

Tesis Doctoral
Ingeniería Automática, Electrónica y de
Telecomunicación

Autonomous Landing of Fixed-Wing **Aircraft on Mobile Platforms**



Autor: Tin Muskardin
Director: Aníbal Ollero Baturone

Ingeniería de Sistemas y Automática
Escuela Técnica Superior de Ingeniería
Universidad de Sevilla

Sevilla, 2019



Tesis Doctoral
Ingeniería Automática, Electrónica y de Telecomunicación

Autonomous Landing of Fixed-Wing Aircraft on Mobile
Platforms

Autor:

Tin Muskardin

Director:

Aníbal Ollero Baturone

Profesor Catedrático

Ingeniería de Sistemas y Automática
Escuela Técnica Superior de Ingeniería
Universidad de Sevilla

2019

Tesis Doctoral: Autonomous Landing of Fixed-Wing Aircraft on Mobile Platforms

Autor: Tin Muskardin
Director: Aníbal Ollero Baturone

El tribunal nombrado para juzgar la Tesis arriba indicada, compuesto por los siguientes doctores:

Presidente:

Vocales:

Secretario:

acuerdan otorgarle la calificación de:

El Secretario del Tribunal

Fecha:

For my family

Acknowledgements

First and foremost I would like to thank my thesis supervisor Prof. Aníbal Ollero for his support throughout the last years. I was always inspired by his commitment and passion for science, and feel very lucky that he has accepted me as his student. I would also like to thank Prof. Alin Albu-Schäffer for hosting me at the DLR Robotics and Mechatronics Institute for so many years. It was always a great pleasure to work at this place and a source of fulfillment. I thank Prof. Gerd Hirzinger for providing the context of this work and for accepting to be on the thesis committee. I would like to express my fullest gratitude to Dr. Christian Ott for his continued trust and support. Finally, I don't have words to describe my gratitude to Dr. Konstantin Kondak, who has always been a role model for me. A true modern scientist. I have learned so much from you.

I would like to thank my colleagues and especially all of my students. You have had an important share in this work. I have enjoyed working with all of you and have learned very much from you.

I want to thank my parents for their love and support throughout all my life. They taught me to be curious and to love the world and its people. Without them I would not be here.

I want to thank my wife for her love. For making me a better person, for giving me confidence, for being my complice. I want to thank my children for teaching me what is important and what is not.

Finally I want to thank my friends for being in my life. Nothing is real if you cannot share it.

*Tin Muskardin
Munich, 2019*

Resumen

En esta tesis se propone un nuevo sistema que permite la operación de aeronaves autónomas sin tren de aterrizaje. El trabajo está motivado por el interés industrial en aeronaves con la capacidad de volar a gran altitud, con más capacidad de carga útil y capaces de aterrizar con viento cruzado.

El enfoque seguido en este trabajo consiste en eliminar el sistema de aterrizaje de una aeronave de ala fija empleando una plataforma móvil de aterrizaje en tierra. La aeronave y la plataforma deben sincronizar su movimiento antes del aterrizaje, lo que se logra mediante la estimación del estado relativo entre ambas y el control cooperativo del movimiento.

El objetivo principal de esta Tesis es el desarrollo de una solución práctica para el aterrizaje autónomo de una aeronave de ala fija en una plataforma móvil. En la tesis se combinan nuevos métodos con experimentos prácticos para los cuales se ha desarrollado un sistema de pruebas específico.

Se desarrollan dos variantes diferentes del sistema de aterrizaje. El primero presta atención especial a la seguridad, es robusto ante retrasos en la comunicación entre vehículos y cumple procedimientos habituales de aterrizaje, al tiempo que reduce la complejidad del sistema. En el segundo se utilizan trayectorias optimizadas del vehículo y sincronización bilateral de posición para maximizar el rendimiento del aterrizaje en términos de requerimientos de longitud necesaria de pista, pero la estabilidad es dependiente del retraso de tiempo, con lo cual es necesario desarrollar un controlador estabilizador ampliado, basado en pasividad, que permite resolver este problema.

Ambas estrategias imponen requisitos funcionales a los controladores de cada uno de los vehículos, lo que implica la capacidad de controlar el movimiento longitudinal sin afectar el control lateral o vertical, y viceversa. El control de vuelo basado en energía se utiliza para proporcionar dicha funcionalidad a la aeronave.

Los sistemas de aterrizaje desarrollados se han analizado en simulación estableciéndose los límites de rendimiento mediante múltiples repeticiones aleatorias. Se llegó a la conclusión de que el controlador basado en seguridad proporciona un rendimiento de aterrizaje satisfactorio al tiempo que suministra una mayor seguridad operativa y un menor esfuerzo de implementación y certificación. El controlador basado en el rendimiento es prometedor para aplicaciones con una longitud de pista limitada. Se descubrió que los

beneficios del controlador basado en el rendimiento son menos pronunciados para una dinámica de vehículos terrestres más lenta.

Teniendo en cuenta la dinámica lenta de la configuración del demostrador, se eligió el enfoque basado en la seguridad para los primeros experimentos de aterrizaje. El sistema de aterrizaje se validó en diversas pruebas de aterrizaje exitosas, que, a juicio del autor, son las primeras en el mundo realizadas con aeronaves reales. En última instancia, el concepto propuesto ofrece importantes beneficios y constituye una estrategia prometedora para futuras soluciones de aterrizaje de aeronaves.

Abstract

In this thesis a new landing system is proposed, which allows for the operation of autonomous aircraft without landing gear. The work was motivated by the industrial need for more capable high altitude aircraft systems, which typically suffer from low payload capacity and high crosswind landing sensitivity. The approach followed in this work consists in removing the landing gear system from the aircraft and introducing a mobile ground-based landing platform. The vehicles must synchronize their motion prior to landing, which is achieved through relative state estimation and cooperative motion control. The development of a practical solution for the autonomous landing of an aircraft on a moving platform thus constitutes the main goal of this thesis. Therefore, theoretical investigations are combined with real experiments for which a special setup is developed and implemented.

Two different landing system variants are developed — the safety-based landing system is robust to inter-vehicle communication delays and adheres to established landing procedures, while reducing system complexity. The performance-based landing system uses optimized vehicle trajectories and bilateral position synchronization to maximize landing performance in terms of used runway, but suffers from time delay-dependent stability. An extended passivity-based stabilizing controller was implemented to cope with this issue. Both strategies impose functional requirements on the individual vehicle controllers, which imply independent controllability of the translational degrees of freedom. Energy-based flight control is utilized to provide such functionality for the aircraft.

The developed landing systems are analyzed in simulation and performance bounds are determined by means of repeated random sampling. The safety-based controller was found to provide satisfactory landing performance while providing higher operational safety, and lower implementation and certification effort. The performance-based controller is promising for applications with limited runway length. The performance benefits were found to be less pronounced for slower ground vehicle dynamics. Given the slow dynamics of the demonstrator setup, the safety-based approach was chosen for first landing experiments. The landing system was validated in a number of successful landing trials, which to the author's best knowledge was the first time such technology was demonstrated

on the given scale, worldwide. Ultimately, the proposed concept offers decisive benefits and constitutes a promising strategy for future aircraft landing solutions.

Contents

<i>Resumen</i>	V
<i>Abstract</i>	VII
<i>Acronyms</i>	XIII
1 Introduction	1
1.1 The Aircraft Landing Problem	3
1.1.1 Standard Landing Procedures	3
1.1.2 Crosswind Landing Procedures	4
1.2 The Cooperative Landing Maneuver	9
1.3 Applicability to Current Aircraft Systems	12
1.3.1 Commercial Aviation	12
1.3.2 High Altitude Platforms and HALE Aircraft	14
1.3.3 Heavier Unmanned Aircraft Systems	16
1.4 Review of Related Applications	16
1.4.1 Novel Aircraft Landing Concepts	17
1.4.2 Automatic Runway Landing	18
1.4.3 Landing of Micro Air Vehicles on Mobile Ground Vehicles	19
1.4.4 Other Related Techniques	20
1.5 Scope of Work	23
1.5.1 Main Contribution	23
1.5.2 Experimental Setup	24
1.5.3 Document Structure	26
1.5.4 Scientific Output and Public Response	27
2 Cooperative Autoland System	31
Chapter Overview	31
2.1 System Requirements	31
2.2 The Landing Procedure	33
2.3 Functional Decomposition	36

2.4	System Design	40
2.4.1	Aircraft Flight Control	40
2.4.2	Ground Vehicle Control	44
2.4.3	Relative State Estimation	45
2.4.4	Approach Control	45
2.4.5	Landing Control	48
2.4.6	Deceleration Control	58
2.4.7	High-Level Mission Control and Fail-Safe System	60
	Chapter Conclusions	60
3	Modeling and Control of Aircraft and Ground Vehicle	61
	Chapter Overview	61
3.1	Aircraft Flight Dynamics Model	61
3.1.1	Coordinate Frames and Notation	61
3.1.2	Rigid Body Equations of Motion	63
3.1.3	Aerodynamics	65
3.1.4	Propulsion	68
3.1.5	Mass and Inertia	69
3.1.6	High Fidelity Modeling via System Identification	70
3.2	Aircraft Flight Control	78
3.2.1	Classical vs Energy-Based Flight Control	78
3.2.2	Waypoint Tracking Control	81
3.2.3	Controller Tuning	81
3.2.4	Qualitative Performance Analysis	82
3.3	Ground Vehicle Dynamics Model	85
3.4	Ground Vehicle Control	90
3.5	Linear Vehicle Models	93
3.5.1	Linear Aircraft Model	93
3.5.2	Linear Ground Vehicle Model	95
	Chapter Conclusions	96
4	Landing Control System	97
	Chapter Overview	97
4.1	Relative State Estimation	97
4.1.1	Networked State Estimation and Lag Compensation	98
4.1.2	Optical Marker Tracking	99
4.1.3	Sensor Data Fusion	101
4.2	Safety-Based Landing Control	103
4.2.1	Longitudinal Cooperative Control	104
4.2.2	Ground Speed Tracking Mode	107
4.2.3	Lateral Cooperative Control	110
4.2.4	Vertical Control and Flare Law	111
4.2.5	Retard and Ground Lock	111

4.2.6	High-Level Mission Control and Fail-Safe System	113
4.2.7	Deceleration	116
4.3	Performance-Based Landing Control	116
4.3.1	Optimized Approach	117
4.3.2	Bilateral Position Synchronization	134
4.3.3	Time Delay Robust Landing Control	136
4.4	Control System Design	150
4.4.1	Frequency Domain Design	152
4.4.2	Time Domain Design	152
4.4.3	Stability Assessment	155
4.4.4	Sensitivity Analysis	160
	Chapter Conclusions	169
5	Simulation Experiments	171
	Chapter Overview	171
5.1	Simulation Environment	171
5.2	Disturbance Rejection Performance Analysis	174
5.3	Disturbance Rejection Robustness Analysis	178
5.4	Landing Performance Analysis	183
5.4.1	Nominal Landing Performance	183
5.4.2	Failsafe Behavior	184
5.5	Landing Robustness Analysis	191
	Chapter Conclusions	195
6	Landing Experiments	197
	Chapter Overview	197
6.1	Preliminary Experiments	197
6.2	Experiments with Hardware-in-the-Loop Setup	202
6.3	Experiments with Final Demonstrator Setup	204
	Chapter Conclusions	209
7	Conclusions and Further Development	211
Appendix A	Simulation Parameters	215
A.1	Controller Parameters	215
A.2	Initial Conditions for Landing Performance Analysis	217
A.3	MPC Controller Settings	218
	<i>List of Figures</i>	221
	<i>List of Tables</i>	227
	<i>Bibliography</i>	229

Acronyms

UAV	Unmanned Aerial Vehicle
MAV	Micro Air Vehicle
UGV	Unmanned Ground Vehicle
GUI	Graphical User Interface
PID	Proportional–Integral–Derivative
DARPA	Defense Advanced Research Agency
ILS	Instrument Landing System
PN	Proportional Navigation
DDPG	Deep Deterministic Policy Gradients
MPC	Model Predictive Control
VL	Virtual Leader
AP	Artificial Potential
FOV	Field Of View
A3R	Automatic Air-to-Air Refueling
BRR	Boom-Receptacle Refueling
PDR	Probe-Drogue Refueling
SISO	Single-Input, Single-Output
6-DoF	Six-Degrees-of-Freedom
RMSD	Root-Mean-Square Deviations
NRMSD	Normalized Root-Mean-Square Deviations
CFD	Computational Fluid Dynamics
VLM	Vortex Lattice Method
EKF	Extended Kalman Filter
IEKS	Iterative Extended Kalman Filter
EKS	Backward Kalman Smoothing
KF	Kalman Filter
GOF	Goodness Of Fit
TIC	Theil's Inequality Coefficient
FAA	Federal Aviation Administration

R2	Coefficient of Determination
RMSE	Root Mean Squared Error
NRMSE	Normalized Root Mean Squared Error
TECS	Total Energy Control System
RTK	Real Time Kinematic
GPS	Global Positioning System
IMU	Inertial Measurement Unit
MIMO	Multiple-Input, Multiple-Output
TDPA	Time Domain Passivity Control Approach
MLDS	Mixed Logical Dynamical System
HIL	Hardware In the Loop
EDMO	Oberpfaffenhofen airport
EDMN	Mindelheim-Mattsies airport
P-P	Position-Position architecture
PO	Passivity Observer
PC	Passivity Controller
TDPN	Time Delay Power Network
P-Fc	Position-Force Computed
GM	Gain Margin
PM	Phase Margin
ITAE	Integral Time Absolute Error
MTOW	Maximum Take-Off Weight
GNSS	Global Navigation Satellite System

1 Introduction

Since the invention of the first fixed-wing aircraft, take-off and landing were commonly performed using some type of undercarriage. Until today most modern aircraft still possess a wheeled landing gear. The reasons for this perseverance do not arise from the landing gear setup being the best solution in terms of safety and efficiency, but are rather of historical nature. The landing gear and its supporting structure are heavy, complex and expensive components. Their mass typically ranges between 6% to 15% of a given aircraft's empty mass, depending on the aircraft type [86]. The landing gear causes high operational costs due to the high maintenance demand, while only being used during take-off, landing and taxiing. Once the aircraft reaches cruise flight, the landing gear is dead weight, reducing the overall performance and efficiency. Another disadvantage of classical landing strategies shows during a landing under crosswind conditions or in turbulence. Either the aircraft must be landed in sideslip configuration with the wings not being level and thus reduced ground clearance, or the fuselage needs to be re-aligned with the runway before or right after touchdown. Such corrective action is typically performed during the last seconds of flight, thus representing a safety critical maneuver with increased associated risk. In any case currently available landing techniques come with operational limitations, which might be very restrictive for certain aircraft types.

An interesting approach regarding the improvement of crosswind landing safety and performance consists in steerable main landing gear systems. Also known as active crosswind assistance systems [123] or crosswind crab systems [114], such method would allow for performing final approach, touchdown, and roll-out in a crabbed flight condition, where the aircraft nose is pointing into the relative wind direction while traveling along the runway with its wings level. This would lead to a decreased crosswind sensitivity, but would involve an increased system weight and complexity, and consequently reduced flight performance and higher maintenance demand.

Another option consists in dropping the landing gear system altogether, or at least parts of it, after take-off and to land on some type of landing skids or reinforced fuselage parts. There are several, mostly historical examples of aircraft where such concept was applied, it did however not turn out to be a reliable landing solution. Figure 1.1 illustrates the detachable landing gear design of the German *Arado Ar 234 Blitz* reconnaissance jet

bomber.



Figure 1.1 Arado Ar 234 V1 taking off from a tricycle trolley, ©.

The main disadvantages consist in the inability to move the aircraft on the ground after touchdown, the lack of braking capability of the landing skids, and the increased risk of damage during the landing. In case of damage, the airfield may be blocked by debris which could lead to a complete operational outage of the airfield during the cleanup.

A further improvement consists in the removal of the landing gear from the aircraft and the introduction of a ground-based landing platform. Such landing system requires an autonomous synchronization of aircraft and landing platform prior to touchdown and thus involves a higher level of complexity. The aircraft takeoff weight could be significantly reduced since the landing gear would be *ground-based*, and crosswind landings could be simplified in the same way as explained for the steerable landing gear, where touchdown can be performed in crabbed flight condition. Given a safe and reliable technical solution, the system would offer decisive advantages for any type of aircraft, where economic, performance, or mobility aspects are of particular importance. Especially for long duration flights where the airborne time is large compared to the number of required landings, and for aircraft with highly weather dependent operational availability, such landing system would be beneficial. Its development constitutes the main goal of this work.

Throughout the rest of this chapter the aircraft landing problem will be analyzed in detail and established landing procedures will be described. The limitations of these procedures will be discussed, and a number of mitigation strategies presented. Next, a cooperative landing maneuver will be introduced and its applicability to current aircraft systems will be evaluated. Finally the experimental setup available for this work will be presented and the scope of the thesis defined.

1.1 The Aircraft Landing Problem

Before addressing the landing of an aircraft on top of a mobile landing platform, it is important to understand established landing procedures for aircraft runway landings. In the following, the basic procedures and corresponding terminology for standard airport traffic patterns, approach patterns and landings are presented [42].

1.1.1 Standard Landing Procedures

Figure 1.2 illustrates a standard left-hand traffic pattern (all turns are made to the left) and provides the corresponding terminology. The pattern altitude is typically $1000ft$ above ground level.

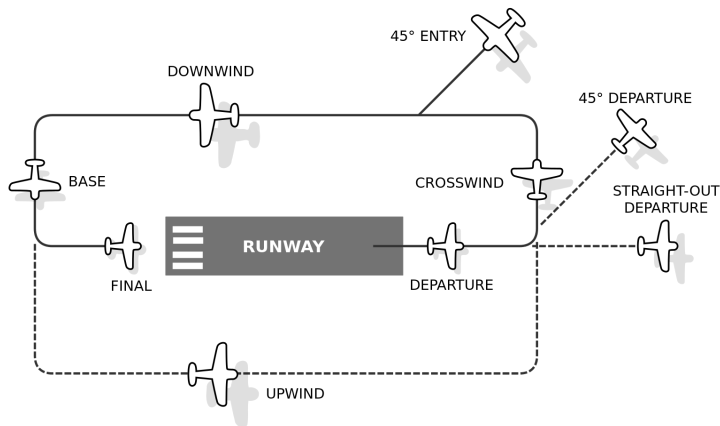


Figure 1.2 "Diagram of a typical left-hand traffic pattern in the US" by Thsutton, licensed under CC BY-SA 2.5 .

When approaching for a normal landing in nominal conditions (engine power available, light wind, final approach into the wind), the traffic pattern is entered in the downwind leg which is usually flown around $1km$ to $1.5km$ off the runway. For a standard landing approach the aircraft should follow the downwind leg until it passes an imaginary 45° line starting from the approach end of the runway, before a medium bank turn is made onto the base leg. The base leg marks the transition from the traffic pattern to the final approach leg, which represents the start of the landing maneuver. During the final approach the aircraft gradually descends (typically on a 3° glideslope) towards the touchdown point on the runway and initiates the flare a few meters above the runway. During the flare the aircraft's pitch attitude is smoothly increased to slow the rate of descent, while the thrust is reduced. Touchdown then happens in a proper landing attitude (pitch angles typically range from 4° to 7°) at an airspeed close to stall speed. After touchdown, the aircraft performs its after-landing roll along the runway centerline until the deceleration to taxiing speed is completed.

1.1.2 Crosswind Landing Procedures

Most modern aircraft are operational beyond ideal atmospheric conditions, but a landing under adverse weather conditions requires special attention and proper procedures. Lateral winds (crosswinds) and non-stationary turbulence effects are the most important disturbances for this application. The established procedures are therefore denominated *crosswind landing procedures* and will be described in the following. Figure 1.3 depicts a heavy crosswind landing of a Boeing 747 at Schiphol airport.



Figure 1.3 "*PH-BFL B747 KLM heavy crosswind landing*" by Maarten Visser, licensed under CC BY 2.0 .

During the final approach a lateral force induced by the crosswind is acting on the aircraft. This force has to be compensated in order to keep the aircraft on a constant course over the runway, which can be achieved either by rotating the aircraft's thrust vector into the wind as shown in Figure 1.4, or by tilting the lift vector as shown in Figure 1.5. Before touchdown the aircraft fuselage needs to be aligned with the runway direction to allow for a safe transition from sustained flight to ground roll. There are basically three recommended and widely employed techniques for the landing of fixed-wing aircraft under crosswind conditions [113, 2, 45], which shall be presented in the following.

De-Crab Landing Technique

The first option consists in using the thrust vector's crosswind component to compensate for the lateral force generated by the crosswind. This method is employed in the so-called *de-crab* and *crab* landing techniques — the *de-crab* technique will be described first. During the final approach the wings are held level, while the heading angle and engine

thrust are adjusted to compensate for the crosswind effect. During the flare (shortly before touchdown) the crab angle is eliminated by applying downwind rudder together with upwind aileron to prevent rolling motion and keep the wings in level position. Touchdown then takes place with the aircraft aligned with the runway direction and wings level. During the final de-crab maneuver some downwind lateral drift is unavoidable. This drift needs to be accounted for by starting the de-crab maneuver with a certain upwind offset from the runway centerline, so that touchdown occurs on the centerline. The procedure is shown in Figure 1.4.

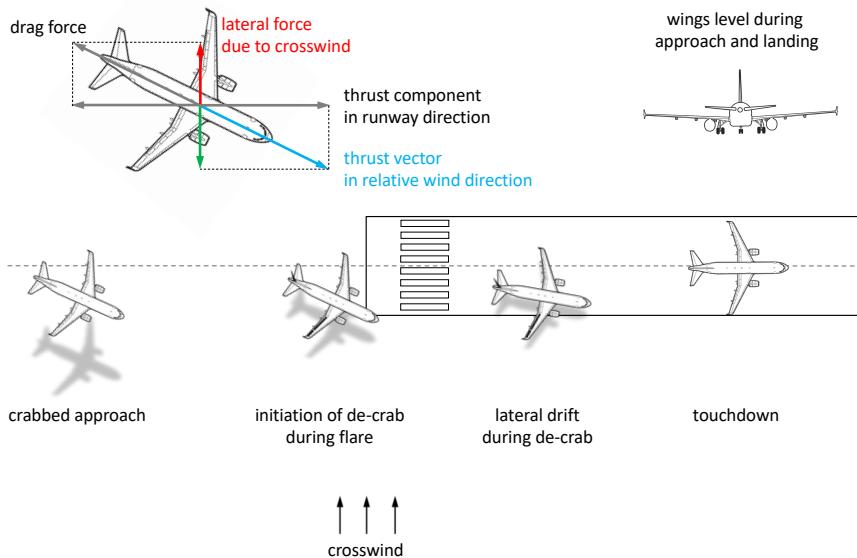


Figure 1.4 De-crab landing technique: 1) wings level during approach, 2) nose points into the wind, 3) position is maintained using engine thrust, 4) during flare downwind rudder is applied to eliminate crab, 5) aileron is applied simultaneously to maintain wings level, 6) at touchdown aircraft, velocity vector and bank angle are aligned with runway, 7) de-crab maneuver results in lateral deviation from point of touchdown.

Sideslip Landing Technique

The second possibility consists in tilting the lift vector sideways and using its crosswind component to compensate for the side force generated by the crosswind. This method is applied during the so-called *sideslip* landing technique as shown in Figure 1.5. Final approach is typically flown in crabbed motion as previously described for the de-crab landing technique, and the aircraft transitions to a steady sideslip condition shortly before

the flare. The sideslip angle is then maintained at a constant value by applying upwind aileron and downwind rudder (such opposite use of aileron and rudder is known as *cross control*), while the lift force is adjusted correctly via the pitch angle, to maintain force equilibrium with the aircraft weight. Precise cross control coordination can be difficult in turbulent weather conditions.

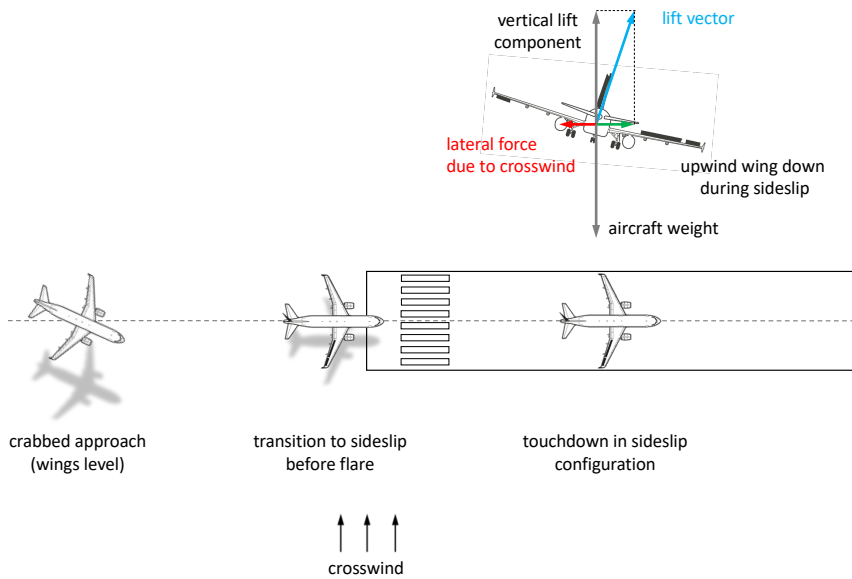


Figure 1.5 Sideslip landing technique: 1) final approach using crab to avoid drift, 2) before flare heading is aligned using aileron into the wind and opposite rudder, 3) aircraft at constant sideslip angle, 4) rudder and aileron must be applied continuously to maintain sideslip angle, 5) aircraft heading aligned with runway during final approach, 6) touchdown usually with upwind main wheels touching the ground first.

Crab Landing Technique

Another possibility is a variation of the de-crab landing technique. Thereby the crabbed aircraft attitude is maintained until touchdown and the aircraft is only realigned with the runway direction once the aircraft landing gear is in contact with the ground. This way the wheel friction is used to minimize the aircraft downwind drift during the final de-crab maneuver on the ground. When landing in dry conditions, the aircraft will move towards the upwind runway edge after touchdown and immediately needs to de-crab and return to the runway centerline during roll-out. The larger the crab angle at touchdown, the larger the deviation from the centerline. Some aircraft manufacturers therefore do not recommend

the use of the crab landing technique on a dry runway [113]. Figure 1.6 illustrates the crab landing technique for wet runway conditions.

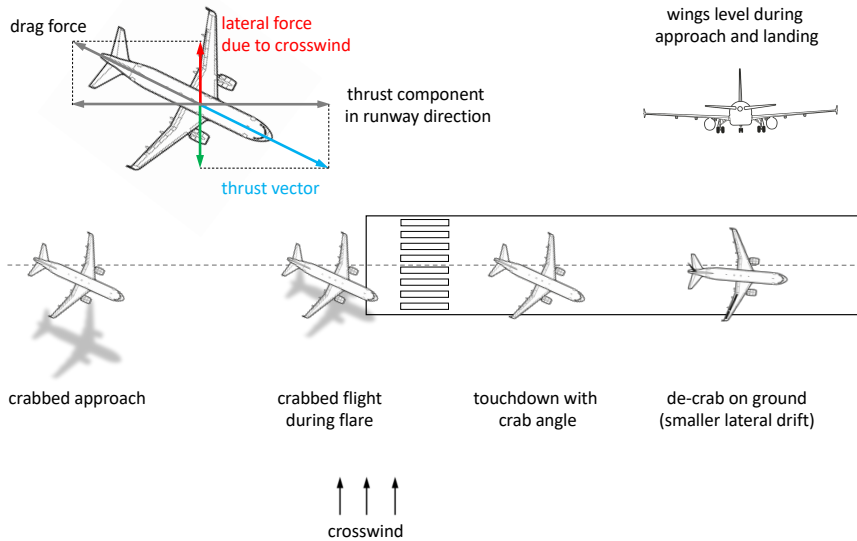


Figure 1.6 Crab landing technique: 1) approach same as during de-crab landing, 2) no de-crab before flare, only after touchdown, 3) on the ground downwind rudder is applied to eliminate crab, 4) aileron is applied immediately to maintain wings level, 5) wheel friction reduces aircraft drift in downwind direction during de-crab on the ground, 6) only recommended on wet or slippery runway, 7) ground-based de-crab maneuver results in smaller lateral deviation from point of touchdown.

Limitations of Standard Landing Techniques

It is important to understand the limitations of the standard landing techniques. Depending on the given aircraft type, limiting factors for the sideslip landing technique include geometrical constraints (maximum attitude angles before tail, wing or engine nacelle ground contact), as well as aerodynamic constraints (available control authority for maintaining a stationary sideslip).

Limiting factors for the de-crab landing technique include the exertion of lateral forces on the landing gear struts and tires caused by inaccurate de-crab control and unavoidable lateral drift during the final de-rotation. Depending on the tire type and pressure these cornering forces can differ, but are generally proportional to the slip angle (angle between wheel orientation and direction of travel) and the weight supported by the wheel. For a

rather small slip angle of 10° the cornering force corresponds to around half the supported weight [42].

To mitigate these limitations and restore some safety margin, a combination of sideslip and de-crab landing techniques is usually applied under heavier crosswind conditions [2, 45]. Such combination also enables a final de-crab maneuver after touchdown in dry conditions, but is limited to tricycle style aircraft — with only the upwind wheel touching the ground at first, a de-rotation becomes possible without damaging the undercarriage. Another mitigation strategy consists in increasing the landing velocity to relax the geometrical constraints and augment control surface effectiveness.

The crab landing technique is not recommended for tailwheel aircraft (conventional landing gear), due to their inherent directional instability during taxiing [42]. Any landing gear configuration, where the aircraft center of gravity lies behind the point of wheel pivoting will exhibit directional instability on the ground and might even lead to a ground loop [122]. Also, the greater tendency of tailwheel aircraft to weathervane during ground operations adds to this problem (tricycle aircraft weathervane as well, but usually to a lesser extent) [42]. Besides the stability issues, a touchdown with non-zero crab angle always exerts lateral loads on the landing gear. As previously mentioned these forces can reach large values, even for smaller crab angles and would be especially harmful to aircraft with a *single-main* or *bicycle* landing gear setup (often found on high performance gliders), where the supported weight and thus lateral forces are distributed among fewer wheels. These limitations basically limit the applicability of the crab landing technique to tricycle aircraft (as found on most modern transport aircraft) and slippery runways.

The above described limitations are especially restrictive for aircraft with large wingspans and low wing loading (and consequently low landing airspeed), since only small roll angles would be acceptable in ground proximity and large crab angles would result even for low crosswinds. In addition, such aircraft types typically possess slow rotational dynamics, which inhibits a fast de-crab maneuver without significant lateral drift. As will be explained in Section 1.3.2, most high altitude aircraft fall into this category. Prominent historical examples include the Lockheed *U-2 Dragon Lady*¹ aircraft, which needs ground based landing assistance via chase cars for wingtip clearance information, and the Boeing *B-52 Stratofortress*² with its large wingspan and negative dihedral. Both aircraft are very sensitive to crosswinds, which in both cases lead to an extension of classical landing techniques, as will be explained in more detail in the Section 1.2. Figure 1.7 illustrates the landing constraints imposed by the design of those aircraft.

Modern examples include HAP and HALE aircraft systems, which are designed for long-term stratospheric flight. These aircraft are of extremely lightweight built, which leads to the above described standard landing techniques being even more restrictive. Figure 1.8 illustrates two examples of such aircraft systems, namely the *Zephyr UAV*³ produced by Airbus (previously QinetiQ), and the Northrop Grumman RQ-4 also known as *Global Hawk*⁴.

¹U-2 Dragon Lady (<https://www.lockheedmartin.com/en-us/products/u2-dragon-lady.html>)

²B52 Stratofortress(<https://www.boeing.com/defense/b-52-bomber/>)

³Zephyr UAV (<https://www.airbus.com/defence/uav/zephyr.html>)

⁴Global Hawk (<https://www.northropgrumman.com/Capabilities/GlobalHawk/Pages/default.aspx>)

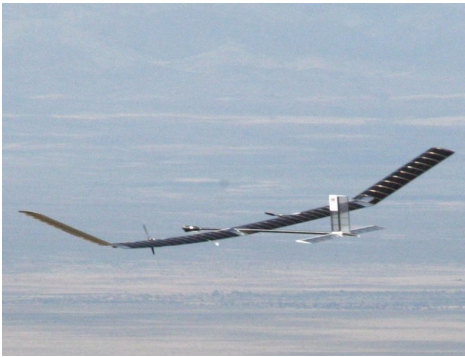


(a) Lockheed U-2 with chase car for landing ground



(b) Boeing B-52 with its large wingspan and negative dihedral, ©.

Figure 1.7 Difficult landing conditions imposed by high altitude aircraft design requirements.



(a) "Zephyr in mid flight during 2008 flight" by QinetiQ Group, licensed under CC BY-NC-ND 2.0.



(b) Global Hawk flying over Edwards Air Force Base in California, ©.

Figure 1.8 HAP aircraft *Zephyr UAV* and HALE UAV *Global Hawk*.

Safe operations would be limited to very favorable weather conditions (low crosswinds), possibly reducing the operational availability to only a few days per year. The solution to the *HAP landing problem* thus requires a new approach and constitutes the main goal of this work.

1.2 The Cooperative Landing Maneuver

As mentioned in the previous section, standard landing techniques might only provide limited possibilities or might not even be applicable for certain aircraft types. A way of further improving the landing performance must be found. Out of the three presented landing techniques, the crab landing technique offers the best performance in terms of associated risk (no de-crab at low altitude) and geometrical constraints, but requires certain

conditions to be met (e.g. wet runway, maximum crab angle) for a safe and effective execution. These requirements are due to the fact that a final de-crab maneuver still is mandatory before the aircraft can perform its roll-out along the runway centerline. With the main landing gear being fixed in forward orientation, any deviation from the runway heading will generate lateral wheel forces at touchdown and thus limits the applicability of ground-based de-crab maneuvers.

A significant improvement of the crab landing technique could be achieved if the final de-crab maneuver was omitted, so that the touchdown and landing roll could be performed in crabbed motion. An interesting approach following this logic consists in steerable main landing gear systems. Such systems are used for either reducing side forces during landing, or entirely eliminating them through *active crosswind assistance* [123], where the undercarriage is rotated by the crab angle to align with the runway direction. To the author's knowledge, there exists only one operational aircraft featuring a full-blown crosswind assistance system — the Boeing B-52 Stratofortress. It possesses a so-called *crosswind crab system* [114], which lets the pilot manually configure a desired landing gear rotation angle in the range of $\pm 20^\circ$. The system enables landing and roll-out in crabbed attitude, and thus allows for wings-level touchdown and roll-out without the need for a de-crab maneuver. Figure 1.9 illustrates the aircraft during a heavy crosswind landing.



Figure 1.9 Boeing B-52 Stratofortress with its steerable main landing gear during a heavy crosswind landing, (©).

Despite the seeming operational simplicity of the solution, the increased weight, mechanical complexity and maintenance demand have prevented the technology from diffusing, although there have been recent efforts to revive the topic [122].

Besides the above described issues related to crosswind landings, another aspect plays an important role for the choice of landing solutions. As previously mentioned, the weight

of the landing gear and associated system components typically ranges between 6% to 15% of a given aircraft's empty weight, which for certain aircraft types like HAPs advances into the region of available payload capacities. A removal of most of the landing gear system weight (certain structural parts are still be required for docking with the ground carriage) would lead to a direct increase in available payload capacity. In the case of the German Aerospace Center's *ELHASPA*⁵ aircraft [66, 65] a payload capacity increase of up to 50% would be possible, while an increase of around 30% is conceivable in the case of the HALE UAV *Global Hawk* [62, 130].

An improvement of flight and landing performance and safety is thus possible if the landing gear is removed from the aircraft, which is the underlying idea of the proposed landing system. As previously described for the *crab landing technique*, the aircraft approaches the ground in crabbed attitude, but instead of performing the touchdown on the runway, it now touches down on a mobile landing platform or *ground carriage* which is moving at the same velocity as the aircraft. The cooperative nature of this approach also allows for cooperative disturbance rejection, where the mobile landing platform adjusts to the disturbed aircraft motion and vice versa. A final de-crab maneuver becomes unnecessary — the aircraft does not have to align to the runway direction, as long as the relative position and velocity of the aircraft and the landing platform are kept close to zero. Touchdown is then performed in crabbed attitude with wings level. The concept is illustrated in Figure 1.10 for DLR's *ELHASPA* aircraft and has been patented under [55].

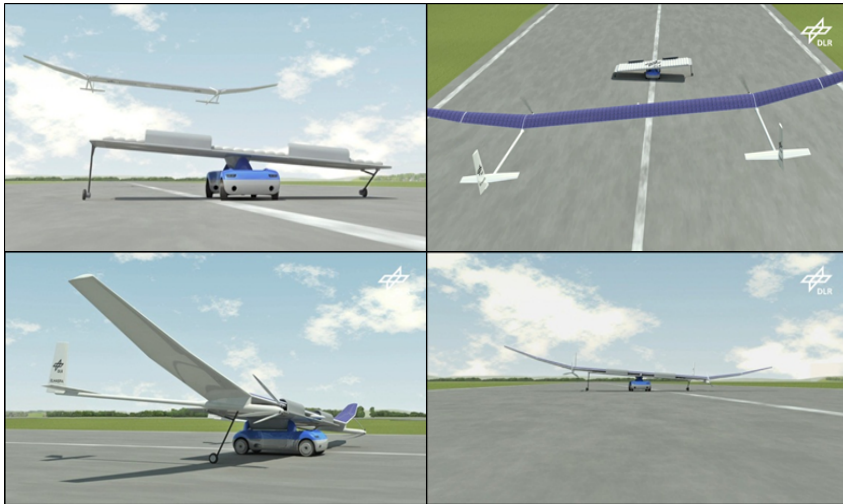


Figure 1.10 High altitude platform *ELHASPA* landing concept (©2016 DLR, reproduced with permission).

The proposed cooperative landing maneuver combines the benefits of a crosswind assistance system with a significant aircraft weight reduction. As will be explained in

⁵Extreme Lightweight High-Altitude Solar Powered Aircraft *ELHASPA*
(<http://www.sfl-gmbh.de/en/blog/projekte/elhaspa/>)

Section 1.3.2, it therefore directly addresses the main inhibitors of commercial HAP operation, namely low payload capacity and high crosswind sensitivity during landing, but also holds the potential of significant improvements for other aircraft types. In the following section three promising fields of application are described, and the specific advantages and current hurdles for the implementation of such ground-based landing system highlighted.

1.3 Applicability to Current Aircraft Systems

1.3.1 Commercial Aviation

The transportation of passengers and cargo by means of aircraft has become a very important business segment in the modern world. Air transport is fast and flexible, when compared to others means of transportation, but also holds several downsides. The environmental impact and operational cost are high, capacity is rather limited, and sustainability is an open issue. Aircraft manufacturers and airlines are putting considerable effort into the optimization of their systems and operational procedures to further maximize performance and efficiency, while yielding only small gains. Nevertheless, there is still room for significant improvement, considering the fact that certain aircraft components and technologies are based on historical developments and available infrastructure, rather than on actual technological feasibility and economic efficiency. As described in Figure 1, the aircraft landing gear is a good example of such system component. One reason leading to this situation can be found in the strict certification procedures applied in aviation, but also in the lack of available alternatives. Recent advances in robotics, sensor and control technology allow for the consideration of new, more efficient landing systems.

The German research project Airport2030 [86] and the European project GABRIEL [102] have investigated the concept of ground-based landing systems for the commercial aviation sector. Such systems could provide significantly higher levels of safety and efficiency and have already caught the attention of the aviation industry. The International Air Transport Association IATA has included the concept into its *Technology Roadmap* [56] and aircraft manufacturer Airbus has added the concept to its *Future by Airbus* program [1].

The example of the landing system developed during the Airport2030 project will be described in more detail to highlight its potential. Figure 1.11 illustrates the concept.

As shown in Figure 1.11, the landing gear is removed from the aircraft and replaced by significantly simpler and lighter mechanical interfaces to connect to a ground carriage, which is mounted onto a landing sled. The ground carriage can move along the landing sled in lateral runway direction and possesses one rotational degree of freedom around its vertical axis. The landing sled itself is mounted onto a dual-rail system, with one rail on each side of the runway, and is electrically driven by two power units (one per rail). During a landing the sled is automatically accelerated to match the position and velocity of the approaching aircraft, while the ground carriage moves laterally to compensate for deviations of the aircraft from the runway centerline. The system then uses its rotational degree of freedom to dynamically adjust to any given crab angle of the aircraft. After



Figure 1.11 Ground-based landing gear system GroLaS designed by German company mb+Partner. (©2012 mb+Partner, reproduced with permission).

touchdown and deceleration, the ground carriage can be detached from the landing sled. It is equipped with its own electrical propulsion system, allowing for emission-free and silent taxiing without the need for running aircraft engines. The system has been patented by the German company mb+Partner under the name GroLaS (Ground-based Landing gear System) and is designed to allow for parallel operation of aircraft with and without landing gear, facilitating the introduction of the new system.

The most important advantages of the system are summarized below [86]:

- Use of ground-based power during take-off, landing and taxiing allows for shorter take-off and landing, eliminates the need for push-back vehicles and thrust reversal and enables the recovery of braking energy during the landing.
- Increase of system efficiency with a reduction of aircraft empty mass by up to 15%, fuel consumption by up to 20% and direct operating cost by up to 12%.
- Lower thrust requirement allows for a decrease of noise levels by up to 35% and carbon dioxide emissions by up to 20%.
- Safer landing procedure in crosswind conditions without the need for de-crab maneuver and automatic disturbance rejection in presence of stronger gusts.

However, there are various issues regarding the implementation of such novel landing systems. The most significant hurdles are listed below [86]:

- Development and investment effort are high with an estimated cost of around 500 million Euros per runway.
- Existing infrastructure has to undergo significant changes, including the installation of the system in pre-defined alternate airports for diversions.

- High certification effort.
- System is aircraft specific and requires adaptation to different aircraft types.
- Introduction phase is expected to be long and difficult and requires an operation of the new landing system in parallel to regular runway landings.

As previously mentioned, a similar solution was developed during the European research project GABRIEL. The main difference with respect to the Airport2030 system, is that the aircraft is actively involved into the synchronization process. The lateral control is done solely by the aircraft, as the landing sled is mounted onto a single-lane maglev track and cannot move sideways. The GroLaS system on the other hand does not include the aircraft into the synchronization control loops in order to keep the changes on the aircraft side as small as possible and thus facilitate the certification process. Nevertheless, the above mentioned advantages and issues are mostly the same for the GABRIEL system and are therefore omitted here. More information can be found in [102].

The biggest issue regarding the introduction of such ground-based landing system concept to the commercial aviation sector stems from the required infrastructural change on existing aircraft and airports, as well as the high certification effort of the highly complex system. These issues will require a significant amount of time to be resolved and a market introduction is not expected before 2030 [56]. The ground-based landing concept is therefore a very promising but long term future technology for the commercial aviation sector. Nevertheless, there are other applications, which could benefit from such landing system in the closer future, where safety requirements and certification issues are less demanding, and where performance aspects are more important than economic efficiency. This is the case for a specific class of unmanned or special purpose aircraft. In the following section one such application will be described in more detail.

1.3.2 High Altitude Platforms and HALE Aircraft

In the recent years, Unmanned Aerial Vehicles (UAVs) have gained widespread interest, and they are now used in many areas. High altitude aircraft, such as HAPs and HALE UAVs are promising in terms of potential applications; however, the development has not yet reached the stage where a commercial use is viable. Intended for a use in the stratosphere, at altitudes in the order of 20km, HAPs typically have large wingspans and very lightweight structures. Equipped with solar cells and sufficient battery capacity, such airplanes can stay airborne day and night, which gives them a theoretically unlimited endurance. This opens up possibilities for various applications, including earth observation, atmospheric science and communication networks. Some of these tasks are now performed by satellites, which are very expensive to build, launch and operate. Apart from saving the cost of a rocket launch, HAPs have more advantages: they are independent of an orbit, and can operate anywhere, either staying at a given location, patrolling over a large area, or relocating as necessary, e.g. in response to a natural disaster. Unlike satellites, aerial vehicles would be able to return to the earth's surface for maintenance or updates, and would also help in diminishing the pestering space debris problem [119, 120].

Over the past years a number of industry and government HAP projects have emerged [66, 28, 81, 35], demonstrating the disruptive potential of such technology. Especially

noteworthy are the investments made and the public attention raised by the internet companies Google and Facebook, in an effort to solve the global connectivity problem [95]. Traditional defense companies have also invested in HAP technology, like Airbus DS with their Zephyr UAV program.

Due to the difficulties inherent to such technology, Airbus today remains the only company with operational HAP aircraft, while a number of competitors ceased their activities. Facebook has decided to stop its *Aquila*⁶ HAP project after a structural failure occurred during the landing approach of the second test flight [116]. Google has announced the end of their Solara HAP program in 2017 due to "*economics and technical feasibility*" issues [14], also having crashed an early prototype in 2015 [115].

One of the main limitations of HAP aircraft are their typically low payload capacity and their high crosswind sensitivity during landing. In order to fly at high altitudes, a low wing loading is necessary, which translates to a large wing area and a low gross weight. However, the batteries required for overnight flight are heavy, and further reduction of the structural mass is very difficult, given the already extreme lightweight design. New approaches are required to lower the empty weight and in turn increase the payload.

As explained in Section 1.2, the approach suggested here is to remove the landing gear. With mission durations of several months, the landing gear, which is only used for take-off and landing, is merely dead weight for most of the time. Without an aircraft-mounted landing gear, new solutions need to be found for take-off and landing. Take-off could be realized using some sort of a temporary gear, which is dropped once the aircraft is airborne. The landing, however, poses a bigger challenge. As mentioned in Section 1.2, the core idea consists in having a landing platform mounted onto a ground vehicle, which is able to accelerate to the aircraft's landing speed and to cooperatively with the aircraft control position and velocity, so that the touchdown on the landing platform takes place without a horizontal velocity component relative to the platform.

This would not only eliminate the need for a landing gear, but also simplify landings in crosswind situations, as the aircraft does not need to align with the runway direction. HAP aircraft are typically very sensitive to wind, due to their size and low weight, in which the proposed landing procedure could greatly improve their operational availability. In the *Aquila* UAV accident report [116] it is stated that "*the operator expected some damage during normal landings*", which is a confirmation of the unsolved landing problem for HAP aircraft.

In contrast to the commercial aviation landing systems described in Section 1.3.1, the proposed HAP landing system makes full use of the dynamic properties of the involved vehicles. Both, aircraft and ground vehicle actively contribute to the optimal execution of the landing maneuver, allowing for a more flexible adaptation to weather conditions or operator needs. In addition, there is a current industrial need [34] for a viable landing solution for ultralight high altitude aircraft. The developed landing system, which will be presented in this work is thus focusing on the high altitude application. Nevertheless the developed technologies have a generic character and are in many aspects valid for any type of fixed-wing aircraft.

⁶Aquila HAP (<https://engineering.fb.com/connectivity/building-communications-networks-in-the-stratosphere/>)

1.3.3 Heavier Unmanned Aircraft Systems

Besides the two applications mentioned above, there is a third class of aircraft system, which could strongly benefit from the use of a ground-based landing system. The main idea is to operate heavier, medium-sized aircraft systems of up to 600kg (German "*Ultraleicht-Klasse*"), independent of any permanently installed infrastructure. Such runway-independent operation is already possible for smaller UAVs (around 20kg MTOW) like the Boeing *ScanEagle*⁷ by means of a catapult-launch and a landing device named *SkyHook*⁸. A somewhat similar, but not yet operational concept is the Aurora *Sidearm*⁹, which is supposed to serve UAVs of up to 200kg takeoff weight. Other, mostly academic and lightweight examples make use of recovery nets [61] or parachutes [118]. For mechanical reasons, it seems quite obvious that such solutions have limited applicability regarding the landing of heavier aircraft systems with up to 600kg. Also, they only represent a landing solution and a second device is required for takeoff.

A ground-based landing system, with a mobile rail- and sled-based architecture (e.g. based on the GroLaS landing system) would allow for runway-independent takeoff and landing of heavier aircraft systems and would thus introduce this aircraft class to new potential fields of application (e.g. automated monitoring of forest fires in remote locations). At the same time it would introduce all the advantages mentioned in the previous sections, such as increased performance (lower aircraft weight) and efficiency (ground-based power during takeoff and landing), simplified crosswind landing (no de-crab), autonomous operation (24/7), short takeoff and landing, and cooperative disturbance rejection. In an effort to develop a first version of such mobile runway system, a German research project called *REALISE*¹⁰ has been initiated by the GroLaS inventor mb+partner in 2016. Its goals is the adaptation of the GroLaS landing system to lighter aircraft systems, and to demonstrate the practical feasibility of the concept with a small UAV of around 20kg. A follow-up project with the goal of industrializing the solution for aircraft of up to 600kg is supposed to start in 2020. Figure 1.12 illustrates the concept.

1.4 Review of Related Applications

This section addresses a number of applications which are technically related to the envisioned cooperative landing concept. Most importantly, state-of-the-art research on novel landing concepts for fixed-wing aircraft is discussed. Then, related work on modern autoland systems, as well as the automatic landing of micro air vehicles (MAVs) on mobile ground vehicles is presented. Lastly, a review of other related applications and techniques (e.g. consensus problems) is provided.

⁷Boeing ScanEagle UAV (<https://www.insitu.com/information-delivery/unmanned-systems/scaneagle>)

⁸Insitu SkyHook

(https://www.insitu.com/images/uploads/pdfs/Skyhook_Universal_ProductCard_PR041615.pdf)

⁹Aurora Sidearm Launch & Recovery System (<https://www.youtube.com/watch?v=2Kj6KnBVTLC>)

¹⁰REALISE - The Mobile Runway System (<https://www.realise.aero>)



Figure 1.12 Conceptual illustration of mobile runway system *ELEKTRAIL*[®] for runway-independent operation of heavier aircraft systems up to 600kg. (©2019 ELEKTRAIL[®] project consortium, reproduced with permission).

1.4.1 Novel Aircraft Landing Concepts

As introduced in Section 1.3.1, several research projects have investigated novel landing concepts in order to overcome the shortcomings of current landing procedures. A notable example can be found in the European-FP7 funded project GABRIEL (Integrated Ground and on-Board system for Support of the Aircraft Safe Take-off and Landing). In GABRIEL, the main proposed concept seeks to assess the feasibility of a ground-based structure to launch and recover large aircraft from and onto a mobile landing pad, which is powered by a magnetic levitation system (Maglev). A preliminary design has been based on a cart-sledge system, where the sledge is permanently connected to a Maglev ramp. The aircraft would be put on a cart attached to the sledge and accelerated until reaching sufficient speed to fly. As no undercarriage would be needed, clear advantages are the cut of required engine take-off thrust and the reduction of the aircraft empty weight, noise, fuel consumption, and chemical emissions [103]. However, despite the work being focused on commercial airliners, only relatively small scale experiments (compared to this thesis) have been demonstrated. Furthermore, the concept does not offer a lateral correction of relative position errors from the ground side. Despite GABRIEL being a compelling concept with several benefits, the main concern is on the rendezvous control, i.e. the guidance of the *landing gear-less* aircraft towards a sledge, which is constrained to move on a straight line [103]. This thesis shows the benefit of lateral position correction along with a larger scale technical demonstration.

GroLaS (Ground-based Landing gear System) is an aircraft carriage system that expands on the concept created for GABRIEL. As described in Section 1.3.1, it consists of a sledge

which accelerates and decelerates subject to a mechanical drive system, on both sides of the runway. This allows for dual usability by conventional aircraft and aircraft with no landing gear. The concept works as follows — immediately after landing, the sledge automatically accelerates the ground carriage to the speed of the approaching aircraft before touchdown and adjusts its position to the aircraft. Then, pins affixed to the ground carriage couple into corresponding interfaces installed on the aircraft. This concept has potential benefits in the reduction of braking distance without the use of reverse thrust [86], while experimental demonstration is yet a work-in-progress. Another interesting concept is REALISE, which is currently under development and funded by the German government. REALISE is a direct derivate from GroLaS and investigates a new technology that allows flexible operation of smaller scale UAVs independent of conventional runway infrastructure. The goal is to create a mobile runway system that is easily set up and operated [97].

Looking for independence from increasingly obsolete aircraft carriers (due to the growing missile threat), the Defense Advanced Research Agency (DARPA) developed the Project SideArm, an effort to create a self-contained, portable apparatus able to horizontally launch and retrieve UAVs from ships, trucks, and so on [25]. Created by Aurora Flight Sciences, it is able to work with aircraft up to 900 pounds. The idea consists in the UAV being launched by a catapult, and retrieved by an aircraft-mounted hook which catches a cable below a large metal rail attached to the ground unit, arresting the aircraft and swinging it upward, where barbs on the aircraft's nose snag into a net [22]. The SideArm system is restricted to fairly low aircraft weights, which limits the applicability of the overall concept. What differentiates the present work from the aforementioned endeavors is that here both aircraft and ground vehicle can cooperate in a bilateral setup throughout the landing maneuver.

1.4.2 Automatic Runway Landing

The problem of automatically landing an aircraft has been studied for several decades. As an example, [17] already discussed the necessity for automated landings in the 1950s. Such automation requires precise aircraft positioning information, which is typically provided by ground infrastructure at the airport known as Instrument Landing System (ILS). Even if the pilot is still required to manually perform the final phase of the landing for lower ILS-categories, auxiliary systems are key to reduce the likelihood of accidents and mishaps [41]. However, deployments of these instrument landing systems often remain challenging. As [70] points out, smaller airfields are typically not equipped with the infrastructure required for automatic landing, such as radar beacons. In addition, altitude measurements from today's GPS sensors are still not accurate enough for the procedure, due to e.g. atmospheric disturbances and interferences, while laser or radio altimeters are rather expensive. Meanwhile, basic technology for flight guidance and control has become increasingly accessible for smaller airplanes over the last decade. Automatic landings are particularly important for UAVs, but maintaining both robustness to uncertain system characteristics and affordability can be challenging. [69] tackles these problems and presents a proof of concept based on a finite state machine, where available autopilot modules are also employed during the final landing phase of the SAGITTA

Demonstrator UAV. To avoid costly upgrades of infrastructure, the emphasis must be on on-board sensing and control. Consequently, several groups developed autoland systems with robotic technologies. A vision-augmented navigation system independent of ground infrastructure is a crucial aspect of the concept. The computer vision systems can detect and track the airport's runway during an approach, using one or more cameras mounted on the aircraft [73]. This information is used to obtain estimates of relative position, velocity, and attitude, which are then fed to the control system guiding the landing operations [50]. A number of nonlinear methods have been applied to landing controllers, such as feedback linearization, intelligent control, robust control, hybrid control and so on. Nevertheless, most aircraft use PID as low-level control technique, and vision-based estimates as high-level input provided to the PID controllers [47]. The use of vision data in the control loops is particularly important if the location of the runway or landing pad is unknown or non-stationary, e.g. the deck of a ship. Classical approaches focus on object recognition using edge detection techniques. Some of the approaches include image segmentation (used to identify lines, object boundaries and curves), image moments (weighted averages of image pixel intensities, generally used to describe objects after segmentation), monocular vision (where each *eye* is used separately), and stereo vision (where 3D information is extracted from images) [47]. Recently, [20] carried out a comparison of a few configurations and showed that vision-based guidance strategies are able to perform accurate landings without GPS or ILS data, especially if a barometric altitude corrective term was added to avoid final horizontal divergence. A very recent effort at the Technical University of Munich was successful in automatically landing a research aircraft, a modified Diamond DA42, with vision-assisted navigation. The researchers designed an optical reference system, using a camera for the normal visible range and an infrared camera for poor visibility conditions¹¹.

1.4.3 Landing of Micro Air Vehicles on Mobile Ground Vehicles

As micro air vehicles became widespread in recent years, innovative solutions have been proposed to alleviate issues like limited range and the difficulty to fully automate deployment and recovery. A solution to the latter is to execute these tasks from a ground vehicle, especially for rescue operations, where a cooperation between aerial and ground stations could save precious mission time [11].

Over the last few years, the idea of integrating ground vehicles and MAVs has been documented in several works. In general, the procedure involves applying a variation of the Kalman filter to fuse INS and visual fiducial data for pose estimation. The landing is typically tackled as a two-stage task — approach and terminal landing. The strategies employed in each phase differ. Although originally developed as a homing guidance law for ballistic missiles, Proportional Navigation (PN) has become an interesting technique for UAV guidance as of late. It is based on the principle that two vehicles are on collision course if their direct line-of-sight holds constant direction as the range closes. The idea is analogous to proportional control, in that a moving pursuer homes in on a moving target by turning with a rate proportional to the turn rate of the line-of-sight [49]. [48] compares PN to pure pursuit and line-of-sight guidance laws, with the former proving to

¹¹www.sciencedaily.com/releases/2019/07/190704191353.htm

be superior to the other two with respect to acceleration required and time to reach the target. A PN-based guidance law was also the choice of [11]. Within close range, PN is no longer efficient, and a smooth transition must be implemented to switch strategies.

For the terminal landing phase on a moving platform, [80] used a computationally cheap, image-based visual servoing approach to track the platform in two-dimensional image space and generate velocity commands. The data is fed to an adaptive sliding mode controller of a quadrotor MAV, which then lands on the moving target. [128] used fiducial markers to detect the landing platform, enhancing the measurements with a sensor fusion approach based on a square-root unscented Kalman filter. [111] implemented a PN-PD scheme, where the system switches from PN to PD when a certain criterion is fulfilled. [11] utilizes a PID controller for the terminal phase, while applying commercially available, low-cost sensors and algorithms combined in a Kalman filter for relative position and velocity estimation. [117] adopted an extended visual-inertial odometry framework for SLAM and a suite of estimation algorithms, while closing the position loop with a model predictive controller.

In [37], researchers were able to land a quadrotor on a moving platform using only on-board sensing and computing, without relying on any ground-based infrastructure. They used computer vision algorithms and multi-sensor fusion for robot localization, platform detection, and motion estimation. [101] also performed the task without any additional infrastructure, relying on multirotor onboard sensing (IMU, altitude sensors and RGB camera) and applying DDPG (Deep Deterministic Policy Gradients), a model-free deep reinforcement learning algorithm capable of learning from continuous state and action spaces. An agent was trained to carry out the landing maneuver on top of a moving platform without any prior human knowledge.

1.4.4 Other Related Techniques

A number of control techniques have been applied to automatic landing and similar application as of late. MPC can be viewed as an optimal controller in the presence of constraints. The control action is the solution of a receding horizon optimal control problem, in which a process model is utilized to predict the system's states subject to constraints. The control input is recalculated at each sampling period to account for new measurements. The technique has become widely popular because it allows for a systematic treatment of constraints, is directly extendable to MIMO systems and delayed systems with almost no modification, accepts both transfer functions and state-space models, and is available in a number of commercial software modules. Once deemed computationally costly for real-time applications, MPC benefited greatly from recent progress in processing, enabling the true power of a technique more powerful than PID control, even for single loops with no constraints [88]. In [51] researchers were able to automatically land a UAV using MPC for the surveillance of internal autopilot controls. [68] used LQR and MPC to improve shipboard landing performance of a fixed-wing UAV. [43] adopted MPC to land a UAV on a moving platform. [125] used MPC with an additional term to develop a modified longitudinal guidance law to reduce the landing risk of an automatic carrier landing system.

Several robust control techniques are also commonly applied to the landing problem, especially the popular mixed $\mathcal{H}_2/\mathcal{H}_\infty$ approach. The \mathcal{H}_2 method is analogous to the famous LQG problem, hence based on quadratic performance. The \mathcal{H}_∞ norm, on the other hand, amounts to the worst case scenario when it comes to disturbance influence on the output. Therefore, mixed $\mathcal{H}_2/\mathcal{H}_\infty$ control represents a state feedback controller, which minimizes the \mathcal{H}_2 norm, such that the \mathcal{H}_∞ norm is inferior to a certain positive number [126]. The controller can also be designed via mixed sensitivity, which corresponds to shaping the sensitivity and complementary sensitivity functions of the system, so that certain performance and robustness criteria are met after the loop is closed. Both \mathcal{H}_2 and \mathcal{H}_∞ optimization may be used [71]. [107] developed a mixed $\mathcal{H}_2/\mathcal{H}_\infty$ controller for an autoland system of a commercial aircraft, where \mathcal{H}_2 was used to obtain an optimal gain for trajectory optimization and \mathcal{H}_∞ to minimize disturbance effects on the performance output. [126] employed mixed $\mathcal{H}_2/\mathcal{H}_\infty$ control for landing as well. The controller is used to ensure the UAV tracks the commanded landing trajectory with preferable qualities, despite the influence of uncertainties and disturbances. Its performance was also compared to a classical, root-locus based controller, where the former was shown to be superior.

Formation control is a fundamental problem for multi-agent systems, where each agent must be commanded at a local level to achieve a collective outcome or maintain a desired structure. Therefore, regarding the underlying control problem, the field exhibits some similarities to the present work. One of the techniques found in the literature consists in the establishment of Virtual Leaders (VL) as a means to move formations held together by Artificial Potentials (AP). Artificial potentials define interaction control forces between neighboring vehicles and seek to enforce a desired spacing between them, using repulsive forces to avoid agent collisions. A virtual leader is merely a moving reference point that influences vehicles in its neighborhood via additional artificial potentials. Such technique can be used to achieve a desired group formation while tracking a path or direct its motion [82]. [32] applied the concept along with MPC to a situation where agents collaboratively adapt the formation and its motion in a distributed manner. Coordinated control for missile guidance is another related area. A PID-based (PI with direct velocity feedback) compensator was used for the low-level coordination control loop in [78]. As previously mentioned, a similar approach found in missile guidance literature consists in the so-called proportional navigation method as described in [93].

Multi-agent rendezvous is another problem related to the scope of this work, since it deals with the dynamics of a relative motion problem. The rendezvous problem requires that each agent arrives at a location simultaneously [16]. In other words, the objective is that a set of agents placed in a certain domain (a closed disk centered at the agent's current position), are all required to meet at the same place and time, within the same domain. Each agent is able to continuously track the positions of all other agents currently within its domain. The problem consists in causing all members to rendezvous through local individual control strategies, without communication between members [85]. The problem might be complex due to characteristics such as quantization, asynchronism, media access, power control, and others. [85] proposed a solution to both the asynchronous and synchronous cases. [24] used proximity graphs to solve the problem for link failures. [131] studied minimalism in sensing and control, assuming each agent moves like a Dubins car and has limited sensor reporting, including only the presence of another agent within some

section of its Field Of View (FOV). A quantized law, in a way that each agent tracks another agent by maintaining it within a given FOV-sector was used. In essence, the cooperative landing scenario represents a similar problem, but is less strict, since the exact location or time of the touchdown are not determined prior to landing, which is only required to take place within the available runway length.

An additional problem affine to this work consists in the control of aerial refueling, particularly autonomous aerial refueling also known as Automatic Air-to-Air Refueling (A3R). The problem is complex due to the close proximity of the vehicles and the dynamic effects they suffer. Besides the time-varying mass and inertia properties during fuel transfer, the wake turbulence caused by the tanker's trailing vortices make the control task very challenging. In formation flight, such effect is beneficial to the follower due to reduced drag, but detrimental to the stability of the receiver. Its dynamics are complex, nonlinear and coupled to those of the tanker. A position-tracking controller is necessary so that the receiver follows the commanded trajectory, which is constant with respect to the tanker's body frame during the station-keeping phase [27]. The two methods widely applied for manned aircraft were also extended to UAVs: Boom-Receptacle Refueling (BRR) and Probe-Drogue Refueling (PDR). The PDR system consists of a hose equipped with a cone-shaped refueling basket, where the pilot of the receiver aircraft has to maneuver the aircraft such that the probe is inserted into the refueling basket. The BRR system consists of an articulated tube located under the rear of the tanker aircraft fuselage. It can be deployed and controlled via two movable airfoils, acting as control surfaces. The pilot of the receiver does not need to actively establish contact with the boom [44]. Historically, PID control has been the most common technique applied to aerial refueling, as can be found in [104]. Nevertheless, robust and optimal control have gained some attention in the field. [19] used the μ – synthesis technique with D-K iteration to design a robust controller for probe-drogue refueling, and considered parametric and non-parametric disturbances, such as mass change and turbulence. Although still incipient for the time being, this should soon become a prosperous research area, due to the need for extended mission time and the proliferation of UAVs.

Other related techniques can be found in the field of automatic net recovery. A vision-based approach has been explored in [100], where a net is used to catch the aircraft while a vision algorithm is used to calculate the bearing angle and completely substitutes the use of differential GPS. In [63] a net suspended by two multirotor UAVs is used to recover a fixed-wing UAV in-air by means of RTK-GPS navigation. In the first case the net is fixed on the ground and the UAV flies directly into it, while in the second case the fixed-wing UAV flies on a straight line, while the multirotor UAVs coordinate the net motion. In both cases the overall motion strategy is unilateral, which means that only one actor performs active coordination control. For the cooperative landing approach discussed in this thesis, the net is attached to the ground vehicle and once the vehicles are aligned, the aircraft performs a *vertical landing* (relative to the landing platform), similar to the helicopter landing case shown in [106], where both vehicles actively contribute to the successful execution of the landing maneuver.

Essentially, the autonomous landing of an aircraft on a moving platform implies a cooperative control problem. Nonetheless, most of the research in the cooperative control literature deals with multi-agent systems [98] and is orthogonal to the proposed cooperative

control strategy. For this class of systems, shared information, role assignment and task division among a group of agents are common difficulties, which require the development of specific techniques and methods for their solution (e.g. consensus problem, graph theory). These solutions do not necessarily apply to the application of interest. Another seemingly related method is a consensus algorithm for distributed systems [87], which is aimed at driving the agents' states to a common value. However, this technique is mostly applicable to problems with many agents, as for example, a swarm of robots, where the agents usually carry a limited sensor suite onboard. On the other hand, for the cooperative landing concept, there are only two agents, namely aircraft and mobile landing platform. Thus, a simpler controller structure can substitute the more complex schemes generally used to solve consensus problems. However, some notable ideas of distributed sensing, computing and control are readily used within the proposed architecture. In [100], a consensus-like scheme is proposed to solve the problem of position synchronization between two cooperative robots, a problem closer to the one addressed in this thesis. However, only position measurements were available for feedback, which is not a restriction in the present application, where velocity is also available for feedback.

For a cooperative control problem involving two agents, the vehicles can be assumed to communicate knowledge over a wireless network. For example, position sensors such as GPS can be deployed for relative state estimation [39]. However, in an outdoor setting, effects due to imperfect communication such as time delays and packet loss are inevitable and can destabilize the overall system. As will be introduced in the following, two solutions to this problem are proposed in this thesis — the first one involves a controller structure that remains stable for any amount of communication time delay. The second approach uses an energy-based passivity controller as a supervisory system to guarantee the stability of the overall system even in the presence of large time delays.

1.5 Scope of Work

1.5.1 Main Contribution

The overall goal of this work consists in the development of an integrated landing solution for extremely lightweight fixed-wing aircraft without a landing gear. The focus is thereby on the practicality and reliability of the solution, while maintaining a generic character and thus the applicability of the developed concept to a larger variety of fixed-wing aircraft. The main contributions of this thesis will be presented in the following.

Concept Development

A first contribution consists in the development of an appropriate concept for solving the current problems inherent to high altitude platforms, namely the low payload capacity and the high crosswind landing sensitivity. The removal of the landing gear system from the aircraft and the introduction of a ground-carriage is identified as a possible solution, requiring a novel landing concept. For its development a thorough analysis of established landing procedures and techniques is performed and their limitations are identified. A new cooperative landing strategy is proposed and a systematic specification of the proposed landing system by means of requirements definition and functional decomposition is

performed. An optimal cooperation strategy is derived, leading to the development of two distinct approaches — the first focussing on the safety of the landing maneuver and the second on its performance (minimization of used runway length). An important contribution consists in this very distinction and the identification of each system's benefits.

Control Strategy Development

The second important contribution is related to the development of the landing control strategy. It includes the identification and implementation of aircraft and ground vehicle controllers suitable for the proposed landing control strategy (requiring independent controllability of the translational degrees of freedom). An energy-based flight controller is implemented to achieve a decoupling of altitude and velocity control on the aircraft side. A reliable and precise relative state estimation system is proposed, based on redundant sensor data. A safety-based landing controller is proposed, which adheres to established landing procedures and has time delay-independent stability properties. A performance-based landing controller is proposed, which utilizes optimized vehicle trajectories and bilateral position synchronization. A stability assessment of both landing controllers in the presence of time delay is performed and a new passivity-based stabilizing controller is proposed for dealing with excessive inter-vehicle communication time delays. The selected aircraft controller also operates on energy level and enables the use of such passivity controller. A state machine-based mission control and fail-safe system is developed to guarantee the operational safety of the landing system.

Modeling, Simulation and Experimental Validation

The third contribution consists in the development of appropriate modeling tools and the validation of the proposed landing system through landing experiments. A high-fidelity aircraft model is developed via time domain system identification. A realistic representation of the flow field around the ground vehicle and landing platform is derived by means of computational fluid dynamics. An optimization-based control design method is applied to the landing controller as well as the individual vehicle controllers. The aircraft model and controller are validated in flight experiments. A realistic, hardware-in-the-loop (HIL) simulation environment is developed, and performance bounds are determined by means of Monte Carlo simulations. The performance bounds are estimated for the given demonstrator setup (semi-autonomous ground vehicle with human driver), and for a hypothetical autonomous ground vehicle (with fast actuators). Finally, the developed landing system is validated through a series of successful real world landing experiments. To the author's best knowledge it was the first time that such technology was demonstrated on the given scale, worldwide. The significant public response to the disseminated results confirms the relevance of this technology.

1.5.2 Experimental Setup

The employed experimental setup has been developed in the scope of the European research project EC-Safemobil¹², where the overall goal was to experimentally demonstrate the feasibility of the envisioned landing concept. For this purpose, a small UAV was used

¹²European Project EC-Safemobil (<http://www.ec-safemobil-project.eu>)

together with a landing platform mounted on a car with a human driver. The demonstrator setup is shown in Figure 1.13 and will be described in detail in the following.



Figure 1.13 Demonstrator setup for landing experiments in the scope of the EC-Safemobil project (©2016 DLR, reproduced by permission).

The proposed landing system used as technology demonstrator consists of an aerial vehicle and a ground vehicle, referred to as UAV and UGV, respectively. Both vehicles are shown in Figure 1.14. The UAV is a commercially available product¹³ with a wingspan of 3.3m and a maximum take-off mass of 21.5kg. It has been equipped with a flight control system and sensors including RTK-GPS (Novatel FlexPak6), IMU (MicroStrain 3DM-GX3-25), pitot tube (Simtec PSS8), and an optical camera (Allied Vision Prosilica GC1380). The ground vehicle is a modified passenger car equipped with an on-board computer as well as RTK-GPS, IMU, a Graphical User Interface (GUI), and optical markers on the roof of the car.

The landing platform is mounted on the car's roof and consists of an aluminum frame and a net. It is 5m wide and 4m long. The net tension is adapted to the weight of the UAV and the expected maximum touchdown impact. The mesh spacing has been chosen in accordance with the UAV wheel dimensions, so that upon touchdown the wheels push through the net and the aircraft fuselage comes to rest on the net. A mechanical locking mechanism (see Figure 4.10) is then used to secure the aircraft safely in the landing platform during the deceleration process. The high level cooperative control loops are closed through a wireless data connection on 5.8GHz, using a lag compensation term for the generation of smooth signals despite communication lags.

¹³Penguin BE UAV (<http://www.uavfactory.com/>)

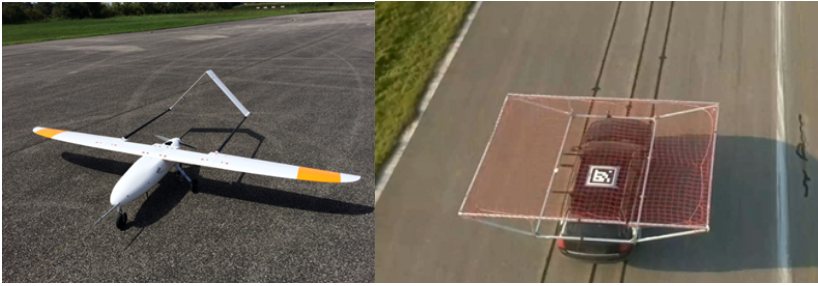


Figure 1.14 Demonstrator setup for landing experiments consisting of aerial vehicle and ground vehicle with roof-mounted landing platform.

1.5.3 Document Structure

The present document is separated into seven different chapters, which shall be introduced briefly in the following.

Chapter 1 introduces the context of the thesis and the industrial need for more capable high altitude unmanned aircraft. The current limitations of such aircraft type are presented and a possible solution is proposed by removing the landing gear from the aircraft and introducing a mobile landing platform. A cooperative landing maneuver is defined, which is based on redundant relative state estimation and cooperative motion control. Current landing techniques and related applications are analyzed to gain deeper insight into the problem. The main research question is formulated, which consists in finding an optimal cooperation strategy for the landing maneuver.

Chapter 2 defines the high-level requirements of the envisioned landing system. A functional decomposition is performed to derive the necessary functional building blocks, which are then mapped to an appropriate controller structure for the cooperative landing controller, as well as the individual vehicle controllers. The process leads to a set of design choices, which include the development of two distinct landing strategies, resulting in a safety-based and a performance-based landing controller. These early design choices constitute the starting point for the detailed development process.

Chapter 3 presents the derivation of the vehicle simulation models and controllers. Two aircraft models are developed — a less accurate but globally valid *simulation model* (valid over the full operational flight envelope), which is based on the aircraft geometry and weight and balance measurements, and a high-fidelity *design model* which is derived via time domain system identification and is suitable for controller design. The flow field around the ground vehicle is modeled by means of computational fluid dynamics and incorporated into the dynamical simulation. The aircraft and ground vehicle controllers are developed according to the functional requirements defined in Chapter 2 and the controller performance is analyzed and validated in simulation and in flight experiments.

Chapter 4 presents the developed relative state estimation system and two variants of the landing control system. The safety-based controller is robust to communication delays and adheres to established landing procedures, while minimizing system complexity. The performance-based controller is based on vehicle trajectory optimization for the approach

phase and bilateral position synchronization during the final landing phase. A time domain passivity-based controller is developed and adapted to the landing application to enforce stability in case of high communication delays. A high-level mission control scheme is developed to autonomously guide the vehicles from approach to landing, as well as a fail-safe system which guarantees the operational safety of the landing system. The controller design procedure is described and its stability assessed in the presence of time delay. A sensitivity analysis in the frequency and time domain is performed and first performance bounds of the developed landing systems are obtained by means of repeated random sampling.

Chapter 5 describes the performed simulations experiments. The developed simulation environment is described, and a detailed performance analysis is presented. This analysis consists of deterministic simulations, which are used to estimate system performance bounds under controlled conditions, as well as stochastic simulations. The stochastic simulation experiments include a large number of test points (around 3000 per test case), and allows for validating the robustness of the system against random variations of given system inputs (e.g. wind gusts, turbulence, initial conditions) and parameters (communication time delay). Based on the results from these simulations the controller architecture (safety-based) for the demonstrator setup is selected.

Chapter 6 describes the performed real world experiments. First a set of preliminary experiments is presented, which analyze individual aspects of the landing system (e.g. validation of relative state estimation system, ground lock concept), before a number of hardware in the loop experiments are shown. The landing is thereby performed with the real ground vehicle and communication systems, and a simulated aircraft, which allows for validating the ground vehicle steering concept and the tuning of the ground vehicle controller. Then, a set of experiments with the full demonstrator setup (including the real aircraft) is performed. First a number of landing attempts on top of a virtual landing platform are performed. The virtual landing platform is elevated to a safe altitude above the real ground vehicle (e.g. 80m offset). Finally the developed landing technology is demonstrated by performing several successful landing experiments including touchdown and deceleration.

Chapter 7 concludes the work and provides an outlook for further development potential.

1.5.4 Scientific Output and Public Response

In the following a list of relevant publications, supervised students, and hyperlinks to different media reports on the EC-Safemobil project results is provided.

Publications

The following publications have been produced during the course of this thesis. They are either directly or indirectly related to the work presented here.

- **Muskardin, T.**, Balmer, G., Persson, L., Wlach, S., Laiacker, M., Lee, J., Ollero, A., Kondak, K.: Landing Fixed-Wing UAV on Mobile Platforms. Springer Handbook of Unmanned Aerial Vehicles (accepted for second edition, 2020)

- **Muskardin, T.**, Balmer, G., Persson, L., Wlach, S., Laiacker, M., Ollero, A., Kondak, K.: A Novel Landing System to Increase Payload Capacity and Operational Availability of High Altitude Long Endurance UAVS. *Journal of Intelligent & Robotic Systems* (2017)
- **Muskardin, T.**, Balmer, G., Persson, L., Wlach, S., Laiacker, M., Ollero, A., Kondak, K.: A Novel Landing System to Increase Payload Capacity and Operational Availability of High Altitude Long Endurance UAV. *International Conference on Unmanned Aircraft Systems* (2016)
- **Muskardin, T.**, et al.: Landing of a Fixed-wing UAV on a Mobile Ground Vehicle. *IEEE International Conference on Robotics and Automation* (2016)
- Persson, L., **Muskardin, T.**, Wahlberg, B.: Cooperative Rendezvous of Ground Vehicle and Aerial Vehicle. *IEEE Conference on Decision and Control* (2017)
- Laiacker, M., Kondak, K., Schwarzbach, M., **Muskardin, T.**: Vision Aided Automatic Landing System for Fixed Wing UAV, *IEEE International Conference on Intelligent Robots and Systems* (2013)
- Garcia, M., **Muskardin, T.**, Viguria, A., Laiacker, M., Ollero, A., Kondak, K.: Analysis and Development of a Reliable Fixed-wing UAV Control System For Mission Profiles with Restricted GPS Availability, *International Conference on Unmanned Aircraft Systems* (2013)
- Balmer, G., **Muskardin, T.**, Wlach, S., Kondak, K.: Enhancing Model-Free Wind Estimation for Fixed-Wing UAV. *IEEE International Conference on Robotics and Automation* (2018)
- Lee, J., **Muskardin, T.**, Ruiz Paez, C., Oettershagen, P., Stastny, P., Sa, I., Siegwart, R., Kondak, K.: Towards Autonomous Stratospheric Flight: A Generic Global System Identification Framework for Fixed-Wing Platforms. *IEEE International Conference on Intelligent Robots and Systems* (2018)
- Coelho, A., Singh, H., **Muskardin, T.**, Balachandran, R., Kondak, K.: Smoother Position-Drift Compensation for Time Domain Passivity Approach based Teleoperation. *IEEE International Conference on Intelligent Robots and Systems* (2018)

Supervised Master and Bachelor Students

The following Master's and Bachelor's student have been supervised during the thesis (in chronological order starting with the first student).

- Georg Balmer (KTH Stockholm, master's thesis): *Modelling and Control of a Fixed-wing UAV for Landings on a Mobile Landing Platform*
- Linnea Persson (KTH Stockholm, master's thesis): *Cooperative Control for Landing a Fixed-Wing Unmanned Aerial Vehicle on a Ground Vehicle*
- Ane Élide Nogueira Frauches Almoia (IFF Campos, bachelor's thesis): *Identificação de um Veículo Aéreo Não Tripulado: Penguin BE* (only in Portuguese)

- Jongseok Lee (ETH Zurich, master's thesis): *High Fidelity Modelling for High Altitude Long Endurance Solar Powered Aircraft*
- Andre Fialho Coelho (IFF Campos, bachelor's thesis): *System Identification and Parameter Space Control Design for a Small Unmanned Aircraft*
- Cristina Ruiz Páez (UCA Cadiz, bachelor's thesis): *Aerodynamic Simulation of Electrical High-Altitude-Long-Endurance Platforms*
- Wilson José de Sá Marques (IFF Campos, bachelor's thesis): *Analysis and Design of a Cooperative Controller for Landing Fixed-Wing UAV on Mobile Platforms*
- Marcus Felipe Fracaroli Pavani (TU Munich, master's thesis): *Implementation and Analysis of a Model Predictive Controller for Landing Fixed-Wing Aircraft on Mobile Platforms*
- Henrique Patusco Gomes da Silva (UFRJ Rio de Janeiro, bachelor's thesis): *Modelagem, Simulação e Controle de uma Aeronave Não Tripulada de Asa Fixa* (only in Portuguese)
- Arti Kalra (ISAE SUPAERO Toulouse, master's thesis): *System Identification and Control Design for a Stratospheric Flight Mission of a Solar Electric Aircraft*

Supervised Interns and Working Students

The following interns and working students have been supervised during the course of the thesis (in chronological order starting with the first student).

- Andre Fialho Coelho (IFF Campos, undergraduate student)
- Maria Kozlova (TU Munich, undergraduate student)
- Luiz Benicio Degli Esposte Rosa (IFF Campos, undergraduate student)
- Eduardo Della Noce (TU Munich, graduate student)

Media Response

The work presented in this thesis was partly funded by the European Commission FP7 ICT Program under the project EC-SAFEMOBIL 288082 (<http://www.ec-safemobil-project.eu>).

The results from the EC-Safemobil project (DLR Press release: https://www.dlr.de/content/en/articles/news/2016/20160118_video-autonomous-landing-at-full-speed_16413.html) have reached a high visibility (100+ press releases) in national and international media:

National:

- <https://www.heise.de/newsticker/meldung/DLR-Autonomer-Flieger-landet-auf-fahrendem-Pkw-3078140.html>
- <https://www.wired.de/collection/latest/mobile-landeflaechen-koennten-leichtere-drohnen-moeglich-machen>

- <http://www.welt.de/wissenschaft/article151178355/Diese-Drohne-landet-auf-einem-fahrenden-Auto-bei-75-km-h.html>

International:

1. <https://www.bbc.com/news/av/technology-35351709/drone-lands-on-moving-car-in-germany>
2. <https://www.wired.com/2016/01/there-may-actually-be-a-good-reason-to-teach-drones-to-land-on-cars/>
3. <http://www.wired.co.uk/news/archive/2016-01/20/drone-lands-on-moving-car>

Further noteworthy reports include:

1. Discovery Channel documentation (link only for USA/Canada, complete video available upon request): <https://www.discovery.ca/video?vid=961169>
2. DLR-Magazine: https://www.dlr.de/dlr/portaldata/1/resources/documents/DLR_Magazin_152/DLR_Magazin-151-152-GB/files/assets/basic-html/index.html#14-15
3. Podcast (in German): <http://www.thedronecast.de/2016/01/tdc-012-dlr-drohne-autonom-fahrendem-pkw/>

2 Cooperative Autoland System

Chapter Overview

Throughout this chapter the high-level system requirements and functional specification of the landing system are derived. The approach is based on classical systems engineering, but was reduced to the minimum scope necessary for the experimental development work shown in this thesis. This process will lead to the definition of the main functional building blocks, which in turn will be mapped to an appropriate landing system architecture in Chapter 4.

2.1 System Requirements

For the definition of functional system requirements it is common practice to start from a high-level perspective and to decompose the system functions in a top-down approach. The main function of the landing system obviously consists in the autonomous and safe landing of a given aircraft with no landing gear on top of a mobile landing platform. This scenario thus represents a dual-agent cooperative control problem, where the two involved vehicles need to be synchronized in position and velocity before a touchdown can take place. It is also desirable to execute the landing maneuver in an efficient and optimized way to minimize the runway used or the time to land. It seems intuitive that through active cooperation of both vehicles such goal should be easier to achieve. It will therefore be required that the landing maneuver is cooperative in nature. The determination of the best suited cooperation strategy constitutes the main research question of this thesis. Different evaluation criteria must be taken into account, including practical and safety aspects, vehicle dynamics, as well as different performance metrics (e.g. runway used). Optimality in this case must be defined as a mix of such criteria.

Table 2.1 provides the high-level functional requirements for the envisioned cooperative landing system.

Table 2.1 Landing system high-level functional requirements.

No.	Requirement Description
S-R1	The landing maneuver shall be autonomous and cooperative.
S-R2	The landing maneuver shall be safe and robust to disturbances.
S-R3	The landing maneuver shall be in line with established aviation procedures.
S-R4	The landing maneuver shall be efficient in terms of used runway.
S-R5	The landing maneuver shall be optimal in terms of safety, practicality and performance.
S-R6	The landing maneuver shall take place within the limits of the landing strip.
S-R7	The landing system shall be robust to communication delays.
S-R8	A fail-safe system shall be in place to cope with unforeseen events. System failures shall be automatically detected, identified and mitigated.
S-R9	An emergency flight termination system shall be available. The activation shall be automatic in case of sensor faults or geofence violations.
S-R10	The landing system shall be operational for horizontal wind velocities of up to $5m/s$.
S-R11	The involved vehicles shall be able to control their translational degrees of freedom independently to correct for disturbances from individual directions.
S-R12	The individual vehicle controllers shall be able to track given higher-level control command from the landing control system.
S-R13	The relative state estimation system shall be precise and reliable. The estimates shall be based on redundant sensor information from different sensors.
S-R14	The aircraft shall be prevented from departing the landing platform after touchdown by means of mechanical locking supported by control action.

In the end, the developed experimental landing system will be validated against these high-level requirements. As previously mentioned, the next step consists in identifying the main functional blocks, based on a priori knowledge of related technical solutions, available system components (such as classical flight controllers and sensors), and established aviation procedures for approach and landing. The examination of existing procedures for manual and automatic landing of aircraft was chosen as starting point for the functional analysis of the envisioned cooperative landing system. A description of common landing techniques was presented in the introduction in Section 1.1, where a possible improvement

was identified and described in Section 1.2. The following section will provide a more detailed look at the procedural implementation of the cooperative landing maneuver and highlight important control system relevant aspects.

2.2 The Landing Procedure

Before going into technical details, a higher-level overview of the envisioned landing procedure will be given. The following scenario shall be considered — the aircraft is flying a standard airport traffic pattern after having returned from a given mission. It is now on standby, waiting for the human ground operator to issue the landing command. At the same time a ground vehicle with a roof-mounted landing platform is standing at the approach end of the runway, ready for acceleration once the landing command is given.

In the following, the focus is on the flight phases related to the actual landing maneuver (once the aircraft starts to descend from the traffic pattern altitude), namely the base leg, final approach, flare, touchdown and landing-roll phases (as previously defined in Figure 1.2). The terminology has been slightly changed, to better adapt to the cooperative nature of the new approach.

The cooperative landing maneuver has been divided into six distinct phases, as illustrated in Figure 2.1:

1. Waypoint tracking
2. Initial descent
3. Final approach
4. Flare and touchdown
5. Ground lock and deceleration
6. Full stop

Phases 1 and 2 guide the aircraft to a pre-defined start point about 300m behind and 20m above the ground vehicle (aircraft and ground vehicle will also be referred to as UAV and UGV in the following), which is waiting at its start/idle position at the runway threshold. The cooperative part of the landing maneuver consists of phases 3 and 4, where the UGV starts accelerating and then moves synchronously with the UAV, until the aircraft touches down on the landing platform. In phases 5 and 6 the UGV is the main actor, decelerating the joined vehicles and safely terminating the maneuver. In these phases, the UAV's control surfaces are used to assist in keeping the UAV safely locked down on the landing platform.

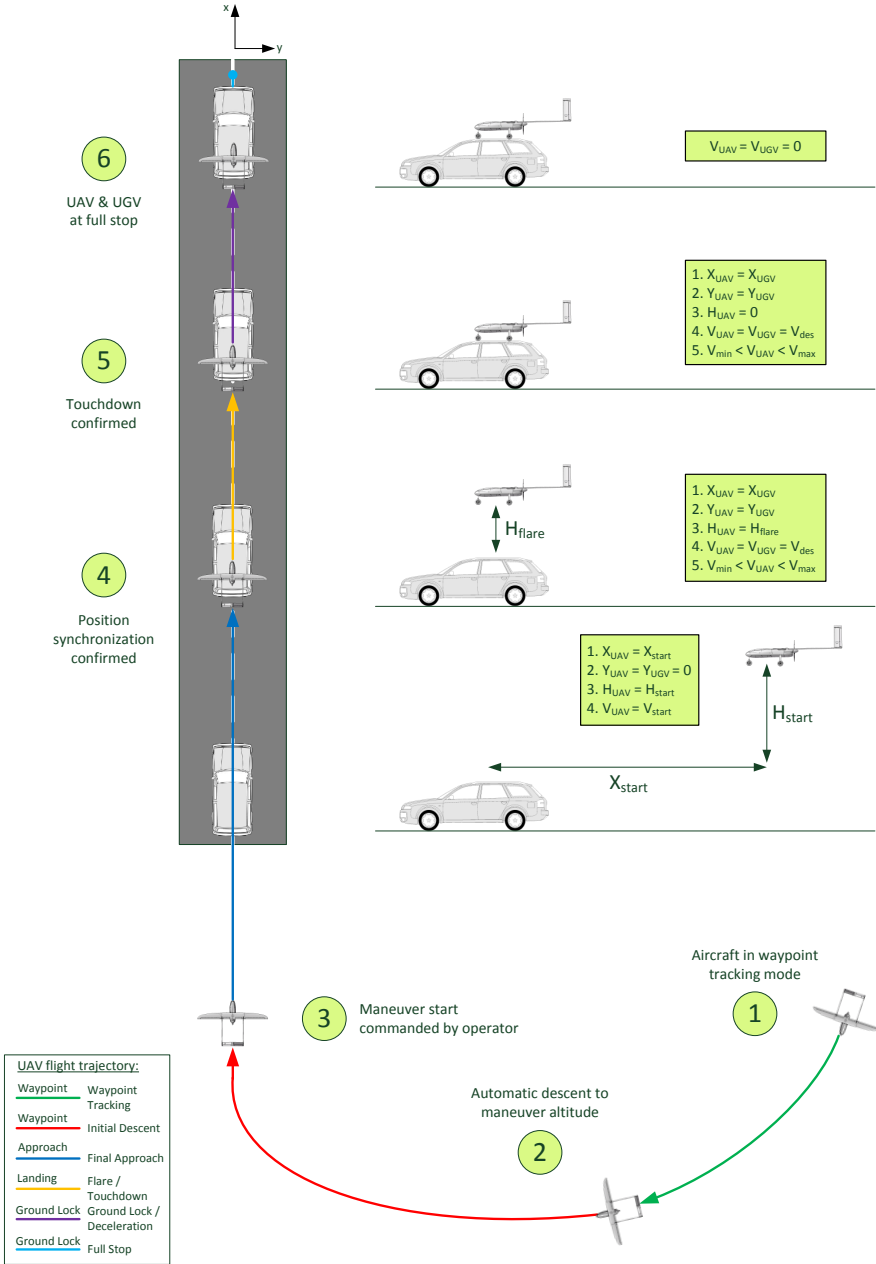


Figure 2.1 The different phases of the cooperative landing maneuver.

It is clear that certain phases of the landing maneuver are cooperative while others are not. The overall landing control system will therefore consist of a mix of different controller modes, where the transitions between the different landing phases are managed via a state machine logic. From a control perspective, the cooperative part of the maneuver can be divided into two main phases — the approach phase (phase 3) and the landing phase (phase 4), both of which are cooperative. The main difference consists in the control objective, where the approach can be interpreted as reference tracking control, while the landing requires position regulation.

The main task during the approach phase consists in the optimal guidance (optimality will be defined later on) of each vehicle towards a synchronized motion condition (*position acquire*). The main task during the landing phase consists in the precise tracking of the aligned vehicle positions despite disturbances, while the final flare maneuver is completed (*position hold/tracking*). Shortly before touchdown the engine is retarded and the so-called *ground lock* phase is entered to quickly transition the UAV from sustained flight condition to minimum lift configuration for a safe and efficient touchdown. Ground lock configuration is maintained after touchdown to avoid accidental lift generation (via nose-down elevator) and to provide pitch damping and braking assistance (important for larger aircraft) during the deceleration phase. The UAV remains in this configuration until a full-stop condition is reached.

Before the approach phase, the individual vehicle controllers are working independently (waypoint tracking for UAV and standby for the UGV), whereas after the landing phase (after touchdown), with the UAV locked onto the landing platform, the UGV performs the deceleration procedure (with possible UAV assistance for roll-out stabilization and drag augmentation).

The transitions between the different maneuver phases must be automatically managed, based on some previously defined conditions (e.g. relative vehicle states, operator commands, communication delay). Also, a failsafe mechanism is required for safe landing system operations — if the UAV for whatever reason significantly deviates from its expected behavior (e.g. large position error due to a strong gust, large communication delay), or if the operator gives an abort command, both vehicles must be commanded to a secure state. In such case a go-around command is automatically issued by the fail-safe system, and the aircraft transitions to flying pre-defined aerodrome circuits, while the UGV is commanded to a full stop. Figure 2.2 illustrates the different phase transition and specifies the switching conditions.

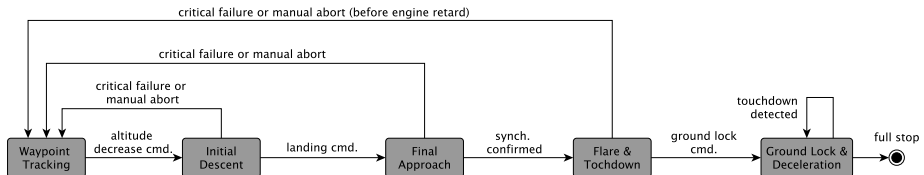


Figure 2.2 Switching conditions for landing maneuver phase transitions.

The main challenge associated with this new landing procedure consists in the cooperative motion control of the vehicles, both of which are complex dynamical systems, while ensuring a safe and efficient execution of the landing maneuver. On the aircraft side, particularly phases 2 to 4 require precise and independent control of all degrees of freedom. For each of the high-level cooperative maneuver phases, a further decomposition into horizontal and vertical controller functions can be made, as will be explained in the following section, where the basic landing system functions are identified and a functional breakdown is developed.

2.3 Functional Decomposition

The classical approach for developing complex systems, as applied in systems engineering, starts with the definition of systems requirements, followed by the derivation of high-level functions, which are then broken down further as required.

As previously mentioned, the definition of these high-level functions is based on a priori knowledge of existing landing procedures, available system components (e.g. sensors, flight controllers, airframes), related applications (see Chapter 1.4), and on engineering judgement. The outcome of such procedure is the definition of all basic system functions, which are required for the operation of the system. Once available, a system specification can be developed, where the different functional blocks are mapped to an appropriate system architecture. In Section 2.1 the high-level functional requirements for the proposed cooperative landing system were provided, while Section 2.2 provided a first decomposition of the landing maneuver into different phases. Throughout this section the functional breakdown for the cooperative landing system will be developed.

At first, the problem at hand is examined from a high-level perspective — for a successful landing maneuver, the two involved vehicles need to be controlled in a cooperative manner from waypoint flight to a full stop on the ground. This transition has already been divided into six different phases. The different phases partly require cooperation between the involved vehicles, and partly don't. The further analysis will therefore be elaborated by considering each vehicle's contribution to the respective maneuver phase.

Waypoint Tracking

The waypoint tracking phase of the landing maneuver quite obviously does not require any cooperation between the two vehicles. The functions required for this phase must therefore be provided by the aircraft alone. The aircraft must be able to follow a route of pre-defined waypoints and must thus possess a waypoint guidance function. During waypoint tracking, the aircraft must be able to maintain and to transition between steady state (unaccelerated) flight conditions. This implies controllability of airspeed, altitude and course angle. This functionality is covered by classical autopilots and is available since the late 1940's [110]. The set of required sub-functions for the UAV during the waypoint tracking phase of the landing maneuver therefore consists of the following functions: *altitude hold*, *speed hold*,

course hold, and *waypoint guidance*. No particular functionality is required of the UGV during this phase. The UGV will remain on *standby* during this phase.

Initial Descent

During the initial descent phase, the aircraft must fly towards the maneuver start point (point 3 in Figure 2.1) while decreasing its altitude from traffic pattern altitude to maneuver start altitude. During the descent, flight path angle and velocity are adjusted as necessary (increased velocity might be required to allow for larger glide angles). No cooperation with the ground vehicle is required during this phase. The set of required sub-functions for the UAV during this phase consequently consists of the following functions: *altitude tracking*, *speed tracking*, and *waypoint guidance*. Again, no special functionality is required of the UGV during this phase, besides the *standby* function.

Final Approach

The final approach phase marks the beginning of the cooperative part of the landing maneuver. The aircraft has reached the maneuver start point in the desired state (position, velocity, attitude) and the ground vehicle initiates its acceleration process. Both vehicles must be able to control the same physical quantities to allow for a precise synchronization. The obvious choice are inertial positions and velocities, relating the motion of each vehicle to the same frame of reference. Another reason for this requirement arises from the so-called *ground vehicle induced disturbance* effect. As the aircraft approaches the ground vehicle, it will enter a disturbed flow region caused by the ground vehicle body itself. The flow over the ground vehicle will result in an increase of flow angle and relative velocity (relative to the undisturbed flow), above the windshield and a decrease above the rear part of the vehicle. This effect acts similar to a wind shear and needs to be compensated for. If the aircraft simply entered the disturbed area while controlling airspeed, it would actively contribute to a mismatch in position alignment. A sudden increase in airspeed would be treated as a control error by the airspeed controller and the aircraft would try to reduce its airspeed by decelerating. Such deceleration is counterproductive for the position synchronization task, where the aircraft should rather increase its airspeed to maintain synchronization. Such behavior can be mitigated by the use of ground speed control, which would actively compensate for the ground vehicle induced disturbance. A detailed analysis of these effects will be presented in Chapter 3.3.

Both vehicles must therefore be able to control ground speed and course angle, which in the case of the aircraft is not an obvious choice, as will be explained in detail in Chapter 4.2.2. The required functionality is provided by the *ground speed tracking* function in case of the UAV, and *speed tracking* for the UGV. Both vehicles require a *course tracking* function.

The vehicles must now synchronize their inertial position and velocity (not airspeed) vectors in the horizontal plane, while the aircraft descends towards the flare initiation altitude. This process represents a cooperative control task and therefore requires the definition of a common functionality on system level. This function will be defined as *position acquire*. During this synchronization maneuver, the aircraft must be able to track a

variable glideslope. The *glideslope tracking* function must be independent of the horizontal position synchronization action to avoid unwanted coupling effects. Intentional coupling might be desired during the final approach, as will be explained in Chapter 4.3.1. As will be explained in detail in 2.4.1, such capability is not directly available for fixed-wing aircraft, since control responses in velocity and altitude are inherently coupled, and requires a dedicated solution. Additionally, both vehicles must be able to track a given course command during the synchronization task to allow for driving the coupled system back towards a course parallel to the runway after a disturbance. This task has a lower priority than the synchronization task and must not interfere with the execution of the primary task. Also, if desired, runway centerline tracking shall be possible. This functionality is included in the position acquire function.

Flare and Touchdown

At the end of the final approach phase horizontal position synchronization is achieved and the aircraft is flying directly above the landing platform (point 4 in Figure 2.1) at the recommended landing airspeed. The aircraft now needs to gradually reduce its vertical velocity before touchdown, which is achieved by means of a flare maneuver. It is mandatory that position synchronization is maintained during the flare maneuver, which again requires independent controllability of the horizontal and vertical degrees of freedom. A system level function is thus defined to maintain position synchronization and to reject external disturbances. This function is called *position synchronization*. During the flare maneuver, the aircraft slowly reduces the flight path angle while maintaining the required ground speed commanded by the position synchronization system. The aircraft is thus required to provide a *flare* function. At the same time, both vehicles must correct for disturbances from arbitrary directions and must therefore be able to independently control all translational degrees of freedom (within certain bounds).

At a certain altitude above the landing platform, the aircraft engine must be retarded (switched off) to avoid damage to the engine or the landing platform. Once the engine is switched off, either velocity or altitude can be controlled, since only one longitudinal control channel (elevator) is left. For this work the priority was given to velocity control, to maintain position synchronization, while vertical control is indirectly provided in a feed forward sense. A detailed description of the implemented control approach will be given in Chapter 4.2.5. The aircraft must therefore be able to control ground speed after engine retard, which will be provided by the *retard* function. No individual functionality is required from the UGV during this maneuver phase.

Ground Lock and Deceleration

Moments before touchdown, the aircraft is still in sustained flight condition while moving in very close proximity to the landing platform. To reduce the associated risk, the aircraft must be able to transition from sustained flight to minimum lift configuration (after touchdown) in a safe and reliable manner. After touchdown, the aircraft must be locked to the landing platform to prevent an unwanted release after first contact was made. These functionalities are provided by the *ground lock* function. After the touchdown the UGV must initiate

the deceleration of the joint vehicles while tracking the runway heading. If desired, the ground vehicle should be capable of guiding the roll-out maneuver back towards the runway centerline, in case touchdown took place with a lateral offset (e.g. due to a wind gust). This is provided by the *braking control* and *course tracking* functions. During the whole duration of the landing roll, the aircraft must provide pitch oscillation damping, and if applicable, active braking assistance. The deceleration process could be shortened by means of appropriate control surface deflection or via spoilers. This functionality is included in the ground lock function.

Full Stop

After the deceleration has been completed, both vehicles must go to a secure state to prevent an unintentional activation of the aircraft or ground vehicle engines. The joint vehicles are then on standby, waiting for a taxiing command issued either by the operator or some automated ground operation system. This is provided by the *secure state* function.

Relative State Estimation

For any of the above described cooperative control tasks to be possible, a reliable and precise relative state estimation system is required. As already mentioned in Table 2.1, the state estimates should be based on redundant sensor information. The relative state estimation system will be described in detail in Chapter 4. For now it will be assumed that the required relative state estimates are available.

High-Level Mission Control and Fail-Safe System

As previously mentioned, the transitions between the different maneuver phases and associated controller modes must be automatically managed. Therefore the pre-defined trigger conditions (e.g. based on relative vehicle state, operator commands, communication delay) must be monitored and the corresponding switching actions executed. This functionality will be provided by the *high-level mission control system*, which will be described in detail in Chapter 4.2.6. At the same time the system must be monitored and systems failures must be automatically detected, identified and mitigated. This functionality must be provided by the *fail-safe system*.

Functional Breakdown Structure

Table 2.2 and Table 2.3 summarize the above defined system functions and classify them into cooperative, non-cooperative and assistive functions. The non-cooperative functions must be provided by the respective vehicle controllers, while the cooperative and assistive functions are provided by the landing control system. The assignment of functions to distinct controllers constitutes the next step towards the system specification. The non-cooperative UAV functions will be provided by the *aircraft flight control system*, while the non-cooperative UGV functions will be provided by the *ground vehicle control system*.

The cooperative control functions related to maneuver phase 3 (final approach) will be associated to the *approach control system*, while the functions related to phases 4 and 5 will be provided by the *landing control system*. Figure 2.3 illustrates the assignment of the different control functions to the high-level controllers, where the respective controller’s activation time is indicated with respect to the different maneuver phases, as well as controller switching conditions and fail-safe behavior.

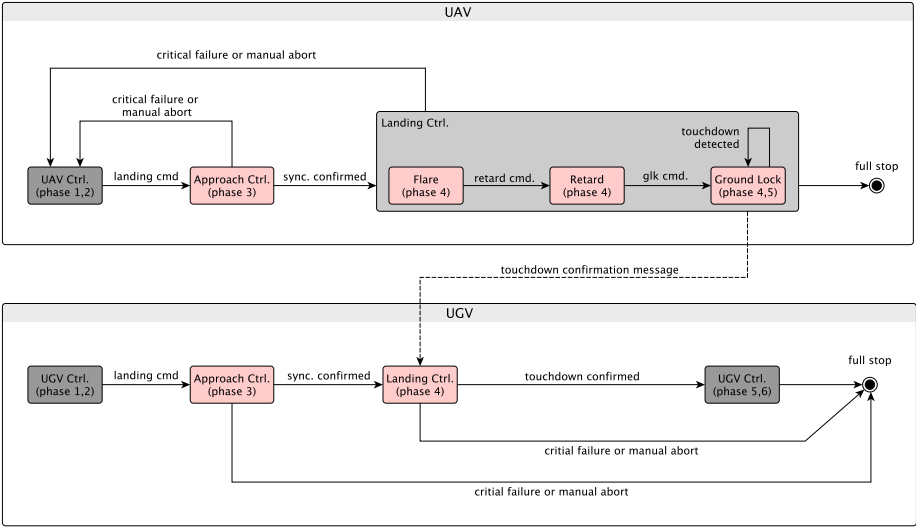


Figure 2.3 Definition of landing controller sub-systems. Gray blocks indicate non-cooperative functions while red boxes indicate cooperative functions.

While most functional requirements for the above mentioned control systems were mentioned throughout this chapter, they shall be formalized and further developed in the following sections. Based on these sub-system requirements the appropriate architecture for each controller can be chosen and implemented on the real system.

2.4 System Design

Throughout the previous section the required landing system functions have been identified, and associated to a number of different sub-systems. The corresponding functional requirements were thereby only implicitly provided. These requirements will now be formalized as sub-system functional requirements, based on which an appropriate system architecture can be developed for each system.

2.4.1 Aircraft Flight Control

Table 2.4 contains a list of aircraft functional requirements which was derived from the previously defined system functions.

Table 2.2 High-level controller functions classified by maneuver phase and vehicle - [c] indicates a cooperative function, [nc] indicated a non-cooperative function, and [a] indicates an assistive function.

No.	Maneuver phase	UAV function	UGV function
1	waypoint tracking [nc]	waypoint guidance [nc] altitude hold [nc] speed hold [nc]	standby [nc]
2	initial descent [nc]	waypoint guidance [nc] altitude tracking [nc] speed tracking [nc]	standby [nc]
3	final approach [c]	position acquire [c] glideslope tracking [nc] ground speed track. [nc] course tracking [nc]	position acquire [c] speed tracking [nc] course tracking [nc]
4	flare and touchdown [c]	position sync. [c] ground speed track. [nc] flare [c] retard [c] ground lock [c]	position sync. [c]
5	ground lock and deceleration [a]	ground lock [a]	braking control [nc] course tracking [nc]
6	full stop [nc]	secure state [nc]	secure state [nc]

Aircraft flight control requires the precise adjustment of aerodynamic forces and moments, which must be used together with propulsion forces and moments to control the motion of the aircraft with respect to the surrounding air mass. The most important aerodynamic parameters include airspeed, angle of attack, and sideslip angle, and are controlled via the different available control surfaces (typically elevator, aileron, and rudder), and the engine thrust. In a steady flight, all external forces (aerodynamics, propulsion, gravity) acting on the aircraft must be in equilibrium, meaning a flight condition where the aircraft's linear and angular velocity vectors are constant. The functional requirements *UAV-R1*, *UAV-R2*, and *UAV-R3* are related to the aircraft's ability to maintain such steady state flight conditions and to transition between them. This functionality is covered by classical autopilots and is available since the late 1940's [110]. Most classical flight controllers [13, 110, 10] would be sufficient to meet this requirement.

UAV-R4 can be solved in different levels of sophistication [38]. The optimal solution will also depend on its integration with the rest of the flight control functions.

Table 2.3 High-level controller functions classified by level of cooperation and vehicle.

Vehicle	Non-Cooperative	Cooperative	Assistive
UAV	waypoint guidance altitude hold speed hold altitude tracking speed tracking glideslope tracking ground speed tracking course tracking secure state	position acquire position sync. flare retard ground lock	ground lock
UGV	standby speed tracking course tracking braking control secure state	position acquire position sync.	-

UAV-R5 is not a common function in aircraft control systems. Ground speed control of course is possible, but care must be taken when integrating it with the airspeed based envelope protection controller.

UAV-R6 requires a more detailed investigation. A decoupling of the aircraft's translational motion (into longitudinal, lateral and vertical directions) with respect to a given inertial frame can be achieved through appropriate control command design. Thereby the short term dynamics (transient behavior) must be taken into consideration, since aircraft dynamics typically include inherent couplings. Especially the decoupling of altitude and speed control poses a bigger problem, since both elevator and throttle produce coupled responses. Coordinated control inputs of elevator and throttle, are required to achieve this goal. An energy-based control approach [74] was chosen, as it was known to provide the needed functionality. In the lateral plane such problem is less pronounced, and a classical control approach was selected here [46]. Chapter 3.2 will provide a detailed explanation of the implemented solution.

During thrust limit operations (full or idle throttle) only the elevator is left to control either altitude or speed. *UAV-R7* requires that speed control is prioritized, which can be achieved through appropriate mode switching at throttle saturation. The reason for this requirement arises from the fact that airspeed constitutes a safety-critical parameter (for lift force generation) and must at all cost be maintained above its minimum value to allow for continued safe flight. It should be noted that indirect control of flight path angle still remains possible, which is due to the natural correlation between flight path angle and airspeed during gliding flight. For a given controlled airspeed there exists only one flight

Table 2.4 Aircraft flight control system functional requirements.

No.	Requirement Description
UAV-R1	The aircraft shall be capable of controlling its airspeed and flight path (flight path angle and course angle)
UAV-R2	The aircraft shall be able to follow a three dimensional waypoint route.
UAV-R3	The aircraft shall possess a flare control system.
UAV-R4	The aircraft shall possess a flight envelope protection system.
UAV-R5	The aircraft shall be capable of controlling ground speed (inertial velocity), while providing continued flight envelope protection.
UAV-R6	The aircraft shall be capable of performing decoupled flight maneuvers in all translational degrees of freedom (longitudinal, lateral and vertical).
UAV-R7	The aircraft shall be capable of controlling velocity even at thrust limits (idle or full thrust), at the expense of glide path control.
UAV-R8	The aircraft shall be capable of performing a fast landing transition (sustained flight to minimum lift configuration after touchdown).
UAV-R9	The aircraft shall assist the ground lock and deceleration phase where suitable (minimum lift, oscillation damping, braking assistance).
UAV-R10	The aircraft shall possess a mechanical locking device to attach itself to the landing platform after touchdown.

path angle which leads to an unaccelerated gliding flight condition (force equilibrium). This correlation can be used during the initial descent phase of the landing maneuver to cover a large altitude difference during a relatively short base leg turn. Therefore the commanded airspeed can be increased, thus allowing for a steeper descent.

UAV-R8 requires that the aircraft transitions quickly from sustained flight to minimum-lift condition. This can be achieved through appropriate control commands, using the available control surfaces, flaps or spoilers. An electric fly-by-wire control system would be beneficial, allowing for arbitrary control surface deflections (e.g. symmetric aileron-up). *UAV-R9* requires that the aircraft provides assistive control and rate damping action after touchdown. Also, if applicable (e.g. for larger aircraft) the available control inputs should be used for active braking assistance.

UAV-R10 requires the design of an appropriate locking device and poses no problem.

Different flight control strategies would be conceivable to meet these aircraft requirements (e.g. PID-control, model-based control, optimal and robust control). In the end, the choice was made in favor of an energy-based MIMO-control approach for the longitudinal motion and a classical cascaded SISO-controller for the lateral motion. This choice is mainly motivated by the physical interpretability of the different controller gains and

the in-flight tuning capability, which favors an iterative development process. Also, the required model specific data is reduced to a minimum and the control architecture is easily transferable to other aircraft systems. Chapter 3.2 will provide a detailed analysis and implementation of the developed flight controller.

2.4.2 Ground Vehicle Control

Table 2.5 contains the functional requirements for the ground vehicle.

Table 2.5 Ground vehicle control system functional requirements.

No.	Requirement Description
UGV-R1	The ground vehicle shall be capable of controlling ground speed and course angle
UGV-R2	The ground vehicle shall be capable of performing decoupled maneuvers in the horizontal plane (in longitudinal and lateral direction)
UGV-R3	The ground vehicle shall possess an appropriate landing platform and locking mechanism to secure the UAV after touchdown.
UGV-R4	The ground vehicle shall be capable of decelerating the joint vehicles to a full stop condition.

UGV-R1 can be provided by either a human driver and an appropriate command interface or by an autonomous vehicle by means of cascaded SISO controllers (PID based). *UGV-R2* can be met by a vehicle with holonomic motion capabilities like the DLR Robomobil¹ or by a classical non-holonomic vehicle. In case of non-holonomic vehicles, a decoupling can be achieved though appropriate control command design and becomes easier at higher velocities, where smaller heading changes are required to produce identical lateral accelerations. Higher velocities (around $75 \frac{km}{h}$) can be expected for the envisioned landing maneuver, given the recommended aircraft landing speed. *UGV-R3* requires specific adaptation to a given experimental setup. In this work the focus was on the analysis and development of a cooperative motion control strategy for a general class of aircraft and ground vehicle systems. The actual demonstrator setup is limited to a rather small aircraft, where a complex, possibly actuated landing platform design is not necessary. The landing platform was therefore designed as a flat surface with a tensioned net to catch the aircraft. For larger and heavier aircraft the specific solution used here is not directly applicable anyway and need to be reworked. The landing platform design therefore developed specifically for the experimental setup presented in Chapter 1.5.2. *UGV-R4* requires that the touchdown is communicated to, or directly detected by the ground vehicle. This functionality is provided by the relative state estimation system, which will be described in detail in Chapter 4.1. Due to the required effort for developing an autonomous vehicle large

¹ROBoMobil - The Robotic Electric Vehicle (<https://www.dlr.de/sr/en/desktopdefault.aspx/tabid-11633/>)

enough to carry the weight of the landing platform and aircraft (100kg for demonstrator setup) the choice in this work was taken in favor for a manned semi-autonomous ground vehicle setup, where a human driver is provided with control commands via a visual interface mounted in the cockpit. A detailed description of the ground vehicle control concept will be given in Chapter 2.4.2.

2.4.3 Relative State Estimation

Table 2.6 contains the functional requirements for the relative state estimation system.

Table 2.6 Relative estimation system functional requirements.

No.	Requirement Description
RSE-R1	The relative state estimation system shall be capable of providing precise and reliable state relative estimates in position, velocity and attitude angles.
RSE-R2	The relative state estimation system shall use redundant sensor information to calculate the relative state estimates.

The relative state estimation systems needs to use sensor data from redundant sources and needs to combine it to produce more reliable results. In this work a combination of GPS and optical camera data will be combined. The camera will be mounted on the aircraft and use a set of optical markers on the landing platform to calculate vision-based relative state estimates. The GPS data will be exchanged between the vehicles through a communication network, while time delays will be take into consideration. These two sources will be combined by means of a sensor fusion algorithm based linear Kalman filtering. A detailed description of the relative state estimation system will be provided in Chapter 4.1.

2.4.4 Approach Control

Table 2.7 contains the functional requirements for the approach control system.

The approach control system will be responsible for guiding to vehicles towards horizontal synchronization, while reducing the altitude of the aircraft. The aircraft is limited in its capability to decrease altitude by its gliding properties. During gliding flight, airspeed and glide angle are naturally correlated and a variation of flight path angle is only possible by varying the airspeed or by setting spoilers (which are not available on the demonstrator aircraft). Since it is desirable to perform the landing at the lower end of the safe flight envelope (at low airspeeds) it is important to maintain a certain margin to the minimum flight path angle (gliding flight). This way the engine remains above idle position and corrections in flight path angle can be performed without compromising speed control. For that reason, the flight path angle is typically limited to rather low values (typically -3°) during a standard approach [113]. The descent of the aircraft from the maneuver

Table 2.7 Approach control system functional requirements.

No.	Requirement Description
AC-R1	The approach control system shall calculate optimal flight- and ground trajectories for the final approach phase of the landing maneuver. Optimality shall be defined by taking into account safety and operational constraints, vehicle dynamics, and runway dimensions.
AC-R2	The approach control system shall calculate appropriate input commands for the respective vehicle controllers, taking into account vehicle performance limitations.
AC-R3	The approach control system shall be capable of rejecting external disturbances and of guiding the vehicles back to the runway centerline if necessary.

start point at the beginning of the final approach phase (point 3 in Figure 2.1) to the flare initiation altitude (point 4 in Figure 2.1) therefore takes a certain amount of time. The same holds for the ground vehicle, which needs to accelerate to match the aircraft's speed. Such acceleration process will typically be performed around the UGV's maximum acceleration to use as little runway as possible, but also requires a certain amount of time. These time slots now need to be placed in way, such that the aircraft arrives at the pre-defined flare initiation altitude at the exact moment, when horizontal synchronization is achieved. Only then would the available runway length be ideally used. The coordination of such well timed maneuver, under the consideration of aircraft and ground vehicle dynamic constraints and the influence of external disturbances, requires some kind of alignment prediction. Such prediction must be based on model specific data, like aircraft and ground vehicle performance data (e.g. minimum UAV flight path angle, maximum UGV acceleration), but also on momentary vehicle states, since external disturbances (e.g. wind) might lead to a deviation from the planned paths. Therefore *AC-R1* and *AC-R2* require the solution of an online alignment prediction during the landing maneuver. The obvious choice for such constrained path planning task would be Model Predictive Control (MPC), since it inherently allows for the calculation of optimized vehicle trajectories while explicitly considering system limitations. Since the solution of a constrained optimization problem usually requires considerable time, it is desirable to keep the model complexity as low as possible, to allow for a faster calculation. As previously described, the overall landing control task was separated into the two sub-tasks *approach control* and *landing control*, which have different control objectives (reference tracking vs. position regulation). Since the MPC would only be used during the final approach phase, and thus not during close proximity operations, a reduced model precision would be acceptable. An integrated MPC-based solution will thus be developed for the approach control phase of the landing maneuver, as will be described in Chapter 4.3.1.

Even though the model complexity will be kept as low as possible, the solution of an MPC problem as part of the overall landing system increases the system complexity and

does not allow for direct physical interpretability and thus heuristic fine tuning during field experiments. A second problem concerns the safety and certifiability of the developed landing system. The vast majority of current civil and military flight control systems still rely on gain scheduled PID-control and not on readily available and more efficient nonlinear control concepts. The perseverance of linear control methods is rooted in the high safety standards and strict certification procedures applied in the aviation sector. Consequently a second, *safety-based* approach controller will be developed alongside the *performance-based* MPC approach. The focus will thereby be on the adherence to established aviation procedures as well as the reduction of system complexity, in favor of a rapid certification and industrialization process. The safety-based approach strategy will be described in detail in Chapter 4.2.

According to standard approach procedures, the aircraft should be controlled along a constant -3° glideslope, at constant recommended landing airspeed. The recommended landing airspeed for a given aircraft weight is provided by the aircraft manufacturer and usually lies at around $1.3 \cdot V_{S0}$, where V_{S0} represents the stall speed in landing configuration [42]. At this low speed the aircraft is susceptible to wind gusts and the precise control of its landing airspeed is a high-priority task. It was therefore decided to develop the safety-based solution with a constant landing airspeed throughout the whole landing maneuver to minimize the associated risk and to adhere to standard landing procedures. The aircraft will therefore only contribute to the lateral vehicle alignment during the safety-based final approach maneuver, while the ground vehicle remains on the runway centerline while adjusting its speed to the aircraft.

It should be noted that a direct control of airspeed is not possible during the final approach of the cooperative landing maneuver, since ground speed control is required for precise vehicle synchronization (see Section 2.3). As the aircraft approaches the landing start point (point 3 in Figure 2.1) it switches from airspeed tracking to ground speed tracking. Even in ground speed tracking mode continued flight at the recommended landing airspeed is required. As ground speed corresponds to the sum of airspeed and wind speed, the ground speed command needs to be dynamically adapted to the current wind conditions (e.g. in the case of a constant headwind, the ground speed command would have to be reduced by the amount of wind speed to yield continuous airspeed tracking). It is thus evident that the appropriate landing ground speed command must be calculated as a function of the current wind conditions. A detailed description of the implemented ground speed tracking method will be presented in Chapter 4.2.2.

The previously described coordination problem of simultaneous aircraft and ground vehicle arrival at the flare initiation point (point 4 in Figure 2.1) can now be solved without the need for a computationally expensive optimization algorithm. Assuming constant aircraft ground speed and constant ground vehicle acceleration, an analytic expression for the required ground vehicle velocity command, only depending on relative vehicle positions, can be derived. Since the above assumptions will not perfectly hold during the final approach process, the prediction is evaluated online at a high rate. In this way, deviations from the assumed conditions are taken into account, and the computed ground vehicle velocity command is continuously updated. This function will be referred to as *linear acceleration control* and will be described in detail in Chapter 4.2.1.

For the safety-based approach, the previously defined system-level function *position*

acquire can be broken down further according to the above described design choices. On the UAV side the required sub-functions are *ground speed tracking (longitudinal)*, *UGV position tracking (lateral)*, and *glideslope tracking (vertical)*. On the UGV side the required sub-functions consist of *linear acceleration control (longitudinal)* and *course tracking (lateral)*. At this point the required UAV and UGV functions have been broken down to the different translational degrees of freedom, where a further decomposition is not necessary. Table 2.8 summarizes the newly defined low-level functions for the safety-based approach, classified by operating direction and vehicle.

Table 2.8 Low-level functions for final safety-based approach classified by operating direction and vehicle.

Operating Direction	UAV	UGV
longitudinal	ground speed tracking	linear acceleration control
lateral	UGV position tracking	course tracking
vertical	glideslope tracking	-

The performance-based approach will produce coordinated control commands in all translational degrees of freedom in an integrated manner, according to a pre-defined objective function and the corresponding optimization constraints. A detailed description of the optimization-based approach controller will be provided in Chapter 4.3.1. Table 2.9 summarizes the design choices for the final approach controller.

2.4.5 Landing Control

Table 2.10 contains the functional requirements for the landing control system.

The main task of the landing control system consists in providing precise horizontal vehicle synchronization (at the recommended landing airspeed) during the final descent of the aircraft. While the aircraft is descending, it must gradually decrease its vertical velocity to allow for a safe touchdown, while maintaining synchronization with the ground vehicle, flying at the recommended landing airspeed, and rejecting disturbances. *LC-R1* thus requires decoupled control of aircraft ground speed, flight path angle, and course angle. On the ground vehicle side, decoupled control of speed and course angle is required. As mentioned in Section 2.4.1, the decoupling of aircraft speed and flight path control will be provided directly by the aircraft flight controller, since it involves strong dynamical couplings and thus requires precisely coordinated control inputs at inner-loop level. Horizontal motion decoupling (with respect to an inertial reference frame, and not a vehicle body frame) will be provided through appropriate directional control command design for both vehicles.

The following scenario shall be considered — the aligned vehicles are moving in the direction of the runway, when the aircraft is pushed sideways due to a sudden wind

Table 2.9 Summary of design choices for the approach controller.

No.	Design choice description
1	Two different approach control strategies will be developed.
2	The first strategy will be referred to as safety-based approach and will be based on established landing procedures. The aircraft will be commanded to fly a constant glideslope at constant recommended landing airspeed (in ground speed tracking mode).
3	The aircraft will perform lateral position synchronization during the approach, the ground vehicle will remain on the runway centerline and adjust its speed to the aircraft.
4	The safety-based approach controller will require the functions <i>ground speed tracking (longitudinal)</i> , <i>UGV position acquire (lateral)</i> , and <i>glideslope tracking (vertical)</i> on the UAV side, while the functions <i>position acquire (longitudinal)</i> and <i>course tracking (lateral)</i> are used on the UGV side.
5	The second strategy will be referred to as performance-based approach and will rely on model predictive control to calculate optimized flight- and ground trajectories.
6	The performance-based approach will cover the functions <i>position acquire (3D)</i> on the UAV side and <i>position acquire</i> on the UGV side.

gust, leading to a lateral position deviation (vertical motion shall be disregarded at the moment). It is now desired to move both vehicles back towards lateral synchronization without affecting the still intact longitudinal synchronization (in runway direction) and speed. The only way to influence the lateral motion of the vehicles is by changing the commanded course angles. Since both aircraft and ground vehicle control the magnitude of their respective speed vector independently of course angle (and thus direction), a change in course angle with constant speed would lead to a rotation of the speed vector, and therefore to a coupling between lateral and longitudinal motion. This coupling could be compensated for by increasing the speed vector during lateral motion, such that its longitudinal component remains constant. This involves a simple transformation of the desired corrective velocity commands from cartesian to spherical coordinates, which directly match both vehicle's available control inputs

It should be noted that such coupling effect will typically generate small velocity deviations for the landing scenario, since only small course angle changes and rather high landing speeds are expected during a nominal landing maneuver (longitudinal speed decrease is proportional to the cosine of course angle). At a landing speed of $21 \frac{m}{s}$ (nominal landing speed for demonstrator setup) a course change of 5.5° would produce a lateral velocity component of around $2 \frac{m}{s}$ (which can be considered fast corrective motion) and would lead to a longitudinal velocity deviation of less than $0.1 \frac{m}{s}$. In the case of stronger

Table 2.10 Landing control system functional requirements.

No.	Requirement Description
LC-R1	The landing control system shall be capable of maintaining precise horizontal position synchronization between the involved vehicles, while the aircraft performs the flare, retard, and ground lock maneuvers.
LC-R2	The landing control system shall calculate appropriate input commands for the respective vehicle controllers.
LC-R3	The landing control system shall be capable of effectively rejecting external disturbances and of guiding the vehicles back to the runway centerline if necessary.
LC-R4	The landing control system shall be robust to communication time delays (due to data exchange between vehicles).
LC-R5	The landing control system shall coordinate the flare, engine retard and ground lock activation sequence.
LC-R6	The landing control system shall be capable of detecting a touchdown and safely securing the aircraft on the landing platform after touchdown.
LC-R7	The landing control system shall produce appropriate control commands for the ground vehicle during the deceleration phase.

head- or crosswinds, where the landing ground speed (in runway direction) would be significantly reduced, larger course angle changes would be required to produce the same lateral velocity. In such case, the coupling would be more pronounced and the above described correction might become necessary. As defined in Table 2.1, the maximum wind speed at which the landing system must remain operational is $5 \frac{m}{s}$. With respect the above described coupling problem, the headwind case represents the worst-case condition, since it leads to the maximum reduction of forward speed. A headwind of $5 \frac{m}{s}$ would lead to a landing ground speed of $16 \frac{m}{s}$, which would then require a course angle of around 7.2° (23% increase) to produce the same lateral corrective action of $2 \frac{m}{s}$. In the case of a $10 \frac{m}{s}$ headwind, the course angle would have to be around 11.5° (52% increase), which is more than double the value of the wind-free case. Of course different aircraft types will have different recommended landing speeds, where the directional coupling worsens with decreasing speed. In case of HAP aircraft systems, landing airspeeds can lie as low as $10 \frac{m}{s}$ or less, which makes them prone to the directional coupling effect. In case of very large and light aircraft, the efficiency of the maneuver might benefit from an increased corrective action from the ground vehicle, rather than the aircraft. It can be concluded that the control command correction is not mandatory for the demonstrator setup ($21 \frac{m}{s}$ landing airspeed) with maximum wind speeds of $5 \frac{m}{s}$, due to the small required variation in absolute course angle values (1.7°). At higher wind speeds or for slow flying aircraft

such correction might prove to be useful.

Additionally, the aircraft will have wind-independent lateral dynamics, in the sense that it will be commanded to fly at a constant airspeed (in ground tracking mode), independent of the resulting ground speed. Since the aerodynamic force and moment generation only depends on aerodynamic quantities (e.g. airspeed, angle of attack and sideslip angle) the generated forces and moments would be the same for any wind condition and would therefore be independent of the ground speed. The above described directional coupling still applies, but the required velocities are reached in the same time, independent of the current ground speed. A non-holonomic ground vehicle on the other hand would suffer from slower lateral dynamics at low ground speeds.

To understand the implications of *LC-R2*, it is necessary to consider the properties of the individual vehicle controllers. Classical aircraft flight control systems are typically designed as cascaded controllers, which fit well to the physical reaction (integrator) chains present in aircraft flight dynamics (e.g. four integration steps from elevator deflection to altitude change). Additionally, autopilots are often separated into longitudinal and lateral control, which is based on the assumption of decoupled flight dynamics (this coupling refers to the aircraft body frame and should not be confused with the previously described directional coupling in the inertial reference frame) [110]. For most fixed-wing aircraft such assumption is valid for flight conditions close to a trim state (force and moment equilibrium with constant velocity and angular velocity vectors), where smooth maneuvers with smaller variations in angle of attack, sideslip angle and angular rates are utilized. Such decoupling was also assumed for this work, since the whole landing maneuver will be performed in a smooth and continuous fashion, and with small variations in speed and flight path angle. Horizontal motion decoupling can therefore be considered as given in terms of both short term (dynamical decoupling) and long term (directional decoupling) control for both vehicles. Decoupling in the vertical plane concerns the aircraft only, and will be provided by the energy-based flight control concept introduced in Section 2.4.1. A detailed description of the implemented vehicle controllers will be given in 3.

The natural architecture of the aircraft flight controller (cascaded PID-control) was continued for the position synchronization controller, which is designed as an additional loop around the existing cascaded vehicle controllers (one per horizontal degree of freedom). Thereby a given relative position error in longitudinal direction is transformed into corrective speed commands (for each vehicle), while a relative position error in lateral direction is converted into corrective course commands. For the vertical direction a variable glideslope flare controller is implemented, allowing for appropriate command generation in terms of flight path angle. Such controller architecture reduces the overall system complexity by mapping a three dimensional MIMO problem to three one dimensional SISO controllers. Additionally, PID-control was chosen for its well-known robustness properties [83], its physical interpretability (allowing for efficient field experiments and heuristic fine tuning), its low computational demand (leading to high sampling rates), and its suitability for existing certification procedures. This results in a distributed feedback control architecture, with simultaneous control actions provided by each vehicle. With these design choices, *LC-R2* can be taken into consideration by designing the cascaded outer-loop position synchronization controller with sufficient bandwidth separation from

the inner-loop velocity and course controllers [10].

Horizontal Position Synchronization

The focus shall now be directed at the design of the horizontal position synchronization controller. The previously defined system-level function *position synchronization* consists of reciprocal position tracking tasks, which can also be formulated as a relative position regulation task. It can thus be broken down into the sub-functions *longitudinal position tracking* and *lateral position tracking* for each vehicle. This distributed synchronization task can now be solved at different levels of cooperation between the involved vehicles. Per vehicle there are four behavioral options (in the horizontal plane):

- The vehicle does not actively contribute to the landing maneuver at all (0).
- The vehicle only actively contributes in the longitudinal direction (X).
- The vehicle only actively contributes in the lateral direction (Y).
- The vehicle actively contributes in both longitudinal and lateral direction (XY).

Since two vehicles are involved in the landing maneuver this adds up to 16 possible combinations, which are shown in Table 2.11.

Table 2.11 Possible modes of cooperation for the position regulation phase of the landing maneuver: no cooperation (0), cooperation in x-direction (X), cooperation in y-direction (Y), cooperation in x and y-direction (XY). The x-axis points in runway direction, the y-axis to the right.

UAV / UGV			
0 / 0	X / 0	Y / 0	XY / 0
0 / X	X / X	Y / X	XY / X
0 / Y	X / Y	Y / Y	XY / Y
0 / XY	X / XY	Y / XY	XY / XY

The problem now consists in finding the optimal solution with respect to safety and performance, which directly corresponds to addressing requirements *LC-R3* and *LC-R4*. Safety in this case is not only related to the stability of the landing control system, but also to the system performance — certain performance bounds must be guaranteed to allow for a safe landing (e.g. touchdown accuracy).

Nevertheless, stability of the position synchronization controller shall be considered first. The control task consists in regulating the relative position errors between the vehicles to zero, while rejecting external disturbances and maintaining desired landing airspeed. The calculation of relative position errors is thereby based on relative state estimates, and thus requires data from both vehicles, which must be provided via a communication channel.

The inclusion of these relative estimates into the feedback loops might therefore induce significant time delay due to the computation time of the relative state estimates themselves and the communication lag. It is a well known fact that the introduction of time delay into an otherwise stable feedback system may cause instability [108]. The effect becomes even more distinct for distributed systems, where the delays are introduced multiple times. With the selected controller design representing a distributed (dual-agent) system, such destabilizing effects must be carefully analyzed and prevented.

It is therefore useful to first classify the different cooperative modes into unilateral and bilateral position tracking cases, where bilateral tracking means that both vehicles actively contribute to the same degree of freedom (e.g. X / X). This distinction is useful, because it allows for directly distinguishing naturally stable cases from potentially unstable ones.

The unilateral case basically represents a classical leader-follower setup, where the follower receives delayed position information from the leader, but no state feedback is sent back to the leader. This represents an open loop configuration which cannot lead to instability, for otherwise stable vehicle systems. As long as the leader is not influenced by the follower's control actions, stability is guaranteed. Also, if the delayed reference value converges, the follower will converge as well, even for varying time delays. It should be noted though that a delayed transmission of the respective vehicle's position information not only causes stability issues, but also induces a position error, which needs to be corrected. In unilateral mode, the follower vehicle basically receives obsolete position information from the leader without knowing it. Its position synchronization controller would successfully follow the delayed reference but in reality, the follower would be located behind the leader by a distance which depends on the current communication delay. Such effect will be accounted for by performing a so-called *lag compensation*, as will be explained in detail in Chapter 4.1.1.

The bilateral case seems promising in terms of performance, but such advantage would come with the drawback of vehicle and communication delay dependent stability and performance properties, which might limit the applicability of the developed landing system to a smaller range of vehicles. Nevertheless, certain vehicle types might benefit from bilateral position tracking control. A detailed analysis will be presented in Chapter 5.

For these reasons it was decided to develop two different landing controllers. The first solution implies a time delay robust synchronization controller in unilateral configuration, and will be referred to as *safety-based* landing controller. The second approach will be based on bilateral position tracking control and will be referred to as *performance-based* landing controller.

The decision to develop two different landing control strategies, was based on system stability aspects. The next step now consists in finding the optimal solution for each strategy in terms of performance and safety, and to further develop the landing control system design.

Table 2.12 shows the different cooperative modes from Table 2.11 classified into unilateral and bilateral modes.

First the safety-based landing controller (unilateral combinations) shall be analyzed. Out of the 9 unilateral modes marked (*) in Table 2.12, the 5 cases including a 0 and a single letter (X or Y) were discarded, since a successful landing requires control of both

Table 2.12 Possible modes of cooperation for the position regulation phase of the landing maneuver classified into unilateral (*) and bilateral (-) modes: no cooperation (0), cooperation in x-direction (X), cooperation in y-direction (Y), cooperation in x and y-direction (XY). The x-axis points in runway direction, the y-axis to the right.

UAV / UGV			
0 / 0 *	X / 0 *	Y / 0 *	XY / 0 *
0 / X *	X / X	Y / X *	XY / X
0 / Y *	X / Y *	Y / Y	XY / Y
0 / XY *	X / XY	Y / XY	XY / XY

horizontal degrees of freedom. Table 2.13 shows the four remaining meaningful unilateral cooperative modes. In the following all four modes will be analyzed in more detail.

Table 2.13 Possible unilateral modes of cooperation for the safety-based landing controller: no cooperation (0), cooperation in x-direction (X), cooperation in y-direction (Y), cooperation in x and y-direction (XY). The x-axis points in runway direction, the y-axis to the right.

UAV / UGV
0 / XY
X / Y
Y / X
XY / 0

In mode (XY / 0), the UAV would follow the UGV in both longitudinal and lateral direction, while the UGV would be commanded to drive at a constant velocity, following the runway centerline. This mode can be compared to a missile guidance task, where only one vehicle needs to follow another, which is not cooperative. For the landing application, this mode can be discarded, since it would rely on the UAV solely to achieve position synchronization, while not making use of the fact that the UGV is hardly affected by atmospheric disturbances. Also, it would not make use of the involved vehicle’s dynamic properties.

The mode (X / Y) would rely on the UAV to provide longitudinal synchronization, while the UGV would take care of the lateral direction, and vice versa for the (Y / X)

mode. In general it makes more sense for the ground vehicle to follow the aircraft's (ground) speed, since the aircraft is limited in its range of admissible airspeeds, while the ground vehicle has no such limitation. Also, the ground vehicle has better control over its longitudinal dynamics (active acceleration and braking), while many aircraft cannot actively brake (only by means of the drag force if no spoilers are available). This can lead to unbalanced longitudinal flight dynamics, making aircraft more prone to tailwind gusts than to headwind gusts. Additionally, the aircraft's lateral dynamics do not depend on its ground speed, but only on its airspeed. Since the aircraft is commanded to follow a constant landing airspeed throughout the flare to touchdown phases, it will have unchanged lateral dynamics independent of its current ground speed. It is therefore desirable to let the ground vehicle perform the longitudinal tracking task, while the aircraft performs the lateral tracking.

There are a number of additional reasons in favor of this choice, which shall be provided as well. Even though the horizontal plane is being analyzed here, it should not be forgotten that the aircraft is performing its flare maneuver during the synchronization process. As previously explained, aircraft dynamics are naturally decoupled in longitudinal and lateral motion, but are strongly coupled in the vertical plane. A decoupling of longitudinal and vertical motion requires the use of an appropriate control approach, where perfect decoupling would require matching aircraft pitch and thrust dynamics, which for most aircraft implies significantly slowed pitch dynamics [74]. High altitude aircraft are especially prone to such retarded longitudinal dynamics, since they typically possess light and efficient propulsion systems and thus slow thrust dynamics. Also, it is very important to precisely track the desired flight path angle and landing airspeed during the flare maneuver, to allow for a safe touchdown at the correct sink rate and airspeed. This task is already hindered by the ground effect, and the ground vehicle induced disturbance and should therefore be kept free of additional control tasks. For these reasons the (Y / X) mode will be favored over the (X / Y) mode.

The (0 / XY) mode would rely on the ground vehicle to perform longitudinal and lateral synchronization, while the aircraft would be commanded to fly at constant recommended landing speed along the runway centerline. Such configuration is not cooperative but could be of interest for aircraft with very slow dynamics and ground vehicles with fast dynamics, as is the case for large commercial aircraft (see Chapter 1.3.1). It should be noted though that the lateral dynamics of a non-holonomic ground vehicle depend on its ground speed, which implies slower dynamics at lower speeds. This will lead to larger and more dynamic heading angle changes at lower velocities, which in turn could lead to a reduced docking accuracy. Also, the ground vehicle control concept should be considered — if the ground vehicle is not fully autonomous (as is the case for the present demonstrator setup) it can be difficult for a human driver to follow control commands in two directions simultaneously. In such case, the best solution would be to maintain runway centerline tracking, while performing the longitudinal position tracking task, as proposed in mode (Y, X).

For the safety-based landing, the previously defined system level function *position synchronization* can be broken down further according to the above described design choices. On the UAV the required sub-functions are *groundspeed tracking (longitudinal)* and *UGV position tracking (lateral)*. On the UGV side the required sub-functions are *UAV position tracking (longitudinal)* and *runway centerline tracking (lateral)*.

With these design choices, *LC-R3* and *LC-R4* have been addressed for the safety-based position synchronization controller. Next, the performance-based landing controller shall be investigated and the remaining bilateral modes from Table 2.12 examined. Table 2.14 shows the meaningful bilateral modes, where modes containing only one operating direction, e.g. (X / X), have been discarded.

Table 2.14 Possible bilateral modes of cooperation for the performance-based landing controller: no cooperation (0), cooperation in x-direction (X), cooperation in y-direction (Y), cooperation in x and y-direction (XY). The x-axis points in runway direction, the y-axis to the right.

UAV / UGV
X / XY
Y / XY
XY / X
XY / Y
XY / XY

All of these modes suffer from the previously described time delay dependency in terms of stability and performance. A stability proof must be provided for any of these modes, while the most promising in terms of performance seems to be the (XY / XY) mode, representing full cooperation in both directions. All other modes are combinatory subsets of the fully coupled configuration. An optimization-based method will be applied in Chapter 4.4.2 to identify the mode with the best performance. In the same chapter the stability of the bilateral landing controller will be analyzed and an appropriate solution will be presented. *LC-R3* and *LC-R4* are therefore addressed for the performance-based position synchronization controller. It should be noted that *LC-R3* cannot be solved in a cooperative way for the vertical direction, since only the aircraft is available to reject a given disturbance. The analysis is therefore focussed on the horizontal plane.

Flare

In the previous section the position synchronization controller has been analyzed and a number of design choices have been made. This resulted in the development of two distinct controller versions, a safety-based and a performance-based position synchronization controller. Now the vertical control approach will be analyzed according to the previously formulated landing controller requirements, and an appropriate solution will be proposed. Throughout the previous section decoupled controllability between the horizontal and vertical motion was assumed, which is a valid assumption if the aircraft controller is capable of providing such decoupling. For the following this will be considered as given. The task of the vertical controller is now to perform a suitable flare maneuver i.e. to

provide appropriate control commands in terms of flight path angle to the underlying aircraft controller such that the desired vertical velocity profile is tracked. *LC-R2* will be taken into consideration by designing the outer-loop flare controller with sufficient bandwidth separation from the underlying aircraft flight path controller.

In contrast to a normal runway landing, it is not desirable to follow a spatially fixed vertical flight path or to reduce the touchdown dispersion with respect to a desired touchdown point on the runway [75], but rather to perform the flare maneuver relative to the current landing platform location, independent of any inertial reference. In the case of a strong headwind, the landing ground speed would be reduced by the amount of the headwind, since landing airspeed would be maintained. The flare controller should therefore only depend on the relative vertical distance between the two vehicles. At the same time, the flare controller must also depend on the relative horizontal positions of the vehicles, to abort the landing attempt and to command a retry if the horizontal synchronization error is too large. A classical flare law, which calculates the desired sink rate based on the current vertical distance between the vehicles in combination with a state machine based fail-safe mechanism is therefore sufficient to meet *LC-R5*, and will be presented in Chapter 4.2.4 and Chapter 4.2.6 respectively.

Engine Retard

Before the aircraft can perform a touchdown on the landing platform, the engine must be switched off to avoid any damage to the engine or the landing platform. This requirement depends on the mechanical setup of the final landing solution, which might not require an engine retard, but in the case of the demonstrator setup an engine retard is required, since the aircraft is touching down on top of a net under tension. The difficulty is now that after engine cut-off only one longitudinal actuator (elevator) is left to control two aircraft states (speed and flight path). As previously mentioned the choice here was taken in favor of speed control, allowing for continued horizontal position synchronization control at the expense of vertical control. The sudden loss of thrust force will lead to a deceleration of the aircraft, which needs to be compensated by an appropriate nose down command. A nose down command will obviously alter the vertical aircraft velocity, which must be controlled in a feed forward manner by choosing the correct retard initiation altitude to produce a desired touchdown velocity. The correct initiation altitude was estimated based on undisturbed measurements at higher altitudes (simulated retard maneuvers), while the influence of the ground effect and ground vehicle induced disturbance were estimated based simulation results. The final solution will be described in detail in Chapter 4.2.5.

Ground Lock

In the last moments before touchdown the aircraft must be transferred from sustained flight to minimum lift configuration (after touchdown) as fast as possible. A sudden wind gust could lead to a local change of relative flow direction or velocity, which might produce a pitching or rolling motion of the aircraft. Such event could easily lead to an uncontrolled contact with the landing platform and might cause a loss of the aircraft. After touchdown the aircraft needs to be secured in the landing platform safely while the joint vehicles are being decelerated to full stop. Therefore the aircraft must detect the touchdown and

send a touchdown confirmation message to the ground vehicle. During the deceleration it is important to safely lock the aircraft to the landing platform so that a release becomes impossible, and to dampen the pitching motion of the aircraft, which is worsened by the elastic properties of the landing platform net. To fulfill *LC-R5* and *LC-R6*, a mechanical locking device was developed, together with an appropriate control surface deflection scheme (symmetric aileron up deflection and nose down elevator) and continued pitch damping control (post-touchdown). The developed solution will be presented in detail in Chapter 4.2.5.

Table 2.15 shows the different low-level functions as they will be used for the safety-based position synchronization controller, while Table 2.16 illustrated the functions of the performance-based landing controller.

Table 2.15 Low-level functions for safety-based landing classified by operating direction and vehicle.

Operating Direction	UAV	UGV
longitudinal	ground speed tracking	UAV position tracking
lateral	UGV position tracking	runway centerline tracking
vertical	flare, retard, ground lock	-

Table 2.16 Low-level functions for performance-based landing classified by operating direction and vehicle.

Operating Direction	UAV	UGV
longitudinal	ground speed tracking UGV position tracking	UAV position tracking
lateral	UGV position tracking	UAV position tracking
vertical	flare, retard, ground lock	-

At this point the main landing controller design choices are available, allowing for a detailed development. Table 2.17 summarizes the design choices for the landing controller.

2.4.6 Deceleration Control

After successful touchdown, the joint vehicles must be decelerated in a safe and efficient manner. Even though the ground vehicle is the main actor during the deceleration process, the aircraft’s active contribution is needed. Firstly, the aircraft should provide a minimum

Table 2.17 Summary of design choices for the landing controller.

No.	Design choice description
1	The position synchronization controllers will be developed as independent SISO-controllers (one per translation degree of freedom) in a distributed setup (on each vehicle).
2	Two different landing control strategies will be developed.
3	The first strategy will be referred to as safety-based landing and will be based on established landing procedures. The aircraft landing airspeed will be commanded to a constant value (in ground speed tracking mode), while a classical flare law will provide a desired vertical velocity profile. The ground vehicle will perform the longitudinal position synchronization task, while the aircraft will be responsible for lateral position synchronization.
4	The safety-based landing controller will require the functions <i>ground speed tracking</i> , <i>UGV position tracking (lateral)</i> , <i>flare</i> , <i>retard</i> , and <i>ground lock</i> on the UAV side, while the functions <i>UAV position tracking (longitudinal)</i> and <i>runway centerline tracking</i> are used on the UGV side.
5	The safety-based landing controller is naturally stable (open-loop control), but does require a lag correction system.
6	The second strategy will be referred to as performance-based landing and will make use of both vehicles to perform the position synchronization task. Disturbances in the horizontal plane will therefore be rejected in a cooperative manner.
7	The performance-based landing controller's stability depends on the communication time delay (closed-loop control), and requires a lag correction system as well as stability provision.
8	Vertical control will be provided by a classical flare control law in combination with a state machine based fail-safe mechanism, which will inhibit a further descent if the required landing accuracy is not met.
9	The performance-based landing controller will cover the function <i>ground speed tracking</i> , <i>UGV position tracking (lateral)</i> , <i>flare</i> , <i>retard</i> , and <i>ground lock</i> on the UAV side and <i>UAV position tracking (longitudinal)</i> and <i>UAV position tracking (lateral)</i> on the UGV side.
10	A horizontal motion decoupling scheme is not required for the safety-based landing, but will be included for the performance-based landing.

lift configuration to reduce the risk of undesired lift forces due to a sudden wind gust. Also, the elastic nature of the landing platform surface causes problems, with the aircraft

tending to increased pitching oscillations, where a constant nose down command would not be sufficient. Active pitch oscillation damping is needed, as well as lift minimization to prevent any positive angles of attack and unstable pitching oscillations. Secondly, for larger aircraft, active braking assistance could be provided by appropriate spoiler, throttle, or surface deflection settings (not relevant for the demonstrator setup). On the ground vehicle side, after the touchdown confirmation message was received from the aircraft, a smooth deceleration profile must be tracked. Also, if a lateral offset had developed during the flare maneuver, the ground vehicle must be capable of drive the joint vehicles toward the runway centerline if desired. *LC-R7* is met by an appropriate design of the outer-loop deceleration controller by providing sufficient bandwidth separation to the UGV inner-loop velocity controller.

2.4.7 High-Level Mission Control and Fail-Safe System

For a safe execution of the landing maneuver, a high-level mission (or maneuver) control scheme must be available to automatically control the transitions between the previously described controller modes (approach, landing, retard, ground lock, deceleration). This high-level mission controller must also contain a fail-safe system, which automatically detects and mitigates failures, such as insufficient landing accuracies, critical communication delays, or sensor faults. The fail-safe system must also contain a flight termination functionality based on geofences to prevent fly-aways. For this purpose a finite state machine will be developed in Chapter 4.2.6.

Chapter Conclusions

Throughout this chapter a functional analysis of the envisioned cooperative landing system was performed and the main design choices formulated. These include the development of two distinct landing control system, where the first is based on safety considerations and the adherence to existing landing procedures, while the second focuses on the performance of the landing maneuver in terms of used runway length. The requirements of the landing control system and the individual vehicle controllers were defined and provide sufficient direction for the following detailed development process.

3 Modeling and Control of Aircraft and Ground Vehicle

Chapter Overview

The design of an efficient cooperative landing strategy requires efficient independent vehicle control. Therefore the dynamics of the involved vehicles need to be analyzed and appropriate control approaches implemented. The following section presents the mathematical models and methods used for simulation and controller synthesis of each vehicle respectively. Nonlinear models, valid over a large region of parameters as well as high fidelity linear models are developed. Appropriate control strategies for both vehicles are analyzed and the respective controllers developed.

3.1 Aircraft Flight Dynamics Model

The aircraft flight dynamics model is based on a rigid body model in six degrees of freedom, with aerodynamic, propulsion and gravitational forces and moments acting on it. The following sections provide an overview of the derived sub-models. In the first step the different reference frames used to describe the aircraft motion are defined.

3.1.1 Coordinate Frames and Notation

When developing the complete aircraft equations of motion, it is very important to precisely describe the attitude and motion of the aircraft with respect to its environment, i.e. the surrounding airmass and the inert ground. Thereby quantities related to the relative motion between the aircraft and its surrounding airmass will be denoted as *aerodynamic quantities* (index a), between the aircraft and the ground as *kinematic quantities* (index k), and

between the airmass and the ground as wind quantities (index w). These quantities are used to describe the aircraft motion and can be expressed in any given reference frame. In the following on the reference frames used in this work are presented, while many more systems can be found in the flight mechanics literature.

- *The geodetic or NED-frame (superscript g):* The geodetic or NED-frame has its origin at the aircraft center of gravity (or another aircraft-fixed reference point) and moves along with the aircraft. The x-axis points to the geographic north pole and is parallel to the local geoid surface, the y-axis points east and is parallel to the local geoid surface while forming a right-handed system with the x and z axes, the z-axis points downwards and is perpendicular to the local geoid surface. It rotates with the *transport rate* to maintain NED-alignment as the aircraft flies along the curved surface of the earth.
- *The body frame (superscript b):* The body frame has its origin at the aircraft center of gravity (or another aircraft-fixed reference point) and moves along and rotates with the rigid body aircraft. The x-axis points towards the aircraft nose and is located in the xz-symmetry plane, the y-axis points to the right (starboard wing) and forms a right-handed system with the x and z axes, while the z-axis points downwards in the aircraft symmetry plane and is perpendicular to the x and y axes.
- *The aerodynamic frame (superscript a):* The aerodynamic frame has its origin at the aircraft center of gravity (or another aircraft-fixed reference point), moves along with the aircraft and rotates with the direction of the relative airflow. The x-axis is aligned with the aerodynamic velocity vector and points in the direction of the negative flow direction. The y-axis points to the right and is perpendicular to the x and z axes. The z-axis points downwards in the symmetry plane of the aircraft and is perpendicular to the xy-plane.
- *The kinematic frame (superscript k):* The kinematic frame (also known as flight path axes) has its origin at the aircraft center of gravity (or another aircraft-fixed reference point), moves along with the aircraft and rotates with the direction of the kinematic (or flight path) velocity vector. The x-axis is aligned with the kinematic velocity vector and points in the same direction. The y-axis points to the right and lies in the earth's horizontal plane. The z-axis points downwards and forms a right-handed system with the x and y axes.
- *The navigation or local runway frame (superscript n):* The navigation frame has its origin at some point on the earth's surface and thus rotates with the angular rate of the earth. The x-axis point to an arbitrary direction (specified with respect to geographic north). The y-axis forms a right-handed system with the x and z axes and is parallel to the local geoid surface. The z-axis points downwards and is perpendicular to the local geoid surface. In the present work (landing scenario) the navigation frame will be used to describe the motion of the aircraft with respect to a given runway, where the x-axis points along the runway centerline in approach direction, the y-axis to the right, and the z-axis downwards. It will therefore also be referred to as local runway frame.

A more detailed description of these coordinate systems can be found in any standard textbook (e.g. [23]), although it should be noted that the definition and use of these axes systems is not consistent over different literature and depending of the specific application additional coordinate systems (e.g. stability frame, experimental frame, sensor frame) might be used.

3.1.2 Rigid Body Equations of Motion

The derivation of the rigid body equations of motion is straightforward and can be found in any textbook on flight dynamics and control like [110, 13, 10]. For the landing application considered in this work, only a local area around the landing strip will be covered by the aircraft, which is why flat earth equations provide sufficient precision. The reference frame of choice is the aircraft body frame, being a commonly used compromise between coordinate transformation effort and physical interpretability. In the following the non-linear, flat-earth, six-degrees-of-freedom (6-DoF), body-axes equations are provided for completeness — a detailed derivation is left out here in favor of a concise document.

The rigid body aircraft possesses six degrees of freedom (6-DOF), including three translational and three rotational dimensions. Each DOF is defined by two states, which consist of velocity and position for the translational DOFs and angular velocity and attitude angles for the rotational DOFs. The derivation of the aircraft equations of motion is based on Newtown's Second Law, which only holds in an inertial reference frame, thus resulting in 12 inertially referenced state variables. These contain the inertial position vector expressed in the navigation frame:

$$\mathbf{p}^n = \begin{bmatrix} p_n \\ p_e \\ p_d \end{bmatrix}_n \quad (3.1)$$

where the indices n , e , and d represent the position vector components in north, east, and down directions. The kinematic velocity vector expressed in the body frame:

$$\mathbf{V}_k^b = \begin{bmatrix} u_k \\ v_k \\ w_k \end{bmatrix}_b \quad (3.2)$$

The attitude angles (Euler Angles) vector (relating the body frame to the geodetic frame):

$$\boldsymbol{\phi}^b = \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix}_b \quad (3.3)$$

The kinematic angular velocity vector expressed in the body frame:

$$\boldsymbol{\omega}^b = \begin{bmatrix} p_k \\ q_k \\ r_k \end{bmatrix}_b \quad (3.4)$$

In the following the index k will be omitted from the inertially referenced state variables in favor of a simpler notation. The states can now be calculated by integrating the following set of coupled, non-linear vector differential equations:

$$\begin{bmatrix} \dot{p}_n \\ \dot{p}_e \\ \dot{p}_d \end{bmatrix} = \begin{bmatrix} c_\theta c_\psi & s_\phi s_\theta c_\psi - c_\phi s_\psi & c_\phi s_\theta c_\psi + s_\phi s_\psi \\ c_\theta s_\psi & s_\phi s_\theta s_\psi + c_\phi c_\psi & c_\phi s_\theta s_\psi - s_\phi c_\psi \\ -s_\theta & s_\phi c_\theta & c_\phi c_\theta \end{bmatrix} \begin{bmatrix} u \\ v \\ w \end{bmatrix} \quad (3.5)$$

$$\begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} = \begin{bmatrix} rv - qw \\ pw - ru \\ qu - pv \end{bmatrix} + \frac{1}{m} \begin{bmatrix} f_x \\ f_y \\ f_z \end{bmatrix} \quad (3.6)$$

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \sin \phi \tan \theta & \cos \phi \tan \theta \\ 0 & \cos \phi - \sin \phi \\ 0 & \frac{\sin \phi}{\cos \theta} & \frac{\cos \phi}{\cos \theta} \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \quad (3.7)$$

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} \Gamma_1 pq - \Gamma_2 qr \\ \Gamma_5 pr - \Gamma_6 (p^2 - r^2) \\ \Gamma_7 pq - \Gamma_1 qr \end{bmatrix} + \begin{bmatrix} \Gamma_3 l + \Gamma_4 n \\ \frac{1}{J_y} m \\ \Gamma_4 l + \Gamma_8 n \end{bmatrix} \quad (3.8)$$

where:

$$\begin{aligned} \Gamma_1 &= \frac{J_{xz}(J_x - J_y + J_z)}{J_x J_z - J_{xz}^2} & \Gamma_5 &= \frac{J_z - J_x}{J_y} \\ \Gamma_2 &= \frac{J_z(J_z - J_y) + J_{xz}^2}{J_x J_z - J_{xz}^2} & \Gamma_6 &= \frac{J_{xz}}{J_y} \\ \Gamma_3 &= \frac{J_z}{J_x J_z - J_{xz}^2} & \Gamma_7 &= \frac{(J_x - J_y)J_x + J_{xz}^2}{J_x J_z - J_{xz}^2} \\ \Gamma_4 &= \frac{J_{xz}}{J_x J_z - J_{xz}^2} & \Gamma_8 &= \frac{J_x}{J_x J_z - J_{xz}^2} \end{aligned} \quad (3.9)$$

and:

$$\begin{aligned} J_x &= \int (y^2 + z^2) dm & J_z &= \int (x^2 + y^2) dm \\ J_y &= \int (x^2 + z^2) dm & J_{xz} &= \int (x \cdot z) dm \end{aligned} \quad (3.10)$$

J_x , J_y , and J_z represent the moments of inertia with respect for the body frame, while J_{xz} represents the product of inertia. Here the rigid body is assumed to be symmetric with respect to its vertical plane (xz-plane), which is a common assumption for aircraft. The remaining off-diagonal elements (J_{xy} and J_{yz}) in the inertia matrix are thus set to zero.

These equations assume that the total external force and moment are applied with respect to the center of mass of the rigid body. As previously mentioned, this external wrench is a sum of all forces and moments acting on the aircraft, where the force vector $\mathbf{f} = (f_x, f_y, f_z)^\top$ and moment vector $\mathbf{m} = (l, m, n)^\top$ in Equations (3.6) and (3.8) contain the aerodynamic, propulsion and gravity forces and the moments, which depend on the aircraft states and

the control inputs (control surface deflections and engine throttle). These components will be developed throughout the following sections.

3.1.3 Aerodynamics

The derivation of an accurate aerodynamics model for fixed-wing aircraft represents an elaborate and costly task, and typically requires detailed numerical simulations, the execution wind tunnel tests, as well as extensive flight testing for system identification. For the early conceptual phase of a given flight control project, it is common to develop a lower fidelity model, based on the aircraft geometry by means of simpler numerical methods. Such approach reduces the development time and effort, thus allowing for a first qualitative analysis of the developed methods. Typically these low-fidelity models are capable of capturing the aircraft dynamical properties (eigenmodes) with sufficient accuracy to allow for a use as a priori models for system identification maneuver design, as will be presented throughout the next sections. Other high-fidelity modeling approaches are typically too costly and time consuming to generate full-envelope models and are rather used for deriving locally valid models for controller synthesis. The geometry-based modeling approach used for this work will be presented in the following, and was based on the work provided in [64].

The aerodynamic forces and moments acting on an aircraft are typically defined in terms of dimensionless aerodynamic coefficients:

$$\begin{aligned} D &= \frac{\rho}{2} V_a^2 \cdot S \cdot C_D & l &= \frac{\rho}{2} V_a^2 \cdot S \cdot c \cdot C_l \\ Y &= \frac{\rho}{2} V_a^2 \cdot S \cdot C_Y & m &= \frac{\rho}{2} V_a^2 \cdot S \cdot c \cdot C_m \\ L &= \frac{\rho}{2} V_a^2 \cdot S \cdot C_L & n &= \frac{\rho}{2} V_a^2 \cdot S \cdot c \cdot C_n \end{aligned} \quad (3.11)$$

where D , Y , and L represent the drag, side and lift forces respectively, while l , m , and n represent the rolling, pitching and yawing moments, ρ the air density, S the planform area of the wing, c the mean chord, and V_a the airspeed. C_D , C_Y , C_L , C_l , C_m , and C_n represent the nondimensional aerodynamic force and moment coefficients, which are typically nonlinear functions of the aerodynamic angles (α and β), the dimensionless rotational rates (\hat{p} , \hat{q} , \hat{r}), and the control surface deflections of aileron, elevator, rudder and flaps (δ_a , δ_e , δ_r , δ_f). These functions are often approximated by multivariate Taylor series, which in case of the lift force coefficient results in a polynomial expression of the form:

$$C_L = C_{L0} + \frac{\partial C_L}{\partial \alpha} \alpha + \frac{\partial C_L}{\partial \alpha^2} \alpha^2 + \frac{\partial C_L}{\partial \hat{q}} \hat{q} + \frac{\partial C_L}{\partial \delta_e} \delta_e + \dots \quad (3.12)$$

where the angular rates are normalized to produce dimensionless moment coefficients (e.g. $\hat{q} = q \cdot c / 2V_a$). The same procedure can be applied to the remaining force and moment coefficients. The different partial derivatives in Equation (3.12) are called aerodynamic stability and control derivatives. Given the polynomial model structure, the main task now consists in the selection of significant monomials from the overall set of candidate regressors, and in estimating the values of the different aerodynamic derivatives. A manual

monomial selection method will be presented in the following, while automated methods (e.g. via stepwise multivariable regression [29]) also exist. The manual approach was favored here to obtain a deeper insight into the physical principles governing the given aircraft's aerodynamics.

In a first step, a number of random samples of the relevant input variables (aerodynamic angles, rotational rates and control surface deflections) are generated. A cosine-like probability density distribution is then used to increase the model precision close to the nominal flight conditions. Then numerical tools, in this case the potential flow solver AVL [31] and the 2D-solver XFOIL [30] (for capturing nonlinear drag effects like friction drag), are used to evaluate the aerodynamic coefficients for all generated inputs. The input bounds considered for the Penguin BE UAV (demonstrator setup) are provided in Table 3.1.

Table 3.1 Considered value ranges for Penguin BE aerodynamic parameters used for geometry-based modeling. These include aerodynamic angles, angular rates and control inputs.

Angles	Rates	Control Inputs
$\alpha \in [-10^\circ, 15^\circ]$	$\hat{p} \in [-0.10, 0.10]$	$\delta_a, \delta_e, \delta_r \in [-20^\circ, 20^\circ]$
$\beta \in [-20^\circ, 20^\circ]$	$\hat{q} \in [-0.03, 0.03]$	$\delta_f \in [0^\circ, 30^\circ]$
	$\hat{r} \in [-0.25, 0.25]$	

The resulting point clouds are fitted in a least-square sense with multi-dimensional polynomials in the input variables, as illustrated in Figure 3.1.

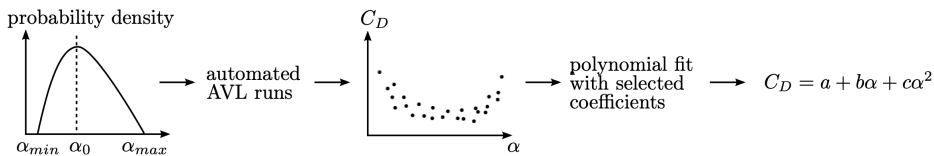


Figure 3.1 Process of geometry-based aerodynamics modeling using the potential flow solver AVL [64]. Assuming a probability density function for the considered value ranges, the AVL software computes point estimates of relevant aerodynamic forces and moments. Then, a polynomial fit with appropriate coefficients is used to explain the data.

The relevant monomials are determined by inspecting the distribution of the aerodynamic coefficients over the input variables, and can also cover higher-order and coupled effects of the form:

$$x_1^{e_1} \cdot \dots \cdot x_n^{e_n} \quad (3.13)$$

where e_1 to e_n represent different possible exponents. To give an example, Figure 3.2a shows a linear dependency on the angle of attack α for the lift coefficient, of the form:

$$C_L(\alpha) = C_{L_0} + C_{L_\alpha} \cdot \alpha \quad (3.14)$$

while the variation of all other model inputs is also visible in the plot. A range of lift coefficient values is found for a given constant value of α , which is due to the changes in the other model inputs. As can be seen in Figure 3.2b, such strong correlation is not directly visible for the aileron deflection. Further dependencies can be found by removing the dominant dependency on alpha from the data and plotting the error $C_L - C_L(\alpha)$ of the initial approximation shown in Equation (3.14) over the different remaining aerodynamic parameters. Investigating the residual total lift coefficient dependency on the aileron

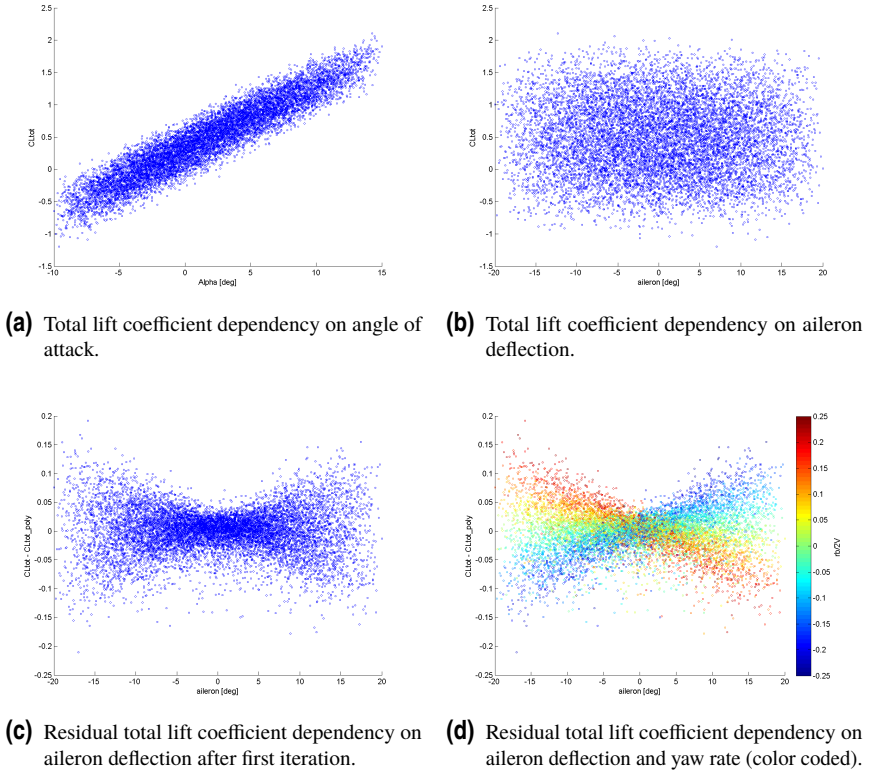


Figure 3.2 Scatter plots of total lift coefficient versus different model inputs before (a,b) and after (c,d) one model structure iteration. These scatter plots are used to find corresponding dependencies of states to aerodynamic forces and moments.

deflection, it seems that the aileron has an influence on the lift coefficient, as shown in Figure 3.2c. Nevertheless, it is not quite clear how, until more information is included.

Figure 3.2d shows the same plot with added coloring according to the yaw rate, which reveals that there is a linear coupling effect between aileron and yaw rate. A monomial of the form $\hat{r} \cdot \delta_a$ should therefore be included in the next iteration of the polynomial fitting. A thorough investigation of all dependencies in such iterative manner provides polynomials that are valid over a large parameter range and can be used to model most of the aircraft's flight envelope. Compared to multidimensional lookup tables, which are traditionally used in flight dynamics modeling, polynomials have the advantage of significantly shorter calculation times, which is particularly important in real time applications.

Six iterations were found to produce sufficiently precise polynomials, where the normalized root-mean-square deviations remained under 10%, as listed in Table 3.2. The final set of monomials is shown in Table 3.3, while the actual values of the corresponding derivatives have been omitted here.

Table 3.2 Root-mean-square deviations (RMSD) and normalized root-mean-square deviations (NRMSD) of polynomial approximations after 6 iterations.

Value	Range	RMSD	NRMSD
C_D	$-0.03 \dots 0.68$	0.0333	4.7%
C_Y	$-0.27 \dots 0.26$	0.0089	6.3%
C_L	$-0.78 \dots 1.79$	0.0039	1.3%
C_l	$-0.17 \dots 0.18$	0.0019	9.5%
C_m	$-1.42 \dots 1.80$	0.0162	1.0%
C_n	$-0.15 \dots 0.17$	0.0098	10.3%

3.1.4 Propulsion

The electric propulsion system consists of two parts, the motor model and the propeller model. The motor model assumes perfect efficiency, with 100% of the electrical power being transformed to mechanical power. Naturally there are some losses in real motors, but electrical motors generally possess high efficiencies, thus allowing for this simple approximation. The conversion from throttle command to electrical power is modeled via a lookup table with power consumption values from static ground tests.

The propeller is modeled using the dimensionless thrust and power coefficients, which are defined as:

$$C_T = \frac{T}{\rho n^2 D^4} \quad \text{and} \quad C_P = \frac{P}{\rho n^3 D^5} \quad (3.15)$$

where ρ represents the atmospheric density, n the propeller angular velocity, D the propeller

Table 3.3 Final set of monomials used in the polynomial approximations of the different aerodynamic coefficients.

$C_D = f(1, \alpha, \hat{q}, \delta_f, \alpha^2, \beta^2, \hat{q}^2, \delta_a^2, \delta_e^2, \delta_r^2, \delta_f^2, \alpha\hat{q}, \beta\hat{r}, \hat{p}\delta_a, \hat{q}\delta_e, \alpha\delta_f, \alpha^3, \alpha\delta_a^2, \alpha^2\delta_f, \alpha\hat{r}\delta_a, \alpha^4, \alpha^2\delta_a^2, \alpha^3\hat{r}\delta_a)$
$C_Y = f(\beta, \hat{p}, \hat{r}, \delta_a, \delta_r, \alpha\beta, \alpha\hat{p}, \beta\hat{q}, \hat{q}\hat{r}, \hat{r}\delta_e, \hat{q}\delta_r, \beta^3, \alpha^2\beta, \beta\delta_a^2, \alpha\beta^3, \alpha^2\beta^3)$
$C_L = f(1, \alpha, \hat{q}, \delta_e, \delta_f, \alpha^2, \beta^2, \hat{q}^2, \hat{r}^2, \alpha\hat{q}, \hat{p}\hat{r}, \alpha\delta_e, \hat{q}\delta_e, \hat{r}\delta_a, \hat{r}\delta_r, \alpha^3, \alpha\beta^2, \alpha\hat{r}^2, \beta^2\delta_e, \beta^2\delta_f)$
$C_l = f(\beta, \hat{p}, \hat{r}, \delta_a, \delta_r, \alpha\beta, \alpha\hat{r}, \alpha\delta_a, \hat{q}\delta_a, \alpha\delta_r, \beta^2\delta_a)$
$C_m = f(1, \alpha, \hat{q}, \delta_e, \delta_f, \beta^2, \hat{q}^2, \delta_a^2, \delta_e^2, \alpha\hat{q}, \beta\hat{r}, \hat{r}\delta_a, \hat{q}\delta_e, \beta\delta_r, \hat{r}\delta_r, \alpha\beta^2, \beta^2\hat{q}, \beta^2\delta_e, \hat{q}^2\delta_e)$
$C_n = f(\beta, \hat{p}, \hat{r}, \delta_a, \delta_r, \alpha\hat{p}, \alpha\hat{r}, \alpha\delta_a, \hat{q}\delta_a, \hat{r}^3, \delta_a^3, \alpha^2\hat{r}, \alpha^2\delta_a, \hat{r}\delta_a^2, \alpha\hat{r}^3, \alpha^3\delta_a, \alpha\delta_a^3, \alpha^2\hat{r}^3, \hat{r}^3\delta_a^2, \alpha^2\delta_a^3, \alpha^3\delta_a^3)$

diameter, T the thrust force, and P the power. The force coefficient C_T and power coefficient C_P are functions of the propeller geometry, the advance ratio $J = \frac{V}{nD}$ (with free stream velocity V), the angular velocity and the Reynolds number. The explicit dependency on angular velocity and Reynolds number is often neglected.

The values of C_T and C_P can either be approximated based on the propeller geometry (typically via the pitch/diameter ratio [64]), or determined experimentally [12]. For the present work, the performance data of the APC 19x11 propeller (as used on the Penguin BE UAV) was provided by the manufacturer as tabular data of $C_T(J)$ and $C_P(J)$ for different angular velocities. The propeller thrust and torque can then be calculated as follows:

$$T = C_T \rho n^2 D^4 \quad \text{and} \quad Q_{prop} = \frac{P}{\omega} = \frac{C_P n^2 D^5}{2\pi} \quad (3.16)$$

Finally, the rotational velocity of the propeller is described by the ordinary differential equation:

$$I \cdot \dot{\omega} = Q_{motor} - Q_{prop} \quad (3.17)$$

where I represents the moment of inertia of the propeller and $\dot{\omega}$ its angular acceleration. The inertia of the motor was neglected as it is significantly smaller than the inertia of the propeller. The propulsion model was compared to static thrust measurements (see Figure 3.3), and showed a good fit for low to medium throttle settings, but overestimates the thrust for high throttle settings.

3.1.5 Mass and Inertia

The modeling of mass and inertia of an electrically powered aircraft is rather straightforward, since these quantities remain constant during the entire flight. The aircraft mass and center of gravity can be directly measured by means of a scale and balancing device (or a number of scales for bigger aircraft), while the estimation of the different moments

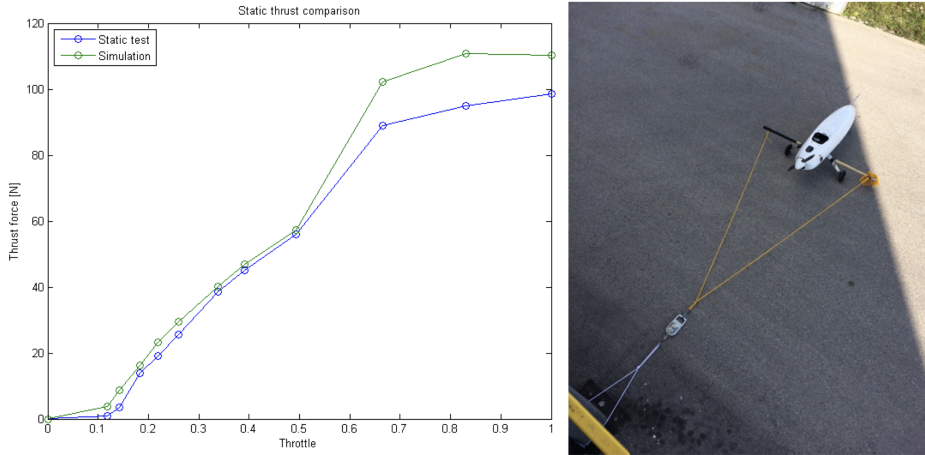


Figure 3.3 Comparison of propulsion model and static thrust measurements (left). The static thrust force against throttle is measured with a force sensor using the real hardware (right). The thrust model predicts the measured static force with a reasonable accuracy.

and products of inertia is more difficult. It is possible to measure the moments of inertia by means of a pendulum setup, where the aircraft is suspended on a torsional spring. The suspended aircraft is then deflected and performs damped oscillations, where the oscillation period can be measured using a stopwatch. The moments of inertia can then be calculated from the measured oscillation period and the known stiffness of the spring as described in [64]. In any case, such measurements are difficult to perform (especially for larger aircraft) and do not allow for the estimation of products of inertia, which in general are nonzero for classical aircraft (in particular J_{xz}).

To overcome these drawbacks the inertia tensor was estimated using another method, which is based on the measurement of as many aircraft component masses as possible. Regarding the structural aircraft components, the mass was assumed to be uniformly distributed in the external surface (with uniform thickness) of the fuselage and in the volumes of the wings and the tail. The weight is concentrated at the centroid of the defined cross-sections in the case of the fuselage and at the quarter-chord point of the distributed sections along the span for the wing and tail, as illustrated in Figure 3.4. All other components are modeled as point masses at their respective locations as shown in Figure 3.4c. This results in a large number of point masses, from which the moments and products of inertia can be calculated.

3.1.6 High Fidelity Modeling via System Identification

Elaborate numerical methods such as Computational Fluid Dynamics (CFD) typically suffer from high computational and cost demands, while simpler methods like Vortex Lattice (VLM), or 3D-Panel methods typically result in low fidelity models due to their

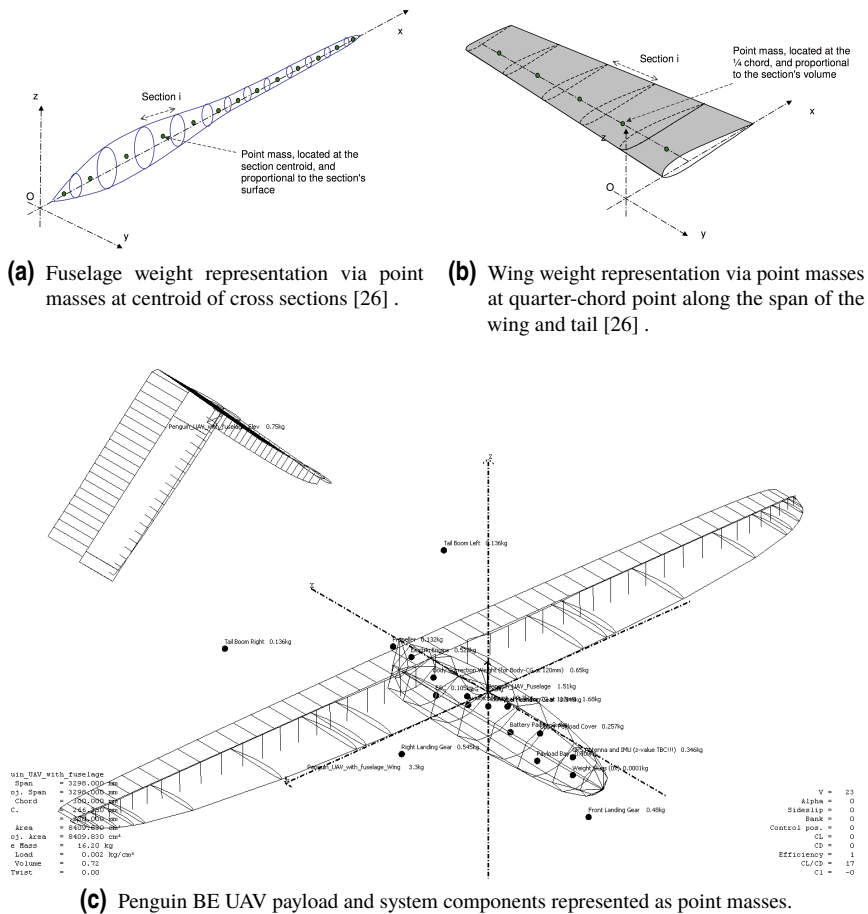


Figure 3.4 Weight and balance estimation for payload and structural aircraft components. The mass is assumed to be uniformly distributed, thus point masses are distributed over the aircraft geometry to compute the moments and products of inertia.

inherent simplifying assumptions. Wind-tunnel testing often lies outside the expertise of the aerial robotics community and also involves constraints due to scaling. In this sense, aircraft system identification is a compelling option for the generation of high fidelity models once the aircraft is able to fly. Here the so-called *Two-Step Method* is used, but other methods such as the Output Error Method can also be employed [58]. It involves data gathering, data compatibility checking, parameter identification and model validation — the process is depicted in Figure 3.5. Further details on the applied approach can be found in [81].

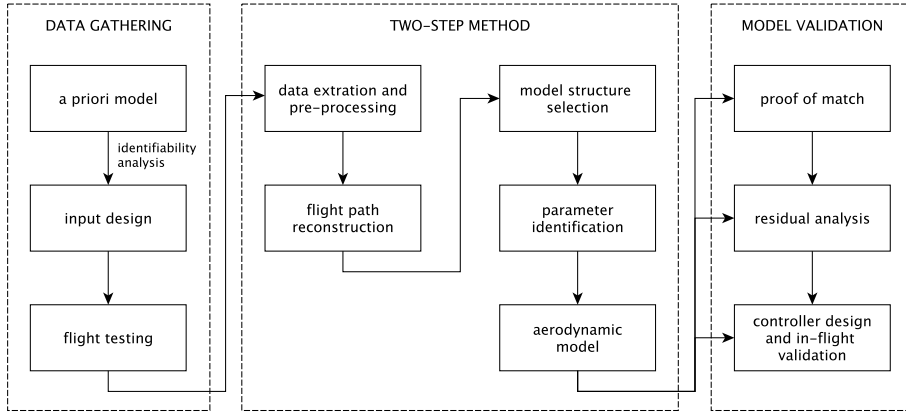


Figure 3.5 Overview of the two-step method for system identification. The individual steps consist of data gathering, data preprocessing, parameter identification and model validation.

Data Gathering

One key criteria for data gathering consists in the identifiability of the aerodynamic parameters, where the recorded data must contain enough information about the parameters to be identified. To meet such criterion, the aircraft is trimmed and the designed control inputs such as *elevator pulse* or *3-2-1-1* are executed to excite the eigenmotion of the aircraft (e.g. phugoid and short period for the longitudinal dynamics). In practice the same system identification maneuvers are repeated several times to mitigate the effects of both the variations in initial conditions and atmospheric disturbances. Then, the data is separated into an identification and validation set. These data sets consist of phugoid (elevator pulse), short period (3-2-1-1) and thrust variation (throttle doublet) maneuvers.

Data Compatibility Check

The data compatibility check within the two-step method has three purposes, namely the estimation of aerodynamic forces and moments from measurements, the correction of measurement errors due to sensor biases and drifts, and the smoothing of the estimates. For this purpose an Iterative Extended Kalman Filter (IEKF) with backward Kalman smoothing (EKS) was implemented [89].

The relevant state vector is defined as $x(t) = [x_a(t), \mathfrak{A}(t), \Theta]^T$ where the aircraft states are defined as $x_a(t) = [u, v, w, p, q, r, \phi, \theta, \psi, h]^T$, with h representing the aircraft altitude above mean sea level. $\mathfrak{A}(t) = [\mathfrak{A}_0, \mathfrak{A}_1, \mathfrak{A}_2]^T$ represents the vector of forces and moments (x-, y- and z-axes in sequential order) as a third order Gauss-Markov process where $\mathfrak{A}_0 = [X, Y, Z, L, M, N]^T$, $\mathfrak{A}_1 = \dot{\mathfrak{A}}_0$ and $\mathfrak{A}_2 = \dot{\mathfrak{A}}_1$. Lastly, the instrumentation error vector includes biases contained in the inertial measurements $\Theta = [\Delta a_x, \Delta a_y, \Delta a_z, \Delta p, \Delta q, \Delta r]^T$. The aircraft states $x_a(t)$ are governed by the aircraft equations of motion as provided in

Equation (3.18). The aircraft is assumed to be a rigid-body with a constant mass. Zero wind is assumed.

$$\begin{aligned}
 \dot{u} &= -qw + rv - g \sin \theta + C_X \\
 \dot{v} &= -ru + pw + g \cos \theta \sin \phi + C_Y \\
 \dot{w} &= qu - pv + g \cos \theta \cos \phi + C_Z \\
 \dot{p} &= pqC_{11} + qrC_{12} + qC_{13} + C_L + C_N C_{14} \\
 \dot{q} &= prC_{21} + (r^2 - p^2)C_{22} - rC_{23} + C_M \\
 \dot{r} &= pqC_{31} + qrC_{32} + qC_{33} + C_L C_{34} + C_N \\
 \dot{\phi} &= p + q \sin \phi \tan \theta + r \cos \phi \tan \theta \\
 \dot{\theta} &= q \cos \phi - r \sin \phi \\
 \dot{\psi} &= q \sin \phi \sec \theta + r \cos \phi \sec \theta \\
 \dot{h} &= u \sin \theta - v \cos \theta \sin \phi - w \cos \theta \cos \phi
 \end{aligned} \tag{3.18}$$

In Equation (3.18) the terms C_X , C_Y , C_Z , C_L , C_M , and C_N represent normalized forces and moments, while C_{ij} represent different constants defined by the aircraft moments of inertia. The detailed derivation and notations of Equation (3.18) can be found in [58] and reference therein. Unlike many widely used definitions of aircraft equations of motion, a numerical differentiation of angular rates is avoided here.

A third order Gauss-Markov process, as shown in Equation (3.19), is an auto-regressive process of order three, where its polynomial coefficients are updated by the recursive algorithm at each time step. \mathbf{w} denotes Gaussian noise.

$$\begin{bmatrix} \dot{\mathbf{a}}_0 \\ \dot{\mathbf{a}}_1 \\ \dot{\mathbf{a}}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{a}_0 \\ \mathbf{a}_1 \\ \mathbf{a}_2 \end{bmatrix} + \begin{bmatrix} \mathbf{w}_{\mathbf{a}_0} \\ \mathbf{w}_{\mathbf{a}_1} \\ \mathbf{w}_{\mathbf{a}_2} \end{bmatrix} \tag{3.19}$$

Accelerometer biases are modeled as Markov processes as shown in Equation (3.20), where $i = x, y, z$. The biases in angular velocities are assumed to be constants ($\dot{\Delta p}, \dot{\Delta q}, \dot{\Delta r} = 0$).

$$\dot{\Delta a}_i = -\frac{1}{\tau} \Delta a_i + \mathbf{w}_{\Delta a_i} \tag{3.20}$$

In Equation (3.20), τ denotes a time constant, where a value of 10 was chosen for the current implementation. Modeling the accelerometer biases as Markov processes is motivated by the reconstructability analysis [89].

Define $y(t) = [u_m, v_m, w_m, p_m, q_m, r_m, \phi_m, \theta_m, \psi_m, h_m, V_{a_m}, a_{x_m}, a_{y_m}, a_{z_m}]^T$ and consequently $z(t_k) = [u_{\text{GPS}}, v_{\text{GPS}}, w_{\text{GPS}}, p_{\text{IMU}}, q_{\text{IMU}}, r_{\text{IMU}}, \phi_{\text{KF}}, \theta_{\text{KF}}, \psi_{\text{KF}}, h_{\text{GPS}}, V_{a_{\text{AP}}}, a_{x_{\text{IMU}}}, a_{y_{\text{IMU}}}, a_{z_{\text{IMU}}}]^T$, which are the vectors of measurements in continuous and discrete time respectively. Equation (3.21) presents the non-trivial parts of the measurement model,

which correspond to measurement biases.

$$\begin{aligned}
 p_m(t) &= p(t) + \Delta p(t) \\
 q_m(t) &= q(t) + \Delta q(t) \\
 r_m(t) &= r(t) + \Delta r(t) \\
 a_{x_m}(t) &= \frac{X(t)}{m} + \Delta a_x(t) \\
 a_{y_m}(t) &= \frac{Y(t)}{m} + \Delta a_y(t) \\
 a_{z_m}(t) &= \frac{Z(t)}{m} + \Delta a_z(t) \\
 V_{a_m}(t) &= K_{v_a} \sqrt{u(t)^2 + v(t)^2 + w(t)^2}
 \end{aligned} \tag{3.21}$$

The Euler angles ϕ_{KF} , θ_{KF} , ψ_{KF} are estimated using a Kalman filter (KF) which runs in real time. Note that the IMU is assumed to be located at the center of gravity of the vehicle in Equation (3.21). In summary, the state and measurement equations are described with $x(t) \in \mathbb{R}^{37 \times 1}$ and $y(t) \in \mathbb{R}^{16 \times 1}$.

Parameter Identification and Validation

Assuming decoupled longitudinal and lateral dynamics, and performing a multivariate Taylor series approximation, the total longitudinal forces and moments (aerodynamics and propulsion) result in Equation (3.22), which is often referred to as linear derivative aerodynamics model.

$$\begin{aligned}
 X &= X_0 + X_u u + X_w w + X_q q + X_{\delta_e} \delta_e + X_{\delta_\tau} \delta_\tau \\
 Z &= Z_0 + Z_u u + Z_w w + Z_q q + Z_{\delta_e} \delta_e \\
 M &= M_0 + M_u u + M_w w + M_q q + M_{\delta_e} \delta_e
 \end{aligned} \tag{3.22}$$

The time histories of aircraft states, control inputs, forces and moments in Equation (3.22) are estimated and smoothened using the EKS. The parameters in Equation (3.22) are also called stability and control derivatives. As Equation (3.22) can be formulated as a linear regression problem, the least squares method can be applied to identify the stability and control derivatives.

A strict separation between the identification and the validation data sets is necessary, where ratios of 6/4 or 8/2 are common in terms of the number of maneuvers contained in each set (ID / VAL). Once identified, the parameters can then be evaluated based on the validation set, where an error analysis is conducted on the output variables (aerodynamic and propulsion forces and moments). Furthermore a forward simulation based on the aircraft equations of motion can be used for further validation. Several statistical metrics such as goodness of fit (GOF), Theil's inequality coefficient (TIC) and FAA standards of fidelity [40] can then be used for the assessment of the model prediction capabilities.

The identified parameters for the Penguin BE aircraft are shown in Table 3.4.

By applying the previously mentioned proof of match and residual analysis, different statistical metrics such as the coefficient of determination (R^2), the root mean squared error (RMSE), and the normalized root mean squared error (NRMSE) can be used. The NRMSE value is normalized with respect to the range of the dataset. The standard definitions of these statistical metrics can be found in [57] and [99].

Table 3.4 Identified stability and control derivatives for Penguin BE linear aerodynamics model as defined in Equation (3.22).

Term	Value	Term	Value	Term	Value
X_0	9.2587	Z_0	-0.4544E2	M_0	-2.1589
X_u	-2.2797	Z_u	-0.0474E2	M_u	0.0928
X_w	1.3422	Z_w	-0.0904E2	M_w	-0.5579
X_q	-1.3519	Z_q	-0.8821E2	M_q	-6.8804
X_{δ_e}	27.3732	Z_{δ_e}	3.3388E2	M_{δ_e}	-48.9397
X_{δ_r}	51.4792				

The validation results are presented in Figure 3.6 and Table 3.5. The predicted longitudinal forces and moments using the identified model are compared to the smoothed estimates of forces and moments obtained through an Iterative Extended Kalman Filter (IEKS) are shown in Figure 3.5. The statistical evaluation is shown in Table 3.5, where the R^2 values do not show a sufficient fit (where 1 represents a perfect fit) due to the presence of noise in the measured data, while the NRMSE values (where 0 represents the best match) can be regarded as sufficient.

Table 3.5 Quantitative statistics for model validation. Averaged R^2 , RMSE and NRMSE for force and moment predictions of the linear model are depicted. The reported values confirm the validity of the aerodynamic model.

Identification				Validation			
Coef	X	Z	M	Coef	X	Z	M
R^2	0.868	0.616	0.422	R^2	0.848	0.665	0.1359
RMSE	3.109	19.779	1.0899	RMSE	4.10	20.63	1.239
NRMSE	0.071	0.067	0.045	NRMSE	0.078	0.067	0.053

Figure 3.7 illustrates the forward simulation results, obtained by applying the developed force and moment model to a linearized rigid body model of the aircraft. The initial state and control input time histories are retrieved from measured flight data and applied to the model. The corresponding statistical metrics are provided in Table 3.6, where TIC and GOF indicators are used (a TIC value of 0 and a GOF value of 1 indicate the best

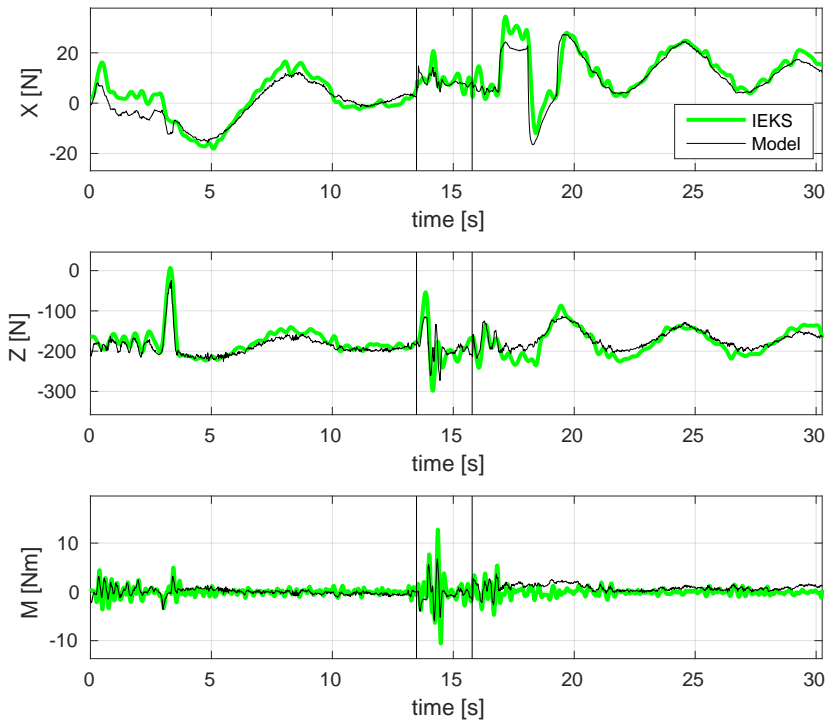


Figure 3.6 Measured versus modeled aircraft responses in terms of forces and moments for different maneuvers (sequence: phugoid, short period, thrust doublet). The results confirm that the aerodynamic forces and moments are accurately predicted.

result). Altogether a good match is observed, where larger discrepancies were found for the thrust variation maneuvers, which deteriorate the overall match. This might be due to the simplified thrust representation as linear and undelayed term in the force polynomial. Another reason might be the large throttle input used during the maneuver, which might have driven the system out of the linear region. An analysis without the thrust variation maneuver has led to TIC values below 0.3.

Another requirement for the model is its suitability for controller synthesis. An optimization-based tuning method has therefore been applied to the nonlinear simulation model from previous section with the newly identified linear derivative aerodynamics model (instead of the VLM-based aerodynamics model). The tuning procedure will be described in detail in Section 3.2.3. The designed aircraft controller has been tested in a number of flight experiments and continuously exhibited good flight performance without the need for re-tuning. The final practical validation of the modeling approach was thus achieved.

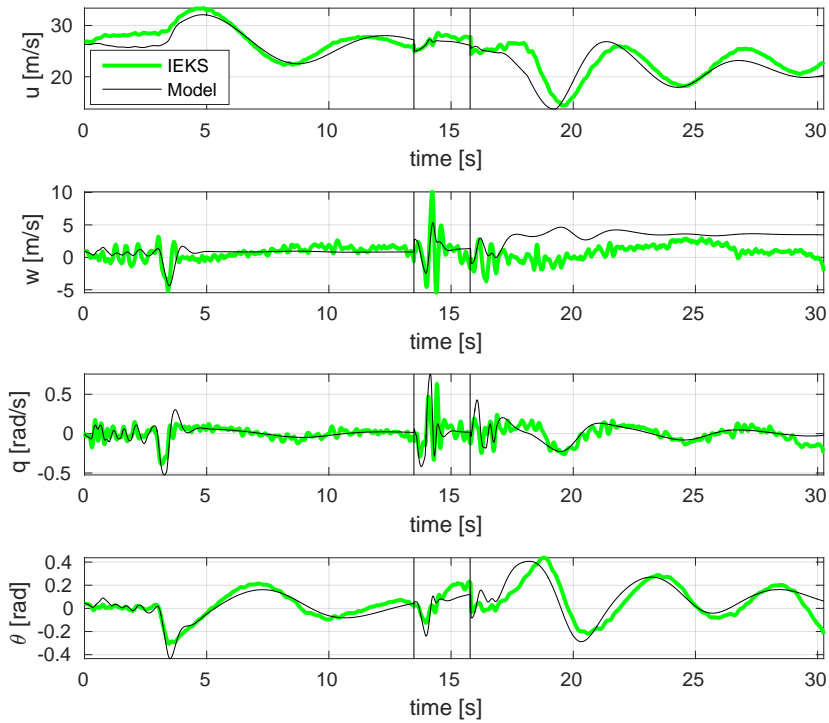


Figure 3.7 Measured versus simulated (linear model) aircraft states for different maneuvers (sequence: phugoid, short period, thrust doublet). The results are computed using forward simulation without the use of any measurements. As depicted, the results confirm that the identified aircraft model accurately explains the measured trajectory.

Table 3.6 TIC and GOF metrics for forward simulation with linear aircraft model. Both the results from identification and validation data sets are computed. Lower values indicate a better fit for TIC, while higher values are better for GOF.

Identification					Validation				
State	u	w	q	θ	State	u	w	q	θ
GOF	0.77	0.72	0.514	0.688	GOF	0.76	0.52	0.52	0.72
TIC	0.03	0.28	0.355	0.254	TIC	0.038	0.453	0.345	0.236

3.2 Aircraft Flight Control

As was introduced in Chapter 2.4.1, the cooperative landing maneuver imposes certain requirements on the aircraft control system, as specified in Table 2.4. From the start it was clear that a classical control approach using three separate SISO loops for altitude, airspeed and course control would not be capable to fulfill the given requirements. Especially *UAV-R6* in Table 2.4 poses a challenging problem for traditional SISO control approaches, since decoupled control of airspeed and altitude can only be achieved through coordinated control inputs of elevator and throttle. Such coordination can only be provided by means of a MIMO control system, thus calling for an appropriate control concept. A flight control strategy based on energy principles and known as *Total Energy Control System (TECS)* is capable of providing such functionality [74]. In parallel to the development of the new energy-based flight controller, a classical SISO-based controller [46] was developed to provide a performance reference. The following sections explain the basic theory of the two control approaches and provide a qualitative performance comparison of the new energy-based control strategy and the classical approach.

3.2.1 Classical vs Energy-Based Flight Control

Conventional flight control systems consist of several single-input, single-output (SISO) control loops, where the engine is typically used for speed control and the elevator for altitude control. Figure 3.8 illustrates such a classical control structure for altitude control.

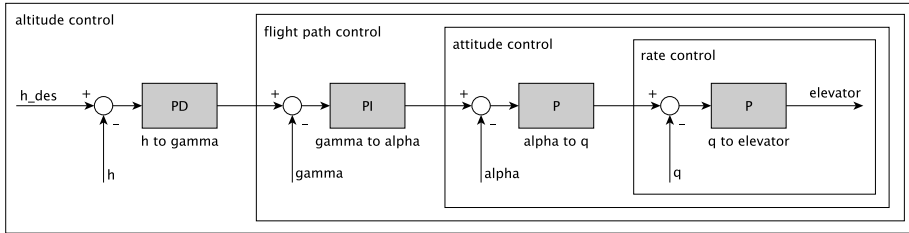


Figure 3.8 Block diagram of a classical longitudinal flight controller. The controller consists of four cascaded loops to control altitude, flight path angle, angle of attack and pitch rate.

A separate SISO-controller is used to control airspeed via the engine, typically consisting of a single PI-loop with a feedforward term:

$$\delta_t = \delta_{t_{FF}} + (K_{V_a} + \frac{1}{s} K_{iV_a}) \cdot (V_{des} - V_a) \quad (3.23)$$

with throttle command δ_t , feed-forward term $\delta_{t_{FF}}$, proportional gain K_{V_a} , integral gain K_{iV_a} , commanded airspeed V_{des} , and current airspeed V_a .

However, this approach does not account for the strong inherent coupling in altitude and speed responses of elevator and thrust commands. Furthermore, large elevator deflections

may drive the airspeed outside the permissible range, due to the limited control authority of the speed control system.

To overcome these limitations of conventional SISO controllers, an improved flight controller, considering energy principles as proposed in [74], [76], and [60], was implemented. Such controller is commonly referred to as total energy control system (TECS) in literature. It consists of an integrated control system using both throttle and elevator in a coordinated manner to decouple altitude and airspeed responses, and does not require a precise model of the aircraft. As explained in Chapter 2.4.5, such decoupling is crucial for the landing maneuver, given the close proximity operations of aircraft and ground vehicle in the final phase of the landing. The basic idea is to use throttle commands to control the aircraft's total energy, i.e. the sum of potential and kinetic energy, and elevator commands to control the distribution between potential and kinetic energy. In practice, the dimensionless energy rates $\gamma = \frac{\dot{h}}{V} = \frac{\dot{E}_{pot}}{mgV}$ and $\dot{V} = \frac{\dot{E}_{kin}}{mgV}$ are used, leading to the following core algorithm:

$$\frac{T_{cmd}}{mg} = K_{TI} \frac{1}{s} (e_\gamma + \frac{e_{\dot{V}}}{g}) - K_{TP} (\gamma + \frac{\dot{V}}{g}) \quad (3.24)$$

$$\theta_{cmd} = K_{EI} \frac{1}{s} (e_\gamma - \frac{e_{\dot{V}}}{g}) - K_{EP} (\gamma - \frac{\dot{V}}{g}) \quad (3.25)$$

where e_γ and $e_{\dot{V}}$ represent the flight path angle and acceleration errors respectively, while the controller outputs consist of the pitch angle command θ_{cmd} and the nondimensional thrust command $\frac{T_{cmd}}{mg}$. The TECS-core output variables are then processed by classical SISO inner loop controllers and transformed into elevator and throttle commands.

The TECS-core input variables γ and \dot{V} are calculated from the outer loop velocity and altitude controllers (in SISO configuration) as follows:

$$\dot{V}_{des} = K_V (V_{des} - V_a) \quad (3.26)$$

$$\dot{h}_{des} = K_h (h_{des} - h) \quad (3.27)$$

$$\gamma_{des} = \frac{\dot{h}_{des}}{V_k} \quad (3.28)$$

Figure 3.9 illustrates the overall longitudinal autopilot structure for the energy-based approach. The commands γ_{des} and \dot{V}_{des} must be limited with respect to the aircraft's operational flight envelope. Aircraft simulation model data is used to estimate drag and maximum available thrust as functions of airspeed, which are then used to limit the control inputs according to:

$$-\frac{D}{mg} < \frac{\dot{V}_{des}}{g}, \gamma_{des} < \frac{T_{max} - D}{mg} \quad (3.29)$$

It should be noted that γ_{des} and \dot{V}_{des} are being limited separately, which would in principle allow for their sum to lie outside of the aircraft performance bounds. In such case a pre-programmed priority logic inside the TECS-core will determine which command to follow. For the landing application it was decided to prioritize the airspeed command.

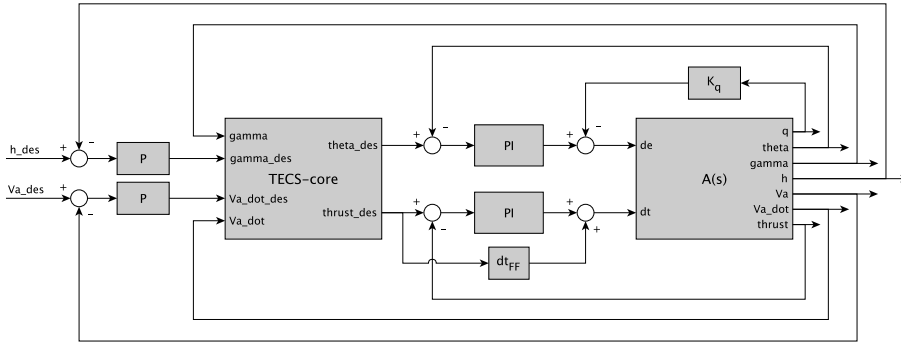


Figure 3.9 Energy-based longitudinal autopilot structure.

Command input rate limiters have also been implemented, as follows:

$$\frac{n_{\min}g}{V_k} < \frac{\ddot{V}_{des}}{g}, \dot{\gamma}_{des} < \frac{n_{\max}g}{V_k} \quad (3.30)$$

where $n_{\min}g$ and $n_{\max}g$ represent the maximum vertical acceleration limits. For manned aircraft these *g-force limits* are typically set to rather low values of around $\pm 0.1g$ to increase passenger comfort, while such low values are not required for unmanned aircraft. For the Penguin BE, the acceleration limits were set to $\pm 1g$.

Finally, stall and overspeed protection limits are implemented. Due to their criticality for flight safety these limits are checked last and are implemented as follows:

$$K_V (V_{\min} - V_a) < \dot{V} < K_V (V_{\max} - V_a) \quad (3.31)$$

where V_{\min} and V_{\max} define the admissible airspeed range, while K_V represents a tuning parameter. For the Penguin BE UAV V_{\min} was set to $V_{\text{stall}} + 2m/s$, while V_{\max} was set to $35m/s$. It should be noted that the stall speed V_{stall} depends on the current aircraft mass and bank angle. The stall speeds for different mass and bank angle combinations were obtained from the aircraft user manual and were implemented by means of a two-dimensional lookup table.

For the lateral-directional control of the aircraft, a conventional cascaded SISO controller using the ailerons as actuators was implemented as shown in Figure 3.10. Rudder control was not used, since the natural directional stability of the aircraft is sufficiently fast to drive the sideslip angle back to zero after a disturbance. As explained in Chapter 2.2, the whole landing maneuver is supposed to happen in a crabbed flight condition, where the aircraft flies in a slip-free condition, always pointing into the direction of the relative wind.

It should be noted that the first proportional controller block produces roll angle commands from a course error. This happens in two steps, where first a desired course rate command is calculated of the form:

$$\dot{\chi}_{des} = K_{\chi} \cdot \chi_{err} \quad (3.32)$$

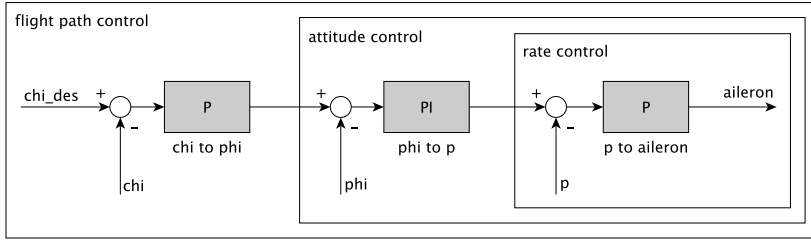


Figure 3.10 Block diagram of the lateral-directional flight controller. The controller consists of three cascaded loops to control course angle, bank angle and roll rate.

while the desired roll angle is then calculated from the course rate command by assuming force equilibrium in a coordinated turn as follows:

$$\phi_{des} = \arctan \left(\frac{V_k \dot{\chi}_{des}}{g} \right) \quad (3.33)$$

3.2.2 Waypoint Tracking Control

A simple waypoint tracking functionality was implemented for the aircraft. Its purpose consists in providing appropriate course commands to guide the aircraft along a pre-defined airfield pattern. Precise path following was not required for the landing application, since a switch-over to the landing controller is performed at a sufficient distance from the cooperative maneuver start point. The algorithm calculates a desired course command by subtracting the current aircraft position from the active waypoint position and calculating the course angle of the relative position vector. As the aircraft approaches the active waypoint, the next waypoint is activated at a circular switching distance of 50m around the currently active waypoint. During waypoint control altitude and speed commands can be provided independently of the current course commands. As will be explained in Chapter 4.2.6 the altitude and speed commands will be set in dependence on the current aircraft position with respect to the pre-defined waypoints.

3.2.3 Controller Tuning

The longitudinal and lateral aircraft controllers developed throughout the last section each consist of a set of cascaded control loops, which can be tuned iteratively by means of successive loop closure [10]. To allow for such approach, it is required that each successive loop is 5 to 10 times slower than its predecessor (starting from the innermost loop), which makes the inner loop appear as unity gain during the design of the outer loop. This principle is also known as *bandwidth separation*. The aircraft inner loops were thereby successively tuned via pole placement and heuristic fine-tuning, while considering sufficient bandwidth separation between loops. The middle-stage of the longitudinal aircraft flight controller contains the energy-based controller module, which provides reference values to the inner

loops. It is mostly model independent and does not require much tuning, since its two proportional gains are physically predetermined to the value of 1, and the remaining two integral gains must have the same value to achieve correct dynamical decoupling of speed and altitude responses [74]. This leaves the outer loops to be designed via pole placement and subsequent manual fine-tuning. Once a flyable simulation model is available as *initial guess*, an optimization-based tuning approach is applied, in accordance to the method described in Chapter 4.4.2. The method is based on performing time domain simulations using the controlled nonlinear aircraft model. The aircraft is then exposed to setpoint changes (steps in speed, altitude, and course angle) and load disturbances (wind gusts) and the responses are recorded. A cost function is assembled based on different performance metrics for each input disturbance — in the case of a setpoint change the following terms are used:

$$f(t_r, OS_{max}, |\dot{u}|, ITAE) = w_1 t_r + w_2 OS_{max} + w_3 |\dot{u}| + w_4 ITAE \quad (3.34)$$

where t_r represents the rise time, OS_{max} the maximum overshoot, $|\dot{u}|$ the maximum rate of change in the control inputs, and $ITAE$ the integral time absolute error. The cost function contributions for the disturbance input consist of slightly different terms, and e.g. do not include rise time, but instead use the settling time. Each term in the cost function is normalized by dividing the cost of each optimization run by the initial (and highest) cost the respective terms produces. In such way it is easy to judge the success of the cost minimization procedure, since only values from 0 to 1 are possible. This normalization also allows for choosing equal weights w_1 to w_4 to allow for equal contributions of all terms. If not normalized, the weights would have to be adapted to the magnitudes of the different terms to yield equal contributions (e.g. rise time typically produces much lower magnitudes than $ITAE$).

A nonlinear constrained optimization algorithm (Nelder-Mead Method [72]) then iteratively varies the different controller gains until the minimal cost is found. The normalization of the cost function inherently leads to matching dynamics in terms of velocity and altitude responses, which are required for a correct dynamical decoupling of the aircraft longitudinal motion via the energy-based flight controller. Additionally, by considering both setpoint changes as well as load disturbances, a balanced tuning result is obtained. It should be noted that the aircraft flight controller possesses more than 15 tunable parameters, which are all simultaneously adjusted by the optimization-based approach. This tuning strategy has led to good in-flight performance with good stability margins, which could not be significantly improved by manual fine-tuning. Table 3.7 contains the aircraft controller gains for the identified high-fidelity model from Section 3.1.6, which were successfully used during the flight experiments.

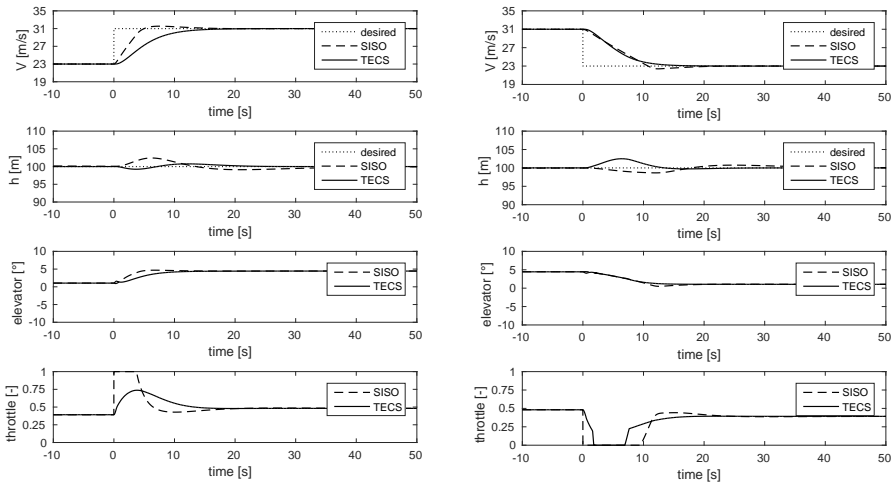
3.2.4 Qualitative Performance Analysis

A qualitative analysis has been performed by means of simulation and flight experiments for both control approaches. The controllers were provided with the same set of flight maneuvers. Figure 3.11 shows two exemplary simulation results for a speed-up and speed-down maneuver at constant commanded altitude, while Figure 3.12 depicts climb and

Table 3.7 Demonstrator aircraft *Penguin BE UAV* flight controller gains after optimization-based tuning.

Parameter	Value	Parameter	Value
K_θ	0.50161	K_p	0.03508
K_τ	0.071775	K_χ	0.74356
$K_{i\tau}$	0.00020063	$K_{i\chi}$	0.034534
δ_{FF}	1.1546	K_{TP}	1
K_{Va}	0.36774	K_{TI}	2
K_ϕ	5.5572	K_{EP}	1
$K_{i\phi}$	0.050287	K_{EI}	2

descent maneuvers at constant commanded airspeed.

**Figure 3.11** Simulation of 8m/s speed-up (left) and speed-down (right) maneuvers at constant commanded altitude for the classical (SISO) and energy based (TECS) flight controllers.

As can be seen on the left side in Figure 3.11 the classical SISO controller, which uses throttle to control airspeed, commands an aggressive throttle increase which leads to an altitude error of around 2.5m, before the altitude controller loop is able to compensate for the undesired cross-coupling. The TECS controller on the other hand is capable of coordinating the maneuver nicely (as can be seen in the elevator and throttle plots), where the altitude error remains below 1m.

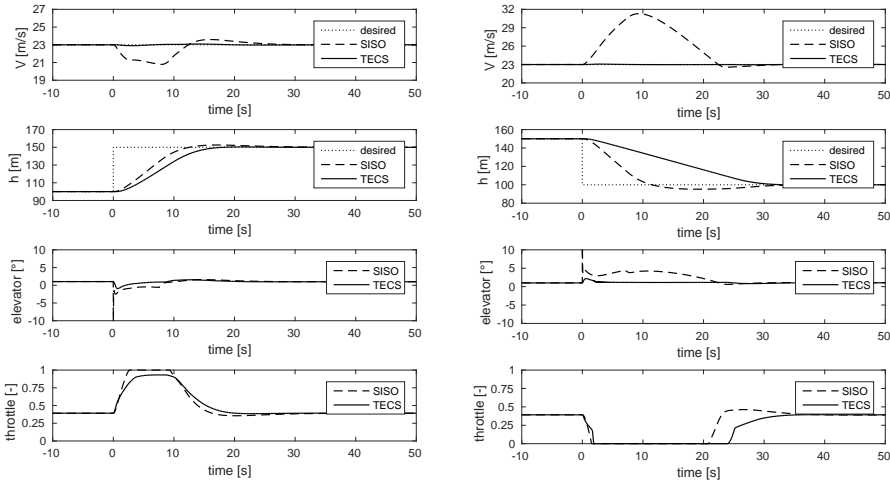


Figure 3.12 Simulation of 50m climb (left) and descent (right) maneuvers at constant commanded airspeed for the classical (SISO) and energy based (TECS) flight controllers.

On the right side of Figure 3.11 the speed-down maneuver is shown. The SISO case shows similar behavior as seen for the speed-up maneuver, where an abrupt throttle command is given, leading to an altitude error of around 1.5m. This error has time to develop, since it can only be compensated for by the altitude controller, once the it becomes visible in the altitude measurement (multiple integration steps later). The performance of the TECS controller at first looks worse than the SISO case, where an altitude error is around 2.5m. After closer investigation it shows that this deviation in altitude is not due to poor controller performance, but to the limit-thrust operation logic included in the TECS controller. As previously mentioned, the TECS controller automatically switches to *speed-via-elevator* mode once the throttle command is saturated (only elevator is left) to control airspeed, while ignoring vertical flight path control. As can be seen on the right side of Figure 3.11 the throttle command reaches zero after a few seconds for the TECS controller, which leads to the immediate use of elevator to follow the speed-down command by pulling the aircraft nose up. The altitude error is decreased once the target airspeed is reached. It should be noted that the velocity step of 8m/s commanded here is rather large, and was chosen this way to illustrate the functionality of the control concept. During the landing experiments much smaller speed variations will be present, therefore expected altitude variation during speed-only maneuvers are much smaller. Also, this behavior is needed to guarantee flight safety, where an adherence to the admissible speed range is absolutely necessary.

Figure 3.12 illustrates exemplary climb and descent maneuvers. As can be seen on the left, the SISO controller is not able to maintain airspeed during the climb maneuver, even though full throttle is commanded. A speed error of around 2m is observed for the

SISO case. The energy-based controller on the other hand reduces the flight path angle as required to maintain airspeed, where practically no error in airspeed is visible. On the right side of Figure 3.12 a descent maneuver is shown, which clearly illustrated the limitations of the SISO control approach. As the elevator controls altitude, the aircraft is commanded to reduce its flight path angle, which naturally leads to a speed increase. The throttle-controlled speed controller then reduces the throttle command to counter the measured speed increase, but reaches a zero throttle command quickly. Once throttle is saturated at zero, the aircraft continues to accelerate since the altitude controller continues to decrease the flight path command. Such situation is very dangerous and could easily drive the aircraft above its structural limits. The energy-based controller on the other hand precisely maintains the airspeed at its commanded value during the whole descent. It implicitly adjusts the flight path angle limit by means of the speed-priority logic during thrust-limit operations, which leads to a safe descent at constant commanded airspeed and minimum possible flight path angle.

The lateral-directional flight controller was not further analyzed here, as it did not present any significant issues regarding its application for the landing maneuver.

After completing the simulation-based performance analysis, the both flight control systems were tested in a number of flight experiments. Only the results for the newly implemented energy-based flight controller will be shown in the following to reduce the size of the present document. It should be noted though that the effects seen during the simulation experiments for the SISO controller were observed in the same way during the flight experiments.

In straight and level flight, the controller was able to keep the altitude within $\pm 1m$ and the airspeed within $\pm 1.5m/s$ of the desired value, as shown in Figure 3.13.

Steps in airspeed and altitude were followed with only small overshoots, and with no coupling effect visible in the other variable. Figure 3.14 shows an exemplary climb and descent maneuver, while Figure 3.15 and Figure 3.16 show different step commands in airspeed.

3.3 Ground Vehicle Dynamics Model

The simulation model for the ground vehicle consists of a kinematic bicycle model for the lateral direction and a dynamic model in the longitudinal direction (with respect to body frame) [67, 7]. This model assumes slip-free motion (no slippage between tires and road) and approximates the front and rear wheel pairs as single wheels, which is considered a valid approximation for the rather smooth ground vehicle motion during the landing application.

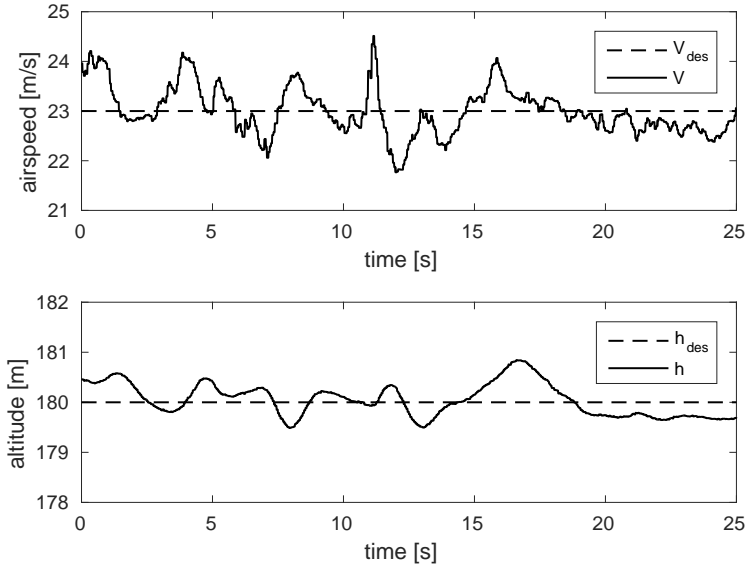


Figure 3.13 Horizontal flight with the energy based flight controller. The altitude error remains within $\pm 1m$, the airspeed error within $\pm 1.5m/s$ (commanded values are displayed as dashed lines, while actual sensor data is represented as a solid line).

The kinematic bicycle model with reference to an inertial frame is given in Equation (3.35):

$$\dot{x} = v \cos(\psi + \lambda) \quad (3.35)$$

$$\dot{y} = v \sin(\psi + \lambda) \quad (3.36)$$

$$\dot{\psi} = \frac{v}{l_r} \sin(\lambda) \quad (3.37)$$

$$\dot{v} = a \quad (3.38)$$

$$\lambda = \tan^{-1} \left(\frac{l_r}{l_f + l_r} \tan(\delta) \right) \quad (3.39)$$

where x and y represent the positions of the ground vehicle center of mass in the inertial frame, v the ground speed of the center of mass, l_r and l_f the distances from the center of mass to the rear and front wheel respectively, ψ the yaw angle and λ the angle between the velocity vector and the ground vehicle's body frame x-axis. This model assumes the acceleration a and the front wheel steering angle δ as inputs.

As acceleration is usually not a directly available input in real vehicles, this model is extended with a dynamic model for the longitudinal motion, considering the external

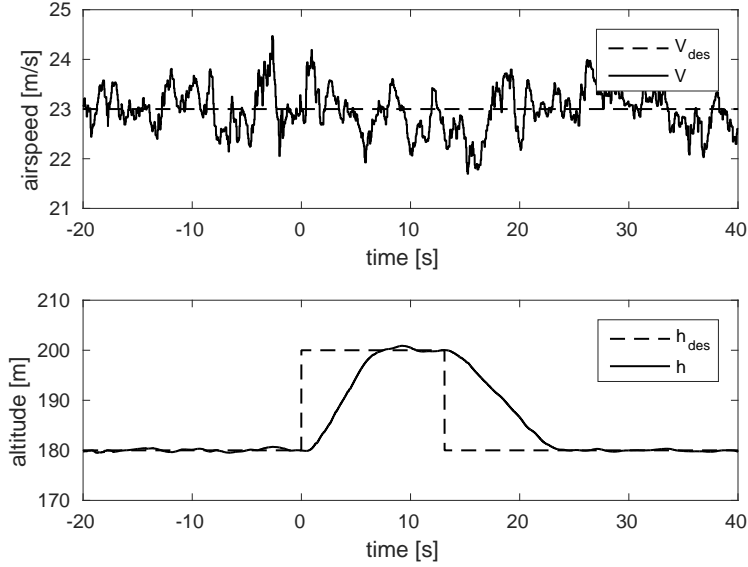


Figure 3.14 Climb and descent with the energy-based flight controller. Both maneuvers are nicely coordinated. The airspeed tracking performance is the virtually same as observed during level flight (commanded values are displayed as dashed lines, while actual sensor data is represented as a solid line).

forces acting on the vehicles as shown in Equation (3.40):

$$m \frac{dv}{dt} = F_{drive} + F_{aero} + F_{fric} + F_g \quad (3.40)$$

where F_{drive} represents the drive force, F_{aero} the aerodynamic drag force, F_{fric} the friction force, and F_g the gravity force. This leads to the following expression:

$$\frac{dv}{dt} = \frac{\alpha_n T (\alpha_n v) \delta_\tau - \frac{1}{2} \rho C_d A v |v| - mg \sin(\theta) - mg C_r \text{sgn}(v)}{m} \quad (3.41)$$

where T is the engine torque (at full throttle), α_n the gear ratio, v the velocity in forward direction, δ_τ the throttle setting, ρ the air density, C_d the drag coefficient, A the front projected area of the ground vehicle, m the ground vehicle mass, g the gravitational acceleration, θ represents the ground slope, C_r the rolling friction coefficient, and $\text{sgn}()$ the sign function.

The engine torque is modeled as:

$$T(\omega) = T_m \left(1 - \beta \left(\frac{\omega}{\omega_m} - 1 \right)^2 \right) \quad (3.42)$$

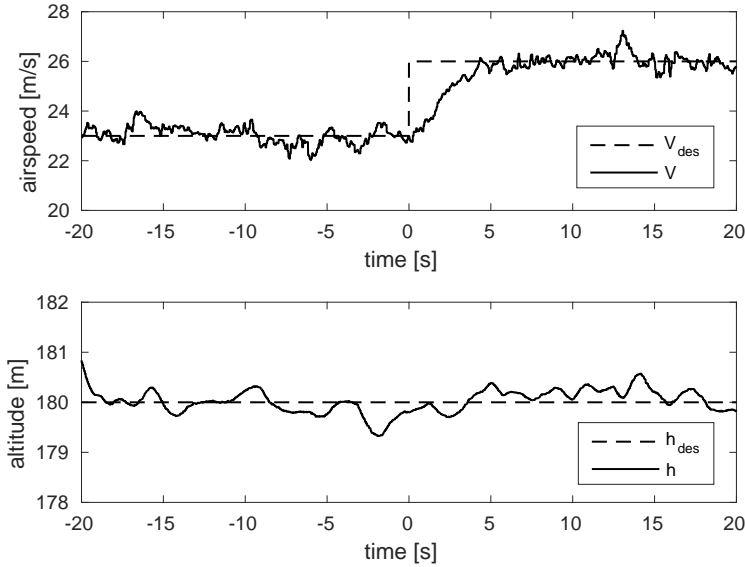


Figure 3.15 A 3m/s speed increase with the energy-based flight controller. The altitude tracking performance is virtually the same as observed during level flight, which validates the correct coordination of elevator and throttle (commanded values are displayed as dashed lines, while actual sensor data is represented as a solid line).

where $T(\omega)$ corresponds to the engine torque at full throttle, ω to the engine angular velocity, and T_m to the maximum torque at maximum engine rotational speed ω_m . These parameters are obtained from the Audi A6 Avant 2.5 TDI data sheet and are shown in Table 3.8. The parameter β is a generic scaling factor and has been set to 0.4 in accordance to [7]. Since the final experiments have been performed in *gear 3* of the automatic transmission system, the drive force is modeled with α_3 only.

With this model, the inputs are the throttle setting and the front wheel steering angle, in accordance to the available control inputs in a regular car. Additionally the front wheel steering angle can be expressed in terms of the steering wheel deflection angle, by means of the steering ratio, which in case of the demonstrator vehicle has a value of 16.2. The brakes have been modeled as negative engine acceleration of equal magnitude, which is considered a reasonable approximation for the given application. The actuation system was modeled as first order low pass filter, as will be described in more detail in Chapter 5.

Table 3.8 shows the technical specifications of the demonstrator ground vehicle.

The aerodynamic disturbance caused by the ground vehicle was identified as an important issue and needed to be analyzed in detail. Therefore a numerical CFD simulation (Computational Fluid Dynamics) has been performed using realistic geometrical models of ground vehicle and landing platform as shown in Figure 3.17. The airflow characteristics have been calculated in three dimensions in order to capture the highest possible

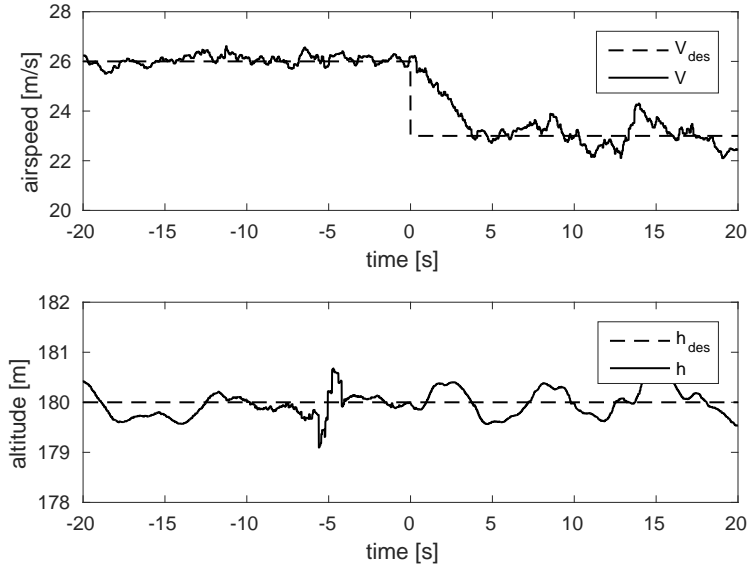


Figure 3.16 A 3m/s speed decrease with the energy based flight controller. Again, good altitude and airspeed tracking performance is observed (commanded values are displayed as dashed lines, while actual sensor data is represented as a solid line).

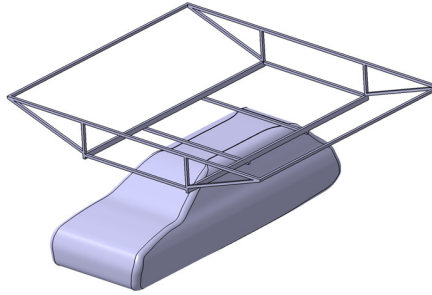


Figure 3.17 Ground vehicle and landing platform model as used in CFD simulations.

detail of the present airflow conditions. The goals of the CFD simulation consisted in the determination of the change of airflow velocity and angle of attack induced by the presence of the UGV (velocity vector magnitude and direction at a specific point in space), as well as the optimal landing trajectory.

Figure 3.18 illustrates the resulting flow situation as well as the selected final landing trajectory. A vertical descent above the middle area of the landing platform induces the smallest variations in airflow magnitude and direction.

Table 3.8 Technical specifications for the demonstrator ground vehicle Audi A6 Avant 2.5 TDI (2000) .

Parameter	Value
max. engine torque	268.1Nm
max. engine speed	460rad/s
mass	1900Kg
max. velocity	219Km/h
gear ratios	[3.6651.9991.40710.742]
axle Ratio	2.984
wheel diameter	631.9mm
l_r	1380mm
l_f	1380mm
wheelbase	2760mm
steering ratio	16.2
drag coefficient	0.32
front projected area	2.19m ²
air density	0.32

Figure 3.19 shows the distribution of both, the induced angle of attack (black lines) and the induced relative velocity (red lines) in the longitudinal and lateral direction of the ground vehicle at 1m above landing platform. These effects have been incorporated into the overall real-time simulation environment as will be described in Section 5.1. A detailed analysis of the results and their impact on the landing system design can be found in [92].

3.4 Ground Vehicle Control

Early simulations have shown that relatively slow ground vehicle dynamics are sufficient for the execution of a successful cooperative landing maneuver. The choice was therefore taken in favor of a manned ‘semi-autonomous’ ground vehicle. In this solution a human driver has to execute the control commands provided by the ground vehicle controller. The human basically replaces the mechanical actuator or driving robot. A Graphical User Interface (GUI) mounted inside the car is used for displaying the corrective action the *human actuator* has to perform. The setup is shown in Figure 3.20.

The GUI has been designed as a two-colored crosshair with the horizontal line indicating throttle commands and the vertical line representing steering wheel deflection commands.

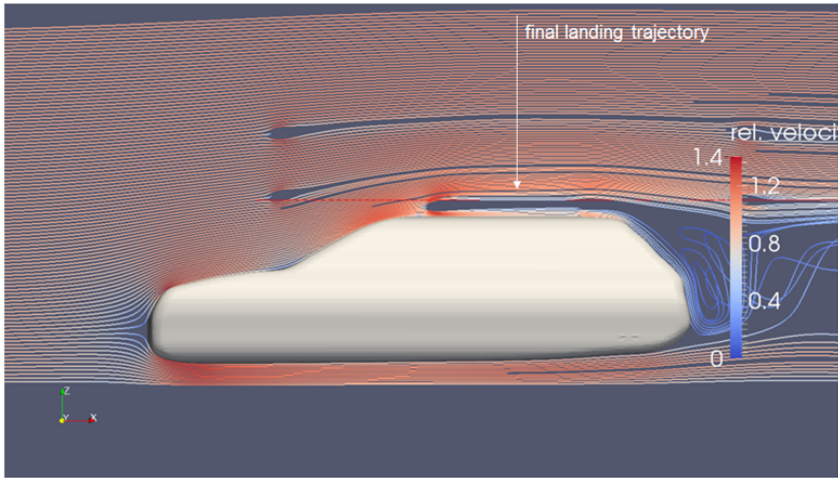


Figure 3.18 Visualization of disturbed flow around the ground vehicle and optimal landing trajectory.

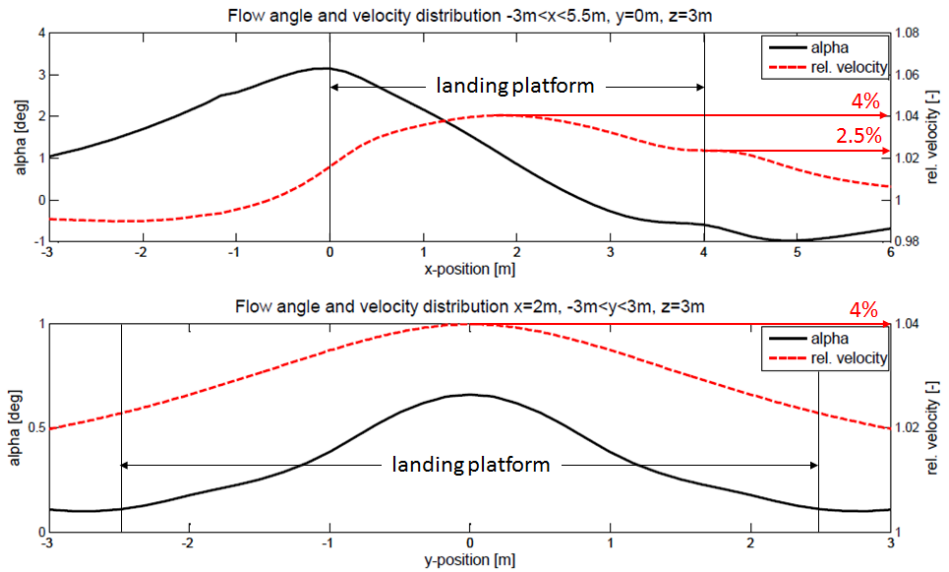


Figure 3.19 Longitudinal (x) and lateral (y) distribution of induced angle of attack (black) and relative velocity (red) at 1m above landing platform.

Both the ground vehicle velocity and course controllers were implemented as proportional controllers, directly translating velocity / course command into throttle / steering wheel deflection commands, with the human driver providing inherent integrating behavior. These controllers allow for the adjustment of the closed-loop ground vehicle dynamics



Figure 3.20 Graphical User Interface (GUI) for human driver commands.

despite the human driver. For the simulation model, integral action had to be added to the ground vehicle speed controller. The ground vehicle has been equipped with the same real-time computer and sensor system (RTK GPS, IMU) as present on the aircraft. Data is exchanged via a wireless network. During first landing experiments with a simulated aircraft, as will be described in detail in Section 6.2, the human driver was capable of following the given throttle commands reasonably well. As shown in Figure 3.21, the actual ground vehicle velocity follows the desired velocity, where the maximum deviation during the last 15s before touchdown remains within a bound of around $0.6 \frac{m}{s}$.

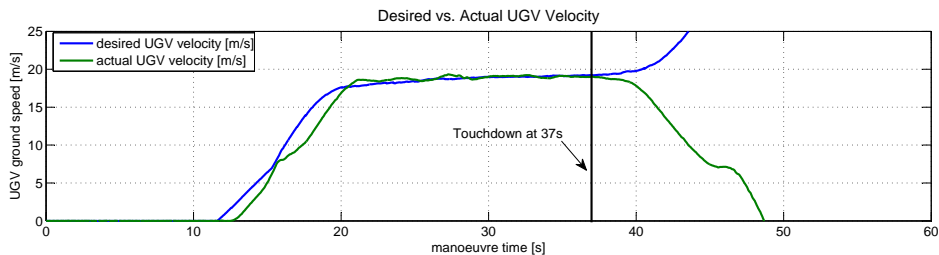


Figure 3.21 Ground vehicle desired vs. actual velocity during a virtual landing experiment (real ground vehicle, virtual aircraft).

The delay in the beginning of the acceleration phase at around 12s into the landing maneuver is due to the delay introduced by the ground vehicle's automatic transmission and the human reaction time. The notch in actual UGV velocity at around 15s is due to the automatic transmission shifting gears. At around 38s, only around one second after the virtual touchdown, the ground vehicle reached the end of the available runway and had to decelerate. This led to an increase in desired UGV velocity, due to the fact that the aircraft simulation was not stopped at touchdown.

For simulation purposes, and for the analysis of future autonomous ground vehicles, a simple ground vehicle autopilot system was developed. Two independent SISO controllers were used for each motion direction. For the longitudinal direction a simple PI-controller

Table 3.9 Demonstrator ground vehicle *Audi A6* controller gains after optimization-based tuning.

Parameter	Value
K_T	0.7894
K_{IT}	0.1084
K_χ	0.895

was sufficient to accurately control ground speed, while for the lateral direction a P-controller was sufficient for accurate course control (higher system type). For controller design, a similar method as will be described in Chapter 4.4.2 for the cooperative position synchronization controller, was applied. Thereby pole placement is used for course pre-tuning based on a desired time domain response, while a nonlinear optimization-based tuning approach is applied in a second step. Thereby the response of the controlled nonlinear vehicle model to a step in reference value and a load disturbance is recorded and a cost function is constructed from different time domain performance metrics (e.g. maximum overshoot, rise time, ITAE). These inputs have been designed in a way to provide a balanced response, where reference tracking and disturbance rejection are considered equally. Then the gains are automatically varied by the optimization algorithm until a minimal cost is found. Table 3.9 depicts the controller gains for the demonstrator ground vehicle.

3.5 Linear Vehicle Models

Throughout this chapter the nonlinear aircraft and ground vehicles models have been derived. For analysis and control design purposes is useful to work with linear models, since linear control theory is well understood and a large variety of tools are available. The linear aircraft model has been obtained though numerical linearization, while the ground vehicle models was derived analytically.

3.5.1 Linear Aircraft Model

The linearization of the aircraft dynamics is performed at steady state flight, where the forces and moments acting on the aircraft rigid body are at equilibrium. When trimmed for straight and level horizontal flight, a subset of the 12 inertial aircraft states $x = (p_n, p_e, p_d, u, v, w, \phi, \theta, \psi, p, q, r)^\top$, as previously defined in Equations (3.5) to (3.8), are constant or zero:

$$\begin{aligned}
 \text{Linear velocities} \quad & u, v, w = \text{const.} \\
 \text{Euler angles} \quad & \phi, \theta, \psi = \text{const.} \\
 \text{Angular rates} \quad & p = q = r = 0 = \text{const.}
 \end{aligned} \tag{3.43}$$

for a corresponding constant input vector $u^* = (\delta_e^*, \delta_\tau^*, \delta_a^*, \delta_r^*)^\top$, where the superscript (*) indicates trim values.

As already mentioned in Chapter 2.4.5, the aircraft dynamics are often treated as decoupled in longitudinal and lateral direction (with respect to the aircraft body frame) in the proximity of a trim condition. This implies decoupled external forces and moments (aerodynamics and propulsion), as well as decoupled rigid body equations of motion. Such decoupling is actually observed after performing the linearization, where the aircraft equations of motion can be split up into two independent equation systems as shown in Equation (3.44):

$$\begin{bmatrix} \dot{x}_{long} \\ \dot{x}_{lat} \end{bmatrix} = \begin{bmatrix} A_{long} & 0 \\ 0 & A_{lat} \end{bmatrix} \begin{bmatrix} x_{long} \\ x_{lat} \end{bmatrix} + \begin{bmatrix} B_{long} & 0 \\ 0 & B_{lat} \end{bmatrix} \begin{bmatrix} u_{long} \\ u_{lat} \end{bmatrix} \quad (3.44)$$

The equations are expressed in the aircraft body frame and are therefore decoupled with respect to the body frame vertical and horizontal planes. The linear state-space model for the demonstrator aircraft *Penguin BE* was derived through numerical linearization, by first trimming the nonlinear aircraft model for straight and level horizontal flight at 21m/s using the Matlab *trim* function, and subsequently applying the Matlab *linmod* function. This results in the following longitudinal model:

$$\begin{bmatrix} \dot{u}_a \\ \dot{w}_a \\ \dot{q}_a \\ \dot{\theta}_a \\ \dot{\omega}_a \end{bmatrix} = \begin{bmatrix} -0.10 & 0.39 & -1.4 & -9.8 & 0.006 \\ -0.64 & -3.6 & 22 & -0.6 & 0 \\ 0.19 & -2.8 & -5.6 & 0 & -0.001 \\ 0 & 0 & 1 & 0 & 0 \\ 21 & 1.2 & 0 & 0 & -2.6 \end{bmatrix} \begin{bmatrix} u_a \\ w_a \\ q_a \\ \theta_a \\ \omega_a \end{bmatrix} + \begin{bmatrix} 0.38 & 0 \\ -7.3 & 0 \\ -65 & 0 \\ 0 & 0 \\ 0 & 2027 \end{bmatrix} \begin{bmatrix} \delta_e \\ \delta_\tau \end{bmatrix} \quad (3.45)$$

with the longitudinal state vector $x_{lon} = (u_a, w_a, q_a, \theta_a, \omega_a)^\top$ and input vector $u_{lon} = (\delta_e, \delta_\tau)^\top$, where u_a and w_a represent the inertial velocity vector components in body x and z axes, q_a and θ_a the pitch rate and pitch angle around the body y-axis, and ω_a the engine's angular velocity. The longitudinal input vector includes the elevator deflection δ_e and throttle setting δ_τ . The index *a* indicates aircraft states, while the index *g* will be used for ground vehicle states. It should be noted that these states represent variations around the linearization trim point. The lateral state-space model is given by:

$$\begin{bmatrix} \dot{v}_a \\ \dot{p}_a \\ \dot{r}_a \\ \dot{\phi}_a \\ \dot{\psi}_a \end{bmatrix} = \begin{bmatrix} -0.28 & 1.9 & -21 & 9.8 & 0 \\ -0.40 & -14 & 2.5 & 0 & 0 \\ 1.3 & -2.1 & -1.2 & 0 & 0 \\ 0 & 1 & 0.08 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} v_a \\ p_a \\ r_a \\ \phi_a \\ \psi_a \end{bmatrix} + \begin{bmatrix} 0.69 & 4.5 \\ -131 & -3.9 \\ -19 & -23 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \delta_a \\ \delta_r \end{bmatrix} \quad (3.46)$$

with the lateral state vector $x_{lat} = (v_a, p_a, r_a, \phi_a, \psi_a)^\top$ and input vector $u_{lat} = (\delta_a, \delta_r)^\top$, where v_a represents the inertial velocity vector components along body y-axis, p_a and ϕ_a the roll rate and roll angle around the body x-axis, r_a and ψ_a the yaw rate and yaw angle around the body z-axis. The lateral input vector includes the aileron deflection δ_a and rudder deflection δ_r .

The poles of the aircraft are shown in Figure 3.22, with the longitudinal modes:

$$\begin{array}{ll} -4.61 \pm 7.86i & \text{short-period mode} \\ -0.03 \pm 0.54i & \text{phugoid mode} \\ -2.66 + 0.00i & \text{engine dynamics} \end{array} \quad (3.47)$$

and lateral modes:

$$\begin{array}{ll} -13.40 + 0.00i & \text{roll subsidence mode} \\ -0.96 \pm 5.19i & \text{dutch roll mode} \\ 0.11 + 0.00i & \text{spiral mode} \end{array} \quad (3.48)$$

The longitudinal mode at $-2.66 + 0.000i$ corresponds to the engine dynamics and was added here for completeness. It is common that many aircraft possess an unstable spiral mode, which is typically quite slow and easily stabilized by a pilot or autopilot.

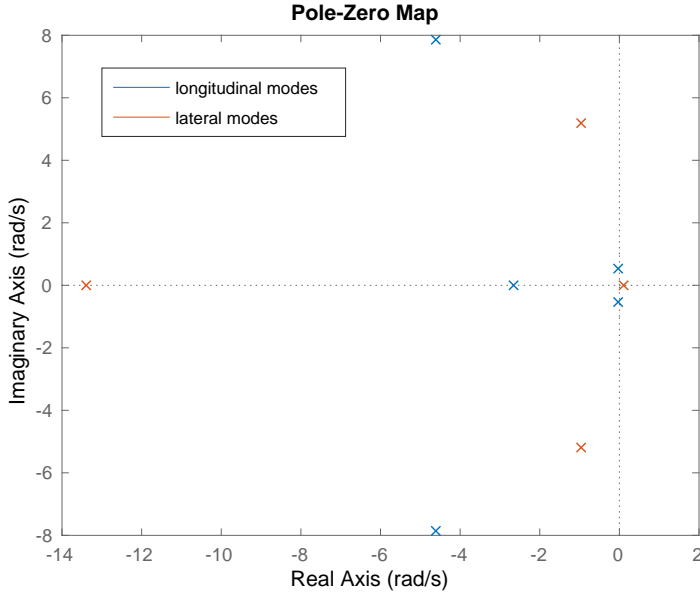


Figure 3.22 Aircraft dynamic modes.

3.5.2 Linear Ground Vehicle Model

As previously mentioned, the state-space model for the ground vehicle was linearized analytically by means of a multivariate Taylor expansion of order one. The partial derivatives for the system and input matrices are then evaluated at a given trim speed, while the vehicle parameters for the specific demonstrator ground vehicle (see Table 3.8) are

inserted. The trim conditions for the ground vehicle have been set as shown in Equation (3.49):

$$\begin{aligned} \text{Ground speed} & \quad v = 21 \text{ m/s} \\ \text{Yaw and slip angles} & \quad \psi, \lambda = 0 \\ \text{Yaw rate} & \quad r = 0 \end{aligned} \quad (3.49)$$

for a corresponding constant input vector $u^* = (f^*, \delta^*)^\top$, where the superscript (*) indicates trim values. The resulting state-space model for the ground vehicle is shown in Equation 3.50:

$$\begin{bmatrix} \dot{u}_g \\ \dot{x}_g \\ \dot{y}_g \\ \dot{\psi}_g \\ \dot{\lambda}_g \end{bmatrix} = \begin{bmatrix} -0.00781 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 21 & 21 \\ 0 & 0 & 0 & 0 & 15.22 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} u_g \\ x_g \\ y_g \\ \psi_g \\ \lambda_g \end{bmatrix} + \begin{bmatrix} 1.7590 & 0 \\ 0 & 0 \\ 0 & 10.5 \\ 0 & 7.6087 \\ 0 & 0.5 \end{bmatrix} \begin{bmatrix} f \\ \delta \end{bmatrix} \quad (3.50)$$

$$\begin{bmatrix} \dot{x}_g \\ x_g \\ y_g \\ \dot{y}_g \\ \lambda_g \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 21 & 21 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} u_g \\ x_g \\ y_g \\ \psi_g \\ \lambda_g \end{bmatrix} \quad (3.51)$$

where u_g represents the inertial velocity vector component along the body x-axis, x_g and y_g the position coordinates in runway x and y directions, ψ_g the yaw angle around the body z-axis, and λ_g the sideslip angle, which is the angle between the velocity vector of the ground vehicle's center of mass and the body x-axis. The ground vehicle possesses 4 poles at the origin and one pole at $-0.0078 + 0.0000i$.

Chapter Conclusions

Throughout this chapter the aircraft and ground vehicle simulation models were developed, as well as appropriate aircraft and ground vehicle control strategies. The tuning of the vehicle controllers was performed, and their performance analyzed in simulation and in flight experiments. The requirements defined in Chapters 2.4.1 and 2.5 were verified, and the vehicle models and controllers are now ready to be used for the development of the cooperative landing controller in the next chapter.

4 Landing Control System

Chapter Overview

Throughout this chapter the cooperative landing control system is developed according to the design choices introduced in Chapter 2.4. First the relative state estimation system is presented, which forms the basis for the cooperative landing control task. The relative state estimates are based on GPS position data, which is exchanged between the vehicles through a wireless network and fused with vision-based states estimates. Then the safety-based landing control system is developed and all its components presented. The safety-based controller is divided into approach and landing control functions which consist of multiple modes, where mode transitions are managed through a dedicated landing state machine. Next, the performance-based landing control system is presented, where the main differences consist in an optimization-based approach and a bilateral position synchronization controller. To guarantee stability of the bilateral control system, where communication time delays enter the feedback loops, an energy-based passivity controller is introduced. After all components of the safety- and performance-based landing controllers are presented, the control system parameters are designed by means of an optimization-based design method and a stability assessment is performed. Finally a sensitivity analysis of the landing controllers is performed in the frequency and time domain, where first performance bounds are estimated by means of a Monte Carlo simulation.

4.1 Relative State Estimation

Before any cooperative control task can be performed, the relative vehicle states must be known. The quality of the relative state estimates directly influences the achievable control precision. A number of different vision-based state estimation methods, including

incremental feature-based motion tracking, RANSAC based motion tracking, as well as techniques based on geometrical models [73] and optical marker tracking have been analyzed and tested. In the end a combination of GPS-based networked state estimation and optical multi marker tracking showed the best performance and was applied to increase the reliability of the relative state estimation system.

4.1.1 Networked State Estimation and Lag Compensation

Each vehicle's sensor data needs to be transmitted via a wireless network and fused with the other vehicle's local sensor data. In our case GPS position and velocity are exchanged together with the corresponding timestamp (GPS Time). This data needs to be synchronized and communication delays compensated for. For effective lag compensation, the amount of signal lag needs to be determined, which is done via the GPS timestamps. This calculation is performed on both vehicles, where the obsolete measurement received via the wireless network is subtracted from the local measurement. In the case of the aircraft the lag estimate is calculated as follows:

$$t_{lag,a} = t_{local,a} - t_{network,g} \quad (4.1)$$

where the indices a and g represent aircraft and ground vehicle related quantities respectively. The calculation is performed in the same way on the ground vehicle. A typical lag profile encountered during landing tests is illustrated in Figure 4.1.

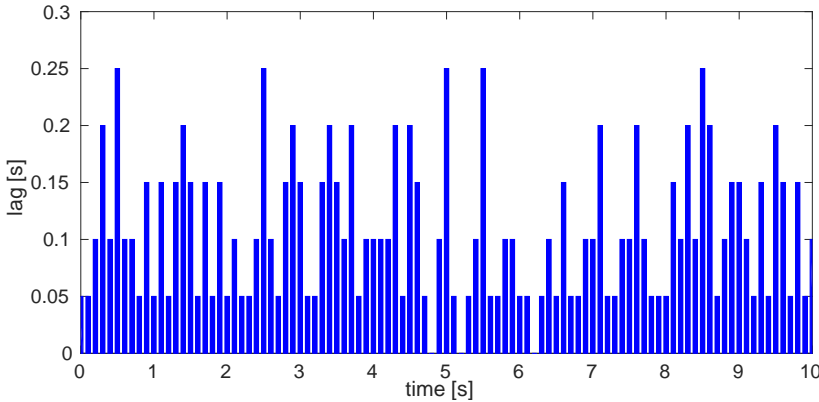


Figure 4.1 Typical lag profile during flight tests.

As can be seen in Figure 4.1, the signal lag estimates are only available in discrete steps of $0.05s$, which corresponds to the used GPS sensor's measurement frequency of $20Hz$. At a landing velocity of around $21 \frac{m}{s}$, this would result in a minimum value of $1.05m$ for the correction of the relative position. The signal lag typically ranges from $0.05s$ to $0.15s$ with occasional outliers at $0.2s$ and more. The main factors influencing the signal lag are the wireless data transmission and the increased amount of signal routing present in the system, as positioning data is exchanged between the two vehicles via the ground station.

Different fusion methods (including Kalman filters and buffer-based synchronization methods) have been analyzed and tested. In the end a very simple method showed the best performance results. In the following, the method will be explained for the aircraft side, but was implemented in the same way for the ground vehicle. The locally measured aircraft position is subtracted from an obsolete ground vehicle position received via the wireless network. The relative position is calculated and the signal lag corrected for, using the above described lag estimation together with the ground vehicle velocity that was measured together with its position (at the same moment in time):

$$\mathbf{x}_{rel}(t) = \mathbf{x}_g(t - t_{lag}) + \mathbf{v}_g(t - t_{lag}) \cdot t_{lag} - \mathbf{x}_a(t) \quad (4.2)$$

with:

$$\mathbf{x}(t) = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad \mathbf{v}(t) = \begin{bmatrix} u \\ v \\ w \end{bmatrix} \quad (4.3)$$

The reference frame (local runway frame) used to describe the above quantities is pointing in runway direction with its x-axis, the y-axis is pointing to the right and the z-axis downwards. Due to the lack of synchronized acceleration measurements, lag compensation was not implemented for the calculation of relative velocities. Here it was assumed that the respective vehicle's velocity did not change from its previous value when lag occurred.

4.1.2 Optical Marker Tracking

The relative state estimation method described in the previous section is based on data exchange via a wireless network. As previously mentioned, this induces significant signal lag which needs to be compensated for. Also, the lag compensation method proposed in Section 4.1.1 is only valid for smaller deviations from the linear forward motion, as obsolete velocities are used for extrapolation. A virtually lag-free optical multi marker detection and localization method was therefore developed and implemented to further increase the robustness and precision of the target estimation [91]. These directly measured relative state estimates are then fused with the networked GPS-based estimates as will be described in Section 4.1.3. This leads to a more reliable system, which is less sensitive to a degraded GPS quality or complete sensor outages.

Fiducial markers are a good tool to detect a known object and localize it relative to a monocular camera. The open-source software suite ARToolkitPlus [124] was chosen for object detection and localization. This algorithm was chosen because it can run in real time on the on-board image processing computer. The method also has a very low false positive detection rate because a checksum is included in the binary pattern used for the markers. This is very important for the reliability and safety of the automatic landing system. A problem can occur when the marker is in frontal direction of the camera. In this position the orientation estimate gets unstable for the angles around the camera x and y axis (z axis pointing in the viewing direction) [112]. This limitation does not affect our application because the unstable angles are not needed for the landing algorithm. Only the angle of the marker around the camera z axis would be needed to align the aircraft heading

to the landing platform on the ground vehicle. Another drawback of this method is that a partly occluded marker might not be detected and localized. To overcome this drawback multiple markers are used. A reference marker is specified and the measurements are provided with respect to this marker. If the position of the reference marker together with one or more other marker is measured, the relative position and orientation with respect to the reference marker will be calculated and stored. If the markers are measured multiple times the measurements will be averaged. If the new measurement differs from the stored measurement by more than a specified value (in this case 5cm), the old measurements will be discarded and only the new measurement is used. If multiple known markers and not the reference marker are in the image a RANSAC algorithm is used to eliminate possible outliers. After the RANSAC is performed the measurements are averaged. Figure 4.2 shows the estimated reference marker pose (large marker in the middle of the roof) and the transformation to a pre-defined point of interest (here the GPS antenna location) during an exemplary landing attempt.

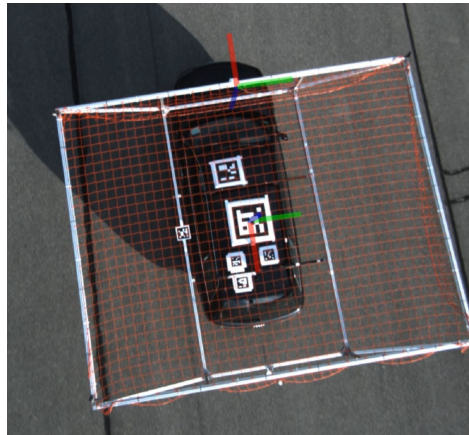


Figure 4.2 Reference marker pose estimate and transformation to desired reference point at GPS antenna mounting position.

As can be seen in Figure 4.2, the vision-based estimation of the GPS antenna location corresponds well to the actual sensor location (white antenna in the middle of the front crossbar). The advantages of this algorithm are a more reliable optical target detection and possibly also higher accuracy of the measured position of the reference marker because multiple measurements are fused. Figure 4.3 shows a comparison of the relative position measurements using two RTK-GPS receivers and the vision-based multiple marker system.

The vision measurements are more accurate the closer the camera gets to the marker, while accuracy also depends on the viewing angle. The detection of the marker is possible from a distance of up to 16m when the target is in the field of view. Between 10s and 15s the target is not inside the camera field of view so no measurements are available. Below 10m relative altitude the target is reliably detected by the vision system. The relative measurements of the two systems are very close during the critical landing phase below 5m . The vision system cannot detect the marker during the last phase below 1m , as can be

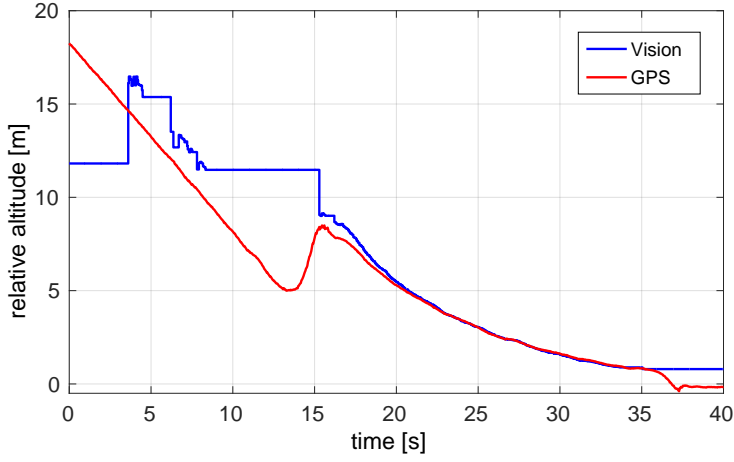


Figure 4.3 Vision-based relative state estimates compared to RTK-GPS measurements.

seen in Figure 4.3. The visual relative position estimate is a good method to compensate for GPS or communication problems during the most critical phase of the landing which significantly increases the reliability of the landing system. Figure 4.4 shows a series of images, illustrating the continuous functionality of the optical multi marker tracking algorithm until touchdown.

4.1.3 Sensor Data Fusion

To use both the marker detection and the relative GPS position, the sensor data is fused using a linear Kalman filter [59]. The fused relative position is then used in the control loops. The state of the filter consists of the three relative position components and relative velocity $x = [x, y, z, u, v, w]$. The state transition matrix is:

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 0 & \Delta t & 0 & 0 \\ 0 & 1 & 0 & 0 & \Delta t & 0 \\ 0 & 0 & 1 & 0 & 0 & \Delta t \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.4)$$

The filter is updated with the measurement available at the update timestep. If both sensors have valid measurements only the relative GPS position is used to update the filter. The

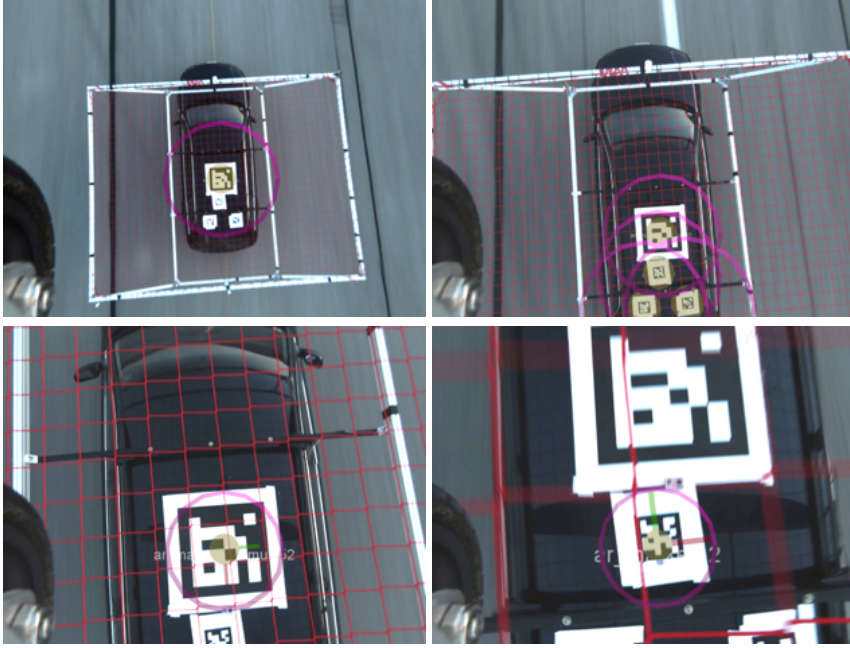


Figure 4.4 Continuous optical marker tracking until touchdown.

system noise covariance matrix is:

$$\mathbf{Q} = \begin{bmatrix} 0.005 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.005 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.0005 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.15 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.15 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.01 \end{bmatrix} \quad (4.5)$$

the measurement noise covariance matrices for GPS and marker tracking are:

$$\mathbf{R}_{GPS} = \begin{bmatrix} 0.4 & 0 & 0 \\ 0 & 0.4 & 0 \\ 0 & 0 & 1.6 \end{bmatrix} \quad \mathbf{R}_{vision} = \begin{bmatrix} 4 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 16 \end{bmatrix} \quad (4.6)$$

The relative velocity estimate is used to estimate the relative position when no new measurement is available. The Kalman filter is running at $20Hz$. This solution was considered to be sufficient, as the combination of the GPS-based relative position estimates and the fiducial marker localization system represents a robust choice.

4.2 Safety-Based Landing Control

As explained in detail Chapter 2.4, it was decided to develop two different landing control strategies, which are implemented as safety-based and the performance-based landing systems. One of the main reasons for this choice is rooted in the fact that the overall system stability is independent of the communication time delay in the safety-based approach, where the vehicles are coupled in a unilateral configuration. The early design choices presented in Tables 2.9 and 2.17 are further analyzed throughout this section and the safety-based landing controller is developed.

As explained in Chapter 2.4, the cooperative landing controller has been divided into two parts, namely horizontal and vertical control. The control task for the horizontal (longitudinal and lateral) position synchronization controller consists in the alignment of the two vehicles with sufficient precision to allow for a safe landing. This controller was designed to be independent of the relative vertical distance. The descent controller on the other hand needs to be dependent on the relative horizontal vehicle positions for safety reasons. The horizontal position synchronization controllers are used in a regulation sense, and will be active once the vehicles are in close proximity to each other. Such position synchronization task represents a multiple-input multiple-output (MIMO) control problem, and has many potential solutions with different degrees of sophistication and complexity. Similar problems are solved in other applications like aircraft formation flight, missile control, ship carrier landing, and aerial refueling, as previously explained in Chapter 1.4, where typical approaches include PID control [78], artificial potentials [82], and optimal control via MPC [33]. In this work, the choice was taken in favor of a PID-based position synchronization controller. This choice was motivated by the physical interpretability of the controller gains and its suitability for field experiments and heuristic fine-tuning. Experience has shown that an iterative validation of different controller functions through early flight experiments leads to a more efficient development process. A PID-based controller is ideally suited for such hands-on methodology.

In all of the following subsections a right-handed coordinate system is used to describe the motion of the aircraft and the ground vehicle with respect to the ground. The reference frame origin is located at the northern end of the runway (black home icon in Figure 4.11), with the x-axis pointing in southward runway direction, the y-axis to the right, and the z-axis downwards. It will be referred to as local runway coordinate system.

The indices a and g are used to associate quantities to the aircraft and ground vehicle respectively. If no additional index is provided the given quantity is referenced to the kinematic (or flight path) reference frame (e.g. $V_a = V_{ka}$).

In the case of horizontal control, the objective is to develop a strategy which forces convergence to zero for the magnitude of the position errors:

$$\begin{bmatrix} X_g - X_a \\ Y_g - Y_a \end{bmatrix}$$

Assuming desired inertial velocities and course angles as possible control inputs to the vehicles, and small course angles χ_a and χ_g , the following approximation can be obtained:

$$\begin{aligned}\dot{X}_{rel} &= V_g \cos \chi_g - V_a \cos \chi_a \\ &\approx V_g - V_a\end{aligned}\quad (4.7)$$

$$\begin{aligned}\dot{Y}_{rel} &= V_g \sin \chi_g - V_a \sin \chi_a \\ &\approx V_g \chi_g - V_a \chi_a\end{aligned}\quad (4.8)$$

It was therefore decided to control x -deviations through velocity inputs, and y -deviation via course angle inputs. As introduced in Chapter 2.4.5, it is assumed that both vehicles are capable of controlling their respective inertial velocities and course angles. This is not necessarily an obvious choice for the aircraft, as will be explained in Section 4.2.2.

Throughout this chapter the relative vehicle positions and velocities are calculated as follows:

$$\begin{aligned}X_{rel} &= X_g - X_a \\ V_{x,rel} &= V_{x,g} - V_{x,a}\end{aligned}\quad (4.9)$$

$$\begin{aligned}Y_{rel} &= Y_g - Y_a \\ V_{y,rel} &= V_{y,g} - V_{y,a}\end{aligned}\quad (4.10)$$

where X_a , X_g , Y_a and Y_g represent the vehicle positions expressed in the local runway frame.

4.2.1 Longitudinal Cooperative Control

Two main phases were considered during the landing. At the beginning of the landing maneuver, the ground vehicle is at rest at the runway threshold, while the aircraft is far behind, approaching the runway. While the aircraft is catching up, the ground vehicle already needs to accelerate to match the aircraft's speed. For this initial acceleration phase, the ground vehicle speed command is calculated under the assumption that the aircraft maintains its speed, while the ground vehicle has a constant acceleration. With these assumptions, and the constraint that speed and position shall be aligned at the same time, the following ground vehicle speed command is derived:

$$V_g^{des} = V_a - \sqrt{2 \cdot a_g \cdot X_{rel}} \quad (4.11)$$

where a_g is the desired (constant) acceleration of the ground vehicle, and V_a is the current aircraft ground velocity, assumed to remain constant. This control law implicitly defines the point at which the ground vehicle starts accelerating, which can be found by setting $V_g = 0$, and solving for the relative position:

$$X_{rel} = \frac{V_a^2}{2 \cdot a_g} \quad (4.12)$$

The start of the ground vehicle acceleration phase depends on the aircraft speed and the commanded ground vehicle acceleration. Assuming a typical approach speed of $21 \frac{m}{s}$ for the aircraft and roughly half the maximum ground vehicle acceleration for a_g , a relative position of around $160m$ is found. Since V_a in this case represents the aircraft's ground speed (and not its airspeed), effects of head- or tailwind are automatically considered and the ground vehicle will adjust its acceleration phase accordingly, as defined in Equations (4.11) and (4.12).

While a constant, relatively high acceleration is ideal for the initial acceleration phase of the ground vehicle, it is not suitable for close proximity operations in the second phase, where small position and velocity errors need to be corrected.

For the second phase an algorithm based on a feedback controller, coordinating the motion of both vehicles was implemented. As introduced in Chapter 2.4.5, this controller is referred to as position synchronization controller, its block diagram is shown in Figure 4.5. A position difference in x-direction (along the runway) is fed through the respective coupled controller blocks C_{XA} and C_{XG} and translated into a velocity command. This additional velocity command is then added to the desired landing ground speed of the aircraft, as well as to the current aircraft ground speed and sent to the aircraft and ground vehicle velocity controllers respectively. The direct feedback of the aircraft ground speed to the ground vehicle speed command acts as a damper on the relative motion of the vehicles and helps to improve the disturbance rejection characteristics of the landing system.

Different approaches could be used for the implementation of the aircraft and ground vehicle coupled controller blocks. During the performed experiments, which will be presented in Section 6, a basic implementation of the two blocks as shown in Figure 4.5 was used, where the velocity commands for the longitudinal direction are calculated of the form:

$$V_g^{des} = V_a + k_1 \cdot X_{rel} \quad (4.13)$$

$$V_a^{des} = V_{land} + k_2 \cdot X_{rel} \quad (4.14)$$

However, as there are relatively tight constraints on the aircraft's airspeed, especially during approach and landing, the control authority is rather limited for the aircraft.

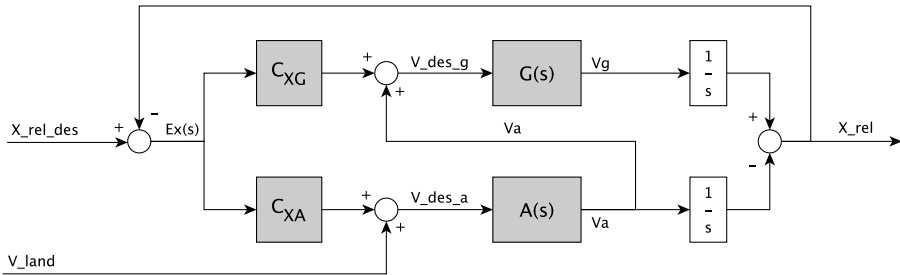


Figure 4.5 High-level longitudinal cooperative control strategy.

A preliminary stability analysis with the linearized system models of the aircraft and

ground vehicle has been performed for the longitudinal direction, as will be presented throughout this section. The objective of the present analysis is to investigate qualitative differences between the different possible feedback structures, and to consolidate the early design choices presented in Chapter 2.4. This first analysis was performed for an undelayed system, while the effects of communication time delays will be analyzed later on in Section 4.4.3. An optimization-based control design procedure for the determination of optimal controller gains will be presented in Section 4.4, which then allows for a detailed quantitative performance analysis of the demonstrator setup, as will be presented in Chapter 5.

The total longitudinal system consisting of aircraft and ground vehicle can be expressed as one single decoupled system of the form:

$$\begin{aligned} \begin{bmatrix} \dot{x}_a \\ \dot{x}_g \end{bmatrix} &= \begin{bmatrix} A_a & 0 \\ 0 & A_g \end{bmatrix} \begin{bmatrix} x_a \\ x_g \end{bmatrix} + \begin{bmatrix} B_a & 0 \\ 0 & B_g \end{bmatrix} \begin{bmatrix} u_a \\ u_g \end{bmatrix} \\ \begin{bmatrix} y_a \\ y_g \end{bmatrix} &= \begin{bmatrix} C_a & 0 \\ 0 & C_g \end{bmatrix} \begin{bmatrix} x_a \\ x_g \end{bmatrix} \end{aligned} \quad (4.15)$$

where x is the vehicle state vector, u the input vector, and y the output vector. The inputs are the desired velocity and path angle of the vehicles, and the outputs are the relative distance and velocity. Given that the longitudinal dynamics are considered isolated from the lateral dynamics, reasonable signals to feedback for the longitudinal control could be velocities, velocity difference and position difference. The system is closed by choosing the input as one of these Equations:

$$\begin{aligned} V_a^{des} &= V_{land} \\ V_g^{des} &= V_a + k_1 \cdot X_{rel} \end{aligned} \quad (4.16)$$

$$\begin{aligned} V_a^{des} &= V_{land} + k_2 \cdot X_{rel} \\ V_g^{des} &= V_a \end{aligned} \quad (4.17)$$

$$\begin{aligned} V_a^{des} &= V_{land} + k_2 \cdot X_{rel} \\ V_g^{des} &= V_a + k_1 \cdot X_{rel} \end{aligned} \quad (4.18)$$

where V_{land} is a ground speed appropriate for landing. In the first structure given in Equation 4.16 the ground vehicle takes over longitudinal positioning control while the aircraft keeps a constant airspeed. In the control structure from Equation 4.17, the aircraft takes over positioning control while the ground vehicle adapts its speed to the ground speed of the aircraft. The final control structure as given in Equation 4.18 combines the two previous approaches and makes both vehicles simultaneously react to differences in position.

Figure 4.6 shows how the respective root locus plots change with the different feedback structures for the longitudinal controller. In Figure 4.6a and Figure 4.6b, the relative

positioning information is only fed back to one vehicle. In Figure 4.6c, both vehicles use the positioning information simultaneously.

As can be seen in these plots, the coupled system remains stable in a wide range of gains k_1 and k_2 , while the overall dynamics of the coupled system are very similar for all three controller structures. The time constants and stability margins were also found to be similar.

In Figure 4.6, the pole locations of the different coupled system defined in Equation 4.16 are shown in dependence of the controller gains k_1 and k_2 .

Figure 4.6a shows the first controller structure as defined in Equation (4.16). In this case the aircraft dynamics are not dependent on the gain k_1 and are therefore not displayed in the plot. The controlled aircraft is considered a stable system and no further analysis was performed at this point. Here the influence of the controller gain on the overall coupled system dynamics was analyzed. As can be seen, the blue pole remains on the real axis for all values k_1 .

Figure 4.6b shows the root locus plot for the controller structure of Equation 4.17 where the relative positions are fed back only to the aircraft. The ground vehicle is commanded to adjust its speed to the current aircraft ground speed, automatically compensating for changing wind conditions. As can be seen, an imaginary part is developed with an increasing controller gain k_2 , which leads to oscillatory dynamics.

Figure 4.6c shows the root locus plot of the fully coupled system as defined through Equation 4.18, where the positioning error is fed back to both vehicles. Again, an imaginary component is developed with increasing controller gains $k_1 = k_2$ (varying both gains simultaneously).

In the end the first controller structure as shown in Figure 4.6a was considered to be favorable to the other approaches. The corresponding block diagram is shown in Figure 4.7, where the aircraft is represented by the transfer function $A(s)$ and the ground vehicle by $G(s)$. The ground vehicle longitudinal position synchronization controller is denominated as C_{XG} , and contains a P-controller as defined in Equation (4.16).

As mentioned before, all three approaches lead to similar overall dynamics with comparable time constants and stability margins. For the first approach the real axis pole does not develop an imaginary component with an increasing controller gain. Overall, this implies lower oscillations and consequently less control activity for both systems. It also represents a simpler and safer practical realization, with the aircraft velocity being constant and independent of the ground vehicle state. Additionally, all three approaches were tested in flight experiments where the first control structure was confirmed to be the best choice for the current experimental setup.

4.2.2 Ground Speed Tracking Mode

Controlling the motion of an aircraft means controlling the aerodynamic forces and moments acting on it. Therefore, the relevant physical quantities are referenced to the surrounding air mass. One such quantity is airspeed, representing a highly safety critical controlled variable in any flight control system. To guarantee safe flight, the airspeed must remain within admissible limits. As presented in Section 3.2.1, the implemented aircraft

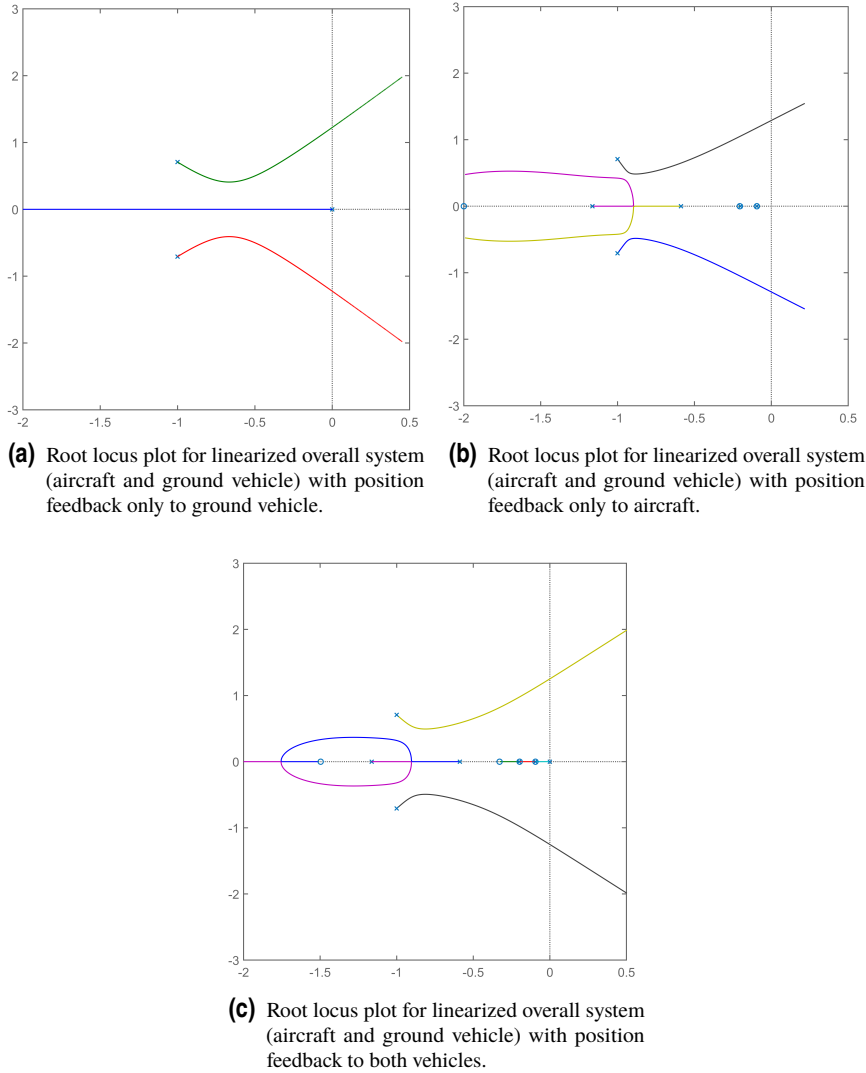


Figure 4.6 Root locus plots of different safety-based position synchronization controller feedback structures.

flight controller possesses such envelope protection capabilities. Consequently, it must be based on airspeed measurements and commands.

Considering the envisioned landing application, it would be beneficial if both vehicles were controlled with respect to the same physical quantities, namely inertial positions and velocities. The cooperative control strategy presented in the previous section is based on such capability. It is therefore necessary to modify the aircraft flight controller in a way

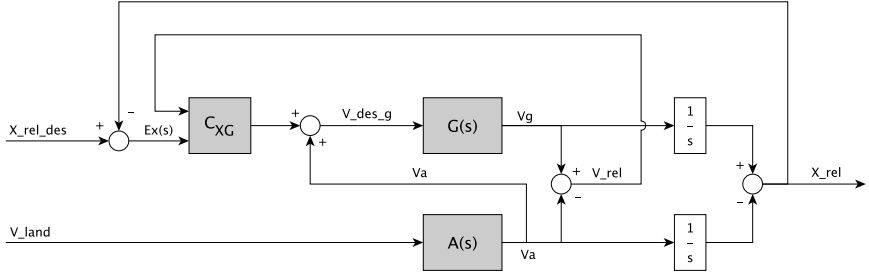


Figure 4.7 Safety-based longitudinal position synchronization controller.

to allow for the tracking of ground speed, while maintaining the full air mass referenced envelope protection functionality.

A ground speed tracking mode has thus been implemented, where the aircraft airspeed command is modified to control the current aircraft ground speed instead of airspeed. The aircraft will then actively compensate for atmospheric disturbances (wind gusts, ground vehicle induced disturbance) while maintaining the commanded ground speed. With the ground vehicle controlling ground speed as well, both vehicles then directly control the same physical quantity, thus facilitating the alignment process. It should be noted that all aircraft envelope protection features remain active and safe flight is still guaranteed, as only the airspeed command is modified while the actual aircraft flight controller remains unchanged.

The modified airspeed command, which is sent to the aircraft flight controller, can be derived as follows:

$$V_A^{des} = (V_k^{des} - V_k) + V_A \quad (4.19)$$

If such command was fed to the current aircraft velocity controller (based on airspeed control error e_{V_A}), only the control error in flight path velocity e_{V_k} would be considered for the generation of actuator commands, as V_A would cancel out:

$$\begin{aligned} e_{V_A} &= V_A^{des} - V_A = (V_k^{des} - V_k) + V_A - V_A \\ &= V_k^{des} - V_k = e_{V_k} \end{aligned} \quad (4.20)$$

The desired ground speed command is then determined as follows:

$$V_k^{des} = V_{k_land} + k \cdot x_{rel} \quad (4.21)$$

$$V_{k_land} = V_{A_land} + \overline{V_k - V_A} = V_{A_land} + V_w \quad (4.22)$$

Here V_{A_land} represents the desired landing airspeed, which is a fixed parameter set by the operator, while the wind estimate V_w represents the averaged difference between the aircraft ground speed and airspeed (denoted as $\overline{V_k - V_A}$), read out at the moment landing mode is activated (shortly after reaching WP2 in Figure 4.11). As a result, V_{k_land} represents the aircraft ground speed command that would approximately produce the airspeed present

at landing mode initiation. A low-pass filter with a cut-off frequency of 0.2Hz was used to obtain an averaged value for the wind estimate V_w . The resulting modified airspeed command required for ground speed tracking is then found as:

$$V_A^{des} = \underbrace{V_{A_land} + k \cdot x_{rel}}_{\text{previous airspeed command}} + V_w - V_k + V_A \quad (4.23)$$

4.2.3 Lateral Cooperative Control

The control approach used for the longitudinal direction, as described in Section 4.2.1, was applied to the lateral direction in a similar fashion. Here, a given lateral deviation is corrected for by commanding an appropriate corrective course action. Figure 4.8 provides the block diagram for the safety-based lateral position synchronization controller.

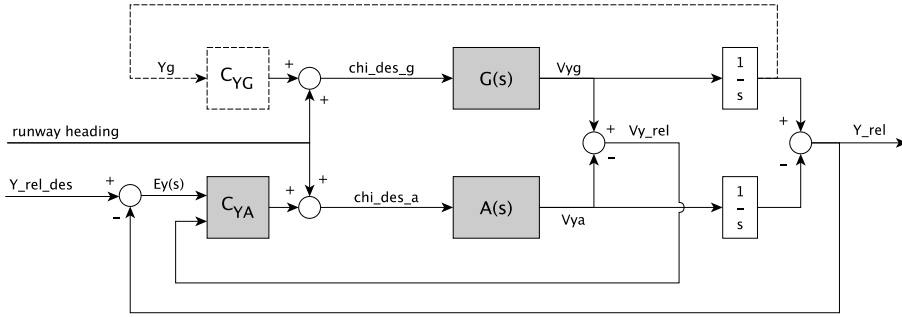


Figure 4.8 Safety-based lateral position synchronization controller with optional runway centerline tracking term (dashed block C_{YG}).

The desired course for the aircraft and the ground vehicle are now calculated as follows:

$$\begin{aligned} \chi_a^{des} = & \psi_{rwy} + k_1 Y_{rel} \\ & + k_2 \frac{1}{s} Y_{rel} \\ & + k_3 V_{y,rel} \end{aligned} \quad (4.24)$$

$$\chi_g^{des} = \psi_{rwy} \quad (4.25)$$

where the feedback of $V_{y,rel}$ is used as derivative term to avoid numerical differentiation and ψ_{rwy} represents the runway heading. As can be seen from these equations, an integral part has been added for the aircraft, as well as a damping term. With this control approach, the vehicles eliminate lateral position differences and keep the course parallel to the runway. In addition, the optional runway centerline tracking term C_{YG} (dashed block in Figure 4.8) can be used to drive the coupled vehicles back to the runway centerline after a disturbance (not used during the landing experiments due to the large available runway width).

4.2.4 Vertical Control and Flare Law

The vertical flight path was implemented as a constant 3° glideslope, combined with a classical flare law as shown in Figure 4.9 and described in [74, 75].

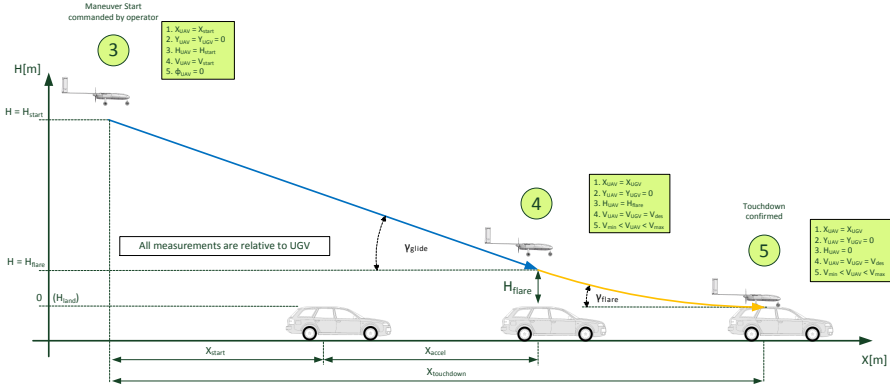


Figure 4.9 Final approach and flare maneuver.

The sink rate during the flare is commanded as:

$$\dot{h}_{flare} = k \cdot (h + h_B) \quad (4.26)$$

The constants k and h_B are calculated under the following constraints:

$$\begin{aligned} \dot{h}_{flare}(h_{flare}) &= \dot{h}_{descent} \\ \dot{h}_{flare}(0) &= 0 \end{aligned} \quad (4.27)$$

This would lead to a relative vertical velocity of zero at touchdown, which in principle is not desirable for a runway landing. In our case, where the landing is performed on top of a moving landing platform, the touchdown vertical velocity is adjusted through the retard and ground lock settings, as will be described in Section 4.2.5. The flare initiation altitude h_{flare} was set to $5m$ (relative altitude).

4.2.5 Retard and Ground Lock

Before a touchdown on top of the landing platform can be attempted, the aircraft engine has to be shut down in order not to damage the aircraft or the landing platform. This so-called *retard* takes place during a critical phase of the landing maneuver, where the aircraft is very close to the landing platform. A sudden loss of thrust force (throttle cut-off) will produce a deceleration of the aircraft, consequently a decrease of lift force and result in an increasing relative velocity and position in x - and z -direction (backward and downward motion).

At the same time the aircraft is already close enough to the ground vehicle to experience the aerodynamic disturbances caused by the ground vehicle (increased airspeed and angle of attack). The ground vehicle induced disturbances will partly compensate the downward motion but increase the backward motion due to the higher aerodynamic drag.

These complex effects have to be compensated by the flight controller, while now (after engine cut-off) only the elevator is available to control either altitude or velocity. The decision was taken in favor of velocity control; the desired vertical velocity at touchdown is achieved by adjusting the altitude at which retard is activated.

In the last moments before touchdown the so-called *ground-lock* configuration is activated. Thereby the ailerons are deflected to the maximum upward position symmetrically to reduce lift, the elevators are deflected downward to half of the maximum travel to generate a nose down moment, and the flaps are fully retracted. This configuration showed the best performance results during a number of ground tests, where the actual aircraft was placed on top of landing platform at around $70 \frac{\text{km}}{\text{h}}$. It was also found that a pitch damper had to remain active in order to reduce possible oscillations once the aircraft is lying on the net of the landing platform, as well as a locking mechanism, assuring that the front landing gear remains locked-in once it crossed the landing platform net, as illustrated in Figure 4.10.



Figure 4.10 Ground tests with aircraft on top of landing platform to determine the best strategy for the ground lock mechanism.

As mentioned before in Section 4.2.4, a non-zero vertical velocity is required for the landing and needs to be configured correctly (ideally 0.2m/s at touchdown). Therefore, the effects of both modes have been analyzed during virtual landing approaches at higher altitudes and the produced additional downward velocities $\dot{h}(t)$ have been determined through direct measurement. The influence of the ground vehicle induced disturbance on the landing performance has been estimated through dynamical simulation experiments (see Chapter 5.1), which included the CFD-data presented in Chapter 3.3. The final (relative) activation altitude was set to 0.6m for *retard mode* and 0.2m for *ground lock mode*.

It should be noted that the presented locking mechanism has been adapted to the

demonstrator setup and is not directly transferable to a larger aircraft. For the final large scale landing system, the ground lock mechanism has to be reworked, since a landing on a stretched net will not be possible anymore. Among other things, this might include the introduction of pre-defined attachment points on the aircraft as well as an attitude alignment and locking system on the ground side.

4.2.6 High-Level Mission Control and Fail-Safe System

A state machine has been developed to provide high-level mission control. The initial state corresponds to waypoint mode, where the aircraft flies an 8-shaped pattern above the airfield (Figure 4.11). The waypoints were set manually, taking into account pilot line-of-sight, obstacles (buildings, trees) and wind conditions. Most of the pattern is flown at an altitude of $110m$ above ground and an airspeed of $23\frac{m}{s}$, except for the section between WP1 and WP2, where the aircraft initiates a rapid descent to $30m$ at an increased airspeed of $28\frac{m}{s}$. The increased airspeed command allows the aircraft to reach the necessary descent angle. The aircraft then stays at $30m$ altitude for a short distance, which gives the operator the opportunity to command the start of the landing maneuver. If no such command is given, the aircraft will climb back to $110m$ and resume the waypoint pattern.

Once the landing is initiated by the operator, the aircraft's actions are determined by its position relative to the landing platform. Figures 4.12 and 4.13 provide an overview of the different zones and their corresponding actions in the longitudinal and lateral directions. The main divisions consist of an *inner fence* and an *outer fence*. As long as the aircraft is located within the inner fence, the landing proceeds normally. Between the inner fence and the outer fence, the approach is suspended (altitude hold), but the aircraft still tries to reach the inner fence in order to resume the landing. Outside of the outer fence, the maneuver is aborted, the aircraft performs a go-around and resumes waypoint flight.

With the aircraft inside the inner fence, the cooperative control described earlier is active, both in longitudinal and lateral direction, and the aircraft is descending. Above the flare height of $h_{flare} = 5m$, the flight path angle is constant at $\gamma_{approach} = -3^\circ$. Once the flare height has been reached, the flare law is active, reducing the vertical speed proportionally to the height above the landing platform. At $0.6m$ above the platform, the motor is turned off, and at $0.2m$ ground lock is activated. Once the motor is turned off, the aircraft is committed to the landing and will not perform any go-around maneuvers.

As can be seen in Figures 4.12 and 4.13, during the flare the inner fence sets relatively tight constraints of $\pm 0.8m$ in both longitudinal and lateral directions. This is necessary to ensure a safe landing, since larger deviations from the center of the landing platform may result in a part of the aircraft hitting the aluminum frame of the platform and getting damaged. Above h_{flare} , the allowed deviation increases as a linear function of the altitude, with a shallow slope behind the car, since it is the direction from which the aircraft is approaching.

If the aircraft leaves the boundary defined by this inner fence before reaching the flare height, the descent is stopped (altitude hold) while the cooperative control in lateral and longitudinal direction is still active. As soon as the aircraft is inside the inner fence again, the descent is resumed. If the inner fence is crossed after h_{flare} has been reached, the cooperative control remains active as well, but the aircraft is commanded to climb to

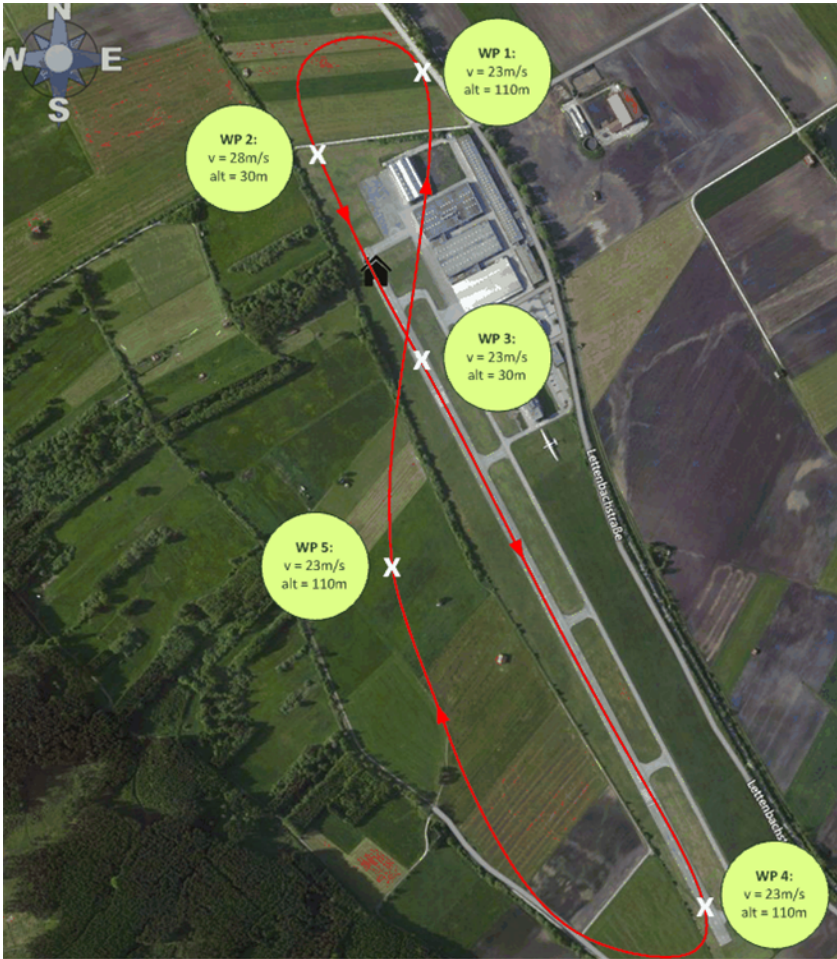


Figure 4.11 Waypoint pattern used for the landing maneuver.

$h_{flare} + 1m$ (retry mode), as it is considered unsafe continue flying at such a low altitude. Flare is only resumed when the aircraft is inside the inner fence and at least $3m$ above the landing platform.

While the aircraft is in flare mode, a retry is also triggered if the communication lag exceeds $0.3s$, resulting in the same actions as if the aircraft had left the inner fence. An excessive lag would result in unreliable data for the relative position estimation of the vehicles, and a landing would be too risky.

If, at any point in the maneuver, the aircraft crosses the outer fence, i.e. reaches the red zones in Figures 4.12 and 4.13, the maneuver is aborted altogether, and the aircraft flies in runway direction while climbing to an altitude of $20m$, before resuming the waypoint pattern. The aircraft will not resume the landing maneuver without a command from the operator.

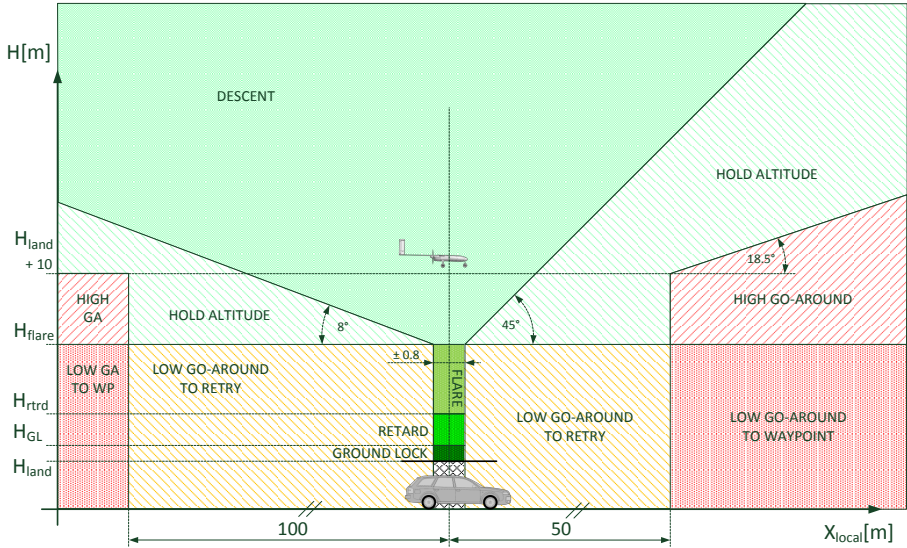


Figure 4.12 Geometrical conditions for state machine logic in the longitudinal direction (side view).

After a landing start command was given, no further input is required from the operator during the entire landing maneuver. However, the operator is able to initiate a manual retry or go-around at any point during the landing maneuver, resulting in the same go-around behavior as caused by an outer fence crossing. During the go-around maneuver, whether it was initiated automatically or manually, the operator can issue a *resume landing* command, if the aircraft is located inside the outer fence.

Additionally a flight termination system has been implemented, which monitors the validity of the GPS data, as well as the aircraft location with respect to pre-defined circular geofences. If the received GPS data is not changing for more than 5s, or if the geofence was violated, the aircraft is commanded to loiter at its current location for a duration of another 5s, before a circular descent is commanded until the aircraft touches the ground. During the loiter, the airspeed is commanded to 23m/s and controlled via the elevator, while the roll angle is commanded to 30° (right turn). The engine is set to a corresponding pre-defined trim value to approximately hold the altitude, since GPS altitude measurements are not available. If GPS data cannot be reestablished within 5s, or if the aircraft does not enter the geofenced area after 5s, the engine is switched off and the aircraft starts descending until it hits the ground. If the GPS failure or the geofence violation are removed during the loiter or circular descent phases, normal waypoint flight is resumed.

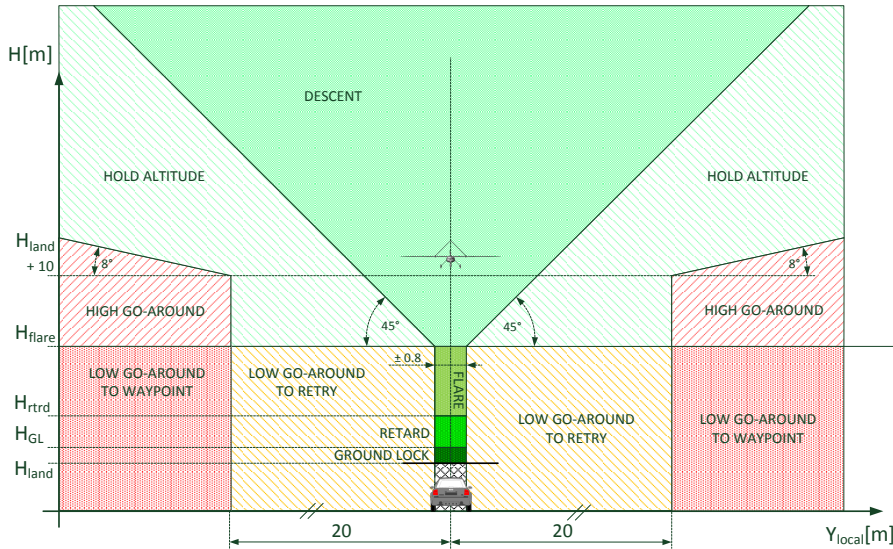


Figure 4.13 Geometrical conditions for state machine logic in the lateral direction (rear view).

4.2.7 Deceleration

After the aircraft has touched down and is locked onto the landing platform, the ground vehicles starts the deceleration process. During the deceleration the ground vehicle is commanded to follow a constant negative acceleration while tracking the runway centerline. During the deceleration process the aircraft actively dampens possible pitching oscillations using its elevators. Depending on the aircraft type active braking assistance could be provided by the aircraft by an appropriate setting of its control surfaces or spoilers.

4.3 Performance-Based Landing Control

Throughout the last section the safety-based landing controller has been presented. This section will describe the performance-based landing controller according to the design choices established in Chapter 2.4. The main differences between the two control strategies consist in the optimization-based approach, where optimal trajectories and control sequences are calculated for both vehicles, and the bilateral position synchronization control, where both vehicles actively contribute to reject disturbances in both horizontal dimensions. The flare controller, as well as the landing state machine remain the same for both methods and will not be presented again. Additionally, given the performance-based

control system's sensitivity to time delays (see Chapter 2.4.5), a time delay robust control approach was developed to stabilize the system in case of high delays by means of energy-based time domain passivity control (TDPA).

4.3.1 Optimized Approach

As introduced in Chapter 2.4.4 an optimization-based controller has been developed for the approach phase of the landing maneuver, where the main task consists in driving the involved vehicles towards alignment in the best possible way. How *best possible* was defined for the present landing application will be explained in the course of this section. The main motivation for choosing an optimal control strategy consists in its capability of capturing a number of constraints like vehicle dynamics, admissible state-space regions, input saturations, terminal states, runway dimensions, and others, while minimizing a given objective function. The objective function can be constructed in a way to obtain desired maneuver properties, which might include a minimal runway usage, short landing time, low controller activity, among others. Another interesting feature consists in the method's capability to work with binary constraints, where the system behavior can be altered based on logical (0,1) conditions. If for example the aircraft is located at a relative position or area with respect to the ground vehicle, a specific behavior can be imposed. Such capability can be used to consider geometric areas during the planning of the vehicle paths, where e.g. the aircraft is not supposed to continue its descent if the alignment in the horizontal plane is not given. In comparison to the state machine introduced in Sections 4.2.6 the optimization-based controller would not only react once a certain condition has already been reached, but would instead predict the course of events and actively avoid undesired states before they occur. In such way undesirable switching conditions, which might even drive otherwise stable subsystems to instability [84], could be avoided.

The optimization-based controller developed in this work represents an extended version of the controller presented in [96]. The main extensions consist in the introduction of adaptive state and input constraints (based on pseudo wind estimates), adaptive cost function weights (based on relative states), and the inclusion of runway elevation profiles for capturing vertical ground vehicle motion. These features will be presented in detail throughout the following sections.

The controller has been implemented in Matlab Simulink using the *YALMIP*¹ toolbox, in combination with the *Gurobi*² optimization solver.

To simplify terminology the following assumptions will be made throughout this section:

- the subscripts a and g will be used to specify quantities associated to the aircraft and ground vehicle respectively
- x and y positions will be referenced to the local runway frame as defined in Chapter 3.1.1
- the altitude of both vehicles will be provided with respect to the negative z-axis of the local runway frame (upwards positive)

¹YALMIP: A Toolbox for Modeling and Optimization in MATLAB (<https://yalmip.github.io>)

²Gurobi Optimization Solver (<https://www.gurobi.com>)

- course angle will be defined with respect to the runway direction (local runway frame x-axis) instead of North direction

In the following the simplified vehicle models used during the optimization will be presented, as well as the general control problem formulation.

Vehicle Models

Since the optimization-based controller is only going to be used during the approach phase of the landing, where a precise representation of short term dynamics is not required, kinematic models were chosen for modeling the aircraft and ground vehicle. The nonlinear kinematic aircraft model is shown in Equations (4.28)

$$\dot{x}_a = v_{k,a} \cos \gamma_a \cos \chi_a \quad (4.28a)$$

$$\dot{y}_a = v_{k,a} \cos \gamma_a \sin \chi_a \quad (4.28b)$$

$$\dot{h}_a = v_{k,a} \sin \gamma_a \quad (4.28c)$$

$$\dot{v}_{k,a} = f_1(v_{k,a}, v_{k,a}^{des}) = a_a \quad (4.28d)$$

$$\dot{\gamma}_a = f_2(\gamma_a, \gamma_a^{des}) \quad (4.28e)$$

$$\dot{\chi}_a = f_3(\chi_a, \chi_a^{des}) \quad (4.28f)$$

where the model inputs are the desired aircraft inertial velocity $v_{k,a}^{des}$, the desired flight path angle γ_a^{des} , and the desired course angle χ_a^{des} . The vehicle states are the position and velocity vectors. The aircraft acceleration a_a will be treated as a state later on to simplify the notation, although it represents a system output. The ground vehicle model is shown in Equation (4.29):

$$\dot{x}_g = v_{k,g} \cos \chi_g \quad (4.29a)$$

$$\dot{y}_g = v_{k,g} \sin \chi_g \quad (4.29b)$$

$$\dot{v}_{k,g} = f_4(v_{k,g}, v_{k,g}^{des}) = a_g \quad (4.29c)$$

$$\dot{\chi}_g = f_5(\chi_g, \chi_g^{des}) \quad (4.29d)$$

where the input is the desired ground vehicle inertial velocity $v_{k,g}^{des}$, and the desired course angle χ_g^{des} , while the states include the ground vehicle position and velocity. The functions f_1 to f_5 represent first order systems of the form:

$$\dot{\psi} = \frac{1}{\tau_\psi} (\psi^{des} - \psi) \quad \forall \quad \psi \in \{v_{k,a}, v_{k,g}, \gamma_a, \chi_a, \chi_g\} \quad (4.30)$$

where τ_ψ is a time constant. Since short term dynamics are not of interest during the approach phase, and due to the approximated dynamics (e.g. v_{des} to v) being rather slow and well damped, the first order system representation constitutes a valid approximation. The time constants have been chosen based on experimental data.

As the optimization must be solved continuously during the landing maneuver, it should be executed at the highest possible sampling rate. The problem will therefore be formulated as quadratic program, which requires linear relationships in its optimization constraints. Consequently, the vehicle models presented in Equations 4.28 and 4.29 are linearized around the landing maneuver trim velocity of $\bar{v}_k = 21 \text{ m/s}$.

With the small angle assumption (valid for the landing application) this results in the following linearized vehicle models:

$$\dot{x}_a = v_{k,a} \quad (4.31a)$$

$$\dot{y}_a = \bar{v}_k \cdot \chi_a \quad (4.31b)$$

$$\dot{h}_a = \bar{v}_k \cdot \gamma_a \quad (4.31c)$$

$$\dot{v}_{k,a} = \frac{1}{\tau_{v,a}} (v_{k,a}^{des} - v_{k,a}) = a_a \quad (4.31d)$$

$$\dot{\gamma}_a = \frac{1}{\tau_{\gamma,a}} (\gamma_a^{des} - \gamma_a) \quad (4.31e)$$

$$\dot{\chi}_a = \frac{1}{\tau_{\chi,a}} (\chi_a^{des} - \chi_a) \quad (4.31f)$$

$$\dot{x}_g = v_{k,g} \quad (4.32a)$$

$$\dot{y}_g = \bar{v}_k \cdot \chi_g \quad (4.32b)$$

$$\dot{v}_{k,g} = \frac{1}{\tau_{v,g}} (v_{k,g}^{des} - v_{k,g}) = a_g \quad (4.32c)$$

$$\dot{\chi}_g = \frac{1}{\tau_{\chi,g}} (\chi_g^{des} - \chi_g) \quad (4.32d)$$

Problem Formulation

For the implementation of these continuous equation systems into the discrete time controller, difference equations will be used, as shown in Equation (4.33):

$$\mathbf{s}(t+ds) = \mathbf{s}(t) + ds \cdot f(\mathbf{s}(t), \mathbf{u}(t)) \quad (4.33)$$

where the state and input vectors are defined as:

$$\begin{aligned} \text{where: } \mathbf{s} &= [x_a, y_a, h_a, v_{k,a}, a_a, \gamma_a, \chi_a, x_g, y_g, v_{k,g}, a_g, \chi_g]^\top \\ \text{and: } \mathbf{u} &= [v_{k,a}^{des}, v_{k,g}^{des}, \gamma_a^{des}, \chi_a^{des}, \chi_g^{des}]^\top \end{aligned} \quad (4.34)$$

The functions f correspond to the right-hand side of Equations 4.31 and 4.32. The notation $\sigma(t+i | t)$ represents a system state or input i iterations into the future, given the information available at time t . Therefore the index i represents the amount of time steps into the future and is limited by the fixed control horizon N where $i \in \{0, 1, 2, \dots, N\}$.

The prediction time step is denominated as ds . The fixed control horizon must be chosen correctly, since a too large horizon would result in excessive computation time (quadratic dependency), while a too low horizon might render the problem infeasible due to constraint violations. For this work the control horizon was fixed to a value of 35, which was based on the results from [96], where feasible sets were used to determine an appropriate time horizon, as well as on simulation experiments.

Next, the error vector is defined, which must be driven to zero before a touchdown can take place:

$$\boldsymbol{\varepsilon} = \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta v \\ \Delta a \\ \Delta \chi \end{bmatrix} = \begin{bmatrix} x_g - x_a \\ y_g - y_a \\ v_{k,g} - v_{k,a} \\ a_g - a_a \\ \chi_g - \chi_a \end{bmatrix} \quad (4.35)$$

$$\boldsymbol{\varepsilon}(t+N | t) \stackrel{!}{=} \mathbf{0} \quad (4.36)$$

In addition, the system inputs and states must remain within certain limits (e.g. runway dimension, vehicle performance limits) and are therefore constrained to lie in some convex sets:

$$\begin{aligned} \mathbf{s} &\subseteq \mathcal{S} = [\mathbf{s}^{min}, \mathbf{s}^{max}] \subseteq \mathbb{R}^n \\ \mathbf{u} &\subseteq \mathcal{U} = [\mathbf{u}^{min}, \mathbf{u}^{max}] \subseteq \mathbb{R}^m \end{aligned} \quad (4.37)$$

For the definition of the objective function a weighting matrix is defined for the states, inputs and input derivatives respectively, as shown in Equation (4.38):

$$\begin{aligned} \mathbf{P} &= \begin{bmatrix} p_x & 0 & 0 & 0 & 0 \\ 0 & p_y & 0 & 0 & 0 \\ 0 & 0 & p_{v_k} & 0 & 0 \\ 0 & 0 & 0 & p_a & 0 \\ 0 & 0 & 0 & 0 & p_\chi \end{bmatrix} & \mathbf{Q} &= \begin{bmatrix} q_{v,a} & 0 & 0 & 0 & 0 \\ 0 & q_{v,g} & 0 & 0 & 0 \\ 0 & 0 & q_{\gamma,a} & 0 & 0 \\ 0 & 0 & 0 & q_{\chi,a} & 0 \\ 0 & 0 & 0 & 0 & q_{\chi,g} \end{bmatrix} \\ & & & (4.38) \\ \mathbf{R} &= \begin{bmatrix} r_{v,a} & 0 & 0 & 0 & 0 \\ 0 & r_{v,g} & 0 & 0 & 0 \\ 0 & 0 & r_{\gamma,a} & 0 & 0 \\ 0 & 0 & 0 & r_{\chi,a} & 0 \\ 0 & 0 & 0 & 0 & r_{\chi,g} \end{bmatrix} \end{aligned}$$

The different matrices are then multiplied with the state, input and input derivative vectors in the form $\mathbf{s}^\top \mathbf{P} \mathbf{s}$, which yields positive costs for positive and negative argument values. In this work diagonal matrices were used, where each diagonal element corresponds to the weight for a specific state or input. Two additional weighting matrices will be introduced later on, namely the slack variable weighting matrix \mathbf{L} and the terminal state weighting matrix \mathbf{W} .

The overall optimization problem can now be formulated as follows:

$$\begin{aligned} \min \quad J = & \sum_{i=0}^N \varepsilon(t+i | t)^\top \cdot \mathbf{P} \cdot \varepsilon(t+i | t) + \\ & \sum_{j=1}^N \mathbf{u}(t+j | t)^\top \cdot \mathbf{Q} \cdot \mathbf{u}(t+j | t) + \\ & \sum_{k=2}^N \dot{\mathbf{u}}(t+k | t)^\top \cdot \mathbf{R} \cdot \dot{\mathbf{u}}(t+k | t) \end{aligned} \quad (4.39)$$

$$\begin{aligned} \text{subject to} \quad \mathbf{c}_1 = & \mathbf{s}(t+i+1 | t) = \mathbf{s}(t+i | t) + ds \cdot f(\mathbf{s}(t+i | t), \mathbf{u}(t+i | t)) \\ \mathbf{c}_2 = & \mathbf{s}(t+i | t) \subseteq \mathcal{S} \quad \forall i \in \{0, 1, 2, \dots, N\} \\ \mathbf{c}_3 = & \mathbf{u}(t+j | t) \subseteq \mathcal{U} \quad \forall j \in \{1, 2, 3, \dots, N\} \\ \mathbf{c}_4 = & \mathbf{s}(t | t) = \mathbf{s}^0 \\ \mathbf{c}_5 = & \mathbf{s}(N | t) = \mathbf{s}^N \end{aligned}$$

where J is the cost function, ε the error vector, \mathbf{P} the error weighting matrix, \mathbf{u} the input vector, \mathbf{Q} the input weighting matrix, $\dot{\mathbf{u}}$ the input derivative vector, and \mathbf{R} the input derivative weighting matrix. The constraint vector \mathbf{c}_1 corresponds to the vehicle kinematic constraints (vehicle models), \mathbf{c}_2 contains the state limits, \mathbf{c}_3 the input limits, \mathbf{c}_4 represents the initial conditions, and \mathbf{c}_5 the terminal state. \mathbf{s}^N represents a desired terminal state vector in \mathcal{S} , which satisfies Equation (4.36) and is required for system stability [96].

The solution of such optimization problem will lead to an optimal trajectory matrix \mathbf{S}^* and control sequence matrix \mathbf{U}^* , where the number of rows corresponds to the number of states and inputs respectively, while the number of columns corresponds to the time horizon N for both matrices. The trajectory matrix therefore has the dimensions $(n \times N)$, while the control sequence matrix is $(m \times N-1)$. Only the first column of \mathbf{U}^* will be sent to the vehicle controllers at each iteration. After one iteration, the next trajectory and control sequence vectors are calculated based on the new sensor information obtained after one time step ds , and the process is repeated.

Soft Constraints

The problem formulation from Equation (4.39) contains so-called *hard constraints*, which when not satisfied render the problem infeasible. If for example one such constraint requires that the altitude of the aircraft remained above a certain hard limit, but if some external disturbance (e.g. wind gust) pushed the aircraft beyond the limit, the optimization would produce an error. To avoid such undesirable behavior *soft constraints* are introduced, based on so-called *slack variables*. Slack variables are used to add some *slack* or *padding* to a given hard constraint. If a previous hard constraint was given as:

$$\mathbf{c}_1 = h_a(t+i | t) \geq h_{safe} \quad (4.40)$$

it is now replaced by the following two constraints:

$$\mathbf{c}_1' = h_a(t + i | t) \geq h_{safe} + \lambda_h^{max} - \lambda_h(t + i | t) \quad (4.41)$$

$$\mathbf{c}_2 = \lambda_h(t + i | t) \leq \lambda_h^{max} \quad (4.42)$$

where the new soft constraint \mathbf{c}_1' contains the slack variable λ_h , while \mathbf{c}_2 represents a new hard constraint on the maximum expected altitude variation λ_h^{max} due to the external disturbance. \mathbf{c}_1' now allows for h_a to vary around $h_{safe} + \lambda_h^{max}$ (depending on the current value of λ_h which results from the overall cost minimization), while \mathbf{c}_2 guarantees that h_a will always be bigger than h_{safe} , as long as the disturbance stays within the expected limits. The objective function must also be extended by the term $l_h \cdot [\lambda_h(t + i | t)]^2$, which drives the slack variable λ_h to zero. l_h thereby represents a component of the slack weighting matrix \mathbf{L} .

If the disturbances lie outside the expected value, the problem might still be rendered infeasible, which would have to be covered by the fail-safe system taking over control. Nevertheless, the use of well-designed soft constraints will solve the problem most of the time, since expected maximum variations due to disturbances are usually known. By adjusting the slack weight l_h , the stiffness properties of the slack region can be modified to better withstand a given disturbance.

Spatial Constraints

The optimization problem defined in Equation (4.39) would already allow for guiding the vehicles towards a synchronized landing condition, but does not yet contain any geometrical constraints. As explained in Chapter 3.3, the aircraft should approach the ground vehicle from straight above to minimize the effects of the ground vehicle induced disturbance and thus maximize landing safety. To produce such flight path, the geometric limitations as implemented in the landing state machine in Section 4.2.6 must be represented in the optimization problem, which can be achieved by introducing binary variables of the form $\mathcal{B} = \{0,1\}$ to the optimization problem formulation. By using binary position feedback (e.g. is the aircraft currently located in area A or B), certain constraints and objective functions terms can be *switched on and off* depending on the current state of the binary variable. If for example the aircraft should only start its descent once the runway threshold is reached (where the runway starts), the following constraint could be added to the optimization problem, where h_a would be minimized in the objective function:

$$\mathbf{c} = h_a(t + i | t) \geq h_{safe} - M \cdot b_h(t + i | t) \quad (4.43)$$

Here h_{safe} represents the desired altitude before the descent, M a sufficiently large number (will be explained in the following), and b_h a binary variable, which is zero when the aircraft has not yet reached the runway threshold and one otherwise. This new constraint will be *active* for $b_h = 0$, and *inactive* for $b_h = 1$ if M is sufficiently large, which in this case means $M > h_{safe}$. It is common to set M to some arbitrary, but sufficiently large number, to guarantee that all affected constraints will be correctly deactivated (in this work $M = 5000$ was chosen). This method is known as *Big-M method* [52]. Instead of only one

binary variable, multiple variables could be included and combined with each other via logical operators to represent more complex scenarios. The optimization problem then becomes a *Mixed Logical Dynamical System (MLDS)*.

Figure 4.14 illustrates the spatial constraints for the landing application:

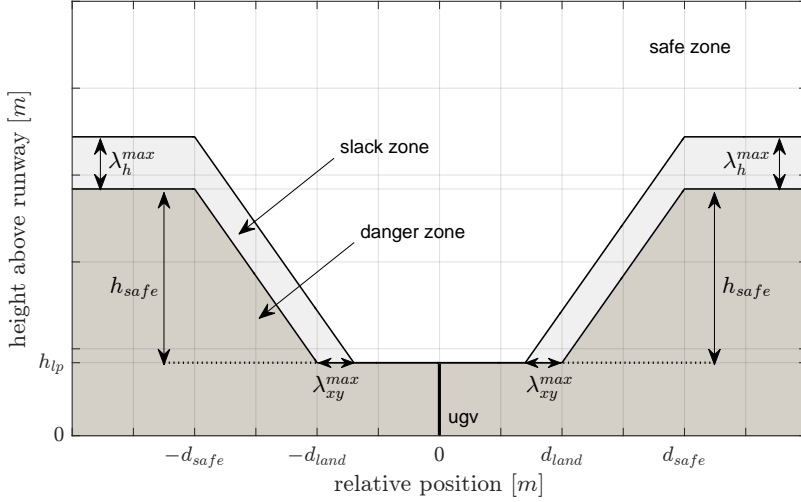


Figure 4.14 Longitudinal spatial constraints for the optimization-based approach controller.

where h_{lp} is the landing platform height, h_{safe} is the safety height above the landing platform, d_{safe} and d_{land} define the slope of the danger zone walls, and λ_h^{max} , and λ_{xy}^{max} are the maximum slack values in vertical and horizontal direction respectively. These parameters can be adjusted depending on the safety requirements of the given landing system setup. To reduce complexity, the different geometrical zones have been designed symmetrically in x and y directions.

With these spatial constraints, the necessary binary variables are defined as follows:

$$b_h(t+i | t) = \begin{cases} 0, & \text{if } H(t+i | t) \geq h_{safe} \\ 1, & \text{if } H(t+i | t) < h_{safe} \end{cases} \quad (4.44a)$$

$$b_x(t+i | t) = \begin{cases} 0, & \text{if aircraft inside landing zone in x-direction} \\ 1, & \text{if aircraft outside landing zone in x-direction} \end{cases} \quad (4.44b)$$

$$b_y(t+i | t) = \begin{cases} 0, & \text{if aircraft inside landing zone in y-direction} \\ 1, & \text{if aircraft outside landing zone in y-direction} \end{cases} \quad (4.44c)$$

where the *landing zone* was defined as the area between the infinite extension of the two inclined danger zone walls. Additionally, the relative vertical distance between the

aircraft and the landing platform was defined as:

$$H(t+i | t) = h_a(t+i | t) - h_g(t+i | t) - h_{lp} \quad (4.45)$$

where h_{lp} corresponds to the landing platform height above the runway. This allows for capturing possible altitude variations of the ground vehicle due to a given runway slope or irregularities (so far the landing platform height was assumed at constant runway altitude). With these definitions and with $\sigma \in \{x, y\}$, the spatial soft constraints for the *safe zone* in Figure 4.14 can now be formulated as follows:

$$H(t+i | t) \geq h_{safe}^\lambda - \lambda_h(t+i | t) - M \cdot b_h(t+i | t) \quad (4.46a)$$

$$H(t+i | t) \geq \frac{h_{safe} \cdot [\Delta\sigma(t+i | t) - d_{land}^\lambda - \lambda_{xy}(t+i | t)]}{d_{safe}^\lambda - d_{land}^\lambda} - M \cdot b_\sigma(t+i | t) \quad (4.46b)$$

$$H(t+i | t) \geq - \frac{h_{safe} \cdot [\Delta\sigma(t+i | t) + d_{land}^\lambda + \lambda_{xy}(t+i | t)]}{d_{safe}^\lambda - d_{land}^\lambda} - M \cdot b_\sigma(t+i | t) \quad (4.46c)$$

where: $h_{safe}^\lambda = h_{safe} + \lambda_h^{max}$
 $d_{safe}^\lambda = d_{safe} - \lambda_{xy}^{max}$
 $d_{land}^\lambda = d_{land} - \lambda_{xy}^{max}$

Here $\Delta\sigma$ corresponds to the position errors between the aircraft and ground vehicle in x and y direction, according to Equation (4.35).

The corresponding hard constraints for new slack variable are introduced as follows:

$$\begin{aligned} \lambda_h(t+i | t) &\leq \lambda_h^{max} \\ \lambda_{xy}(t+i | t) &\leq \lambda_{xy}^{max} \end{aligned} \quad (4.47)$$

One additional constraint is now required to complete the spatial constraint definition by capturing the dependencies between the sub-equations of Equation (4.46). Since either Equation (4.46a) or Equations (4.46b) and (4.46c) must be fulfilled at the same time, the following constraint is added:

$$b_h(t+i | t) \cdot [b_x(t+i | t) + b_y(t+i | t)] = 0 \quad (4.48)$$

At this point the complete MPC is defined, and could be used to calculate optimal trajectories and control sequences for the involved vehicles.

The main issue with this problem formulation consists in the computational demand. As can be seen in Equations (4.44) and (4.46) the overall optimization problem now requires the calculation of three binary variables, under consideration of the new constraint from Equation (4.48). This implies the solution of a MILP in every iteration, and was found to be computationally too demanding for the intended landing application, with computation times of more than 5s for a given time horizon of $N = 35$ and sample time of 0.4s [96].

Controller Decoupling

One possible way of reducing the high computational demand was proposed in [96], where the optimization problem was divided into two smaller sub-problems, one for the horizontal plane and one for the vertical plane. As can be seen in Equations (4.28) and (4.29) the vehicles operate in different state spaces, where both vehicles control the horizontal states, while only the aircraft acts in the h and γ dimensions. The vertical states are coupled with the horizontal states only via Equation (4.28c) and the spatial constraints defined in Equation (4.46). By performing a horizontal optimization first, Equation (4.28c) is disregarded, which leads to the assumption of independent controllability of speed and vertical flight path on the aircraft side, since vertical states are not yet known during the horizontal optimization. The flight control approach presented in Chapter 3.2 is especially well suited, since it provides such functionality even on short term dynamical level.

The binary constraints from Equation (4.46) are equally disregarded, since they also contain vertical states. As will be explained later on, the definition of the binary variables in Equation (4.46) needed to be modified for the decoupled approach, to only depend on d_{safe} , since the vertical states are not yet known during the horizontal optimization. The decoupling strategy is depicted in Figure 4.15, and shall be explained in the following.

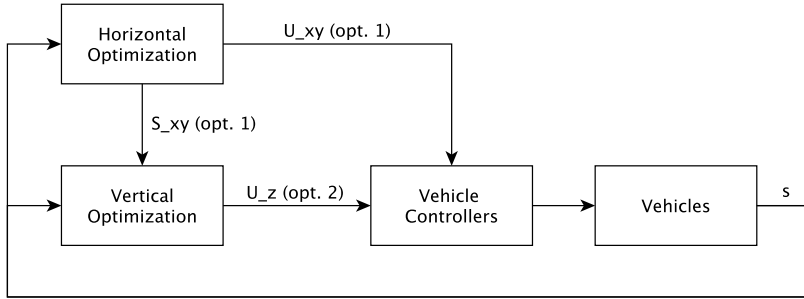


Figure 4.15 Controller decoupling strategy where the horizontal optimization is executed first (opt. 1), while the vertical optimization is executed second (opt. 2).

First the horizontal optimization is performed, where the vertical state equations are removed from the kinematic vehicle models, such that the following states and inputs remain to be optimized:

$$\begin{aligned} \mathbf{s}_{\mathbf{xy}} &= [x_a, y_a, v_{k,a}, a_a, \chi_a, x_g, y_g, v_{k,g}, a_g, \chi_g]^\top \\ \mathbf{u}_{\mathbf{xy}} &= [v_{k,a}^{des}, \chi_a^{des}, v_{k,g}^{des}, \chi_g^{des}]^\top \end{aligned} \quad (4.49)$$

This results in the horizontal state trajectory $\mathbf{S}_{\mathbf{xy}}^*$ and input sequence $\mathbf{U}_{\mathbf{xy}}^*$ of the form:

$$\begin{aligned} \mathbf{S}_{\mathbf{xy}}^* &= [\mathbf{s}_{\mathbf{xy}}^*(t | t), \quad \mathbf{s}_{\mathbf{xy}}^*(t + 1 | t), \mathbf{s}_{\mathbf{xy}}^*(t + 2 | t), \dots, \mathbf{s}_{\mathbf{xy}}^*(t + N | t)] \\ \mathbf{u}_{\mathbf{xy}}^* &= [\mathbf{u}_{\mathbf{xy}}^*(t + 1 | t), \mathbf{u}_{\mathbf{xy}}^*(t + 2 | t), \mathbf{u}_{\mathbf{xy}}^*(t + 3 | t), \dots, \mathbf{u}_{\mathbf{xy}}^*(t + N | t)] \end{aligned} \quad (4.50)$$

With the information about the future evolution of the horizontal states, the binary variables can be calculated for each time step. It should be noted that since the relative altitude is not taken into account during the horizontal optimization, the binary variables in Equation (4.44) are not being evaluated during the optimization and the spatial constraints Equation (4.46) are therefore disregarded. The horizontal optimization is thus not slowed by the consideration of binary variables during the optimization — the binary variables are calculated afterwards.

As previously mentioned, the definition of the binary variables needed to be modified since no vertical state predictions are available in \mathbf{S}_{xy}^* . The evaluation of the binary conditions must therefore be based on the relative horizontal states Δx and Δy only, which are compared against d_{safe} as follows:

$$b_h(t+i | t) = \begin{cases} 0, & \text{if } |\Delta x^*| > d_{safe} \quad \text{or} \quad |\Delta y^*| > d_{safe} \\ 1, & \text{if } |\Delta x^*| \leq d_{safe} \quad \text{and} \quad |\Delta y^*| \leq d_{safe} \end{cases} \quad (4.51a)$$

$$b_{xy}(t+i | t) = \begin{cases} 0, & \text{if } |\Delta x^*| \leq d_{safe} \quad \text{and} \quad |\Delta y^*| \leq d_{safe} \\ 1, & \text{if } |\Delta x^*| > d_{safe} \quad \text{or} \quad |\Delta y^*| > d_{safe} \end{cases} \quad (4.51b)$$

The previously defined landing zone (trapezoidal shape in Figure 4.14) now becomes a rectangle of edge length $2 \cdot d_{safe}$. This change does not affect the functionality of the spatial soft constraints as defined in Equation (4.46), but only affects the timing of their activation, which is now slightly delayed. It should be noted that the extended logical evaluation shown in Equation (4.51) allows for consolidating the binary variables b_x and b_y into the horizontal binary variable b_{xy} , and removing the previously introduced constraint shown in Equation (4.48).

In such way, the complete binary variable trajectory $\mathbf{B}^* = [\mathbf{b}_h^*, \mathbf{b}_{xy}^*]^\top$ is available for the vertical optimization a priori, which significantly reduces the computational demand of the problem. The vertical optimization then only needs to consider the vertical states and inputs:

$$\begin{aligned} \mathbf{s}_z &= [h_a, \gamma_a]^\top \\ u_z &= \gamma_a^{des} \end{aligned} \quad (4.52)$$

which finally leads to the predicted vertical state trajectory \mathbf{S}_z^* and control sequence \mathbf{U}_z^* of the form:

$$\begin{aligned} \mathbf{S}_z^* &= [\mathbf{s}_z^*(t | t), \mathbf{s}_z^*(t+1 | t), \mathbf{s}_z^*(t+2 | t), \dots, \mathbf{s}_z^*(t+N | t)] \\ \mathbf{u}_z^* &= [u_z^*(t+1 | t), u_z^*(t+2 | t), u_z^*(t+3 | t), \dots, u_z^*(t+N | t)] \end{aligned} \quad (4.53)$$

The overall result can then be assembled as follows:

$$\begin{aligned} \mathbf{S}^* &= [\mathbf{S}_{xy}^*, \mathbf{S}_z^*]^\top \subseteq \mathcal{S} \\ \mathbf{U}^* &= [\mathbf{u}_{xy}^*, \mathbf{u}_z^*]^\top \subseteq \mathcal{U} \end{aligned} \quad (4.54)$$

During a series of Hardware In the Loop (HIL) experiments, consistent computation times of around 0.15s were found, which corresponds to a sampling rate of around 6Hz, and

was deemed sufficiently fast for the envisioned approach control task.

Adaptive Limits

The fixed input limits introduced in Equation (4.37) might cause undesirable behavior in the presence of wind. As introduced in Chapter 2.3, the coordinated motion of both vehicles is controlled via inertial velocity commands, while the aircraft controller must still operate in terms of airspeed to allow for continued envelope protection. To achieve such functionality the ground speed tracking mode was introduced in Chapter 4.2.2. It allows for tracking ground speed commands only if they result in safe airspeeds, which means that in case the optimized control sequence would command the aircraft to fly beyond its operational limits, the envelope protection system would prevent this to happen. The aircraft would thus not follow the predicted optimal trajectory in such cases, which would deteriorate the performance of the optimized landing approach. The current wind situation must therefore be considered in the optimization problem, which requires some type of wind estimation.

The wind vector is defined as the difference between the ground speed and airspeed vectors. Since only a simple pitot tube is available on the aircraft to measure the airspeed's body frame x-component, a real wind estimate cannot be calculated. Instead an averaged scalar *pseudo wind estimate* is calculated by subtracting the scalar airspeed measurement v_a from the ground speed vector's absolute value v_k , and feeding the result through a low pass filter with time constant $\tau_w = 0.8s$:

$$\hat{v}_w(t + ds) = \hat{v}_w(t) + ds \frac{v_k(t) - v_a(t) - \hat{v}_w(t)}{\tau_w} \quad (4.55)$$

It should be noted that this quantity does not represent a real wind estimate, but rather corresponds to the required ground speed command change to maintain a given airspeed at the current flight path angle and course angle. This information can be used to design adaptive minimum and maximum limits for the ground speed command, thus allowing for the inherent consideration of vehicle performance limits during the optimization. In such way the envelope protection mechanism would not be triggered due to airspeed violations commanded out of the optimization. As the demonstrator aircraft possesses an aerodynamically stable design (where the aircraft returns to a constant trim angle of attack after a disturbance) and the whole landing maneuver is flown in crabbed condition (at zero sideslip angle) an improved wind estimate as shown in [9] was investigated, where the trim values of the angle of attack and sideslip angle were provided to a discrete-time Kalman filter as *pseudo measurements*. The approach allows for the estimation of a full wind vector, but the benefit was eventually found to be minor for the landing scenario, which underlines the validity of the simpler pseudo wind estimate for the present application.

The adaptive limits for the aircraft ground speed and flight path angle are calculated as follows:

$$v_{k,a}^{a,min} = v_{k,a}^{f,min} + \hat{v}_w \leq v_{k,a} \leq v_{k,a}^{f,max} + \hat{v}_w = v_{k,a}^{a,max} \quad (4.56a)$$

$$\gamma_a^{a,min} = \arcsin\left(\frac{\dot{h}_a^{min}}{v_{k,a}^{f,max} + \hat{v}_w}\right) \leq \gamma_a \leq \arcsin\left(\frac{\dot{h}_a^{max}}{v_{k,a}^{f,max} + \hat{v}_w}\right) = \gamma_a^{a,max} \quad (4.56b)$$

where: $\dot{h}_a^{min} = v_{k,a}^{f,max} \cdot \sin(\gamma_a^{f,min})$

and: $\dot{h}_a^{max} = v_{k,a}^{f,max} \cdot \sin(\gamma_a^{f,max})$

where the superscripts a and f indicate adaptive and fixed limits respectively, while the subscript a indicates aircraft quantities. The adaptive flight path angle limits were designed under the assumption that the same minimum vertical velocity should be achievable independently of the current wind situation. Figure 4.16 illustrates the effect of a headwind on the resulting minimum flight path angle, where the black vectors represent the wind-free case, the red vectors the resulting situation with fixed limit, and the blue vectors show the adaptive case. The subscripts indicate either ground speed (k) or airspeed (a). The pitot tube was assumed to be aligned with the airspeed vector to simplify the illustration, which is a valid assumption for trimmed flight conditions with small angles of attack.

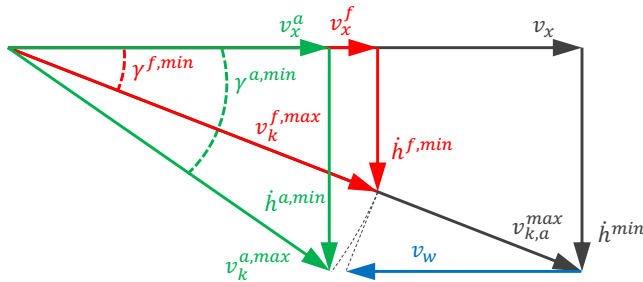


Figure 4.16 Resulting minimum flight path angles in headwind conditions for fixed (red) and adaptive (green) input limits. The black vectors illustrate the wind-free case.

The black vectors in Figure 4.16 represent a flight condition with minimum flight path angle and maximum ground speed for the wind-free case. Here the ground speed and airspeed vectors $v_{k,a}^{max}$ are identical. In the headwind case, the resulting maximum ground speed would be reduced by the amount of the pseudo wind estimate to maintain the maximum admissible airspeed, which corresponds to the ground speed in the wind-free case. With fixed flight path angle limits, such condition would result in a reduced minimum vertical velocity $\dot{h}^{f,min}$, as illustrated by the red vectors. With adaptive limits, the minimum vertical velocities \dot{h}^{min} and $\dot{h}^{a,min}$ have the same magnitude, while the magnitude of the ground speed vector is maintained.

It should be noted that the flight path angle is also covered by the envelope protection

system and cannot assume arbitrary values. Hence, for strong winds the optimization might still be limited by the envelope protection. For a nominal ground speed of 23m/s , and the currently implemented flight path angle limit of -5° , a headwind of 5m/s (worst case assumption as defined in Table 2.1) would lead to an adaptive flight path angle limit of around -6.4° , if vertical velocity was maintained. Since this still lies within the performance range of the demonstrator aircraft, the flight path angle protection is not expected to cause any issues at this point.

To allow for a correct use of the adaptive aircraft limits, the ground vehicle maximum speed limit also needed to be replaced by an adaptive limit in accordance to Equation (4.56a). The lower ground vehicle limit is zero anyway and does not require any modification, as shown in Equation (4.57):

$$v_{k,g}^{a,min} = v_{k,g}^{f,min} = 0 \frac{m}{s} \leq v_{k,g} \leq v_{k,g}^{f,max} + \hat{v}_w = v_{k,g}^{a,max} \quad (4.57)$$

Adaptive and Terminal Weights

In addition to the adaptive input limits introduced in the last section, another extension to the optimization problem presented in Equation (4.39) was performed. The constant objective function weights from Equation (4.38) have been extended by adaptive terms to compensate for the dependency on initial conditions and add more flexibility for shaping the system behavior. The adaptive weights are therefore designed as follows:

$$p_\sigma^a = p_\sigma^k + p_\sigma^l \cdot |\sigma_g^0 - \sigma_a^0| \quad \forall \sigma \in \{x, y, v_k, a, \chi\} \quad (4.58)$$

where k indicates a constant weight term, l a constant multiplicative factor of the adaptive weight term, and $|\Delta\sigma^0|$ the relative state of the vehicles, where the superscript 0 indicates the value of the different states σ right before a given iteration. The introduction of these adaptive state weights has proven to be very beneficial in terms of system robustness and overall performance.

In addition to the adaptive weights presented above, terminal weights have been added to the objective function to reduce certain terminal states. The obvious states chosen here were the terminal ground vehicle position, leading to a reduction in used runway length, and the lateral ground vehicle position, driving the ground vehicle towards the runway centerline at the end maneuver of the maneuver:

$$w_x \cdot [x_g(t + N | t)]^2 \quad (4.59a)$$

$$\sum_{i=0}^N w_y \cdot [y_g(t + i | t)]^2 \quad (4.59b)$$

where the terminal weights w_x and w_y are elements of the terminal weighting matrix \mathbf{W} . The addition of other terminal weights is conceivable to achieve a desired landing system behavior, which allows for manifold shaping possibilities and represents one of the main advantages of the optimization-based control approach.

Runway Elevation Prediction

The ground vehicle cannot control its vertical degree of freedom, but can passively move in the vertical dimension due to an uneven runway. Variable runway slopes and runway undulation are quite common, and height differences of $15m$ and more from one runway end to the other are possible (e.g. *EDMO* airport). The ground vehicle will therefore exhibit some vertical motion, which might influence the desired touchdown sink rate. As the runway elevation data is time invariant, it is quite easily measured and included in the optimization problem. With the predicted ground vehicle states S_{xy}^* from the horizontal optimization run, the ground vehicle altitude can be calculated from the recorded runway elevation data and forwarded to the vertical optimization as predicted ground vehicle altitude trajectory h_g^* . Figure 4.17 illustrates the elevations profiles of the airports in Oberpfaffenhofen (*EDMO*) and Mindelheim-Mattsies (*EDMN*), which were used for performing experiments. At

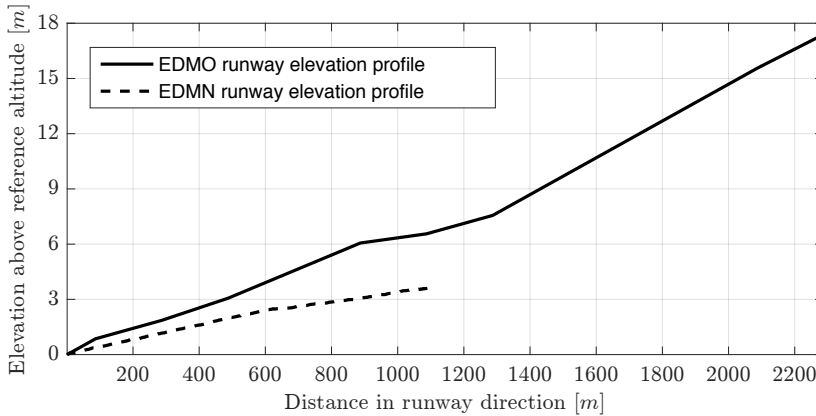


Figure 4.17 Runway elevation profiles of Oberpfaffenhofen airport (*EDMO*) and Mindelheim-Mattsies airport (*EDMN*).

this point all necessary sub-components of the developed optimization-based approach controller were derived. In the following sections the complete controller formulation will be provided, separated into the horizontal and vertical approach controllers.

Complete Horizontal Approach Controller

The complete horizontal approach controller will be provided in the following. The horizontal objective function is defined as:

$$\begin{aligned}
 J_{xy} = & \sum_{i=0}^N [\lambda_{xy}^s(t+i|t)]^\top \cdot \mathbf{L}_{xy} \cdot [\lambda_{xy}^s(t+i|t)] + \\
 & \sum_{i=0}^N [\varepsilon(t+i|t)]^\top \cdot \mathbf{P}_{xy} \cdot [\varepsilon(t+i|t)] + \\
 & \sum_{j=1}^N [\mathbf{u}_{xy}(t+j|t)]^\top \cdot \mathbf{Q}_{xy} \cdot [\mathbf{u}_{xy}(t+j|t)] + \\
 & \sum_{k=2}^N [\dot{\mathbf{u}}_{xy}(t+k|t)]^\top \cdot \mathbf{R}_{xy} \cdot [\dot{\mathbf{u}}_{xy}(t+k|t)] + \\
 & \sum_{j=1}^N [\mathbf{s}_{xy}(t+N|t)]^\top \cdot \mathbf{W}_{xy} \cdot [\mathbf{s}_{xy}(t+N|t)]
 \end{aligned} \quad (4.60)$$

where \mathbf{L}_{xy} represents the slack weighting matrix, \mathbf{P}_{xy} the error weighting matrix, \mathbf{Q}_{xy} the input and \mathbf{R}_{xy} the input derivative weighting matrices, and \mathbf{W}_{xy} the terminal weighting matrix.

The constraints contain the horizontal state equations of the vehicle models from Equations (4.28) and (4.29):

$$\mathbf{s}_{xy}(t+i+1|t) = \mathbf{s}_{xy}(t+i|t) + ds \cdot f(\mathbf{s}_{xy}(t+i|t), \mathbf{u}_{xy}(t+i|t)) \quad (4.61)$$

the input and state limits, where the state limits were formulated as soft constraints to add some robustness:

$$\begin{aligned}
 \mathbf{s}_{xy}(t+i|t) & \leq \mathbf{s}_{xy}^{\max} + \lambda_{xy}^s(t+i|t) \\
 \mathbf{s}_{xy}(t+i|t) & \geq \mathbf{s}_{xy}^{\min} - \lambda_{xy}^s(t+i|t) \\
 \mathbf{u}_{xy}(t+j|t) & \leq \mathbf{u}_{xy}^{\max} \\
 \mathbf{u}_{xy}(t+j|t) & \geq \mathbf{u}_{xy}^{\min}
 \end{aligned} \quad \begin{aligned} & \forall i \in \{0, 1, 2, \dots, N\} \\ & \forall j \in \{1, 2, 3, \dots, N\} \end{aligned} \quad (4.62)$$

$$\text{where: } \lambda_{xy}^s = [\lambda_{x_a}, \lambda_{y_a}, \lambda_{v_{k,a}}, \lambda_{a_a}, \lambda_{\chi_a}, \lambda_{x_g}, \lambda_{y_g}, \lambda_{v_{k,g}}, \lambda_{a_g}, \lambda_{\chi_g}]^\top$$

the initial condition and terminal constraints:

$$\mathbf{S}_{xy}(t|t) = \mathbf{s}_{xy}^0 \quad (4.63a)$$

$$\varepsilon(t+N|t) = \mathbf{0} \quad (4.63b)$$

Complete Vertical Approach Controller

The complete vertical approach controller will be presented in the following. The vertical objective function is defined as:

$$\begin{aligned}
 J_z = & \sum_{i=0}^N [\lambda_z^s(t+i | t)]^\top \cdot \mathbf{L}_z \cdot [\lambda_z^s(t+i | t)] + \\
 & \sum_{i=0}^N p_h \cdot [h_a(t+i | t) - h_g^*(i) - h_{lp}]^2 \cdot b_h^*(i) + \\
 & \sum_{i=0}^N p_\gamma \cdot [\gamma_a(t+i | t)]^2 \cdot b_{xy}^*(i) + \\
 & \sum_{j=1}^N [\mathbf{u}_z(t+j | t)]^\top \cdot \mathbf{Q}_z \cdot [\mathbf{u}_z(t+j | t)] + \\
 & \sum_{k=2}^N [\dot{\mathbf{u}}_z(t+k | t)]^\top \cdot \mathbf{R}_z \cdot [\dot{\mathbf{u}}_z(t+k | t)]
 \end{aligned} \tag{4.64}$$

where: $\lambda_z^s = [\lambda_{\gamma_a}, \lambda_h, \lambda_{xy}]^\top$

where \mathbf{L}_z represents the slack weighting matrix, p_h and p_γ the weights for the altitude and flight path angle respectively, \mathbf{Q}_z the input and \mathbf{R}_z the input derivative weighting matrices. The superscript \star represents already predicted values from the horizontal optimization run. As can be seen in Equation (4.64), the altitude and flight path angle contributions only become active once the binary variables b_h^* and b_{xy}^* are equal to one, which corresponds to the aircraft being located within the landing zone. It should be noted that b_h^* and b_{xy}^* always have opposed values, which implies that the aircraft will try to maintain a small flight path angle while outside the landing zone, while reducing altitude inside the landing zone. This does not mean that the aircraft will only start descending once it enters the landing zone, since the entire descent is being predicted at each optimization run, and must be completed within N timesteps.

The optimization constraints contain the vertical state equations of the linear vehicle models from Equations (4.28) and (4.29):

$$\mathbf{s}_z(t+i+1 | t) = \mathbf{s}_z(t+i | t) + ds \cdot f(\mathbf{s}_z(t+i | t), \mathbf{u}_z(t+i | t)) \tag{4.65}$$

the input and state limits, where the flight path angle limit was incorporated as soft constraint to increase system robustness:

$$\begin{aligned}
 h_a(t+i | t) &\leq h_a^{max} \\
 h_a(t+i | t) &\geq h_a^{min} + h_g^*(i) \\
 \gamma_a(t+i | t) &\leq \gamma_a^{max} + \lambda_\gamma(t+i | t) \\
 \gamma_a(t+i | t) &\geq \gamma_a^{min} - \lambda_\gamma(t+i | t) \\
 \lambda_\gamma(t+i | t) &\leq \lambda_\gamma^{max} \\
 \gamma_a^{des}(t+j | t) &\leq \gamma_a^{des,max} \\
 \gamma_a^{des}(t+j | t) &\geq \gamma_a^{des,min}
 \end{aligned} \quad \forall i \in \{0,1,2,\dots,N\} \quad (4.66)$$

$$\forall j \in \{1,2,3,\dots,N\}$$

where the ground vehicle altitude h_g^* is an input from the previously executed horizontal optimization run. The vertical constraints further include the initial constraints:

$$\mathbf{S}_z(t | t) = \mathbf{s}_z^0 \quad (4.67)$$

and terminal constraint on the desired sink rate at touchdown \dot{h}_{td} :

$$\frac{h_a(t+N | t) - h_a(t+N-1 | t)}{ds} = \dot{h}_{td} \quad (4.68)$$

as well as the spatial constraints representing the danger zone from Equation (4.46), with $\sigma \in \{x,y\}$:

$$H(t+i | t) \geq h_{safe}^\lambda - \lambda_h(t+i | t) - M \cdot b_h^*(t+i | t) \quad (4.69a)$$

$$H(t+i | t) \geq \frac{h_{safe} \cdot [\Delta\sigma^*(t+i | t) - d_{land}^\lambda - \lambda_{xy}(t+i | t)]}{d_{safe}^\lambda - d_{land}^\lambda} - M \cdot b_{xy}^*(t+i | t) \quad (4.69b)$$

$$H(t+i | t) \geq - \frac{h_{safe} \cdot [\Delta\sigma^*(t+i | t) + d_{land}^\lambda + \lambda_{xy}(t+i | t)]}{d_{safe}^\lambda - d_{land}^\lambda} - M \cdot b_{xy}^*(t+i | t) \quad (4.69c)$$

where: $h_{safe}^\lambda = h_{safe} + \lambda_h^{max}$

$$d_{safe}^\lambda = d_{safe} - \lambda_{xy}^{max}$$

$$d_{land}^\lambda = d_{land} - \lambda_{xy}^{max}$$

and the corresponding slack variable constraints:

$$\begin{aligned}
 \lambda_h(t+i | t) &\leq \lambda_h^{max} \\
 \lambda_{xy}(t+i | t) &\leq \lambda_{xy}^{max}
 \end{aligned} \quad (4.70)$$

The variables $\Delta\sigma^*$, b_h^* and b_{xy}^* are now constant inputs received from the preceding horizontal optimization. At this point the complete optimization-based approach controller is defined and ready to be integrated into the overall performance-based landing controller. As previously mentioned, the predictive controller will only be used until the vehicles have reached a synchronized horizontal state for the first time. Then control will be passed over to the bilateral position synchronization controller, which runs at a higher sampling rate to effectively reject disturbances. The integration of the optimal approach controller into the overall landing control system and the mentioned switching mechanism will be explained in the following section. A thorough performance analysis of the performance-based landing controller via deterministic and stochastic simulation experiments, and its comparison to the safety-based landing controller will be presented in Chapter 5.

Integration in Overall Landing Control System

As the name indicates, the optimal approach controller will only be used during the approach phase of the landing maneuver. As the predictive controller includes the full landing trajectory (until touchdown) in its optimization runs, at some point a switch over to the performance-based position synchronization controller must be performed. This transition is performed automatically by the landing state machine once the aircraft reaches the flare initiation altitude $h_{rel} = h_{flare}$. At the same time, the flare controller is activated and takes over vertical control. The retard and ground lock modes remain the same as already presented for the safety-based controller, where the engine is switched off at $h_{rel} = 0.6m$, and the aircraft go into ground lock configuration $0.2m$ above the landing platform. The deceleration as well remains unchanged. It should be noted that the operation of the approach controller is monitored by the fail-safe system, to capture cases where the optimization returns erroneous values in case of an infeasible problem (e.g. due to an excessive wind gust). This functionality was added to the landing state machine, where the system is temporarily switched back to the safety-based controller if the optimal approach controller output is found to be faulty.

4.3.2 Bilateral Position Synchronization

As introduced in Chapter 2.4.5, a bilateral position synchronization control structure will be applied for the performance-based landing controller. Once the aircraft completed the approach phase and reached horizontal position synchronization at flare initiation altitude, the bilateral position tracking controller is automatically activated. Figure 4.18 shows the block diagram for the performance-based longitudinal position synchronization controller.

It should be noted that in contrast to the safety-based controller from Figure 4.7, the output of the longitudinal cooperative controllers C_{XG} and C_{XA} are now calculated in terms of inertial runway axes (i.e. cartesian x-direction) instead of the previous ground speed vector magnitudes (commanded in the kinematic reference frame). A subsequent transformation into the spherical command structure (V_k and χ) as expected by the lower-level vehicle controllers is performed in the blocks *command transformation* as shown in Equation 4.71. Additionally, only the x-component of the aircraft velocity is fed back to the UGV instead of the velocity vector magnitude V_k , as was the case for the safety-based

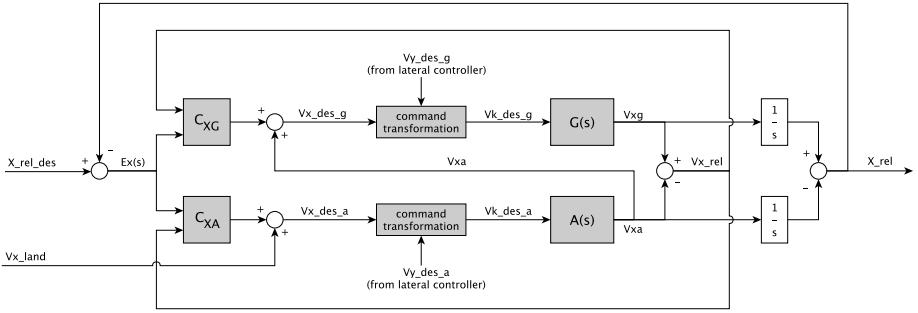


Figure 4.18 Performance-based longitudinal position synchronization controller.

controller.

$$V_k(t) = \sqrt{V_x^2(t) + V_y^2(t)}$$

$$\chi(t) = \arctan\left(\frac{V_y}{V_x}\right) \quad (4.71)$$

Figure 4.19 shows the block diagram of the performance-based lateral position synchronization controller.

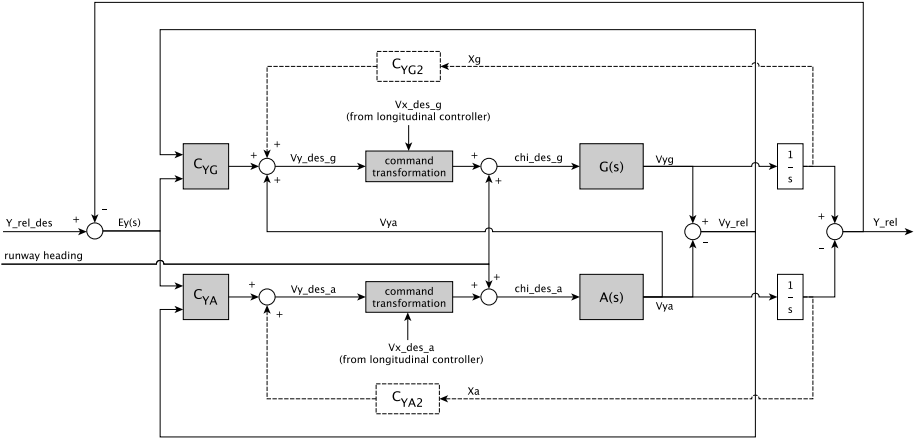


Figure 4.19 Performance-based lateral position synchronization controller with optional runway centerline tracking terms (dashed blocks C_{YG2} and C_{YA2}).

In comparison to the safety-based lateral control architecture from Figure 4.8, the performance-based controller contains a velocity feedback term V_{ya} which acts as a damper on the relative lateral motion the same way it was implemented for the longitudinal synchronization controllers shown in Figure 4.7 and 4.18. The optional runway centerline tracking terms C_{YG2} and C_{YA2} (dashed blocks in Figure 4.19) can be used to drive the coupled

vehicles back to the runway centerline after a disturbance. During the simulated landing experiments with the demonstrator setup, these terms were neglected due to the relatively large runway width of 20m. The command transformation blocks are implemented in the same way as described for the longitudinal controller, according to Equation 4.71.

The overall ground speed command sent to the ground vehicle speed controller is therefore calculated as:

$$V_{k,g}^{des} = \sqrt{(V_{x,a} + C_{XG}(X_{rel}^{des} - X_{rel}))^2 + (V_{y,a} + C_{YG}(Y_{rel}^{des} - Y_{rel}))^2} \quad (4.72)$$

while the control command sent to the aircraft's ground speed tracking controller (as defined in Section 4.2.2) is:

$$V_{k,a}^{des} = \sqrt{(V_{x,land} + C_{XA}(X_{rel}^{des} - X_{rel}))^2 + (C_{YA}(Y_{rel}^{des} - Y_{rel}))^2} \quad (4.73)$$

where C_{XG} , C_{YG} , C_{XA} , and C_{YA} were implemented as PID-controllers. As explained in Section 2.4.5, PID control was chosen mainly due to its extensive usage in aerospace applications (e.g. [78]), its physical interpretability and thus suitability for field experiments, its certifiability and its disturbance rejection capabilities [83].

The course commands for the ground vehicle and aircraft are calculated as follows:

$$\chi_g^{des} = \arctan\left(\frac{V_{y,a} + C_{YG}(Y_{rel}^{des} - Y_{rel})}{V_{x,a} + C_{XG}(X_{rel}^{des} - X_{rel})}\right) + \psi_{rwy} \quad (4.74)$$

$$\chi_a^{des} = \arctan\left(\frac{C_{YA}(Y_{rel}^{des} - Y_{rel})}{V_{x,land} + C_{XA}(X_{rel}^{des} - X_{rel})}\right) + \psi_{rwy} \quad (4.75)$$

where ψ_{rwy} represents the runway heading for the given landing approach direction. The design of the safety-based and performance-based position synchronization controllers will be presented in the following section.

4.3.3 Time Delay Robust Landing Control

The presence of time delay in the feedback loops of the performance-based landing control system was identified as a possible cause of instability early on in Chapter 2.4.5. The stability assessment, which will be performed in Section 4.4.3, has shown that rather high communication delays of more than 5s are tolerable for the given demonstrator setup. The physical delays due to wireless communication typically range in the area of 50ms and less, but the overall signal routing and processing can induce higher delays, as can be seen in Figure 4.1. In extreme cases, where large amounts of data (e.g. live telemetry and video streaming) are being transferred over the same communication channel, or where range limits are being reached and high package loss is present, even temporary high delays of more than 5s are possible. This has been observed during flight experiments at various occasions, where large amounts of data were streamed from the vision PC on the aircraft to the ground station over the same wireless network, temporarily increasing the estimated delays up to around 7s. Such high delays might drive the landing system to instability and

thereby jeopardize the system's operational safety. Although the issue with the vision PC was solved, similar situations might occur requiring the presence of a fail-safe mechanism to guarantee system safety. The control strategy chosen in this work is the *Time Domain Passivity Control Approach* (TDPA), which is based on the monitoring of energy flows in the system, and actively dissipating any given active energy by means of adaptive damping elements. In such way, overall system passivity and consequently stability is enforced. TDPA was chosen due to its known robustness properties, low conservatism, its physical interpretability, and its flexibility regarding implementation [53, 105]. Throughout this section the concept will be presented and its application to the landing control system described. A performance analysis demonstrating the fail-safe behavior in the presence of high time delays will be presented in Chapter 5.4.2.

Time Domain Passivity Control Approach

The Time Domain Passivity Control Approach was first developed in the field of teleoperation, where a remote manipulator (e.g. in space) is operated by a human via a delayed communication channel. Thereby, the teleoperation system is typically used in a bilateral setup, where the human operator exerts torques on a haptic (master) device, which are then transformed into position (or equivalently velocity) commands and transmitted to the controller of the remote manipulator (slave) through a communication channel. The slave controller transforms the received velocity commands into force commands, which are then applied to the remote environment. At the same time the commanded (or measured) force is sent back through the communication channel to the haptic device on the master side, which can then be felt by the human operator. This concept is known as *position-force computed (P-Fc)* teleoperation architecture and is illustrated in Figure 4.20:

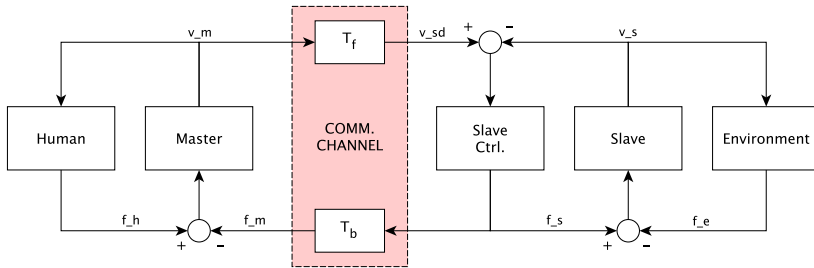


Figure 4.20 Block diagram of a classical *position-force computed (P-Fc)* teleoperation architecture.

where f_s and f_m represent the computed slave force and the delayed feedback force to the master, v_m and v_s the master and slave velocities, f_h and f_e the forces applied by the human operator and the environment, and T_f and T_b the forward and backward time delays.

The delayed signals v_{sd} and f_m are defined as follows:

$$v_{sd}(t) = v_m(t - T_f) \quad (4.76)$$

$$f_m = f_s(t - T_b) \quad (4.77)$$

where v_{sd} represents the desired slave velocity. The slave controller is typically implemented as PD-controller on position, or equivalently as PI-controller on velocity.

It should be noted that even though the P-Fc architecture is not directly applicable to the landing scenario, where precise bilateral position synchronization is required rather than a highly transparent haptic feedback, it shall be presented in detail here, as it best demonstrates the basic concepts behind TDPA. In a second step, the so-called *position-position architecture (P-P)* will be introduced, which will form the basis for the adaptation of the method to the landing application.

As can be seen in Figure 4.20, the feedback loops are closed via a communication channel, which naturally introduces time delay to the system. For such a delayed closed-loop system overall stability cannot be guaranteed, even if the individual sub-systems are stable. One approach to solve this problem consists in considering the overall system passivity, which represents a sufficient condition for stability, since the interconnection of passive systems is also passive [3]. In a typical TDPA setup the communication channel is the only non-passive element, while the master and slave devices are assumed to be passive. The energy flows into and out of the communication channel are then monitored by means of a so-called *passivity observer (PO)*, while any observed active energy is removed via an adaptive damping element, the *passivity controller (PC)*.

One important requirement of TDPA consists in the fact that flow and effort pairs, so-called *power conjugated variables*, must be used as control variables which are then transferred back and forth through the communication channel to allow for the calculation of the required energy flows. In the case of mechanical systems, these pairs include forces and velocities, where energy is calculated as the integral of the product of force and velocity. The identification of the corresponding flow and effort pairs can be difficult to retrieve from a block diagram for certain system setups, which are more complex than the P-Fc architecture shown in Figure 4.20. It is therefore useful to express the system architecture in a circuit representation using electro-mechanical analogies (current-velocity, voltage-force), which then allows for directly identifying the correct power-conjugated pairs at the ports of the communication channel [3]. The circuit representation of the P-Fc teleoperation system from Figure 4.20 is shown in Figure 4.21, where f_h , f_s , and f_e are the forces applied by the human, the slave controller, and the environment, v_m and v_s the velocities of the master and slave devices, v_{sd} the delayed master velocity used as reference for the slave controller, and f_m the delayed slave computed force applied to the master device. Z_m , Z_s , and Z_e are the impedances of the master device, the slave device, and the environment, which have been represented as resistances here. K_d and K_p are the derivative and proportional gains of the slave position controller (equivalent to PI-control on velocity).

The communication channel is represented by a pair of *Time Delay Power Networks (TDPNs)*, which mask the underlying communication elements (delayed dependent flow and effort sources) used for the transmission of the power-conjugated variables. TDPNs

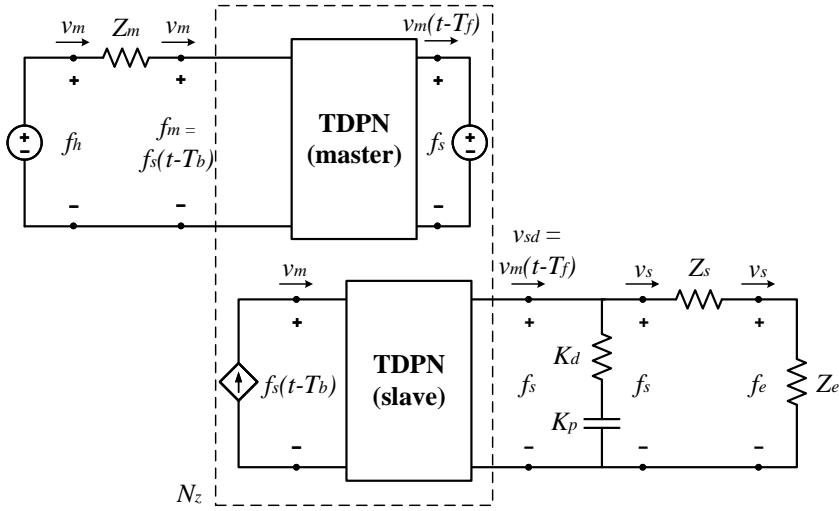


Figure 4.21 Circuit representation of position-force computed (P-Fc) teleoperation architecture.

are two-port networks, which can be used to represent effects such as varying time delays, package loss, and jitter, and constitute the basis for a generalized stability treatment for any type of time-delayed system [3]. To guarantee stability, both the master and slave side TDPNs must be passivated.

The port variables for both, the master and slave side TDPNs are $f_m(t) = f_s(t - T_b)$ and $v_m(t)$ at the left port, while the variables at the right port include $f_s(t)$ and $v_{sd} = v_m(t - T_f)$. It should be noted that such obvious mapping is not given for other architectures (e.g. P-P architecture), which makes it more difficult to identify the correct power-conjugated pairs for passivity checking. More details on this issue will be provided in Section 4.3.3.

Passivity of Time Delay Power Networks

The introduction of TDPNs in the last section allows for investigating the stability of the teleoperation system by analyzing the energy flows at the ports of the TDPN. Figure 4.22 illustrates the energy flow through the TDPN, where E^M and E^S are the accumulated energies on the left (master) and right (slave) hand side of the TDPN respectively, $f_1(k)$, $v_1(k)$, and $f_2(k)$, $v_2(k)$ the force and velocity pairs at instant k on the left and right ports.

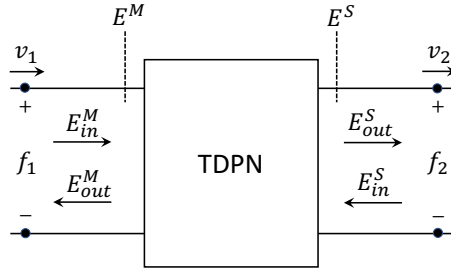


Figure 4.22 Energy flow through the TDPN.

The TDPN will be passive if:

$$E^N(k) = E^M(k) + E^S(k) \geq 0, \quad \forall k \geq 0 \quad (4.78)$$

where:

$$E^M(k) = E_{in}^M(k) - E_{out}^M(k) = \Delta T \sum_{j=0}^k f_1(j)v_1(j), \quad \forall k \geq 0 \quad (4.79)$$

$$E^S(k) = E_{in}^S(k) - E_{out}^S(k) = \Delta T \sum_{j=0}^k -f_2(j)v_2(j), \quad \forall k \geq 0 \quad (4.80)$$

$E_{in}^M(k)$, $E_{out}^M(k)$, $E_{in}^S(k)$, and $E_{out}^S(k)$ represent the energy flows into and out of the TDPN at time step k on the master and slave side respectively. These quantities constitute monotonically increasing and positive defined functions, which can be calculated as follows:

$$E_{in}^M(k) = \begin{cases} \Delta T f_1(k)v_1(k) + E_{in}^M(k-1), & \text{if } f_1(k)v_1(k) > 0 \\ E_{in}^M(k-1), & \text{else} \end{cases} \quad (4.81)$$

$$E_{out}^M(k) = \begin{cases} -\Delta T f_1(k)v_1(k) + E_{out}^M(k-1), & \text{if } f_1(k)v_1(k) < 0 \\ E_{out}^M(k-1), & \text{else} \end{cases} \quad (4.82)$$

$$E_{in}^S(k) = \begin{cases} \Delta T f_2(k)v_2(k) + E_{in}^S(k-1), & \text{if } f_2(k)v_2(k) > 0 \\ E_{in}^S(k-1), & \text{else} \end{cases} \quad (4.83)$$

$$E_{out}^S(k) = \begin{cases} -\Delta T f_2(k)v_2(k) + E_{out}^S(k-1), & \text{if } f_2(k)v_2(k) < 0 \\ E_{out}^S(k-1), & \text{else} \end{cases} \quad (4.84)$$

Unfortunately Equation (4.78) cannot be directly evaluated in a real (delayed) system, since the real-time values of the energies at either side of the TDPN are not available at the same time. It is therefore necessary to observe the energy flows at each port individually, which can be accomplished by taking the direction of the energy flows into account, as

will be shown in the following. Equation (4.78) can be rearranged as follows:

$$E^N(k) = E_{in}^M(k) - E_{out}^S(k) + E_{in}^S(k) - E_{out}^M(k) = E^{L2R}(k) + E^{R2L}(k) \geq 0, \quad \forall k \geq 0 \quad (4.85)$$

where $E^{L2R}(k)$ and $E^{R2L}(k)$ represent the left-to-right and right-to-left energy flows respectively. A sufficient condition to fulfill Equation (4.85) and ensure overall passivity is now given by:

$$E^{L2R}(k) \geq 0, \quad \forall k \geq 0 \quad (4.86)$$

$$E^{R2L}(k) \geq 0, \quad \forall k \geq 0 \quad (4.87)$$

Since Equations (4.86) and (4.87) are not observable on one side of the TDPN in a delayed system, the following observable conditions are used instead:

$$E_{obs}^{L2R}(k) = E_{in}^M(k - T_f(k)) - E_{out}^S(k) \geq 0, \quad \forall k \geq 0, \quad (4.88)$$

$$E_{obs}^{R2L}(k) = E_{in}^S(k - T_b(k)) - E_{out}^M(k) \geq 0, \quad \forall k \geq 0, \quad (4.89)$$

where $E_{obs}^{L2R}(k)$ represents the energy flow from left to right, observed on the right side of the TDPN, and $E_{obs}^{R2L}(k)$ the energy flow from right to left, observed on the left side of the TDPN. $T_f(k)$ and $T_b(k)$ are the forward and backward delays, respectively. It must now be shown that fulfilling Equations (4.88) and (4.89) is a sufficient condition to fulfill Equations (4.86) and (4.87). With the energy flows E_{in}^M and E_{in}^S entering the TDPN being monotonic and positive defined functions, their delayed values will always be upper bounded by their undelayed values:

$$E_{in}^M(k - T_f(k)) \leq E_{in}^M(k), \quad \forall k \geq 0 \quad (4.90)$$

$$E_{in}^S(k - T_b(k)) \leq E_{in}^S(k), \quad \forall k \geq 0 \quad (4.91)$$

which leads to the observed energy flows from Equations (4.88) and (4.89) being upper bounded by their actual values:

$$E_{obs}^{L2R}(k) \leq E^{L2R}(k), \quad \forall k \geq 0 \quad (4.92)$$

$$E_{obs}^{R2L}(k) \leq E^{R2L}(k), \quad \forall k \geq 0 \quad (4.93)$$

This implies that the passivity conditions from Equations (4.86) and (4.87), and consequently also Equation (4.78) are fulfilled if:

$$E_{obs}^{L2R}(k) \geq 0, \quad \forall k \geq 0 \quad (4.94)$$

$$E_{obs}^{R2L}(k) \geq 0, \quad \forall k \geq 0 \quad (4.95)$$

Using Equations (4.88) and (4.89) these relations can be more intuitively expressed as:

$$E_{out}^S(k) \leq E_{in}^M(k - T_f(k)), \quad \forall k \geq 0 \quad (\text{observed on slave side}) \quad (4.96)$$

$$E_{out}^M(k) \leq E_{in}^S(k - T_b(k)), \quad \forall k \geq 0 \quad (\text{observed on master side}) \quad (4.97)$$

which implies that the flows out of the TDPN must be upper bounded by their respective input flows. It should be noted that the sufficient condition introduced in Equations (4.86) and (4.87) adds some conservatism to the TDPA, which is further increased by the error between observed and actual energy flows as shown in Equations (4.92) and (4.93). Nevertheless, the conservatism included in TDPA is still lower than in many other approaches and is usually outweighed by the flexibility and model independent applicability of the approach [8].

Passivity Observer and Passivity Controller

In the previous section the appropriate observable energy flows for checking system passivity have been identified. Now the passivity observer (PO) and passivity controller (PC) can be defined and added to the teleoperation system. The PO will check the passivity conditions defined in Equations (4.88) and (4.89) on each side (one PO per side) of the communication network, while taking into account previous energy dissipated by the master and slave PCs. The observed energies W_M on the master, and W_S on the slave side are:

$$W_M(k) = E_{in}^S(k - T_b(k)) - E_{out}^M(k) + E_{PC}^M(k - 1) \quad (4.98)$$

$$W_S(k) = E_{in}^M(k - T_f(k)) - E_{out}^S(k) + E_{PC}^S(k - 1) \quad (4.99)$$

where $E_{PC}^M(k - 1)$ and $E_{PC}^S(k - 1)$ are the energies dissipated by the master and slave PCs up to the previous time step.

The passivity controllers are located at each port of the TDPN and adaptively dissipate any active energy observed by the passivity observers. For the P-Fc architecture, an impedance-type PC is used on the master side to dissipate energy by modifying the force and conserving velocity, while an admittance-type PC is used on the slave side, modifying velocity, while conserving force. The placement of the PO-PC pairs is shown in Figure 4.23.

For the admittance-type PC on the slave side, the velocity command $\hat{v}_{sd}(k) = v_m(k - T_f(k))$ coming from the master is modified according to:

$$v_{sd}(k) = \hat{v}_{sd}(k) + \beta(k)f_s(k) \quad (4.100)$$

where β represents an adaptive damping coefficient, defined as:

$$\beta(k) = \begin{cases} 0 & \text{if } W_S(k) > 0 \\ -\frac{W_S(k)}{\Delta T f_s^2(k)} & \text{else, if } |f_s(k)| > 0 \end{cases} \quad (4.101)$$

and the energy dissipated by the PC on the slave side is

$$E_{PC}^S(k) = \Delta T \sum_{j=0}^k \beta(j)f_s^2(j). \quad (4.102)$$

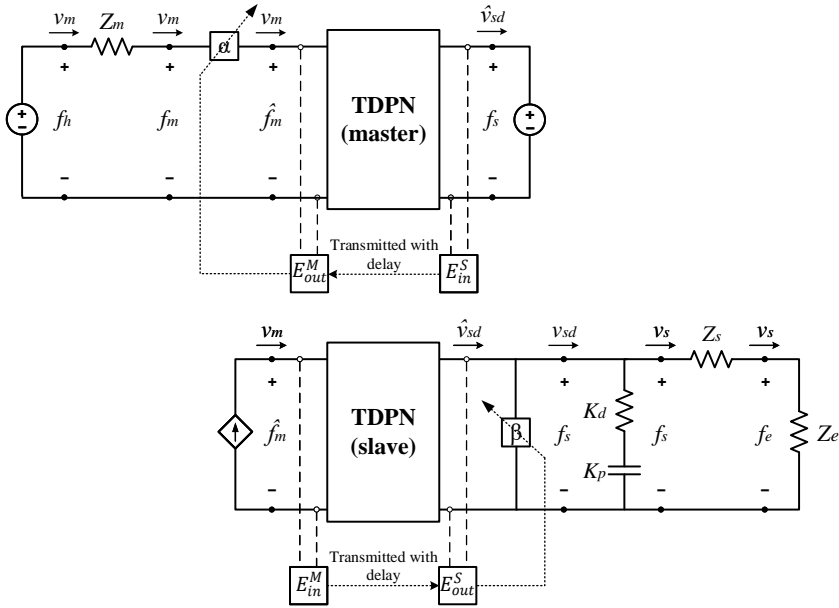


Figure 4.23 P-Fc teleoperation architecture with PO-PC pairs in impedance and admittance configuration on the master and slave side respectively.

In the same way the master side PC can be derived, where an impedance-type PC modifies the force command received from the slave while conserving velocity:

$$f_m(k) = f_{md}(k) + \alpha(k)v_m(k) \quad (4.103)$$

where α represents an adaptive damping coefficient, defined as:

$$\alpha(k) = \begin{cases} 0 & \text{if } W_S(k) > 0 \\ -\frac{W_M(k)}{\Delta T v_m^2(k)} & \text{else, if } |v_m(k)| > 0 \end{cases} \quad (4.104)$$

It should be noted that ideal dependent flow and effort sources are used in combination with the TDPNs in Figures 4.21 and 4.23, which are capable of providing and absorbing infinite amounts of energy. Any active energy flowing towards the source will therefore not affect the overall system passivity [6], thus only requiring passivity control at the opposite port of the flow and effort sources. Additionally, it is important to highlight the validity of energy-based TDPA in the presence of variable time delays and package loss [105]. Due to the monotonic and positive defined nature of the energy flows entering the communication network, as defined Equations (4.90) and (4.91), the delayed quantities will always be upper bounded by their undelayed values, thus ensuring passivity.

Also, TDPA is in general less conservative than other available methods, where the amount of dissipated energy is set to a pre-defined worst-case value (e.g. wave variable

approach [94, 8]). The adaptive passivity controller used in TDPA only removes the exact (and thus minimal) amount of active energy, continuously estimated by the passivity observer. Also, energy-based TDPA was found to be less conservative and thus more efficient than power-based TDPA [129]. Overall, only little knowledge of the involved systems and external forces and moments is required for a use of TDPA, which leads to a straightforward implementation and allows for robust performance even under varying time delay, jitter, and package loss. Other control methods struggle to cope with such communication channel properties, which are difficult to model and predict. A downside of all passivity-enforcing controllers consists in a possible mismatch of perceived task impedances and therefore reduced transparency (for a definition of transparency in the teleoperation context see [79]), but this issue is not of particular importance for the landing scenario considered here, since haptic feedback is not required.

Position-Position Architecture

The P-Fc architecture presented in the previous sections is not applicable to the landing application, since force commands cannot be directly applied to either of the involved vehicles, where available control inputs include desired velocities, course angles and flight path angle (on the aircraft side). Also, precise bilateral position synchronization between aircraft and ground vehicle must be achieved and maintained throughout the whole landing maneuver. Such task requires the exchange of position information in both directions, which differs from the P-Fc approach, where only a force signal is sent back to the master side. As previously mentioned, a variation of the P-P teleoperation architecture could be suitable here, where position data is exchanged in both directions. Nevertheless, the use of vehicle positions (or equivalently velocities) as control variables, instead of the power-conjugated pairs velocity and force used in the P-Fc architecture, does not allow for calculating the relevant energy flows from the system inputs and outputs directly (which are now all velocities). The identification of the correct flow and effort pairs for passivity checking is thus not obvious for the P-P architecture. The system must first be transferred into a suitable circuit representation, similar to the P-Fc case shown in Figure 4.21, to allow for the identification of the correct power-conjugated pairs and hence the stability treatment of the system [5].

A short overview of the P-P teleoperation concept shall be given in the following, before its adaptation to the landing scenario is presented in Section 4.3.3. Figure 4.24 illustrates the block diagram of the P-P teleoperation architecture.

The signals f_m and f_s represent the computed master and slave forces, v_m and v_s the master and slave velocities, f_h and f_e the forces applied by the human operator and the environment, and T_f and T_b the forward and backward time delays. The delayed signals v_{sd} and f_m are defined as follows:

$$v_{sd}(t) = v_m(t - T_f) \quad (4.105)$$

$$v_{md}(t) = v_s(t - T_b) \quad (4.106)$$

where v_{md} and v_{sd} represent the desired master and slave velocities respectively. The master and slave controllers are typically implemented as PI-controllers on velocity, or

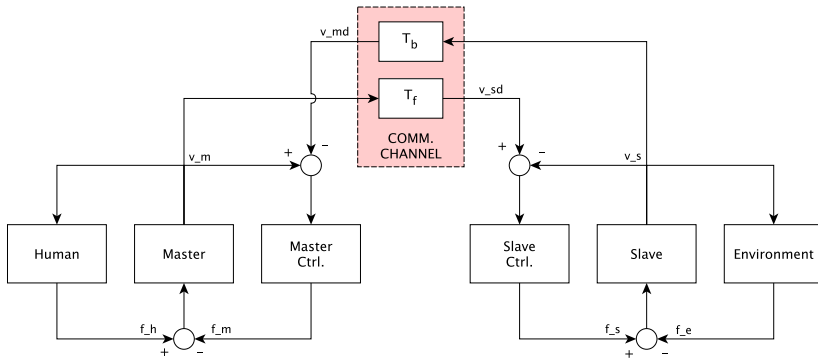


Figure 4.24 Block diagram of P-P teleoperation architecture.

equivalently as PD-controllers on position.

As can be seen in Figure 4.24, the block diagram of the P-P architecture does not provide sufficient information about the causality (input / output correlation of flow and effort variables) of the communication channel in terms of energy flows. The circuit representation of the P-P architecture, augmented with a pair of TDPNs, is therefore introduced as shown in Figure 4.25.

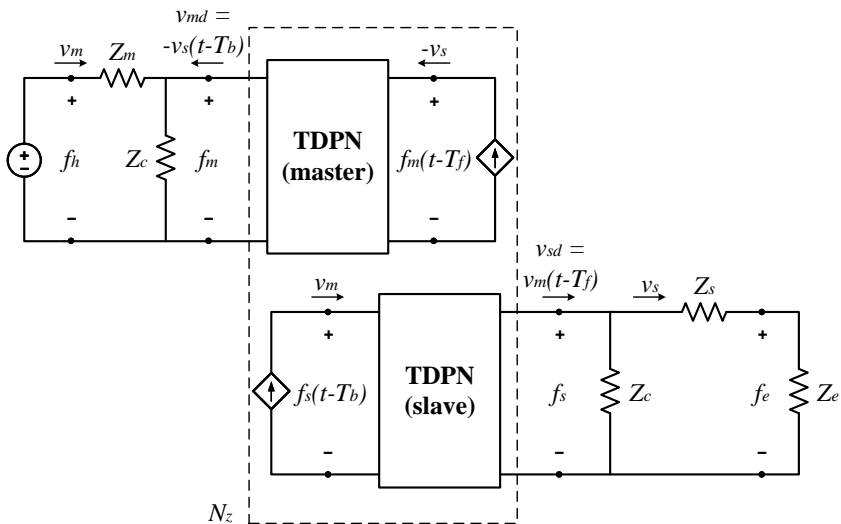


Figure 4.25 Circuit representation of P-P teleoperation architecture with TDPN-augmented communication network.

The port variables for the master side TDPN are now found as $f_m(t)$ and $v_s(t - T_b)$ at the left port, while the variables at the right port include $f_m(t - T_f)$ and $v_s(t)$.

The port variables for the slave side TDPN are $f_s(t - T_b)$ and v_m at the left port, while the variables at the right port include $f_s(t)$ and $v_m(t - T_f)$.

It can easily be seen that the master and slave side TDPNs exhibit different power-conjugated pairs, which was not the case for the P-Fc architecture shown in Figure 4.21. The circuit representation is thus an important tool for the derivation of power-port conform electrical network structures. It should also be noted that now the exchanged force signals have no direct impact on the vehicle's motion control systems, but are used for passivity control purposes only.

Once the correct network representation of the system is found, the application of appropriate PO-PC pairs for the passivation of the communication network is straightforward, as previously shown for the P-Fc architecture. The main difference is that for the P-P architecture admittance-type PCs must be used on both the master and slave sides. It should be noted that the use of admittance type passivity controllers, which modify the received velocity commands in case active energy is detected, leads to the generation of position drifts between master and slave. Position synchronization can be re-established by means of so-called *drift compensators*, which add the removed energy back to the system once an *energy-gap* is available [18, 4]. As will be shown in the next section, such position drift compensation can be provided by the position synchronization controllers on the aircraft and ground vehicle, and no additional functions will be required. The working principle of the cooperative landing controllers was even used to develop a new position drift compensator for classical teleoperation [21].

TDPA for Landing Control

The cooperative control architecture presented in Section 4.3.2 must now be transferred to a suitable circuit representation to identify the correct power-conjugated pairs for passivity checking. The developed circuit representation is shown in Figures 4.26 and 4.27 for the aircraft (index a, UAV) and ground vehicle (index g, UGV) sides, respectively.

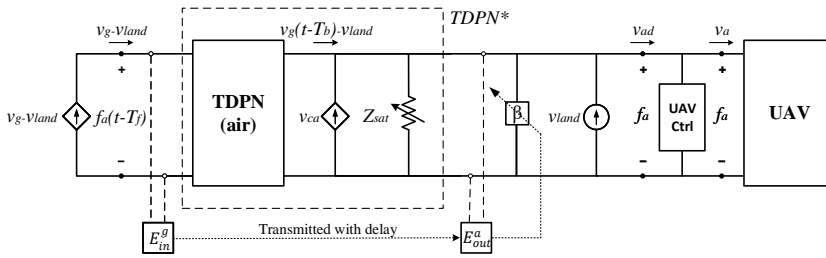


Figure 4.26 Circuit representation of aircraft position synchronization controller with passivity controller and augmented TDPN-based communication network representation.

The representation shown in Figures 4.26 and 4.27 resembles the P-P architecture presented in the previous section, where both systems share velocity signals and have their own controllers to track the velocity of the other system received through the channel.

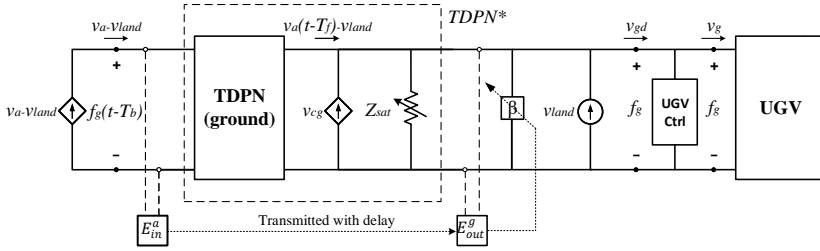


Figure 4.27 Circuit representation of ground vehicle position synchronization controller with passivity controller and augmented TDPN-based communication network representation.

In contrast to the P-P teleoperation concept no human operator is present in the landing application, which renders the classical master / slave assignments irrelevant, since both vehicles act as master and slave devices simultaneously in a distributed setup.

In the landing context, the forward time delay T_f refers to the communication delay from the aircraft to the ground vehicle, while the backward delay T_b refers to the opposite direction.

As previously mentioned, each vehicle transmits its current velocity through the communication channel, which serves as reference to the other vehicle's position controller. The current ground speeds of aircraft and ground vehicle are v_a and v_g , respectively, where the index k was dropped in favor of a concise notation. The commanded aircraft and ground vehicle forces f_a and f_g , which are controller-internal variables, are also exchanged in order to monitor the energy flow of the channel.

v_{land} represents the desired landing velocity with respect to the ground, which is assumed to be constant and independent of the vehicle states. Consequently, v_{land} does not have to be checked for passivity and can be added after the passivity controller in the circuit representation as a constant current source. The overall velocity command forwarded to the vehicles thus consists of two parts — the constant landing velocity v_{land} , which is not a source of instability, and the velocity commands v_{ca} and v_{cg} provided by the vehicle's cooperative controllers, which are functions of the relative position between the vehicles. The relative position vector is calculated based on delayed data received via the communication channel, which introduces time delay, jitter, and package loss to the system. Due to its potential to cause instability, the velocity commands from the cooperative controllers must also be covered by the PO-PC.

The passivity controller is represented as variable impedance β in Figures 4.26 and 4.27.

Since saturation blocks are used in the vehicle controllers to keep the vehicles within safe velocity ranges, variable impedances Z_{sat} were added to both circuits. Those impedances are infinite in case the velocities are within the safe ranges. In case the commanded velocities are too high, the value of Z_{sat} is adjusted to comply with the admissible command value ranges.

UAV Ctrl and *UGV Ctrl* represent the vehicle controllers, whose inputs are desired ground speeds.

The velocities flowing towards the vehicle controllers after Z_{sat} are therefore saturated versions of $v_{ca} + v_g(t - T_b) - v_{land}$ for the aircraft, and $v_{cg} + v_a(t - T_f) - v_{land}$ for the ground vehicle, where in accordance to Equations (4.72) and (4.73), the position synchronization commands v_{ca} and v_{cg} are defined as:

$$v_{ca} = -v_g(t - T_b) + v_{land} + \sqrt{(C_{XA}(\hat{x}_{rel,a}))^2 + (C_{YA}(\hat{y}_{rel,a}))^2} \quad (4.107)$$

$$v_{cg} = -v_a(t - T_b) + \sqrt{(v_{xa} + C_{XG}(\hat{x}_{rel,g}))^2 + (C_{YG}(\hat{y}_{rel,g}))^2} \quad (4.108)$$

where C_{XA} , C_{YA} , C_{XG} , and C_{YG} refer to the cooperative position synchronization controllers on the aircraft and ground vehicle respectively, as defined in Section 4.3.2. In contrast to Equations (4.72) and (4.73) the relative positions $\hat{x}_{rel,a}$ and $\hat{x}_{rel,g}$ are directly fed back, thus removing X_{rel}^{des} and Y_{rel}^{des} from the equations. This implies desired relative x and y positions of zero, which represents the typical use case. Also, v_{land} in Equation (4.107) was added after the command transformation, which is equivalent to adding $v_{x,land}$ before the command transformation as shown in Figure 4.18 and Equation (4.73). This was done to simplify the application of the circuit representation to the system, where v_{land} can be treated as a constant parameter.

$\hat{x}_{rel,a}$, $\hat{x}_{rel,g}$, $\hat{y}_{rel,a}$, and $\hat{y}_{rel,g}$ represent the estimated values of the current relative vehicle position, calculated on each vehicle individually, by extrapolating the other vehicle's delayed position as previously introduced in Section 4.1.1. The relative vehicle position estimates in the longitudinal direction are calculated as:

$$\hat{x}_{rel,a} = x_g(t - T_b) + T_b v_{xg}(t - T_b) - x_a \quad (4.109)$$

$$\hat{x}_{rel,g} = x_g - x_a(t - T_f) - T_f v_{xa}(t - T_f) \quad (4.110)$$

where analog expressions can be derived for the lateral direction. It should be noted that each vehicle's position information is transmitted together with its velocity and energy information, and is not derived by integrating the received velocity signal. This helps to increase the system's robustness against package loss, since a loss of velocity data would lead to a permanent drift in the integrated position, while lost position data would only affect the system temporarily during the package loss. It should also be noted that full position and velocity vectors are transmitted over the channel, where all vector components are accessible after reception (e.g. v_{xa} , v_{ya} , v_{za}), while the circuit representation shown in Figures 4.26 and 4.27 is based on the scalar values relevant for passivity control.

As can be seen in Equations (4.107) and (4.108), the commanded inertial velocities v_{ca} and v_{cg} depend on the relative vehicle states in x and y direction. Therefore these velocity commands contain the longitudinal and lateral motion information and can be monitored by a single PO-PC pair per vehicle. The passivation of the course commands is therefore not necessary. Also no passivation of the vertical motion is required, since only the aircraft controls the vertical direction, which corresponds to an open loop system. If an actuated landing platform with active vertical control was added in a future development, the z -axis

would have to be taken into account.

It is important to note that the signals $v_g(t - T_b) - v_{land}$ and $v_a(t - T_f) - v_{land}$ transmitted through the forward and backward communication channels (see Figures 4.26 and 4.27), and relevant for passivity control, are modified on the receiving side by subtracting the first two terms of the cooperative controllers v_{ca} and v_{cg} , as defined in Equations (4.107) and (4.108). Only the saturated version of the last term is left to be checked by the passivity controller, before it is forwarded to the lower-level vehicle controllers. This way not only the communication channel is passivated, but also the cooperative position synchronization controllers. Such setup can be interpreted as an augmented TDPN (dashed boxes in Figures 4.26 and 4.27), where the in and out energy flows are passivated by the PO-PC pairs at the TDPN ports in the same way as shown for the P-P architecture. This system architecture was chosen to allow for the adaptation of the TDPA concept to the landing control system, and does not imply any limitations regarding its applicability. It should be noted that for the undelayed case and perfectly synchronized vehicles the in and out energy pairs will match, and the transmitted signals will remain untouched by the PC. As will be shown during the stability assessment of the delayed system in Section 4.4.3, the specific demonstrator setup remains stable for time delays of more than 5s. Given this rather large margin, it was decided to activate the passivity controller only for time delays above 0.5s, thus mitigating the effects of conservatism inherent to passivity-based control. One should also note that the velocities checked by the PO-PC pairs on both sides of the TDPNs represent variations of the vehicle velocities around v_{land} on one side, and variation of the commanded velocities around v_{land} on the other side. This operating point shift (subtraction of v_{land} from the absolute velocity signals) was necessary to allow for the distinction of inward and outward energy flows based on the sign of the power conjugated signal, as is usual in classical teleoperation setups [105]. The system thus resembles a P-P teleoperation system where commanded and actual velocities are used in the energy computations. Those velocities are multiplied by the commanded forces and their delayed versions in order to monitor the energy flow of the system.

Force-Based Vehicle Control

As explained in the previous sections, one requirement for the application of TDPA is that power conjugated variables (velocity and force) are used as control signals. Unlike most robotic manipulators, classical aircraft and ground vehicle control systems do not allow for explicit force control, where instead velocity, course angle and flight path angle represent the available control variables. Nevertheless, the use of the energy-based flight controller introduced in Chapter 3.2, allows for retrieving the required force command. The ground vehicle does not allow for explicit force control either, which was circumvented by using model-based force estimation. For a further improvement the use of a force observer, which would allow for more precise force estimates, could be investigated to overcome this problem.

Integration in Overall Landing Control System

As will be analyzed in Section 4.4.3, the system will remain stable for rather high time delays of more than 5s. As the intention of the passivity control system developed throughout

this section is to serve as a fail-safe system, it only needs to be activated if higher time delays are observed. As previously mentioned, it was therefore decided to set the activation threshold to a measured time delay of $500ms$, where the landing state machine introduced in Section 4.2.6 monitors the current communication delay and activates the passivity controller once the limit is exceeded.

4.4 Control System Design

Throughout this section the cooperative landing controller will be designed and a linear stability and sensitivity analysis will be performed. This section is based on the *a priori* vehicle models in state space form, as provided in Chapter 3.5. The high fidelity models from system identification were not yet available at the time of this study, but the control system design procedure was applied in the same way to the high-fidelity models once they were available before the landing experiments took place. The results of the high-fidelity model simulation experiments will be presented in Chapter 5. Nevertheless, the conclusions drawn here are considered valid for the same *class* of vehicles, since the geometry-based modeling approach used to derive the *a priori* models was found to be accurate enough, with deviations below 20% in terms of aerodynamic parameters and below 10% for inertia parameters [64]. Also, the effects of time delay on the system stability will be analyzed and a fail-safe strategy will be presented.

Figure 4.28 illustrates the different controller modes and activation sequences during a safety-based (mode 1) and performance-based (mode 2) landing maneuver.

As can be seen, there are only two position tracking functions active during a safety-based approach, namely the *UGV position tracking (lateral)* function on the aircraft side and the *runway centerline tracking* function on the ground vehicle side. Both of these functions act in the lateral direction and are implemented as unilateral outer-loop controllers, which require appropriate tuning but do not pose any problem in terms of time delay dependent stability. The longitudinal approach functions are operated in open-loop and therefore do not require any tuning on motion coordination level. As can be seen in Figure 4.28 the lateral position tracking modes from the approach phase remain active during the different landing modes (flare, retard, ground lock), while the function *UAV position tracking (longitudinal)* is added on the ground vehicle side during the landing phase. The main task during the approach phase is to align the vehicle's positions in the longitudinal direction, while a lateral misalignment is tolerable during the early phase of the approach, since the vehicles are still far apart and have sufficient time left to achieve lateral synchronization. No special controller mode will therefore be developed for the lateral position tracking tasks during the approach phase — instead the lateral tracking controllers for the landing phase will be used during the approach as well. The position tracking controllers will therefore be tuned for the relative position regulation task of the landing phase, where the main goal is effective disturbance rejection. The same controller gains will be used for lateral position tracking during the approach phase.

For the performance-based landing maneuver a separation of approach and landing phase is more evident, since no PID-based position tracking controller will be active during the approach. The control commands during the approach phase will be calculated by the

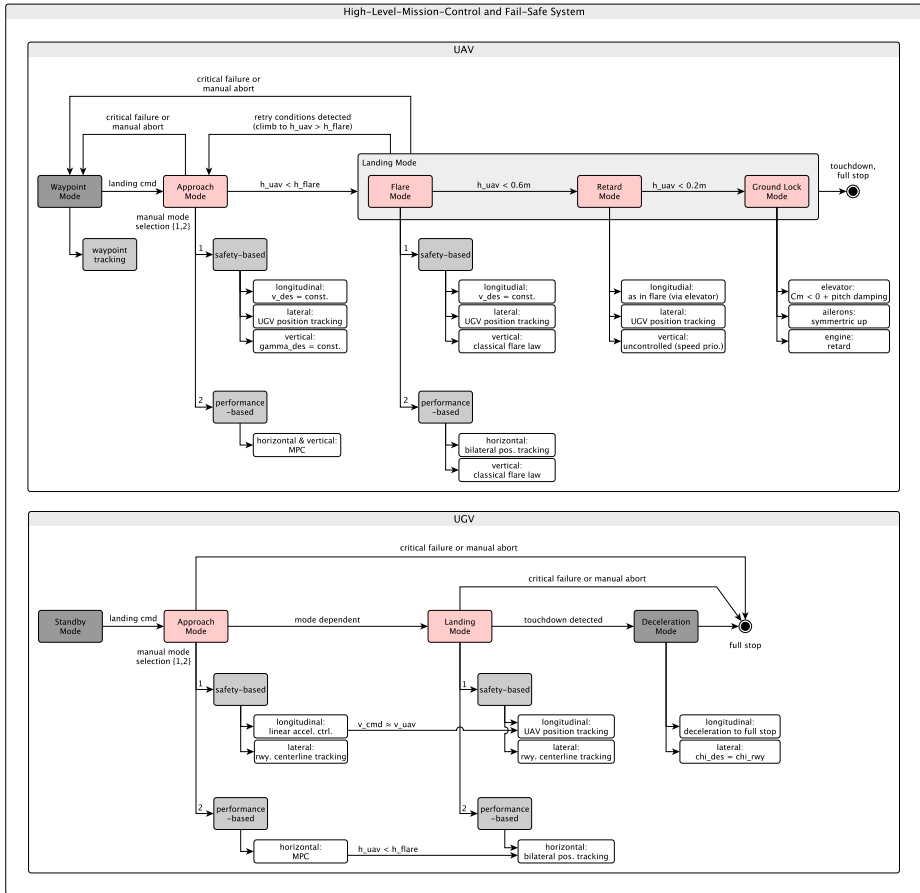


Figure 4.28 Controller modes and switching conditions during safety-based and performance-based landing maneuver.

optimization-based controller presented in Section 4.3.1, where its tuning was already performed. After the approach phase, the performance-based position synchronization controller is activated and remains active until the touchdown is achieved. During the retard phase (engine off) the horizontal position synchronization controller is still functional (speed via elevator), while the vertical flare controller is deactivated due to the missing engine control channel after engine retard.

It is therefore sufficient to tune the horizontal position synchronization controller for the safety-based and performance-based landing maneuver. These controllers will be tuned for disturbance rejection and not for reference tracking which might result in a reduced control performance when the landing maneuver is initiated with large lateral position errors. As previously mentioned, this can be tolerated since the vehicles are still far apart during the early approach.

It is now important to perform an appropriate tuning, which allows for a just comparison of the disturbance rejection characteristics of the two landing controllers variants (safety-based vs. performance-based). The approach taken in this work consists of performing a frequency domain pre-tuning of both architectures using the loop shaping methodology. In such way both controllers are tuned to present a similar frequency response, which also makes them similar in their dynamic behavior in the time domain [109]. This pre-tuning will then serve as starting point for an optimization-based tuning method in the time domain, where the nonlinear system response to a wind disturbance is recorded and a cost function assembled with different pre-defined performance metrics (e.g. maximum overshoot, rise time, settling time, ITAE). The cooperative controller gains are then iterated until the lowest cost is found. The following sections will describe the approach in more detail.

4.4.1 Frequency Domain Design

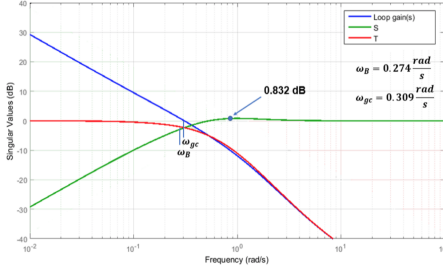
As previously mentioned, the first step consists in the tuning of both the safety-based and performance-based architectures to exhibit similar frequency responses. The open-loop frequency response can be characterized in terms of different functions and metrics — here the gain crossover frequency of the loop transfer function ω_{gc} (zero crossing of gain curve), the system bandwidth ω_B (sensitivity function crossing $-3dB$ from below), and the sensitivity peak (maximum value of sensitivity function) will be used [109]. Figures 4.29a and 4.29b illustrate the results of the loop shaping design for the safety-based and performance-based longitudinal position synchronization controllers and Figures 4.29c and 4.29d for the lateral controllers.

As can be seen in Figure 4.29, the frequency responses of both the safety-based and performance-based controllers are very similar, which implies similar dynamical behavior in the time domains as well. The main metrics used for the similarity assessment are the gain crossover frequency, the system bandwidth and the sensitivity peak. This tuning is considered to be a fair starting point for the subsequent optimization-based fine-tuning procedure in the time domain.

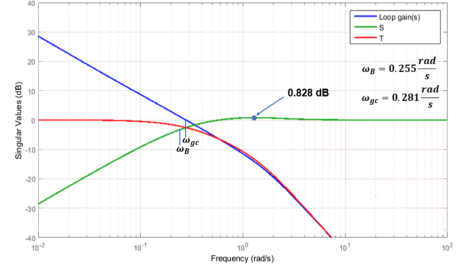
4.4.2 Time Domain Design

With both synchronization controllers tuned to present similar dynamics in the previous section, an optimization-based fine-tuning procedure is now applied. Therefore the system response to a disturbance is recorded for both controller structures (safety-based and performance-based) and a cost function established based on a number of time domain performance metrics. The different controller gains are then varied by solving a nonlinear optimization problem until the minimal cost is found. The setup used during the optimization looks as follows — a wind disturbance is applied to the vehicles, which are already aligned in the horizontal plane, while the aircraft is flying at constant altitude. The defined cost function is shown in Equation 4.111:

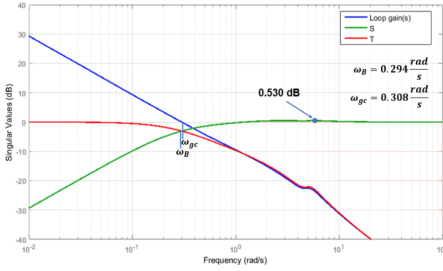
$$f(OS_{max}, US_{max}, |\dot{u}|, ITAE) = w_1 OS_{max} + w_2 US_{max} + w_3 |\dot{u}| + w_4 ITAE \quad (4.111)$$



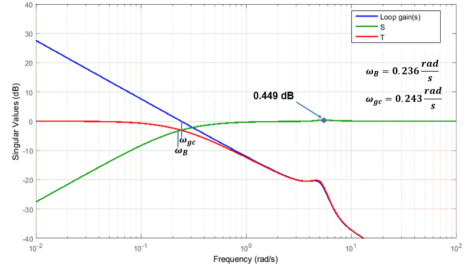
(a) Loop shape of safety-based longitudinal synchronization controller.



(b) Loop shape of performance-based longitudinal synchronization controller.



(c) Loop shape of safety-based lateral synchronization controller.



(d) Loop shape of performance-based lateral synchronization controller.

Figure 4.29 Frequency responses of safety-based and performance-based position synchronization controllers after loop shaping design.

where OS_{max} represents the maximum overshoot, US_{max} the maximum undershoot, \dot{u} the derivative of the control command, and ITAE a performance criteria based on the integral of the product of time and absolute error [54]. The Nelder-Mead method was used to solve the nonlinear constrained multivariable optimization problem [72]. Being a direct search method, this heuristic approach is not based on differentiation and gradient calculation, but evaluates only the function value.

The external disturbance used for the optimization is a *one-minus-cosine* wind gust model [15, 36], which is defined as shown in Equation (4.112):

$$V_{wind} = \begin{cases} 0, & x < 0 \\ \frac{V_m}{s} \left(1 - \cos\left(\frac{\pi x}{d_m}\right) \right), & 0 \leq x \leq d_m \\ V_m, & x > d_m \end{cases} \quad (4.112)$$

where V_m represents the gust amplitude, d_m the gust length, x the distance traveled, and V_{wind} the resulting wind velocity in the aircraft body frame.

To analyze the effectiveness of the optimization-based tuning method, the stability margins before and after its applications will be analyzed in Section 4.4.4. The stability

margins are expected to decrease after the optimization is performed, since the tuning should result in a more performant system with higher bandwidth.

The final value of the cost function after optimization is also an indicator for the success of the tuning method, since it represents the sum of different performance measures in the time domain. After applying the optimization-based tuning method to both architectures, the safety-based controller exhibited a final cost of 10505.6, while the performance-based controller ended up at 4134.98, representing less than half of the safety-based controller's value. It is possible to compare these values, since the same cost function and weights were used for tuning both architectures. It should be noted that since only a wind gust was used for perturbing the previously aligned vehicles, the tuning is optimized for disturbance rejection and not for setpoint tracking. Since the position synchronization controller's purpose is to maintain position alignment during the flare to touchdown phases, such tuning represents the appropriate approach. The lateral tracking functions will also be active during the safety-based approach, where a tuning for disturbance rejection might not result in the best performance. Nevertheless, as previously mentioned, a somewhat reduced performance is tolerable during the early phase of the approach, when the vehicles still are far apart. No specific tuning for the lateral approach controllers is therefore required. Tables 4.1 and 4.2 present the cooperative controller gains for the safety-based and performance-based position synchronization controllers respectively. It should be noted that the derivative gains of the longitudinal and lateral position synchronization controllers have different signs than the proportional and integral gains. This is due to the fact that the measured relative velocity is used as derivative term on the relative position directly, i.e. it is not subtracted from a desired relative velocity signal as is the case for the relative position feedback. A numerical differentiation of the relative position signal would lead to equal signs.

Table 4.1 Optimized position synchronization controller gains for safety-based architecture.

Operating direction	Controller gain	Optimized value
<i>Longitudinal UGV</i>	K_{pXG}	0.3388
	K_{iXG}	0
	$K_{p\dot{X}G}$	-0.0003787
<i>Lateral UAV</i>	K_{pYA}	-0.88948
	K_{iYA}	-21.5695
	$K_{p\dot{Y}A}$	0.20062

Table 4.2 Optimized position synchronization controller gains for performance-based architecture.

Operating direction	Controller gain	Optimized value
<i>Longitudinal UGV</i>	K_{pXG}	0.3844
	K_{iXG}	0
	$K_{p\ddot{X}G}$	-0.2701
<i>Lateral UGV</i>	K_{pYG}	0.7907
	K_{iYG}	0.002971
	$K_{p\ddot{Y}G}$	-1.6877
<i>Longitudinal UAV</i>	K_{pXA}	-0.2366
	K_{iXA}	-0.09508
	$K_{p\ddot{X}A}$	0.7221
<i>Lateral UAV</i>	K_{pYA}	-1.7616
	K_{iYA}	-0.00357
	$K_{p\ddot{Y}A}$	0.0589

4.4.3 Stability Assessment

Up to this point, the safety and performance-based landing controllers were designed for the undelayed case. In this section the effects of communication time delay on the overall system performance and stability are analyzed. Given the delay-independent stability properties of the safety-based (unilateral) controller, only the performance-based (bilateral) position tracking controller is considered here. A stability analysis is performed for both the longitudinal and lateral position synchronization controllers with the gains from Table 4.2, while only the longitudinal analysis is shown in full length. The lateral derivation is presented in a shortened version for the sake of brevity.

Longitudinal Stability Analysis

Since the analysis is performed using linear vehicle models, which have been linearized around a horizontal forward flight trim condition ($V_x^* = 21\text{m/s}$ and $V_y^* = 0\text{m/s}$), the directional coupling corrections term (*command transformation* block in Figure 4.18) is not considered during the block diagram reduction. Figure 4.30 shows the block diagram of the longitudinal performance-based position synchronization controller with the lag corrections terms as previously defined in Chapter 4.1.1. As explained in Chapter 4.1.1, the time delay can be directly estimated since both GPS receivers (on UAV and UGV)

are running on synchronized GPS time. The time scale resolution is limited by the GPS receiver's sampling rate, since a new measurement is only saved at multiples of the sampling time step ($50ms$ in case of the demonstrator setup). The goal is now to derive the transfer function from the system input V_{land} to the state X_{rel} (bold signals in Figure 4.30). The relative velocity feedback to the position synchronization controllers (previously used as derivative term) has been omitted in favor of a simpler derivation, and has been replaced by a classical derivative term.

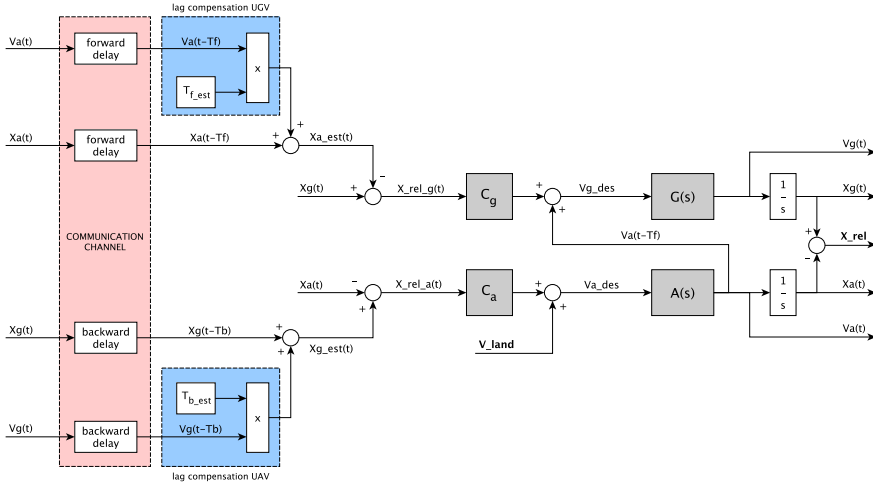


Figure 4.30 Longitudinal position synchronization controller with lag correction terms.

From Figure 4.30 the following relations can be derived:

$$X_g = \frac{G}{s} [V_a(t - T_f) + C_g(X_g - (X_a(t - T_f) + T_f V_a(t - T_f)))] \quad (4.113)$$

$$X_a = \frac{A}{s} [V_{land} + C_a(X_g(t - T_b) + T_b V_g(t - T_b) - X_a)] \quad (4.114)$$

and with:

$$V_g(t - T_b) = sX_g e^{-sT_b} \quad (4.115)$$

$$V_a(t - T_f) = sX_a e^{-sT_f} \quad (4.116)$$

the transfer functions from X_a to X_g and from V_{land} and X_g to X_a are found:

$$X_g = \frac{Ge^{-sT_f}(s - C_g(1 + sT_f))}{s - GC_g} X_a = H_{ga} X_a \quad (4.117)$$

$$X_a = \frac{A}{s + AC_a} V_{land} + \frac{AC_a e^{-sT_b}(1 + sT_b)}{s + AC_a} X_g = H_{av} V_{land} + H_{ag} X_g \quad (4.118)$$

Substituting Equation (4.117) into Equation (4.118) and vice versa yields:

$$X_g = H_{ga}H_{av}V_{land} + H_{ga}H_{ag}X_g \quad (4.119)$$

$$X_a = H_{av}V_{land} + H_{ag}H_{ga}X_a \quad (4.120)$$

and thus the transfer functions from V_{land} to X_g and X_a :

$$\frac{X_g}{V_{land}} = \frac{H_{ga}H_{av}}{1 - H_{ag}H_{ga}} \quad (4.121)$$

$$\frac{X_a}{V_{land}} = \frac{H_{av}}{1 - H_{ag}H_{ga}} \quad (4.122)$$

Subtracting Equation 4.122 from Equation 4.121 yields the transfer function from V_{land} to X_{rel} , which allows for the stability analysis of the coupled control system:

$$\frac{X_g}{V_{land}} - \frac{X_a}{V_{land}} = \boxed{\frac{X_{rel}}{V_{land}} = \frac{H_{av}(H_{ga} - 1)}{1 - H_{ag}H_{ga}}} \quad (4.123)$$

The transfer functions for the longitudinal controlled vehicle models $G(s)$ and $A(s)$ have been derived from the nonlinear vehicle models introduced in Chapter 3, and were approximated as second order functions, as shown in Equations 4.124 and 4.125:

$$G(s) = \frac{0.1181}{s^2 + 0.5345s + 0.1181} \quad (4.124)$$

$$A(s) = \frac{0.4943}{s^2 + 0.949s + 0.4943} \quad (4.125)$$

The longitudinal position synchronization controllers C_g and C_a are implemented as PID-controllers with the gains shown in Table 4.2, the exponentials in the transfer functions were expressed as rational functions using a Padé approximation of order 5 [121]. The pole zero map and step response plot for the coupled system with increasing time delays are shown in Figures 4.31 and 4.32 respectively.

As can be seen in Figure 4.31 the maximum time delay before the system becomes unstable is found to be 5.7s. The undelayed system is asymptotically stable — most of its poles are scattered over a large area in the left half plane as shown in Figure 4.31, while a few poles are located close to the origin. The step response for the undelayed case confirms the stable behavior as can be seen in Figure 4.32.

Lateral Stability Analysis

Figure 4.33 illustrates the block diagram for the lateral stability analysis. As can be seen, the command transformation terms have been maintained to produce control commands in terms of course angle, as required by the low-level vehicle controllers. For the lateral analysis the transfer function from the system input Y_{rel}^{des} to the state Y_{rel} (bold signals in Figure 4.33) must be derived. As for the longitudinal case, the relative velocity feedback

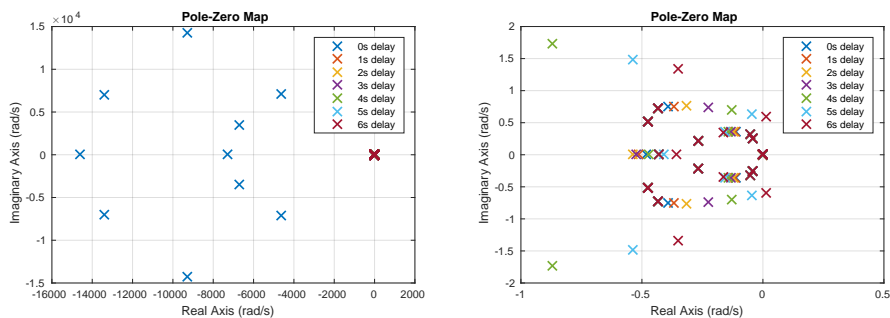


Figure 4.31 Pole-zero plot for transfer function X_{rel}/V_{land} of performance-based position synchronization controller with increasing communication time delays. The system remains stable for time delays of up to 5.7s.

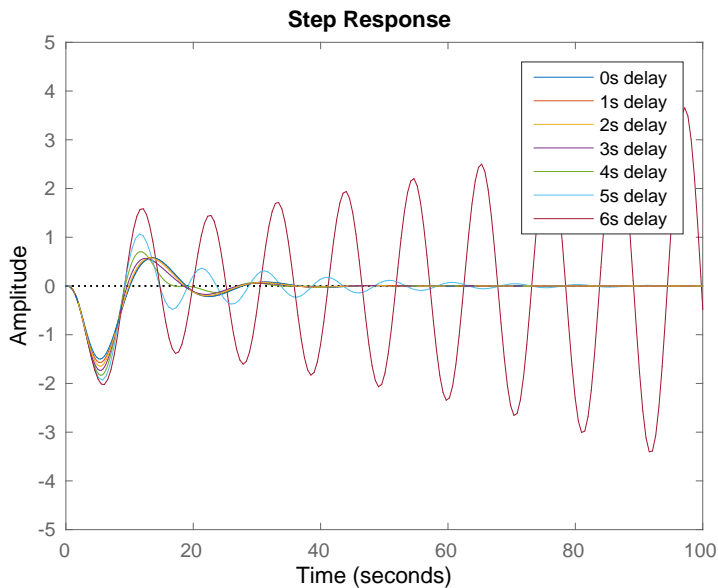


Figure 4.32 Step responses for transfer function X_{rel}/V_{land} of performance-based position synchronization controller with increasing communication time delays. The system remains stable for time delays of up to 5.7s.

to the position synchronization controllers (previously used as derivative term) has been omitted in favor of a simpler derivation, and has been replaced by a classical derivative term.

Since linear models are used for the analysis, where the linearization was performed around a horizontal forward flight trim condition ($V_x^* = 21m/s$ and $V_y^* = 0m/s$), and with

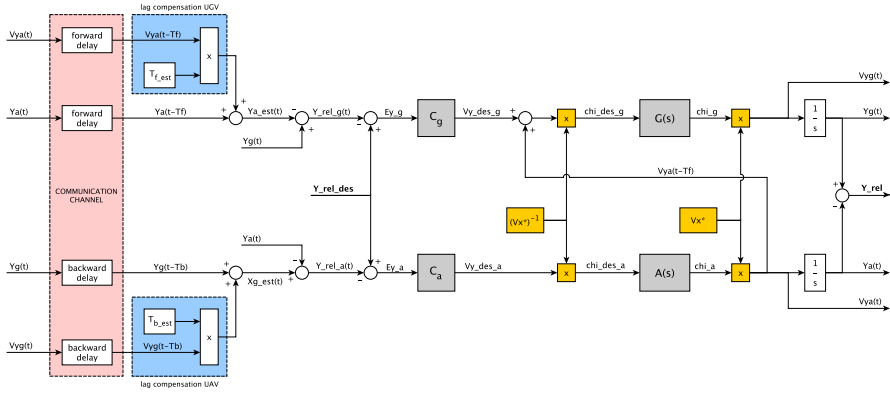


Figure 4.33 Lateral position synchronization controller with lag correction terms.

the small angle approximation, the angular transformation from Equation (4.71) reduces to a linear expression of the form:

$$\chi = \frac{V_y}{V_x^*} \quad (4.126)$$

The command transformation terms (orange blocks in Figure 4.33) therefore cancel out during the block diagram reduction. The derivation of the desired transfer function is performed in the same way as shown for the longitudinal case, while intermediate steps are omitted here. From Figure 4.33 the following relations can be derived:

$$Y_a = \frac{AC_a}{s - AC_a} Y_{rel}^{des} - \frac{AC_a e^{-sT_b} (1 + sT_b)}{s - AC_a} Y_g = H_{ay} Y_{rel}^{des} + H_{ag} Y_a \quad (4.127)$$

$$Y_g = \frac{GC_g}{s + GC_g} Y_{rel}^{des} + \frac{Ge^{-sT_f} (s + C_g (1 + sT_f))}{s + GC_g} Y_a = H_{gy} Y_{rel}^{des} + H_{ga} Y_a \quad (4.128)$$

Substituting Equation (4.127) into Equation (4.128) and vice versa yields the transfer functions from Y_{rel}^{des} to Y_g and Y_a :

$$\frac{Y_g}{Y_{rel}^{des}} = \frac{H_{gy} + H_{ga} H_{ay}}{1 + H_{ga} H_{ag}} \quad (4.129)$$

$$\frac{Y_a}{Y_{rel}^{des}} = \frac{H_{ay} - H_{ag} H_{gy}}{1 + H_{ag} H_{ga}} \quad (4.130)$$

Subtracting Equation (4.130) from Equation (4.129) finally leads to the transfer function from Y_{rel}^{des} to Y_{rel} , which allows for analyzing the stability of the lateral coupled control system:

$$\frac{Y_g}{Y_{rel}^{des}} - \frac{Y_a}{Y_{rel}^{des}} = \frac{Y_{rel}}{Y_{rel}^{des}} = \frac{H_{ay}(H_{ga} - 1) + H_{gy}(H_{ag} + 1)}{1 + H_{ag} H_{ga}} \quad (4.131)$$

As for the longitudinal analysis, the transfer functions for the lateral controlled vehicle models $G(s)$ and $A(s)$ have been derived using the nonlinear vehicle models introduced in Chapter 3, and were approximated as second order systems, as shown in Equations 4.132 and 4.133:

$$G(s) = \frac{13.7}{s^2 + 16.31s + 13.7} \quad (4.132)$$

$$A(s) = \frac{0.09593}{s^2 + 0.3807s + 0.0944} \quad (4.133)$$

The lateral position synchronization controllers C_g and C_a are again implemented as PID-controllers with the gains shown in Table 4.2. Analog to the longitudinal case, the exponentials in the transfer functions were expressed as rational functions using a Padé approximation of order 5. The pole zero map and step response plot for the coupled lateral system with increasing time delays are shown in Figures 4.34 and 4.35 respectively.

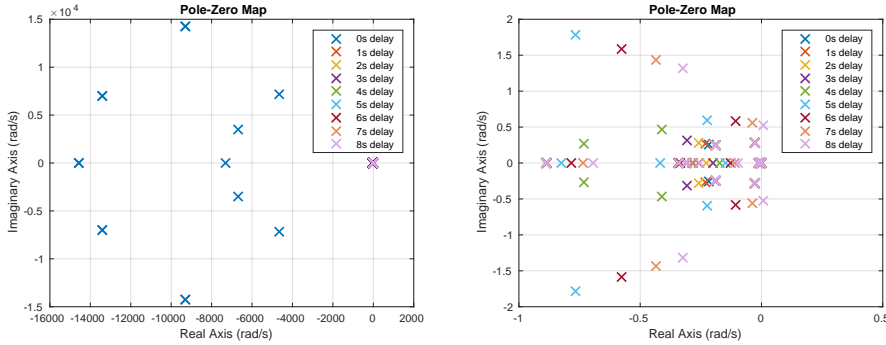


Figure 4.34 Pole-zero plot for transfer function Y_{rel}/Y_{rel}^{des} of lateral performance-based position synchronization controller with increasing communication time delays. The system remains stable for time delays of up to 7.7s.

The lateral dynamics present a similar behavior as the longitudinal dynamics, where the maximum tolerable time delay was estimated to be 7.7s. The undelayed case is stable, as can be seen in Figure 4.35. The direct velocity feedback from the aircraft to the ground vehicle was found to have a positive effect on the tolerable time delay (increase of around 1s), acting as a damper on the relative motion of the vehicles.

4.4.4 Sensitivity Analysis

In the following section the sensitivity of the landing system to external disturbances will be analyzed using the linear models from Chapter 3.5. The analysis will be divided into two parts — first a frequency domain analysis will be performed, investigating the frequency response in terms of the loop transfer function, the sensitivity function and the complementary sensitivity function. In a second step a stochastic time domain analysis

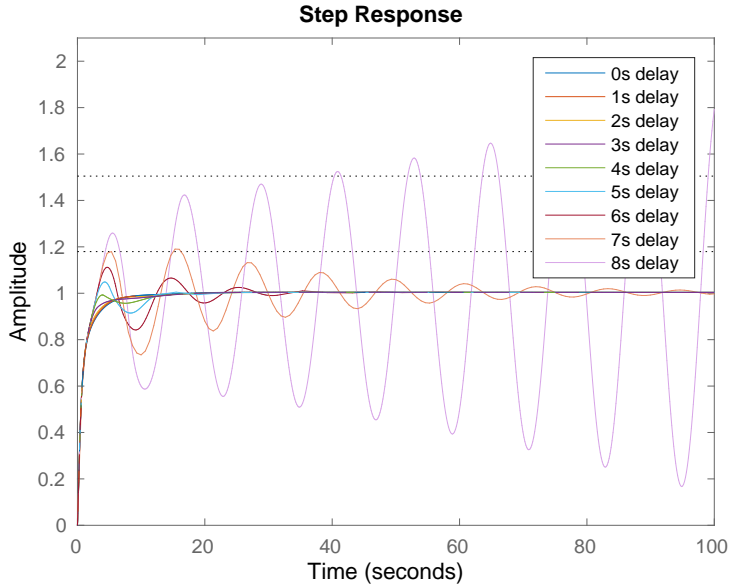


Figure 4.35 Step responses for transfer function Y_{rel}/Y_{rel}^{des} of lateral performance-based position synchronization controller with increasing communication time delays. The system remains stable for time delays of up to 7.7s.

will be performed by means of Monte Carlo simulations. It should be noted that at this point only linear vehicle models are used, since only smaller deviations from the trim conditions are investigated. For completeness, a nonlinear robustness analysis with the identified (high fidelity) vehicle models will be shown in Chapter 5.3.

Frequency Domain Analysis

In the frequency domain, the sensitivity of a given system is typically analyzed in terms of the sensitivity and complementary sensitivity functions, as will be explained in the following. A basic feedback loop is shown in Figure 4.36, where C is the controller, P the plant, r the reference signal, e the error signal, u the control signal, d the load disturbance, y the system output, and n the noise.

The transfer functions from the system inputs to the error signal can be derived as shown in Equations (4.134) to (4.138). The system output y can be expressed as:

$$y = (1 + PC)^{-1}PCr + (1 + PC)^{-1}d - (1 + PC)^{-1}PCn \quad (4.134)$$

Expressing Equation (4.134) in terms of the error signal leads to:

$$e = r - y = (1 + PC)^{-1}r - (1 + PC)^{-1}d + (1 + PC)^{-1}PCn \quad (4.135)$$

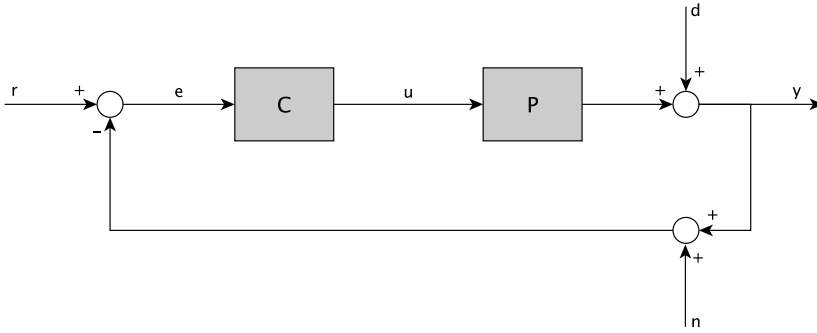


Figure 4.36 Basic feedback control loop.

and with the definitions of the sensitivity function S and complementary sensitivity function T :

$$\frac{e}{r} = \frac{1}{(1 + PC)} = S \quad (4.136)$$

$$\frac{y}{r} = \frac{PC}{(1 + PC)} = T \quad (4.137)$$

the transfer function from the different system inputs (reference r , disturbance d and noise n) to the error is found as:

$$e = r - y = Sr - Sd + Tn \quad (4.138)$$

For the landing application, the reference signal r (i.e. X_{rel}^{des}) will typically be zero and changing with frequency zero, since the landing is supposed to happen with zero relative position, and the disturbances d will mostly be due to atmospheric turbulence or runway slope changes, and thus only changing slowly (low frequency changes). These inputs would therefore mainly act in the lower frequency region (typically below the gain crossover frequency ω_{gc}). The noise input is a high-frequency signal originating from sensor noise (typically above the gain crossover frequency ω_{gc}). As can be seen in Equation (4.138), it would therefore be desirable for S to be small for low frequencies to provide good reference tracking and disturbance rejection, while T should be low at high frequencies for noise attenuation. At the same time Equation (4.137) requires T to be high at low frequencies to allow for effective reference tracking. In addition, from Equations 4.136 and 4.137 the following relation can be derived:

$$S + T = 1 \quad (4.139)$$

which has important implications on the design of the closed loop behavior, since S and T cannot be designed independently. If fast reference tracking is desired, T should be high, but this would also reduce the noise attenuation qualities. A tradeoff has to be found, which is usually not a problem due to the large bandwidth separation between the different input signals.

Important parameters of the frequency response regarding the system's robustness are the gain crossover frequency ω_{gc} , the curve slope at crossover, the peak value of the sensitivity function M_s and the high-frequency roll-off (curve slope at high frequencies). The sensitivity peak is calculated according to Equation (4.140).

$$M_s = \max_{0 \leq \omega < \infty} |S(j\omega)| = \max_{0 \leq \omega < \infty} \left| \frac{1}{1 + G(j\omega)C(j\omega)} \right| \quad (4.140)$$

The inverse of the maximum value of the sensitivity function $1/M_s$ represents the minimum distance from the Nyquist curve of the loop transfer function to the critical $-1 + 0j$ point, and is thus a direct measure for system robustness.

In the following the frequency responses of the safety-based and performance-based position synchronization controllers will be analyzed after the application of the optimization-based tuning procedure described in Section 4.4.2. These results will then be compared against the frequency responses before optimization as shown in Figure 4.29 and conclusions on the system performance and robustness will be drawn.

First, the stability margins of the systems before the optimization-based tuning (after frequency-based loop shaping design) and after optimization were calculated to verify the effectiveness of the tuning procedure. As previously mentioned, it is expected that the margins will decrease, since the optimization-based tuning should drive the systems towards higher performance (higher bandwidth) and thus lower stability margins. Table 4.3 shows the gain (GM) and phase margins (PM) for the longitudinal position synchronization controllers before and after the optimization-based tuning procedure. As can be seen, the optimization-based procedure did drive the systems towards lower stability margins and thus higher bandwidth and performance as expected.

Table 4.3 Stability margins for the longitudinal position synchronization controllers before and after the optimization-based tuning procedure.

	Margins	Before optimization	After optimization
safety-based	GM	16.2	14.6
	PM	72.4	68.8
performance-based	GM	19.1	17
	PM	77.3	73.9

Figure 4.37 shows the loop shapes of the safety-based and performance-based synchronization controllers after the optimization-based tuning procedure was applied.

To judge the performance of the system, the gain crossover frequency ω_{gc} and the bandwidth ω_b are conclusive measures, characterizing the system in terms of response speed. The sensitivity peak M_s is a direct robustness indicator, where values of up to 6dB are acceptable [109], while lower values are preferable. The maximum value of the

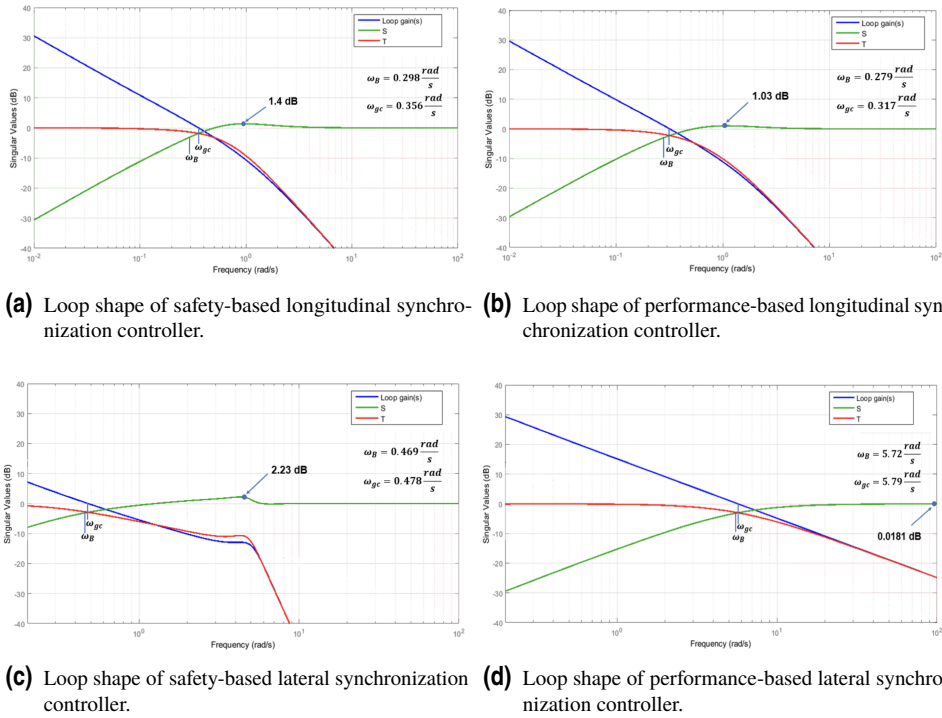


Figure 4.37 Frequency responses of safety-based and performance-based position synchronization controllers after optimization-based tuning.

sensitivity function is directly related to the system's disturbance rejection qualities, where a higher peak value leads to worse disturbance rejection properties.

Figures 4.37a and 4.37b show the loop shape of the safety-based and performance-based longitudinal controllers respectively. As can be seen, the response speed is similar, while the performance-based controller exhibits a lower sensitivity peak (25% reduction).

Figures 4.37c and 4.37d depict the lateral frequency response. As can be seen, the performance-based controller exhibits much faster dynamics with an increase in bandwidth by more than a factor of 10, when compared to the safety-based controller. The performance-based controller also presents a significantly lower sensitivity peak (more than factor 100). For the performance-based controller these measures imply a faster system response and increased robustness in terms of low frequency disturbances, which in turn leads to a decrease in sensor noise attenuation. The situation is inverted for the safety-based controller, which exhibits worse disturbance rejection characteristics, but higher noise attenuation. The high noise attenuation of the safety-based controller can also be observed through the steep roll-off at high frequencies, as shown in Figure 4.37c, while the high-frequency roll-off is much shallower for the performance-based controller as visible in Figure 4.37d.

In the following section, the observations made in the frequency domain will now be compared against the results of the Monte Carlo analysis.

Time Domain Analysis

For the robustness analysis in the time domain the linear vehicle models from Chapter 3.5 have been used, which had been linearized at a trim velocity of 21m/s . The initial conditions for the Monte Carlo simulation were therefore set to a synchronized motion condition with aligned positions in x and y directions at a constant velocity of 21m/s and constant aircraft altitude. Then a discrete wind gust and atmospheric turbulence were added to the system, where the turbulence was modeled as normally distributed white noise ($\mu = 0\text{m/s}$, $\sigma^2 = 1\text{m/s}$) passed through a Dryden filter [77]. The wind gust has been applied in longitudinal (tailwind) and lateral (crosswind left) directions with a magnitude of 5m/s each. The effects of gust disturbances are analyzed first. In a second step normally distributed random time delay ($\mu = 0.15\text{s}$, $\sigma^2 = 0.0001\text{s}$) was introduced to the system to analyze the effects of time delay on the system performance. The mean value of $\mu = 0.15\text{s}$ for the random time delay has been derived from real time delay estimates as encountered during flight tests (see Figure 4.1), representing a worst case scenario. A number of 1000 simulation experiments have been performed for each controller architecture and disturbance effect (turbulence and time delay), which was considered to be sufficient.

Figures 4.38 to 4.41 illustrate the results of the Monte Carlo analysis, where the x-axis represents the magnitude of the sampled value (e.g. seconds in case of a measured settling time) plotted onto a discrete number of equally spaced value intervals or *bins*, while the y-axis represents the normalized frequency of occurrence. The number of bins is chosen as the square root of the total number of different observation values, while the normalization is performed by dividing the total number of occurrences (frequency of occurrence before normalization) for one specific observation by the total number of trials (here 1000) and the width of one bin (or value interval) for each different observation value. An empirical distribution was then estimated from the samples and is also plotted in the figures. It should be noted that since linear models have been used for this analysis, normally distributed random input variables will also result in normally distributed outputs. A nonlinear Monte Carlo simulation with the identified high-fidelity aircraft model will be presented in Chapter 5.3 to determine realistic performance bounds for the specific demonstrator setup used in the landing experiments in Chapter 6.3. The analysis presented here is meant to provide more general conclusion on the different performance properties of the two proposed synchronization controllers, where it is sufficient to use more generic models which are representative for a given vehicle class, rather than a specific vehicle type. The use of the a priori models from Chapter 3.5 are therefore deemed to be appropriate for this purpose.

Figure 4.38 illustrates the results in terms of maximum overshoot and undershoot for both architectures, after a randomly varying longitudinal tailwind gust with a mean value of 5m/s has hit the synchronized vehicles. As can be seen in Figure 4.38, the performance-based controller is more effective in terms of disturbance rejection in both overshoot and undershoot. In case of the overshoot a mean value of 0.347m is encountered for the performance-based controller, while the safety-based controller exhibits a mean value

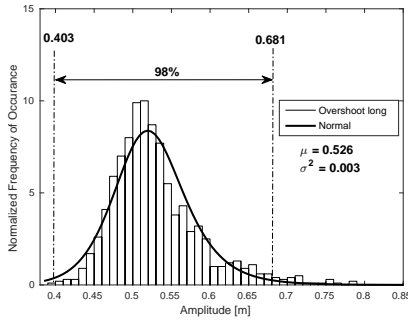
of $0.526m$. This corresponds to an improvement of 50%. The variance is also lower for the performance-based controller, resulting in a tightened confidence interval of $(0.298m, 0.396m)$, versus the broader confidence interval of $(0.403m, 0.681m)$ for the safety-based controller. The confidence interval can be interpreted as a performance bound, in the sense that the real system response is contained in the given bound with a confidence of 98%. In case of the undershoot, an improvement in terms of mean value of around 30% is found for the performance-based controller, while the variance has decreased by a factor of two. It should be noted that the maximum values for the undershoot are larger than those for the overshoot. This is due to the fact that a tailwind gust is applied and that relative position is calculated by subtracting the aircraft position from the ground vehicle position. As undershoot is associated to negative relative position values, and overshoot to positive values, the tailwind gust first causes a large undershoot (the aircraft is pushed forward), while the second oscillation corresponds to the overshoot.

Figure 4.39 illustrates the system response in terms of settling time for longitudinal and lateral gust disturbances, where Figures 4.39a and 4.39b correspond to the longitudinal disturbance, while Figures 4.39c and 4.39d illustrate the lateral response. In the longitudinal direction only a small improvement in favor of the performance-based controller is visible with a mean settling time of $15.21s$, against the slightly higher value of $15.96s$ in case of the safety-based controller. This corresponds to an improvement of around 5%. The performance bounds have basically not changed, where a range of $(12.82s, 18.02s)$ is found for the performance-based controller versus the bounds of $(13.38s, 18.58s)$ in case of the safety-based controller.

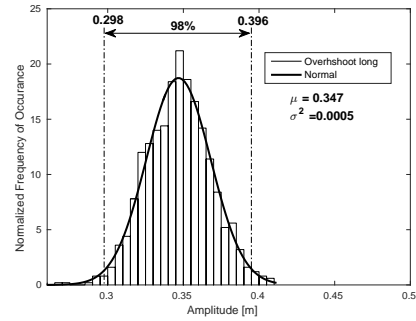
In the lateral direction a shorter settling time is found, where the performance-based controller again exhibits slightly better disturbance rejection properties, with a mean settling time of $4.87s$ versus $5.71s$ for the safety-based controller. For the lateral direction this represents an improvement of around 17%. The variance is around 30% lower for the performance-based controller.

Figure 4.40 illustrates the responses in terms of integral time absolute error (ITAE) for a crosswind gust. This performance metric reduces the influence of the early transient phase after the gust, and emphasizes the long term response. As can be seen, the performance-based controller presents a lower mean value of 21.87 , when compared to the value of 29.59 of the safety-based architecture. This represents an improvement of around 25% in favor of the performance-based approach. The difference in variance and thus performance bounds is even more pronounced, where the performance-based controller exhibits performance bounds more than 30 times smaller than the safety-based controller, leading to a more consistent performance.

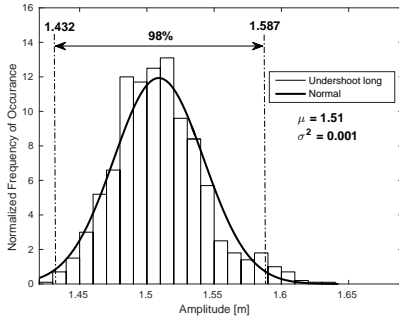
Finally Figure 4.41 illustrates the effect of time delay on the maximum undershoot performance metric, which has exhibited the strongest differences of all considered metrics. As previously mentioned, time delay was modeled as a normally distributed random variable with a mean value of $\mu = 0.15s$ and variance of $\sigma^2 = 0.0001s$. The introduction of time delay has led to a worsened system performance for both controller architectures as expected, where an increase of around 20% was found in terms of mean value, and around 60% in terms of performance bounds. Both controllers were almost equally affected by the time delay.



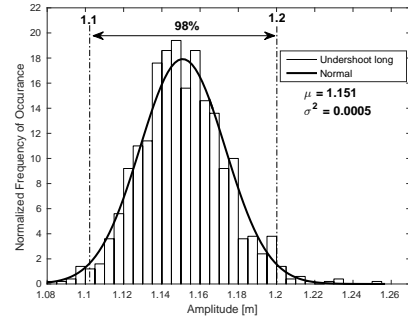
(a) Overshoot for safety-based controller after gust in longitudinal direction.



(b) Overshoot for performance-based controller after gust in longitudinal direction.



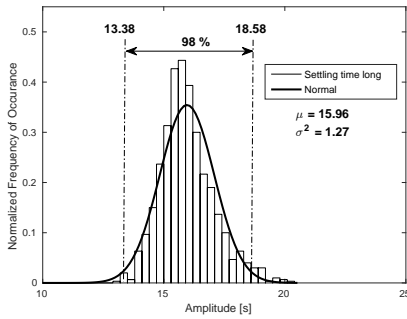
(c) Undershoot for safety-based controller after gust in longitudinal direction.



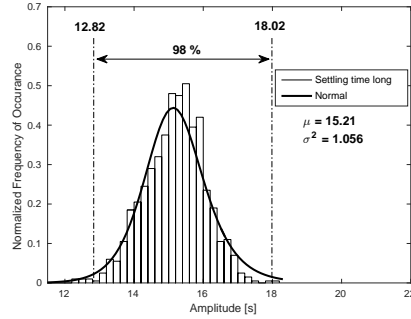
(d) Undershoot for performance-based controller after gust in longitudinal direction.

Figure 4.38 Maximum overshoot and undershoot for safety-based and performance-based position synchronization controllers after a 5 m/s tailwind gust in longitudinal direction. The interval of confidence is 98%, with mean value μ and variance σ^2 .

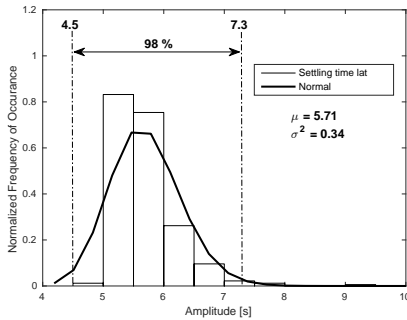
These results are in line with the observations from the frequency domain analysis, where the performance-based controller exhibited a higher bandwidth, which makes it faster but also more sensitive to high frequency noise. The time domain analysis also reveals higher robustness against low frequency disturbances, which is directly related to the lower sensitivity peak found in the frequency domain analysis. Overall the performance-based control architecture was found to have better performance (as expected), where the improvement seems more pronounced in the lateral direction than in the longitudinal direction. Interestingly, settling time after a longitudinal gust is not much affected by the different controller architectures. In summary, the performance-based controller presented an improvement of around 50% in terms of overshoot, 30% for undershoot, 5% for longitudinal settling time, 17% for lateral settling time, and 25% in terms of the ITAE performance metric. Both architectures exhibited a similar performance decrease due to



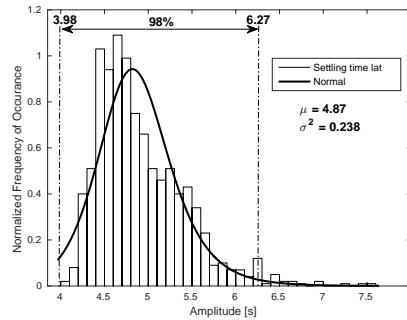
(a) Settling time for safety-based controller after gust in longitudinal direction.



(b) Settling time for performance-based controller after gust in longitudinal direction.



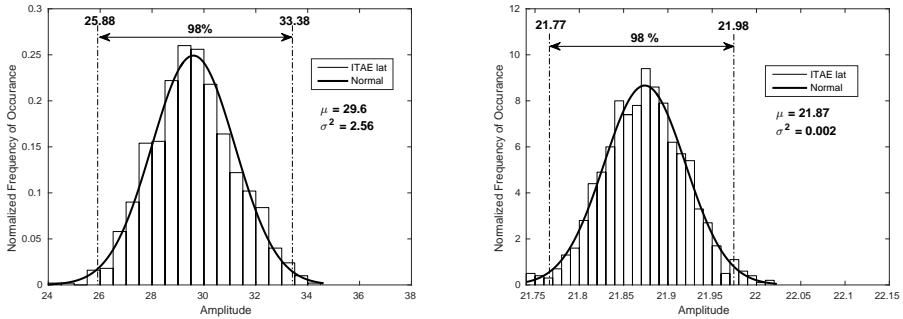
(c) Settling time for safety-based controller after gust in lateral direction.



(d) Settling time for performance-based controller after gust in lateral direction.

Figure 4.39 Settling time performance metrics for safety-based and performance-based position synchronization controllers after gusts in longitudinal and lateral directions. The interval of confidence is 98%, with mean value μ and variance σ^2 .

time delay. If a mean improvement is calculated by weighting each performance metric equally, an overall improvement of around 25% is found in favor of the performance-based controller. It is nevertheless important to consider the absolute values, where e.g. the 50% improvement in maximum overshoot implies an actual improvement of less than 20cm. Of course there will most probably be multiple gust events during a landing maneuver, such that these effects add up to produce a bigger difference between the applied architectures. However, when choosing a control architecture for the landing system, the higher implementation effort of the performance-based controller should be considered. As described in Chapter 2.4.5, system stability becomes time delay dependent for the performance-based controller. Consequently additional precautions had to be taken to ensure safe operation, while the safety-based controller is more straightforward to implement, better suited for certification, and stability is not dependent on communication



(a) ITAE for safety-based controller after lateral gust. (b) ITAE for performance-based controller after lateral gust.

Figure 4.40 ITAE performance metrics for lateral wind gusts of $5m/s$. The interval of confidence is 98%, with mean value μ and variance σ^2 .

time delay. The choice for one or the other position synchronization controller architecture must therefore be taken under consideration of safety, implementation and certification effort, available runway length, and available sensor quality. If the available sensors are not too noisy, the available runway length is plenty and the implementation effort is to be reduced, the safety-based approach will in most cases provide sufficient performance. For the demonstrator setup, where a human driver is executing the calculated control commands, the safety-based approach additionally has the benefit of a reduced driver workload, where only the longitudinal control commands would be varied. For these reasons the safety-based control approach was chosen for the landing experiments, which will be shown in Chapter 6. Nevertheless the performance-based approach might be required for more restrictive environments with a shorter available runway or stronger atmospheric disturbances and for vehicle types with lower safety and certification requirements. A good example of such scenario is the landing of heavier, medium sized UAV on a mobile runway system as introduced earlier in Chapter 1.3.3). As previously mentioned, the landing system stability issue had to be solved for a safe application of the performance-based landing system, where high time delays might cause the system to become unstable as shown in Figure 4.32. The required fail-safe functionality has been developed in Section 4.3.3 by considering the passivity of the landing system.

Chapter Conclusions

In this chapter two possible solutions to the cooperative landing problem were developed. The first system, the *safety-based* landing control system, was developed

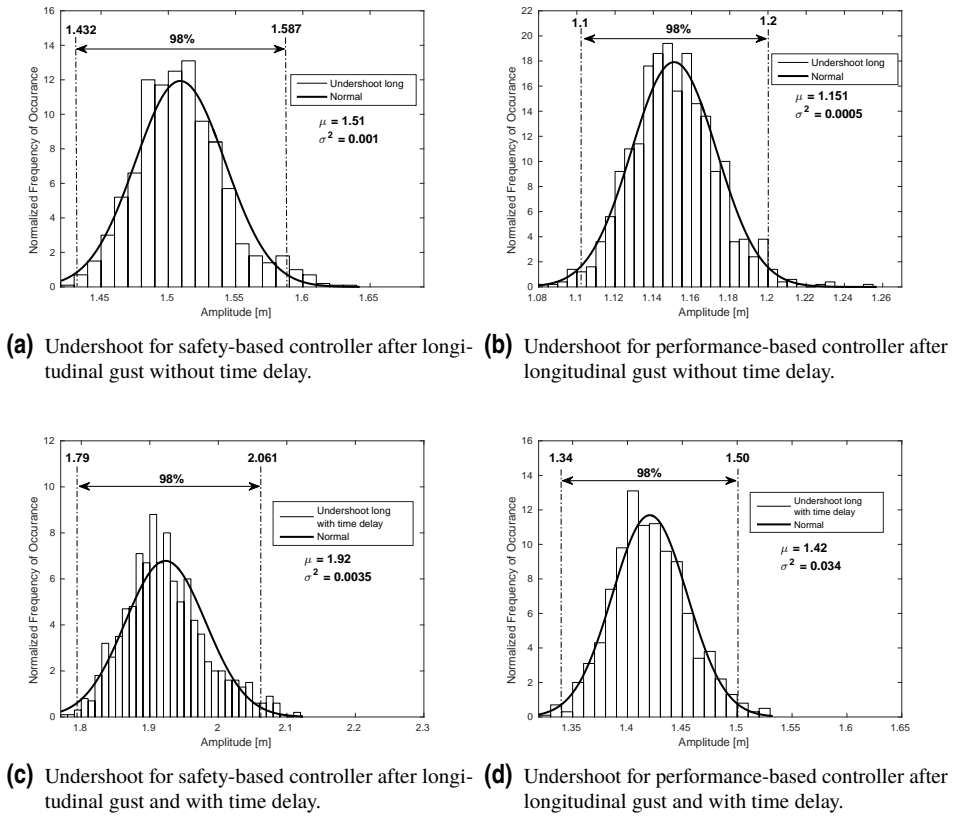


Figure 4.41 Undershoot responses for longitudinal wind gusts without time delay and in the presence of random time delay with a mean value of 0.15s. The interval of confidence is 98%, with mean value μ and variance σ^2 .

with the focus on maneuver safety, as the name indicates. Here the adherence to existing landing procedures and the reduction of system complexity were of particular importance, as well as the practicality of the solution in terms of implementation effort and certifiability. The second system, the *performance-based* landing control system, was developed with the focus on optimizing the landing maneuver in terms of runway usage and robustness to external disturbances. A first performance comparison between the two systems in terms of disturbance rejection (in the horizontal plane) indicates an overall improvement of around 25% in favor of the performance-based landing controller. A full landing performance analysis (including the approach and final descent phases) will be performed in Chapter 5.

5 Simulation Experiments

Chapter Overview

Before any real landing experiments are attempted, the developed methods need to be tested in simulation and possible risks and operational limits identified. In this Chapter a realistic simulation environment is implemented. It includes all relevant effects, such as wind, turbulence, aerodynamic disturbances by the ground vehicle, communication delays, and a human driver interface for the ground vehicle (for real-time simulation experiments). Extensive simulations of the landing maneuver are performed, and robust performance bounds are estimated. The performed simulation analyses show that the cooperative control concept is valid and the landing maneuver exhibits a robust performance despite disturbances and system failures. The following sections will describe the developed simulation environment, the performed simulation experiments and results.

5.1 Simulation Environment

The simulation environment is developed as a design and risk assessment tool and therefore needs to contain all relevant effects. All incorporated effects will be described in the following sections.

Ground Vehicle Induced Disturbances

The ground vehicle induced aerodynamic disturbances have been modeled and analyzed in detail in Chapter 3.3. As a result the velocity vectors in 3D-space around the ground vehicle are known. This means that the magnitude (airspeed) and direction (angle of attack and sideslip angle) of the airflow velocity vector at a given point in space can be retrieved

from a three dimensional vector field. The relevant vector field has been reduced to a cubic volume of 12m edge length and a spacing of 0.2m considering the size and local extent of the encountered effects around the ground vehicle and for computation time reasons. This leads to a data set of 226981 points, each containing a $[x,y,z]$ velocity vector. The relevant volume and data points are shown in Figure 5.1.

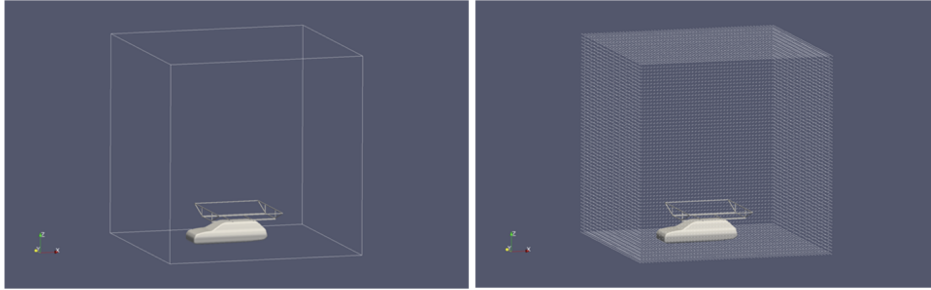


Figure 5.1 Representation of ground vehicle induced aerodynamic disturbance in real-time simulation environment.

For the integration into the simulation environment, the vector field was split up into three cubic scalar matrices (each containing one of the three vector components) suitable for a use in 3D-lookup tables. Intermediate values are determined using linear interpolation methods. The ground vehicle disturbance effect is then scaled to the ground vehicle airflow velocity and forms an additive contribution to any present wind effects. Thereby a couple of minor approximations have been applied — the ground vehicle effect is considered to be stationary at a given airflow velocity and independent of the aircraft position.

Wind Effects

Atmospheric disturbances are modeled in the simulation environment. The following effects are captured:

- Atmospheric conditions (calculation of static pressure, temperature, density and speed of sound at current MSL altitude)
- Background wind (a three dimensional variable wind vector can be set)
- Turbulence (using Karman turbulence models)
- Wind shear (calculation of body angular rates due to wind and turbulence)

Any of these effects can be activated or deactivated depending on their relevance for a given simulation experiment.

Communication Delays

During the cooperative phase of the landing maneuver the control loops of the coupled aircraft and ground vehicle controllers are closed through a wireless network. Any wireless

data link is subject to signal delays. For a realistic representation of the real world conditions this communications delay must be considered in the simulation environment. The information exchanged over the wireless network consists of the positions and velocities of the two vehicles. The relative position and velocity are then calculated locally by the respective vehicle's on-board computers. Therefore the signal lag seen by each of the vehicles can differ slightly.

Human Actuator

As mentioned in Sections 1.5.2 and 3.4, a graphical user interface was mounted inside the ground vehicle and is used for displaying the corrective action the *human actuator* has to perform. To get a better insight into the actual capabilities of a human actuator for this application, the human involvement has been considered during the simulation experiments. This means that the human has to adjust the given control commands during the simulation, just as he would during the real world experiment. All relevant simulation experiments have therefore been performed in real-time to guarantee realistic reaction times and stress levels for the driver.

The human actuator interface for real-time simulation has been implemented using a RC controller as input device. Instead of a gas pedal and steering wheel (as present in the real ground vehicle) one of the 2-axis controller joysticks was used.

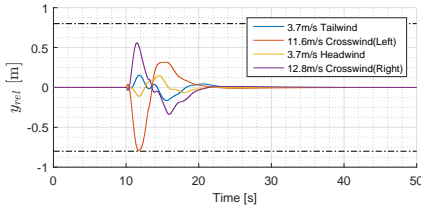
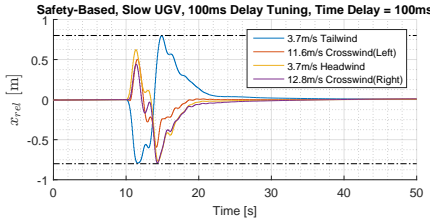
During the simulation a cross-hair is shown on the screen, just as it would be during the real landing experiments. The position of the cross-hair is calculated from the difference between the commanded throttle and steering wheel angle and the current values. Vertical deviations correspond to throttle (forward/backward) inputs and horizontal deviations to steering wheel inputs (left/right). The human has to bring the cross-hair to the middle point (0,0) in order to follow the commanded control inputs. Different visualization concepts have been investigated where the current solution was found to be the most intuitive, considering the fact that two parameters have to be transmitted to the human while he is simultaneously performing the task of driving the car. A conclusion from early simulation experiments was that the driver should be assisted by a co-pilot for safety reasons. It is important to monitor the real-time capability of the system during simulation to guarantee for a realistic mode of operation with realistic reaction times and stress levels for the driver. Therefore a real-time display is used, showing any delays between simulation time and clock time.

5.2 Disturbance Rejection Performance Analysis

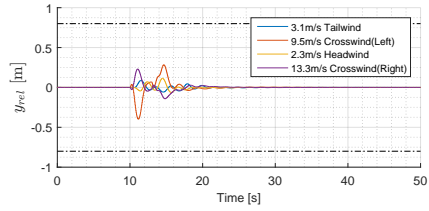
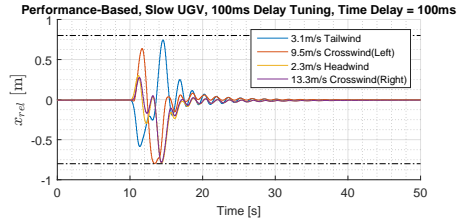
In this section the disturbance rejection characteristics of the safety-based and performance-based landing systems are analyzed using the simulation environment described in the previous section. Therefore, a set of deterministic simulation experiments are performed, where the already aligned vehicles are exposed to a horizontal wind gust disturbance. The principal gust directions are investigated, i.e. headwind, crosswind, and tailwind gusts. The gust intensity is increased until the relative x or y positions hit the retry limit of $0.8m$, as defined in Figures 4.12 and 4.13. The gust duration is set to $3s$ for all simulation cases. This represents the worst case scenario, where the aircraft is already performing the flare maneuver at lower relative altitude (where the precision requirements imposed by the landing state machine are highest) when the gust event occurs. The intent is to find the maximum magnitude of the external disturbance, which can still be tolerated by the landing system without entering retry mode, and thus causing an interruption of the landing maneuver.

Two use cases are analyzed — first slow ground vehicle actuator dynamics are assumed to represent the demonstrator setup with the human driver. The ground vehicle actuator is therefore modeled as first-order low-pass filter with a cut-off frequency of $2.5rad/s$. This value was identified from road tests with the human-controlled ground vehicle. The second use case represents an autonomous ground vehicle with a fast actuation system, with a cut-off frequency of $15.7rad/s$. All aircraft control surfaces (aileron, elevator, rudder) are modeled as first-order low-pass filters with a cut-off frequency of $15.7rad/s$ in correspondence to [127], where very similar servo motors were used and their parameters identified. Both landing controllers (safety-based and performance-based) were tuned for a nominal communication delay of $100ms$, which represents the expected mean value measured during outdoor experiments (see Figure 4.1).

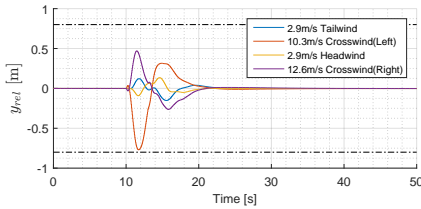
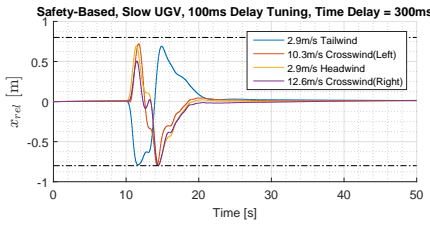
Figure 5.2 illustrates the controller performance comparison (safety vs. performance-based) for the slow ground vehicle actuator for the nominal $100ms$ delay case, as well as the $300ms$ worst case. As can be seen, the tolerable gust magnitudes are similar for both controllers for the nominal delay case, as well as the worst case. The safety-based controller slightly outperforms the performance-based controller, which can be explained by the additional delay inflicted by the slow ground vehicle actuator dynamics. This delay is injected into the bilateral control loops of the performance-based controllers on top of the communication delay, which significantly lowers the controller performance. This effect is not visible for the safety-based controller, since it operates in two parallel unilateral loops (one for each horizontal dimension). The destabilizing effect of time delay on the performance-based controller can clearly be seen in Figure 5.2d, where the system starts to develop an oscillatory behavior at higher time delays. It can also be observed that the longitudinal error represents the limiting factor for both architectures, except the crosswind case for the safety-based controller.



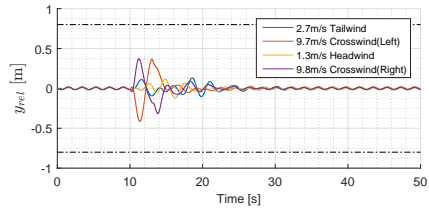
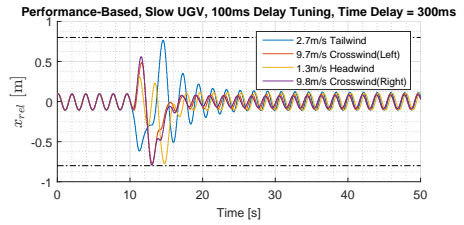
(a) Safety-based controller, slow ground vehicle actuator, 100ms delay.



(b) Performance-based controller, slow ground vehicle actuator, 100ms delay.



(c) Safety-based controller, slow ground vehicle actuator, 300ms delay.



(d) Performance-based controller, slow ground vehicle actuator, 300ms delay.

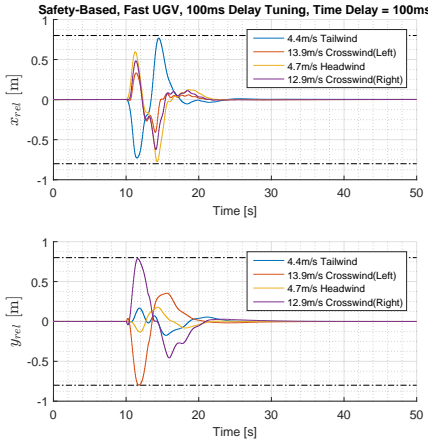
Figure 5.2 Horizontal relative vehicle positions during gust event with **slow ground vehicle actuator**. The gust disturbance magnitude is increased until the system reaches the retry limits of $0.8m$ in x and y direction. The legend shows the estimated disturbance rejection performance bounds. The *slow* ground vehicle actuator (human driver) is modeled as a first-order low-pass filter with a cut-off frequency of $2.5rad/s$.

Figure 5.3 shows the situation for a fast ground vehicle actuator, which represents an autonomous vehicle. Again, the tolerable wind gusts have similar magnitudes, but now the performance-based controller exhibits a slightly better performance. As can be seen in Figures 5.3a and 5.3c the safety-based controller presents similar lateral limits as encountered in the corresponding slow ground vehicle cases shown in Figures 5.2a and 5.2c. This is due to the fact that only the aircraft is responsible for the lateral motion control in the safety-based architecture, while of course the ground vehicle dynamics do not influence the aircraft behavior.

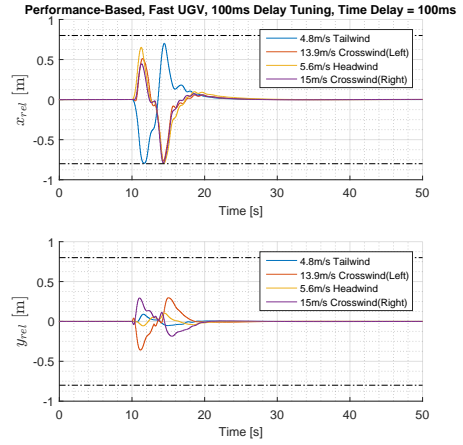
In general it can be observed that the performance-based controller offers slightly better longitudinal gust rejection properties in case of a fast ground vehicle actuator. For a slow ground vehicle actuator this advantage is not seen. Overall, the longitudinal direction does not present a significant difference between the two architectures. This can be explained by the slow overall ground vehicle dynamics (even with fast actuators) and its protracted transients. The lateral direction on the other hand benefits from the performance-based approach at low communication delays. In case of the slow ground vehicle actuator a slight improvement of $0.5m/s$ in terms of crosswind resistance is found, while the fast ground vehicle actuator exhibits an improvement of $2.1m/s$. For higher delays, the performance-based controller performance quickly degrades, and even becomes inferior to the safety-based design for slow ground vehicle actuation.

It can be concluded that the performance-based controller does not provide a significant advantage in terms of disturbance rejection in the case of slow ground vehicle actuation, and even performs worse in the longitudinal direction. For the fast (autonomous) ground vehicle actuation case an improvement of around 15% in terms of crosswind tolerance is found at lower communication delays. In this case the longitudinal disturbance rejection performance is found to be very similar for both architectures. For higher delays of around $300ms$ the safety-based controller catches up and offers the same performance.

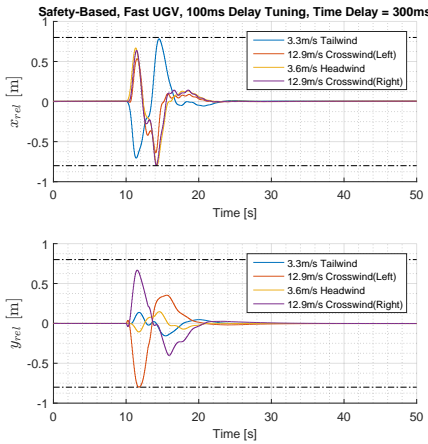
It should be noted that this analysis only considers single isolated gust events, while gusts typically occur many times during a landing maneuver. The previously identified slight advantage of the performance-based controller might therefore add up to a significant difference in overall landing performance. Such cumulative effects will be analyzed in Section 5.4, where the whole landing maneuver is considered. Additionally, the encountered gusts will vary in strength, direction and duration, and communication time delays will not be constant, which is not represented in the above analysis. In order to allow for the estimation of robust performance bounds a more detailed analysis must be performed. For this reason, a Monte Carlo simulation is performed in the next section, where randomly varied disturbances and time delays are applied to the system.



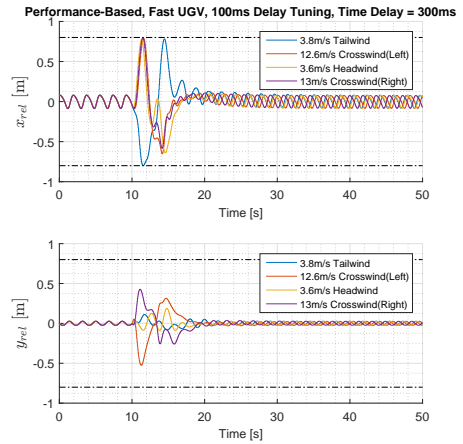
(a) Safety-based controller, fast ground vehicle actuator, 100ms delay.



(b) Performance-based controller, fast ground vehicle actuator, 100ms delay.



(c) Safety-based controller, fast ground vehicle actuator, 300ms delay.

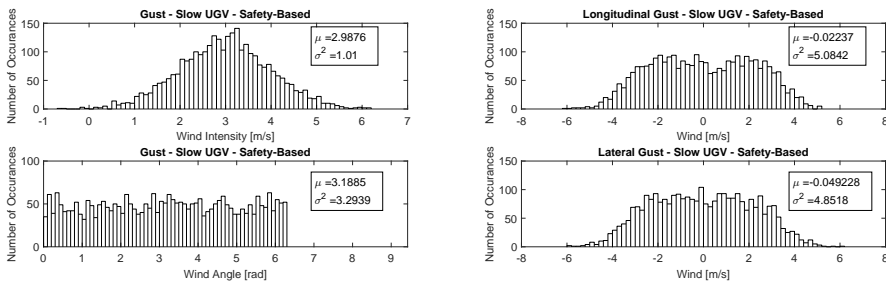


(d) Performance-based controller, fast ground vehicle actuator, 300ms delay.

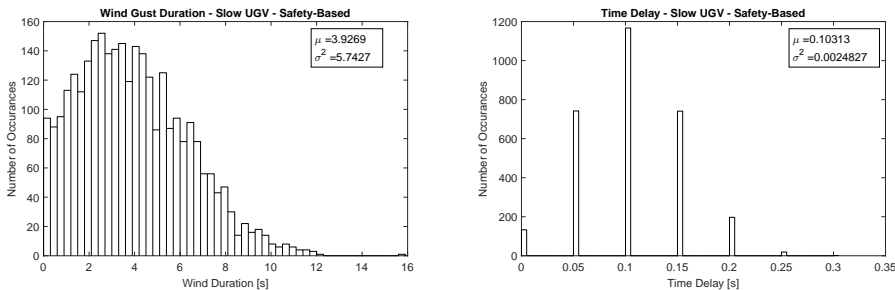
Figure 5.3 Horizontal relative vehicle positions during gust event with **fast ground vehicle actuator**. The gust disturbance magnitude is increased until the system reaches the retry limits of $0.8m$ in x and y direction. The legend shows the estimated disturbance rejection performance bounds. The *fast* ground vehicle actuator (autonomous ground vehicle) is modeled as a first-order low-pass filter with a cut-off frequency of $15.7rad/s$.

5.3 Disturbance Rejection Robustness Analysis

Throughout this section the results of a disturbance rejection robustness comparison between the safety-based and performance-based landing controllers are presented. The corresponding Monte Carlo simulations contain 3000 tests cases each, where given system inputs and parameters are randomly varied. The aligned vehicles are exposed to a random wind gust in the horizontal plane and randomly varying communication time delays. Thereby the wind gust intensity is varied, as well as the gust direction and duration. Figure 5.4 depicts these random variables for the safety-based controller simulation with slow ground vehicle actuation.



(a) Random wind gust intensity and direction in polar coordinates. (b) Random wind gust intensity and direction in cartesian coordinates.



(c) Random wind gust duration. (d) Randomly varying time delay. Discrete values were chosen in accordance to the real estimates.

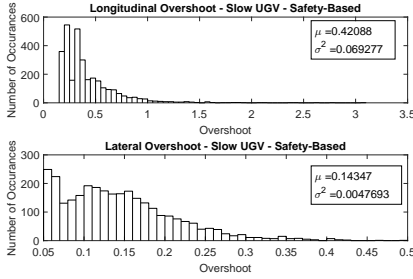
Figure 5.4 Random system inputs and parameters as used in Monte Carlo simulation of wind gust rejection robustness analysis. These distributions correspond to the safety-based simulation for the slow ground vehicle actuation case, but are virtually identical for all other test cases. Further illustrations are thus omitted for brevity.

As can be seen in Figure 5.4a, the wind gust intensity is randomly varied around a mean value of 3 m/s , which represents nominal operating conditions, while covering the worst case requirement of 5 m/s (see SR10 in Table 2.1). The wind gust direction is uniformly

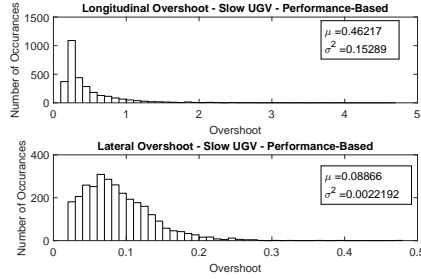
distributed and does not favor a specific direction as shown in Figure 5.4c, while the gust duration was varied around a mean value of $3s$. The time delay is also randomly varied around a mean value of $100ms$, which was previously estimated as the nominal delay from outdoor measurements. One should note the discrete representation of the different time delay samples, which was chosen in accordance to the estimates available on the real system, where the GPS receivers run on synchronized GPS-time at $20Hz$ and thus provide a new measurement every $50ms$. The same procedure was applied for the fast ground vehicle actuator and the performance-based cases, while the corresponding random variables are not plotted here, since they consist of virtually the same distributions as shown for the safety-based case.

Figure 5.5 shows the results for the slow ground vehicle actuation case in terms of different time domain performance indicators, including maximum overshoot, maximum undershoot, settling time, and integral time absolute error (ITAE). Based on these metrics a generalized and robust disturbance rejection performance comparison can be performed.

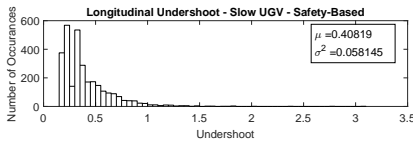
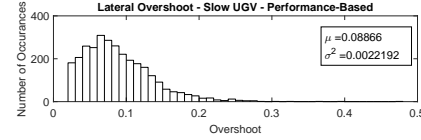
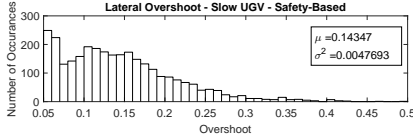
As can be seen in Figure 5.5, the performance-based controller exhibits a better performance in most of the metrics besides the maximum longitudinal overshoot, where the safety-based controller presents an advantage of around 9%. An improvement of around 40% is found for the maximum overshoot in lateral direction, which sounds like a lot but only represents an improvement of around $5cm$ in terms of absolute value. The maximum undershoot results in a similar situation as the overshoot case. The settling time is around 15% shorter in longitudinal and 85% in lateral direction for the performance-based controller. Again, it should be noted that this corresponds to an improvement of around $0.5s$ and $0.75s$ for the longitudinal and lateral directions, respectively. The ITAE metric exhibits an improvement in favor of the performance-based controller of around 12% in longitudinal and 50% in lateral direction. Overall, the performance-based controller provides a slightly better performance in all metrics but the maximum overshoot. In terms of absolute values the improvement is much less significant, and must be contextualized with respect to the much higher implementation and certification effort for the performance-based controller. Also, if the available runway is sufficiently long, the improvement might not be meaningful. Given the slow actuator dynamics of the demonstrator ground vehicle and the available runway of more than $1km$, these results point towards the implementation of the safety-based architecture for the demonstrator setup. This assessment shall be verified during the landing robustness analysis in Section 5.5.



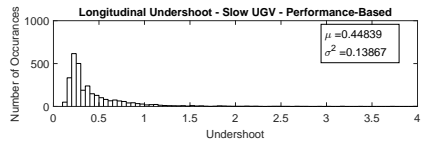
(a) Maximum overshoot (safety-based).



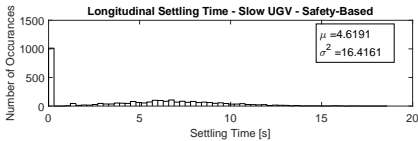
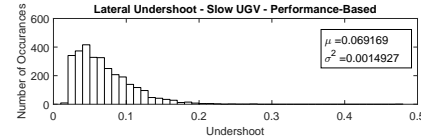
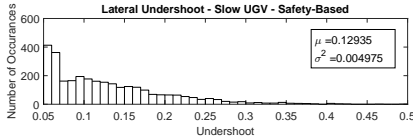
(b) Maximum overshoot (performance-based).



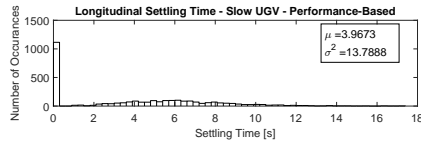
(c) Maximum undershoot (safety-based).



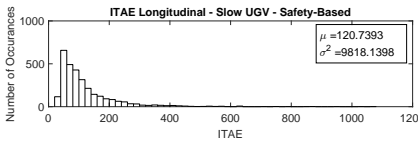
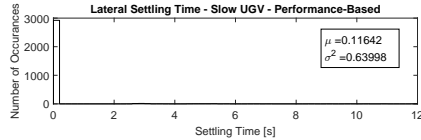
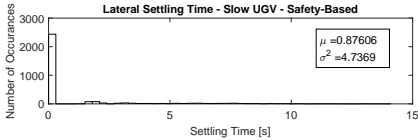
(d) Maximum undershoot (performance-based).



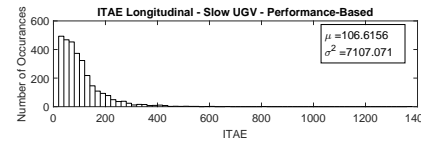
(e) Settling time (safety-based).



(f) Settling time (performance-based).



(g) ITAE (safety-based).



(h) ITAE (performance-based).

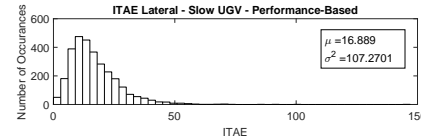
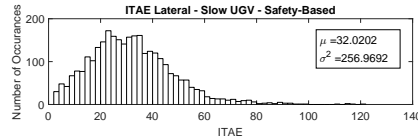
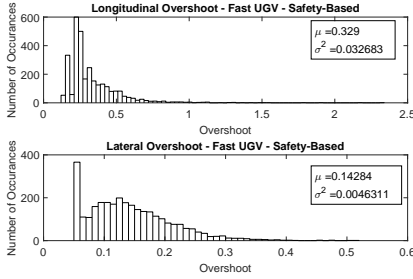
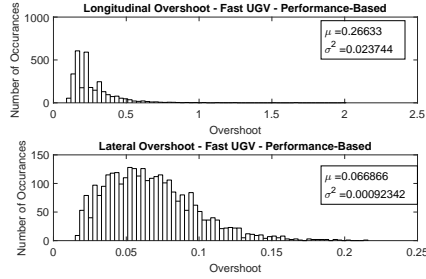


Figure 5.5 Results of Monte Carlo simulation for wind gust rejection robustness analysis for **slow ground vehicle actuator**. Different time domain performance metrics are shown for the safety-based and performance-based landing controllers. The analysis is performed in the horizontal plane.

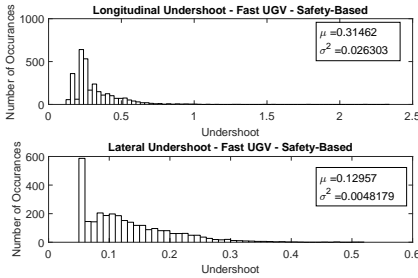
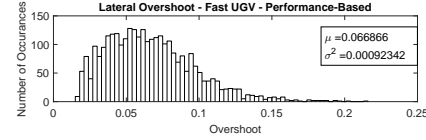
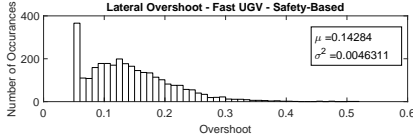
Figure 5.6 illustrates the results for the fast ground vehicle actuation case. In this case the advantage of the performance-based architecture is consistently visible, with an improvement of around 17% in terms of maximum longitudinal overshoot and 50% in terms of maximum lateral overshoot. Again, the undershoot case exhibits a similar relative performance as observed in the overshoot case. The settling time is improved by around 48% in favor of the performance-based controller in longitudinal direction, and around 95% for the lateral direction. Again, this improvement must be seen in the context of absolute values, where an improvement in settling time of around 0.75s for the longitudinal and 0.88s for the lateral direction is found. The ITAE metric presents an improvement of around 11% for the performance-based controller in the longitudinal direction, while an improvement of around 50% is found for the lateral direction. Overall, the performance-based architecture does seem to offer a considerable improvement in terms of disturbance rejection. As previously mentioned, the performance gain at each single gust event might add up to a decisive advantage in overall landing performance, which will be analyzed in the landing performance analysis throughout the following sections.



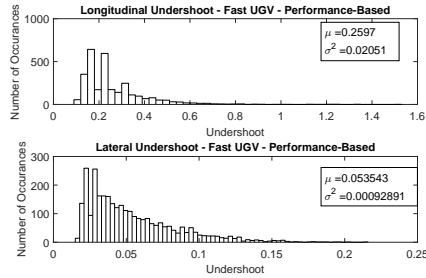
(a) Maximum overshoot (safety-based).



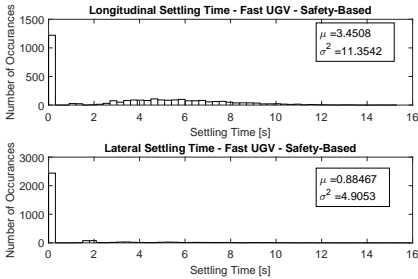
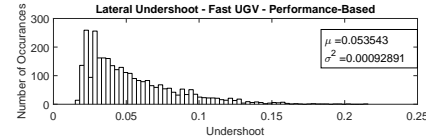
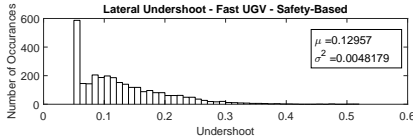
(b) Maximum overshoot (performance-based).



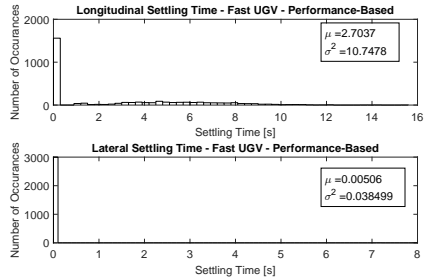
(c) Maximum undershoot (safety-based).



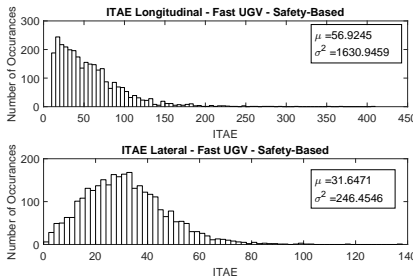
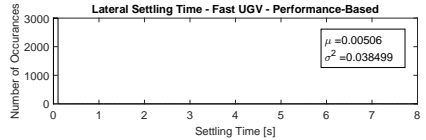
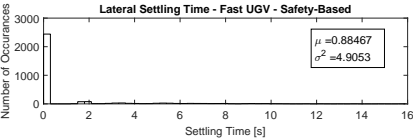
(d) Maximum undershoot (performance-based).



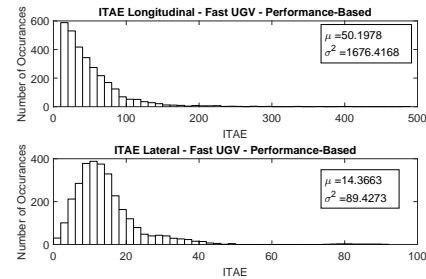
(e) Settling time (safety-based).



(f) Settling time (performance-based).



(g) ITAE (safety-based).



(h) ITAE (performance-based).

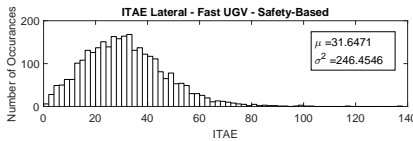


Figure 5.6 Results of Monte Carlo simulation for wind gust rejection robustness analysis for **fast ground vehicle actuator**. Different time domain performance metrics are shown for the safety-based and performance-based landing controllers. The analysis is performed in the horizontal plane.

5.4 Landing Performance Analysis

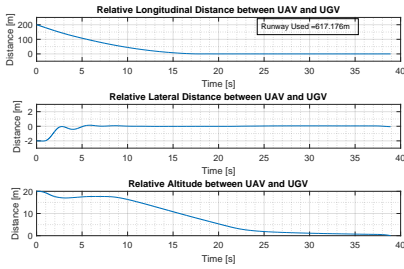
Throughout the last section the disturbance rejection characteristics for both the safety-based and performance-based landing controllers were analyzed in the horizontal plane (with the aircraft flying at constant altitude). This section presents the analysis of the full landing performance from maneuver initiation command to touchdown. First, a deterministic analysis under controlled conditions is performed to investigate the nominal and fail-safe behavior of the two landing systems. The analyzed failure cases include excessive wind gusts during the approach and flare phases of the landing, as well as an excessive communication delay. In the next section, a stochastic analysis will be carried out to determine robust performance bounds for the overall landing system and allow for final conclusions.

5.4.1 Nominal Landing Performance

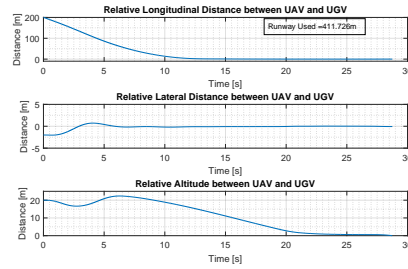
Figure 5.7 illustrates a nominal landing maneuver with the safety-based and performance-based landing systems for the demonstrator setup (slow ground vehicle actuator), where no wind is present. As can be seen in Figures 5.7a and 5.7b, the performance-based controller is capable of performing the landing in a shorter time of around 28s, while the safety-based controller requires around 38s. This represents an improvement of around 26% or 10s for the performance-based controller. Also the runway usage is lower in the case of the performance-based system, where a distance of around 412m is required to complete the landing. The safety-based landing system accomplishes the landing within 617m. In terms of runway usage this corresponds to a performance benefit of 33% or 205m for the performance-based landing system.

Figures 5.7c and 5.7d show the aircraft trajectory with respect to the ground vehicle, as well as the different geometrical areas represented in the landing state machine (see Chapter 4.2.6). It should be noted that the relative vehicle distance in all figures throughout this section is of opposed sign when compared to the previous definition of relative vehicle position, where the aircraft position was subtracted from the ground vehicle position (see Equation 4.2). This is done solely to provide a more intuitive representation where x-axis values increase towards the right, and has no functional implication. The initial descent at the beginning of the simulation is due to an aircraft trim setting which does not perfectly match the simulation initial conditions, which is negligible for the present analysis.

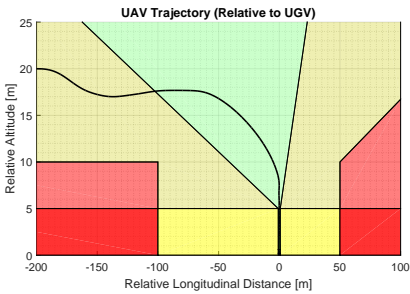
Figures 5.7e and 5.7f depict the flare phase of the landing maneuver. As can be seen, the landing is accomplished with no problems during nominal conditions. In the following section the fail-safe behavior of the two landing systems will be analyzed.



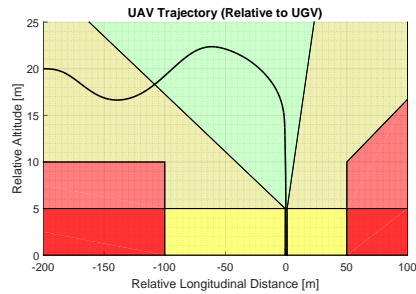
(a) Nominal landing performance (safety-based).



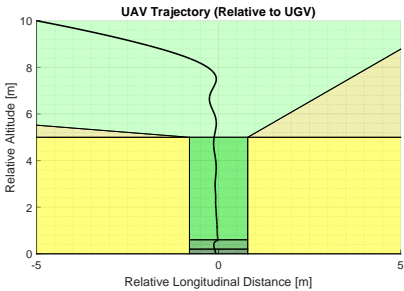
(b) Nominal landing performance (performance-based).



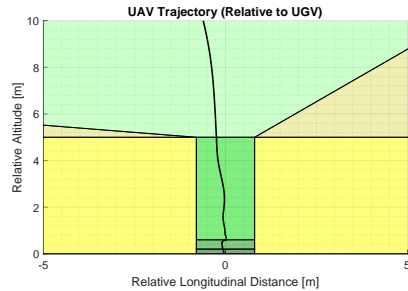
(c) Nominal landing trajectory (safety-based).



(d) Nominal landing trajectory (performance-based).



(e) Nominal flare trajectory (safety-based).



(f) Nominal flare trajectory (performance-based).

Figure 5.7 Nominal landing performance for safety-based and performance-based landing systems (side view).

5.4.2 Failsafe Behavior

Throughout this section the fail-safe behavior of the developed landing systems is analyzed, where three different failure cases are investigated. The first failure case contains an excessive tailwind gust, which hits the system during the approach phase. The second failure case contains an excessive crosswind gust during the flare phase, while the third failure case covers an excessive communication delay.

Excessive Tailwind Gust During Approach

In Figure 5.8 the system's fail-safe behavior is shown for the tailwind case. Thereby a tailwind gust of $5m/s$ (infinite duration) hits the system $5s$ after the landing is initiated (corresponding to a relative altitude of around $15m$). As can be seen in Figures 5.8a and 5.8b, the performance-based controller performs better and produces a successful landing after around $40s$, while using around $830m$ of runway. The safety-based controller as well is capable of performing a successful landing but requires around $52s$ and $1015m$ of runway length. This implies an improvement of around 23% or $12s$ in terms of landing time, and around 18% or $185m$ in terms of used runway.

Figures 5.8c and 5.8d illustrate the relative aircraft trajectory during the landing maneuver. As can be seen, the aircraft is pushed into the *hold altitude* zone and holds its altitude until the system is able to recover from the disturbance. Then the landing continues normally. **It should be noted that the hold altitude zone (ocher areas in Figures 5.8c to 5.8f) is deactivated in case of the performance-based controller, since the optimization-based approach controller contains the necessary spatial constraints in its formulation. The retry and go-around modes remain active.** The state machine zone is still depicted for visual orientation. The performance-based controller as well is pushed forward, but is not affected by the hold altitude zone and continues its descent during the horizontal position recovery. This implies a decisive advantage, since no descent time is sacrificed during the horizontal position recovery as was the case for the safety-based maneuver. It should be noted, that the safety-based controller requires the interaction with the landing state machine to achieve a successful landing after the gust event, while the performance-based controller resolves the disturbance without *external* interaction. This leads to a continuous control signal throughout the whole maneuver, which improves the landing performance.

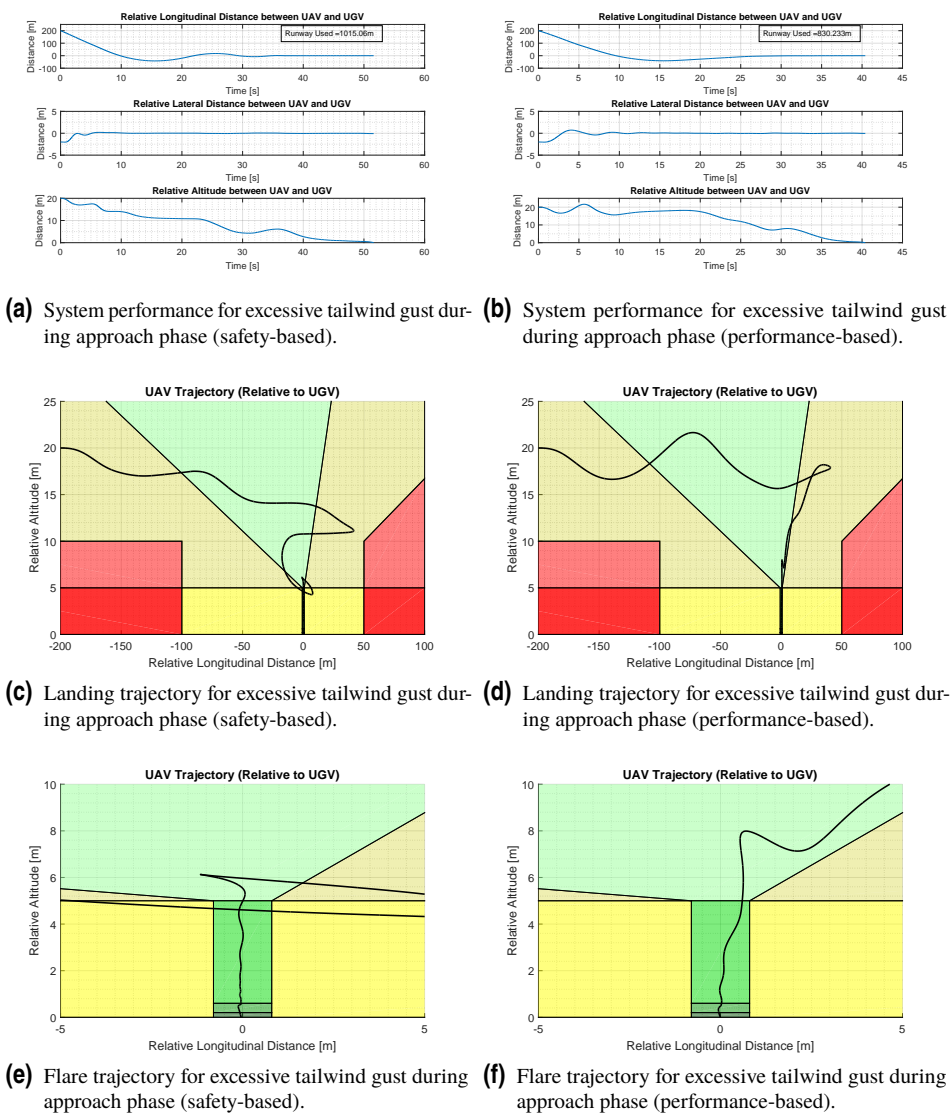


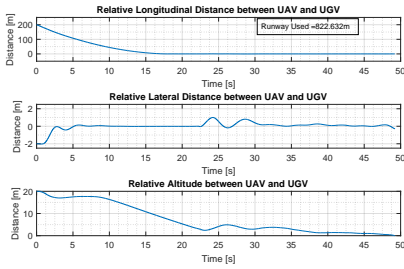
Figure 5.8 Fail-safe behavior of the safety-based and performance-based landing systems for an excessive tailwind gust during the approach phase. A tailwind of 5m/s hits the system 5s after maneuver start, which corresponds to a relative altitude of around 15m (side view).

Excessive Crosswind During Flare

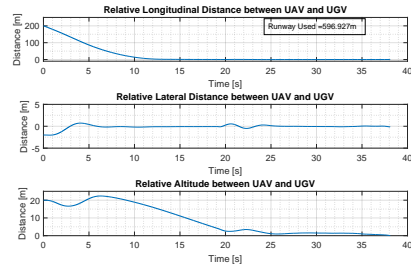
In Figures 5.9 and 5.10 the landing system's fail-safe behavior is shown for the crosswind case. Thereby a crosswind gust of 15m/s intensity and 10s duration hits the system at a relative altitude of 4m (during the flare phase). As can be seen in Figures 5.9a and 5.9b, the performance-based controller performs better and produces a successful landing after around 38s , while using around 597m of runway. The safety-based controller as well is capable of performing a successful landing but requires around 49s and 823m of runway length. This implies an improvement of around 23% or 11s in terms of landing time, and around 27% or 226m in terms of used runway.

Figures 5.9c and 5.9d illustrate the relative aircraft trajectories during the landing maneuver, which are basically the same as shown in Figures 5.7c and 5.7d for the nominal landing case. This is due to the crosswind event taking place after the approach phase, once the aircraft is already in flare mode. As can be seen in Figures 5.9e and 5.9f, the aircraft is not pushed into the retry zone in the longitudinal direction. Nevertheless, it can be anticipated from Figure 5.9e that a retry is triggered in the case of the safety-based controller, but not in the performance-based case. The reason is found in Figure 5.10a, where the retry zone violation is clearly visible for the safety-based controller, while the performance-based controller remains within the nominal landing zone, as shown in Figure 5.10b. The performance-based position synchronization controller thus exhibits better performance rejection properties, as expected.

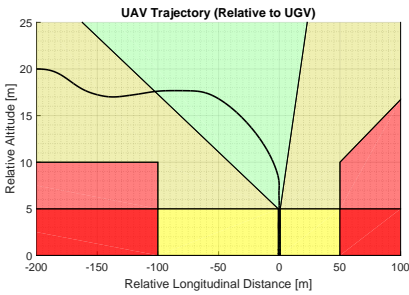
Overall, it can be said that both architectures are able to cope with the analyzed failure tests cases (tailwind and crosswind gusts), while the performance-based architecture exhibits an average landing performance improvement of around 25% for both failure tests cases. In the next section the communication delay failure case is analyzed.



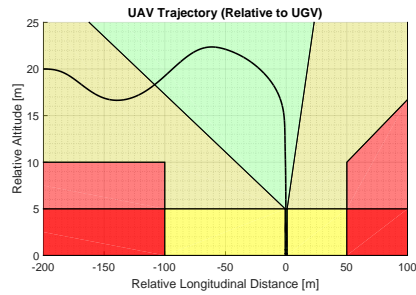
(a) System performance for excessive crosswind gust during flare phase (safety-based).



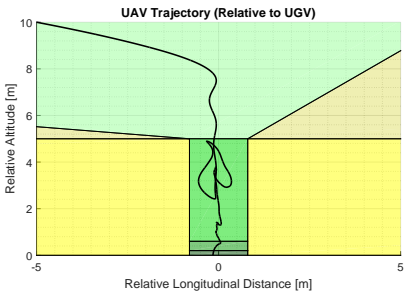
(b) System performance for excessive crosswind gust during flare phase (performance-based).



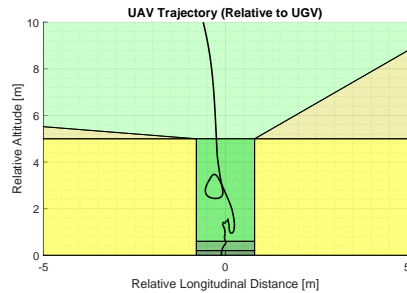
(c) Landing trajectory for excessive crosswind gust during flare phase (safety-based).



(d) Landing trajectory for excessive crosswind gust during flare phase (performance-based).



(e) Flare trajectory for excessive crosswind gust during flare phase (safety-based).

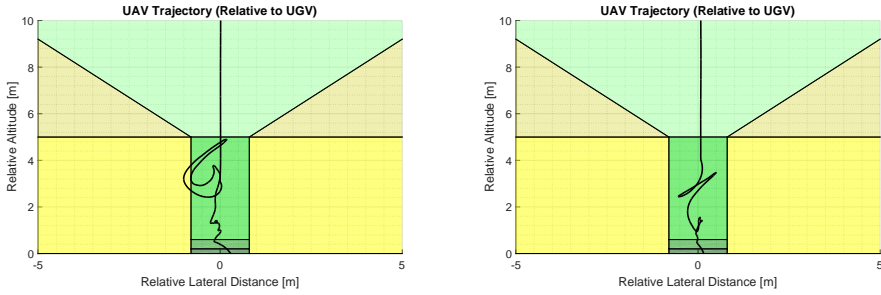


(f) Flare trajectory for excessive crosswind gust during flare phase (performance-based).

Figure 5.9 Fail-safe behavior of the safety-based and performance-based landing systems for an excessive crosswind gust during the flare phase. A crosswind (right) of 15m/s gust hits the system at a relative altitude of 4m (side view).

Excessive Communication Delay

Throughout the previous two sections the landing systems' fail-safe behavior due to excessive wind gust disturbances was analyzed and compared. Given the time delay independent

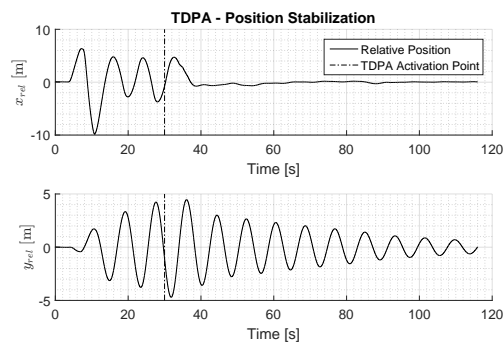


(a) Flare trajectory for excessive crosswind gust during flare phase (safety-based). (b) Flare trajectory for excessive crosswind gust during flare phase (performance-based).

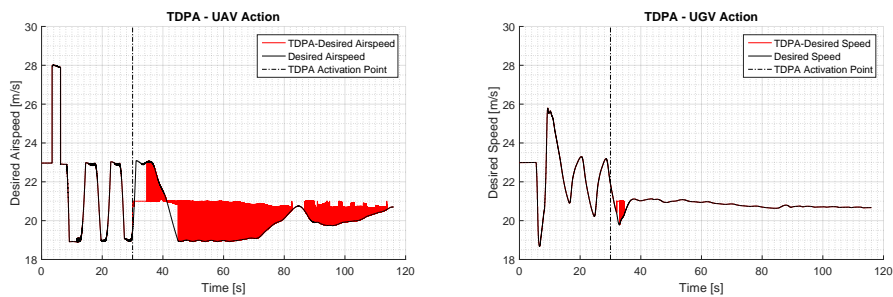
Figure 5.10 Fail-safe behavior of safety-based and performance-based landing systems for excessive crosswind gust during flare phase. A crosswind (right) of 15m/s gust hits the system at a relative altitude of 4m (rear view).

stability properties of the safety-based controller, this section only investigates the fail-safe behavior of the performance-based controller in the presence of high communication time delays. The analysis is performed in the horizontal plane, where the aircraft is flying at a constant altitude above the ground vehicle, while the vehicles are aligned in horizontal positions and velocities. Then the communication time delay is increased to a very high value (4s round trip delay) and a crosswind gust is applied to disturb the system. As can be seen in Figure 5.11a, the system exhibits an unstable behavior after the crosswind gust (left) event at around 5s into the simulation. At around 30s the passivity controller (TDPA) is manually activated. As can be seen in Figure 5.11a stability is restored, where the relative longitudinal position quickly converges to small values, while the lateral position takes a long time to converge. Nevertheless, stability is enforced, which was the main requirement for the passivity-based controller. Figures 5.11b and 5.11c illustrate the activity of the passivity controllers on the aircraft and ground vehicle sides, where the red lines represent the modified velocity commands. The red areas in Figure 5.11b correspond to a permanent interference of the passivity controller, which basically cuts down the desired airspeed command to the pre-defined nominal operating point of 21m/s . The area is depicted as a filled color due to the high frequency switching action of the passivity controller. As can be seen in Figure 5.11c, the ground vehicle passivity-controller exhibits a lower activity, which is mainly due to the higher inertia and damping contained in the ground vehicle dynamics. Overall, the passivity-based stabilizing controllers fulfill their design purpose and drive the system back to stability.

At this point the deterministic nominal and fail-safe performance analysis is complete. This analysis allowed for gaining important insight into the time domain characteristics of the different landing systems, but only covers specific and few test cases. It therefore does not allow for drawing general conclusions about the expectable overall performance of the system. For the determination of robust performance bounds a stochastic landing performance analysis is carried out in the next section.



(a) A crosswind gust (left) hits the synchronized system at 5s into the simulation. At around 30s into the simulation (vertical dashed line) the passivity-controller (TDPA) is activated and stabilizes the system.

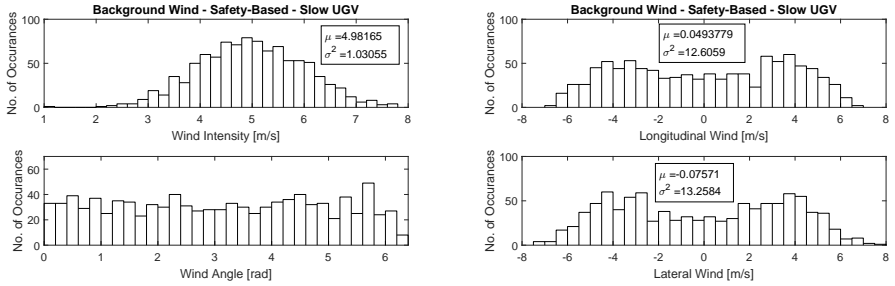


(b) Control action of passivity-controller (TDPA) on the aircraft. (c) Control action of passivity-controller (TDPA) on the ground vehicle.

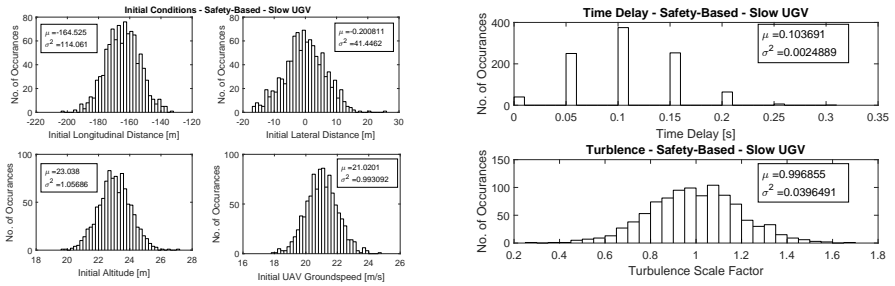
Figure 5.11 Fail-safe behavior of the performance-based landing system for excessive communication time delay. A round trip delay of 4s was applied and a crosswind gust was used to perturb the previously aligned vehicles.

5.5 Landing Robustness Analysis

In this section the results of a landing robustness comparison between the safety-based and performance-based landing controllers are presented. The corresponding Monte Carlo simulations contain 3000 test points for the simulation test cases with fast ground vehicle actuation, and 1000 test points for simulation test cases with slow ground vehicle actuation. The given system inputs and parameters are randomly varied, as illustrated in Figure 5.12.



(a) Random wind intensity and direction in polar coordinates. (b) Random wind intensity and direction in cartesian coordinates.



(c) Random initial conditions in terms of relative positions and aircraft velocity. (d) Randomly varying time delay and turbulence scale factor. Discrete values were chosen in accordance to the real estimates.

Figure 5.12 Random system inputs and parameters as used in Monte Carlo simulation for landing robustness analysis. These distributions correspond to the safety-based simulation for the slow ground vehicle actuation case, but are virtually identical for all other test cases. Further illustrations are thus omitted for brevity.

In contrast to the stochastic disturbance rejection analysis no discrete wind gust was used here to maintain a generic character of the results. Instead, turbulence (von Karman model) is added to the background wind and multiplied with a scale factor randomly varying around 1 with a variance of 0.04, as shown in Figure 5.12d.

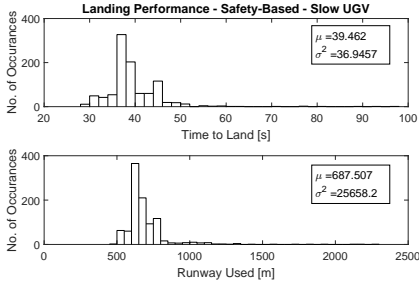
As can be seen in Figure 5.12a, the wind intensity is randomly varied around a mean

value of $5m/s$ (which corresponds to the worst-case design requirement S-R10 from Table 2.1), while the wind direction is uniformly distributed and does not favor a specific direction. The time delay is also randomly varied around a mean value of $100ms$, which was previously estimated as the nominal delay from outdoor measurements. Again, discrete time delay values were assumed in accordance to the estimates available on the real systems. Also the initial aircraft position is randomly varied to cover the inaccuracy in reaching the desired landing start point. The longitudinal relative initial position varies around a mean value of $165m$, the relative lateral position around $0m$, and the relative vertical position around $23m$, as shown in Figure 5.12c. These mean values and corresponding variances were taken from a number of flight experiments, which were performed to measure the maneuver start point attainment accuracy. Also the aircraft initial velocity is varied to increase the robustness of the results. The same procedure was applied for the fast ground vehicle actuator and the performance-based cases, while the corresponding random variables are not plotted here, since they consist of virtually the same distributions as shown for the safety-based case.

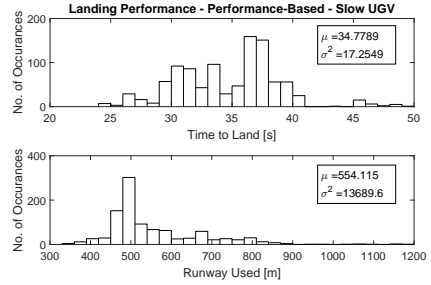
Figure 5.13 shows the results for the slow ground vehicle actuation case in terms of different time domain performance indicators, including time to land, runway used, and position and velocity error at touchdown. Based on these metrics a generalized and robust landing performance comparison can be performed.

As can be seen in Figure 5.13, the performance-based controller exhibits a better performance in all of the metrics, despite some metrics exhibiting very similar values. An improvement of around 12% is found for the landing time, which corresponds to an absolute improvement of around 5s. The performance-based system uses around 20% less runway, which corresponds to around 113m. The position error at touchdown is reduced by around 60% in longitudinal and 23% in lateral direction for the performance-based controller. Again, it should be noted that this only corresponds to an improvement of around 8.5cm and 1.7cm for the longitudinal and lateral directions, respectively. The velocity error at touchdown is virtually the same in longitudinal and reduced by 88% in lateral direction for the performance-based controller. It should be noted that this corresponds to an improvement of only 6cm/s for the lateral direction. The vertical velocity at touchdown is almost identical for both controllers, where a mean value of around $-0.9m/s$ is found. The observed variances are in general significantly smaller for the performance-based controller, which implies a more accurate and consistent performance.

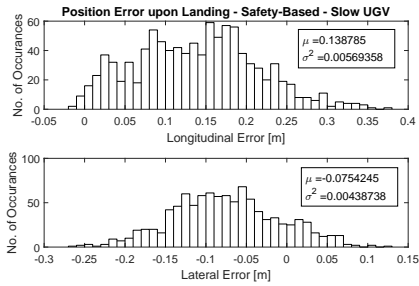
Overall the performance-based controller provides a slightly better performance in all metrics. In terms of absolute values the improvement is much less pronounced, and should again be put into context with respect to the much higher implementation and certification effort for the performance-based controller. Also, if the available runway is sufficiently long, the improvement might not be very meaningful. **Given the slow actuator dynamics of the demonstrator ground vehicle and the available runway of more than 1km it was decided to implement the safety-based architecture for first landing experiments.**



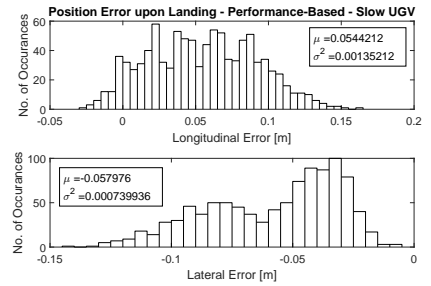
(a) Landing performance (safety-based).



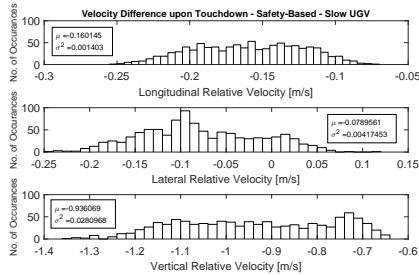
(b) Landing performance (performance-based).



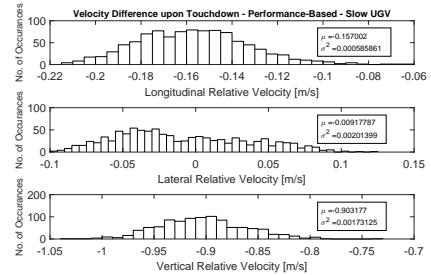
(c) Position error at touchdown (safety-based).



(d) Position error at touchdown (performance-based).



(e) Velocity error at touchdown (safety-based).



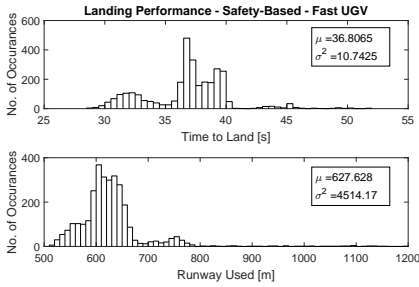
(f) Velocity error at touchdown (performance-based).

Figure 5.13 Results of Monte Carlo simulation for robust landing performance comparison for **slow ground vehicle actuator**. Different time domain performance metrics are shown for the safety-based and performance-based landing controllers.

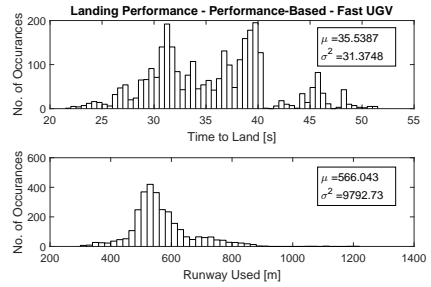
Nevertheless, an outlook on the expectable performance in case of an autonomous ground vehicle (fast actuation case) shall be provided. Therefore the same analysis as shown for the human-actuated (slow actuation case) ground vehicle is repeated for the fast ground vehicle case. The results are shown in Figure 5.14.

Again, the performance-based controller exhibits a better performance in all of the metrics, despite some metrics exhibiting very similar values. The similarities are even more pronounced than for the slow ground vehicle case shown in Figure 5.13. An improvement of only 3.5% is found in terms of landing time, which corresponds to an absolute improvement of around 1.2s. The performance-based system uses around 10% less runway distance, which corresponds to around 61m. The position error at touchdown is reduced by around 55% in longitudinal and 27% in lateral direction for the performance-based controller. Again, it should be noted that this only corresponds to an improvement of around 4.5cm and 8cm for the longitudinal and lateral directions, respectively. Interestingly, the velocity error at touchdown in longitudinal direction is 39% lower for the safety-based controller, while a reduction of around 89% is found in the lateral direction in favor of the performance-based controller. It should be noted that this corresponds to an improvement of only 6cm/s for the lateral direction. The vertical velocity at touchdown is almost identical for both controllers, where a mean value of around $-0.9m/s$ is found. The observed variances are in general significantly smaller for the performance-based controller, which implies a more accurate and consistent performance.

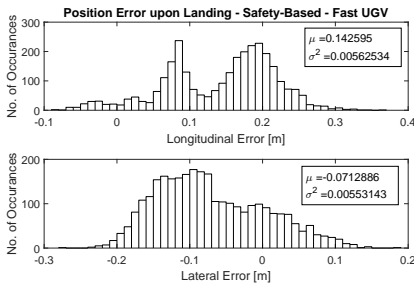
Overall, the performance-based controller again provides a slightly better performance in all metrics besides the longitudinal velocity error at touchdown, which corresponds to a very small difference in absolute values. Interestingly, when comparing the landing performance for the slow ground vehicle actuator (Figures 5.13a and 5.13b) with the fast actuation case (Figures 5.14a and 5.14b), no significant improvement is found. **These results indicate, that the semi-autonomous human actuation concept constitutes a valid approach, and that an automation effort of the current ground vehicle would not lead to a significant improvement.** It should be noted, that a considerable improvement can be expected for a ground vehicle with faster overall dynamics (not only faster actuators).



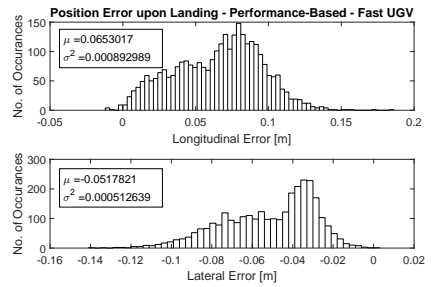
(a) Landing performance (safety-based).



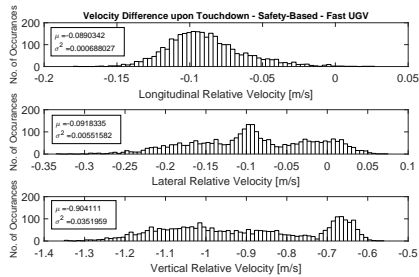
(b) Landing performance (performance-based).



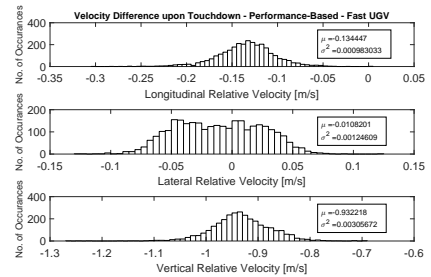
(c) Position error at touchdown (safety-based).



(d) Position error at touchdown (performance-based).



(e) Velocity error at touchdown (safety-based).



(f) Velocity error at touchdown (performance-based).

Figure 5.14 Results of Monte Carlo simulation for robust landing performance comparison for **fast ground vehicle actuator**. Different time domain performance metrics are shown for the safety-based and performance-based landing controllers.

Chapter Conclusions

Throughout this chapter a large amount of simulation experiments have been performed (10000+ tests points). The disturbance rejection performance of the safety-based and performance-based landing system was investigated by means of deterministic simulation experiments, which allowed for a detailed investigation of the system dynamics. Then robust disturbance rejection performance bounds were estimated by means of extensive Monte Carlo simulations. In a second step the overall landing maneuver was analyzed in a deterministic way to get a deeper insight into the overall landing performance under controlled nominal and failure conditions. The implemented fail-safe logic (contained in the landing state machine and the passivity-based stabilizing controller) was analyzed and validated. In a final step robust landing performance bounds were estimated. The main conclusion of the performed simulation experiments is that for the current demonstrator setup, the safety-based landing system will provide a very satisfactory performance. In addition the reduced system complexity and thus increased operational safety constitutes an important argument in favor of the safety-based solution. The safety-based landing system was therefore chosen and implemented for the final landing experiments. The performance-based landing system was found to provide a potential performance improvement of around 20%, which is not a decisive factor for this work, since a sufficiently long runway is available. For short runways or future mobile runway solutions as proposed in Chapter 1.3.3, the performance-based system represents an interesting option. Also, it was found that a faster ground vehicle actuation system would not significantly improve the landing performance for the demonstrator setup, which validates the semi-autonomous ground vehicle steering concept utilizing a human driver. The analysis presented in this chapter did not include a ground vehicle with faster overall dynamics, which might benefit more from the performance-based approach. Such analysis remains a topic for future work.

6 Landing Experiments

Chapter Overview

This chapter provides an overview of the performed experiments which lead to the successful demonstration of the developed landing concept. As concluded in the previous chapter, all landing experiments were performed with the safety-based landing controller. Before the actual landing with the final demonstrator setup was attempted, a set of preliminary real world experiments were performed to analyze the landing platform concept and other aspects of the landing maneuver individually. The preliminary tests included structural testing of the landing platform during high velocity road tests, a qualitative investigation of the ground vehicle disturbance effect by means of visualization (using wool threads) and by close proximity flight with a smaller RC-aircraft, the validation of the vision and network-based state estimation system, as well as the validation of the ground lock concept. In a second step a reduced hardware in the loop setup was used to gain further insights into the expectable system behavior, where a number of experiments with the real ground vehicle and a simulated aircraft were performed. These tests allowed for a validation of the ground vehicle steering concept. Then the final demonstrator setup was used in further preliminary tests, which included landing experiments on top of a virtual landing platform, where the aircraft perceives the landing platform at a given vertical distance above the real landing platform. Finally a set of real landing experiments is presented, which allowed for the validation of the developed landing concept.

6.1 Preliminary Experiments

A set of preliminary experiments have been performed prior to the final landing experiments to confirm the simulation results and to analyze specific aspects of the overall system.

These experiments will be presented in the following.

Validation of Landing Platform Concept

The developed landing platform concept has been tested in a series of road trials to confirm its safe structural and dynamical properties. Figure 6.1 shows the final ground vehicle setup with the landing platform mounted on top of the car.



Figure 6.1 Landing platform mounted on top of ground vehicle.

In this configuration the ground vehicle was exposed to different longitudinal and lateral accelerations and disturbances for velocities up to 120km/h . The lateral disturbances were deliberately produced through sudden steering wheel inputs at high velocities as shown in the middle picture of Figure 6.2. The actual landing maneuver will take place at around 70km/h , leaving a very comfortable safety margin. The ground vehicle with the mounted landing platform showed good and stable behavior in all test cases, where potential adverse effects of the landing platform on the ground vehicle dynamics were not encountered.



Figure 6.2 Landing platform validation tests performed at velocities of up to 120km/s .

Visualization of Flow Field around Ground Vehicle

The purpose of the next experiment was to analyze the ground vehicle induced aerodynamic disturbance and to confirm the results obtained from the CFD simulation described in Chapter 3.3. Wool threads have been attached to the landing platform at different characteristic locations and to a long stick which was then manually moved to locations of interest. The experiment has been performed at the desired landing velocity of around 70km/h . This type of analysis only provides a qualitative evaluation of the present airflow situation but allowed for a rough confirmation of the results found in simulation. The

bell shaped form of the disturbance (a positive angle of attack over the windshield and a negative angle of attack over the rear part of the ground vehicle) and the extent of the overall disturbance in negative z-direction (upwards) were confirmed.



Figure 6.3 Flow visualization with wool threads.

Close Proximity Flight with Small UAV

The following experiments were performed to further investigate the ground vehicle induced disturbance effect on a flying vehicle. Therefore, a small unmanned aircraft of less than 1kg takeoff weight was manually piloted and placed into the disturbed area. It was important to position the safety pilot in the close proximity of the ground vehicle during the whole experiment, which was accomplished by means of a follower vehicle. To guarantee good visibility for the safety pilot a convertible car was used. The experimental setup is shown in Figure 6.4.



Figure 6.4 Flight experiments with small manually controlled aircraft inside ground vehicle induced disturbance effect.

The small aircraft was manually flown towards the ground vehicle in a slow and steady manner to observe the expected effect as precisely as possible. The encountered ground

vehicle induced disturbance was found to be present and, as expected, locally limited to the very close proximity of the ground vehicle. The feedback from the manual pilot was that the disturbance did not pose any major problems during the steady approach and subsequent flight on top of the landing platform. It was therefore decided to attempt a manual landing with the small aircraft, which was completed with success. This experiment allowed for a more reliable risk assessment (with a lower risk rating) regarding a landing with the much heavier and thus less wind-prone demonstrator aircraft.

Validation of Ground Lock Concept

The ground lock concept was tested in a number of indoor and outdoor experiments. Figure 6.5 illustrates a drop test, where the correct net tension was determined and the reliable functioning of the mechanical locking device tested.



Figure 6.5 Drop testing of ground lock mechanism.

Additionally a number of road tests were performed, where the aircraft was placed onto the landing platform during still-stand, and accelerated to landing airspeed (around 70km/h) together with the ground vehicle. Then, while moving at high speed, the ground lock concept was investigated for the post-touchdown and deceleration phases with the aircraft lying freely on top of the landing platform (secured with a slack rope at the nose of the aircraft). These tests have been described in more detail in Chapter 4.2.5.

Validation of Relative State Estimates

The relative state estimation system has been tested under the final real world conditions. The image shown in Figure 6.6 has been taken from the aircraft. As can be seen, the optical marker is located under the landing platform net and is thus slightly obscured. This effect might impair the performance and accuracy of the developed estimation methods. Therefore, a set of tests have been conducted under these final conditions in order to evaluate the marker recognition and tracking performance.

Figure 6.7 shows a series of images taken at 30 frames per second. Each image corresponds to one consecutive frame. As can be seen the marker is tracked continuously and the relative position is calculated.



Figure 6.6 Optical marker as seen from the aircraft.

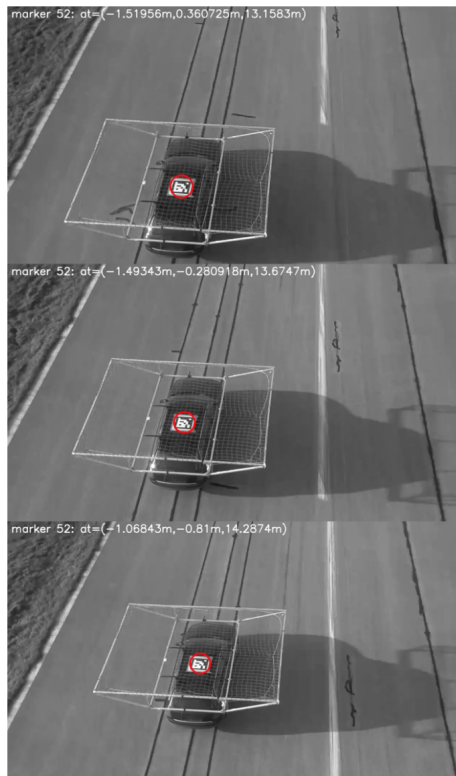


Figure 6.7 Continuous tracking of optimal markers.

Conclusions of Preliminary Experiments

The following conclusions were drawn for the preliminary real world experiments presented in the previous sections:

- The landing platform is well suited for the planned cooperative landing maneuver

- The ground vehicle induced disturbance does not seem to pose a major problem, as was expected from the CFD simulation results
- The ground lock concept is valid and the locking mechanism reliable
- The vision-based relative state estimation approach works reliably under real world conditions

Also, important insight into different operational aspects (e.g. overall experimental setup, operational roles, rules and procedures, safety-pilot positioning and communication) of the planned cooperative landing experiment has been gained throughout the preliminary experiments.

6.2 Experiments with Hardware-in-the-Loop Setup

As described in Section 5, the proposed landing system has been successfully tested in a large number of simulation experiments. In the next step, a so-called Hardware In the Loop (HIL) test setup has been prepared, where the real ground vehicle is used together with a *virtual aircraft* [92]. This virtual aircraft is running as a real-time process on a second machine (identical to the aircraft flight control computer) inside the ground vehicle, which is connected to the communications network via a wireless data link. This machine has been located inside the ground vehicle to create similar signal paths as could be expected during the final phase of the real landing experiments (close proximity of both vehicles).

A set of 15 HIL-landing experiments have been successfully performed, where different controller configurations were tested and system parameters tuned. The concept allowed for an investigation of the expectable landing system performance with very little risk involved, since only a virtual aircraft was performing the landing maneuver. Figure 6.8 shows the results of an exemplary landing experiment with the safety-based landing controller. The x-axis points in the direction of the runway, the y-axis to the right and the z-axis downwards. The desired landing airspeed was set to 21m/s .

The main goals for the HIL-experiments consisted in the determination of a first set of realistic ground vehicle controller gains and the validation of the human actuator concept for ground vehicle steering. As can be seen in the top of Figure 6.8, the landing maneuver starts with the aircraft at around 300m behind the ground vehicle, which is at rest at the beginning of the maneuver. At around 12 seconds into the landing maneuver, the ground vehicle starts accelerating and the relative velocity in runway direction starts decreasing. Around second 21 both vehicles are aligned in position and velocity and remain in this condition until the touchdown is achieved at around 37 seconds. At touchdown (relative z-position is equal to zero) the remaining position deviation in y-direction (lateral to the runway) is around 0.7m , while the position error in x-direction is close to zero. As the aircraft simulation is not stopped at touchdown, relative position and velocity start to increase again. Overall, it can be concluded that the ground vehicle steering concept involving the human driver seems to be a valid solution.

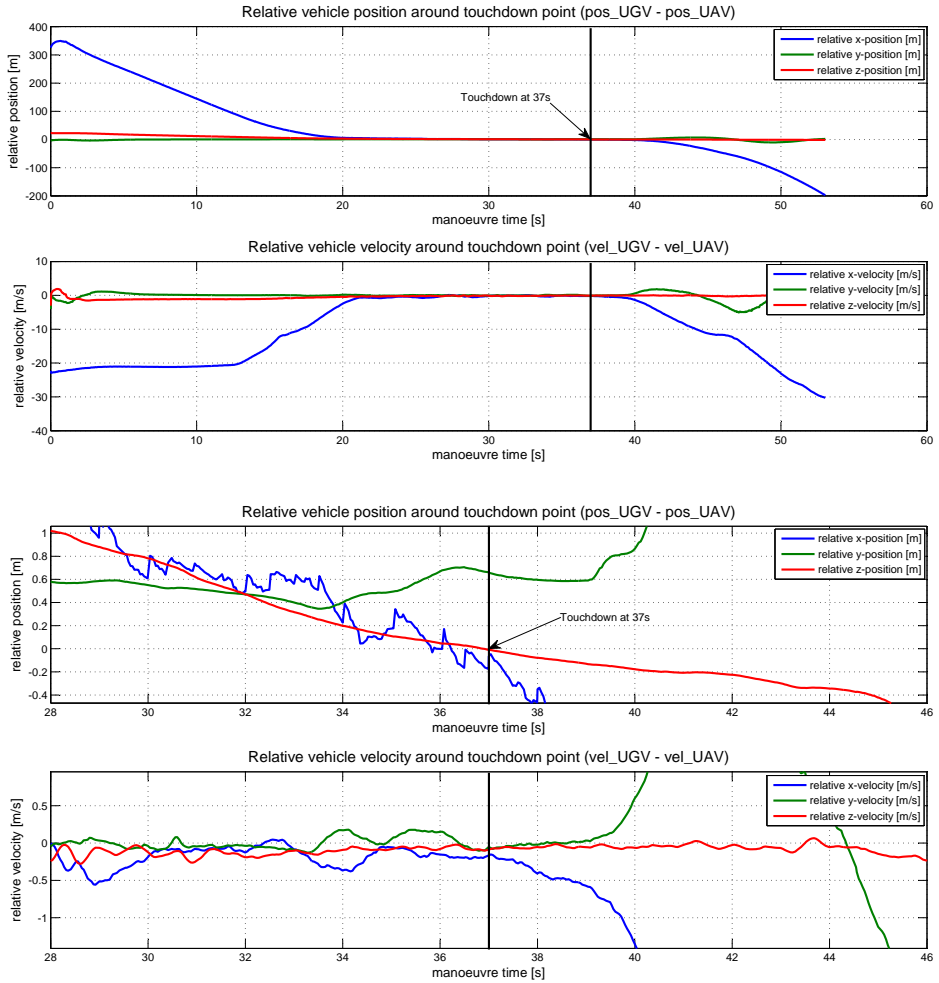


Figure 6.8 Relative vehicle position and velocity for a virtual landing maneuver with the safety-based landing controller. The overall maneuver is shown at the top, while a detail around touchdown is shown at the bottom.

6.3 Experiments with Final Demonstrator Setup

As described in the previous sections, the proposed landing system has been successfully tested in a number of Hardware In the Loop (HIL) experiments, where the real ground vehicle was used together with a virtual aircraft. The concept allowed for an investigation of the expectable landing system performance with very little risk involved, since only a virtual aircraft was performing the landing maneuver. With the successful completion of the HIL-experiments, the next set of experiments with the full demonstrator setup, as illustrated in Figure 6.9, was initiated. These experiments included a series of virtual landing trials without a physical touchdown (real aircraft lands on top of a vertically offset *virtual* ground vehicle), as well as the final landing experiments with an actual touchdown. A large number of virtual landing experiments were performed, while the offset between the virtual and real ground vehicle was successively reduced to the point, where the aircraft would perform its virtual touchdown at only $1m$ above the real landing platform. This low altitude trials also allowed for validating the positioning accuracy within the ground vehicle induced disturbance effect, which was found to lie well below $\pm 0.8m$. In the following only the complete landing experiments (with touchdown) will be described in detail for the sake of brevity.



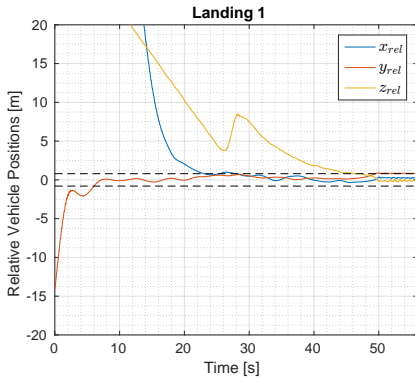
Figure 6.9 Experimental setup for final landing experiments.

The landing was successfully performed multiple times. The required landing precision of $\pm 0.8m$ was consistently achieved. As described in Section 4.2.6, the landing maneuver was automatically controlled via a high-level state machine logic. During a typical suc-

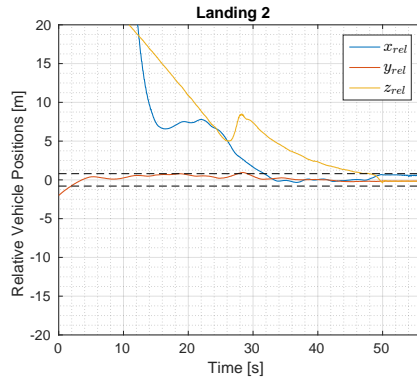
successful landing maneuver sequence, the different states are passed through successively in the following order: WaypointMode, Descent, Flare, Retard, and GroundLock. This sequence will generate a descent path within the previously defined *inner fence* (different green areas in Figures 4.12 and 4.13). The desired landing airspeed was set to 21m/s throughout all performed landing experiments. In the following, two exemplary landing attempts are described in detail.

The relative vehicle positions and velocities during the last 50s before touchdown are shown in Figure 6.10 for both exemplary landing attempts. The first attempt (*Landing 1*) is shown on the left, while the second attempt (*Landing 2*) is shown on the right. *Landing 1* will be described first. As can be seen in Figure 6.10a, the flare phase was entered (relative altitude below 5m) at around 25s. At that moment the required precision of $\pm 0.8\text{m}$ in x- and y-direction (horizontal dotted lines) was not given and an automatic retry command was issued. The aircraft is thus commanded to climb above 6m in order to start a second approach. During the second approach the relative x- and y-errors remain within the required bounds and the landing is achieved with a touchdown at around 50s. The exponential flare law is nicely visible in the altitude profile between second 33 and 48. The effects of retard and ground lock mode activation can be seen during the last two seconds before the touchdown at around 48s, where the relative altitude decreases rapidly. After second 50 the aircraft is lying in the landing platform (slightly negative relative altitude due to downwards extended net), the small oscillations occurred during the deceleration process. The relative vehicle velocities during the first exemplary landing attempt are shown in Figures 6.10c and 6.10e. As can be seen, the maneuver starts with a relative x-velocity (blue line) of around 21m/s , where the ground vehicle starts accelerating at around 9s and reaches the aircraft velocity at around 19s. The aircraft is descending with a constant vertical velocity (yellow line) of around -1m/s between second 8 and 26, which corresponds to the commanded 3° descent slope. The retry command can be clearly seen at around 27s, where the aircraft is commanded to climb above 6m for a second attempt. The vertical velocity profile during the flare is also visible as an exponential curve between seconds 33 and 48, as shown in Figure 6.10e). The small oscillations after the touchdown at around 52s are due to the aircraft bouncing slightly on top of the landing platform during the deceleration process.

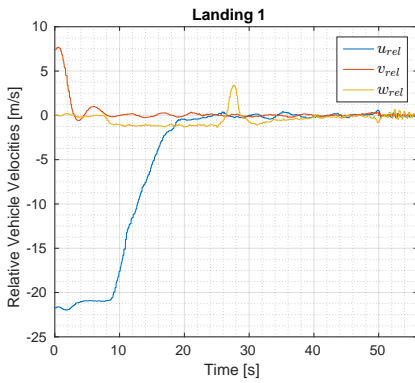
Figure 6.10b shows the relative vehicle positions for the second successful landing attempt. As can be seen, flare mode is activated at around 26s, when the flare initiation altitude of 5m is reached. At that moment, the relative x-position deviation still lies at around 5m which is far beyond the required bound of $\pm 0.8\text{m}$. This large deviation is due to a rather poor initial tracking performance of the human driver, where the car is accelerated too much as can be seen in Figure 6.11b. Anyway, the driver is able to recover the initial error and continues to track the given commands reasonably well from second 26 onwards. The automatic retry command can be seen in Figure 6.10b at around 26s, where the aircraft is commanded to climb before the second attempt is started. Flare mode is entered for a second time at around 33s, where the flare initiation altitude of 5m is reached. In the meantime the two vehicles were able to align and the relative x- and y-positions lie within the required bounds. The altitude profile nicely shows the exponential character of the flare law until the touchdown is produced at around 50s (yellow line crosses zero).



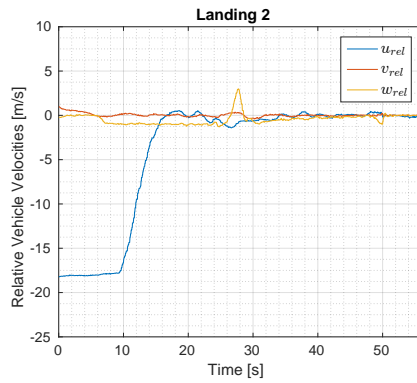
(a) Relative vehicle positions (Landing 1).



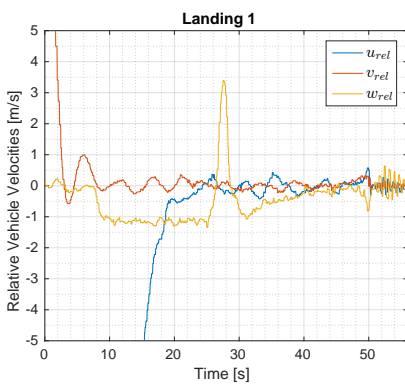
(b) Relative vehicle positions (Landing 2).



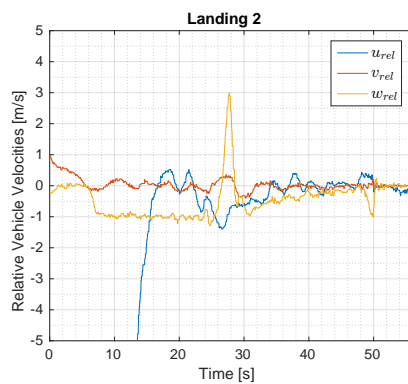
(c) Relative vehicle velocities (Landing 1).



(d) Relative vehicle velocities (Landing 2).



(e) Zoom of relative vehicle velocities (Landing 1).

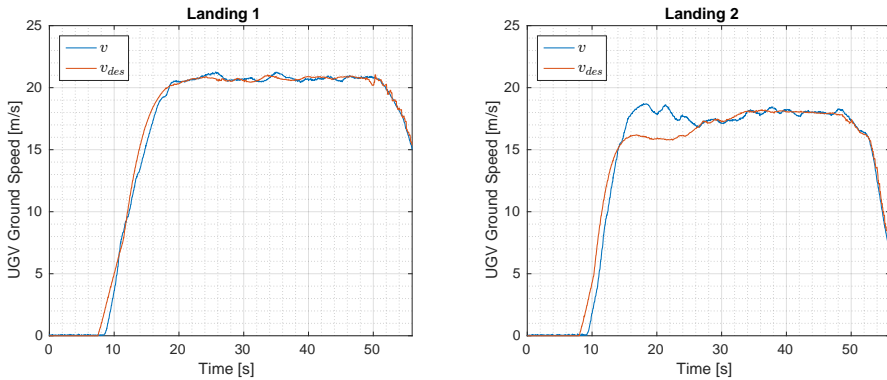


(f) Zoom of relative vehicle velocities (Landing 2).

Figure 6.10 Relative vehicle positions and velocities in local runway frame during exemplary landing attempts with safety-based landing controller. Touchdown occurs at $t = 50s$.

Again, the rather sudden drop of altitude in the last two seconds before touchdown is due to the additional vertical velocity induced by the retard and ground lock modes. As can be seen, the relative position deviation in x- and y-direction was well below $\pm 0.8m$ between second 32 and 48. In the last two seconds before touchdown, the relative x-position is slightly increased by around $0.5m$, which is again caused by the human driver. As can be seen in Figure 6.11b, there is a small upwards notch in actual ground vehicle velocity at around 48s, right before the deceleration process starts. The relative vehicle velocities during the second landing attempt are shown in Figures 6.10d and 6.10f. As can be seen, the maneuver starts with a relative velocity in x-direction of around $18m/s$, where the ground vehicle starts accelerating at around 10s and reaches the aircraft's velocity at around 17s. In comparison to the first landing attempt, where the relative velocity was around $21m/s$, a decrease has occurred. This indicates changed wind conditions, where a headwind of $3m/s$ was present during the second landing. Again, the retry command is clearly visible, with a spike in the relative vertical velocity (yellow line) at around 28s. The 3° descent slope is visible again, represented by the $-1m/s$ constant vertical velocity between second 7 and 26. The second attempt is initiated at around 33s, where the exponential flight path starts. The touchdown is produced at around 50s, as shown in Figure 6.12.

The ground vehicle velocity tracking performance during the selected landing attempts is shown in Figure 6.11, demonstrating the validity of the human actuator concept. Figure 6.12 illustrates the successful touchdown as seen from the landing platform.



(a) Ground vehicle desired vs. actual velocity (Landing 1). (b) Ground vehicle desired vs. actual velocity (Landing 2).

Figure 6.11 Ground vehicle desired vs. actual velocity for the selected landing attempts.

After the first set of landing experiments were concluded [90], a couple of changes have been implemented to further improve the performance of the landing system [91]. The initially applied constant climb angle of 10° during a low-retry (below h_{flare}) was affecting the system performance, as it would induce a rather abrupt climb maneuver, even if the aircraft was still high above the landing platform. Although the flight controller is



Figure 6.12 Successful touchdown on top of the landing platform.

able to decouple aircraft responses in velocity and altitude, sudden changes are not always perfectly handled. An adaptive climb angle was therefore introduced, where the UAV would only climb abruptly when close to the landing platform and smoother when further away, as shown in Equation (6.1).

$$\gamma_{climb} = (h_{flare} - h) \cdot \frac{\pi}{72} \quad (6.1)$$

This results in a hold altitude behavior at $h = h_{flare}$ and a commanded climb angle of 12.5° at $h = 0m$. Therefore, if the aircraft is pushed out of the inner fence margins just after crossing h_{flare} at $5m$ altitude, it will try to recover the horizontal misalignment while maintaining altitude and minimizing parasitic effects on the horizontal controller performance.

The performance of the vision-based state estimates is compared to GPS data and the fused solution in Figure 6.13 for yet another successful landing attempt.

As can be seen, all three solutions lie close together with very small deviations during the last meters before touchdown, while no retry command was issued during this landing attempt. The fiducial marker tracking works well down to a relative altitude of around $1m$, where the markers cannot be tracked anymore. It should be noted that even though a very precise RTK-GPS system was used during these experiments, the relative position estimates depend on the accuracy of the lag estimate which could possibly induce smaller errors. The comparison of the GPS-based solution to the purely vision-based estimates confirms the proper functionality of the lag correction method. The fused solution is optimally suited for a use in the coupled controller, as it accounts for GPS or vision system

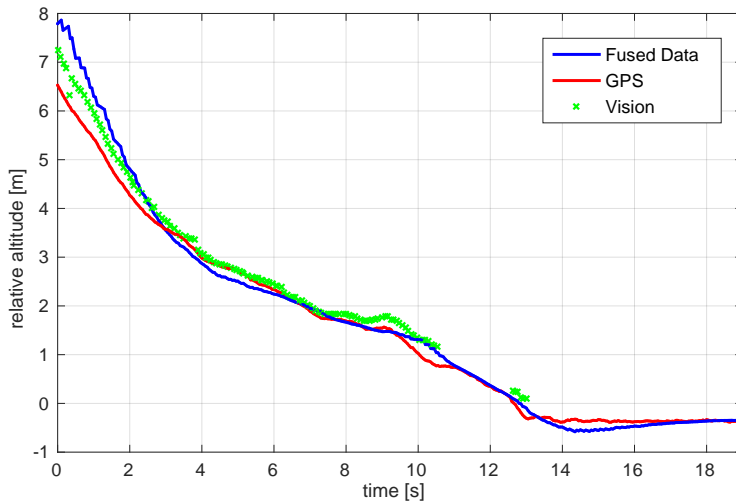


Figure 6.13 Comparison of relative altitude estimates based on 1. GPS data exchanged via the network (red), 2. pure optical marker tracking (green), and 3. combined solution by sensor data fusion (blue). Around $t = 13s$ the touchdown is performed.

outages and ensures a continuous mode of operation. A landing with degraded GPS performance is therefore also possible, which significantly increases the reliability and safety of the landing system.

Chapter Conclusions

The validity of the developed landing concept was demonstrated in multiple successful landing experiments. The achieved landing accuracy was within $\pm 0.8m$, which is precise enough to perform safe and reproducible landing maneuvers with the current setup. Synchronized close proximity flight was possible with positioning accuracies of less than $0.8m$ despite the ground vehicle induced aerodynamic disturbance, which was found to correspond well to the predictions from CFD simulations. Thereby the aircraft was fully immersed in the ground vehicle induced aerodynamic disturbance. The flight controller efficiently compensated for the ground vehicle induced disturbance. The relative state estimation algorithm performed well and the fiducial marker tracking method produced precise relative state estimates in the centimeter range during the most critical part of the landing, where GPS dropouts are most likely to happen ($1m < h_{rel} < 8m$). The lag compensation method worked reliably. The ground lock system performed as expected and secured the aircraft in the landing platform during the deceleration phase until a full stop was reached.

It can be concluded that autonomous landings on the mobile landing platform are feasible with the current experimental setup and that the system is suitable for a practical application.

7 Conclusions and Further Development

This thesis investigated the main technical inhibitors regarding a viable commercial application of high altitude platforms and proposes a possible solution. High altitude platforms suffer from low payload capacity and highly weather dependent operational availability, where especially the crosswind landing sensitivity poses problems. The goal of this thesis thus consisted in the development of a new landing system for high altitude platforms, which solves the aforementioned problems. The main results can be grouped into three areas, which will be described in detail in the following. An outlook on possible future developments is also provided.

Concept Development

First an appropriate landing concept was developed, including the removal of the landing gear from the aircraft and the introduction of a ground-based mobile landing platform. Currently established landing procedures and their limitations, as well as related applications from the field of cooperative motion control were analyzed. This led to the development of two distinct landing strategies — the safety-based and the performance-based strategies. Each approach follows a different logic, where the safety-based landing concept adheres to established landing procedures and keeps the system complexity low, while offering time delay independent stability properties. The focus is on the operational safety of the landing system. The performance-based landing concept promises the minimization of the used runway, but suffers from time delay-dependent stability characteristics. A reduced systems engineering approach was applied for the development of both systems, including the definition of high-level requirements and a functional decomposition for the identification of the required functional building block on a lower level. The system specification was then derived and formed the basis of the subsequent detailed development process.

Control Strategy Development

The functional requirements from the concept development phase were then used for the detailed development of the individual vehicle control systems and the cooperative landing controller. The developed cooperative landing strategies required independent controllability of the translational degrees of freedom for both involved vehicles. In the case of the aircraft this functionality is not obvious and was achieved through the implementation of an energy-based flight control system. A reliable relative state estimation system based on fiducial markers and GPS sensor data was developed and tested in outdoor experiments. The safety-based and performance-based landing control systems were developed and their stability was assessed. The inter-vehicle communication time delay was found to affect the stability of the performance-based controller. A passivity-based stabilizing controller, extending on the classical time domain passivity control approach, was developed to enforce stability in case of high communication delays. The application of the energy-based passivity-control approach was enabled by an appropriate choice of the aircraft flight controller, which also operates on energy level. Finally, a high-level mission control and fail-safe system was developed to automatically guide the vehicles through the different landing phases and to ensure the operational safety of the landing system.

Modeling, Simulation and Experimental Validation

The developed control concept was analyzed and validated in realistic simulation and landing experiments. The simulation experiments included a deterministic performance analysis, which allowed for a thorough investigation of the landing system behavior on dynamical level, as well as a stochastic analysis for the determination of robust performance bounds. The safety-based landing system showed very satisfactory performance and was able to cope with the worst-case wind conditions specified in the system requirements. The performance-based landing system was found to provide an overall improvement of around 20% in terms of used runway, when compared to the safety-based solution. A practical relevance of this advantage mainly depends on the available runway length and the required landing distance, which in the case of the given experimental setup (slow aircraft on long runway) was not an issue. The analysis was performed for a slow ground vehicle actuation case, representing the semi-autonomous human-driven ground vehicle, and a fast actuation case, representing a hypothetical autonomous vehicle. For the given demonstrator setup, no significant improvement through the use of an autonomous ground vehicle was found, which validated the semi-autonomous steering concept as used for the demonstrator setup. The results from the simulation analyses led to the selection of the safety-based architecture for the demonstrator setup. A number of hardware in the loop experiments were performed with the real ground vehicle and communication systems and a virtual aircraft. This allowed for validating the ground vehicle steering concept in practice. Then, a series of virtual landing experiments with the full experimental setup were performed, where the real aircraft landed on a (vertically offset) virtual landing platform. Finally, several successful landing experiments were conducted, thus validating the proposed landing system. To the author's best knowledge it was the first time that such

technology was demonstrated on the given scale, worldwide. The public response to the disseminated results was significant and confirms the relevance of this technology.

Future Development

Possibilities for a future development of the system include the extension of the landing concept to larger and heavier aircraft systems. This implies the development of an appropriate mechanical locking device with pre-defined docking points on the ground and aircraft side, which are able to withstand the given structural loads. This also requires the introduction of an actuated landing platform, which can adapt its attitude and position (within certain limits) to the aircraft motion, to allow for an improved docking accuracy. Such system will be developed in the follow-up project *ElektRail* (see Chapter 1.3.3), where a mobile runway system will be developed to allow for an infrastructure independent operation of heavier unmanned aircraft systems. In this setup the mobile landing platform will be fully automated and possesses much faster dynamics than the demonstrator setup used in this work. Such mobile runway system could also be used to enable very short takeoff and landing capabilities for smaller manned fixed-wing aircraft and thus extend the current urban air mobility concepts, which are dominated by vertical takeoff and landing systems, with more efficient fixed-wing solutions. Also, such system could be used in the maritime sector to enable the takeoff and landing of fixed-wing aircraft on medium-sized ships.

Appendix A

Simulation Parameters

The following chapter presents the controller settings used for all simulation experiments throughout this chapter.

A.1 Controller Parameters

Table A.1 Aircraft controller gains.

Parameter	Value	Parameter	Value
K_q	0.0502	K_ϕ	4.2968
K_θ	0.1511	$K_{i\phi}$	0.0076
$K_{i\theta}$	0.0001	K_χ	0.4566
K_τ	1.3614	$K_{i\chi}$	0.0002
$K_{i\tau}$	1.0251	K_{KTP}	1
δ_{FF}	0.4926	K_{KTI}	1.1711
K_{Va}	0.5175	K_{KEP}	1
K_h	0.2046	K_{KEI}	1.1711
K_p	0.1166		

Table A.2 Ground vehicle controller gains.

Parameter	Value	Parameter	Value
K_T	0.3699	K_S	1.5801
K_{iT}	0.0201	K_{iS}	0.0001

Table A.3 Safety-based position synchronization controller gains (fast ground vehicle actuator).

Parameter	Value	Parameter	Value
K_{pXG}	-4.984259	K_{pYA}	2.525233
K_{iXG}	-0.034398	K_{iYA}	0.041100
K_{vXG}	-2.591996	K_{vYA}	3.973529

Table A.4 Performance-based position synchronization controller gains (fast ground vehicle actuator).

Parameter	Value	Parameter	Value
K_{pXG}	-4.532453	K_{pYG}	-4.961942
K_{iXG}	-0.071747	K_{iYG}	-0.000746
K_{vXG}	-2.174629	K_{vYG}	-4.917977
K_{pXA}	0.034519	K_{pYA}	4.853377
K_{iXA}	0.177558	K_{iYA}	2.581068
K_{vXA}	3.038446	K_{vYA}	3.108276

Table A.5 Safety-based position synchronization controller gains (slow ground vehicle actuator).

Parameter	Value	Parameter	Value
K_{pXG}	-4.9532	K_{pYA}	2.5261
K_{iXG}	-0.0389	K_{iYA}	0.0226
K_{vXG}	-4.9930	K_{vYA}	3.4193

Table A.6 Performance-based position synchronization controller gains (slow ground vehicle actuator).

Parameter	Value	Parameter	Value
K_{pXG}	-4.8540	K_{pYG}	-1.7120
K_{iXG}	-0.0012	K_{iYG}	-0.0043
K_{vXG}	-4.9554	K_{vYG}	-2.0906
K_{pXA}	0.0001	K_{pYA}	1.6661
K_{iXA}	0.0977	K_{iYA}	1.0645
K_{vXA}	0.9661	K_{vYA}	1.9031

A.2 Initial Conditions for Landing Performance Analysis

$$\mathbf{s}^0 = \begin{bmatrix} x_{uav}^0 \\ y_{uav}^0 \\ h_{uav}^0 \\ v_{k,uav}^0 \\ a_{uav}^0 \\ \gamma_{uav}^0 \\ \chi_{uav}^0 \\ x_{ugv}^0 \\ y_{ugv}^0 \\ h_{ugv}^0 \\ v_{k,ugv}^0 \\ a_{ugv}^0 \\ \chi_{ugv}^0 \end{bmatrix} = \begin{bmatrix} -200.0 & m \\ 2.0 & m \\ h_{ugv}^0 + h_{lp} + 20.0 & m \\ 21.0 & m/s \\ 0 & m/s^2 \\ 0 & \text{deg} \\ 0 & \text{deg} \\ 0 & m \\ 0 & m \\ 0 & m \\ 0 & m/s \\ 0 & m/s^2 \\ 0 & \text{deg} \end{bmatrix} \quad (\text{A.1})$$

A.3 MPC Controller Settings

Time Constants for First-Order Vehicle Models

$$\begin{aligned}
 \bar{v}_k &= 21.0 \text{ m/s} \\
 \tau_{\gamma, uav} &= 2.14 \text{ s} \\
 \tau_{v, uav} &= 3.44 \text{ s} \\
 \tau_{\chi, uav} &= 2.44 \text{ s} \\
 \tau_{v, ugv}^{fast} &= 1.39 \text{ s} \\
 \tau_{\chi, ugv}^{fast} &= 1.73 \text{ s} \\
 \tau_{v, ugv}^{slow} &= 1.59 \text{ s} \\
 \tau_{\chi, ugv}^{slow} &= 1.91 \text{ s}
 \end{aligned} \tag{A.2}$$

State and Input Limits for Aircraft and Ground Vehicle

$$\begin{bmatrix} x_a^{min} \\ y_a^{min} \\ h_a^{min} \\ v_{k,a}^{min} \\ a_a^{min} \\ \gamma_a^{min} \\ \chi_a^{min} \\ x_g^{min} \\ y_g^{min} \\ v_{k,g}^{min} \\ a_g^{min} \\ \chi_g^{min} \end{bmatrix} = \begin{bmatrix} -1000.0 & m \\ -1000.0 & m \\ 0 & m \\ 19.0 & m/s \\ -2.0 & m/s^2 \\ -5.0 & \text{deg} \\ -10.0 & \text{deg} \\ 0 & m \\ -3.0 & m \\ 0 & m/s \\ -2.0 & m/s^2 \\ -10.0 & \text{deg} \end{bmatrix} \leq \mathbf{s} \leq \begin{bmatrix} 1000.0 & m \\ 1000.0 & m \\ 100.0 & m \\ 35.0 & m/s \\ 2.0 & m/s^2 \\ 5.0 & \text{deg} \\ 10.0 & \text{deg} \\ 1000.0 & m \\ 3.0 & m \\ 35.0 & m/s \\ 2.0 & m/s^2 \\ 10.0 & \text{deg} \end{bmatrix} = \begin{bmatrix} x_a^{max} \\ y_a^{max} \\ h_a^{max} \\ v_{k,a}^{max} \\ a_a^{max} \\ \gamma_a^{max} \\ \chi_a^{max} \\ x_g^{max} \\ y_g^{max} \\ v_{k,g}^{max} \\ a_g^{max} \\ \chi_g^{max} \end{bmatrix} \tag{A.3}$$

$$\begin{bmatrix} v_{k,a}^{des,min} \\ \gamma_a^{des,min} \\ \chi_a^{des,min} \\ v_{k,g}^{des,min} \\ \chi_g^{des,min} \end{bmatrix} = \begin{bmatrix} 21.0 & m/s \\ -5.0 & \text{deg} \\ -10.0 & \text{deg} \\ 0 & m/s \\ -10.0 & \text{deg} \end{bmatrix} \leq \mathbf{u} \leq \begin{bmatrix} 25.0 & m/s \\ 5.0 & \text{deg} \\ 10.0 & \text{deg} \\ 25.0 & m/s \\ 10.0 & \text{deg} \end{bmatrix} = \begin{bmatrix} v_{k,a}^{des,max} \\ \gamma_a^{des,max} \\ \chi_a^{des,max} \\ v_{k,g}^{des,max} \\ \chi_g^{des,max} \end{bmatrix} \tag{A.4}$$

Danger Zone Parameters

$$\begin{aligned}
h_{safe} &= 5.0 \text{ m} \\
d_{safe} &= 0.8 \text{ m} \\
d_{land} &= 0.6 \text{ m} \\
\lambda_h^{max} &= 2.0 \text{ m} \\
\lambda_{xy}^{max} &= 0.4 \text{ m}
\end{aligned} \tag{A.5}$$

Fixed State Weights

$$p_h = 1.2327, \quad p_\gamma = 5.5201 \tag{A.6}$$

Adaptive State Weights

$$\begin{aligned}
p_x^k &= 8.767, & p_y^k &= 6.5703, & p_{v_k}^k &= 9.6992, & p_a^k &= 2.3279, & p_\chi^k &= 4.4647 \\
p_x^l &= 2.65078, & p_y^l &= 1.23267, & p_{v_k}^l &= 98.4503, & p_a^l &= 44.3752, & p_\chi^l &= 47.3969
\end{aligned} \tag{A.7}$$

Input Weights

$$q_{v_{k,a}} = 0.0012568, \quad q_{v_{k,g}} = 0.0012568, \quad q_{\gamma_a} = 1.2327, \quad q_{\chi_a} = 1.2327, \quad q_{\chi_g} = 1.2327 \tag{A.8}$$

Input Derivative Weights

$$r_{v_{k,a}} = 10.7369, \quad r_{v_{k,g}} = 7.61641, \quad r_{\gamma_a} = 14.8691, \quad r_{\chi_a} = 99.3144, \quad r_{\chi_g} = 99.3144 \tag{A.9}$$

Terminal Weights

$$w_x = 14.8691, \quad w_y = 0.0012568 \tag{A.10}$$

List of Figures

1.1	Arado Ar 234 V1 taking off from a tricycle trolley, ©	2
1.2	"Diagram of a typical left-hand traffic pattern in the US" by Thsutton, licensed under CC BY-SA 2.5	3
1.3	"PH-BFL B747 KLM heavy crosswind landing" by Maarten Visser, licensed under CC BY 2.0	4
1.4	De-crab landing technique: 1) wings level during approach, 2) nose points into the wind, 3) position is maintained using engine thrust, 4) during flare downwind rudder is applied to eliminate crab, 5) aileron is applied simultaneously to maintain wings level, 6) at touchdown aircraft, velocity vector and bank angle are aligned with runway, 7) de-crab maneuver results in lateral deviation from point of touchdown	5
1.5	Sideslip landing technique: 1) final approach using crab to avoid drift, 2) before flare heading is aligned using aileron into the wind and opposite rudder, 3) aircraft at constant sideslip angle, 4) rudder and aileron must be applied continuously to maintain sideslip angle, 5) aircraft heading aligned with runway during final approach, 6) touchdown usually with upwind main wheels touching the ground first	6
1.6	Crab landing technique: 1) approach same as during de-crab landing, 2) no de-crab before flare, only after touchdown, 3) on the ground downwind rudder is applied to eliminate crab, 4) aileron is applied immediately to maintain wings level, 5) wheel friction reduces aircraft drift in downwind direction during de-crab on the ground, 6) only recommended on wet or slippery runway, 7) ground-based de-crab maneuver results in smaller lateral deviation from point of touchdown	7
1.7	Difficult landing conditions imposed by high altitude aircraft design requirements	9
1.8	HAP aircraft <i>Zephyr UAV</i> and <i>HALE UAV Global Hawk</i>	9
1.9	Boeing B-52 Stratofortress with its steerable main landing gear during a heavy crosswind landing, ©	10
1.10	High altitude platform ELHASPA landing concept (©2016 DLR, reproduced with permission)	11

1.11	Ground-based landing gear system GroLaS designed by German company mb+Partner. (©2012 mb+Partner, reproduced with permission)	13
1.12	Conceptual illustration of mobile runway system <i>ELEKTRAIL</i> ® for runway-independent operation of heavier aircraft systems up to 600kg. (©2019 ELEKTRAIL® project consortium, reproduced with permission)	17
1.13	Demonstrator setup for landing experiments in the scope of the EC-Safemobil project (©2016 DLR, reproduced by permission)	25
1.14	Demonstrator setup for landing experiments consisting of aerial vehicle and ground vehicle with roof-mounted landing platform	26
2.1	The different phases of the cooperative landing maneuver	34
2.2	Switching conditions for landing maneuver phase transitions	35
2.3	Definition of landing controller sub-systems. Gray blocks indicate non-cooperative functions while red boxes indicate cooperative functions	40
3.1	Process of geometry-based aerodynamics modeling using the potential flow solver AVL [64]. Assuming a probability density function for the considered value ranges, the AVL software computes point estimates of relevant aerodynamic forces and moments. Then, a polynomial fit with appropriate coefficients is used to explain the data	66
3.2	Scatter plots of total lift coefficient versus different model inputs before (a,b) and after (c,d) one model structure iteration. These scatter plots are used to find corresponding dependencies of states to aerodynamic forces and moments	67
3.3	Comparison of propulsion model and static thrust measurements (left). The static thrust force against throttle is measured with a force sensor using the real hardware (right). The thrust model predicts the measured static force with a reasonable accuracy	70
3.4	Weight and balance estimation for payload and structural aircraft components. The mass is assumed to be uniformly distributed, thus point masses are distributed over the aircraft geometry to compute the moments and products of inertia	71
3.5	Overview of the two-step method for system identification. The individual steps consist of data gathering, data preprocessing, parameter identification and model validation	72
3.6	Measured versus modeled aircraft responses in terms of forces and moments for different maneuvers (sequence: phugoid, short period, thrust doublet). The results confirm that the aerodynamic forces and moments are accurately predicted	76
3.7	Measured versus simulated (linear model) aircraft states for different maneuvers (sequence: phugoid, short period, thrust doublet). The results are computed using forward simulation without the use of any measurements. As depicted, the results confirm that the identified aircraft model accurately explains the measured trajectory	77
3.8	Block diagram of a classical longitudinal flight controller. The controller consists of four cascaded loops to control altitude, flight path angle, angle of attack and pitch rate	78
3.9	Energy-based longitudinal autopilot structure	80

3.10	Block diagram of the lateral-directional flight controller. The controller consists of three cascaded loops to control course angle, bank angle and roll rate	81
3.11	Simulation of $8m/s$ speed-up (left) and speed-down (right) maneuvers at constant commanded altitude for the classical (SISO) and energy based (TECS) flight controllers	83
3.12	Simulation of $50m$ climb (left) and descent (right) maneuvers at constant commanded airspeed for the classical (SISO) and energy based (TECS) flight controllers	84
3.13	Horizontal flight with the energy based flight controller. The altitude error remains within $\pm 1m$, the airspeed error within $\pm 1.5m/s$ (commanded values are displayed as dashed lines, while actual sensor data is represented as a solid line)	86
3.14	Climb and descent with the energy-based flight controller. Both maneuvers are nicely coordinated. The airspeed tracking performance is the virtually same as observed during level flight (commanded values are displayed as dashed lines, while actual sensor data is represented as a solid line)	87
3.15	A $3m/s$ speed increase with the energy-based flight controller. The altitude tracking performance is virtually the same as observed during level flight, which validates the correct coordination of elevator and throttle (commanded values are displayed as dashed lines, while actual sensor data is represented as a solid line)	88
3.16	A $3m/s$ speed decrease with the energy based flight controller. Again, good altitude and airspeed tracking performance is observed (commanded values are displayed as dashed lines, while actual sensor data is represented as a solid line)	89
3.17	Ground vehicle and landing platform model as used in CFD simulations	89
3.18	Visualization of disturbed flow around the ground vehicle and optimal landing trajectory	91
3.19	Longitudinal (x) and lateral (y) distribution of induced angle of attack (black) and relative velocity (red) at 1m above landing platform	91
3.20	Graphical User Interface (GUI) for human driver commands	92
3.21	Ground vehicle desired vs. actual velocity during a virtual landing experiment (real ground vehicle, virtual aircraft)	92
3.22	Aircraft dynamic modes	95
4.1	Typical lag profile during flight tests	98
4.2	Reference marker pose estimate and transformation to desired reference point at GPS antenna mounting position	100
4.3	Vision-based relative state estimates compared to RTK-GPS measurements	101
4.4	Continuous optical marker tracking until touchdown	102
4.5	High-level longitudinal cooperative control strategy	105
4.6	Root locus plots of different safety-based position synchronization controller feedback structures	108
4.7	Safety-based longitudinal position synchronization controller	109
4.8	Safety-based lateral position synchronization controller with optional runway centerline tracking term (dashed block C_{YG})	110

4.9	Final approach and flare maneuver	111
4.10	Ground tests with aircraft on top of landing platform to determine the best strategy for the ground lock mechanism	112
4.11	Waypoint pattern used for the landing maneuver	114
4.12	Geometrical conditions for state machine logic in the longitudinal direction (side view)	115
4.13	Geometrical conditions for state machine logic in the lateral direction (rear view)	116
4.14	Longitudinal spatial constraints for the optimization-based approach controller	123
4.15	Controller decoupling strategy where the horizontal optimization is executed first (opt. 1), while the vertical optimization is executed second (opt. 2)	125
4.16	Resulting minimum flight path angles in headwind conditions for fixed (red) and adaptive (green) input limits. The black vectors illustrate the wind-free case	128
4.17	Runway elevation profiles of Oberpfaffenhofen airport (EDMO) and Mindelheim-Mattsies airport (EDMN)	130
4.18	Performance-based longitudinal position synchronization controller	135
4.19	Performance-based lateral position synchronization controller with optional runway centerline tracking terms (dashed blocks C_{YG2} and C_{YA2})	135
4.20	Block diagram of a classical <i>position-force computed (P-Fc)</i> teleoperation architecture	137
4.21	Circuit representation of position-force computed (P-Fc) teleoperation architecture	139
4.22	Energy flow through the TDPN	140
4.23	P-Fc teleoperation architecture with PO-PC pairs in impedance and admittance configuration on the master and slave side respectively	143
4.24	Block diagram of P-P teleoperation architecture	145
4.25	Circuit representation of P-P teleoperation architecture with TDPN-augmented communication network	145
4.26	Circuit representation of aircraft position synchronization controller with passivity controller and augmented TDPN-based communication network representation	146
4.27	Circuit representation of ground vehicle position synchronization controller with passivity controller and augmented TDPN-based communication network representation	147
4.28	Controller modes and switching conditions during safety-based and performance-based landing maneuver	151
4.29	Frequency responses of safety-based and performance-based position synchronization controllers after loop shaping design	153
4.30	Longitudinal position synchronization controller with lag correction terms	156
4.31	Pole-zero plot for transfer function X_{rel}/V_{land} of performance-based position synchronization controller with increasing communication time delays. The system remains stable for time delays of up to $5.7s$	158
4.32	Step responses for transfer function X_{rel}/V_{land} of performance-based position synchronization controller with increasing communication time delays. The system remains stable for time delays of up to $5.7s$	158
4.33	Lateral position synchronization controller with lag correction terms	159
4.34	Pole-zero plot for transfer function Y_{rel}/Y_{rel}^{des} of lateral performance-based position synchronization controller with increasing communication time delays. The system remains stable for time delays of up to $7.7s$	160

4.35	Step responses for transfer function Y_{rel}/Y_{rel}^{des} of lateral performance-based position synchronization controller with increasing communication time delays. The system remains stable for time delays of up to $7.7s$	161
4.36	Basic feedback control loop	162
4.37	Frequency responses of safety-based and performance-based position synchronization controllers after optimization-based tuning	164
4.38	Maximum overshoot and undershoot for safety-based and performance-based position synchronization controllers after a $5m/s$ tailwind gust in longitudinal direction. The interval of confidence is 98%, with mean value μ and variance σ^2	167
4.39	Settling time performance metrics for safety-based and performance-based position synchronization controllers after gusts in longitudinal and lateral directions. The interval of confidence is 98%, with mean value μ and variance σ^2	168
4.40	ITAE performance metrics for lateral wind gusts of $5m/s$. The interval of confidence is 98%, with mean value μ and variance σ^2	169
4.41	Undershoot responses for longitudinal wind gusts without time delay and in the presence of random time delay with a mean value of $0.15s$. The interval of confidence is 98%, with mean value μ and variance σ^2	170
5.1	Representation of ground vehicle induced aerodynamic disturbance in real-time simulation environment	172
5.2	Horizontal relative vehicle positions during gust event with slow ground vehicle actuator . The gust disturbance magnitude is increased until the system reaches the retry limits of $0.8m$ in x and y direction. The legend shows the estimated disturbance rejection performance bounds. The <i>slow</i> ground vehicle actuator (human driver) is modeled as a first-order low-pass filter with a cut-off frequency of $2.5rad/s$	175
5.3	Horizontal relative vehicle positions during gust event with fast ground vehicle actuator . The gust disturbance magnitude is increased until the system reaches the retry limits of $0.8m$ in x and y direction. The legend shows the estimated disturbance rejection performance bounds. The <i>fast</i> ground vehicle actuator (autonomous ground vehicle) is modeled as a first-order low-pass filter with a cut-off frequency of $15.7rad/s$	177
5.4	Random system inputs and parameters as used in Monte Carlo simulation of wind gust rejection robustness analysis. These distributions correspond to the safety-based simulation for the slow ground vehicle actuation case, but are virtually identical for all other test cases. Further illustrations are thus omitted for brevity	178
5.5	Results of Monte Carlo simulation for wind gust rejection robustness analysis for slow ground vehicle actuator . Different time domain performance metrics are shown for the safety-based and performance-based landing controllers. The analysis is performed in the horizontal plane	180
5.6	Results of Monte Carlo simulation for wind gust rejection robustness analysis for fast ground vehicle actuator . Different time domain performance metrics are shown for the safety-based and performance-based landing controllers. The analysis is performed in the horizontal plane	182

5.7	Nominal landing performance for safety-based and performance-based landing systems (side view)	184
5.8	Fail-safe behavior of the safety-based and performance-based landing systems for an excessive tailwind gust during the approach phase. A tailwind of $5m/s$ hits the system $5s$ after maneuver start, which corresponds to a relative altitude of around $15m$ (side view)	186
5.9	Fail-safe behavior of the safety-based and performance-based landing systems for an excessive crosswind gust during the flare phase. A crosswind (right) of $15m/s$ gust hits the system at a relative altitude of $4m$ (side view)	188
5.10	Fail-safe behavior of safety-based and performance-based landing systems for excessive crosswind gust during flare phase. A crosswind (right) of $15m/s$ gust hits the system at a relative altitude of $4m$ (rear view)	189
5.11	Fail-safe behavior of the performance-based landing system for excessive communication time delay. A round trip delay of $4s$ was applied and a crosswind gust was used to perturb the previously aligned vehicles	190
5.12	Random system inputs and parameters as used in Monte Carlo simulation for landing robustness analysis. These distributions correspond to the safety-based simulation for the slow ground vehicle actuation case, but are virtually identical for all other test cases. Further illustrations are thus omitted for brevity	191
5.13	Results of Monte Carlo simulation for robust landing performance comparison for slow ground vehicle actuator . Different time domain performance metrics are shown for the safety-based and performance-based landing controllers	193
5.14	Results of Monte Carlo simulation for robust landing performance comparison for fast ground vehicle actuator . Different time domain performance metrics are shown for the safety-based and performance-based landing controllers	195
6.1	Landing platform mounted on top of ground vehicle	198
6.2	Landing platform validation tests performed at velocities of up to $120km/s$	198
6.3	Flow visualization with wool threads	199
6.4	Flight experiments with small manually controlled aircraft inside ground vehicle induced disturbance effect	199
6.5	Drop testing of ground lock mechanism	200
6.6	Optical marker as seen from the aircraft	201
6.7	Continuous tracking of optimal markers	201
6.8	Relative vehicle position and velocity for a virtual landing maneuver with the safety-based landing controller. The overall maneuver is shown at the top, while a detail around touchdown is shown at the bottom	203
6.9	Experimental setup for final landing experiments	204
6.10	Relative vehicle positions and velocities in local runway frame during exemplary landing attempts with safety-based landing controller. Touchdown occurs at $t = 50s$	206
6.11	Ground vehicle desired vs. actual velocity for the selected landing attempts	207
6.12	Successful touchdown on top of the landing platform	208
6.13	Comparison of relative altitude estimates based on 1. GPS data exchanged via the network (red), 2. pure optical marker tracking (green), and 3. combined solution by sensor data fusion (blue). Around $t = 13s$ the touchdown is performed	209

List of Tables

2.1	Landing system high-level functional requirements	32
2.2	High-level controller functions classified by maneuver phase and vehicle - [c] indicates a cooperative function, [nc] indicated a non-cooperative function, and [a] indicates an assistive function	41
2.3	High-level controller functions classified by level of cooperation and vehicle	42
2.4	Aircraft flight control system functional requirements	43
2.5	Ground vehicle control system functional requirements	44
2.6	Relative estimation system functional requirements	45
2.7	Approach control system functional requirements	46
2.8	Low-level functions for final safety-based approach classified by operating direction and vehicle	48
2.9	Summary of design choices for the approach controller	49
2.10	Landing control system functional requirements	50
2.11	Possible modes of cooperation for the position regulation phase of the landing maneuver: no cooperation (0), cooperation in x-direction (X), cooperation in y-direction (Y), cooperation in x and y-direction (XY). The x-axis points in runway direction, the y-axis to the right	52
2.12	Possible modes of cooperation for the position regulation phase of the landing maneuver classified into unilateral (*) and bilateral (-) modes: no cooperation (0), cooperation in x-direction (X), cooperation in y-direction (Y), cooperation in x and y-direction (XY). The x-axis points in runway direction, the y-axis to the right	54
2.13	Possible unilateral modes of cooperation for the safety-based landing controller: no cooperation (0), cooperation in x-direction (X), cooperation in y-direction (Y), cooperation in x and y-direction (XY). The x-axis points in runway direction, the y-axis to the right	54
2.14	Possible bilateral modes of cooperation for the performance-based landing controller: no cooperation (0), cooperation in x-direction (X), cooperation in y-direction (Y), cooperation in x and y-direction (XY). The x-axis points in runway direction, the y-axis to the right	56

2.15	Low-level functions for safety-based landing classified by operating direction and vehicle	58
2.16	Low-level functions for performance-based landing classified by operating direction and vehicle	58
2.17	Summary of design choices for the landing controller	59
3.1	Considered value ranges for Penguin BE aerodynamic parameters used for geometry-based modeling. These include aerodynamic angles, angular rates and control inputs	66
3.2	Root-mean-square deviations (RMSD) and normalized root-mean-square deviations (NRMSD) of polynomial approximations after 6 iterations	68
3.3	Final set of monomials used in the polynomial approximations of the different aerodynamic coefficients	69
3.4	Identified stability and control derivatives for Penguin BE linear aerodynamics model as defined in Equation (3.22)	75
3.5	Quantitative statistics for model validation. Averaged R^2 , RMSE and NRMSE for force and moment predictions of the linear model are depicted. The reported values confirm the validity of the aerodynamic model	75
3.6	TIC and GOF metrics for forward simulation with linear aircraft model. Both the results from identification and validation data sets are computed. Lower values indicate a better fit for TIC, while higher values are better for GOF	77
3.7	Demonstrator aircraft <i>Penguin BE UAV</i> flight controller gains after optimization-based tuning	83
3.8	Technical specifications for the demonstrator ground vehicle Audi A6 Avant 2.5 TDI (2000)	90
3.9	Demonstrator ground vehicle <i>Audi A6</i> controller gains after optimization-based tuning	93
4.1	Optimized position synchronization controller gains for safety-based architecture	154
4.2	Optimized position synchronization controller gains for performance-based architecture	155
4.3	Stability margins for the longitudinal position synchronization controllers before and after the optimization-based tuning procedure	163
A.1	Aircraft controller gains	215
A.2	Ground vehicle controller gains	216
A.3	Safety-based position synchronization controller gains (fast ground vehicle actuator)	216
A.4	Performance-based position synchronization controller gains (fast ground vehicle actuator)	216
A.5	Safety-based position synchronization controller gains (slow ground vehicle actuator)	217
A.6	Performance-based position synchronization controller gains (slow ground vehicle actuator)	217

Bibliography

- [1] Airbus SAS (2016). Future by airbus. smart tech and interaction zones. <http://www.airbus.com/innovation/future-by-airbus>.
- [2] Airbus SE (2008). Airbus flight operations briefing notes - landing techniques - crosswind landings. *FOB Reference : FLT_OPS - LAND - SEQ 05 - REV 03 - MAR. 2008*.
- [3] Artigas, J. (2014). *Time domain passivity control for delayed teleoperation*. PhD thesis, Ph. D. dissertation, Universidad Politecnica Madrid-UPM.
- [4] Artigas, J., Ryu, J.-H., and Preusche, C. (2010a). Position drift compensation in time domain passivity based teleoperation. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4250–4256. IEEE.
- [5] Artigas, J., Ryu, J.-H., and Preusche, C. (2010b). Time domain passivity control for position-position teleoperation architectures. *Presence: Teleoperators and Virtual Environments*, 19(5):482–497.
- [6] Artigas, J., Ryu, J.-H., Preusche, C., and Hirzinger, G. (2011). Network representation and passivity of delayed teleoperation systems. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 177–183. IEEE.
- [7] Aström, K. J. and Murray, R. M. (2010). *Feedback systems: an introduction for scientists and engineers*. Princeton university press.
- [8] Balachandran, R., Artigas, J., Mehmood, U., and Ryu, J.-H. (2016). Performance comparison of wave variable transformation and time domain passivity approaches for time-delayed teleoperation: Preliminary results. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 410–417. IEEE.
- [9] Balmer, G., Muskardin, T., Wlach, S., and Kondak, K. (2018). Enhancing model-free wind estimation for fixed-wing uav. In *2018 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 1242–1247. IEEE.

- [10] Beard, R. W. and McLain, T. W. (2012). *Small unmanned aircraft: Theory and practice*. Princeton university press.
- [11] Borowczyk, A., Nguyen, D.-T., Phu-Van Nguyen, A., Nguyen, D. Q., Saussié, D., and Le Ny, J. (2017). Autonomous landing of a multirotor micro air vehicle on a high velocity ground vehicle. *IFAC-PapersOnLine*, 50(1):10488–10494.
- [12] Brandt, J. and Selig, M. (2011). Propeller performance data at low reynolds numbers. In *49th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, page 1255.
- [13] Brockhaus, R. (2013). *Flugregelung*. Springer-Verlag.
- [14] Burgess, M. (2017). Google quietly shut its internet drone project last year. <https://www.wired.co.uk/article/google-project-titan-web-drones>.
- [15] Chalk, C. R., Neal, T., Harris, T., Pritchard, F. E., and Woodcock, R. J. (1969). Background information and user guide for mil-f-8785b (asg), 'military specification-flying qualities of piloted airplanes'. Technical report, CORNELL AERONAUTICAL LAB INC BUFFALO NY.
- [16] Chandler, P. R., Pachter, M., and Rasmussen, S. (2001). Uav cooperative control. In *Proceedings of the 2001 American Control Conference*.(Cat. No. 01CH37148), volume 1, pages 50–55. IEEE.
- [17] Charnley, W. (1958). A survey of approach and landing aids. *Proceedings of the IEE-Part B: Radio and Electronic Engineering*, 105(9S):333–343.
- [18] Chawda, V., Van Quang, H., O'Malley, M. K., and Ryu, J.-H. (2014). Compensating position drift in time domain passivity approach based teleoperation. In *2014 IEEE Haptics Symposium (HAPTICS)*, pages 195–202. IEEE.
- [19] Chen, B., Dong, X., Xu, Y., and Lin, Q. (2007). Disturbance analysis and flight control law design for aerial refueling. In *2007 International Conference on Mechatronics and Automation*, pages 2997–3002. IEEE.
- [20] Clos-Cot, L., Templai, J., Vanek, B., and Patartics, B. (2018). Vision-based control and guidance structure for automatic landing simulations. In *2018 IEEE Aerospace Conference*, pages 1–15. IEEE.
- [21] Coelho, A., Singh, H., Muskardin, T., Balachandran, R., and Kondak, K. (2018). Smoother position-drift compensation for time domain passivity approach based teleoperation. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5525–5532. IEEE.
- [22] Condon, J. (2017). Darpa's new project sidearm snatches drones from the sky. <https://www.thedrive.com/tech/7445/darpas-new-project-sidearm-snatches-drones-from-the-sky>.

- [23] Cook, M. V. (2012). *Flight dynamics principles: a linear systems approach to aircraft stability and control*. Butterworth-Heinemann.
- [24] Cortés, J., Martínez, S., and Bullo, F. (2006). Robust rendezvous for mobile autonomous agents via proximity graphs in arbitrary dimensions. *IEEE Transactions on Automatic Control*, 51(8):1289–1298.
- [25] DARPA (2017). Sidearm prototype catches full-size unmanned aerial system flying at full speed. <https://www.darpa.mil/news-events/2017-02-06>.
- [26] Deperrois, A. (2013). Analysis of foils and wings operating at low reynolds numbers. *Guidelines for XFLR5*.
- [27] Dogan, A., Sato, S., and Blake, W. (2005). Flight control and simulation for aerial refueling. In *AIAA Guidance, Navigation, and Control Conference and Exhibit*, page 6264.
- [28] d’Oliveira, F. A., Melo, F. C. L. d., and Devezas, T. C. (2016). High-altitude platforms—present situation and technology trends. *Journal of Aerospace Technology and Management*, 8(3):249–262.
- [29] Draper, N. R. and Smith, H. (1998). *Applied regression analysis*, volume 326. John Wiley & Sons.
- [30] Drela, M. and Youngren, H. (2000). Xfoil. <http://raphael.mit.edu/xfoil>.
- [31] Drela, M. and Youngren, H. (2004). Athena vortex lattice (avl). <http://raphael.mit.edu/avl>.
- [32] Droge, G. (2015). Distributed virtual leader moving formation control using behavior-based mpc. In *2015 American Control Conference (ACC)*, pages 2323–2328. IEEE.
- [33] Dunbar, W. B. and Murray, R. M. (2006). Distributed receding horizon control for multi-vehicle formation stabilization. *Automatica*, 42(4):549–558.
- [34] EC-Safemobil Project Consortium. Deliverable d2.1 system requirements.
- [35] Elektra Solar GmbH (2019). Solar-powered unmanned and optionally piloted aircraft system.
- [36] Etele, J. and Fusina, G. (2006). *Overview of wind gust modelling with application to autonomous low-level UAV control*. Defence R&D Canada.
- [37] Falanga, D., Zanchettin, A., Simovic, A., Delmerico, J., and Scaramuzza, D. (2017). Vision-based autonomous quadrotor landing on a moving platform. In *2017 IEEE International Symposium on Safety, Security and Rescue Robotics (SSRR)*, pages 200–207. IEEE.
- [38] Falkena, W., Borst, C., Chu, Q., and Mulder, J. (2011). Investigation of practical flight envelope protection systems for small aircraft. *Journal of Guidance, Control, and Dynamics*, 34(4):976–988.

- [39] Fax, J. A. and Murray, R. M. (2004). Information flow and cooperative control of vehicle formations. *IEEE transactions on automatic control*, 49(9):1465–1476.
- [40] Federal Aviation Administration (1991). *Airplane simulator qualification (AC 120-40C)*. FAA Advisory Circular, Tech. Rep.
- [41] Federal Aviation Administration (2011). *Airplane Flying Handbook (FAA-H-8083-3A)*. Skyhorse Publishing Inc.
- [42] Federal Aviation Administration (2016). *Airplane Flying Handbook (FAA-H-8083-3B)*. Skyhorse Publishing Inc.
- [43] Feng, Y., Zhang, C., Baek, S., Rawashdeh, S., and Mohammadi, A. (2018). Autonomous landing of a uav on a moving platform using model predictive control. *Drones*, 2(4):34.
- [44] Fezans, N. and Jann, T. (2017). Modeling and simulation for the automation of aerial refueling of military transport aircraft with the probe-and-drogue system. In *AIAA Modeling and Simulation Technologies Conference*, page 4008.
- [45] Flight Safety Foundation (2000). Approach and landing accident reduction - briefing note 8.7 - crosswind landings. *Flight Safety Digest*.
- [46] Garcia, M., Muskardin, T., Viguria, A., Laiacker, M., Ollero, A., and Kondak, K. (2013). Analysis and development of a reliable fixed wing uav control system for mission profiles with restricted gps availability. In *2013 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 599–608. IEEE.
- [47] Gautam, A., Sujit, P., and Saripalli, S. (2014). A survey of autonomous landing techniques for uavs. In *2014 international conference on unmanned aircraft systems (ICUAS)*, pages 1210–1218. IEEE.
- [48] Gautam, A., Sujit, P., and Saripalli, S. (2015). Application of guidance laws to quadrotor landing. In *2015 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 372–379. IEEE.
- [49] Girard, A. R. and Kabamba, P. T. (2015). Proportional navigation: optimal homing and optimal evasion. *SIAM Review*, 57(4):611–624.
- [50] Gong, X., Abbott, L., and Fleming, G. (2006). A survey of techniques for detection and tracking of airport runways. In *44th AIAA Aerospace Sciences Meeting and Exhibit*, page 1436.
- [51] Gripp, J. D. B. and Sampaio, U. (2014). Automatic landing of a uav using model predictive control for the surveillance of internal autopilot’s controls. In *International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 1219–1224.
- [52] Griva, I., Nash, S. G., and Sofer, A. (2009). *Linear and nonlinear optimization*, volume 108. Siam.

- [53] Hannaford, B. and Ryu, J.-H. (2002). Time-domain passivity control of haptic interfaces. *IEEE Transactions on Robotics and Automation*, 18(1):1–10.
- [54] Harold, K. and Randal, A. (2011). *Simulation of dynamical system with MATLAB and Simulink*. CRC Press.
- [55] Hirzinger, G. (2012). Start- und Landehilfe für Leichtbauflugzeuge (DE102010051112A1). *German Patent*.
- [56] International Air Transport Association (2013). IATA technology roadmap. *June*.
- [57] Jategaonkar, R., Behr, R., Gockel, W., and Zorn, C. (2006). Data analysis of phoenix reusable launch vehicle demonstrator flight test. *Journal of aircraft*, 43(6):1732–1737.
- [58] Jategaonkar, R. V. (2015). *Flight vehicle system identification: A time-domain methodology*. American Institute of Aeronautics and Astronautics, Inc.
- [59] Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. *Journal of basic Engineering*, 82(1):35–45.
- [60] Kastner, N. and Looye, G. (2013). Generic tecs based autopilot for an electric high altitude solar powered aircraft. *CEAS EuroGNC*.
- [61] Kim, H. J., Kim, M., Lim, H., Park, C., Yoon, S., Lee, D., Choi, H., Oh, G., Park, J., and Kim, Y. (2013). Fully autonomous vision-based net-recovery landing system for a fixed-wing uav. *IEEE/ASME Transactions On Mechatronics*, 18(4):1320–1333.
- [62] Kinzig, B. et al. (2009). Global hawk systems engineering case study.
- [63] Klausen, K., Fossen, T. I., and Johansen, T. A. (2018). Autonomous recovery of a fixed-wing uav using a net suspended by two multirotor uavs. *Journal of Field Robotics*, 35(5):717–731.
- [64] Klöckner, A. (2013). Geometry based flight dynamics modelling of unmanned airplanes. In *AIAA Modeling and Simulation Technologies (MST) Conference*, page 5154.
- [65] Klöckner, A., Leitner, M., Schlabe, D., and Looye, G. (2013). Integrated modelling of an unmanned high-altitude solar-powered aircraft for control law design analysis. In *Advances in Aerospace Guidance, Navigation and Control*, pages 535–548. Springer.
- [66] Klöckner, A., Schlabe, D., and Looye, G. (2012). Integrated simulation models for high-altitude solar-powered aircraft. In *AIAA modeling and simulation technologies conference*, page 4717.
- [67] Kong, J., Pfeiffer, M., Schildbach, G., and Borrelli, F. (2015). Kinematic and dynamic vehicle models for autonomous driving control design. In *Intelligent Vehicles Symposium (IV), 2015 IEEE*, pages 1094–1099. IEEE.

- [68] Koo, S., Kim, S., Suk, J., Kim, Y., and Shin, J. (2018). Improvement of shipboard landing performance of fixed-wing uav using model predictive control. *International Journal of Control, Automation and Systems*, 16(6):2697–2708.
- [69] Kügler, M. E. and Holzapfel, F. (2017). Autoland for a novel uav as a state-machine-based extension to a modular automatic flight guidance and control system. In *2017 American Control Conference (ACC)*, pages 2231–2236. IEEE.
- [70] Kügler, M. E., Mumm, N. C., Holzapfel, F., Schwithal, A., and Angermann, M. (2019). Vision-augmented automatic landing of a general aviation fly-by-wire demonstrator. In *AIAA Scitech 2019 Forum*, page 1641.
- [71] Kwakernaak, H. (2002). Mixed sensitivity design. *IFAC Proceedings Volumes*, 35(1):61–66.
- [72] Lagarias, J. C., Reeds, J. A., Wright, M. H., and Wright, P. E. (1998). Convergence properties of the nelder–mead simplex method in low dimensions. *SIAM Journal on optimization*, 9(1):112–147.
- [73] Laiacker, M., Kondak, K., Schwarzbach, M., and Muskardin, T. (2013). Vision aided automatic landing system for fixed wing uav. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2971–2976. IEEE.
- [74] Lambregts, A. (1983). Vertical flight path and speed control autopilot design using total energy principles. In *Guidance and Control Conference*, page 2239.
- [75] Lambregts, A. and Creedon, J. (1980). Development and flight evaluation of automatic flare laws with improved touchdown dispersion. In *Guidance and Control Conference*, page 1757.
- [76] Lambregts, A. A. (2006). Generalized automatic and augmented manual flight control. In *Berlin Technical University Colloquium*, volume 19.
- [77] Langelaan, J. W., Alley, N., and Neidhoefer, J. (2011). Wind field estimation for small unmanned aerial vehicles. *Journal of Guidance, Control, and Dynamics*, 34(4):1016–1030.
- [78] Lavretsky, E. (2002). F/a-18 autonomous formation flight control system design. In *AIAA Guidance, Navigation, and Control Conference and Exhibit*, page 4757.
- [79] Lawrence, D. A. (1993). Stability and transparency in bilateral teleoperation. *IEEE transactions on robotics and automation*, 9(5):624–637.
- [80] Lee, D., Ryan, T., and Kim, H. J. (2012). Autonomous landing of a vtol uav on a moving platform using image-based visual servoing. In *2012 IEEE international conference on robotics and automation*, pages 971–976. IEEE.
- [81] Lee, J., Muskardin, T., Pacz, C. R., Oettershagen, P., Stastny, T., Sa, I., Siegwart, R., and Kondak, K. (2018). Towards autonomous stratospheric flight: A generic

- global system identification framework for fixed-wing platforms. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6233–6240. IEEE.
- [82] Leonard, N. E. and Fiorelli, E. (2001). Virtual leaders, artificial potentials and coordinated control of groups. In *Proceedings of the 40th IEEE Conference on Decision and Control (Cat. No. 01CH37228)*, volume 3, pages 2968–2973. IEEE.
- [83] Li, Y., Ang, K. H., and Chong, G. C. (2006). Pid control system analysis and design. *IEEE Control Systems Magazine*, 26(1):32–41.
- [84] Lin, H. and Antsaklis, P. J. (2005). Stability and stabilizability of switched linear systems: A short survey of recent results. In *Proceedings of the 2005 IEEE International Symposium on, Mediterrean Conference on Control and Automation Intelligent Control, 2005.*, pages 24–29. IEEE.
- [85] Lin, J., Morse, A. S., and Anderson, B. D. (2003). The multi-agent rendezvous problem. In *42nd IEEE International Conference on Decision and Control (IEEE Cat. No. 03CH37475)*, volume 2, pages 1508–1513. IEEE.
- [86] Lütjens, K., Lau, A., Pfeiffer, T., Loth, S., Gollnick, V., Klimek, H., Greßmann, B., Löwa, S., and Blank, C. (2012). *Airport2030-Lösungen für den effizienten Lufttransport der Zukunft*. Deutsche Gesellschaft für Luft-und Raumfahrt-Lilienthal-Oberth eV.
- [87] Lynch, N. A. (1996). *Distributed algorithms*. Elsevier.
- [88] Maciejowski, J. M. (2002). *Predictive control: with constraints*. Pearson education.
- [89] Mulder, J., Chu, Q., Sridhar, J., Breeman, J., and Laban, M. (1999). Non-linear aircraft flight path reconstruction review and new advances. *Progress in Aerospace Sciences*, 35(7):673–726.
- [90] Muskardin, T., Balmer, G., Persson, L., Wlach, S., Laiacker, M., Ollero, A., and Kondak, K. (2016). A novel landing system to increase payload capacity and operational availability of high altitude long endurance uav. In *2016 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 495–504.
- [91] Muskardin, T., Balmer, G., Persson, L., Wlach, S., Laiacker, M., Ollero, A., and Kondak, K. (2017). A novel landing system to increase payload capacity and operational availability of high altitude long endurance uavs. *Journal of Intelligent & Robotic Systems*, 88(2-4):597–618.
- [92] Muskardin, T., Balmer, G., Wlach, S., Kondak, K., Laiacker, M., and Ollero, A. (2016). Landing of a fixed-wing uav on a mobile ground vehicle. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1237–1242. IEEE.
- [93] Nesline, F. W. and Zarchan, P. (1981). A new look at classical vs modern homing missile guidance. *Journal of Guidance and Control*, 4(1):78–85.

- [94] Niemeyer, G. D. (1996). *Using wave variables in time delayed force reflecting teleoperation*. PhD thesis, Massachusetts Institute of Technology.
- [95] O'Brien, M. (2015). Google, facebook in race to build high-altitude aircraft.
- [96] Persson, L., Muskardin, T., and Wahlberg, B. (2017). Cooperative rendezvous of ground vehicle and aerial vehicle using model predictive control. In *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, pages 2819–2824. IEEE.
- [97] Realise Project Consortium (2019). Runway independent automatic launch and landing system. <https://www.realise.aero>.
- [98] Ren, W., Beard, R. W., and Atkins, E. M. (2005). A survey of consensus problems in multi-agent coordination. In *Proceedings of the 2005, American Control Conference, 2005.*, pages 1859–1864. IEEE.
- [99] Rencher, A. C. (2003). *Methods of Multivariate Analysis, 2nd Edition*. Wiley, wiley series in probability and statistics edition.
- [100] Rodriguez-Angeles, A. and Nijmeijer, H. (2003). Cooperative synchronization of robots via estimated state feedback. In *42nd IEEE International Conference on Decision and Control (IEEE Cat. No. 03CH37475)*, volume 2, pages 1514–1519. IEEE.
- [101] Rodriguez-Ramos, A., Sampedro, C., Bavle, H., Moreno, I. G., and Campoy, P. (2018). A deep reinforcement learning technique for vision-based autonomous multi-rotor landing on a moving platform. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1010–1017. IEEE.
- [102] Rohacs, D., Voskuijl, M., and Siepenkotter, N. (2014). Evaluation of landing characteristics achieved by simulations and flight tests on a small-scaled model related to magnetically levitated advanced take-off and landing operations. In *ICAS 2014: Proceedings of the 29th Congress of the International Council of the Aeronautical Sciences, St. Petersburg, Russia, 7-12 September 2014*. Citeseer.
- [103] Rohacs, J. and Rohacs, D. (2018). Problems and barriers impeding the implementation of maglev assisted aircraft take-off and landing concept. *Journal of Transportation Technologies*, 8(02):91.
- [104] Ross, S. M., Pachter, M., Jacques, D. R., Kish, B. A., and Millman, D. R. (2006). Autonomous aerial refueling based on the tanker reference frame. In *2006 IEEE Aerospace Conference*, pages 22–pp. IEEE.
- [105] Ryu, J.-H., Artigas, J., and Preusche, C. (2010). A passive bilateral control scheme for a teleoperator with time-varying communication delay. *Mechatronics*, 20(7):812–823.
- [106] Saripalli, S., Montgomery, J. F., and Sukhatme, G. S. (2003). Visually guided landing of an unmanned aerial vehicle. *IEEE transactions on robotics and automation*, 19(3):371–380.

- [107] Shue, S.-P. and Agarwal, R. K. (1999). Design of automatic landing systems using mixed h/h control. *Journal of Guidance, Control, and Dynamics*, 22(1):103–114.
- [108] Sipahi, R., Niculescu, S.-I., Abdallah, C. T., Michiels, W., and Gu, K. (2011). Stability and stabilization of systems with time delay. *IEEE Control Systems Magazine*, 31(1):38–65.
- [109] Skogestad, S. and Postlethwaite, I. (2007). *Multivariable feedback control: analysis and design*, volume 2. Wiley New York.
- [110] Stevens, B. L., Lewis, F. L., and Johnson, E. N. (2015). *Aircraft control and simulation: dynamics, controls design, and autonomous systems*. John Wiley & Sons.
- [111] Tan, R. and Kumar, M. (2014). Tracking of ground mobile targets by quadrotor unmanned aerial vehicles. *Unmanned Systems*, 2(02):157–173.
- [112] Tanaka, H., Sumi, Y., and Matsumoto, Y. (2012). A high-accuracy visual marker based on a microlens array. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4192–4197. IEEE.
- [113] The Boeing Company (2005). 737 - 600/700/800/900 Flight Crew Training Manual. (FCT 737 NG (TM)).
- [114] The Boeing Company (2006). *B-52H Flight manual, T-O. 1B-52H-1*, change 22 edition.
- [115] The National Transportation Safety Board (2015). Accident report, aircraft: Titan aerospace holdings inc solara 50, registration: N950ta. https://www.nts.gov/_layouts/ntsb.aviation/brief.aspx?ev_id=20150505X85410&key=1.
- [116] The National Transportation Safety Board (2016). Accident report, aircraft: Facebook uk ltd aquila, registration: N565aq, ntsb identification: Dca16ca197. https://www.nts.gov/_layouts/ntsb.aviation/brief.aspx?ev_id=20160701X62525&key=1.
- [117] Tzoumanikas, D., Li, W., Grimm, M., Zhang, K., Kovac, M., and Leutenegger, S. (2019). Fully autonomous micro air vehicle flight and landing on a moving target using visual-inertial estimation and model-predictive control. *Journal of Field Robotics*, 36(1):49–77.
- [118] UAV Factory (2019). Penguin c uas.
- [119] Union of Concerned Scientists (2017). Ucs satellite database. <https://www.ucsusa.org/nuclear-weapons/space-weapons/satellite-database>.
- [120] United Nations Office for Outer Space Affairs (2018). Online index of objects launched into outer space.
- [121] Van Loan, C. F. and Golub, G. H. (1983). *Matrix computations*. Johns Hopkins University Press.

- [122] Vechtel, D. (2019). How future aircraft can benefit from a steerable main landing gear for crosswind operations. *CEAS Aeronautical Journal*.
- [123] Vechtel, D., Meissner, U. M., and Hahn, K.-U. (2014). On the use of a steerable main landing gear for crosswind landing assistance. *CEAS Aeronautical Journal*, 5(3):293–303.
- [124] Wagner, D. and Schmalstieg, D. (2007). Artoolkitplus for pose tracking on mobile devices.
- [125] Wang, L., Zhang, Z., Zhu, Q., and Dong, R. (2019). Longitudinal automatic carrier landing system guidance law using model predictive control with an additional landing risk term. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 233(3):1089–1105.
- [126] Wang, R., Zhou, Z., and Shen, Y. (2007). Flying-wing uav landing control and simulation based on mixed h₂/h_∞. In *2007 International Conference on Mechatronics and Automation*, pages 1523–1528. IEEE.
- [127] Weiser, C. (2017). Robust flight control design for the faser uav. Master's thesis, Technische Universität München.
- [128] Yang, S., Ying, J., Lu, Y., and Li, Z. (2015). Precise quadrotor autonomous landing with sruf vision perception. In *2015 IEEE international conference on robotics and automation (ICRA)*, pages 2196–2201. IEEE.
- [129] Ye, Y., Pan, Y.-J., and Hilliard, T. (2013). Bilateral teleoperation with time-varying delay: A communication channel passification approach. *IEEE/ASME Transactions on Mechatronics*, 18(4):1431–1434.
- [130] Yi, Z. and Heping, W. (2006). A study of structure weight estimating for high altitude long endurance (hale) unmanned aerial vehicle (uav). *ICAS-Secretariat-25th Congr. Int. Counc. Aeronaut. Sci*, 3:1784–1789.
- [131] Yu, J., LaValle, S. M., and Liberzon, D. (2011). Rendezvous without coordinates. *IEEE Transactions on Automatic Control*, 57(2):421–434.