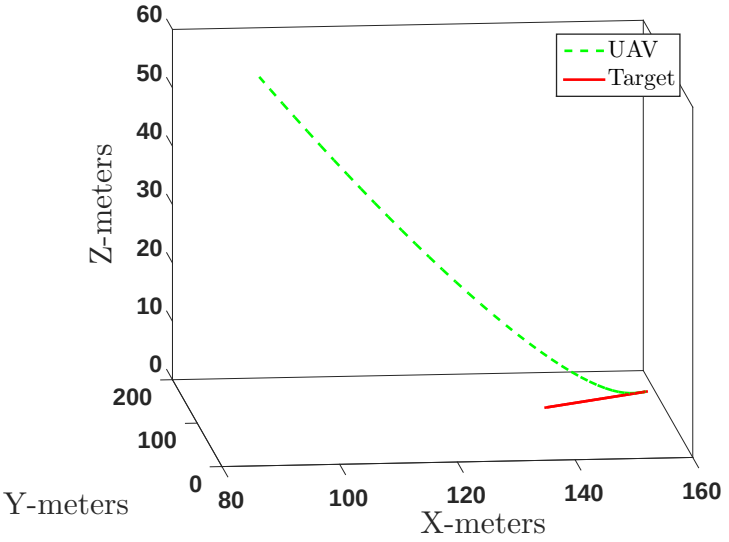
$$\begin{bmatrix} x_t(n) \\ y_t(n) \\ \dot{x}_t(n) \\ \dot{y}_t(n) \end{bmatrix},$$



(a)



(b)



(c)

Figure 2.8: MATLAB results showing the trajectory followed by the UAV to land, for target heading of (a) 0 deg (b) 30 deg (c) 60 deg

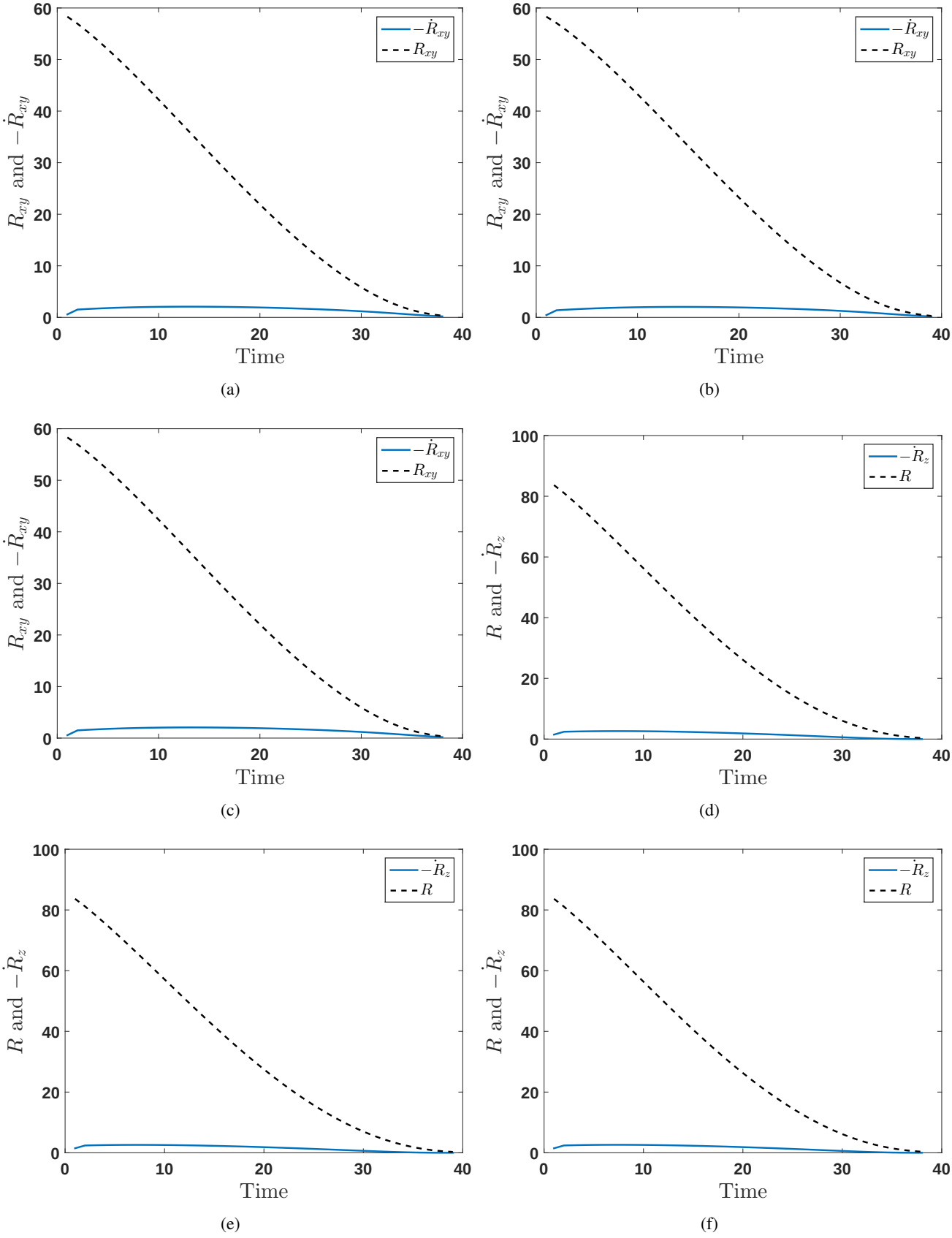


Figure 2.9: Closing velocity profile with time in MATLAB, for a target heading of (a) 0 deg in xy plane (b) 30 deg in xy plane (c) 60 deg in xy plane (d) 0 deg in z direction (e) 30 deg in z direction (f) 60 deg in z direction.

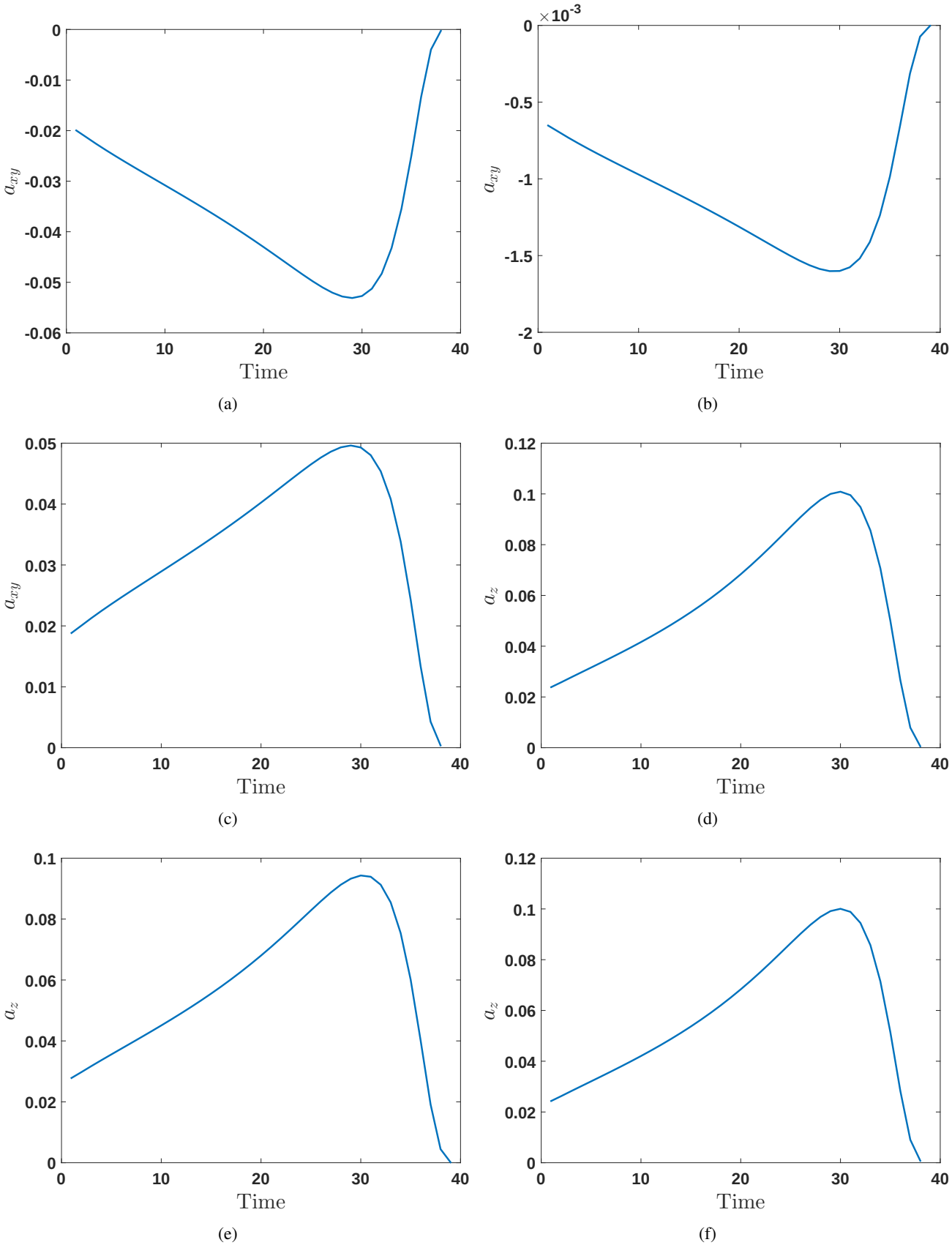


Figure 2.10: Lateral acceleration profile with time in MATLAB, for a target heading of (a) 0 deg in xy plane (b) 30 deg in xy plane (c) 60 deg in xy plane (d) 0 deg in z direction (e) 30 deg in z direction (f) 60 deg in z direction.

and the measurement vector $Z_t(n)$

$$\begin{bmatrix} x_t(n) \\ y_t(n) \end{bmatrix}.$$

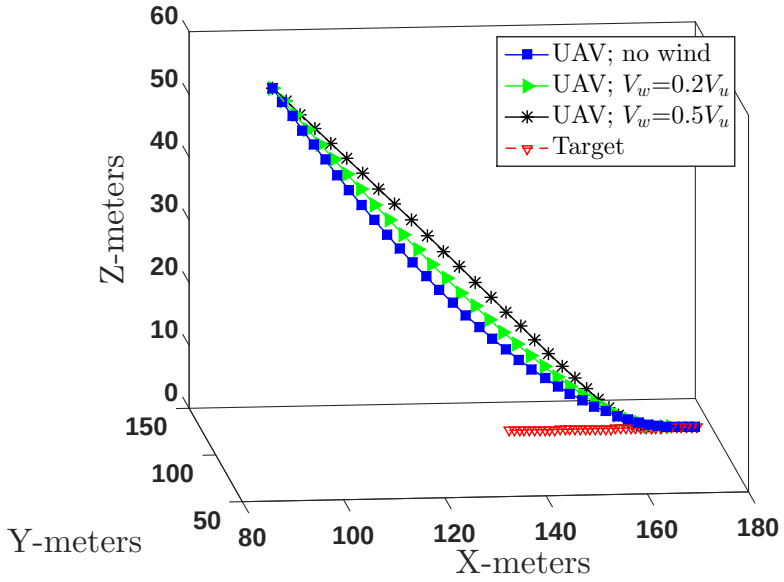
Noise was added to the target position in the form of measurement noise ($\mathcal{N}(0,\ 1.5^2)$) and to target velocity in the form of process noise ($\mathcal{N}(0,\ 0.4^2)$) , both were considered to be non correlated and zero mean white gaussian noise. Figures 2.15 and 2.16 show the estimated target trajectory and the simulated trajectory followed by the UAV to land on the target. UAV was able to land on the target in the presence of noise while also meeting the desired terminal conditions.

2.4.3 ROS Simulation results

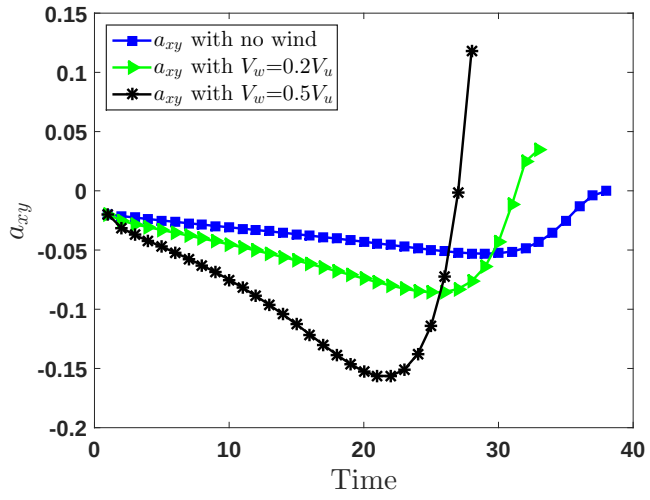
Simulations were also carried out in ROS [81] quadrotor simulator (hector-quadrotor) [82][83]. It is based on gazebo and contains packages related to modeling, simulation and control of quadrotors. Figure 2.17 shows the active ROS nodes and topics involved. The main node, *landing guidance* is subscribed to the ROS topic *ground truth to tf/pose* which gives the position and orientation of the quadrotor within the gazebo world simulator. *landing guidance* generates the necessary guidance commands and the resulting velocity commands are published in order to guide the UAV to land on the target. Simulations were started with an initial UAV and target velocity of 2 m/s and 1 m/s respectively. R_{xy}^0 was 50 meters and the UAV started at an altitude of 40 meters above the ground. The simulations were terminated when $R \leq 0.1$ meters with $\left|\dot{R}_{xy}\right|$ of ≤ 0.02 m/s. Trajectory followed by the UAV for a heading angle of 0 deg is shown in 2.18. Closing velocity and lateral acceleration profiles are shown in figures 2.19 and 2.20 respectively.

2.5 Hardware setup

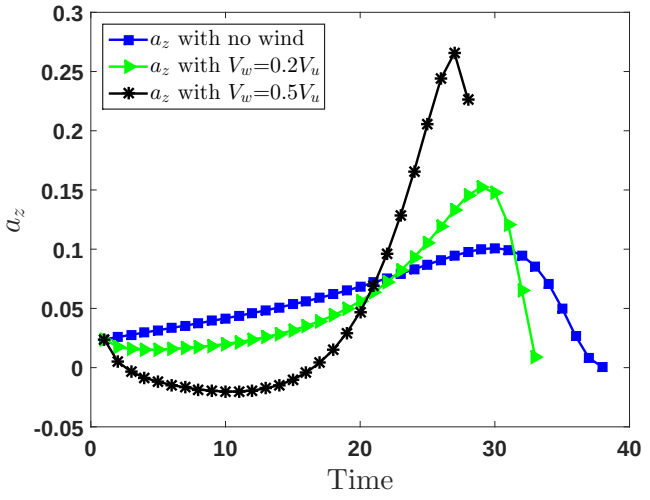
We use an off the shelf quadcopter (3DR IRIS+) [84] for hardware validation of our landing algorithm. The landing algorithm runs on a linux based embedded computer – Odroid U3 [85]. This is mounted on the top of the quadrotor and connected to a downward facing 720P camera through a USB connection. The details of the hardware setup and the information flow is shown in Figure 2.21 and the quadrotor with the associated accesories is shown in Figure 2.22. We implement our approach using ROS [86] and use OpenCV [87] to implement the target detection. Guidance commands are sent to the quadrotor over a USB link using ROS package mavros which converts ROS messages to mavlink format and vice versa, to aid communication with the autopilot.



(a)

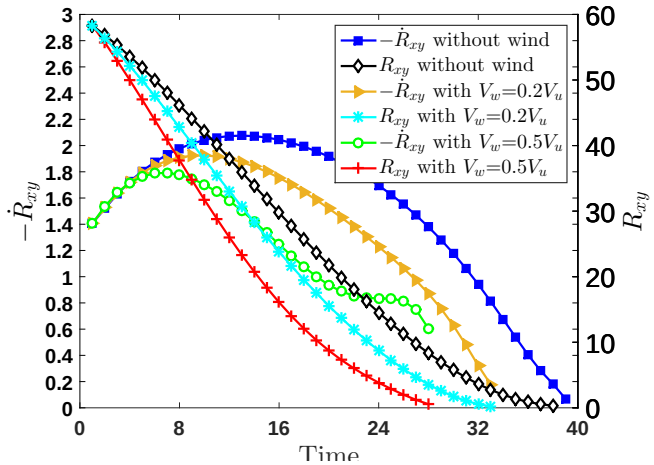


(b)

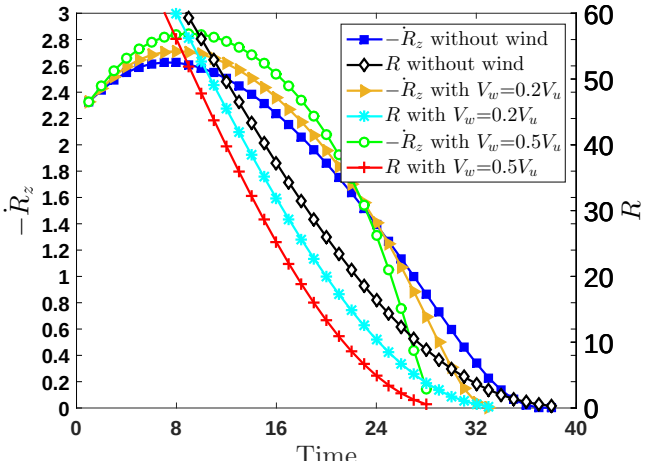


(c)

Figure 2.11: Effect of wind disturbances on the vehicle while landing on a target moving with heading of 0 deg. (a) Trajectory followed by the UAV (b) lateral acceleration profile with time in MATLAB, in xy plane (c) vertical acceleration profile with time in MATLAB, in z direction.

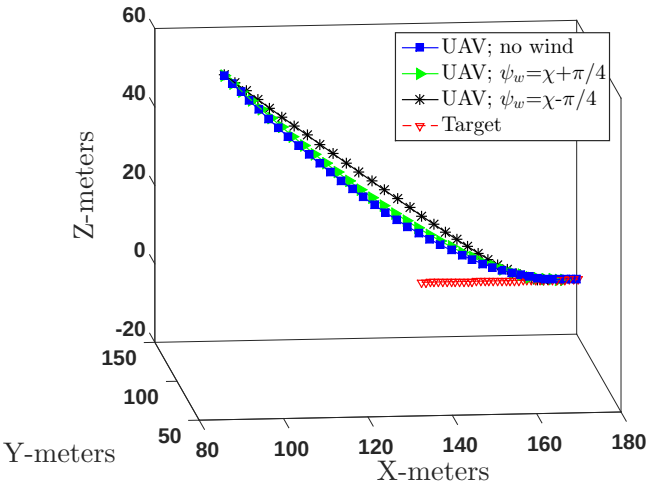


(a)

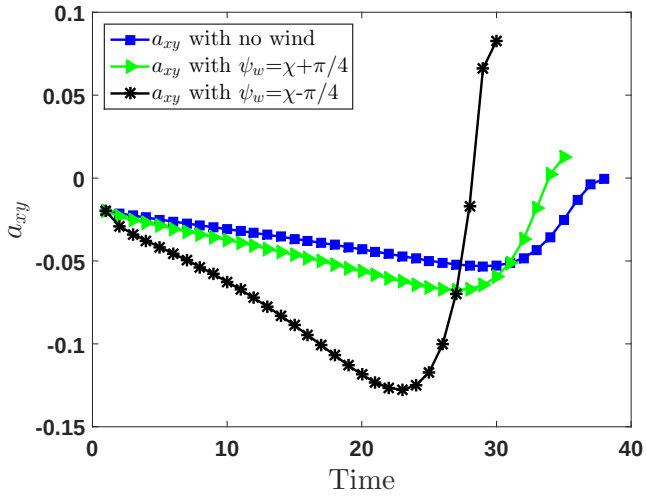


(b)

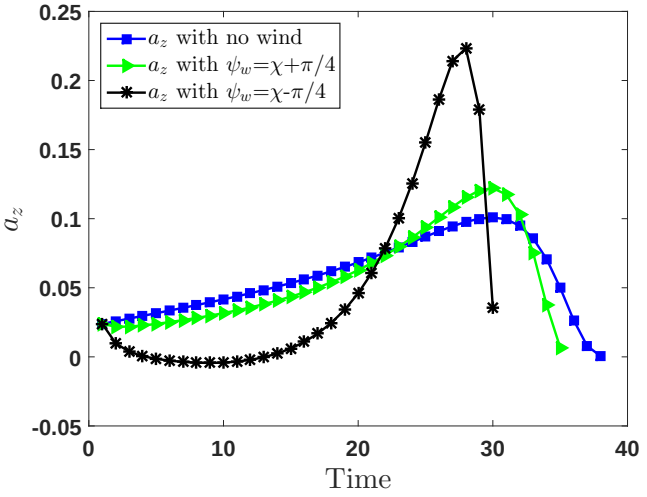
Figure 2.12: Comparison of effect of wind disturbances for a target heading of $0\,\mathrm{deg}$ on (a) closing velocity profile with time in MATLAB, in xy plane (b) closing velocity profile with time in MATLAB, in z direction.



(a)



(b)



(c)

Figure 2.13: Effect of varying wind direction on landing for a target heading of $0\,\mathrm{deg}$. (a) Trajectory followed by the UAV (b) Lateral acceleration profile with time in MATLAB, in xy plane (c) Vertical acceleration profile with time in MATLAB, in z direction.

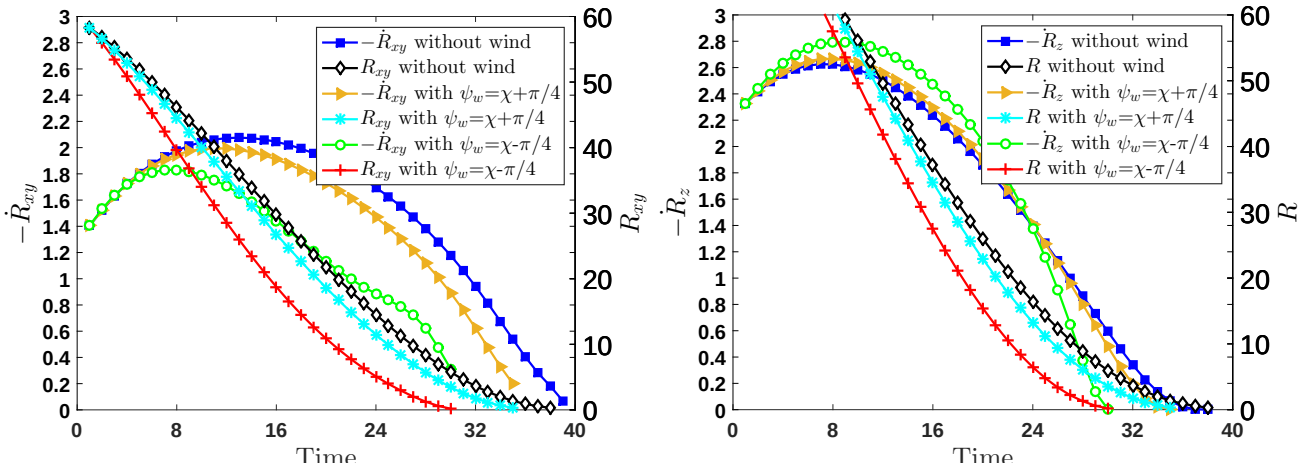


Figure 2.14: Comparison of effect of varying direction of wind disturbances for a target heading of 0 deg on (a)closing velocity profile with time in MATLAB, in xy plane (b) closing velocity profile with time in MATLAB, in z direction.

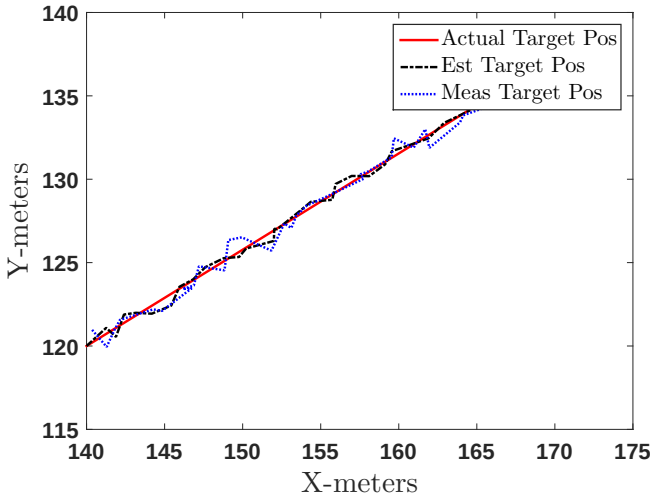


Figure 2.15: Kalman Filter tracking performance shown by the actual, measured and estimated target trajectories

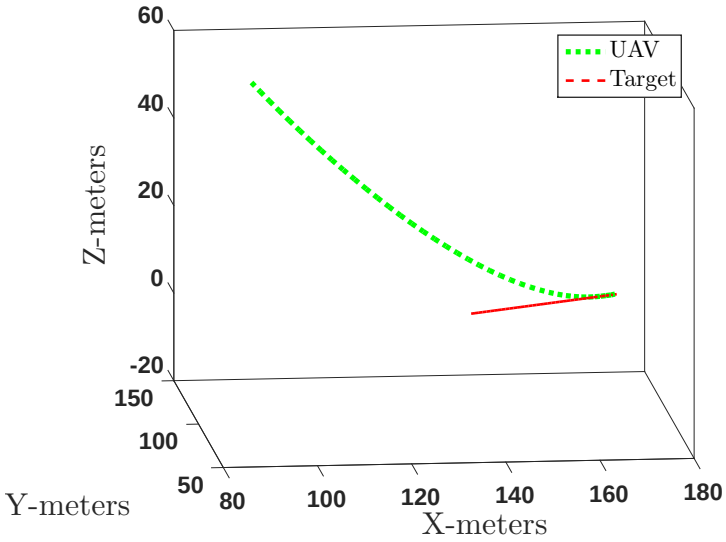


Figure 2.16: Trajectory followed by the UAV when using estimated target information

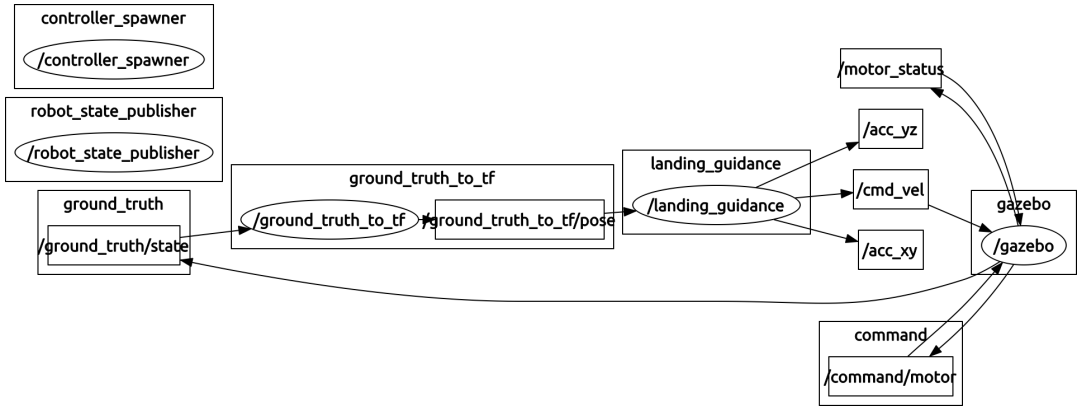


Figure 2.17: ROS nodes and topics involved in ROS simulations

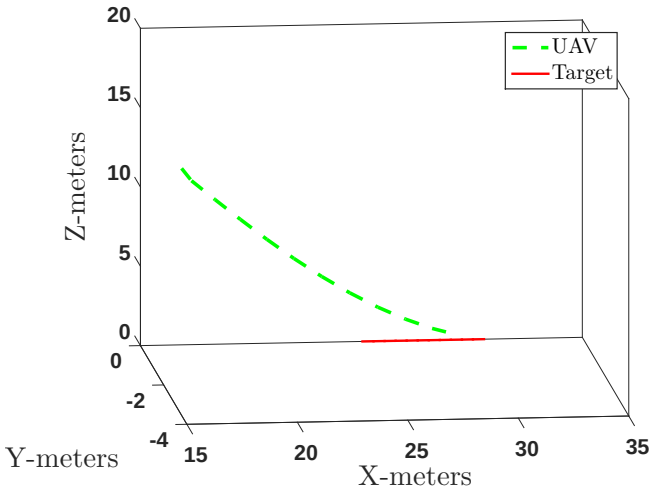
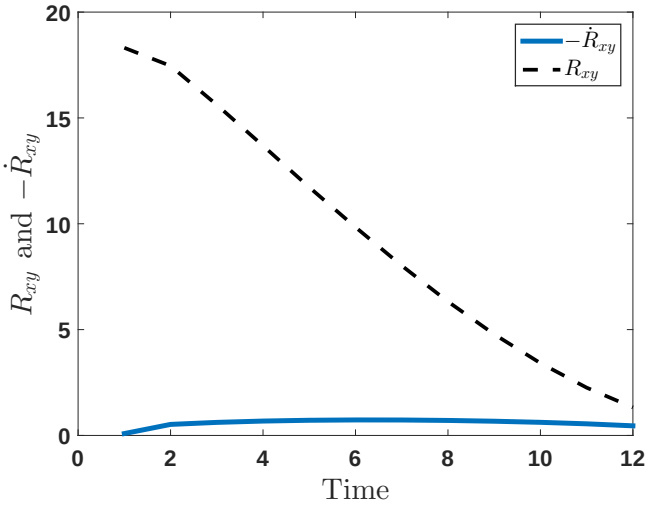
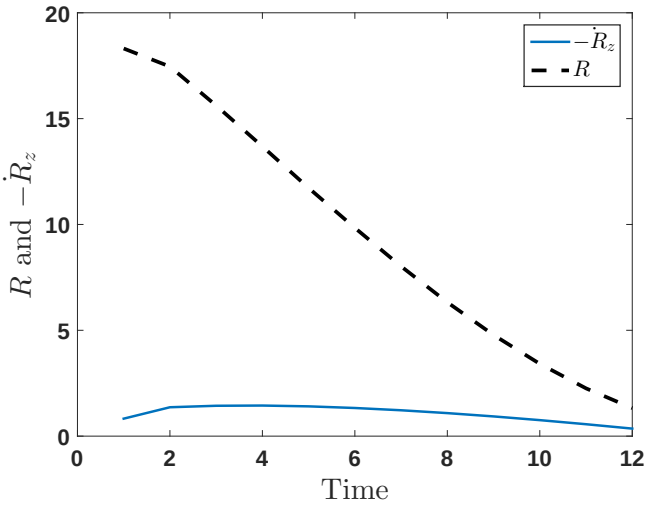


Figure 2.18: ROS results showing the trajectory followed by UAV to land on the target moving with 0 deg heading.

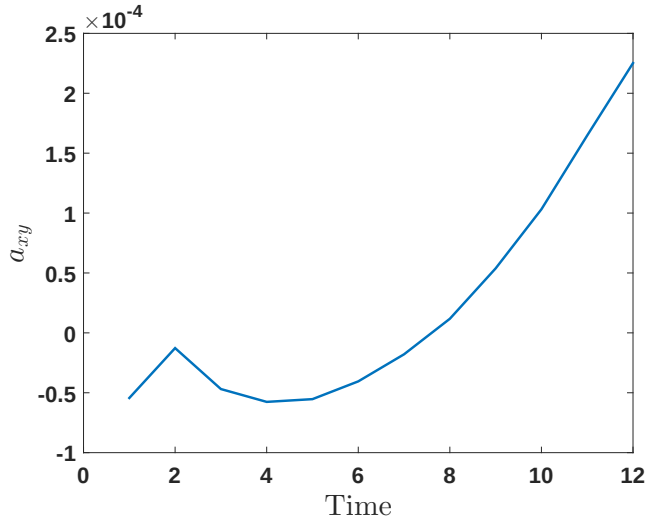


(a)

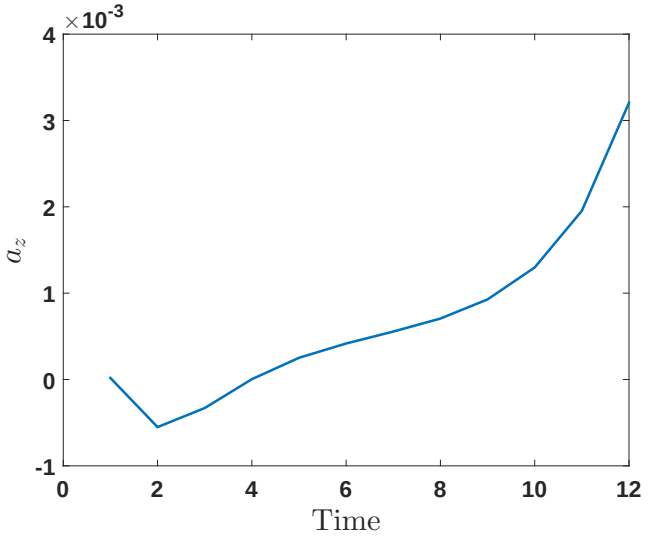


(b)

Figure 2.19: Closing velocity profiles for a target heading of 0 deg in ROS (a) in xy plane with time (b) in z direction with time.



(a)



(b)

Figure 2.20: Lateral acceleration profiles for a target heading of 0 deg in ROS (a) in xy plane with time (b) in z direction with time.

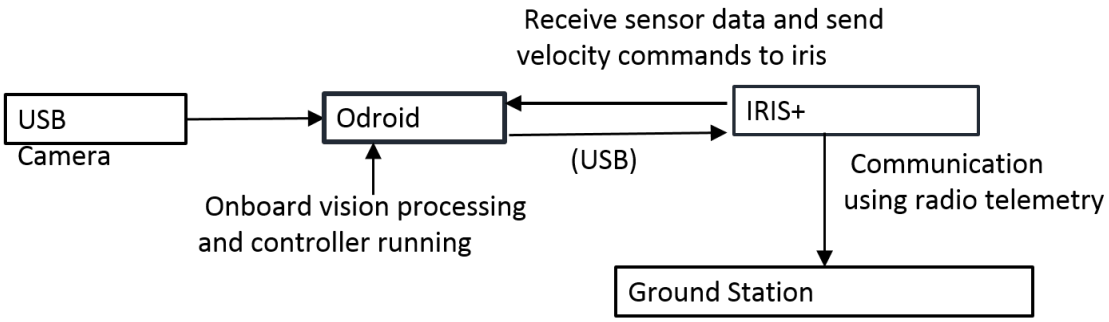


Figure 2.21: Hardware setup and information flow diagram consisting of 3DR IRIS+ (quadrotor), Odroid U3 and USB camera

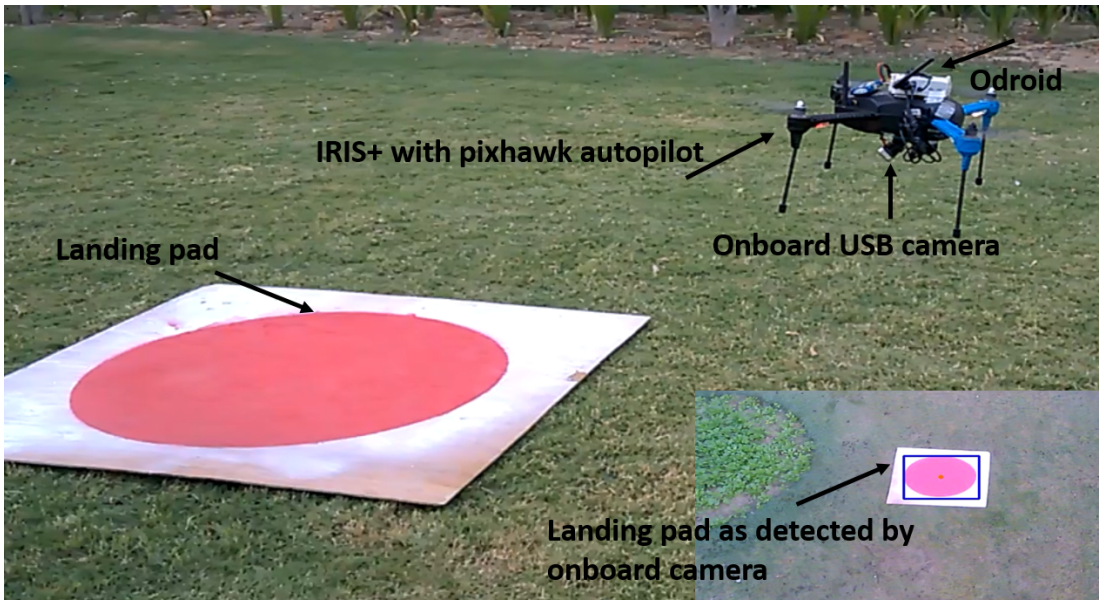


Figure 2.22: Quadcopter with the whole hardware interface about to land on the target

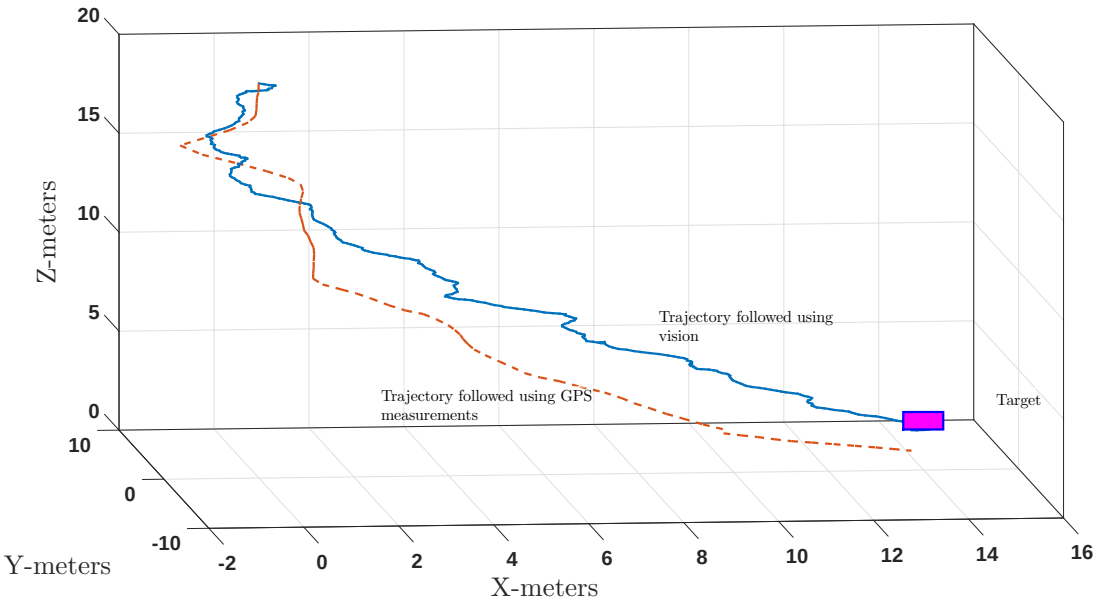


Figure 2.23: Comparison of the trajectories followed by the quadcopter to land on the target using vision and GPS measurements

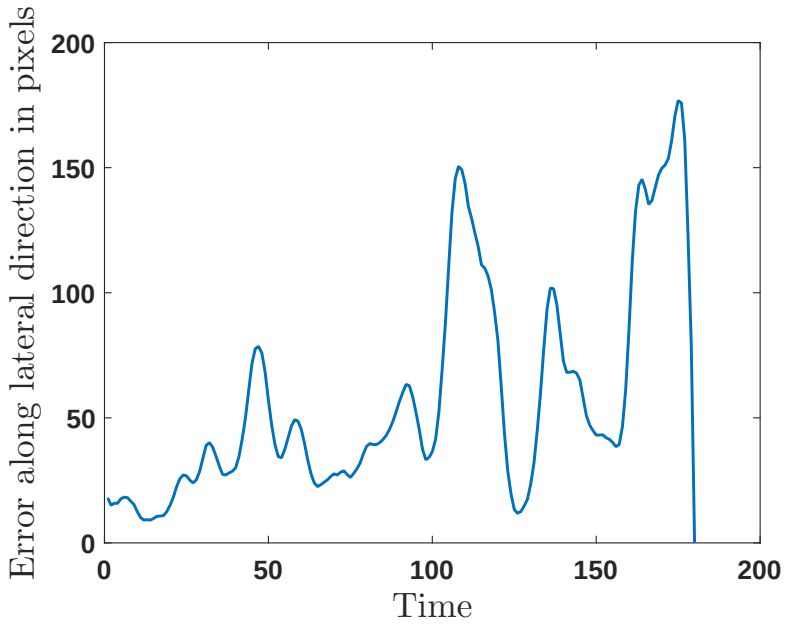
2.6 Hardware results

In order to validate our algorithm, we performed experiments with the hardware setup described in Section 2.5. Two sets of experiments were performed. In the first case, GPS coordinates of the target were assumed to be known and guidance commands were generated based on GPS measurements. This was done in order to validate our guidance algorithm without vision in the control feedback loop. In the second experiment, target was detected using vision, and based on the extracted information guidance commands were issued to the quadcopter. To initiate the landing process, the quadcopter was commanded to takeoff to an altitude of 20 meters. The target was stationary and the image data was processed online.

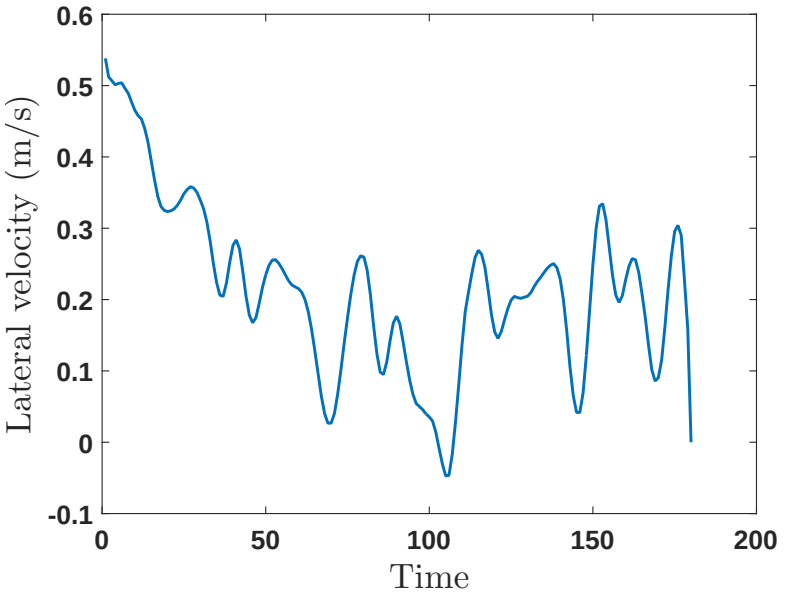
Figure 2.22 shows a typical image obtained from the downward pointing camera on the quadcopter and the desired landing target as identified by the vision system (bounding box in the Figure). At each step of the landing process, the guidance system generates (a) velocity commands in xy plane with respect to image frame (b) velocity commands in z direction based on the altitude. These commands were converted to velocity north, east and down commands based on the heading of the quadcopter. Figure 2.23 shows the trajectories followed by quadrotor with GPS measurements and vision based input. The quadcopter was able to land on the target precisely using vision information, as opposed to when GPS measurements were used. The magnitude of the total error in XY plane in case of GPS measurements was found out to be 3 meters. Figures 2.24 and 2.25 show the performance of vision based controller. The variation of velocity in both lateral and longitudinal directions is according to the pixel error in respective directions. Figure 2.26 shows the performance of the closing velocity controller with GPS input. It can be observed that the variation in closing velocity is same as that obtained through simulations and the closing velocity can be seen to be approaching zero towards the end. As observed from the hardware results with stationary target, the guidance based approach enables the quadrotor to land on the landing pad accurately. The integrated closing velocity controller provides velocity commands in reference to quadrotor position from target and also achieves zero closing velocity, necessary for event-less landing.

2.6.1 Landing in the presence of target occlusion

One of the major challenges of the landing process is losing the sight of the target. This can happen if the target gets partially or fully occluded due to obstacles and/or goes out of field of view. It results in erroneous estimate of the target, causing deviation from the desired trajectory. To overcome this situation, we use Kalman Filter (KF) to achieve a robust estimate of the landing pad parameters. The output of detection algorithm (x_t and y_t centroid pixel coordinates of the target), are given as input to the estimator. The extracted target parameters

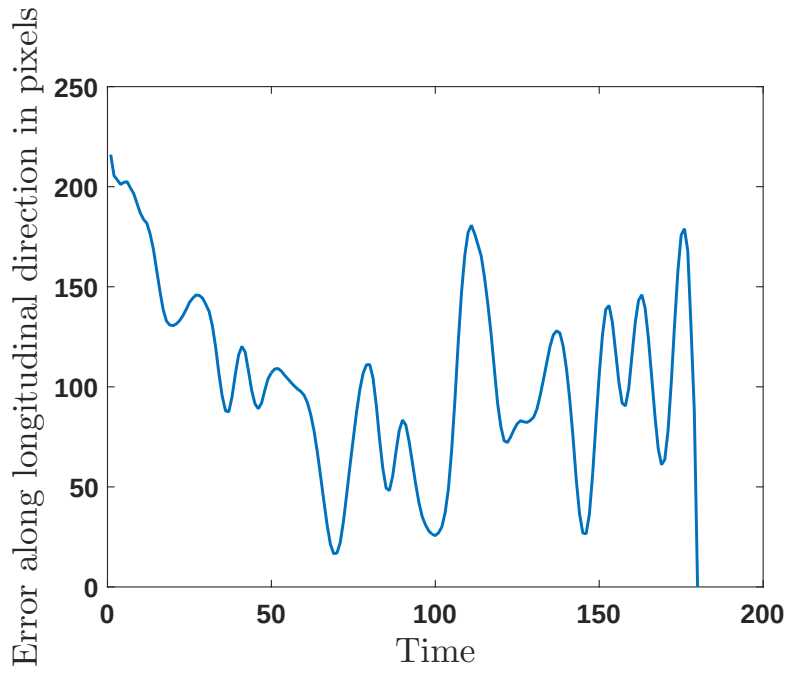


(a)

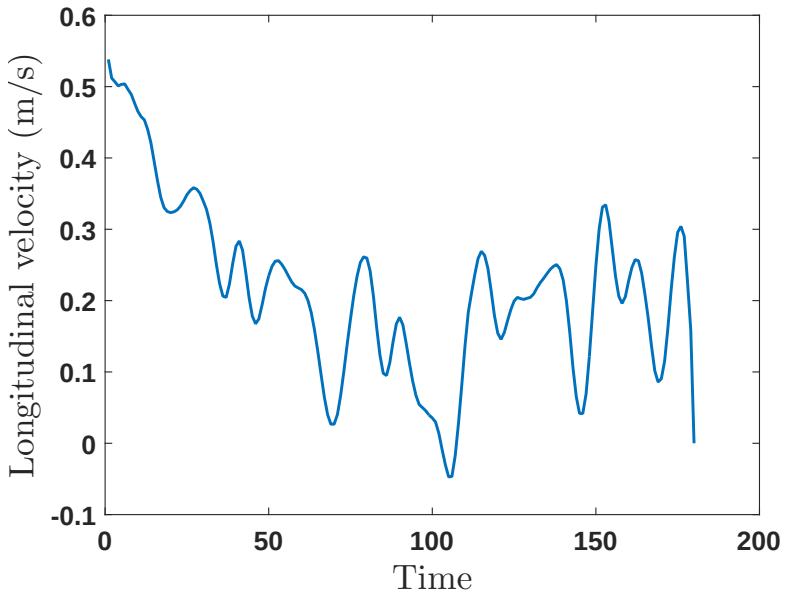


(b)

Figure 2.24: (a) Error output by controller in lateral direction (b) Lateral velocity profile.

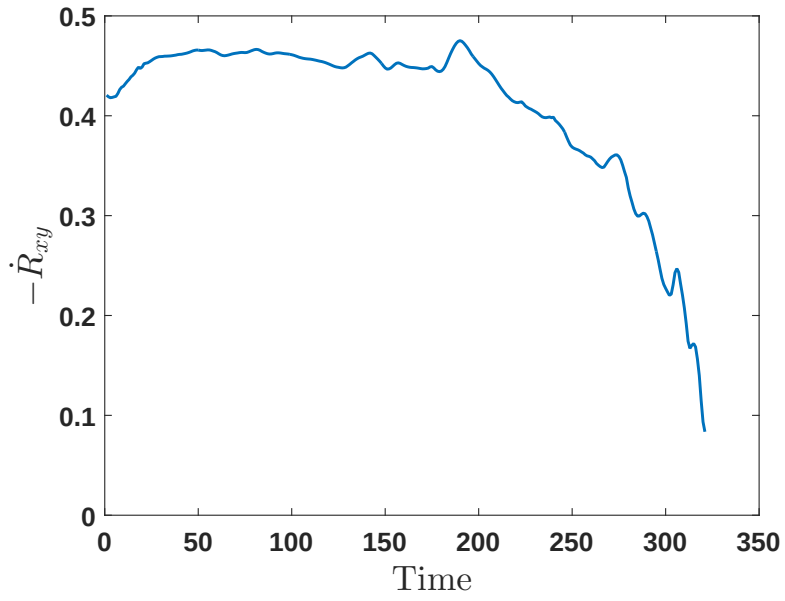


(a)

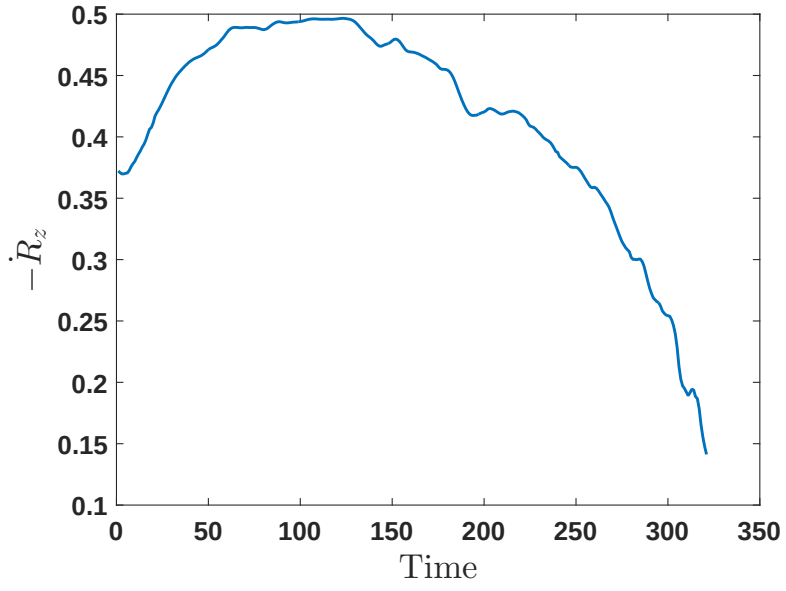


(b)

Figure 2.25: (a) Error output by controller in longitudinal direction (b) Longitudinal velocity profile

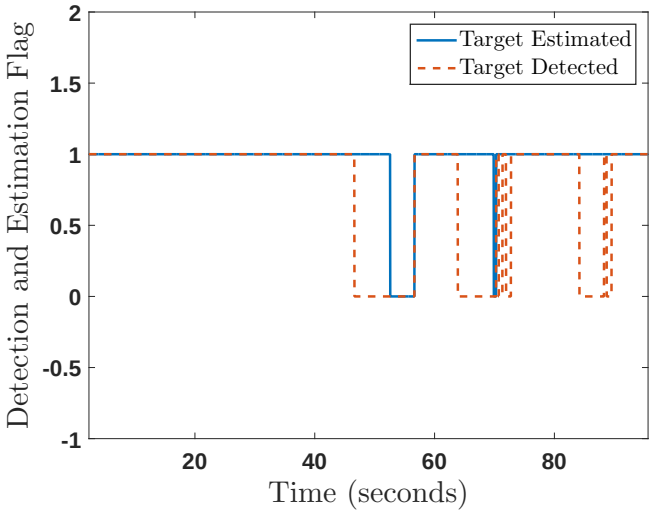


(a)

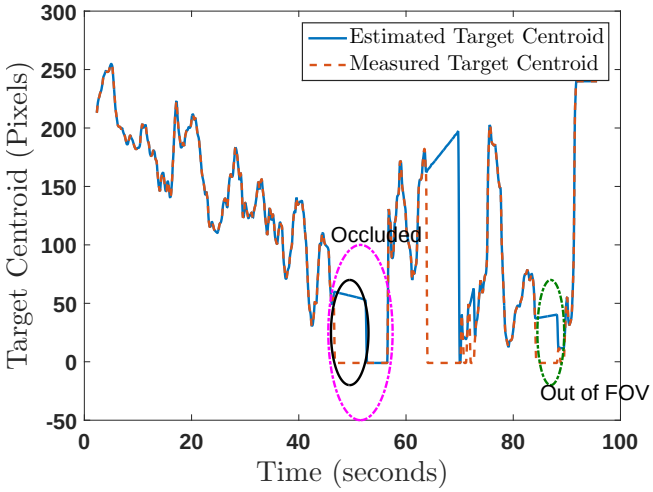


(b)

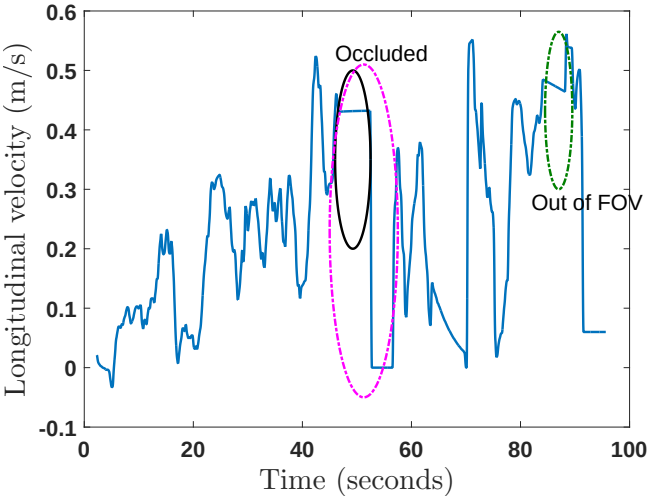
Figure 2.26: Closing velocity profile with time from GPS measurements (a) in xy plane (b) in z direction



(a)



(b)



(c)

Figure 2.27: (a) Detection and Estimation Flag (b) Measured and estimated target center in longitudinal direction (c) Velocity commands based on vision estimate.

using vision are used to generate appropriate guidance commands for the quadrotor.

We have performed an experiment, where we fully occlude the target during certain phases of the flight. Fig. 2.27 shows one such occluded target phase of 10 seconds. In Figure 2.27(a), a zero value of detection flag indicates that the target is occluded and/or out of field of view. When the target is occluded, no measurements are received and the predicted estimate by KF is used for 60 frames (6 seconds approx.) (empirically chosen) to generate the guidance commands. This is represented by the value of 1 for estimation flag even when the detection flag is 0. After 60 frames the quadrotor goes into hover mode (0 value of estimation flag), until the landing pad is detected again.

Figs. 2.27(b) and 2.27(c) show the estimated target centroid and the velocity commands generated during the experiment. It can be observed that appropriate velocity commands are generated, even when the target is occluded. A zero velocity command is generated when the estimation flag becomes 0 causing the quadrotor to change to hover mode. Figs. 2.27(b) and 2.27(c) also show the scenario where the landing target is out of field of view, which occurs typically towards the end of the landing phase. In this case also, appropriate velocity commands are generated, which enable the quadrotor to fly towards the target and land accurately.

2.6.2 Landing on moving target in a straight line

Next, we validate the log polynomial controller for a moving target scenario. In this experiment scenario, the quadcopter was commanded to takeoff to an altitude of 10 meters and the target was moved in an approximate straight line motion with an approximate velocity of $0.5m/s$ (as given by the autopilot mounted on the target) in longitudinal direction and negligible movement in lateral direction. In case of a moving target tracking and landing, velocity of the UAV must be controlled according to the velocity of the target. Thus, it is important to compute the target velocity correctly. We use a KF for a robust estimate of the landing pad parameters (x_{img}, y_{img}), to deal with sudden changes in detection output, resulting from wind disturbances and target occlusion. The relative velocity on the image frame is calculated by the slope of the image positions. We can express the target velocity as given below:

$$\begin{bmatrix} v_{(tx)} \\ v_{(ty)} \end{bmatrix} = \begin{bmatrix} v_{qx_{t-\delta(t)}} \\ v_{qy_{t-\delta(t)}} \end{bmatrix} + \begin{bmatrix} K_x & 0 \\ 0 & K_y \end{bmatrix} \begin{bmatrix} \cos(\psi) & -\sin(\psi) \\ \sin(\psi) & \cos(\psi) \end{bmatrix} \begin{bmatrix} \dot{x_{img}} \\ \dot{y_{img}} \end{bmatrix}. \tag{2.56}$$

where (K_x, K_y), represent the constants for transformation between the local NED coordinate and the image plane. The computed target velocity components are used to design the lateral and longitudinal velocity control subsytems independently using the log polynomial velocity profile. For altitude control, velocity profile is

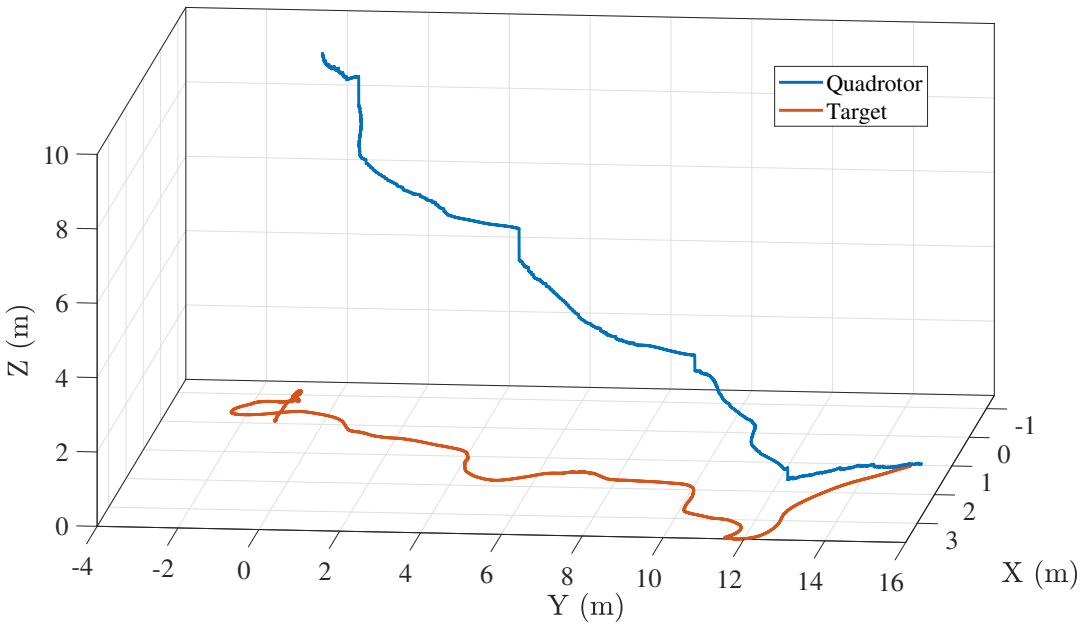


Figure 2.28: Trajectory followed by the Quadcopter to land on a moving target

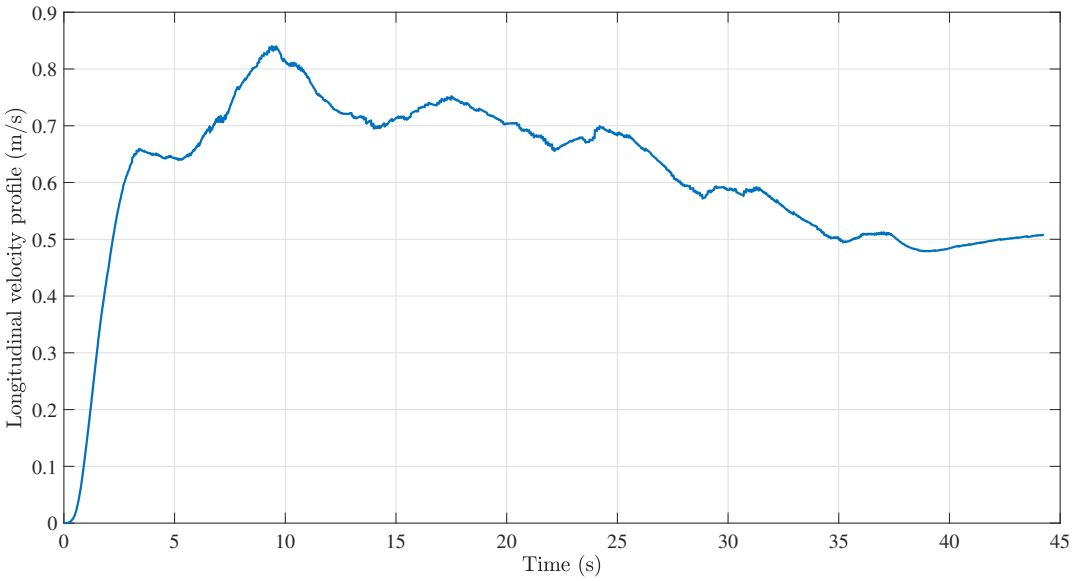
given as,

$$V_z = V_{z0} \log_n \left(1 + ax + bx^2\right) \tag{2.57}$$

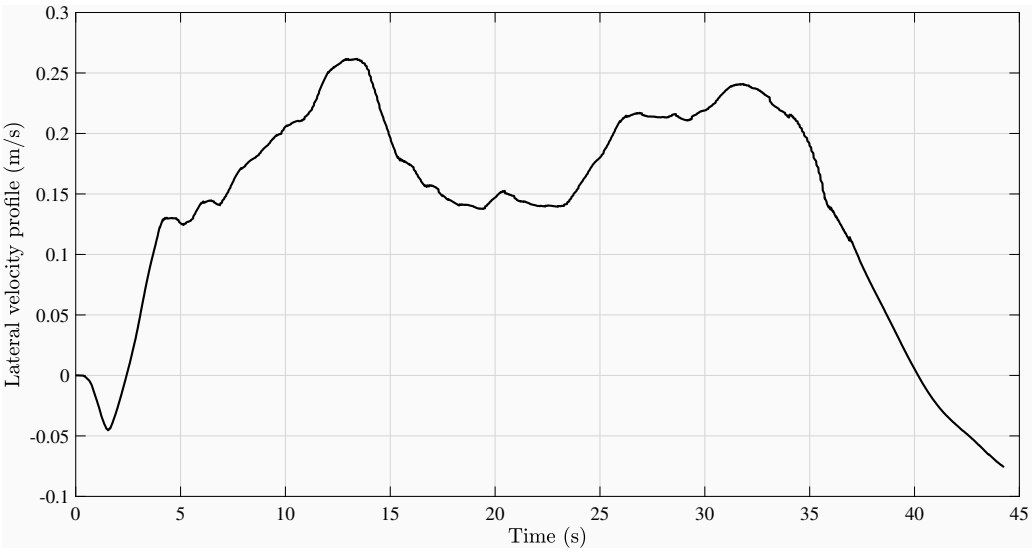
where x represents the ratio of current to initial altitude, and V_{z0} represents the initial velocity in z direction. Further, a flag was set for the quadrotor to descend such that quadrotor was commanded a z velocity only if this flag was equal to 1 as given in Equations. (2.59) and (2.60)

$$\begin{aligned} flag_{descend} &= 1, \text{if } R_{ut} \leq \frac{fp}{1.5}, && (2.58) \\ V_z &= V_{z0} \log_n \left(1 + ax + bx^2\right), \text{if } flag_{descend} = 1, && (2.59) \\ V_z &= 0, \text{if } flag_{descend} = 0 && (2.60) \end{aligned}$$

The flag was set to 1, if the distance between UAV and the target (R_{ut}) was less than $\frac{fp}{1.5}$, where fp is the camera footprint in meters at the respective altitude. In case of the distance being greater than $\frac{fp}{1.5}$, descend flag was set to 0 and thus the commanded V_z was also equal to zero. Fig. 2.28 shows the trajectory followed by the quadrotor to land on the target. Figs. 2.29(a) and 2.29(b) show the longitudinal and lateral velocity profiles of the quadrotor. It can be seen that the longitudinal velocity profile approaches the target speed towards the end and the lateral velocity profile approaches 0 thus resulting in zero closing velocity for a successful landing.



(a)



(b)

Figure 2.29: Quadrotor velocity profile in case of moving target landing (a) Longitudinal velocity profile (b) Lateral velocity profile

2.7 Conclusion

In this chapter, a pure pursuit guidance based landing approach for quadrotors is proposed for an accurate autonomous landing. A novel log polynomial closing velocity controller is integrated with pure pursuit guidance to allow faster landing with minimum closing velocity. The efficacy of the proposed approach is validated exhaustively through simulations. We also performed validation of the proposed approach for a stationary as well as a moving landing pad.

Chapter 3

Analysis of log polynomial profile for vertical landing

3.1 Introduction

Autonomous landing is a critical part of the missions associated with these applications. Prospective landing guidance strategies desire a smooth velocity variation with zero closing velocity at touchdown. Further, a greater control over the landing time requires a flexibility in controlling the acceleration and the velocity profiles of the vehicle. Guidance methods used in Refs. [88, 89, 90, 91, 92] are preliminary in nature and meet only terminal constraints of landing without offering any explicit control over the velocity profile. On the other hand, bio-inspired approaches based on tau guidance have been used in quadrotor applications since they offer flexibility in the landing process. In these methods, guidance commands are generated based on time-to-go, typically denoted by τ. Guidance methods for achieving soft or hard landing based on the nature of $\dot{\tau}$ (constant or varying) were developed in Refs. [93] [94]. Though $\dot{\tau}$ based approach provides some flexibility to the nature of landing, absence of control over parameters like rate of decay of velocity, results in a strictly decreasing profile which poses a limitation in terms of time to land. This limitation was addressed in [95] by using intrinsic tau method which includes time to land as an additional controllable dimension. The intrinsic tau approach results in an initially accelerating velocity profile followed by deceleration till touchdown. However, this flexibility was based on the assumption that the vehicle must start in a hover state. Yang et al. [96] proposed an improved intrinsic tau guidance strategy for decentralized 4D trajectory generation. The improvement is based on adding an initial velocity term to the intrinsic tau guidance framework so as to aid the vehicle to start with an arbitrary initial velocity and reach the desired velocity in given time. However, the equation relating the

guidance gain with maximum velocity is an exponential function of the gain and hence imposing a maximum velocity constraint over the landing process would typically require an iterative numerical method to obtain the desired gain.

It is also noteworthy that, though time-based approaches bring in necessary flexibility in guidance design, any unexpected event or system delays would result in re-planning from a new commanded time horizon. The landing process is inherently a distance-based maneuver planning and with the advent of highly accurate and low weight on-board altimeters [97, 98, 99, 100] a distance-based velocity profile can provide a robust and practical alternate.

In this chapter, we consider a vertical landing scenario and theoretically analyze the log polynomial function of distance velocity controller. The decay characteristics of the log function, inherently guarantee zero closing velocity and acceleration at touchdown. Analytic conditions are deduced on the two design parameters for achieving, strictly decreasing, initially accelerating and hovering velocity profiles, respectively. Closed-form expressions as function of design parameters are derived for maximum velocity, and engagement time constraints, respectively. As a major design advantage of the proposed method, the choice of velocity profile (strictly decelerating, initially accelerating or hovering), maximum velocity constraint, and engagement time can be simultaneously analyzed and visualized in the 2-dimensional design parameter space.

The chapter is arranged as follows: The problem is briefly described in Section 3.2 followed by the proposed velocity profile in Section 3.3. Section 3.4 discusses analysis in detail, the resulting flight characteristics. Simulations are presented in Section 3.5. Section 3.6 contains concluding remarks.

3.2 Problem Description

Consider an aerial flight scenario as shown in Figure 3.1, where a quadrotor is trying to land vertically on a stationary landing point with an instantaneous distance R and instantaneous downward velocity V_u. At the beginning of the landing phase, the quadrotor has an initial downward velocity of V_{u0} and it is at an altitude of R_0 above the landing point. In order to have a safe landing, the distance between the vehicle and the landing point must decay to zero with zero closing velocity at touchdown. That is, for a final time t_f,

$$\lim_{t \to t_f} R = \lim_{t \to t_f} -\dot{R} = 0 \tag{3.1}$$

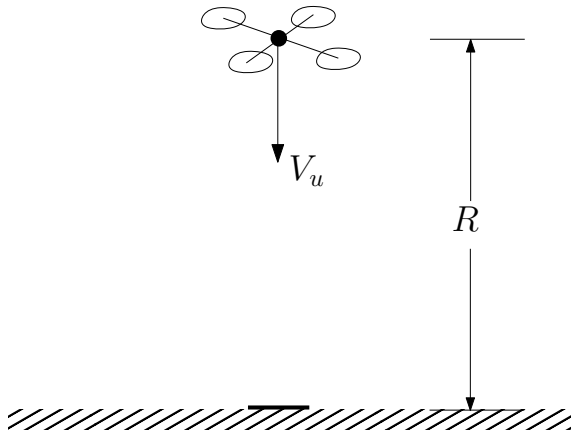


Figure 3.1: Flight scenario of a quadrotor descending on landing point.

where, $-\dot{R}$ is the closing velocity between the vehicle and the landing point. For vertical landing on a stationary landing point,

$$\dot{R} = -V_u \tag{3.2}$$

where, V_u is the downward velocity of the quadrotor. Using Equation (3.2) in Equation (3.1), the terminal requirement can be expressed as,

$$\lim_{t \to t_f} R = \lim_{t \to t_f} V_u = 0 \tag{3.3}$$

Prospective velocity profiles are desired for a smooth variation of the vehicle velocity satisfying the initial conditions and the final constraints of Equation (3.3). Hovering, as a subproblem, requires a similar approach to a hovering altitude R_d, with $R_d > 0$. The next section introduces a new velocity profile and analyses its basic properties.

3.3 Log polynomial Velocity Profile

A log of polynomial velocity profile as given in Equation (2.40) is

$$V_u = V_{u0} \log_n \left(1 + ax + bx^2\right) \tag{3.4}$$

where, x represents the ratio of current and initial distance to landing point and is defined as,

$$x = \frac{R}{R_0}, \;\; 0 \leq x \leq 1 \tag{3.5}$$

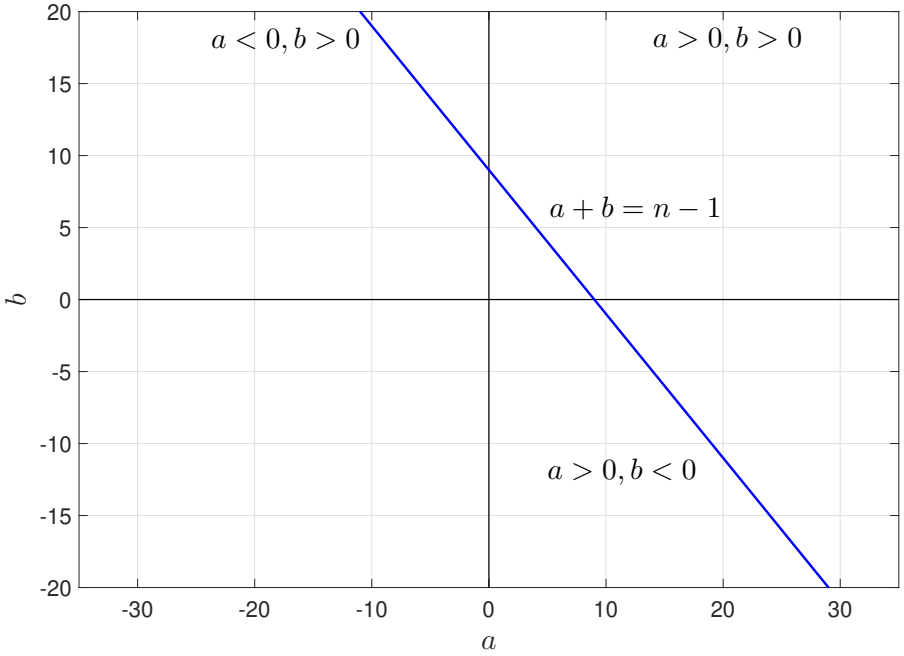


Figure 3.2: Solution in $a - b$ design space.

and the constants a and b are related to base of the log n by,

$$a + b \quad = \quad n - 1, \quad n > 1 \tag{3.6}$$

Initially at $t = t_0$, $R = R_0$, which using Equation (3.5), implies $x = 1$. Using Equation (3.6) in Equation (3.4) with $x = 1$ leads to,

$$\lim_{x \to 1} V_u = V_{u0} \log_n (n) = V_{u0} \tag{3.7}$$

Equation (3.7) shows that the proposed profile satisfies the initial condition. As the vehicle approaches the landing point, the distance R and and hence the ratio x reduces. The touch-down velocity can be obtained as,

$$\lim_{x \to 0} V_u = \lim_{x \to 0} V_{u0} \log_n \left(1 + ax + bx^2\right) = 0 \tag{3.8}$$

Hence, the closing velocity logarithmically decays to zero as the distance to the landing point goes to zero. The design parameters a and b offer flexibility to the overall landing process. Figure 3.2 shows the solution represented by Equation (3.6) spanning across the three feasible quadrants in the two parameter (a, b) design space.

3.4 Analysis of the velocity profile

This section analysis the effect of design parameter variation in the three quadrants on the resulting flight profiles. This is followed by an analysis of the engagement time characteristics.

3.4.1 Velocity profiles

Using Equation (3.4), rate of change of velocity can be deduced as,

$$\dot{V_u} = V_{u0}\left(\frac{a+2bx}{1+ax+bx^2}\right)\dot{x} \tag{3.9}$$

Using Equation (3.2) and Equation (3.5) in Equation (3.9) results in,

$$\dot{V_u} = -\left(\frac{V_{u0}}{R_0}\right)\left[V_u\left(\frac{a+2bx}{1+ax+bx^2}\right)\right] \tag{3.10}$$

Equation (3.10) presents the downward acceleration of the vehicle as a function of the distance to landing point. From Equation (3.8) and Equation (3.10), it can be readily deduced that

$$\lim_{x\to 0}\dot{V_u} = \lim_{x\to 0} -\left(\frac{V_{u0}}{R_0}\right)\left[V_u\left(\frac{a+2bx}{1+ax+bx^2}\right)\right] = 0 \tag{3.11}$$

And hence, using Equation (3.11), the acceleration goes to zero, as the velocity of vehicle approaches zero near touchdown.

3.4.1.0.1 Quadrant 1 ($\mathrm{a} \geq 0, \mathrm{b} \geq 0$**)** Any feasible solution pair (a, b) satisfying Equation (3.6) for $a \geq 0, b \geq 0$ leads to,

$$\frac{a+2bx}{1+ax+bx^2} \geq 0, \quad \text{for } 0 \leq x \leq 1 \tag{3.12}$$

Using Equation (3.12) in Equation (3.10) leads to,

$$\dot{V_u} < 0, \quad \text{for } a \geq 0, b \geq 0 \tag{3.13}$$

Hence using Equation (3.13), in the first quadrant of the design space, a landing governed by Equation (3.8) is achieved with a continuously decelerating velocity profile. Figure 3.3 shows three velocity profiles for different values of a. Using Equation (3.4), it can be observed that for $0 \leq x \leq 1$, a higher value of a (and hence, using

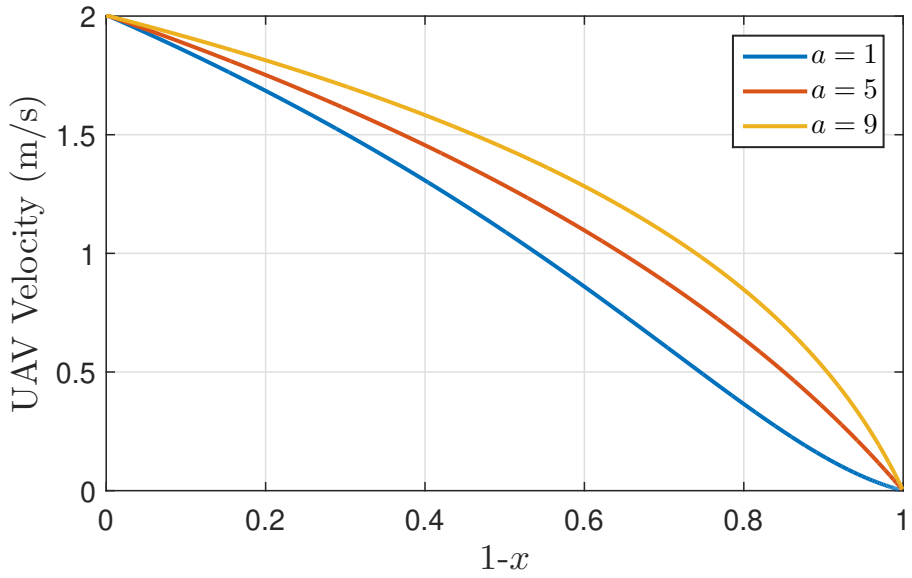


Figure 3.3: Strictly decreasing velocity profile variation with increasing values of a.

Equation (3.6), a lower value of b) leads to a lower loss of velocity with distance.

3.4.1.0.2 Quadrant 4 ($a > 0, b < 0$): Using Equation (3.6) in the component $1 + ax + bx^2$ of Equation (3.10) leads to,

$$1 + a(x - x^2) + (n-1)x^2 \Rightarrow 1 + ax + bx^2 > 0, \quad \text{for } a > 0,\ n > 1 \text{ and } 0 \leq x \leq 1 \tag{3.14}$$

Using Equation (3.10) and Equation (3.14), $\dot{V}_u \leq 0$ if,

$$a + 2bx \geq 0 \quad \Rightarrow a - 2|b|x > 0 \tag{3.15}$$

Since $0 \leq x \leq 1$, the condition,

$$2|b|x \leq 2|b|, \tag{3.16}$$

is always true. Equation (3.16) implies that Equation (3.15) will always hold true if,

$$a - 2|b| \geq 0, \ \Rightarrow a + 2b \geq 0, \ \ a > 0, b < 0 \tag{3.17}$$

Equation (3.17) governs the condition for landing with a continuous deceleration in this quadrant.

For considerable initial ranges and a low of velocity, a decelerating velocity profile can lead to a very high landing time. An initially accelerating velocity profile is desired in such cases. Using Equation (3.10), the initial rate of change of velocity can be deduced as,

$$\dot{V}_{u0} = \lim_{x \to 1} -\left(\frac{V_{u0}}{R_0}\right)\left[V_u\left(\frac{a+2bx}{1+ax+bx^2}\right)\right] = -\left(\frac{V_{u0}}{R_0}\right)\left[V_u\left(\frac{a+2b}{1+a+b}\right)\right] \tag{3.18}$$

Using Equation (3.6) in Equation (3.18) implies, $\dot{V}_{u0} > 0$ if,

$$a + 2b < 0 \tag{3.19}$$

The maximum downward velocity that can be achieved by the vehicle and the distance up to which it accelerates are controlled by choosing appropriate values of a or b that satisfy Equation (3.6) and Equation (3.19). The point at which the vehicle stops accelerating, $\dot{V}_u$ is zero and substituting that in Equation (3.10) leads to,

$$x = -\frac{a}{2b} \tag{3.20}$$

Accordingly, the maximum downward velocity can be obtained by using Equation (3.20) in Equation (3.4) as,

$$V_{u_{max}} = V_{u0} \log_n \left(1 - \frac{a^2}{4b}\right) \tag{3.21}$$

A feasible solution pair (a, b) should satisfy the constraints given in Eqns. (3.6), (3.19) for an initially accelerating profile, and it results in a maximum velocity as governed by Equation (3.21). Theorem 3 proves the existence of an initially accelerating velocity solution.

Theorem 3. *For a given V_{u0} and a desired $V_{u_{max}} > V_{u0}$, there exists a unique pair of design parameters a and b which results in an initially accelerating profile leading to a subsequent landing satisfying Equation* (3.8)*.*

Proof. Equation (3.21) can be rearranged as,

$$1 - \frac{a^2}{4b} = n^M \tag{3.22}$$

where, M is ratio $\frac{V_{u_{max}}}{V_{u0}}$ satisfying $M > 1$. Using Equation (3.6) in Equation (3.22) leads to,

$$(n - 1 - b)^2 = 4(1 - n^M)b \tag{3.23}$$

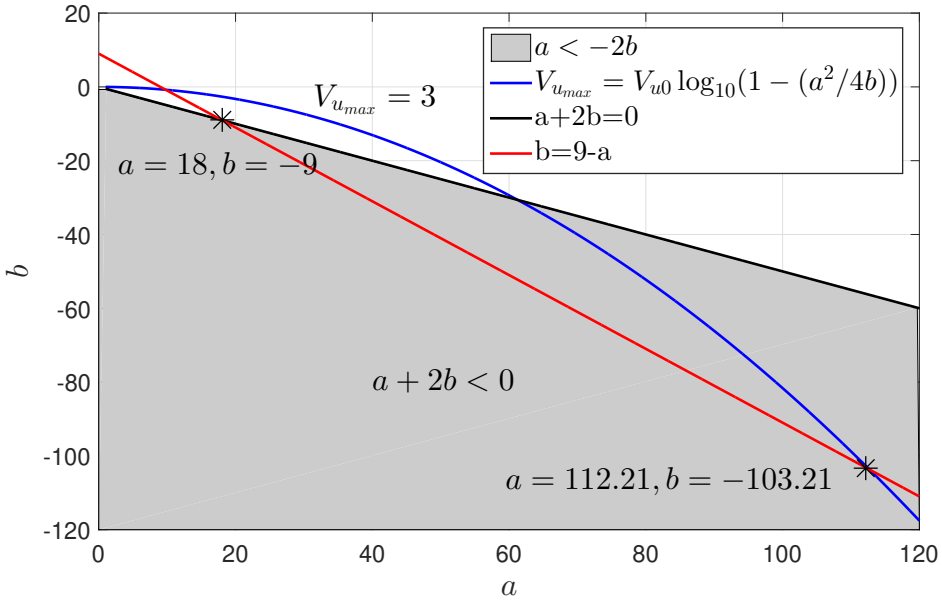


Figure 3.4: Illustration of solution space and possible solutions for accelerating profile, $V_{u_{max}}$=3 m/s.

Solving the quadratic Equation (3.23) for b results in,

$$b = (n + 1 - 2n^M) \pm 2\sqrt{(n - n^M)(1 - n^M)} \tag{3.24}$$

It can be observed from Equation (3.24) that, for $n > 1, M > 1$, b is always real and the two solutions are considered as candidates subsequently.

The first solution is given by,

$$b = (n + 1 - 2n^M) - 2\sqrt{(n - n^M)(1 - n^M)} \tag{3.25}$$

For an initially accelerating profile, substituting for this solution in Equation (3.19) leads to,

$$\begin{aligned} b = (n + 1 - 2n^M) - 2\sqrt{(n - n^M)(1 - n^M)} \leq -(n - 1) \text{ or,} \\ -\sqrt{(n - n^M)(1 - n^M)} \leq (n^M - n) \end{aligned} \tag{3.26}$$

Since both $\sqrt{(n - n^M)(1 - n^M)}$ and $n^M - n$ are positive, Equation (3.26) is true for all $n > 1, M > 1$. To analyse the uniqueness of the solutions, verifying Equation (3.6) and Equation (3.19), for the other solution of Equation (3.24) leads to,

$$b = (n + 1 - 2n^M) + 2\sqrt{(n - n^M)(1 - n^M)} \leq -(n - 1) \tag{3.27}$$

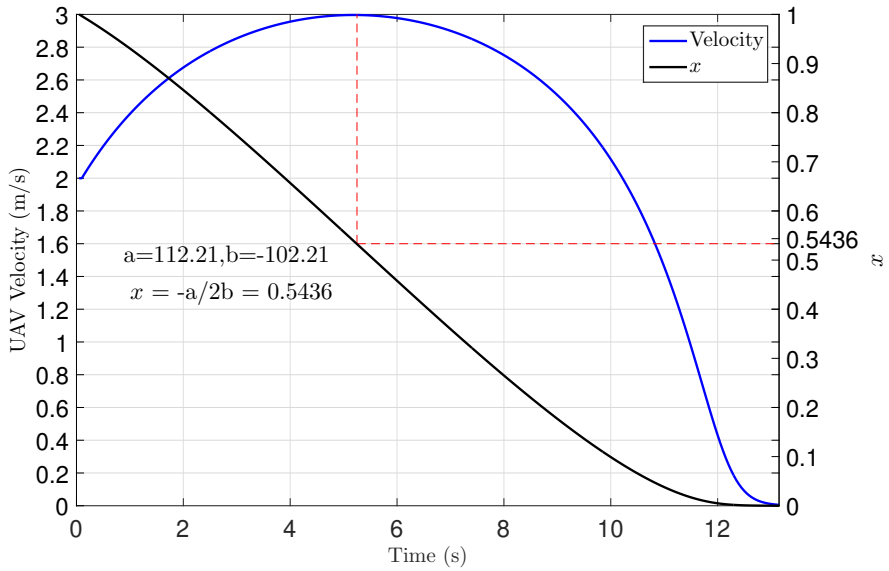


Figure 3.5: Typical velocity variation with initial acceleration.

which can be further simplified to obtain,

$$(n^M - 1) \leq (n^M - n) \tag{3.28}$$

$$\Rightarrow -1 \leq -n \tag{3.29}$$

which does not hold for $n > 1$. □

Considering the initially accelerating velocity profile, Figure 3.4 presents an example problem in the fourth quadrant of $a - b$ design space. The constants and engagements parameters are chosen as, $n = 10$, $V_{u0} = 2$ m/s, $R_0 = 30$ m, and $V_{u_{max}} = 3$ m/s, respectively. Using Equation (3.6) with $n = 10$, the solution for a and b is represented by the red line in Figure 3.4. Any feasible design pair has to lie on this line. Representing the initially accelerated profile, the constraint of Equation (3.19) is shown as the shaded region in Figure 3.4. The desired maximum velocity curve of Equation (3.21) is plotted by the blue line in Figure 3.4. The two candidate solutions are plotted by the starred marker out of which $(a, b) = (112.21, -103.21)$, satisfies all the constraints. Corresponding to the solution pair, the velocity variation is plotted in Figure 3.5. As governed by Eqns. (3.20) and (3.21), a maximum downward velocity of $V_{max} = 3$ is achieved at $x = -\frac{a}{2b} = 0.5436$.

3.4.1.0.3 Quadrant 2 ($a < 0, b > 0$): It can be noted that, another solution for $V_u = 0$ can be deduced using Equation (3.8) as,

$$x = -\frac{a}{b} \tag{3.30}$$

This solution corresponds to a zero velocity at a non zero altitude, resulting in hover state. Proposition 1 gives the proof of existence of the hovering velocity profile solution.

Proposition 1. *Subject to Eqns.* (3.6) *and* (3.4)*, solutions in the second quadrant of the design space correspond to hovering flight profiles settling at a hover altitude* $x = \frac{-a}{b}$.

Proof. For the vehicle to be in hover, $V_u = 0$ at some $x > 0$. Using Equation (3.6), the solution pair (a, b), given $a < 0, b > 0$ satisfies,

$$|a| < |b| \tag{3.31}$$

which ensures $x \leq 1$. Hence, using Equation (3.8), the hovering condition can be expressed as,

$$\lim_{x \to -\frac{a}{b}} V_u = 0; \quad a < 0, b > 0 \tag{3.32}$$

□

During the flight x varies between,

$$-\frac{a}{b} \leq x \leq 1 \tag{3.33}$$

Next, the nature of velocity profile in hover region is discussed. In the hover region,

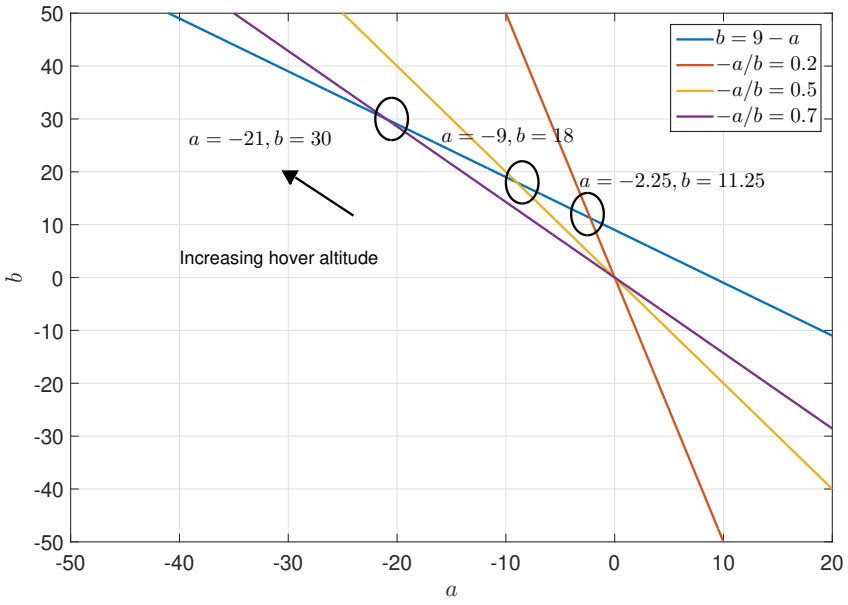
$$x \geq -\frac{a}{b} \Rightarrow ax \geq -bx^2 \Rightarrow ax + bx^2 \geq 0 \tag{3.34}$$

The minimum possible value that $a + 2bx$ can take in the hover region ($\frac{-a}{b} \leq x \leq 1$) can be obtained as,

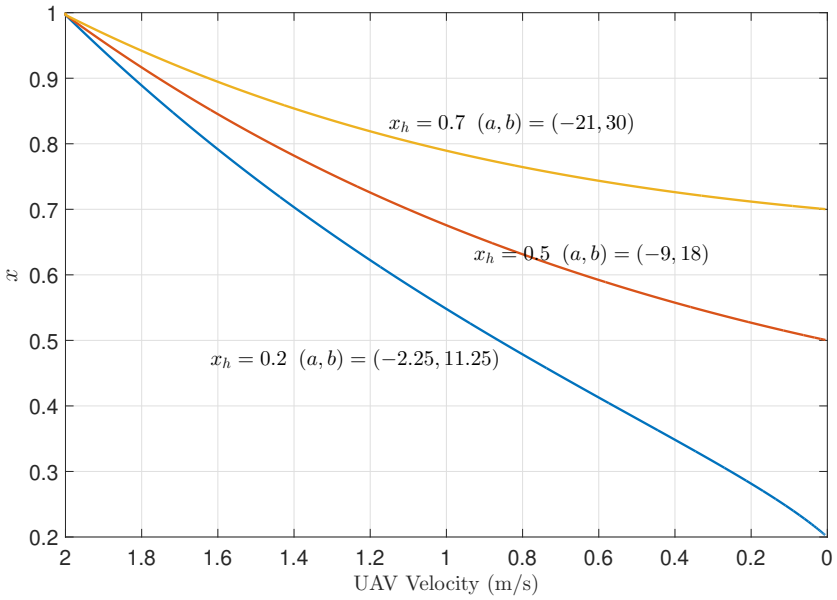
$$a + 2b\left(\frac{-a}{b}\right) = -a \tag{3.35}$$

Equation (3.35) subject to Equation (3.33) for $a < 0$ results in,

$$a + 2bx > 0. \tag{3.36}$$



(a) Solutions for hovering altitudes $x = 0.2$, 0.5 and 0.7.



(b) Normalized altitude variation for for hovering altitudes $x = 0.2$, 0.5 and 0.7.

Figure 3.6: Hovering scenarios in Quadrant 2.

Using Eqns. (3.34) and (3.36) in Equation (3.10) leads to, $\dot{V}_u < 0$ for this quadrant. Hence, the hover profile corresponds to a strictly decreasing velocity. To achieve a desired hovering altitude, a feasible solution pair (a, b) can be obtained by solving Eqns. (3.6) and (3.30) as represented in Figure 3.6(a). Using Equation (3.6) in Equation (3.30) leads to,

$$x = -\frac{a}{n-1-a}. \tag{3.37}$$

For finite but very high values of a, $n-1$ becomes negligible and hovering altitude tends to 1. As an example scenario, desired hovering altitudes corresponding to $x_1 = 0.2$, $x_2 = 0.5$, $x_3 = 0.7$ are considered and solutions pairs are plotted in Figure 3.6(a). Figure 3.6(b) shows the velocity variation for the corresponding hover altitudes.

3.4.2 Engagement Time Characteristics

Choice of design parameters (a, b) also effects the average velocity and hence the time taken to complete the landing process. Since the velocity is not an explicit function of time, an approximate average velocity of the vehicle ($V_{u_{avg}}$) can be deduced as,

$$V_{u_{avg}} = \frac{\int_0^1 V(x)\ \mathrm{d}x}{\int_0^1\ \mathrm{d}x} \tag{3.38}$$

Using Equation (3.4) in Equation (3.38) leads to,

$$V_{u_{avg}} = \int_0^1 V_{u0} \log_n \left(1 + ax + bx^2\right)\ \mathrm{d}x \tag{3.39}$$

Integral in Equation (3.39) takes the following two forms based on the value of $4b - a^2$.

$$V_{u_{avg}} = \frac{V_{u0}}{\log(n)} \left(\left[\left(x + \frac{a}{2b}\right) \log(1 + ax + bx^2) - 2x + \frac{\sqrt{4b-a^2}\left(\tan^{-1}\left(\frac{a+2bx}{\sqrt{4b-a^2}}\right)\right)}{b} \right]_0^1 \right),$$
$$\text{for } \ 4b - a^2 \geq 0,\ b \neq 0 \tag{3.40}$$

and,

$$V_{u_{avg}} = \frac{V_{u0}}{\log(n)} \left(\left[\left(x + \frac{a}{2b} \right) \log(1 + ax + bx^2) - 2x + \frac{\sqrt{a^2 - 4b} \left(\tanh^{-1} \left(\frac{a+2bx}{\sqrt{a^2-4b}} \right) \right)}{b} \right]_0^1 \right), \quad \text{for } 4b - a^2 < 0,\ b \neq 0 \tag{3.41}$$

Applying the limits to Eqns. (3.40) and (3.41), the final integral forms are obtained.

$$V_{u_{avg}} = V_{u0} \left[\frac{2\sqrt{4b - a^2} \left(\tan^{-1} \left(\frac{\sqrt{4b-a^2}}{a+2} \right) \right)}{2b \log(n)} + \frac{2b \left(\log(n) - 2 \right) + a \log(n)}{2b \log(n)} \right], \quad \text{for } 4b - a^2 \geq 0,\ b \neq 0 \tag{3.42}$$

and,

$$V_{u_{avg}} = V_{u0} \left[\frac{2\sqrt{a^2 - 4b} \left(\tanh^{-1} \left(-\frac{\sqrt{a^2-4b}}{a+2} \right) \right)}{2b \log(n)} + \frac{2b(\log(n) - 2) + a \log(n)}{2b \log(n)} \right], \quad \text{for } 4b - a^2 < 0\ b \neq 0 \tag{3.43}$$

Equation (3.39) reduces to,

$$V_{u_{avg}} = \int_0^1 V_{u0} \log_n(1 + ax)\ \mathrm{d}x = \frac{V_{u0}}{\log(n)} \left(\frac{1}{a} - \frac{\log(1+a)}{a^2} \right), \quad \text{for } b = 0 \tag{3.44}$$

Using Equation (3.6) and partially differentiating Equation (3.42) with respect to a leads to,

$$\frac{\partial V_{u_{avg}}}{\partial a} = \frac{V_{u0}}{\log(n)} \left[\frac{-1}{n - 1 - a} + \left(\tan^{-1} \left(\frac{4n - (a+2)^2}{a+2} \right) \frac{2(n-1) - a(n+1)}{(n-1-a)^2 \sqrt{4n - (a+2)^2}} \right) + \frac{\log(n)(n-1)}{2(n-1-a)^2} \right] \tag{3.45}$$

Using Equation (3.6) and partially differentiating Equation (3.43) with respect to a leads to,

$$\frac{\partial V_{u_{avg}}}{\partial a} = \frac{V_{u0}}{\log(n)} \left[\frac{-1}{n - 1 - a} + \left(\tanh^{-1} \left(\frac{(a+2)^2 - 4n}{a+2} \right) \frac{2(n-1) - a(n+1)}{(n-1-a)^2 \sqrt{4n - (a+2)^2}} \right) + \frac{\log(n)(n-1)}{2(n-1-a)^2} \right] \tag{3.46}$$

Right hand sides of Eqns. (3.45) and (3.46) were evaluated using the *fmincon* MATLAB ® solver with *Sequential Quadratic Programming* (SQP) as the input algorithm and the minimum solution was found out to be

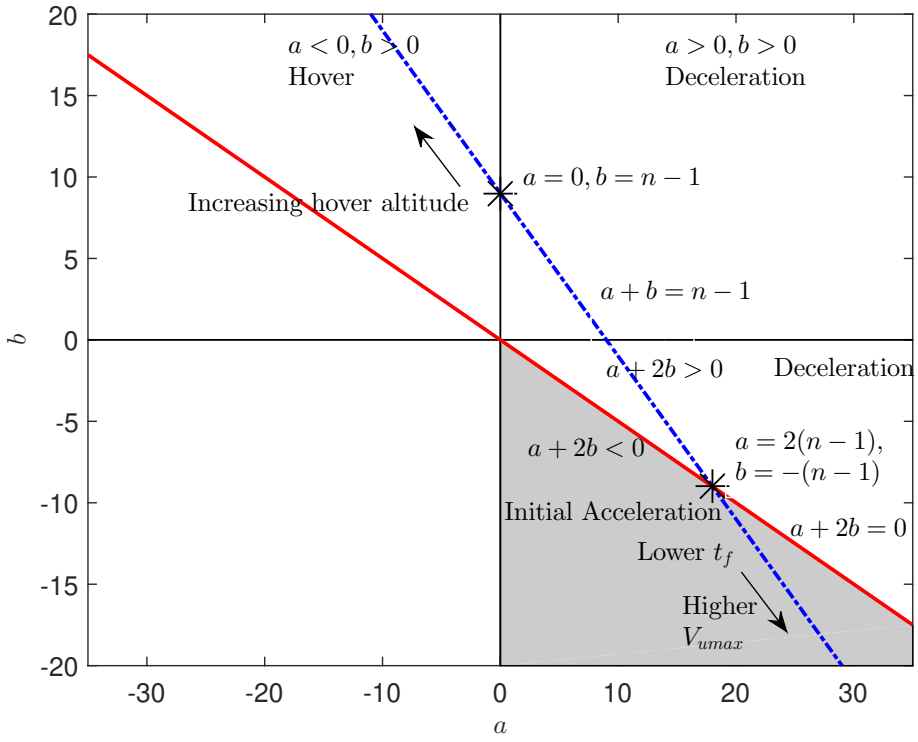


Figure 3.7: Complete solution space and various flight profiles.

greater than 0 in both the cases. Hence,

$$\frac{\partial V_{u_{avg}}}{\partial a} > 0, \text{ for } a \geq 0, \ b \neq 0 \tag{3.47}$$

Further, for the case $b = 0$, differentiating Equation (3.44) with respect to a we get,

$$\frac{\partial V_{u_{avg}}}{\partial a} = \frac{V_{u0}}{\log(n)} \left(\frac{1}{a} - \frac{\log(1+a)}{a^2} \right) \tag{3.48}$$

Using $b = 0$ in Equation (3.6) leads to,

$$a = n - 1 \tag{3.49}$$

Using Equation (3.49) in Equation (3.48) leads to,

$$\frac{\partial V_{u_{avg}}}{\partial a} = \frac{V_{u0}}{\log(n)} \left(1 - \frac{\log(n)}{n-1} \right) \tag{3.50}$$

which is greater than 0 for $n > 1$. Hence, using Eqns. (3.47) and (3.50), and it is concluded that average downward velocity is an increasing function of the design parameter a, for all $a \geq 0, n > 1$.

(a, b)	Velocity Profile
$0 \leq a \leq 2(n-1), -(n-1) \leq b \leq (n-1)$	Continuous decelerating landing profile
$a > 2(n-1), b < -(n-1)$	Initially accelerating landing profile
$a < 0, b > (n-1)$	Continuous decelerating hovering profile

Table 3.1: Intervals of (a, b) for various flight profiles

3.4.3 Summary of the Solution Regions and Discussion

Figure 3.7 shows the complete solution space consisting of different velocity profiles. The solution governed by Equation (3.6) is shown in blue. The shaded area in the fourth quadrant represents the region satisfying the initial acceleration condition given by Equation (3.19). The solution line crosses into this region at $(a, b) = (2(n-1), -(n-1))$. As the design pair moves from this point forward into the shaded region, a higher maximum velocity is achieved, which can also be observed from Equation (3.21). The solution region lying between $(a, b) = (2(n-1), -(n-1))$ and $(a, b) = (0, n-1)$ corresponds to a strictly decelerating velocity profile. Note that the magnitude of $a+2b$ increases moving from fourth to the first quadrant and this results in a higher deceleration as governed by Equation (3.10). Hovering can be achieved in the continuously decelerating profile region ($a+2b > 0$) of the second quadrant. As a increases in magnitude in this region, hovering altitude also increases as governed by Equation (3.30). Table 3.1 presents the solution intervals and corresponding flight profiles.

3.5 Simulation results

While the basic velocity profiles were simulated as part of the discussion in Section IV, additional simulations are carried out for studying the engagement time and autopilot dynamics aspects. Simulations are carried out for a vertical landing scenario with a stationary landing point. The vehicle is at a distance of $R_0 = 60$ m from the landing point and with an initial downward velocity of $V_{u0} = 2$ m/s. All simulations are carried out for the log base $n = 10$, and terminated for $R \leq 0.01$ m.

3.5.1 Case 1: Engagement Time Characteristics

This case considers the effect of the choice of the design parameters on the engagement time. The simulation also considers a maximum speed constraint of 5 m/s. The variation in landing time with a is plotted in Figure 3.8. In accordance with Eqns. (3.47) and (3.50), the simulation result shows that the engagement time decreases

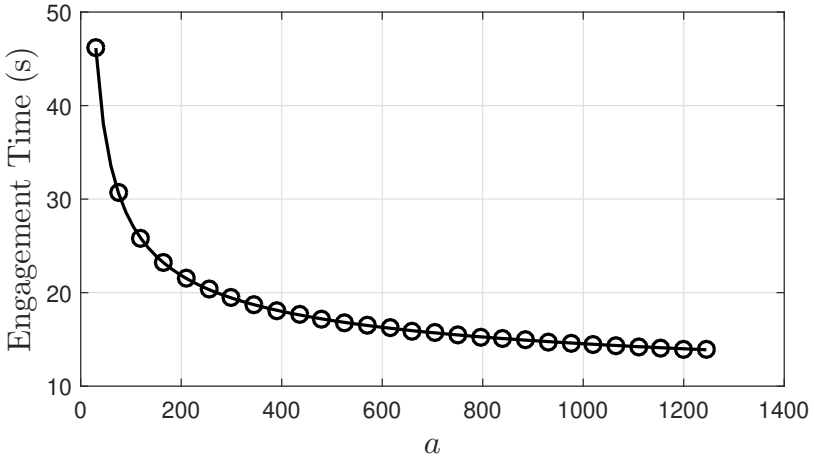


Figure 3.8: Results for Case 1: Variation of engagement time with design parameter a.

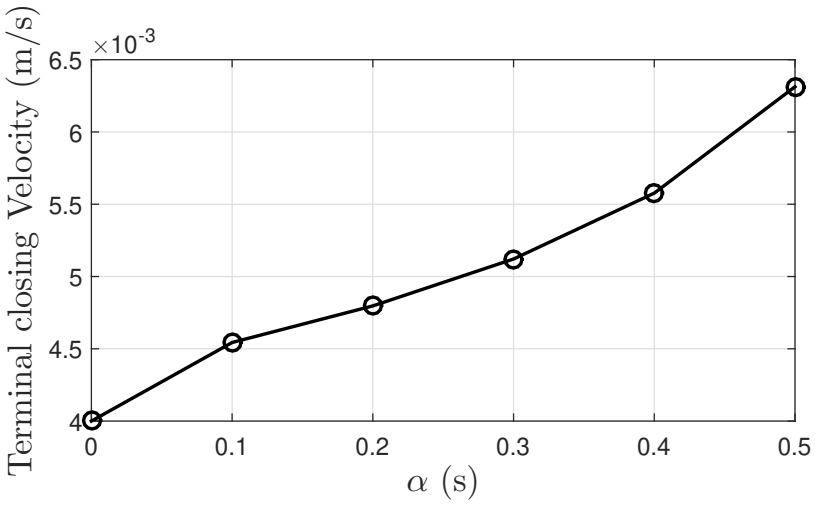
with the increasing choice of parameter a.

3.5.2 Case 2: Robustness with respect to Autopilot Lag

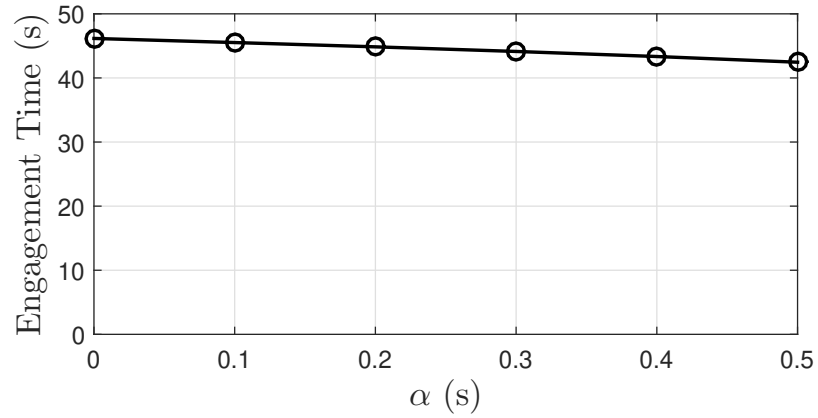
For practical applications, robustness with respect to system delays is desired for any prospective velocity profile. The velocity variation with a first order autopilot dynamics is modeled as,

$$\frac{V_{u_a}}{V_{u_c}} = \frac{1}{s\alpha + 1} \tag{3.51}$$

where V_{u_c} is the commanded vehicle velocity and α is the time constant. Simulations are carried out for $\alpha = 0.1$ s to $\alpha = 0.5$ s. Terminal closing velocity variation is plotted in Figure 3.9(a) showing negligible impact speeds for the range of α. The robustness is attributed to the fact that the proposed velocity profile of Equation (4) is a function of the instantaneous distance to the landing point. Due to this inherent feedback nature of the commanded velocity profile, the vehicle performs desired landing in the presence delays caused by autopilot dynamics. Variation in engagement time is plotted in Figure 3.9(b). Slower deceleration caused due to autopilot delay has an overall impact of a lower engagement time for higher values of α. Results highlight the robustness of the proposed velocity profile with respect to uncompensated systems delays.



(a) Terminal closing speed variation with time constant



(b) Engagement time variation with time constant

Figure 3.9: Results for Case 2: Robustness with respect to autopilot dynamics.

3.6 Conclusions

The log polynomial velocity profile method allows one to propose a desired velocity profile which is a log of polynomial function of the normalized distance to the landing point with two design parameters. Closed-form design parameter solutions are derived for strictly decreasing, initially accelerating and and hovering velocity profiles. Approximate landing time and maximum velocity expressions are derived as a function of the design parameters. Simulations are presented for generating the flight profiles and verifying the robustness of the proposed method to uncompensated autopilot dynamics. The method offers significant advantage in simultaneously analysing multiple flight constraints in the two design parameter space and could potentially address wider future applications including planetary landing scenarios.

Chapter 4

Compensated trajectory shaping guidance for landing and qualitative robustness evaluation

In Chapter 2, we developed a pursuit based guidance law with a log polynomial function based velocity controller for autonomous landing with vision. We showed the efficacy of the landing system using simulations and hardware experiments. However, the landing system is limited to landing on flat regions and the approach towards the target is fixed due to the nature of the pursuit guidance law. If is often essential to have some flexibility in the approaching direction of the vehicle towards the target due to terrain restrictions. For instance, landing on a moving target which is moving in on an incline surface due to terrain. In such scenarios, the previously proposed approach may not always be useful. Therefore, to address this issue, we developed a compensated trajectory shaping guidance for landing on targets moving on inclined surfaces. Further, there is a need to evaluate extensively the landing methodology to ensure the controller degrades gracefully in unforeseen conditions, and a decision to abort and/or reinitialize the landing process can be made accordingly.

In this chapter, we describe the compensated trajectory shaping guidance for landing and evaluate the robustness of the autonomous landing on a moving target using the proposed guidance law along with log polynomial velocity profiler. We describe the performance based on simulations and hardware experiments.

Figure 4.1: Hardware setup consisting of the quadrotor attempting to land on the target mounted on a ground rover. The quadrotor is equipped with a camera (point Grey Chameleon 3) and Odriod U3+ embedded computer

4.1 Landing approach using trajectory shaping guidance

The aim of the guidance law is to intercept the target accurately. Trajectory shaping guidance [101] is used in applications, where it is desirable to shape the trajectory at the time of interception. For example, to intercept the target at a desired approach angle. In our proposed framework we explore these desirable properties of the trajectory shaping guidance to achieve autonomous landing of a quadrotor on a moving target.

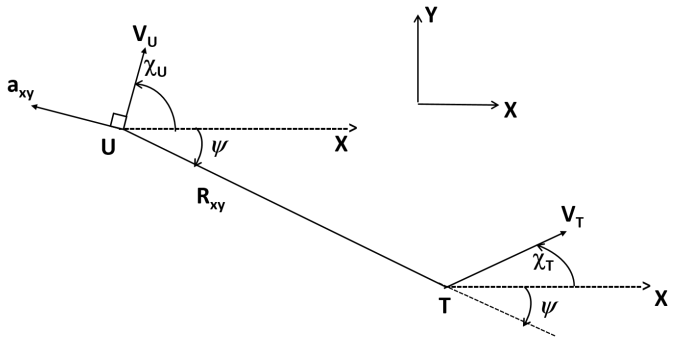


Figure 4.2: Planar engagement geometry of UAV-target.

The trajectory shaping guidance is given by,

$$a_{xy} = 4V_u\dot{\psi} + \frac{2V_u\left(\psi - \psi_f\right)}{t_{go}}, \tag{4.1}$$

where t_{go} is the estimated time to land on the moving target. The first term is the proportional navigation

component of the guidance law. Proportional navigation works on the principle that UAV velocity vector should rotate at a rate proportional to the LOS rate and in the same direction in order to eventually drive the LOS rate to zero and intercept the landing target. The second term drives the LOS angle to a desired impact angle. The combination of these terms gives a better control of the landing trajectory as compared to pure pursuit, especially in case of complex target trajectories or landing on an inclined plane. However, the trajectory shaping guidance is derived under the assumption of a compensated V_u. Though this approach gives a trajectory and terminal impact angle control to land on a moving target, an explicit control over UAV speed is required for a successful landing.

4.1.1 Compensation for variable speed

The condition to drive the LOS rate ($\dot{\psi}$) to zero can be obtained from equation (2.2) as,

$$V_t \sin(\chi_t - \psi) = V_u \sin(\chi_u - \psi) \tag{4.2}$$

let $(\chi_u - \psi)$=δ. Then,

$$\dot{\delta} = \dot{\chi}_u,\ \dot{\psi} = 0 \tag{4.3}$$

Differentiating equation (4.2) on both the sides with respect to time, we get,

$$V_t \cos(\chi_t - \psi)\dot{\psi} = V_u \cos(\delta)\dot{\delta} + \dot{V}_u \sin(\delta), \tag{4.4}$$

substituting, $\dot{\psi} = 0$ in the above equation, we get,

$$V_u\dot{\delta} = -\dot{V}_u \tan(\delta) \tag{4.5}$$

Since aim is to drive the LOS rate to zero, the lateral acceleration due to a variable speed must be compensated for, in the guidance term. Thus, the complete guidance term for compensated trajectory shaping guidance principle is given by:

$$a_{xy} = 4V_u\dot{\psi} + \frac{2V_u\left(\psi - \psi_f\right)}{t_{go}} - \dot{V}_u \tan\left(\chi_u - \psi\right) \tag{4.6}$$

To the best of our knowledge, an integrated guidance framework with a speed, trajectory control with speed compensation and impact angle control has not been developed for autonomous landing applications on a

moving target.

4.1.2 Example scenario

We now show that the compensated trajectory shaping guidance is better than no compensation through a simulated example scenario. Figures 4.3(a) shows the trajectory of the quadrotor landing on a moving target with and without compensation in trajectory shaping guidance. From the figure, we can see that the compensated UAV trajectory is smooth and takes less time to land compared to no-compensation condition. Further, from Figure 4.3(b), we can see that the compensated trajectory shaping guidance approaches the desired LOS angle earlier that no-compensation guidance law. The simulations were carried out using MATLAB with the following parameters $V_{u0} = 2m/s$, $V_t = 1m/s$, $\psi_f = -90\,\mathrm{deg}$ and terminal criteria was $R_{xy} \leq 0.01m$. From this simple simulation, it can be seen that trajectory followed by the UAV with compensation can help in achieving efficient landing.

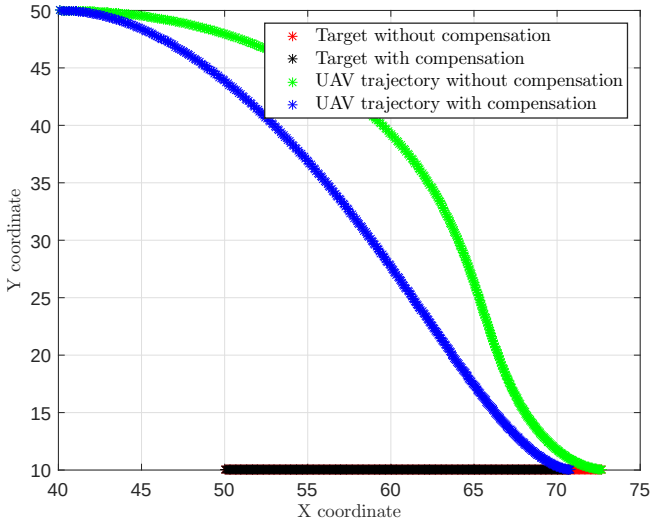
4.2 Robustness evaluation

The developed compensated trajectory shaping guidance law needs to be evaluated for robustness under different target motion characteristics through simulation and experiments. Initially, we will evaluate the performance in simulation followed by the experimental validation.

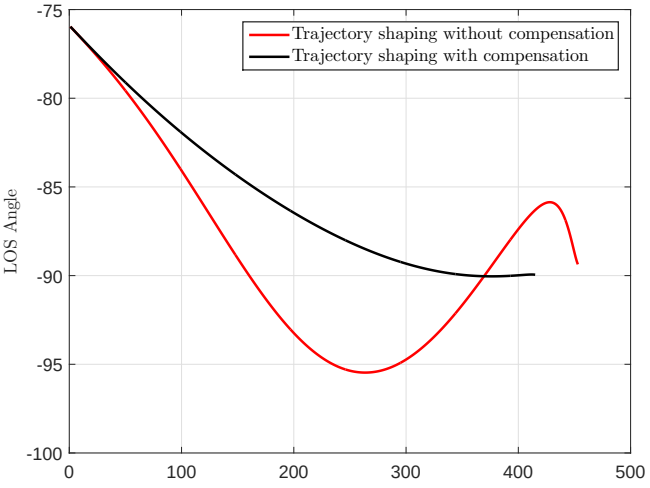
4.2.1 Simulation-based evaluation

4.2.1.1 Simulation setup

In order to validate the robustness of the guidance law with the log polynomial velocity profiler, we use the Microsoft Airsim [102] simulator for evaluation. Airsim is a MAV simulator that uses Unreal Engine [103] as its base, thus being able to simulate realistic scenes and environment. The quadrotor vehicle modeled in airsim is exposed to various physical phenomena such as gravity, magnetic field etc and consists of an efficient physics engine that can run its update loop at high frequency, which is desirable for enabling realtime simulation scenarios such as high speed quadrotor control. Further, the MAV simulated within airsim had a forward and downward facing camera and sensor models such as accelerometer, gyroscope, barometer, magnetometer and GPS. Onboard stream (of 640x480 resolution) of the downward camera was used for target detection. We utilize open-source libraries available in OpenCV in order to implement vision based landing target detection.



(a)



(b)

Figure 4.3: a)Trajectory comparison of trajectory shaping guidance with and without speed compensation, $\psi_f = -90\,\text{deg}$, $\alpha_{u0} = 0\,\text{deg}$. b) LOS angle profile comparison with and without speed compensation , $\psi_f = -90\,\text{deg}$, $\alpha_{u0} = 0\,\text{deg}$

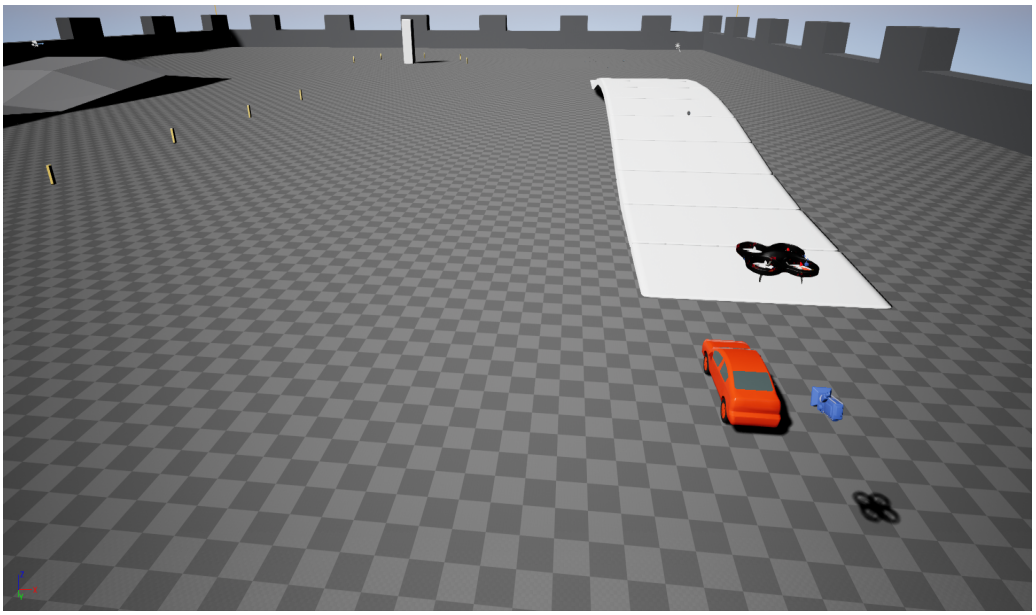


Figure 4.4: A snapshot of the simulation setup consisting of the airsim flying quadrotor vehicle and a ground vehicle as the landing target

The landing target was in the form of a moving car simulated in unreal engine. Figure 4.4 shows a screenshot of the simulation environment consisting of the flying quadrotor vehicle modeled in airsim and a ground vehicle which served as the moving landing target.

Using the Airsim simulator, we evaluate the robustness of the landing in the presence of wind disturbances. In each scenario, the quadrotor is initialized at an altitude of 20 meters. The landing target is a red colored moving ground vehicle and it is detected using the downward camera of the Airsim quadrotor. The target is assumed to be of known color and detection is performed using color based blob detection. The different scenarios for the evaluation are (a) landing on the straight moving target (b) landing on an inclined slope (c) landing on a target which is moving in a circular trajectory. The evaluation of these scenarios is given in detail below.

4.2.1.2 Landing on a straight line moving target

In this case, we evaluate the performance of the system when the target is moving in a straight line motion with a constant speed. Simulations with different target speeds of 1m/s, 2 m/s and 3 m/s are carried out. Figure 4.5 shows the trajectory followed by the quadrotor with the vision based target input when the target is moving with a constant speed of 2 m/s. With vision in the loop, the quadrotor follows an efficient trajectory towards the target. Figures 4.8 and 4.9 show the closing velocity profile and the descend velocity of the UAV respectively. It can be observed that both converge to zero as the vehicle lands on the target.

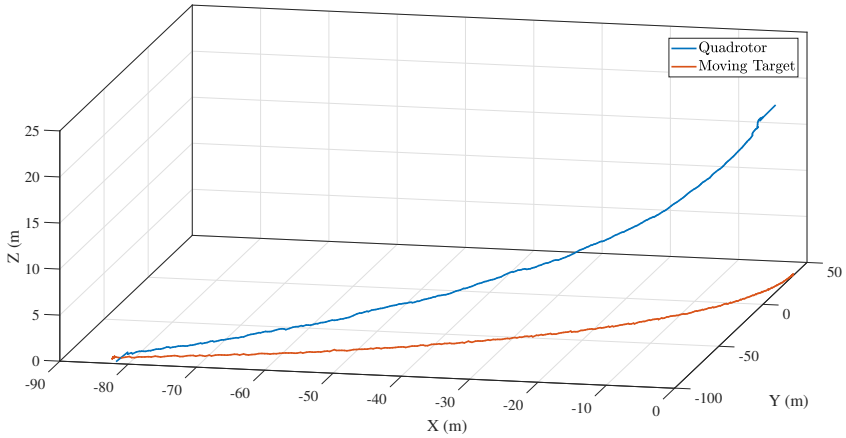


Figure 4.5: Landing trajectory followed by UAV when ground vehicle moves in a near straight line trajectory at 2 m/s.

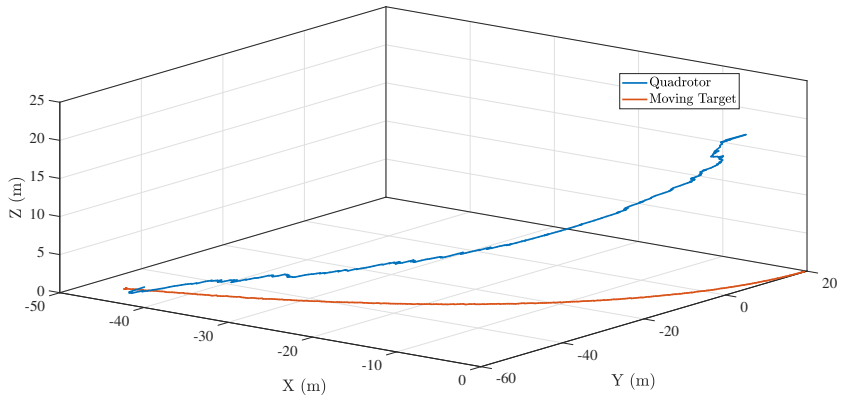


Figure 4.6: Landing trajectory followed by UAV when ground vehicle moves in a near straight line trajectory at 1 m/s.

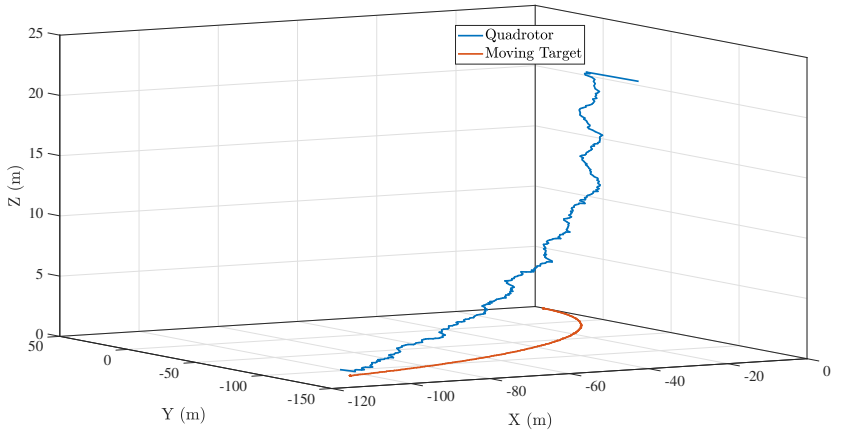


Figure 4.7: Landing trajectory followed by UAV when ground vehicle moves in a near straight line trajectory at 3 m/s.

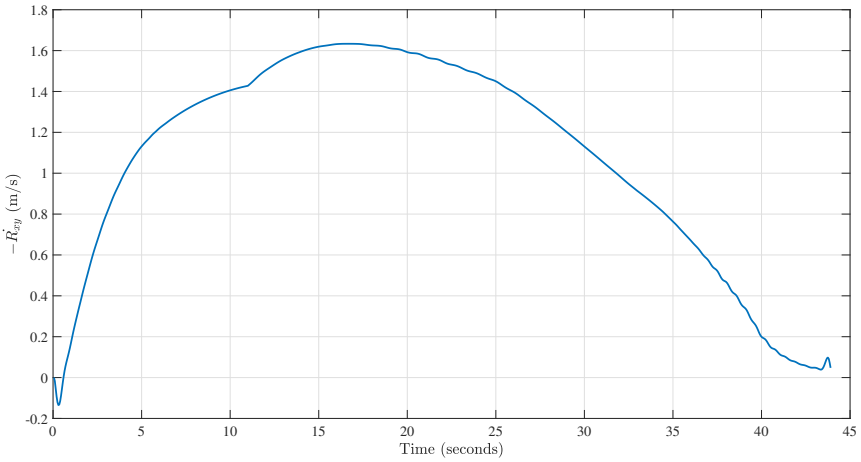


Figure 4.8: Closing velocity profile (xy plane) of the UAV when vehicle moves in a near straight line trajectory at 2 m/s.

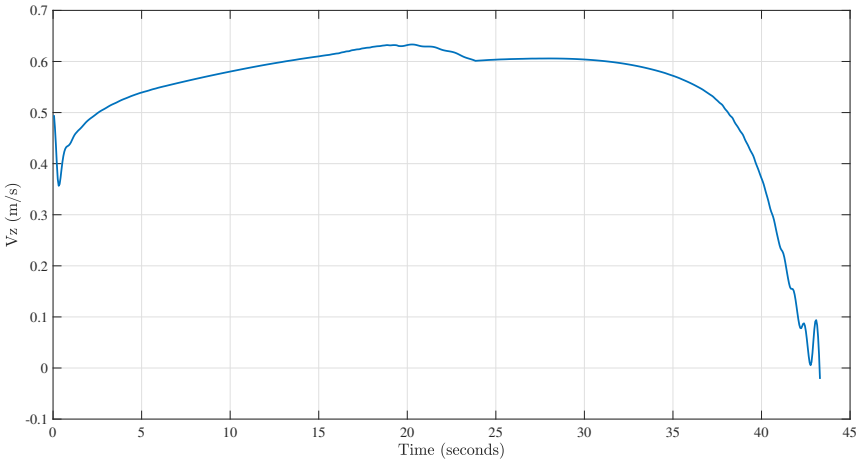


Figure 4.9: Descend velocity of the UAV when ground vehicle moves in a near straight line trajectory at 2 m/s.

When the target speed is 1m/s or 3m/s, the quadrotor was able to land on the target successfully as shown in Figures 4.6 and 4.7 respectively. The velocity profiles are similar to that of the simulation with the target speed of 2m/s and hence we do not show the trajectories.

4.2.1.3 Landing on a circular trajectory moving target

In the case, we analyze the performance of the quadrotor landing when the target is moving in a circular trajectory of radii 5 and 15m with speed of 2m/s. The quadrotor tracks the target's trajectory to attempt to land on it successfully. Figure 4.10 shows the trajectory followed by the vehicle while landing on the target successfully. As observed from the figure, the quadrotor tracks the trajectory of the target vehicle and efficiently

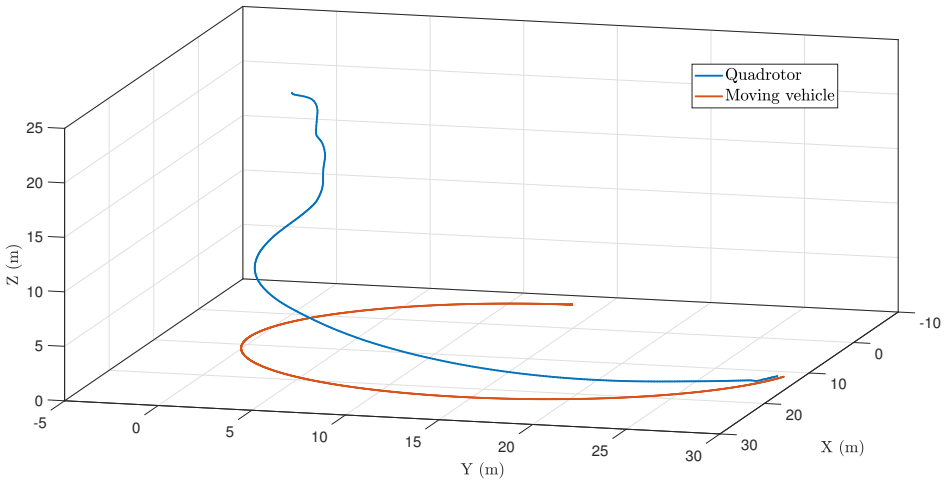


Figure 4.10: Landing trajectory followed by UAV when ground vehicle moves along a curved path (15 m radius).

adjusts itself to follow along the same path and finally land on the target accurately. Figures 4.11 and 4.12 show the closing velocity profile and the descend velocity of the UAV respectively. It can be observed that both converge to zero as the vehicle lands on the target. Figure 4.13 shows a similar trend of trajectory, followed by successful landing when the landing target was made to follow a circular path of smaller radius (5 m approx.)

The results from three scenarios show that the proposed algorithm is robust against all variations and performs efficiently. For moving targets, this is verified with varying target speeds in the range of 1m/s to 3m/s. Further, the quadrotor adjusts its maneuvering in presence of dynamically moving targets with disturbances like non uniform surfaces, complex ground vehicle trajectories along curved paths and is able to successfully land on targets in outdoor environment.

We tried to analyze the landing when the target is performing a tight loiter at 2m/s and the loiter radius in 5m. Figure 4.13 show the trajectory of the vehicle tracking the target and descending towards the target. The resultant simulation was successful. Since the curvature of the target is high, the quadrotor's descent rate is not monotonically decreasing.

4.2.1.4 Landing on an inclined slope

The last deals with landing on inclined slope. In this case, the target slowly travels on an inclined slope. This test shows the performance of the vehicle against terrain changes. Robustness of the landing on an inclined surface was tested by introducing a $30\,\mathrm{deg}$ slope in the target trajectory. Target was made to follow a straight line trajectory with the slope in between at a constant speed of $2m/s$. Figure 4.14 shows the performance

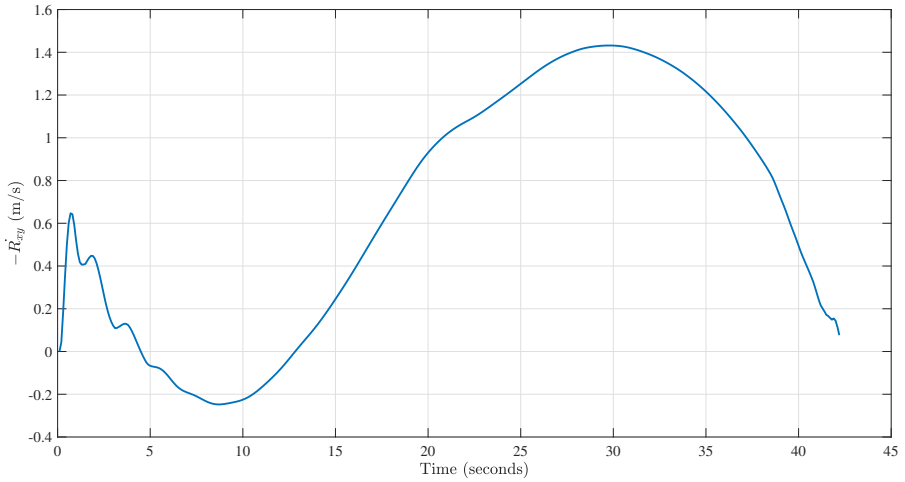


Figure 4.11: Closing velocity profile (xy plane) of the UAV when ground vehicle moves along a curved path (15 m radius).

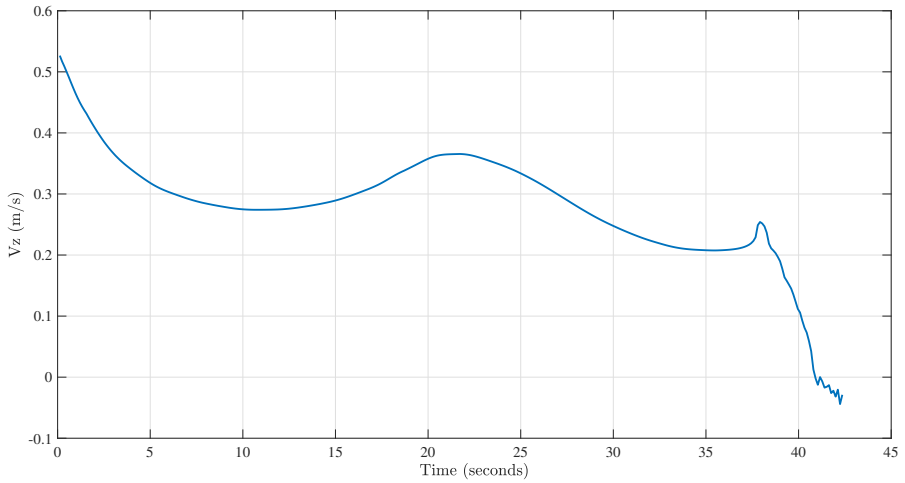


Figure 4.12: Descend velocity of the UAV when ground vehicle moves along a curved path (15 m radius).

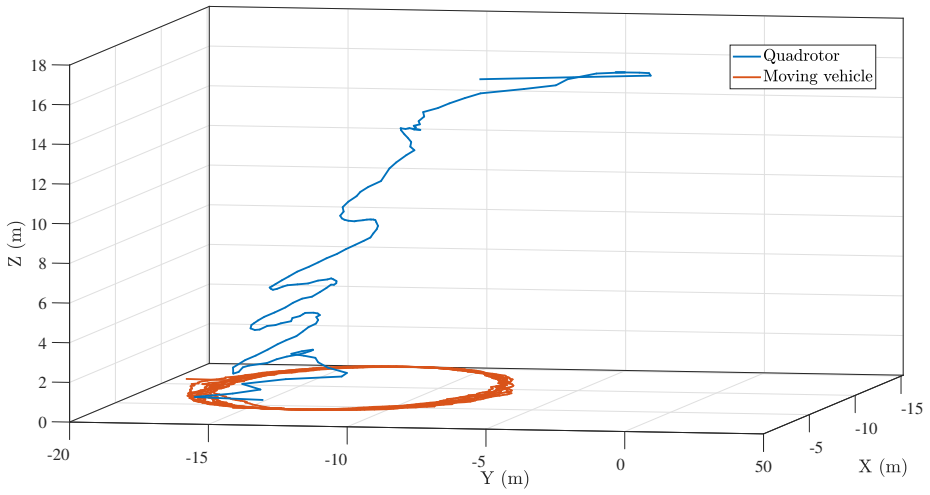


Figure 4.13: Landing trajectory followed by UAV when ground vehicle moves along a curved path (5 m radius)

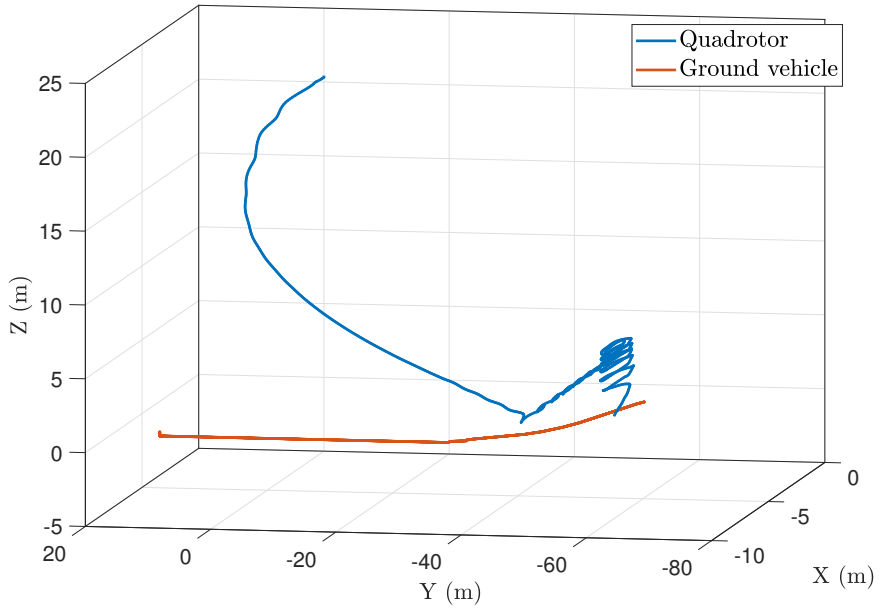


Figure 4.14: Trajectory followed by the quadrotor while trying to land on a non uniform surface.

of the guidance algorithm, when attempting to land on a non uniform surface. The disturbance in quadrotor trajectory at the end is due to change in altitude of the target. The proposed method, requires knowledge of terminal desired LOS angle (surface inclination). In this simulation, we assumed that we have no knowledge of the slope, instead it is calculated continuously by the change in quadrotor altitude. Since this estimate is continuously changing as the quadrotor moves up the slope, it leads to some disturbance in the quadrotor trajectory at the time of landing. It can be seen from Figure 4.15 that the magnitude of LOS angle at the time of landing is close to the desired terminal LOS angle ($90\,\mathrm{deg} -$angle of inclination $= 60\,\mathrm{deg}$).

4.2.2 Experimental evaluation

We have performed a series of experiments that involved outdoor data collection, data processing and closed loop controller validation with onboard processing. Hardware setup along with the target detection and tracking methodology is explained in Sections 2.5 and 2.6. The hardware setup along with the landing target is shown in Figure 4.1. We perform two sets of experiments to validate the efficacy of proposed controller; the first set of experiments is aimed at validating the compensated trajectory shaping guidance for moving target and second set of experiments test the robustness of the said controller in the presence of external disturbance. In the first set of experiments, the UAV starts at an initial distance of $R0$ from the target and an initial altitude of 15 meters. Target is moving in a straight line with a nearly constant ground speed of 0.5 m/s. These experiments are conducted in the presence of negligible wind conditions or any external disturbances. Figures 4.16 and 4.16b show the UAV-target line-of-sight (LOS) distance profile ($R0 = 12$) (in XY plane) and relative velocity profile

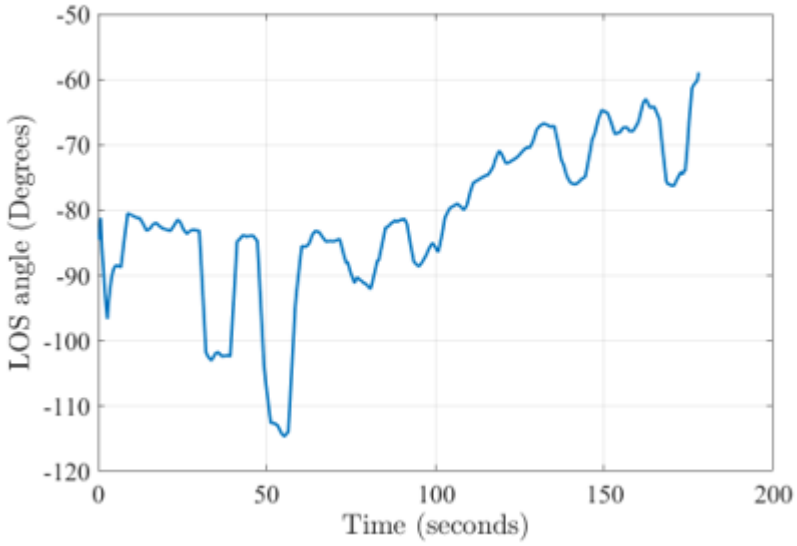
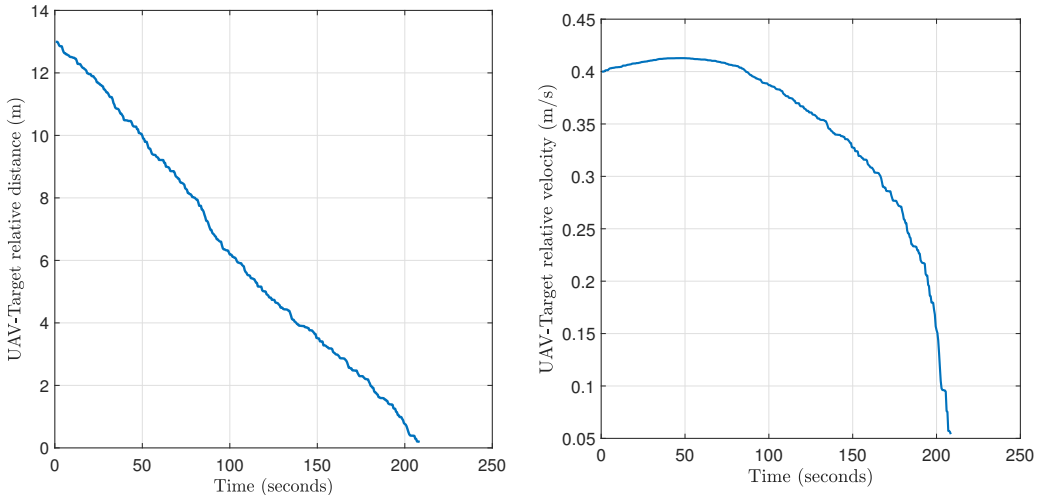
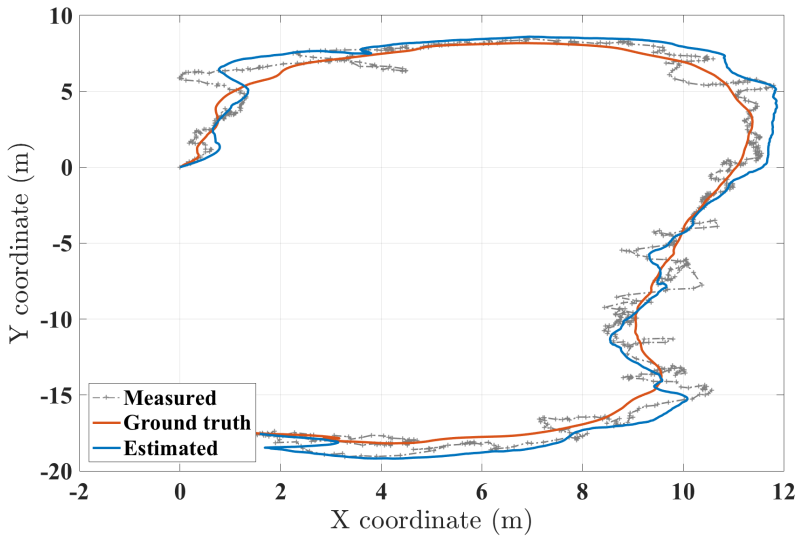


Figure 4.15: LOS angle profile (Terminal desired LOS angle : -60 degree)

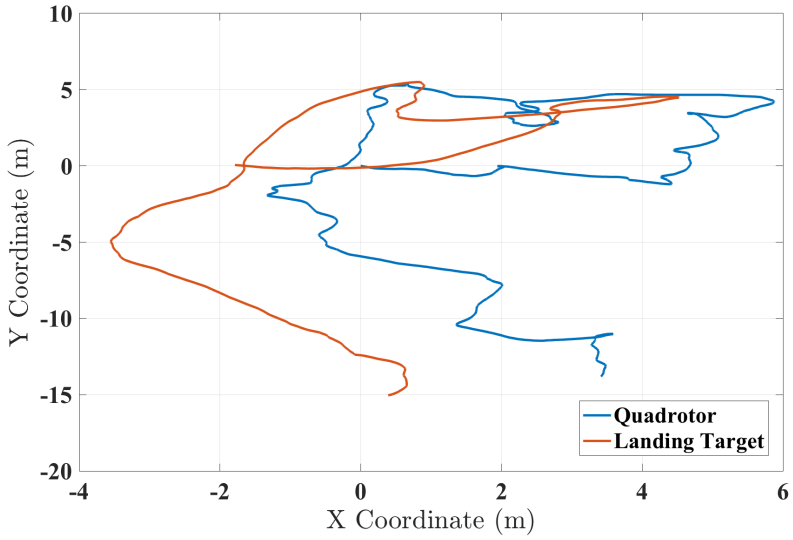


(a) (b)

Figure 4.16: (a) LOS distance profile and (b) relative velocity profile of the quadrotor when tracking a target in a straight line

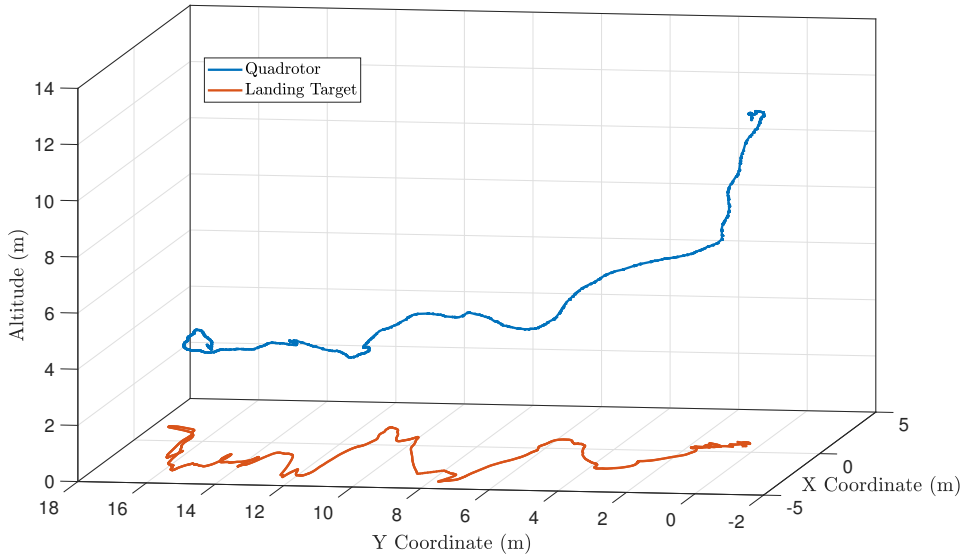


(a)

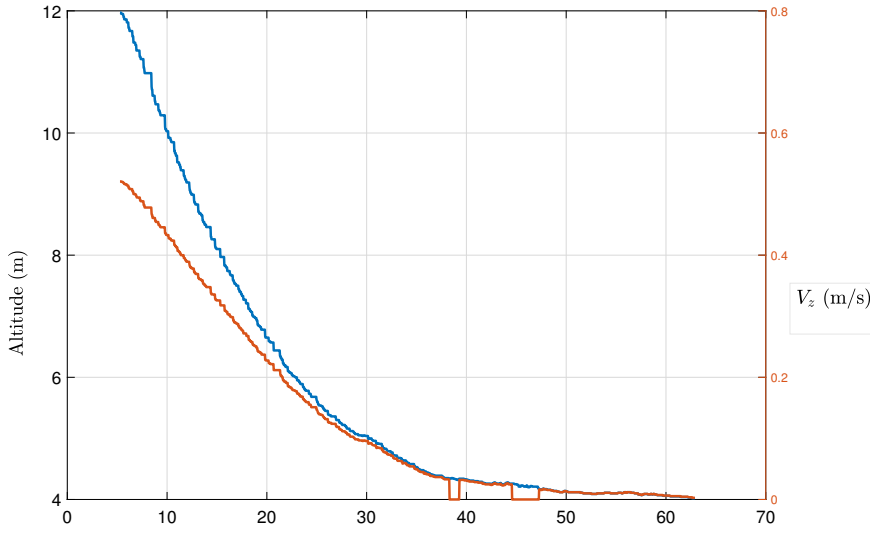


(b)

Figure 4.17: a) Trajectory of the ground rover as estimated by the quadcopter b) Trajectory of the quadrotor tracking the landing target in the presence of high winds (5m/s steady)



(a)



(b)

Figure 4.18: (a) UAV trajectory and (b) descent velocity while tracking the target and descending till 4 meters from the ground

respectively. It can be seen that, as the LOS distance decreases to zero, relative velocity between UAV and target also decreases to zero. The log of polynomial controller provides a fine grained control of UAV velocity during the landing process. Using this, an initial accelerating profile ($a = 30, b = -21, n = 10$) is designed to track the target in a time efficient manner, while a continuously decelerating profile ($a = 6, b = 3, n = 10$) is chosen as UAV descends onto the target.

The second set of experiments are aimed at testing the robustness of the landing process. These set of experiments are comprised of different scenarios pertaining to target trajectory and the presence of wind. In the first experiment, landing target was made to follow a random trajectory with a near constant speed and the quadrotor was made to manually follow the target at a constant altitude of 12 meters. This experiment was performed in the presence of strong wind conditions with gusts as high as 6 m/s [106]. We have first processed the image data offline. A Kalman Filter is used to obtain a robust estimate of trajectory from noisy measurements. Figure 4.17a shows the ground truth, estimated and measured trajectories followed by the landing target. We have then performed a landing experiment in similar high velocity wind conditions, where the loop between vision and control is closed and the processing is done onboard to test the tracking. In this case, both UAV and target start at the same position in XY plane. UAV is initially at an altitude of 12 meters above the target. The target is maneuvered manually in random trajectory and UAV tracks the target. Figure 4.17b shows the trajectory followed by the quadrotor and the landing target. Although, an offset can be observed between the two trajectories due to high winds, it can be seen that even in the presence of disturbance, UAV follows a similar trend as the target, highlighting the controller's efficiency in tracking the target.

In the final experiment, target is maneuvered manually in a random sinusoidal trajectory. UAV was commanded to follow the target based on vision measurements and descent till an altitude of 33% of its initial altitude. All other experimental conditions are the same as the previous scenario. Figures 4.18a and 4.18b show the trajectory followed by the UAV and the descend velocity profile. In this case for descent velocity, hovering velocity profile parameters ($a = -4.43, b = 9.43$) were chosen in the log polynomial controller. It can be observed that the UAV tracks the target while descending to an altitude of 4 meters. At this point V_z becomes zero and UAV continues to track the target at a constant altitude.

4.3 Discussion

From our experiments, we observed that environmental disturbances such as wind and abrupt changes in target speed or trajectory result in similar effects from the perspective of a camera image. For example, high winds or target changing its speed both result in estimation errors or target moving out of camera field of view.

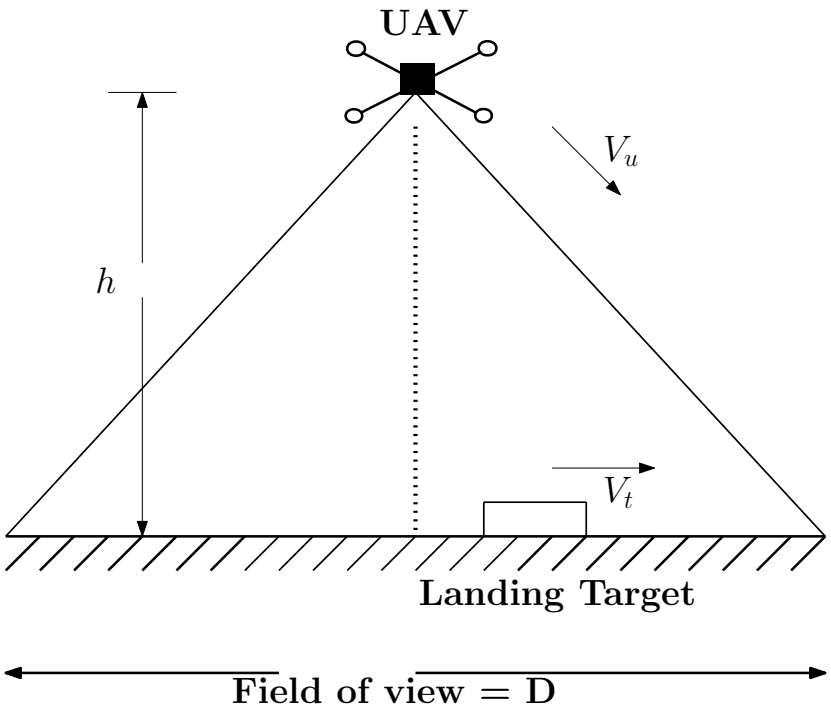


Figure 4.19: A depiction of a quadrotor trying to land on a moving target with velocity V_t

Thus, studying the effect of perturbations from computer vision perspective aids in evaluating the conditions to abort and/or reinitialize the landing process, reducing the dependence on knowledge of source of external disturbance. This helps towards developing robust landing strategies, agnostic to external disturbance. We present our approach here with a brief example of a mid-landing scenario. Consider a landing scenario as shown in Figure 4.19, where a quadrotor (with a velocity V_u), and with a downward facing camera, is attempting to land on a target moving in straight line with velocity V_t. At time t, quadrotor is at a distance of R from the landing target and an altitude h meters above the ground. Let D be the field of view of the camera in meters at a height h above the ground at time t. The condition for the landing target going out of field of view can be expressed as, $R \geq D \pm \delta, -\dot{R} > 0$. Given such a scenario, the target remains within the field of view of the as long as the UAV velocity is higher than the target velocity and a near zero relative velocity is maintained in the final stages of landing. Given the knowledge of camera parameters and quadrotor velocity profile, the condition for the target to go out of field of view can be estimated.

Whenever the target moves out of FOV, we command a zero z velocity. Let h_{OF} be the altitude at which the target went out of field of view. At this instant if V_t was the estimated target velocity and V_u was the calculated commanded velocity, then time required by the target to cover D_{OF} meters (which is the field of view of camera when the target was lost) is given by,

$$T = \frac{D_{OF}}{V_t} \tag{4.7}$$

UAV is commanded a velocity V_u based on the estimated target velocity, such that $V_u \geq V_t$. If the target doesn't come in the view even after t seconds, such that $t \geq T$, it can be safely said that the target has been lost and landing must be aborted or reinitialized using some strategy, for example increasing the altitude.

4.4 UAV rendezvous application

In the previous sections, we demonstrated the capability of log polynomial velocity controller for a robust landing when integrated with compensated trajectory shaping guidance. As concluded in Chapter 3, log polynomial controller offers significant advantage in simultaneously analyzing multiple flight constraints in the two design parameter space and could potentially address wider future applications including planetary landing scenarios. Docking is another operation, which plays an important role in multiple practical applications, for example energy refueling in case of a long flight. In this section, we show the application of proposed controller for potential docking applications. We perform MATLAB simulations using the proposed controller integrated with compensated trajectory shaping guidance for both stationary and moving targets and show the efficacy of the controller for different desired terminal LOS angles (ψ_f).

Simulation results

In this section we present the result of 2D Matlab simulations. For stationary target case, V_{u0} and R_0 were initialized to $2m/s$ and $41meter$ respectively. Termination criteria was $R \leq 0.01$ m. Performance of the guidance law was evaluated for $\psi_f = 80, 120, -90, -120$ and -170 deg. Figure 4.20 shows the trajectories followed by the UAV for intercepting the stationary target for different ψ_f. Figures 4.21 and 4.22 show the closing velocity and LOS angle profiles respectively. It can be seen that the desired ψ_f is achieved in all the cases with closing velocity going to zero. Note that in the case of a stationary target the log polynomial controller speed variation is given by,

$$V_u = V_{u0} \log_{10}(1 + ax + bx^2 + cx^3), \tag{4.8}$$

where V_{u0} is the initial UAV velocity. For moving target case, V_t was a constant equal to 1m/s and the guidance law was tested for $\psi_f = 90, 120, -90, -120, -170$ deg. Rest of the simulation parameters were same as the stationary case. Figure 4.23 shows the trajectories followed by the UAV for intercepting the moving target for different ψ_f. Figures 4.24 and 4.25 show the closing velocity and LOS angle profiles respectively. It can be seen that the desired ψ_f is achieved in all the cases with closing velocity going to zero for a moving target as

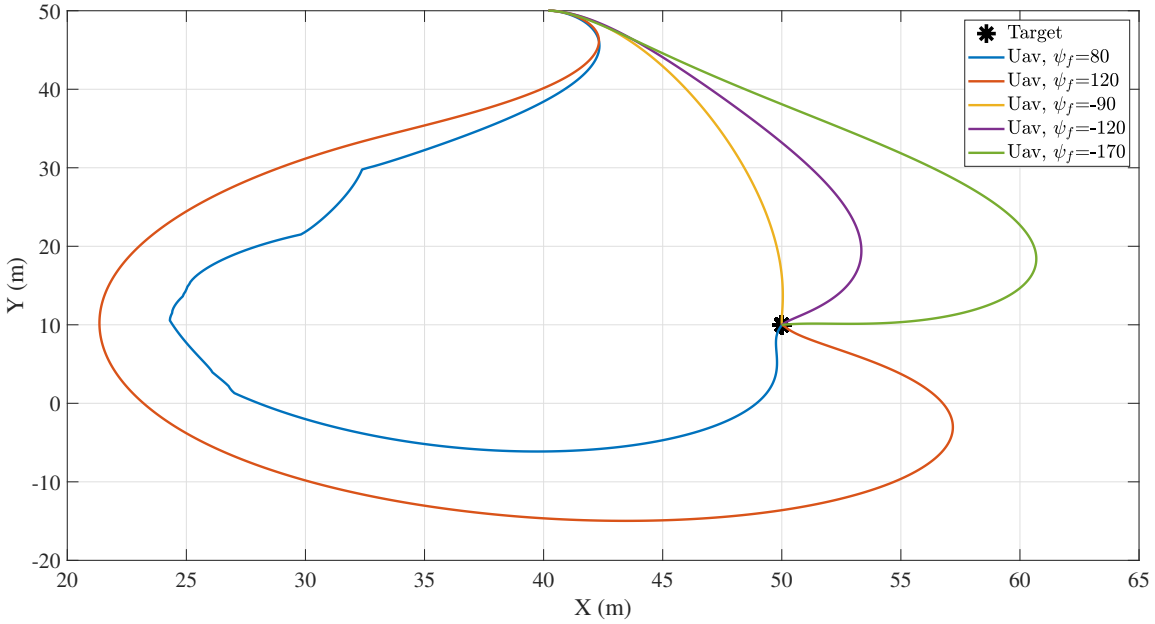


Figure 4.20: Trajectory followed by UAV to intercept a stationary target for different ψ_f

well.

Additionally, we also examined the performance of the guidance principle for variation in log polynomial controller parameters for a fixed ψ_f. This was done for a stationary target scenario, for $\psi_f = -150\,\text{deg}$. Other Simulation parameters were the same as the previous cases. Figures 4.26 and 4.27 show the trajectory and closing velocity profiles of the UAV for $(a_1, b_1, c_1) = (10, 150, -191)$ and $(a_2, b_2, c_2) = (70, 130, -191)$ respectively. It can be observed that, closing velocity in both the cases converges to zero. Also, for higher value of a and corresponding lower value of b, time to intercept the target decreases and a higher $V_{u_{max}}$ is achieved. This is in agreement with the analytical results in Chapter 3. Also, it can be observed from the trajectory variation that compensation to the trajectory is applied in accordance with the magnitude of $\dot{V}_u$.

4.5 Conclusions

In this chapter, we proposed a compensated trajectory shaping guidance law that enables the quadrotor to land on the target having different motion characteristics and varying surface inclinations. We evaluated the performance of the guidance law in a realistic simulator and through field experiments. The results show that the compensated trajectory shaping guidance with log polynomial function based velocity controller performs well. We have further evaluated the use of log polynomial function based velocity controller for rendezvous application where the vehicle needs to dock into a stationary target. The results show that the vehicle docks to

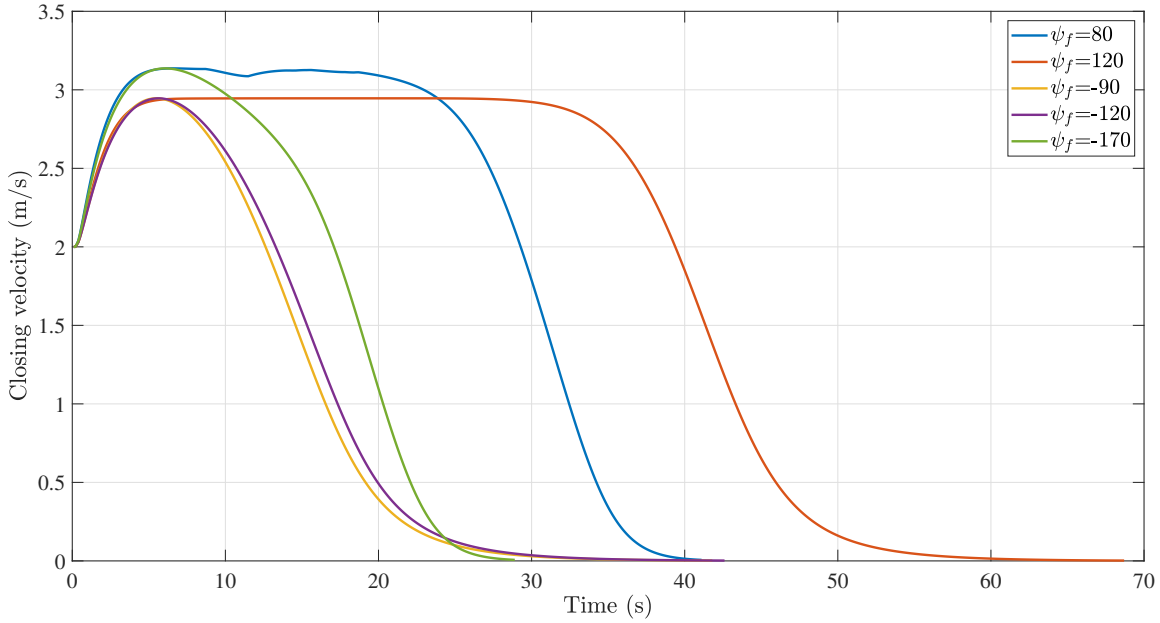


Figure 4.21: Closing velocity profile of the UAV for different impact LOS angles (ψ_f), stationary target

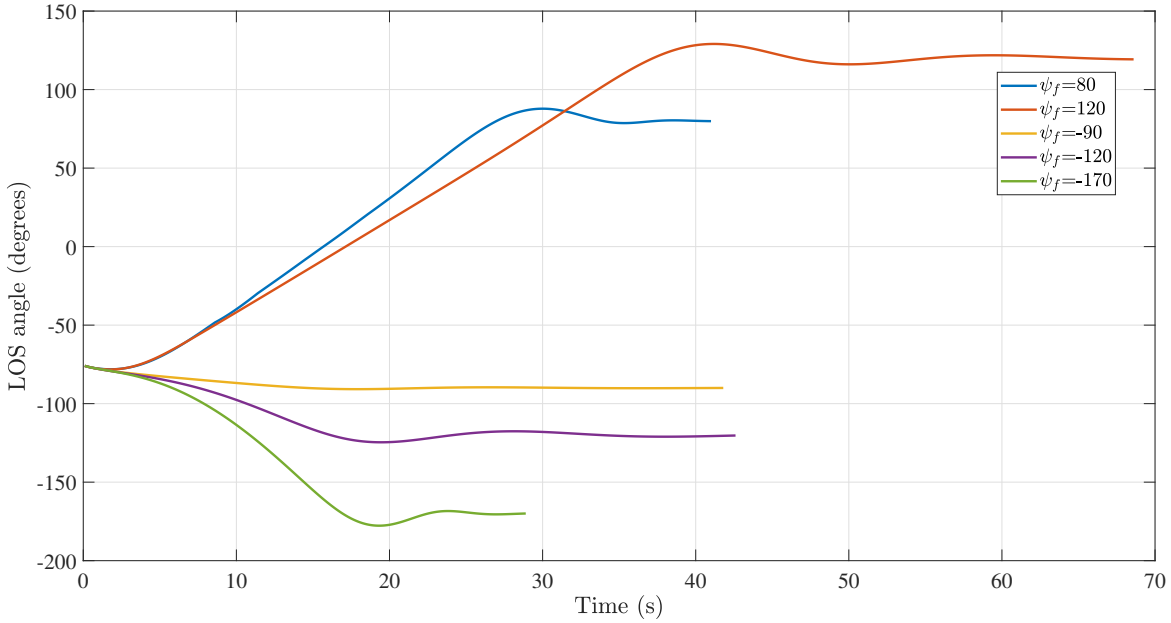


Figure 4.22: LOS angle profile of UAV for different ψ_f, stationary target

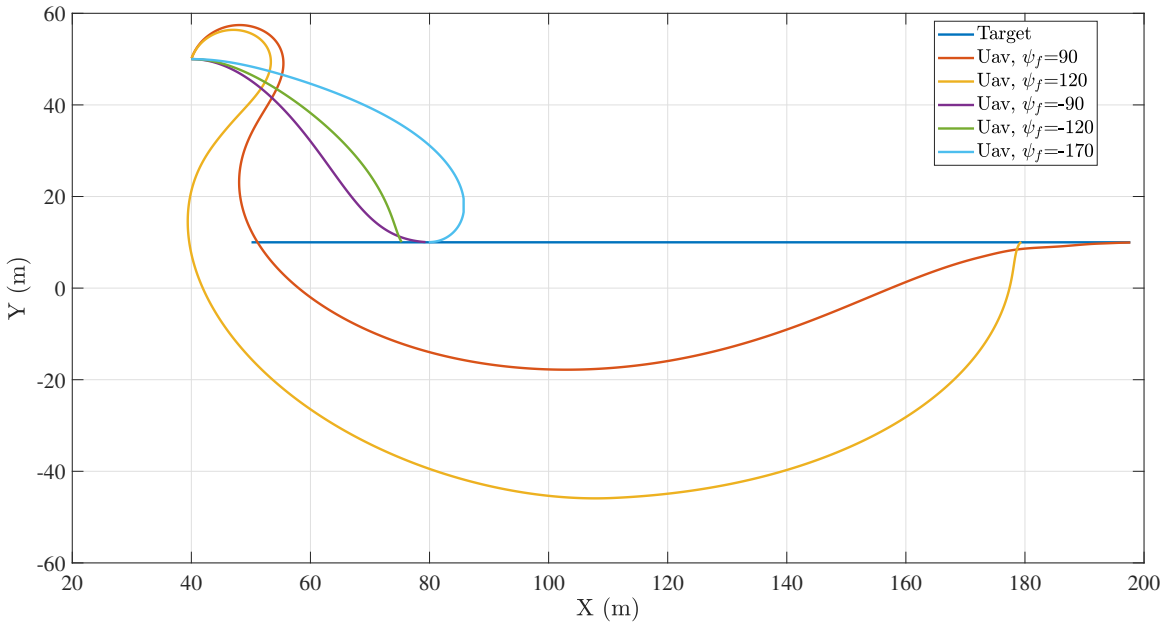


Figure 4.23: Trajectory followed by UAV to intercept the target for different ψ_f

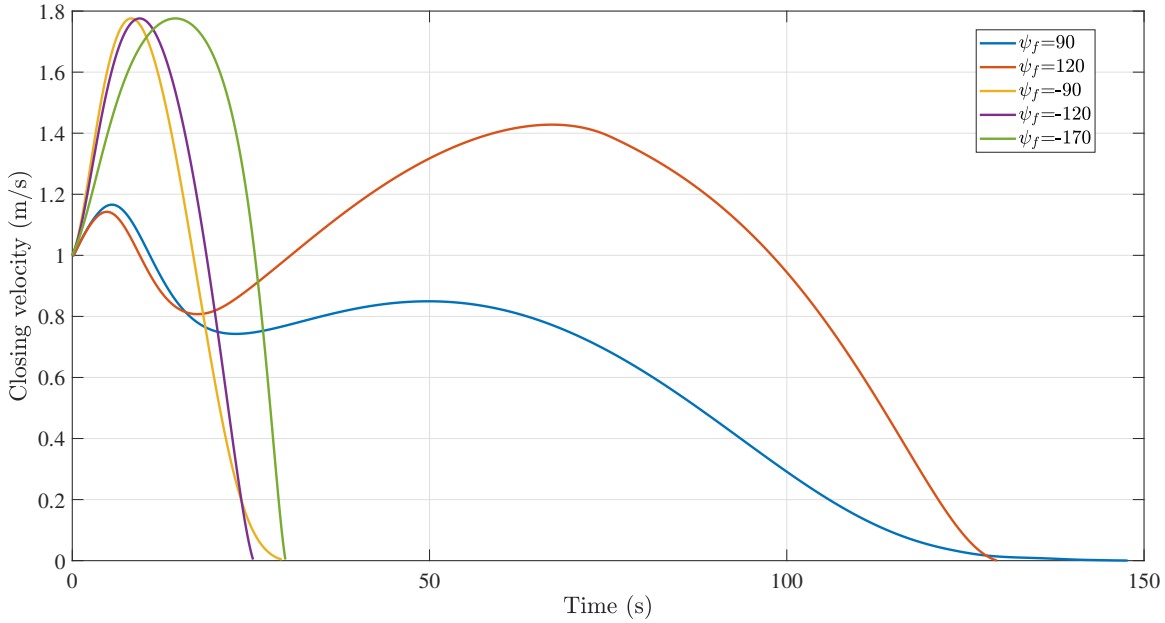


Figure 4.24: Closing velocity profile of the UAV for different impact LOS angles (ψ_f)

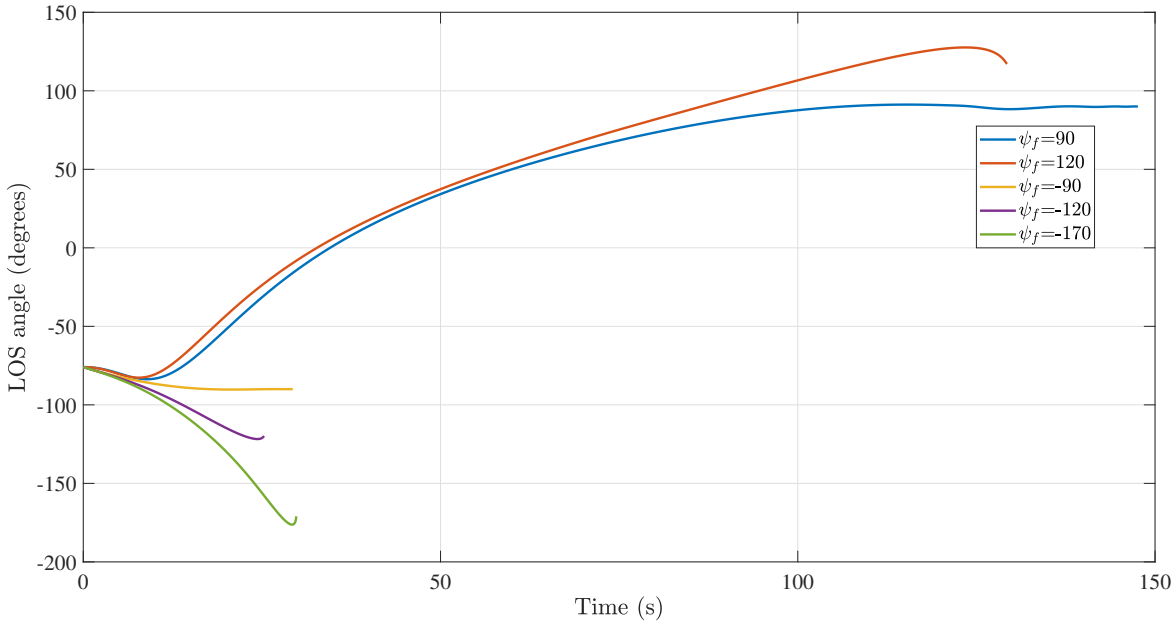


Figure 4.25: LOS angle profile of UAV for different ψ_f

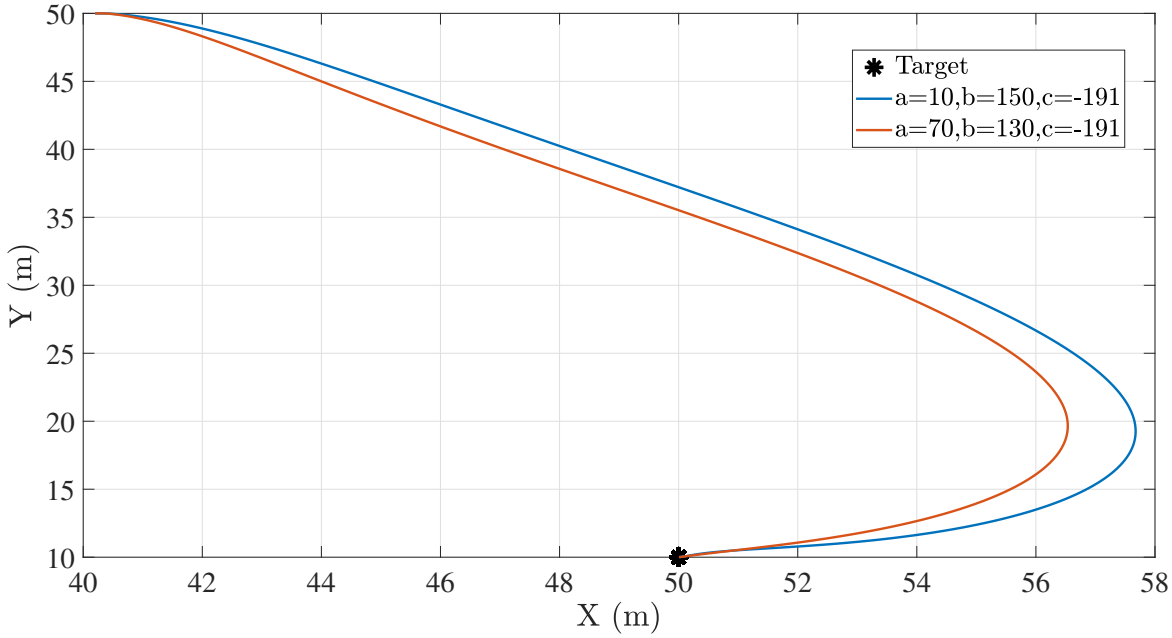


Figure 4.26: Trajectory of the UAV for a stationary target and two sets of (a, b, c) parameters, $\psi_f = -150$

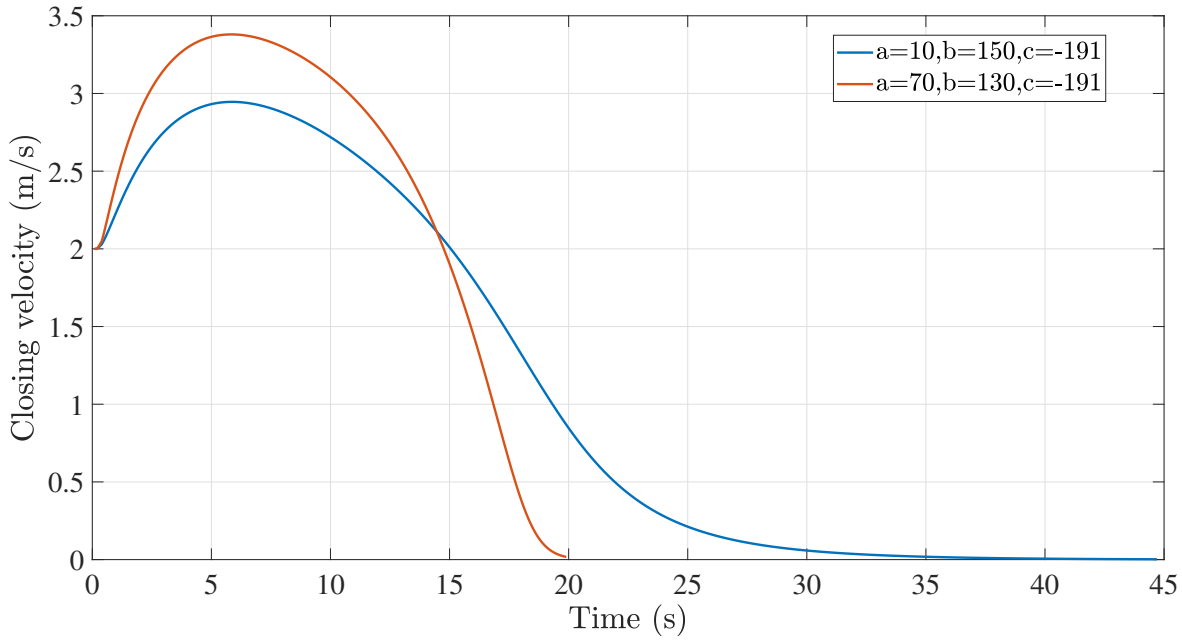


Figure 4.27: Closing velocity profile of the UAV for a stationary target and two sets of (a, b, c) parameters, $\psi_f = -150$

target satisfying the docking requirements.

Chapter 5

Conclusions

In this work, we focused on developing vision assisted autonomous landing strategy for a quadrotor, to address several pertinent issues relevant to autonomous landing, and to improve upon the state of the art. The main contributions of this thesis are summarized below:

1. In the second chapter, we developed a vision assisted pure pursuit based guidance strategy for autonomous landing of a quadrotor. We developed a novel log polynomial speed profile integrated with pure pursuit guidance for accurately landing the vehicle on the target, while meeting the required closing velocity characteristics. The guidance law is applicable to both stationary and moving targets. We performed simulations and carried out field experiments to demonstrate the efficacy of the guidance law.

2. In the third chapter, we provided a detailed analysis of the proposed log polynomial velocity profile. The analysis was carried out considering a vertical landing scenario to evaluate the impact of log polynomial function parameters on guidance characteristics. Closed-form expressions are derived for maximum velocity, and engagement time constraints as function of design parameters. The analysis provides a visual representation of the choice of velocity profile (strictly decelerating, initially accelerating or hovering), maximum velocity constraint, and engagement time in two-dimensional design parameter space. The analysis results of a vertical landing scenario help us in devising the landing behaviors in three dimensions (x, y, z) and can give insights to the overall landing process to suit the needs of the application.

3. Performance of pure pursuit guidance law for landing applications is limited to flat regions and fixed direction towards the target. In the fourth chapter, we addressed this limitation by proposing a compensated trajectory shaping guidance integrated with log polynomial controller, for landing on non flat regions and adding a constraint on the terminal LOS angle. Further, we investigate the effects of environment distur-

bances and varying target characteristics like, environmental uncertainties, uneven landing surface and constraints imposed by target localization errors, uncertain motion of the target, to validate the robustness of our proposed landing strategy. We carried out qualitative analysis of the log polynomial guidance law in simulations under varying environmental conditions and movement of the target. The simulations were performed in, Microsoft Airsim with unreal engine. The robustness of the guidance law was further analyzed through, outdoor experiments. These studies showed that the proposed guidance law is capable of landing on stationary and moving targets, in reasonably adverse environmental conditions. We also explore the extension of the proposed guidance law to rendezvous applications and demonstrate its efficacy through 2D matlab simulations.

Some of the future research directions are highlighted below:

1. As the vehicle descends, there is a high probability of the target moving out of field of view. Although, one can use an estimator, at times, the target may not be visible within a given time threshold. Therefore, using an active gimbal control to search for the target would be ideal. One potential direction of future research is on designing a gimbal-based active landing methodology for landing on highly maneuvering targets.

2. The proposed log polynomial velocity profile has several advantages on designing the velocity given a landing time. Therefore, this profiler can be used for emergency landing and also for planetary landing. Integrating the profiler with the dynamics and kinematics of other type of aircrafts and spacecrafts is another direction of future research.

   We derived closed form expressions for maximum velocity and engagement time constraints for the proposed profile. Evaluating the profiler for minimum engagement time objective for vertical landing and comparing the proposed method with an optimal control based methodology is also an interesting future research direction.

3. Coupling of estimation and control loops and evaluating the performance of proposed controller with respect to observability analysis is another future direction.

4. Although, this thesis focused on landing on an inclined and flat surfaces, one challenge is to perform a ship based landing where the target is having 3D motion along with wind disturbances that are function of the sea-states.

5. Another important area of research for landing is to decide when to abort landing. Although this is an important decision-making problem, there is not adequate work on this problem. One may have to

use machine learning techniques to learn when to make the decision. Some potential machine learning techniques are learning by demonstration and reinforcement learning.

6. In the current approach, the localization and stabilization of the aircraft is based on GPS. One potential extension is to land in GPS-denied environments using vision. This may require estimation of the target as well the POSE of the aircraft relative geometry space rather than earth-fixed inertial frame.

7. In the current approach, landing target considered is neutral. Since the log polynomial gives us flexibility of different velocity profiles, extending the controller for cooperative landing problem with controlled motion of the target is also an interesting research direction.

8. A natural extension is to apply the proposed guidance-laws and the log polynomial velocity controller for fixed-wing aircrafts. Since the fixed-wing aircraft requires minimum velocity to ensure it does not stall, the log polynomial velocity controller may need to be modified.

Bibliography

[1] L. Geng, Y. Zhang, J. Wang, J. Fuh, and S. Teo, "Mission planning of autonomous uavs for urban surveillance with evolutionary algorithms," in *IEEE International Conference on Control and Automation*, Hangzhou, China, 2013, pp. 828–833.

[2] S. Waharte and N. Trigoni, "Supporting search and rescue operations with uavs," in *International Conference on Emerging Security Technologies*, Canterbury, UK, 2010, pp. 142–147.

[3] S. Herwitz, L. Johnson, S. Dunagan, R. Higgins, D. Sullivan, J. Zheng, B. Lobitz, J. Leung, B. Gallmeyer, M. Aoyagi *et al.*, "Imaging from an unmanned aerial vehicle: agricultural surveillance and decision support," *Computers and electronics in agriculture*, vol. 44, no. 1, pp. 49–61, 2004.

[4] G. Grenzdörffer, A. Engel, and B. Teichert, "The photogrammetric potential of low-cost uavs in forestry and agriculture," *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 31, no. B3, pp. 1207–1214, 2008.

[5] A. R. Girard, A. S. Howell, and J. K. Hedrick, "Border patrol and surveillance missions using multiple unmanned air vehicles," in *IEEE Conference on Decision and Control*, vol. 1, Atlantis, Bahamas, 2004, pp. 620–625.

[6] V. Ramanathan, M. V. Ramana, G. Roberts, D. Kim, C. Corrigan, C. Chung, and D. Winker, "Warming trends in asia amplified by brown cloud solar absorption," *Nature*, vol. 448, no. 7153, pp. 575–578, 2007.

[7] M. Nagai, T. Chen, R. Shibasaki, H. Kumagai, and A. Ahmed, "Uav-borne 3-d mapping system by multisensor integration," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 47, no. 3, pp. 701–708, 2009.

[8] C. S. Sharp, O. Shakernia, and S. S. Sastry, "A vision system for landing an unmanned aerial vehicle," in *IEEE International Conference on Robotics and Automation*, vol. 2, Seoul, Korea, 2001, pp. 1720–1727.

[9] M. Garg, A. Kumar, and P. Sujit, "Terrain-based landing site selection and path planning for fixed-wing uavs," in *International Conference on Unmanned Aircraft Systems*. IEEE, 2015, pp. 246–251.

[10] J. Park, Y. Kim, and S. Kim, "Landing site searching and selection algorithm development using vision system and its application to quadrotor," *IEEE Transactions on Control Systems Technology*, vol. PP, no. 99, pp. 1–1, 2014.

[11] R. Brockers, P. Bouffard, J. Ma, L. Matthies, and C. Tomlin, "Autonomous landing and ingress of micro-air-vehicles in urban environments based on monocular vision," in *Micro-and Nanotechnology Sensors, Systems, and Applications III*, vol. 8031, 2011, p. 803111.

[12] S. Scherer, L. Chamberlain, and S. Singh, "Online assessment of landing sites," in *AIAA Infotech@ Aerospace*, no. AIAA 2010-3358, Atlanta, Georgia, 2010.

[13] A. Kushleyev, B. MacAllister, and M. Likhachev, "Planning for landing site selection in the aerial supply delivery," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, San Francisco, California, USA, 2011, pp. 1146–1153.

[14] M. Likhachev and A. Stentz, "Ppcp: Efficient probabilistic planning with clear preferences in partially-known environments," in *National Conference on Artificial Intelligence*, vol. 21, no. 1, Boston, Massachusetts, 2006, p. 860.

[15] S. Griffiths, J. Saunders, A. Curtis, B. Barber, T. McLain, and R. Beard, "Maximizing miniature aerial vehicles," *IEEE Robotics & Automation Magazine*, vol. 13, no. 3, pp. 34–43, 2006.

[16] P. Yan, M. Ding, C. Zhou, and C. Zheng, "A path replanning algorithm based on roadmap-diagram," in *World Congress on Intelligent Control and Automation*, vol. 3, Hangzhou, China, 2004, pp. 2433–2437.

[17] R. Brockers, M. Hummenberger, S. Weiss, and L. Matthies, "Towards autonomous navigation of miniature uav," in *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, Columbus, Ohio, 2014, pp. 645–651.

[18] S. Kurnaz and O. Çetin, "Autonomous navigation and landing tasks for fixed wing small unmanned aerial vehicles," *Acta Polytechnica Hungarica*, vol. 7, no. 1, pp. 87–102, 2010.

[19] B. Erginer and E. Altug, "Modeling and pd control of a quadrotor vtol vehicle," in *IEEE Intelligent Vehicles Symposium*, Istanbul, Turkey, 2007, pp. 894–899.

[20] V. Lippiello, F. Ruggiero, and D. Serra, "Emergency landing for a quadrotor in case of a propeller failure: A pid based approach," in *IEEE International Symposium on Safety, Security, and Rescue Robotics*, 2014, pp. 1–7.

[21] C. Ha and J.-k. Kim, "Automatic landing under wind shear turbulence in adaptive gain scheduled pid control optimized in genetic algorithm," in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, 2005, p. 6346.

[22] H. Khalil, *Nonlinear Systems*. Prentice Hall PTR, 2002.

[23] B. Prasad B and S. Pradeep, "Automatic landing system design using feedback linearization method," in *AIAA Infotech@ Aerospace 2007 Conference and Exhibit*, no. AIAA 2007-2733, California, 2007.

[24] H. Voos and B. Nourghassemi, "Nonlinear control of stabilized flight and landing for quadrotor uavs," in *Workshop on Advanced Control and Diagnosis, Zielona Góra, Poland*, 2009.

[25] B. T. Burchett, "Feedback linearization guidance for approach and landing of reusable launch vehicles," in *American Control Conference*. IEEE, Portland, OR, USA, 2005, pp. 2093–2097.

[26] T. Lee and Y. Kim, "Nonlinear adaptive flight control using backstepping and neural networks controller," *Journal of Guidance, Control, and Dynamics*, vol. 24, no. 4, pp. 675–682, 2001.

[27] D. Venkateswara Rao and T. H. Go, "Automatic landing system design using sliding mode control," *Aerospace Science and Technology*, vol. 32, no. 1, pp. 180–187, 2014.

[28] D. Lee, H. J. Kim, and S. Sastry, "Feedback linearization vs. adaptive sliding mode control for a quadrotor helicopter," *International Journal of control, Automation and systems*, vol. 7, no. 3, pp. 419–428, 2009.

[29] Y. Huang, T. Kuo, and H. Way, "Robust vertical takeoff and landing aircraft control via integral sliding mode," *IEE Proceedings-Control Theory and Applications*, vol. 150, no. 4, pp. 383–388, 2003.

[30] S. Bouabdallah and R. Siegwart, "Backstepping and sliding-mode techniques applied to an indoor micro quadrotor," in *IEEE International Conference on Robotics and Automation*, Barcelona, Spain, 2005, pp. 2247–2252.

[31] B. Ahmed and H. R. Pota, "Backstepping-based landing control of a ruav using tether incorporating flapping correction dynamics," in *American Control Conference*. IEEE, Seattle, WA, 2008, pp. 2728–2733.

[32] B. Ahmed, H. R. Pota, and M. Garratt, "Flight control of a rotary wing uav using backstepping," *International Journal of Robust and Nonlinear Control*, vol. 20, no. 6, pp. 639–658, 2010.

[33] S. Yoon, Y. Kim, and S. Park, "Constrained adaptive backstepping controller design for aircraft landing in wind disturbance and actuator stuck," *International Journal of Aeronautical and Space Sciences*, vol. 13, no. 1, pp. 74–89, 2012.

[34] M. Livchitz, A. Abershitz, U. Soudak, and A. Kandel, "Development of an automated fuzzy-logic-based expert system for unmanned landing," *Fuzzy Sets and Systems*, vol. 93, no. 2, pp. 145–159, 1998.

[35] K. Nho and R. K. Agarwal, "Automatic landing system design using fuzzy logic," *Journal of Guidance, Control, and Dynamics*, vol. 23, no. 2, pp. 298–304, 2000.

[36] M. A. Olivares-Méndez, I. F. Mondragón, P. Campoy, and C. Martinez, "Fuzzy controller for uav-landing task using 3d-position visual estimation," in *IEEE International Conference on Fuzzy Systems*, Barcelona, Spain, 2010, pp. 1–8.

[37] S. Malaek, N. Sadati, H. Izadi, and M. Pakmehr, "Intelligent autolanding controller design using neural networks and fuzzy logic," in *Asian Control Conference*, vol. 1, Melbourne, Victoria, Australia, 2004, pp. 365–373.

[38] J.-S. Jang, "Anfis: adaptive-network-based fuzzy inference system," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 23, no. 3, pp. 665–685, 1993.

[39] R. Mori, S. Suzuki, Y. Sakamoto, and H. Takahara, "Analysis of visual cues during landing phase by using neural network modeling," *Journal of aircraft*, vol. 44, no. 6, pp. 2006–2011, 2007.

[40] T. J. Koo and S. Sastry, "Hybrid control of unmanned aerial vehicles for autonomous landing," in *AIAA Unmanned Unlimited*, no. AIAA 2003-6541, San Diego, California, 2003.

[41] E. Frazzoli, M. A. Dahleh, and E. Feron, "A hybrid control architecture for aggressive maneuvering of autonomous helicopters," in *IEEE Conference on Decision and Control*, vol. 3, Arizona, USA, 1999, pp. 2471–2476.

[42] S.-P. Shue and R. K. Agarwal, "Design of automatic landing systems using mixed h_2/h_∞ control," *Journal of Guidance, Control, and Dynamics*, vol. 22, no. 1, pp. 103–114, 1999.

[43] F. Liao, J. L. Wang, E. K. Poh, and D. Li, "Fault-tolerant robust automatic landing control design," *Journal of Guidance, Control, and Dynamics*, vol. 28, no. 5, pp. 854–871, 2005.

[44] R. Wang, Z. Zhou, and Y. Shen, "Flying-wing uav landing control and simulation based on mixed h_2/h_∞," in *International Conference on Mechatronics and Automation*, Heilongjiang, China, 2007, pp. 1523–1528.

[45] R. Lungu and M. Lungu, "Application of h2/hâĹđ and dynamic inversion techniques to aircraft landing control," *Aerospace Science and Technology*, vol. 46, pp. 146–158, 2015.

[46] K. E. Wenzel, P. Rosset, and A. Zell, "Low-cost visual tracking of a landing place and hovering flight control with a microcontroller," in *International Symposium on UAVs, Reno, Nevada, USA*. Springer, June 2009, pp. 297–311.

[47] M. F. Sani and G. Karimian, "Automatic navigation and landing of an indoor ar. drone quadrotor using aruco marker and inertial sensors," in *International Conference on Computer and Drone Applications*. IEEE, 2017, pp. 102–107.

[48] W. E. Green, P. Y. Oh, K. Sevcik, and G. Barrows, "Autonomous landing for indoor flying robots using optic flow," in *ASME International Mechanical Engineering Congress and Exposition*, 2003, pp. 1347–1352.

[49] T. Merz, S. Duranti, and G. Conte, "Autonomous landing of an unmanned helicopter based on vision and inertial sensing," in *Experimental Robotics IX*. Springer, 2006, pp. 343–352.

[50] T. Daquan and Z. Hongyue, "Vision based navigation algorithm for autonomic landing of uav without heading & attitude sensors," in *IEEE International Conference on Signal-Image Technologies and Internet-Based System*, Shanghai, China, 2007, pp. 972–978.

[51] A. Miller, M. Shah, and D. Harper, "Landing a uav on a runway using image registration," in *IEEE International Conference on Robotics and Automation*, Pasadena, California, 2008, pp. 182–187.

[52] X. Pan, D.-q. Ma, L.-l. Jin, and Z.-s. Jiang, "Vision-based approach angle and height estimation for uav landing," in *Congress on Image and Signal Processing*, vol. 3, Hainan, China, 2008, pp. 801–805.

[53] I. F. Mondragon, P. Campoy, J. F. Correa, and L. Mejias, "Visual model feature tracking for uav control," in *IEEE International Symposium on Intelligent Signal Processing*, Alcala de Henares, Spain, 2007, pp. 1–6.

[54] "Depth estimation of markers for uav automatic landing control using stereo vision with a single camera," in *International Conference on Information and Communication Technology for Embedded System*, 2014.

[55] S. Saripalli, J. F. Montgomery, and G. S. Sukhatme, "Visually guided landing of an unmanned aerial vehicle," *IEEE Transactions on Robotics and Automation,*, vol. 19, no. 3, pp. 371–380, 2003.

[56] S. Saripalli and G. S. Sukhatme, "Landing on a moving target using an autonomous helicopter," in *Field and Service Robotics*. Springer, 2006, pp. 277–286.

[57] S. Lange, N. Sunderhauf, and P. Protzel, "A vision based onboard approach for landing and position control of an autonomous multirotor uav in gps-denied environments," in *International Conference on Advanced Robotics*, Munich, Germany, 2009, pp. 1–6.

[58] O. Shakernia, Y. Ma, T. J. Koo, and S. Sastry, "Landing an unmanned air vehicle: Vision based motion estimation and nonlinear control," *Asian journal of control*, vol. 1, no. 3, pp. 128–145, 1999.

[59] G. Xu, Y. Zhang, S. Ji, Y. Cheng, and Y. Tian, "Research on computer vision-based for uav autonomous landing on a ship," *Pattern Recognition Letters*, vol. 30, no. 6, pp. 600–605, 2009.

[60] D. B. Barber, S. R. Griffiths, T. W. McLain, and R. W. Beard, "Autonomous landing of miniature aerial vehicles," *Journal of Aerospace Computing, Information, and Communication*, vol. 4, no. 5, pp. 770–784, 2007.

[61] B. Herissé, T. Hamel, R. Mahony, and F.-X. Russotto, "Landing a vtol unmanned aerial vehicle on a moving platform using optical flow," *IEEE Transactions on Robotics*, vol. 28, no. 1, pp. 77–89, 2012.

[62] D. Iwakura, W. Wang, K. Nonami, and M. Haley, "Movable range-finding sensor system and precise automated landing of quad-rotor mav," *Journal of System Design and Dynamics*, vol. 5, no. 1, pp. 17–29, 2011.

[63] S. Arora, S. Jain, S. Scherer, S. Nuske, L. Chamberlain, and S. Singh, "Infrastructure-free shipdeck tracking for autonomous landing," in *IEEE International Conference on Robotics and Automation*, Karlsruhe, Germany, 2013, pp. 323–330.

[64] M. Kim and Y. Kim, "Multiple uavs nonlinear guidance laws for stationary target observation with waypoint incidence angle constraint," *International Journal of Aerospace Science*, vol. 14, no. 1, pp. 67–74, 2013.

[65] B.-M. Min, M.-J. Tahk, D. H. Shim, and H. Bang, "Guidance law for vision-based automatic landing of uav," *International Journal of Aeronautical and Space Sciences*, vol. 8, no. 1, pp. 46–53, 2007.

[66] H. J. Kim, M. Kim, H. Lim, C. Park, S. Yoon, D. Lee, H. Choi, G. Oh, J. Park, and Y. Kim, "Fully autonomous vision-based net-recovery landing system for a fixed-wing uav," *IEEE/ASME Transactions on Mechatronics*, vol. 18, no. 4, pp. 1320–1333, 2013.

[67] S. Yoon, H. J. Kim, and Y. Kim, "Spiral landing trajectory and pursuit guidance law design for vision-based net-recovery uav," in *AIAA Guidance Navigation and Control Conference*, Chicagi, Illinios, 2009, pp. AIAA 2009–5682.

[68] S. Khantsis and A. Bourmistrova, "Uav controller design using evolutionary algorithms," in *AI 2005: Advances in Artificial Intelligence*. Springer, Sydney, Australia, 2005, pp. 1025–1030.

[69] S. Huh and D. H. Shim, "A vision-based landing system for small unmanned aerial vehicles using an airbag," *Control Engineering Practice*, vol. 18, no. 7, pp. 812–823, 2010.

[70] N. Gageik, T. Müller, and S. Montenegro, "Obstacle detection and collision avoidance using ultrasonic distance sensors for an autonomous quadrocopter," *University Of Wurzburg, Aerospace Information Technology (Germany) Wurzburg September*, 2012.

[71] S. Scherer, L. Chamberlain, and S. Singh, "First results in autonomous landing and obstacle avoidance by a full-scale helicopter," in *IEEE International Conference on Robotics and Automation*, Minnesota, USA, 2012, pp. 951–956.

[72] M. C. Martin and H. P. Moravec, "Robot evidence grids." DTIC Document, Tech. Rep., 1996.

[73] S. Scherer, D. Ferguson, and S. Singh, "Efficient c-space and cost function updates in 3d for unmanned aerial vehicles," in *IEEE International Conference on Robotics and Automation*, Kobe, Japan, 2009, pp. 2049–2054.

[74] S. Krause, "Multi-purpose environment awareness approach for single line laser scanner in a small rotorcraft ua," *Journal of Intelligent & Robotic Systems*, vol. 65, no. 1-4, pp. 587–601, 2012.

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

[76] A. Cho, J. Kim, S. Lee, S. Choi, B. Lee, B. Kim, N. Park, D. Kim, and C. Kee, "Fully automatic taxiing, takeoff and landing of a uav using a single-antenna gps receiver only," in *International Conference on Control, Automation and Systems*, Oct 2007, pp. 821–825.

[77] F.-B. Hsiao, S.-H. Huang, and M.-T. Lee, "The study of real-timed gps navigation accuracy during approach and landing of an ultralight vehicle," in *International Conference on Recent Advances in Space Technologies*, Nov 2003, pp. 375–384.

[78] A. Cesetti, E. Frontoni, A. Mancini, P. Zingaretti, and S. Longhi, *A Vision-Based Guidance System for UAV Navigation and Safe Landing using Natural Landmarks*. Dordrecht: Springer Netherlands, 2010, pp. 233–257.

[79] K. P. Valavanis and G. J. Vachtsevanos, *Handbook of Unmanned Aerial Vehicles*. Springer Publishing Company, Incorporated, 2014.

[80] M. Abramowitz and I. A. Stegun, *Handbook of mathematical functions: with formulas, graphs, and mathematical tables*. Courier Corporation, 1964, vol. 55.

[81] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "Ros: an open-source robot operating system," in *ICRA workshop on open source software*, vol. 3, no. 3.2, 2009, p. 5.

[82] A. Koenig, "gazebo-ros wiki," 2013.

[83] J. Meyer, A. Sendobry, S. Kohlbrecher, U. Klingauf, and O. von Stryk, *Comprehensive Simulation of Quadrotor UAVs Using ROS and Gazebo*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 400–411.

[84] https://3dr.com/kb/iris/.

[85] http://www.hardkernel.com/main/products/prdt_info.php?g_code=g138745696275.

[86] E. Games, "Robot operating system," *Online: www.ros.org*, 2008.

[87] *The OpenCV Reference Manual*, 2nd ed., Itseez, April 2014.

[88] W. Kong, D. Zhang, X. Wang, Z. Xian, and J. Zhang, "Autonomous landing of an uav with a ground-based actuated infrared stereo vision system," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Tokyo, Japan, 2013, pp. 2963–2970.

[89] H. Lee, S. Jung, and D. H. Shim, "Vision-based uav landing on the moving vehicle," in *International Conference on Unmanned Aircraft Systems*, Arlington, VA, USA, 2016, pp. 1–7.

[90] K. Ling, "Precision landing of a quadrotor uav on a moving target using low-cost sensors," in *Master's Thesis, Department of Mechanical and Mechatronics Engineering*, University of Waterloo, Ontario, Canada, 2014, url: http://hdl.handle.net/10012/8803.

[91] J. de Lafontaine, D. Neveu, and K. Lebel, "Autonomous planetary landing using a lidar sensor: The closed-loop system," in *International ESA Conference on Guidance, Navigation and Control Systems*, Loutraki, Greece, 2006, pp. 3.1–3.10.

[92] A. Borowczyk, D.-T. Nguyen, A. Phu-Van Nguyen, D. Q. Nguyen, D. Saussié, and J. Le Ny, "Autonomous landing of a quadcopter on a high-speed ground vehicle," *Journal of Guidance, Control, and Dynamics*, vol. 40, no. 9, pp. 2373–2380, 2017.

[93] M. T. Alkowatly, V. M. Becerra, and W. Holderbaum, "Bioinspired autonomous visual vertical control of a quadrotor unmanned aerial vehicle," *Journal of Guidance, Control, and Dynamics*, vol. 38, no. 2, pp. 249–262, 2014.

[94] P. Xie, O. Ma, and Z. Zhang, "A bio-inspired approach for uav landing and perching," in *AIAA Guidance, Navigation, and Control Conference*, Boston, MA, 2013, pp. AIAA 2013–5108.

[95] F. Kendoul, "Four-dimensional guidance and control of movement using time-to-contact: Application to automated docking and landing of unmanned rotorcraft systems," *International Journal of Robotics Research*, vol. 33, no. 2, pp. 237–267, 2014.

[96] Z. Yang, Z. Fang, and P. Li, "Decentralized 4d trajectory generation for uavs based on improved intrinsic tau guidance strategy," *International Journal of Advanced Robotic Systems*, vol. 13, no. 3, pp. 1–13, 2016.

[97] (2017) Dji official. http://www.dji.com/phantom.

[98] "Parrot official," https://www.parrot.com, 2017.

[99] (2017) Lightware official. http://lightware.co.za/shop2017/drone-altimeters/51-lw20-100-m.html.

[100] T. Zhao and H. Jiang, "Landing system for ar.drone 2.0 using onboard camera and ros," in *Chinese Guidance, Navigation and Control Conference*, Aug Nanjing, China, 2016, pp. 1098–1102.

[101] N. Shneydor, *Missile Guidance and Pursuit: Kinematics, Dynamics and Control*, ser. Horwood engineering science series. Horwood Pub., 1998. [Online]. Available: https://books.google.co.in/books?id=8X7CQgAACAAJ

[102] S. Shah, D. Dey, C. Lovett, and A. Kapoor, "Airsim: High-fidelity visual and physical simulation for autonomous vehicles," *CoRR*, vol. abs/1705.05065, 2017.

[103] E. Games, "Unreal engine," *Online: https://www. unrealengine. com*, 2007.

[104] A. Gautam, P. Sujit, and S. Saripalli, "Autonomous quadrotor landing using vision and pursuit guidance," *IFAC World Congress*, vol. 50, no. 1, pp. 10 501–10 506, Toulouse, France, 2017.

[105] A. Gautam, A. Ratnoo, and P. Sujit, "Log polynomial velocity profile for vertical landing," *Journal of Guidance, Control, and Dynamics*, vol. 41, no. 7, pp. 1617–1623, 2018.

[106] (2017) Wind data. https://darksky.net.