Cranfield University

School of Aerospace, Transport and Manufacturing

MSc Autonomous Vehicle Dynamics & Control

MSc Thesis

COUNTERING IMPROVISED CRUISE MISSILES

Maria Morante Soria

Academic Year 2015 – 2016

Supervisor: Dr Hyo – Sang SHIN

December, 2016

Cranfield University

School of Aerospace, Transport and Manufacturing

MSc Autonomous Vehicle Dynamics & Control

MSc Thesis

Academic Year 2015 – 2016

COUNTERING IMPROVISED CRUISE MISSILES

Maria Morante Soria

Supervisor: Dr Hyo – Sang SHIN

December, 2016

This thesis is submitted in partial fulfilment of the requirements for the degree of MSc in Autonomous Vehicle Dynamics & Control.

Abstract

Two types of guidance and control designs are compared under a linear quadratic formulation: a separated approach and an integrated one. Analysing the integrated approach, two types of autopilot – guidance laws are explored: a single – loop and a two – loop configuration. In both cases, there is a full feedback on the guidance loop. However, the two – loop scheme presents an inner autopilot loop designed separately from the outer guidance loop. It is shown that both integrated schemes achieve identical end – game scenario performance if and only if the number of guidance commands is identical to the number of controllers. Hence, the integrated two – loop configuration is preferred since it ensures the inner stability of the airframe, whether the guidance loop is active or not. It is also proven that the performance of the integrated designs is higher than the one of the traditional separated approach. The guidance laws are illustrated using an exo – atmospheric dual control missile model.

Acknowledgements

First of all, I would like to thank my supervisor, Dr Hyo–Sang Shin, and MBDA UK for their continuous support and advice all along these months. I would also like to express my gratitude to the different professors I have met in the Polytechnic University of Valencia. Not to mention all the friends I have made there and with whom I have lived unforgettable experiences. I would also like to thank all the wonderful friends I have made in Cranfield. I thoroughly enjoyed each and every one of the moments together. A special thank you goes to Ben, without whom this experience would not have been the same. Finally, I would like to express my deepest gratitude to my parents, whose pride makes it all worthwhile, and without whom this accomplishment would not have been possible.

Contents

List of Figures

List of Tables

Abbreviations

Symbol	Description
APN	Augmented Proportional Navigation
CG	Center of Gravity
DAC	Dual Aerodynamic Control
G&C	Guidance & Control
IGA	Integrated Guidance and Autopilot
IGC	Integrated Guidance and Control
LOS	Line Of Sight
LQD	Linear Quadratic Differential
LQDG	Linear Quadratic Differential Game
LQR	Linear Quadratic Regulator
LQT	Linear Quadratic Tracking
MIMO	Multi Input Multi Output
MP	Minimum Phase
ND	Non Dimensional
NMP	Non Minimum Phase
OGL	Optimal Guidance Law
PN	Proportional Navigation
SDRE	State Dependent Riccati Equation
SISO	Single Input Single Output
SMC	Sliding Mode Control
STT	Skit To Turn
TDC	Time Delay Control
TVC	Thrust Vector Control
ZEM	Zero Effort Miss

Notation

Latin letters

Symbol	Description	Units
a_E	Evader acceleration	m/s^2
a_P	Pursuer acceleration	m/s^2
a_{EN}	Evader acceleration ⊥ to initial LOS	m/s^2
a_{PN}	Pursuer acceleration ⊥ to initial LOS	m/s^2
b	Penalty imposed by the miss distance	
c	Penalty imposed by the impact angle	
$\hat{c}_n$	Autopilot design gain – Nose control	
$\hat{c}_t$	Autopilot design gain – Tail control	
$\mathbf{F}$	State space model matrix	
$\mathbf{G}$	State space model matrix	
I	Moment of inertia	kg m^2
J	Optimal cost index	1/s
$\mathbf{K}$	Optimal gain control matrix	
$K_{\dot{\theta}_n}$	Autopilot design gain – Nose control	
$K_{\dot{\theta}_t}$	Autopilot design gain – Tail control	
$\mathbf{k}_G$	Optimal guidance gains	
l	Distance from the center of mass	m
M	Pitch moment	Nm
M_{δ_n}	Autopilot design gain – Nose control	1/s^2
M_{δ_t}	Autopilot design gain – Tail control	1/s^2
m	Mass	kg
N'	Proportional navigation constant	
$\mathbf{P}$	Ricatti equation matrix solution	
p	Roll rate	deg/s
$\mathbf{Q}_f$	State weighting matrix	
q	Pitch rate	deg/s
$\mathbf{R}$	Control weighting matrix	
r	Range	m
r	Yaw rate	deg/s
T	Thrust	N
t	Time	s

t_f	Flight time	s
t_{Go}	Time to go	s
$\mathbf{u}$	Control input vector	
u	Missile speed in OX_b axis	m/s
v	Missile speed in OY_b axis	m/s
w	Missile speed in OZ_b axis	m/s
V_c	Closing speed	m/s
V_E	Evader speed	m/s
V_P	Pursuer speed	m/s
$\mathbf{x}$	State vector	
$\mathbf{x}_G$	Guidance state vector	
$\mathbf{x}_D$	Dynamic state vector	
$\mathbf{x}_S$	Servo state vector	
y	Relative displacement normal to initial LOS	m
z	Zero effort miss distance	m

Greek letters

Symbol	**Description**	**Units**
α	Angle of attack	deg
α_T	Total angle of attack	deg
β	Sideslip angle	deg
γ	Flight path angle	deg
δ	Control surface deflection angle	deg
δ_n	Nose fin deflection angle	deg
δ_t	Tail fin deflection angle	deg
δ_n^c	Nose control fin deflection angle	deg
δ_t^c	Tail control fin deflection angle	deg
Λ_G	Effective navigation ratio	
λ	Angle between the initial and the current LOS	deg
ϕ	Aerodynamic roll angle	deg
θ	Pitch angle	deg
τ	Time constant	s
τ_n	Nose servo delay	s
τ_t	Tail servo delay	s

Superscripts

Symbol	Description
*	Optimal
APN	Augmented Proportional Navigation
B	Body axes
c	Command
Max	Maximum
OG	Optimal Guidance
PN	Proportional Navigation
W	Wind axes

Subscripts

Symbol	Description
0	Initial values
1L	Single – loop
2L	Two – loop
A	Autopilot
c	Canard
CG	Center of gravity
D	Dynamic systems
E	Evasive target
G	Guidance loop
λ	Perpendicular to LOS
N	Perpendicular to LOS
n	Nose
P	Pursuer missile
r	Radial – Along the LOS
S	Servo systems
t	Tail
Tot	Total

Chapter 1

Introduction

1.1 Background and motivation

Historically, homing missiles relied on both better manoeuvrability strategies and higher velocities over their targets in order to achieve interception. However, new missions require the interceptor to overcome different types of threats that necessarily amplify the requirements of the missile design.

Nowadays, missiles have to fly longer and faster. The design of an interceptor must be improved significantly in order to be more effective and efficient when engaging and destroying the adversary. As a result, it becomes mandatory to magnify the missile aerodynamics and exploit different types of guidance and control configurations.

To this end, the usual guidance and control designs use a decoupled architecture. This approach is generally done assuming that spectral separation holds. However, the rapid variations that occur during the end – game phase indicate that separation does not hold when it gets close to interception and this may result in instability. Thereafter, the miss distance can be increased significantly.

On the other hand, the integrated guidance and control scheme facilitates the level of synergy between the guidance and control functions. Accounting for the coupling between the subsystems, an optimal combination will significantly enhance the performance of the interceptor missiles, maximising the lethality.

Additionally, there are several considerations to be made with regard to the missile's architecture. Interceptors with aerodynamic control present either canard or tail control configurations. However, an appropriate combination of canard and tail control results in an optimum utilisation. This is due to the fact that both the overshoot and settling problems of the canard – control configuration as well as the non – minimum phase effect of the tail – control configuration are eliminated when the dual configuration is used.

This is why this thesis presents a hybrid combination of a thrust vector controller added to a nose jet device as an alternative to canard or tail control. In essence, a thrust vector controller provides additional forces and moments to the missile and quickly aligns the velocity vector. However, once the motor is entirely burned out, only the tail control remains available.

1.2 Research objectives

The purpose of this thesis is to assure the effectiveness of the integrated guidance and control design over the traditional one using a dual – controlled missile. This overall objective can be divided into five main areas of study, which are:

1. **Description of the terminal or end – game phase,** which is generally the most critical one. In essence, a missile flight consists of three phases: boost, mid – course and terminal. The complexity of each phase depends upon the mission assigned to the missile. However, during the end – game engagement, the missile is required to have a high degree of accuracy and it may be required to manoeuvre to maximum capability.

2. **Development of a dynamics model for a dual – controlled missile:** Its hybrid configuration (a nose jet device added to a thrust vector control missile) is presented as an alternative to canard or tail control.

3. **Research on the separated guidance and control approach:** An optimal guidance law is proposed in order to formulate a two – loop guidance law for a dual control missile. Furthermore, an evaluation and comparison of the performance of the separated guidance law versus the integrated laws is carried out.

4. **Research on the integrated guidance and control approach:** It accounts for the coupling between the flight control and guidance functions. The hypothesis of spectral separation between the subsystems is no longer valid. A comparative analysis of the integrated single – loop scheme versus the integrated two – loop scheme is performed. In the latter, two different autopilots schemes are explored.

5. **Autopilot design for the integrated two – loop scheme:** Modern control systems require multiple controllers and multiple terminal costs or objectives in order to meet the goal that is to achieve small miss distances.

   As a result, the autopilot will be illustrated using two different autopilot schemes:

   5.1. **Single – input autopilot design:** The guidance law issues an acceleration command, which is the input to the autopilot. An appropriate vectorial control law is used to divide the acceleration command into two equivalent signals.

   5.2. **Multi – input autopilot design:** The guidance law issues a 2D vector aiming to control each controller individually.

1.3 An overview of the thesis

This thesis is organised in seven different chapters, briefly described below, plus auxiliary material enclosed in the appendices. The structure appears as follows:

- **Chapter 1: Introduction.** Exploration of the challenges in the interceptor's design due to highly manoeuvrable threats. The thesis objectives are also described here, in addition to a brief summary of each chapter and its contributions.

- **Chapter 2: Literature review.** This chapter presents a literature review based on the research objectives. In addition, it introduces the dual – controlled missile as the future alternative after reviewing the limitations of both canard and tail control configurations.

- **Chapter 3: Problem formulation.** The design assumptions are provided in this chapter; followed by a schematic view of the end – game engagement scenario.

- **Chapter 4: Autopilot – guidance design.** This chapter provides the general form of the different guidance laws schemes studied: a separated two – loop autopilot – guidance design, an integrated single – loop guidance design and an integrated two – loop autopilot – guidance design. A theorem regarding a necessary condition to achieve the same performance for both integrated schemes is also presented.

- **Chapter 5: Case study.** Formulation of the dynamic model of a dual – controlled exo – atmospheric missile with both nose and tail controllers; followed by the end – game engagement scenario parameters and the kinematic equations. Moreover, the guidance law schemes provided in the previous chapter are applied to the DAC missile.

- **Chapter 6: Simulation results.** The separated two – loop autopilot guidance law is compared to the integrated guidance law via a Pareto front. Additionally, the integrated single – loop design is also compared to the integrated two – loop design with the help of a Pareto front and sample runs. In the latter case, the study is carried out using two different autopilot schemes that vary in function of the number of inputs.

- **Chapter 7: Conclusions and future work.** Presentation of the thesis conclusions and further recommendations.

- **References.**

- **Appendix A: Theorem proof.** Illustration of the proof of a theorem presented in Section 4.5. This theorem ensures that, under given conditions, the integrated single – loop multi – input scheme and the integrated two – loop multi – input scheme achieve identical end – game performance.

- **Appendix B: Matlab$^{®}$ code.** This chapter encompasses all the Matlab files used to formulate the different guidance law schemes in addition to several functions required to perform the analysis.

- **Appendix C: Simulink$^{®}$ model.** Presentation of the two autopilot schemes for the two – loop case along with the diagrams of the separated autopilot – guidance design, the integrated single – loop guidance design and the integrated two – loop autopilot – guidance design.

Chapter 2

Literature review

2.1 Introduction

The purpose of this chapter is to review the literature in relation to the research objectives listed in Section 1.2. Therefore, the limitations of both canard and tail control configurations are briefly described in order to introduce the dual – controlled missile as the future alternative.

On the other hand, the literature review is focused on clarifying how the dual control problem has been formulated in former studies. For that purpose, research on both the traditional guidance and control approach and the integrated one is carried out in Sections 2.3 and 2.4, respectively.

2.2 Dual control missiles

There are several considerations to be made with regard to the latest interceptor architectures. Firstly, missiles with aerodynamic control are capable of varying their trajectory by deflecting movable fins in order to generate lateral acceleration (normal to the missile longitudinal axis). These interceptors present either canard or tail control configurations: a canard control interceptor deflects its forward control surfaces to manoeuvre, whereas a tail – controlled one acts in a similar way on its rear control surfaces instead.

Canard control configurations are commonly used for manoeuvrability purposes in case of short – range air – to – air missiles; whereas tail – controlled interceptors are effective when the missiles need to make initial sharp turns since they are capable of saturating at high angles of attack. However, the latter system generates an initial small lift in the direction opposite to the main lift, which results in a response delay. This non – minimum phase effect is due to a zero in the right half s – plane of the corresponding linear transfer function and its characteristics represent a significant challenge. As a result of this, authors in [6] proved that a canard configuration provides a higher performance than a tail control one.

Table 2.1 summarises the advantages and disadvantages of both configurations.

	Advantages	**Disadvantages**
Canard	(1) Higher lift at small angles of attack. (2) Higher moment arm due to deflection of the flow in the rear surfaces. (3) Easier to manufacture. (4) Easier to modify the design. (5) More efficient design of the control and the seeker systems at the missile ogive.	(1) Saturation at low angles of attack. (2) Interaction of the vortices created by the forward control surfaces with the tail. (3) Oscillations are present in the response. (4) Long settling time required. (5) Loss of stability at higher velocities.
Tail	(1) Low aerodynamic induced drag. (2) Low induced rolling moment. (3) High effectivity at high angles of attack. (4) Easy deflection of the control surfaces used to control the roll channel.	(1) Smaller trim lift. (2) Delay in the response (non – minimum phase effect). (3) Compromise between manoeuvrability and stability. (4) Inefficient implementation of the control system.

Table 2.1: Comparison of canard and tail control configurations.

The table suggests that both configurations are complementary, and the optimum utilisation could be an appropriate combination of canard and tail control.

As a matter of fact, authors in [18] achieved promising results for a dual control missile autopilot using linear optimal control techniques. In this study, the missile was considered to operate in divert mode, where both controls are deflected in the same direction (Figure 2.1). Therefore, opposite directions of canard and tail controls were not considered.

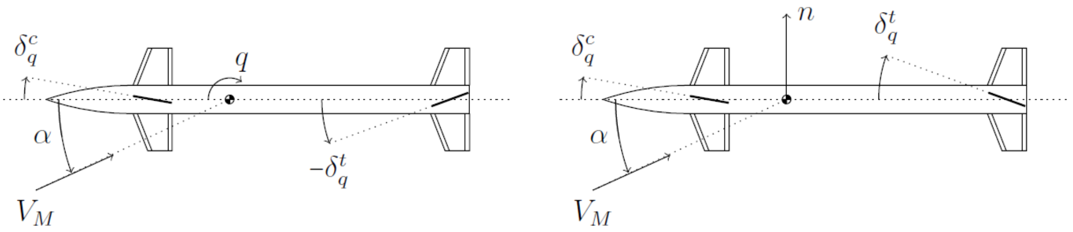


Figure 2.1: Dual control and divert mode for a missile with movable forward and rear controls (taken from [40]).

The results obtained in [18] are shown in Figures 2.2 and 2.3, where the closed – loop response of the autopilot for the dual configuration showed better performance than the canard – only or tail – only configurations. It shows that both the overshoot and settling problems of the canard – control configuration as well as the non – minimum phase effect of the tail – control configuration are eliminated.

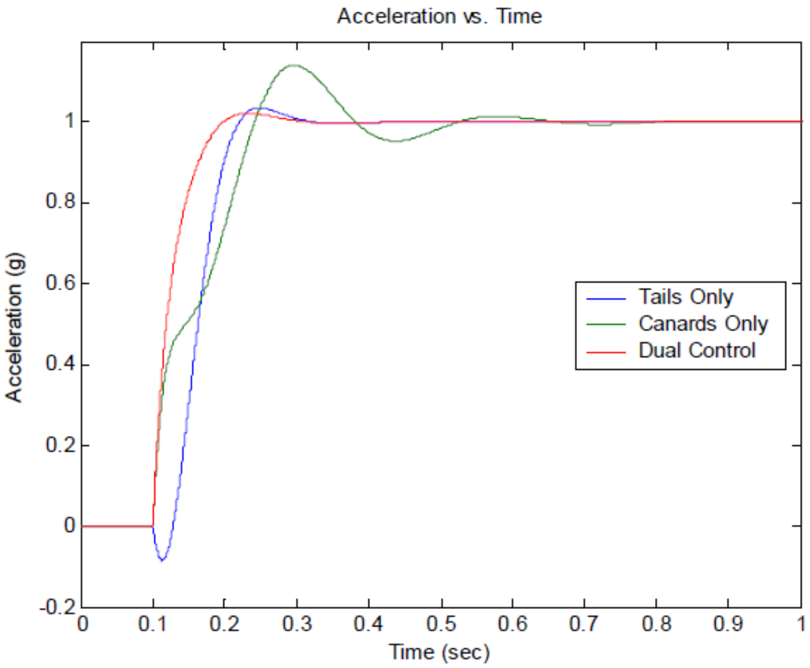


Figure 2.2: Acceleration vs. time for canard, tail, and both control configurations [18].

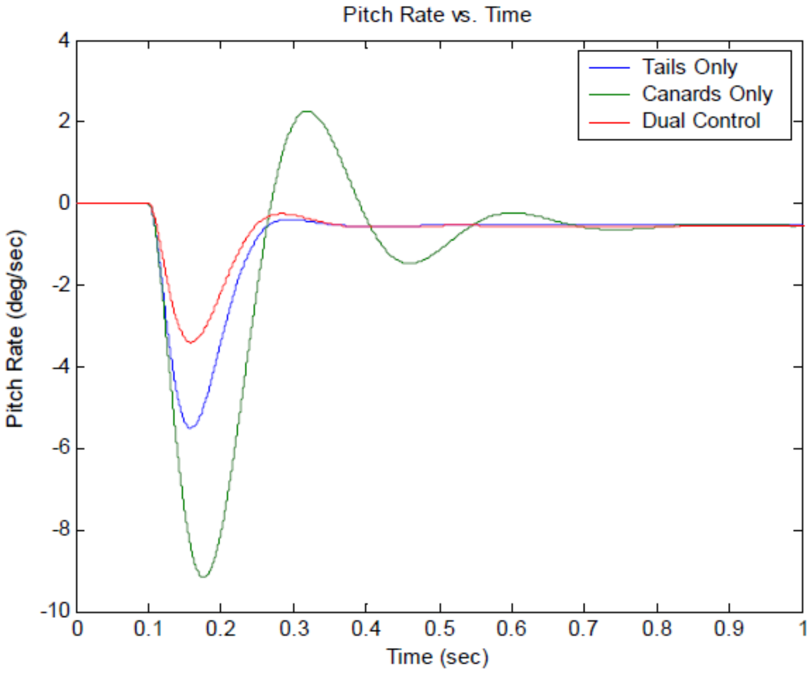


Figure 2.3: Pitch rate vs. time for canard, tail, and both control configurations [18].

Also, from Figure 2.3, it can be appreciated that the missile operates in divert mode, as already mentioned above. As a result of this, the interceptor's lift increases and rotations are minor in comparison with the more traditional approaches. However, dual control missiles operating in divert mode struggle to achieve high angles of attack or high levels of lateral acceleration. Hence, divert mode is typically used for small corrections right before the engagement.

To sum up, integrating both types of control does really enhance the performance of the missile. This is why this thesis uses a dual control missile. In particular, a hybrid combination of a thrust vector controller added to a nose jet device is used. This missile quickly aligns the velocity vector since the thrust vector controller provides additional forces and moments to the interceptor. However, once the motor is entirely burned out, only the tail control remains available.

Additionally, this thesis uses an exo – atmospheric thrust vector controller missile since the aerodynamic forces can be considered to be negligible.

Such missiles are primarily used in the following engagement scenarios:

1. Surface – to – air exo – atmospheric missions. The targets are mainly strategic ballistic missiles operating in the end – game phase and at the high end of the atmosphere.

2. Air – to – air endo – atmospheric tail attack defence; since tail aerodynamic control may not always be sufficient to achieve interception.

3. Ground – to – air endo – atmospheric defence. In this case, the thrust vector control provides manoeuvring capability after launching the missile when the dynamic pressure is low.

The most popular hybrid application is shown in Figure 2.4. It is a Jet Vanes' design known as Sidewinder 9 – X. Its actuator was used to deflect both control surfaces resulting in an increase of the interceptor's reaction capability. However, the interceptor's initial non – minimum phase effect was also amplified.

Figure 2.4: Aim 9 – X Sidewinder.

2.3 Separated guidance and control architecture

Figure 2.5 represents the guidance and control loop for an interceptor missile with a traditional decoupled architecture. This architecture separates the guidance from the flight control function, presented below in two consecutive blocks in the figure. The aim of the given guidance and control loop is to achieve minimum miss distance.

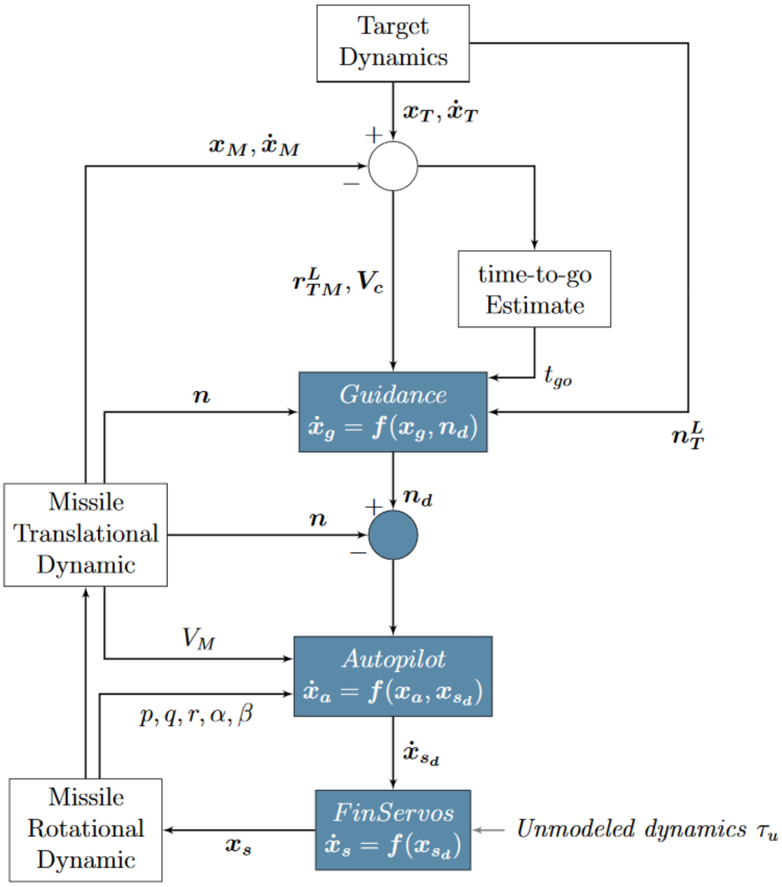

Target Dynamics
$x_T, \dot{x}_T$
+
$x_M, \dot{x}_M$
−
time-to-go Estimate
r^L_{TM}, V_c
t_{go}
n
Guidance $\dot{x}_g = f(x_g, n_d)$
n^L_T
Missile Translational Dynamic
+
n_d
n
−
V_M
Autopilot $\dot{x}_a = f(x_a, x_{s_d})$
p, q, r, α, β
$\dot{x}_{s_d}$
Missile Rotational Dynamic
x_s
FinServos $\dot{x}_s = f(x_{s_d})$
Unmodeled dynamics τ_u

Figure 2.5: Schematic view of a separated autopilot – guidance approach (taken from [40]).

Here, is a short description of the guidance and flight control functions. First, the guidance block takes the calculation of the acceleration command (n^L_d) in real time and on board of the missile. This acceleration command is based on both the engagement kinematic equations and the missile dynamics.

Then, the autopilot block transforms the acceleration command (n^L_d) into an appropriate missile's response (x_{s_d}). In other words, the autopilot is a feedback loop itself since its purpose is to monitor the achieved acceleration (n^B) and to generate the appropriate fin deflections [14]. The biggest delay introduced in the guidance and control loop is due to the autopilot's initial time response.

Nowadays, the great majority of interceptor missiles use a variant of proportional navigation. According to [4], a proportional navigation approach has been proved to be optimal for non – manoeuvring targets and missiles with ideal dynamics. But on the other hand, an augmented proportional navigation is the most frequently used approach when the target performs a constant manoeuvre; whereas an optimal guidance law is implemented when the missile is represented by a first – order dynamic system [3, 18, 21].

Former research has focused on the formulation of optimal guidance laws for the dual – controlled missile. Levy et al., in [15], provided a solution to the end – game engagement scenario linearised about the initial LOS for a separated two – loop autopilot – guidance design. However, this approach did not predict the missile's response accurately since it was designed based on a first – order autopilot model. As a result, the separated approach did not issue an appropriate guidance command and its performance was proven to be inferior than the one of the integrated approach.

2.4 Integrated guidance and control architecture

Figure 2.6 presents the integrated approach in a schematic way, where the control input both guides the missile and stabilises all the internal states of the interceptor. The integrated autopilot – guidance scheme is a finite – horizon control problem that depends on the time remaining until interception (t_{Go}).

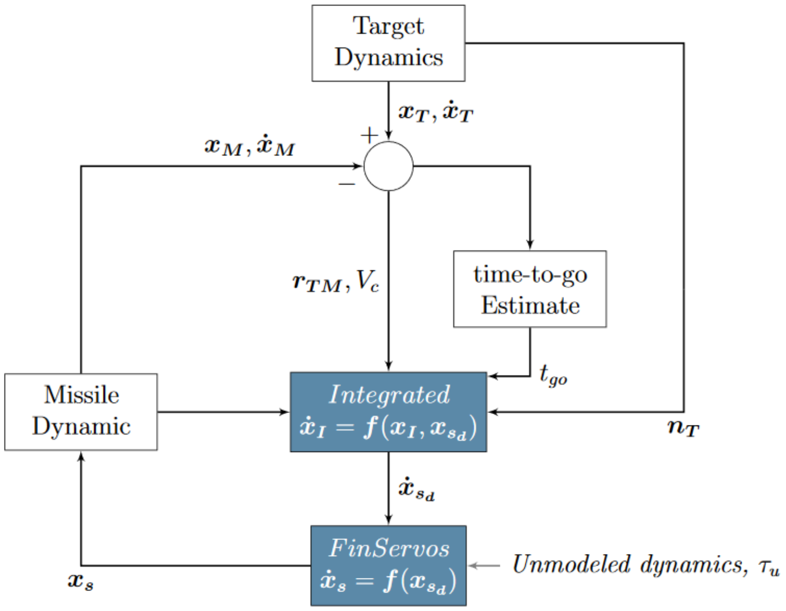

Target Dynamics
$\boldsymbol{x}_T, \dot{\boldsymbol{x}}_T$
+
$\boldsymbol{x}_M, \dot{\boldsymbol{x}}_M$
−
$\boldsymbol{r}_{TM}, V_c$
time-to-go Estimate
t_{go}
Missile Dynamic
Integrated $\dot{\boldsymbol{x}}_I = \boldsymbol{f}(\boldsymbol{x}_I, \boldsymbol{x}_{s_d})$
$\boldsymbol{n}_T$
$\dot{\boldsymbol{x}}_{s_d}$
FinServos $\dot{\boldsymbol{x}}_s = \boldsymbol{f}(\boldsymbol{x}_{s_d})$
Unmodeled dynamics, τ_u
$\boldsymbol{x}_s$

Figure 2.6: Schematic view of an integrated autopilot – guidance approach (taken from [40]).

Exploiting the coupling that exists between the guidance and flight control subsystems improves the performance of the interceptor. This idea of modifying the scheme presented in Figure 2.5 combining both subsystems is not new. In fact, a large amount of research has been carried out over the last 30 years in relation to this topic.

According to [14], the term ‘integrated guidance law’ has been used in the scientific literature in two different meanings. In its first meaning, ‘integrated’ means a single – loop guidance scheme that directly generates the control deflections, having a full – state feedback [12, 17, 21, 22, 30]. However, in its second meaning, the term ‘integrated’ means having an inner autopilot loop designed apart from the outer guidance loop, but also with a full – state feedback [32].

Starting from the basic definitions, an integrated single – loop guidance law is capable of ensuring the interception of the target as well as the inner stability of the airframe. An integrated two – loop autopilot – guidance law can be defined as a two – loop guidance and control system in which the design includes both the engagement kinematics and the missile dynamics.

Authors in [12, 16, 27, 28, 29, 30] have investigated the solution to the integrated single – loop and two – loop guidance – autopilot for dual control missiles. Table 2.2 summarises the numerous studies concerning this topic.

Reference	**Aerodynamic model**	**G&C**	**Control method**
[12]	Constant coefficients	IGA	SMC
[16]	Constant coefficients	IGA	LQR
[27]	Transfer functions	IGA	LQD
[28]	Transfer functions	IGA	LQD
[29]	Transfer functions	IGA	LQD
[30]	Constant coefficients	IGA	SMC
[33]	Constant coefficients	IGA	SMC

Table 2.2: References on DAC missile guidance.

Shima et al., in [28], derived a differential game guidance law taking into account the controllers’ bounds. It was proven that the canard channel should predominate over the tail’s channel in order to enhance the interceptor’s performance. The treatment of the bounds was done both directly [28], and indirectly through weighting factors in the cost index [29].

In the latter cases, the studies were focused on exo – atmospheric hybrid missiles but the results are extensive to DAC missiles. Figure 2.7 presents the aerodynamic model used in this research, where both control channels are represented by linear transfer functions. It can be seen that the guidance law directly introduces the commands to the canard and tail control channels.

Integrating the guidance and the flight control systems can also be achieved by using the sliding mode control technique. Shima et al., in [30], applied the sliding mode control method to derive an integrated single – loop guidance law for a dual control missile. The interceptor’s model was based on two control surfaces and a single terminal cost. The objective was to minimise a quadratic cost function given that there was a penalty imposed by the miss distance.

The SMC methodology was also used by Gutman and co – workers in [9], where the use of an additional controller and the need to derive two sliding manifolds ensured a higher performance of the interceptor.

In [33], back – stepping and high – order sliding mode control techniques were used in an integrated two – loop guidance – autopilot architecture. The inner loop was constrained to track the LOS rate command in the presence of uncertainties; whereas the purpose of the outer guidance loop was to the maintain the sliding surface close to zero.

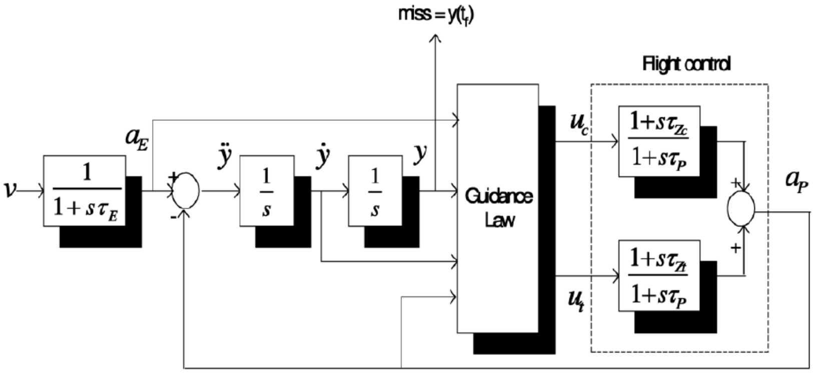


Figure 2.7: Dual control missile model [29].

Furthermore, it is valuable to briefly consider the approaches taken in several research studies with regard to canard – only configurations, since these approaches are extensive to dual control missiles. Besides, a canard – only configuration may be preferred over a dual – controlled one in research such as [18]. Hence, it is essential to take into consideration these approaches. Table 2.3 reviews the literature on the subject.

Reference	Aerodynamic model	G&C	Control method
[2]	Constant coefficients	IGA	OGL
[6]	Transfer functions	IGA	LQD
[9]	Transfer functions	IGA	LQD
[12]	Transfer functions	IGA	SMC
[15]	Transfer functions	IGA	LQD
[17]	Transfer functions	IGA	LQR
[21]	Constant coefficients	IGA	LQD
[22]	Transfer functions	IGA	TDC

Table 2.3: References on canard – control missile guidance.

Firstly, in [2], a variety of proportional navigation guidance laws were obtained in a closed – form solution. The transverse acceleration command was combined with the dynamics of the airframe in order to derive an autopilot control law. Park and co – workers, on the other hand, derived a time – delay system in [22] aiming to design an integrated single – loop scheme. In this case, the guidance law depended on the controller's sampling time.

Menon and Ohlmeyer [17] implemented the feedback linearisation technique combined with a linear control design method to derive the pseudo controller variable in an integrated single – loop guidance scheme. The transformation of the non – linear system into a linear, time – invariant one was achieved by differentiating the state – space equations.

Palumbo and co – workers [21] implemented an integrated single – loop design formulating the problem as a finite – horizon dynamic game and using partial – state information. The goal was to minimise a given performance index under the worst engagement scenario (target manoeuvre and measurement disturbances).

Shima et al. [15] demonstrated that the integrated single – loop and two – lop scheme achieved the same performance for a single – input single – output system (a single guidance command and terminal cost).

Finally, in [6], Gutman used the SMC technique to derive a canard configuration missile and demonstrate that a canard – only configuration provided better performance than the tail – only configuration. The sliding manifold was established by a zero effort miss term obtained from the differential game formulation. The same methodology was used by Shima et al., in [12], in order to derive an integrated single – loop guidance law for a canard control missile, where the objective was to minimise a quadratic cost function.

2.5 Conclusions

The latest research shows that the most frequently used methods to design autopilots for dual control missiles have been the non – linear second – order sliding mode one [12, 13, 33] in addition to the non – linear State Dependent Riccati Equation [4, 7, 37]. As a matter of fact, from the revision of MIMO non – linear methods, SDRE techniques emerge as the most suitable ones since these techniques are capable of handling the non – minimum phase effect of tail configurations.

On the other hand, linear control techniques including linear quadratic tracking [1, 18] or proportional plus integral optimal regulators [1] have also been widely presented in the literature.

As a matter of fact, a linear quadratic formulation is used in this thesis to design the autopilot for a dual control missile. The main purpose is to demonstrate the efficiency of integrated guidance approaches versus traditional ones, since the integrated approach is still being challenged in view of the fact that some research proves that it is not capable of providing any improvement in the performance versus a two – loop design in an idealised engagement scenario [16].

In order to address this challenge, this thesis deals with pitch control but does not address roll control. Additionally, the aerodynamic cross – control coupling effects due to interferences between the canard and the tail, and the effect of noises or radome errors are not analysed.

Chapter 3

Problem formulation

3.1 Introduction

This chapter describes the design assumptions and the linearised end – game geometry used for the analysis of the guidance laws. The scenario presented is based on a perfect information end – game scenario between a missile and a manoeuvring target. It is assumed that all missile and guidance variables are available for control feedback and are noise – free.

Following the design assumptions and the interception scenario description, the set of linear equations of motion is detailed and the theoretical solution framework to be used is summarised.

3.2 Design assumptions

The missile model and guidance laws are formulated based on the following assumptions:

1. It is considered a skid – to – turn roll – stabilised missile. The three – dimensional motion of such missile can be interpreted separating its motion into two perpendicular channels since the pitch and yaw channels can be treated as an independent two – dimensional problem. Therefore, only planar motion is considered.

2. Both the pursuer missile and the evasive target have linear dynamics.

3. Also both the pursuer missile and the evasive target have constant speeds.

4. It is assumed that perfect information of the future target's manoeuvre strategy is known and that it performs a constant manoeuvre.

$$\dot{a}_E = 0 \tag{3.1}$$

5. The plant disturbance is given to be zero.

6. The missile's and target's deviations from the collision triangle are negligible during the end – game phase. In this manner, the requirement on the impact angle is satisfied and, consequently, the relative end – game trajectory can be linearised about the initial line of sight (LOS).

3.3 End – game scenario description

The Figure 3.1 presents a schematic diagram of the planar engagement scenario. In this figure, the Cartesian reference frames are found to be $X_{1F} - O_{1F} - Z_{1F}$. It can be seen that the X axis is parallel to the initial line of sight (LOS_0) and the Z axis is aligned to the relative displacement between the adversaries (y). In other words, the Z axis is perpendicular to the initial LOS.

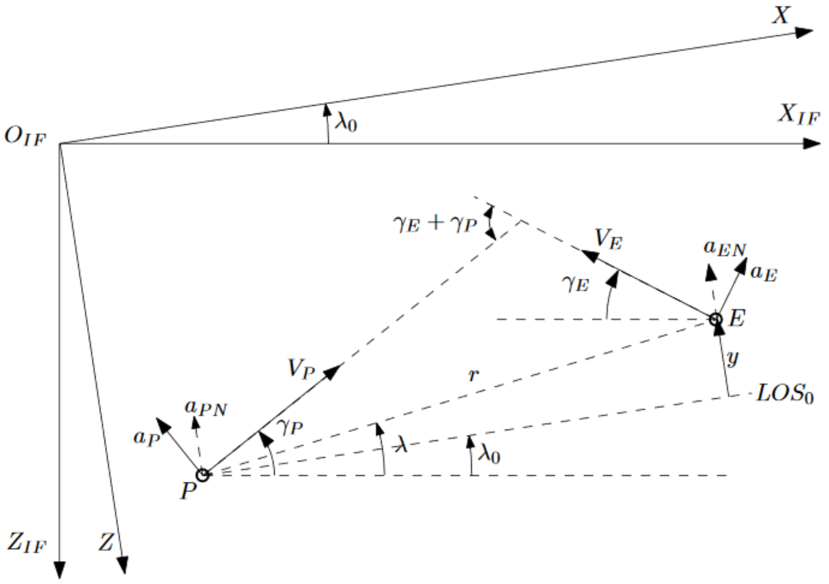


Figure 3.1: Planar engagement scenario.

In the schematic diagram, two subscripts can be found. These subscripts P and E, respectively represent the pursuing missile and the evading target. V denotes the speed whereas a represents the acceleration. Thus, a_{PN} and a_{EN} are respectively the pursuer and evader acceleration normal to the initial LOS. Additionally, γ represents the path angle.

Furthermore, r denotes the range between the missile and the target. The angle between the LOS and the X_{1F} axis is λ. Thus, λ_0 represents the angle between the initial LOS and the X_{1F} axis. Finally, the variable y represents the relative displacement between the missile and the target normal to the initial line of sight.

The pursuer's and evader's accelerations normal to LOS_0 are:

$$a_{PN} = a_P \cos(\gamma_{P0} - \lambda_0) \tag{3.2}$$

$$a_{EN} = a_E \cos(\gamma_{E0} + \lambda_0) \tag{3.3}$$

On the other hand, considering $\Delta\gamma_c$ is the impact angle error:

$$\Delta\gamma_c = \gamma_E + \gamma_P - \gamma_c \tag{3.4}$$

Hence, the kinematics equation can be written as follows:

$$\ddot{y} = a_{EN} - a_{PN} \tag{3.5}$$

$$\Delta\dot{\gamma}_c = \dot{\gamma}_E + \dot{\gamma}_P \tag{3.6}$$

where

$$\dot{\gamma}_E = \frac{a_E}{V_E} \tag{3.7}$$

$$\dot{\gamma}_P = \frac{a_P}{V_P} \tag{3.8}$$

Furthermore, we have assumed that the target performs a constant manoeuvre and that perfect information of the future target's manoeuvre strategy is known. These assumptions imply:

$$\dot{a}_E = 0 \tag{3.9}$$

3.4 Linear equations of motion

There are several equations of motion, and these equations can be divided into three types such as the kinematics or guidance equations, the equations of dynamics and the equations of servo model.

1. *Kinematic or guidance equations.* The guidance state vector contains the kinematic states of the engagement scenario. It includes the relative displacement between the missile and the target, the separation rate, the evader's normal acceleration and the impact angle error.

$$\mathbf{x_G} \in R^{n_G \times 1} \tag{3.10}$$

2. *Dynamic equations.* By definition the dynamic state vector contains the dynamic states of the engagement scenario. For instance, the missile body angle and the missile body angle rate.

$$\mathbf{x_D} \in R^{n_D \times 1} \tag{3.11}$$

3. *Servo model equations.* The equations of motion take into account the servo dynamics. In this manner, the nose and tail controller deflections are included in the servo state vector.

$$\mathbf{x_S} \in R^{n_S \times 1} \tag{3.12}$$

As it happens the equations of dynamics are coupled with the equations of servo model, but the reverse is not possible. Therefore:

$$\begin{bmatrix} \mathbf{\dot{x}_D} \\ \mathbf{\dot{x}_S} \end{bmatrix} = \begin{bmatrix} \mathbf{A_{DD}} & \mathbf{A_{DS}} \\ [\mathbf{0}] & \mathbf{A_S} \end{bmatrix} \begin{bmatrix} \mathbf{x_D} \\ \mathbf{x_S} \end{bmatrix} + \begin{bmatrix} [\mathbf{0}] \\ \mathbf{B_S} \end{bmatrix} \mathbf{\tilde{u}} \tag{3.13}$$

In the above given expression, $\mathbf{A_{DD}} \in R^{n_D \times n_D}$, $\mathbf{A_{DS}} \in R^{n_D \times n_S}$ and $\mathbf{A_S} \in R^{n_S \times n_S}$. Note that $\mathbf{\tilde{u}}$ is the input provided to the servo and $[\mathbf{0}]$ is a zero matrix of appropriate dimensions $(n_S \times n_D)$.

The set of kinematic equations is given by:

$$\mathbf{\dot{x}_G} = \begin{bmatrix} \mathbf{A_{GG}} & \mathbf{A_{GD}} & \mathbf{A_{GS}} \end{bmatrix} \begin{bmatrix} \mathbf{x_G} \\ \mathbf{x_D} \\ \mathbf{x_S} \end{bmatrix} \tag{3.14}$$

At the end, these equations can be summarised and there the general set of equations can be found and presented as following:

$$\begin{bmatrix} \mathbf{\dot{x}_G} \\ \mathbf{\dot{x}_D} \\ \mathbf{\dot{x}_S} \end{bmatrix} = \begin{bmatrix} \mathbf{A_{GG}} & \mathbf{A_{GD}} & \mathbf{A_{GS}} \\ [\mathbf{0}] & \mathbf{A_{DD}} & \mathbf{A_{DS}} \\ [\mathbf{0}] & [\mathbf{0}] & \mathbf{A_S} \end{bmatrix} \begin{bmatrix} \mathbf{x_G} \\ \mathbf{x_D} \\ \mathbf{x_S} \end{bmatrix} + \begin{bmatrix} [\mathbf{0}] \\ \mathbf{B_S} \end{bmatrix} \mathbf{\tilde{u}} \tag{3.15}$$

where

$$\mathbf{x} = \begin{bmatrix} \mathbf{x_G} & \mathbf{x_D} & \mathbf{x_S} \end{bmatrix}^T \tag{3.16}$$

$$\mathbf{F} = \begin{bmatrix} \mathbf{A_{GG}} & \mathbf{A_{GD}} & \mathbf{A_{GS}} \\ [\mathbf{0}] & \mathbf{A_{DD}} & \mathbf{A_{DS}} \\ [\mathbf{0}] & [\mathbf{0}] & \mathbf{A_S} \end{bmatrix} \tag{3.17}$$

$$\mathbf{G} = \begin{bmatrix} [\mathbf{0}] \\ \mathbf{B_S} \end{bmatrix} \tag{3.18}$$

The general set of equations $(\mathbf{\dot{x}} = \mathbf{F}\,\mathbf{x} + \mathbf{G}\,\mathbf{\tilde{u}})$ is time−varying. However, for simplicity, the time dependency is not explicitly written.

3.5 Conclusions

This chapter showcases a perfect information end – game scenario between a missile and a manoeuvring target as a two – dimensional problem. Furthermore, the formulation of the guidance laws is based on the assumptions of linear dynamics and constant speed for both adversaries; and a relative end – game trajectory linearised about the initial LOS.

In addition, the linear equations of motion have been introduced, taking into account the kinematic states of the engagement scenario (y, $\dot{y}$, a_{EN} and $\Delta\gamma_c$); the dynamic states (θ and $\dot{\theta}$) and both controller's deflections (δ_n and δ_t).

In later chapters different types of autopilot – guidance schemes for multi – input single – output missile systems are presented. The guidance laws are exemplified using a dual – controlled missile.

Chapter 4

Autopilot – guidance design

4.1 Introduction

This chapter provides the separated two – loop autopilot – guidance law, the integrated single – loop guidance law and the integrated two – loop autopilot – guidance law. In addition, a theorem is presented concerning a condition for achieving the same performance for the integrated single – loop design and the integrated two – loop design.

Note that the formulation of the guidance laws presented in this chapter as well as the theorem are valid for both linear quadratic single – input single – output systems and linear quadratic multi – input multi – output systems.

4.2 Separated two – loop autopilot – guidance law

Traditional missile guidance laws are designed independently of the autopilot. In its most basic form, the guidance law during the end – game phase requires that the missile generates an acceleration perpendicular to the speed's missile. This acceleration is equal in module to:

$$u = N' \, V_M \, \dot{\lambda} \tag{4.1}$$

where N' is the navigation constant and $\dot{\lambda}$ is the angle rate between the X_{1F} axis and the LOS.

Nevertheless, it is well – known that usual derived guidance laws, such as proportional navigation, are highly effective when the target has less manoeuvre capability. As targets increase their capabilities, the missile requires a more sophisticated guidance law.

In order to address this challenge, a guidance law using linear – quadratic optimal control theory is derived to obtain an analytic feedback solution. In general, optimal control techniques optimise the missile performance using a cost function.

The linear quadratic optimisation problem is formulated as [15]:

$$\min_{u(t)} J\left(\mathbf{x}(\mathbf{t_0}),\, \mathbf{u}(\mathbf{t_0}),\, \mathbf{t_0}\right) \;=\; \mathbf{x}^T(\mathbf{t_f})\, \mathbf{Q_f}\, \mathbf{x}\,(\mathbf{t_f}) + \int_{t_0}^{t_f} \mathbf{u}^T(\mathbf{t})\, \mathbf{R}\, \mathbf{u}\,(\mathbf{t})\; dt \tag{4.2}$$

subject to:

$$\dot{\mathbf{x}} \;=\; \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \tag{4.3}$$

In other words, the objective is to find a minimum control $\mathbf{u}\,(\mathbf{t})$ during the time interval $[t_0, t_f]$ that minimises the quadratic cost function of the final relative displacement between the pursuing missile and the evading target and subject to the specified dynamic constraints.

The plant will be designed using the decoupled guidance and control architecture shown in Figure 4.1.

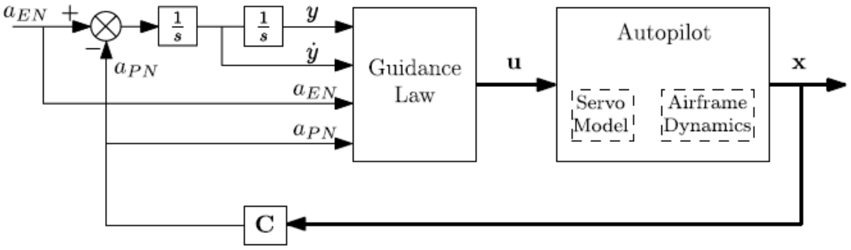


Figure 4.1: Block diagram of a separated two – loop autopilot – guidance law [15].

The outer guidance loop is stated as a solution of the finite – horizon control problem. Therefore, it is designed assuming a simplified low – order model of the closed – loop autopilot dynamics. The inner autopilot loop is treated as a solution of the infinite – horizon control problem, designed to follow the guidance acceleration commands. The problem is stated in a general two – dimensional form.

The separated guidance law is based on a low – order autopilot dynamics model. However, it has to be taken into account that when using ideal or first – order approximation of the autopilot dynamics, the airframe response isn't described accurately. This implies a deterioration of the end – game performance.

The formulation of the guidance law based on linear – quadratic optimal control and a low – order approximation of the autopilot is presented further in Section 5.6.

4.3 Integrated single – loop guidance law

This section formulates the guidance law's optimisation problem for the integrated single – loop scheme. The diagram is presented in Figure 4.2. It can be clearly seen that the guidance law is directly related to the dynamics of the airframe. As a result of this, the information of the internal states of the missile can be used by the guidance law.

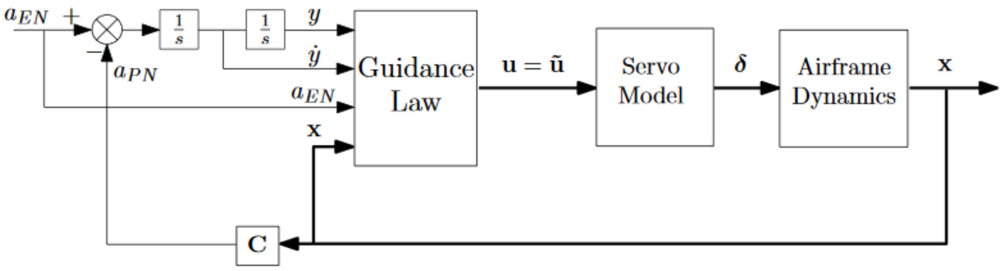


Figure 4.2: Block diagram of an integrated single – loop guidance law [15].

The guidance law is defined as a solution to the finite – time control problem. Different types of cost functions can be applied in order to achieve the control requirements. The quadratic cost function can be minimised and the expression is given here below:

$$\min_{u(t)} J \;=\; \mathbf{x}^T(\mathbf{t_f})\,\mathbf{Q_f}\,\mathbf{x}\,(\mathbf{t_f}) + \int_{t_0}^{t_f} \mathbf{u}^T(\mathbf{t})\,\mathbf{R}\,\mathbf{u}\,(\mathbf{t})\;dt \tag{4.4}$$

The optimal controller is also known as a linear feedback law and the expression is given by:

$$\mathbf{u}^* \;=\; \tilde{\mathbf{u}}^* \;=\; -\mathbf{R}^{-1}\,\mathbf{G}^{\mathbf{T}}\,\mathbf{P}\,\mathbf{x} \tag{4.5}$$

Substituting Equation 4.5 into Equation 3.15:

$$\dot{\mathbf{x}} \;=\; \left[\mathbf{F} - \mathbf{G}\,\mathbf{R}^{-1}\,\mathbf{G}^{\mathbf{T}}\,\mathbf{P}\right]\,\mathbf{x}(\mathbf{t}) \tag{4.6}$$

where **P** is known as the solution to the differential Riccati equation.

$$-\dot{\mathbf{P}} \;=\; \mathbf{P}\,\bar{\mathbf{F}} + \bar{\mathbf{F}}^{\mathbf{T}}\,\mathbf{P} - \mathbf{P}\,\mathbf{G_A}\,\mathbf{R_A^{-1}}\,\mathbf{G_A^T}\,\mathbf{P} + \tilde{\mathbf{Q}} \tag{4.7}$$

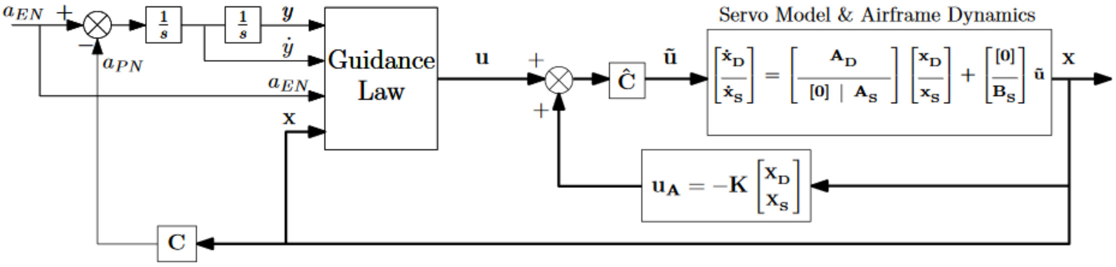
$$\mathbf{P}\,(\mathbf{t_f}) \;=\; \mathbf{Q_f} \tag{4.8}$$

4.4 Integrated two – loop autopilot – guidance law

In the two – loop design approach, the guidance system and the autopilot are designed individually as shown in Figure 4.3.

Here the input to the servo is no longer the guidance command as shown in Figure 4.2, but instead it is an equivalent controller as a result of the inner autopilot loop. It can be described with the help of the following terms:

$$\tilde{\mathbf{u}} \;=\; \hat{\mathbf{C}}\;(\,\mathbf{u} + \mathbf{u_A}\,) \tag{4.9}$$



a_{EN}
+
−
$\frac{1}{s}$
$\frac{1}{s}$
y
$\dot{y}$
a_{PN}
a_{EN}
x
Guidance Law
u
+
+
$\hat{\mathbf{C}}$
$\tilde{\mathbf{u}}$
Servo Model & Airframe Dynamics
$\begin{bmatrix}\dot{\mathbf{x}}_\mathbf{D}\\ \dot{\mathbf{x}}_\mathbf{S}\end{bmatrix} = \begin{bmatrix}\mathbf{A_D}\\ [0] \mid \mathbf{A_S}\end{bmatrix}\begin{bmatrix}\mathbf{x_D}\\ \mathbf{x_S}\end{bmatrix} + \begin{bmatrix}[0]\\ \mathbf{B_S}\end{bmatrix}\tilde{\mathbf{u}}$
x
$\mathbf{u_A} = -\mathbf{K}\begin{bmatrix}\mathbf{x_D}\\ \mathbf{x_S}\end{bmatrix}$
C

Figure 4.3: Block diagram of an integrated two – loop autopilot – guidance law [15].

Another formulation of the controller equation, equivalent to the previous one is:

$$\tilde{\mathbf{u}} \;=\; \hat{\mathbf{C}}\;(\,\mathbf{u} + \mathbf{u_A}\,) \;=\; \hat{\mathbf{C}}\left(\mathbf{u} - \begin{bmatrix}\mathbf{k_D} & \mathbf{k_S}\end{bmatrix}\begin{bmatrix}\mathbf{x_D}\\ \mathbf{x_S}\end{bmatrix}\right) \tag{4.10}$$

where $\mathbf{k_D} \in R^{m_G \times n_D}$, $\mathbf{k_S} \in R^{m_G \times n_S}$ and $\mathbf{u_A} \in R^{m_G \times 1}$. The input to the missile's autopilot is $\tilde{\mathbf{u}} \in R^{m \times 1}$ where $\hat{\mathbf{C}} \in R^{m \times m_G}$.

Hence the integration of the cost function can be represented by a function of $\tilde{\mathbf{u}}$. Therefore, the following expression:

$$J \;= \mathbf{x}^T(t_f)\,\mathbf{Q_f}\,\mathbf{x}(t_f) + \int_t^{t_f} \tilde{\mathbf{u}}^T(\tau)\,\mathbf{R}\,\tilde{\mathbf{u}}(\tau)\;d\tau \tag{4.11}$$

And substituting Equation 4.10 in Equations 3.15 – 3.18 provides the resulting set of equations:

$$\dot{\mathbf{x}} \;=\; \mathbf{F_A}\,\mathbf{x} + \mathbf{G_A}\,\mathbf{u} \tag{4.12}$$

$$\mathbf{x} \;=\; \begin{bmatrix}\mathbf{x_G^T} & \mathbf{x_D^T} & \mathbf{x_S^T}\end{bmatrix}^\mathbf{T} \tag{4.13}$$

where

$$\mathbf{F_A} \;=\; \begin{bmatrix}\mathbf{A_{GG}} & \mathbf{A_{GD}} & \mathbf{A_{GS}}\\ [\mathbf{0}] & \mathbf{A_{DD}} & \mathbf{A_{DS}}\\ [\mathbf{0}] & -\mathbf{B_S}\,\hat{\mathbf{C}}\,\mathbf{k_D} & \mathbf{A_S} - \mathbf{B_S}\,\hat{\mathbf{C}}\,\mathbf{k_S}\end{bmatrix} \tag{4.14}$$

$$\mathbf{G_A} \;=\; \begin{bmatrix}[\mathbf{0}]\\ \mathbf{B_S}\,\hat{\mathbf{C}}\end{bmatrix} \tag{4.15}$$

4.5 Guidance laws theorem

The theorem is formulated for the general form of the guidance laws presented in Sections 4.2, 4.3 and 4.4.

Definition 1. *A system is considered to be SISO if it has a single terminal cost (input) and a single controller (output). Furthermore, a system is considered to be MIMO if it has multiple terminal costs and multiple controllers.*

Theorem 1. *The solution of the linear quadratic optimisation problem of the integrated single – loop design (4.4 – 4.8) and of the integrated two – loop design (4.9 – 4.15) is identical for SISO and MIMO guidance systems. The necessary and sufficient condition is that $\hat{\boldsymbol{C}}$ is non singular.*

Proof. See Appendix A.2. □

Remark 1. *If $\hat{\boldsymbol{C}} \in R^{m \times m_G}$ is a non singular matrix, it must be a square matrix. For instance, its dimensions must be $m_G \times m_G$. Thus, the number of guidance commands must be identical to the number of inputs to the servo.*

Remark 2. *Recall that $\tilde{\boldsymbol{u}} \in R^{m \times 1}$ and $\boldsymbol{u} \in R^{m_G \times 1}$. If $m_G < m$, then $\hat{\boldsymbol{C}}$ constrains the command input to the autopilot model. In this manner, the guidance law cannot issue the appropriate command to each of the controllers since it loses degrees of freedom. For instance, a missile with multiple controls (both nose and tail controls) with a guidance law that issues a scalar acceleration command.*

4.6 Conclusions

This chapter explored two different types of guidance and control designs: a traditional separated approach and an integrated one. Moreover, the integrated approach, both a single – loop guidance law and a two – loop autopilot – guidance law were presented.

Furthermore, a theorem has been introduced showing that the necessary and sufficient condition to obtain the same performance for both integrated linear quadratic optimisation problems is that $\hat{\mathbf{C}}$ is a non singular matrix. The proof of Theorem 1 can be found in Appendix A.2.

Chapter 5

Case study

5.1 Introduction

The separated two – loop autopilot – guidance law, the integrated single – loop guidance law and the integrated two – loop autopilot – guidance law were introduced in their general form in Sections 4.2, 4.3 and 4.4. Thereafter, this chapter presents the formulation of the three different guidance systems applied to our missile's model.

Regarding the missile's model, it is a dual – controlled exo – atmospheric missile with both nose and tail controllers. Thus, the objective is to provide its linearised dynamic model, followed by the kinematic equations and the end – game scenario parameters. Also this chapter explores the corresponding autopilot designs (Section 5.5) depending on the number of inputs.

5.2 Dynamics model

The pursuer's model is based on a thrust vector control missile, where a nose jet device has been added. The standard configuration of the dual – controlled missile chosen is given in Figure 5.1.

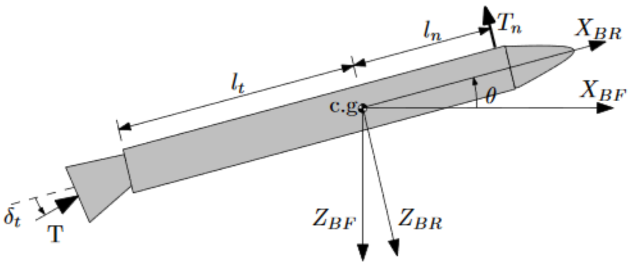


Figure 5.1: Configuration of a dual – controlled missile.

As mentioned above, when describing the end – game scenario, the frame $X - O - Z$ has its X axis aligned with the initial LOS whereas the Z axis is perpendicular to it.

In Figure 5.1, the coordinate system $X_{BF} - CG - Z_{BF}$ is considered to be parallel to the $X - O - Z$ frame and the origin of the frame is considered to be at the center of gravity of the missile. Regarding the $X_{BR} - CG - Z_{BR}$ coordinate system, it is a rotating frame linked to the missile's center of gravity. In this manner, the X_{BR} axis is always aligned with the longitudinal axis of the missile.

In its basic configuration, the missile has a nose controller (a thrust vector controller) and a tail controller. The missile's body angle is defined by θ; whereas δ_n and δ_t are the nose controller and the tail controller deflections. Furthermore, T and T_n denote the thrust and the nose jet force. Lastly, we define l_n and l_t respectively as the distance from the center of mass to the nose jet and the distance from the center of mass to the thrust vector controller.

The missile has been defined as an exo−atmospheric missile. Therefore, since the density is low outside the atmosphere, the aerodynamic forces and the wind can be ignored.

As it has been mentioned previously, the missile's thrust vector can be projected along the X and Z axis. The thrust force is described in Figure 5.2.

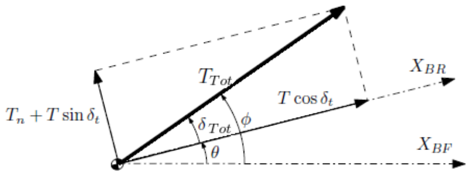


Figure 5.2: Thrust vector for a dual−controlled missile.

The two components are alongside and perpendicular to the missile's longitudinal axis X_{BR}. In this manner, letting δ_{Tot} denote the angle between the X_{BR} axis and the total thrust vector, and ϕ the angle between the X_{BF} axis and the total thrust vector, it can be proceeded in order to obtain the missile's dynamic model.

From the above considerations, the thrust vector components can be projected and these are written as follows:

$$T_t = T \sin\delta_t \tag{5.1}$$

$$T_x = T \cos\delta_t \tag{5.2}$$

Here, T_x denotes the axial thrust force. The total thrust force perpendicular to the longitudinal axis X_{BR} is given by:

$$T_y = T_n + T_t = T_n + T \sin\delta_t \tag{5.3}$$

Then, the total thrust force is given by:

$$T_{Tot} = \sqrt{T_x^2 + (T_n + T_t)^2} \tag{5.4}$$

And the angle between the X_{BR} axis and the total thrust vector is:

$$\sin\delta_{Tot} \;=\; \frac{T_n + T\sin\delta_t}{T_{Tot}} \tag{5.5}$$

If δ_t is considered very small, we can assume that $\sin\delta_t \cong \delta_t$ and $\cos\delta_t \cong 1$. The equation of moment with respect to the center of gravity can be written with the following expression:

$$I_{CG}\,\ddot{\theta} \;=\; T_n\, l_n - T\,\delta_t\, l_t \tag{5.6}$$

Define:

$$\delta_n \;\triangleq\; \frac{T_n}{T} \tag{5.7}$$

$$\delta_t \;\triangleq\; \frac{T_t}{T} \tag{5.8}$$

$$M_{\delta_n} \;\triangleq\; \frac{T\,l_n}{I_{CG}} \tag{5.9}$$

$$M_{\delta_t} \;\triangleq\; \frac{T\,l_t}{I_{CG}} \tag{5.10}$$

Substituting Equations 5.7 and 5.8 in Equation 5.4 and rearranging the total thrust, one gets:

$$T_{Tot} \;=\; T\sqrt{1+\left(\frac{T_n}{T}+\frac{T_t}{T}\right)^2} \tag{5.11}$$

Assuming that δ_n and δ_t are small angles and taking into account only first order terms:

$$T_{Tot} \;\cong\; T\sqrt{1+\delta_n^2+2\,\delta_n\,\delta_t+\delta_t^2} \;\simeq\; T \tag{5.12}$$

Recall that the angle between the thrust vector and the X_{BR} axis is given by:

$$\delta_{Tot} \;\simeq\; \delta_n + \delta_t \tag{5.13}$$

Substituting 5.9 and 5.10 in Equation 5.6:

$$\ddot{\theta} \;=\; M_{\delta_n}\,\delta_n - M_{\delta_t}\,\delta_t \tag{5.14}$$

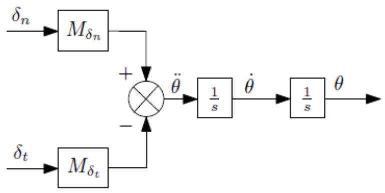


Figure 5.3: Dynamics for a dual−controlled missile.

In the same manner, we assume ϕ to be small in order to obtain a set of linear equations. Thus, we assume that $\sin\phi \cong \phi$. The pursuer's acceleration perpendicular to the initial LOS can be obtained using the perpendicular thrust component.

$$a_{PN} = \frac{T}{m}\phi \tag{5.15}$$

where

$$\phi = \theta + \delta_{Tot} \simeq \theta + \delta_n + \delta_t \tag{5.16}$$

Assuming first order servo models for both nose and tail controllers:

$$\delta_n = \frac{1}{\tau_n s + 1}\delta_n^c \tag{5.17}$$

$$\delta_t = \frac{1}{\tau_t s + 1}\delta_t^c \tag{5.18}$$

The dynamic state−space formulation results:

$$\begin{bmatrix} \dot{\mathbf{x}}_\mathbf{D} \\ \dot{\mathbf{x}}_\mathbf{S} \end{bmatrix} = \begin{bmatrix} \mathbf{A}_\mathbf{DD} & \mathbf{A}_\mathbf{DS} \\ [\mathbf{0}] & \mathbf{A}_\mathbf{S} \end{bmatrix} \begin{bmatrix} \mathbf{x}_\mathbf{D} \\ \mathbf{x}_\mathbf{S} \end{bmatrix} + \begin{bmatrix} [\mathbf{0}] \\ \mathbf{B}_\mathbf{S} \end{bmatrix} \tilde{\mathbf{u}} \tag{5.19}$$

where the dynamic state vector $\mathbf{x}_\mathbf{D}$, the servo model state vector $\mathbf{x}_\mathbf{S}$ and the control input vector $\tilde{\mathbf{u}}$ are defined as follows:

$$\mathbf{x}_\mathbf{D} = \begin{bmatrix} \theta & \dot{\theta} \end{bmatrix}^T \tag{5.20}$$

$$\mathbf{x}_\mathbf{S} = \begin{bmatrix} \delta_n & \delta_t \end{bmatrix}^T \tag{5.21}$$

$$\tilde{\mathbf{u}} = \begin{bmatrix} \delta_n^c & \delta_t^c \end{bmatrix}^T \tag{5.22}$$

Finally, the state space matrices are:

$$\mathbf{A}_\mathbf{D} = \begin{bmatrix} \mathbf{A}_\mathbf{DD} & \mathbf{A}_\mathbf{DS} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & M_{\delta_n} & -M_{\delta_t} \end{bmatrix} \tag{5.23}$$

$$\mathbf{A_S} = \begin{bmatrix} -\frac{1}{\tau_n} & 0 \\ 0 & -\frac{1}{\tau_t} \end{bmatrix} \tag{5.24}$$

$$\mathbf{B_S} = \begin{bmatrix} \frac{1}{\tau_n} & 0 \\ 0 & \frac{1}{\tau_t} \end{bmatrix} \tag{5.25}$$

Remark 3. *In a TVC missile, the mass, speed and inertia are time – varying. However, since the guidance law is being formulated for the terminal phase, these variables can be assumed to be constant.*

5.3 Kinematics equations

As stated before the kinematic equations are:

$$\ddot{y} = a_{EN} - a_{PN} = a_E \cos\left(\gamma_{E0} + \lambda_0\right) - a_P \cos\left(\gamma_{P0} - \lambda_0\right) \tag{5.26}$$

$$\Delta\dot{\gamma}_c = \dot{\gamma}_E + \dot{\gamma}_P = \frac{a_E}{V_E} + \frac{a_E}{V_E} \tag{5.27}$$

since

$$\Delta\gamma_c = \gamma_E + \gamma_P - \gamma_c \tag{5.28}$$

Assuming perfect information of the future target's manoeuvre strategy and considering that the target's manoeuvre is constant, then the following equation can be written:

$$\dot{a}_E = 0 \tag{5.29}$$

Furthermore, the pursuing missile's acceleration has been obtained in Equations 5.15 and 5.16 $\left(a_{PN} = \frac{T}{m}\phi\right)$. Therefore the following mathematical expressions:

$$a_{PN} = \mathbf{C}\begin{bmatrix} \mathbf{x_D} \\ \mathbf{x_S} \end{bmatrix} = \begin{bmatrix} \frac{T}{m} & 0 & \frac{T}{m} & \frac{T}{m} \end{bmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \\ \delta_n \\ \delta_t \end{bmatrix} \tag{5.30}$$

where

$$\mathbf{C} = \begin{bmatrix} \frac{T}{m} & 0 & \frac{T}{m} & \frac{T}{m} \end{bmatrix} \tag{5.31}$$

Finally, using Equations 5.26, 5.27 and 5.29, the kinematics state – space formulation is given by:

$$\dot{\mathbf{x}}_{\mathbf{G}} = \mathbf{A}_{\mathbf{G}}\,\mathbf{x}_{\mathbf{G}} + \mathbf{B}_{\mathbf{G}}\,\tilde{\mathbf{u}} \tag{5.32}$$

where

$$\mathbf{x}_{\mathbf{G}} = \begin{bmatrix} y & \dot{y} & a_{EN} & \Delta\dot{\gamma}_c \end{bmatrix}^T \tag{5.33}$$

$$\mathbf{A}_{\mathbf{G}} = \begin{bmatrix} \mathbf{A}_{\mathbf{GG}} & \mathbf{A}_{\mathbf{GD}} & \mathbf{A}_{\mathbf{GS}} \end{bmatrix} = \begin{bmatrix} \mathbf{A}_{\mathbf{GG}} & \mathbf{A}_{\mathbf{G,DS}} \end{bmatrix} \tag{5.34}$$

$$\mathbf{A}_{\mathbf{GG}} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{V_E\ \cos(\gamma_{E0}+\lambda_0)} & 0 \end{bmatrix} \tag{5.35}$$

$$\mathbf{A}_{\mathbf{G,DS}} = \begin{bmatrix} [\mathbf{0}] \\ -\mathbf{C} \\ [\mathbf{0}] \\ \frac{\mathbf{C}}{V_P\ \cos(\gamma_{P0}-\lambda_0)} \end{bmatrix} \tag{5.36}$$

$$\mathbf{B}_{\mathbf{G}} = [\mathbf{0}] \tag{5.37}$$

5.4 Scenario parameters

The scenario parameters can be represented with the help of Table 5.1.

Parameter	**Value**	**Units**
y_0	10	m
V_P	300	m/s
V_E	300	m/s
a_{EN}	10	m/s^2
T/m	120	m/s^2
γ_c	10	deg
γ_{P0}	0	deg
γ_{E0}	0	deg
θ_0	0	deg
M_{δ_n}	200	1/s^2
M_{δ_t}	200	1/s^2
τ_n	0.1	s
τ_t	0.15	s

Table 5.1: Scenario parameters values.

5.5 Autopilot design

The objective of the autopilot is to ensure that the missile follows the guidance acceleration commands issued by the guidance law. Essentially, the autopilot block for the two – loop case is designed in order to obtain a given damping ratio and gain margin. Two different schemes will be presented based on the number of inputs to the block.

Besides, in contrast with the two – loop autopilot scheme, the autopilot for the single – loop case is designed for the sake of consistency with the two – loop autopilot scheme.

5.5.1 Autopilot design for the single – loop case

The design gains for the single – loop scheme are given in Table 5.2. As mentioned above, these gains are chosen in accordance with the multi – input two – loop autopilot – guidance scheme.

Gain set	$\mathbf{K}_{\dot\theta_\mathbf{n}}$	$\mathbf{K}_{\dot\theta_\mathbf{t}}$	$\hat{\mathbf{c}}_\mathbf{n}$	$\hat{\mathbf{c}}_\mathbf{t}$
1	0.04	0.04	0.26	-0.1

Table 5.2: Single – loop autopilot design gains.

5.5.2 Autopilot design for the two – loop case

5.5.2.1 Single – input autopilot scheme

In the classical guidance – autopilot design, the guidance law issues a single acceleration command. This acceleration command is divided into two equivalent command signals and is delivered to the controllers by the vectorial control law given in Equation 5.38.

$$\hat{\mathbf{C}}_\mathbf{V} = \begin{bmatrix}\hat{c}_n & \hat{c}_n\end{bmatrix}^T \tag{5.38}$$

Figure 5.4 illustrates the single – input autopilot scheme, where the input is the acceleration command (a_P^c) issued by the guidance law.

Additionally, Table 5.3 presents the design gains for the autopilot scheme. In the single – input case, the guidance law is forced to use a combination of both controllers for all the values of α (the weight on the nose control command). Therefore, in order to investigate the limiting cases, the gains $\hat{c}_n$ and $\hat{c}_t$ are changed instead. From now on, 'Autopilot X' will refer to the gain set number X of the single – input scheme.

Basically, the first gain set is chosen for the sake of the consistency with the two – loop multi – input autopilot scheme; whereas the second set is established so as to analyse the resulting effects when the same control effort is enforced for both controllers ($|\hat{c}_n| = |\hat{c}_t| = 0.1$). Finally, the third set is defined in order to investigate the limiting case where there is only canard – control.

Gain set	$\mathbf{K}_{\dot{\theta}_n}$	$\mathbf{K}_{\dot{\theta}_t}$	$\hat{\mathbf{c}}_\mathbf{n}$	$\hat{\mathbf{c}}_\mathbf{t}$
1	0.04	0.04	0.26	-0.1
2	0.035	0.035	0.1	-0.1
3	0.04	0.04	20	-0.1

Table 5.3: Single – input autopilot design gains.

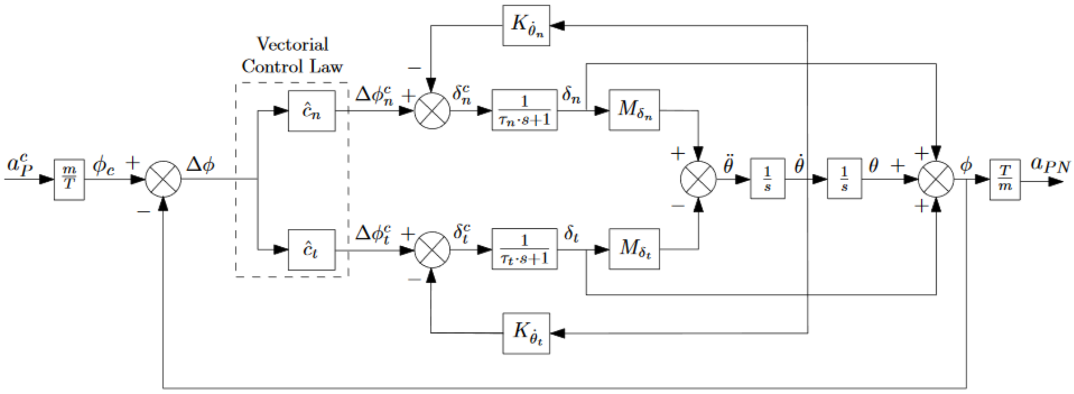


Figure 5.4: Autopilot design for the two – loop single – input guidance – autopilot scheme.

Figure 5.5 presents the missile's step response. It exemplifies the 3 autopilot versions given in Table 5.3. In all cases, the absence of the non – minimum phase effect due to the tail – only configuration can be appreciated at the beginning of the missile's response. Besides, it can be seen that these configurations ensure zero steady – state error to constant acceleration command inputs.

Furthermore, it can be appreciated that 'Autopilot 1' presents a peak response of 1.07 m/s^2 and a settling time to within 2% of 1.7 seconds. On the other hand, 'Autopilot 2' has a settling time to within 2% of 1.08 seconds. Lastly, it can be seen that 'Autopilot 3' has a peak response of 1.05 m/s^2 and a settling time to within 2% of 1.65 seconds.

5.5.2.2 Multi – input autopilot scheme

According to theorem 4.5, the number of guidance commands has to be identical to the number of controllers in order to achieve the same performance than the one of the single – loop autopilot case.

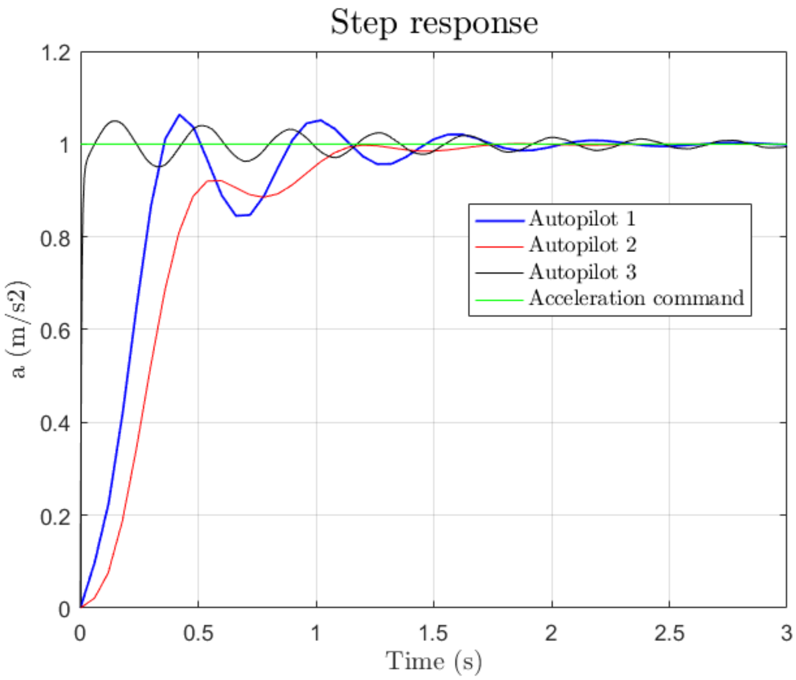


Figure 5.5: Step response for the two – loop single – input guidance – autopilot scheme.

Figure 5.6 illustrates the multi – input autopilot design, where two different inputs can be appreciated: the nose and the tail acceleration commands. This results in two degrees of freedom since each controller is controlled separately by the corresponding acceleration command.

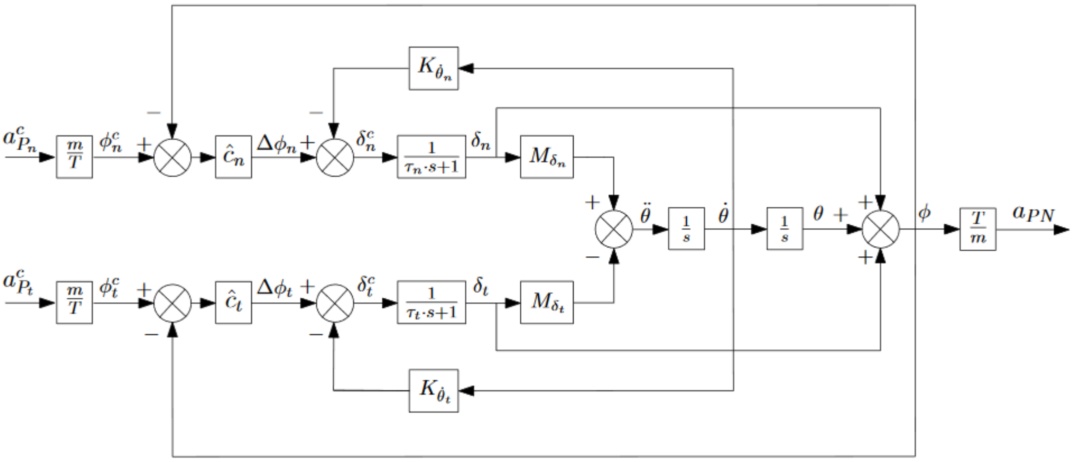


Figure 5.6: Autopilot design for the two – loop multi – input guidance – autopilot scheme.

The chosen design gains are shown in Table 5.4.

Gain set	$\mathbf{K}_{\dot{\theta}_\mathrm{n}}$	$\mathbf{K}_{\dot{\theta}_\mathrm{t}}$	$\hat{\mathbf{c}}_\mathbf{n}$	$\hat{\mathbf{c}}_\mathbf{t}$
1	0.04	0.04	0.26	-0.1

Table 5.4: Multi – input autopilot design gains.

Additionally, Figure 5.7 shows the step response for the two – loop multi – input guidance – autopilot scheme. The autopilot response has a peak amplitude of 1.07 m/s^2 and a settling time to within 2% of 1.7 seconds.

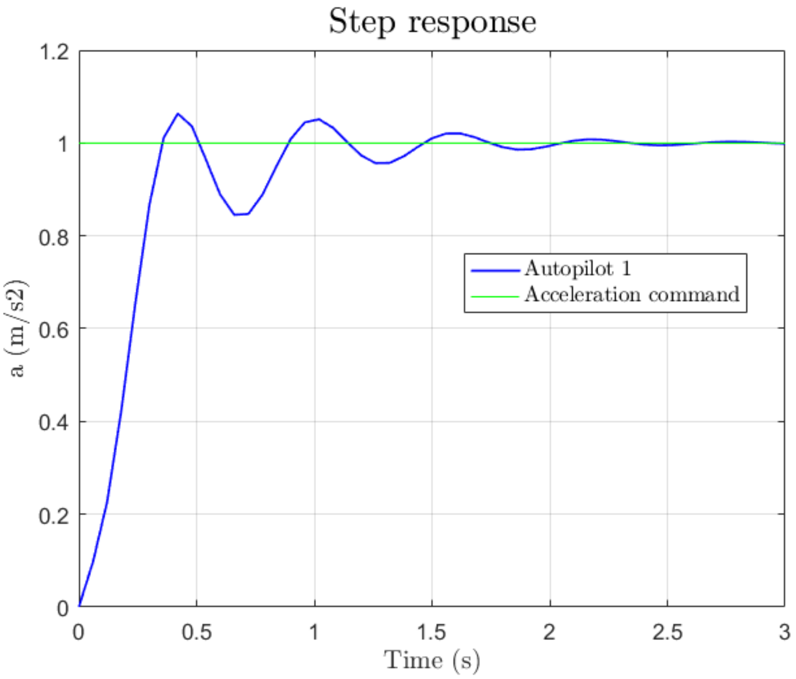


Figure 5.7: Step response for the two – loop multi – input guidance – autopilot scheme.

5.6 Separated two – loop autopilot – guidance law formulation

As seen before, when applying the optimal control theory to missile guidance, the results lead to a better end – game performance than when using the classical guidance laws such as proportional navigation. In this manner, when using OGL, the missile achieves smaller miss distances against manoeuvring targets.

Therefore, in this section, OGL is derived assuming that the adversaries are in the end – game phase. Note that we consider a separated two – loop autopilot – guidance design based on a low – order approximation of the autopilot dynamics. Thus, it is assumed that the missile responds to the acceleration command via a first – order transfer function:

$$\frac{a_P(s)}{a_c(s)} = \frac{1}{\tau\, s + 1} \tag{5.39}$$

The state vector of the linearised two dimensional problem includes the missile acceleration and is given by:

$$\mathbf{x} = \begin{bmatrix} y & \dot{y} & a_E & a_P \end{bmatrix}^T \tag{5.40}$$

Therefore, the linear quadratic optimisation problem formulated as:

$$\min_{u(t)} J\left(\mathbf{x}(\mathbf{t_0}),\, \mathbf{u}(\mathbf{t_0}),\, \mathbf{t_0}\right) = \mathbf{x}^T(\mathbf{t_f})\, \mathbf{Q_f}\, \mathbf{x}\,(\mathbf{t_f}) + \int_{t_0}^{t_f} \mathbf{u}^T(\mathbf{t})\, \mathbf{R}\, \mathbf{u}\,(\mathbf{t})\; dt \tag{5.41}$$

is subject to:

$$\begin{bmatrix} \dot{y} \\ \ddot{y} \\ \dot{a}_E \\ \dot{a}_P \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{-1}{\tau} \end{bmatrix} \begin{bmatrix} y \\ \dot{y} \\ a_E \\ a_P \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{1}{\tau} \end{bmatrix} u \tag{5.42}$$

The performance weighting matrices are:

$$R = 1 \tag{5.43}$$

$$\mathbf{Q_f} = \begin{bmatrix} b^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{5.44}$$

where b is the penalty imposed by the miss distance. In other words, b is the penalty on the relative displacement between the adversaries when $t = t_f$. Therefore, a large value of b implies smaller miss distances. For perfect interception, it is necessary to assume $b \rightarrow \inf$.

Finally, the closed – form solution of the guidance law is [19]:

$$u = a_P^c = \frac{\Lambda_G}{t_{Go}^2}\left[y + \dot{y}\, t_{Go} + \frac{1}{2}\, a_E\, t_{Go}^2 - a_E\, \tau^2 \left(e^{-\left(\frac{t_{Go}}{\tau}\right)} + \left(\frac{t_{Go}}{\tau}\right) - 1 \right) \right] \tag{5.45}$$

where the navigation ratio Λ_G is given by:

$$\Lambda_G = \frac{6\left(\frac{t_{Go}}{\tau}\right)^2\left(e^{-\left(\frac{t_{Go}}{\tau}\right)}+\left(\frac{t_{Go}}{\tau}\right)-1\right)}{2\left(\frac{t_{Go}}{\tau}\right)^3+3+6\left(\frac{t_{Go}}{\tau}\right)-6\left(\frac{t_{Go}}{\tau}\right)^2-12\left(\frac{t_{Go}}{\tau}\right)e^{-\left(\frac{t_{Go}}{\tau}\right)}-3e^{-2\left(\frac{t_{Go}}{\tau}\right)}} \tag{5.46}$$

Furthermore, the zero effort miss distance is given by:

$$z = y+\dot{y}\,t_{Go}+\frac{1}{2}\,a_E\,t_{Go}^2-a_E\,\tau^2\left(e^{-\left(\frac{t_{Go}}{\tau}\right)}+\left(\frac{t_{Go}}{\tau}\right)-1\right) \tag{5.47}$$

Optimal guidance laws are considered as an optimisation of proportional navigation laws with two additional gains in the linear model. Therefore, OGL can deal with high manoeuvring targets.

However, the acceleration command is issued by the outer guidance loop, which means that there is no feedback available of the target state to the control system. Furthermore, the autopilot introduces a delay as the autopilot and the guidance systems are operating at a different frequency; and this delay has a considerable effect on the miss distance.

This is why it is necessary to study a different approach where both the guidance system and the autopilot are integrated so as to achieve a better end – game performance. This approach will be formulated below in Sections 5.7 and 5.8.

5.7 Integrated single – loop guidance law formulation

The application of the modern control theory allows a new approach for studying the relation between the missile guidance law and the autopilot based on the integration of both subsystems, which means that there is no separation between them.

The result is a guidance law that issues a two dimensional vector with the nose and tail deflection commands. Besides, the guidance law takes into consideration both the missile and the target states.

The general form of the single – loop guidance law optimisation problem was already presented in Section 4.3. This section deals with the formulation of the guidance law optimisation problem applied to the chosen test case – a dual control missile.

Using the kinematic equations (5.32 - 5.37) described in Section 5.3 and the dynamic model equations (5.19 - 5.25) described in Section 5.2, the state – space representation of the linear set of equations becomes:

$$\dot{\mathbf{x}} = \mathbf{F}\,\mathbf{x}+\mathbf{G}\,\mathbf{u} \tag{5.48}$$

$$\tilde{\mathbf{u}} = \mathbf{u} = \begin{bmatrix}\delta_n^c & \delta_t^c\end{bmatrix}^T \tag{5.49}$$

where

$$\mathbf{x} = \begin{bmatrix}y & \dot{y} & a_{EN} & \Delta\gamma_c & \theta & \dot{\theta} & \delta_n & \delta_t\end{bmatrix}^T \tag{5.50}$$

$$\mathbf{F} = \begin{bmatrix}\mathbf{A_{11}} & \mathbf{A_{12}} \\ [\mathbf{0}] & \mathbf{A_{22}}\end{bmatrix} \tag{5.51}$$

$$\mathbf{A_{11}} = \begin{bmatrix}0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{V_E\ \cos(\gamma_{E0}+\lambda_0)} & 0\end{bmatrix} \tag{5.52}$$

$$\mathbf{A_{12}} = \begin{bmatrix}0 & 0 & 0 & 0 \\ \frac{-T}{m} & 0 & \frac{-T}{m} & \frac{-T}{m} \\ 0 & 0 & 0 & 0 \\ \frac{T}{m}\,\frac{1}{V_P\ \cos(\gamma_{P0}-\lambda_0)} & 0 & \frac{T}{m}\,\frac{1}{V_P\ \cos(\gamma_{P0}-\lambda_0)} & \frac{T}{m}\,\frac{1}{V_P\ \cos(\gamma_{P0}-\lambda_0)}\end{bmatrix} \tag{5.53}$$

$$\mathbf{A_{22}} = \begin{bmatrix}0 & 1 & 0 & 0 \\ 0 & 0 & M_{\delta_n} & -M_{\delta_t} \\ 0 & 0 & -\frac{1}{\tau_n} & 0 \\ 0 & 0 & 0 & -\frac{1}{\tau_t}\end{bmatrix} \tag{5.54}$$

$$\mathbf{G} = \begin{bmatrix}[\mathbf{0}] & [\mathbf{0}] \\ \frac{1}{\tau_n} & [\mathbf{0}] \\ [\mathbf{0}] & \frac{1}{\tau_t}\end{bmatrix} \tag{5.55}$$

Finally, the chosen weight matrices are:

$$\mathbf{R} = \begin{bmatrix}\alpha & 0 \\ 0 & 1\end{bmatrix} \tag{5.56}$$

$$\mathbf{Q_f}(i,j) = \begin{cases} b^2 & \text{if } \ i=j=1 \\ c^2 & \text{if } \ i=j=4 \\ 0 & \text{otherwise}\end{cases} \tag{5.57}$$

where $1 \leq i,\, j \leq 8$.

5.8 Integrated two – loop autopilot – guidance law formulation

In Section 4.4 the general form of the two – loop guidance law optimisation problem was introduced. In this section, the optimisation problem is applied to our chosen test case, and it is exemplified using the two autopilot schemes presented in Section 5.5.

Again, the input to the autopilot is an equivalent controller, which can be rewritten as a function of the state vector and the guidance controller as follows:

$$\tilde{\mathbf{u}} = \hat{\mathbf{C}}\,(\mathbf{u} + \mathbf{u_A}) = \hat{\mathbf{C}}\left(\mathbf{u} - \begin{bmatrix}\mathbf{k_D} & \mathbf{k_S}\end{bmatrix}\begin{bmatrix}\mathbf{x_D}\\ \mathbf{x_S}\end{bmatrix}\right) \tag{5.58}$$

$$\tilde{\mathbf{u}} = \mathbf{C_{Eq}}\,\mathbf{x} + \mathbf{D_{Eq}}\,\mathbf{u} \tag{5.59}$$

where

$$\mathbf{x} = \begin{bmatrix}\mathbf{x_G^T} & \mathbf{x_D^T} & \mathbf{x_S^T}\end{bmatrix}^{\mathbf{T}} \tag{5.60}$$

$$\mathbf{C_{Eq}} = \begin{bmatrix}0 & -\hat{\mathbf{C}}\,\mathbf{k_D} & -\hat{\mathbf{C}}\,\mathbf{k_S}\end{bmatrix} \tag{5.61}$$

$$\mathbf{D_{Eq}} = \hat{\mathbf{C}} \tag{5.62}$$

Substituting Equation 5.58 into Equation 4.11:

$$J = \mathbf{x}^T(t_f)\,\mathbf{Q_f}\,\mathbf{x}(t_f) + \int_t^{t_f}\left(\mathbf{u}^T\,\mathbf{R_A}\,\mathbf{u} + 2\,\mathbf{x}^T\,\mathbf{S_A}\,\mathbf{u} + \mathbf{x}^T\,\mathbf{Q_A}\,\mathbf{x}\right)(\tau)\,d\tau \tag{5.63}$$

where

$$\mathbf{Q_A} = \begin{bmatrix}[\mathbf{0}] & [\mathbf{0}]\\ [\mathbf{0}] & \mathbf{C_{Eq}^T}\,\tilde{\mathbf{R}}\,\mathbf{C_{Eq}}\end{bmatrix} \tag{5.64}$$

$$\mathbf{S_A} = \begin{bmatrix}[\mathbf{0}]\\ \mathbf{C_{Eq}^T}\,\tilde{\mathbf{R}}\,\mathbf{D_{Eq}}\end{bmatrix} \tag{5.65}$$

$$\mathbf{R_A} = \mathbf{D_{Eq}^T}\,\tilde{\mathbf{R}}\,\mathbf{D_{Eq}} \tag{5.66}$$

It is necessary to take into account the fact that the guidance commands in the multi – input autopilot scheme are the acceleration commands; whereas in the single – input autopilot scheme, the guidance commands are the deflection commands.

Therefore, the acceleration command has to be multiplied by the appropriate factor (m/T) in order to obtain the deflections.

$$\phi^c = \frac{m}{T} a_P^c \tag{5.67}$$

Using the kinematic equations (5.32 - 5.37) from Section 5.3 and the dynamic model equations (5.19 - 5.25) from Section 5.2, the state – space representation results:

$$\dot{\mathbf{x}} = \mathbf{F_A}\,\mathbf{x} + \mathbf{G_A}\,\mathbf{u} \tag{5.68}$$

where $\mathbf{x}$ is defined in Equation 5.50. The matrix $\mathbf{F_A}$ is given by:

$$\mathbf{F} = \begin{bmatrix} \mathbf{A_{11}} & \mathbf{A_{12}} \\ [\mathbf{0}] & \mathbf{A_{22}} \end{bmatrix} \tag{5.69}$$

$$\mathbf{A_{22}} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & M_{\delta_n} & -M_{\delta_t} \\ -\frac{\hat{c}_n}{\tau_n} & -\frac{K_{\dot{\theta}_n}}{\tau_n} & -\frac{1+\hat{c}_n}{\tau_n} & -\frac{\hat{c}_n}{\tau_n} \\ -\frac{\hat{c}_t}{\tau_t} & -\frac{K_{\dot{\theta}_t}}{\tau_t} & -\frac{\hat{c}_t}{\tau_t} & -\frac{1+\hat{c}_t}{\tau_t} \end{bmatrix} \tag{5.70}$$

The weight matrices are:

$$\mathbf{R} = \begin{bmatrix} \alpha & 0 \\ 0 & 1 \end{bmatrix} \tag{5.71}$$

$$\mathbf{Q_f}(i,j) = \begin{cases} b^2 & \text{if } i = j = 1 \\ c^2 & \text{if } i = j = 4 \\ 0 & \text{otherwise} \end{cases} \tag{5.72}$$

where $1 \leq i,\ j \leq 8$.

5.8.1 Single – input autopilot scheme

The guidance command, as it has been explained before, is determined by the autopilot scheme. In the single – input case, the guidance command is the pursuer's acceleration command $u = a_P^c$.

Note that the guidance law issues a scalar (a single command) as it is shown in Figure 5.4. In this manner, the acceleration is divided into two equivalent commands using the vectorial control law given by $\hat{c}_n$ and $\hat{c}_t$. Thus, the equivalent controllers are given by the following expression:

$$\tilde{\mathbf{u}} = \begin{bmatrix} \hat{c}_n \frac{m}{T} \\ \hat{c}_t \frac{m}{T} \end{bmatrix} u - \begin{bmatrix} \hat{c}_n & K_{\dot{\theta}_n} & \hat{c}_n & \hat{c}_n \\ \hat{c}_t & K_{\dot{\theta}_t} & \hat{c}_t & \hat{c}_t \end{bmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \\ \delta_n \\ \delta_t \end{bmatrix} \tag{5.73}$$

where u is the acceleration command. As mentioned above, the matrices $\mathbf{C_{Eq}}$ and $\mathbf{D_{Eq}}$ are given by the autopilot type.

$$\mathbf{C_{Eq}} = \begin{bmatrix} [\mathbf{0}] & \hat{c}_n & K_{\dot{\theta}_n} & \hat{c}_n & \hat{c}_n \\ [\mathbf{0}] & \hat{c}_t & K_{\dot{\theta}_t} & \hat{c}_t & \hat{c}_t \end{bmatrix} \tag{5.74}$$

$$\mathbf{D_{Eq}} = \begin{bmatrix} \hat{c}_n \frac{m}{T} \\ \hat{c}_t \frac{m}{T} \end{bmatrix} \tag{5.75}$$

The state space matrix $\mathbf{G_A}$ is also determined by the autopilot block diagram. Therefore, in the single – input case:

$$\mathbf{G_A} = \begin{bmatrix} [\mathbf{0}] \\ \frac{\hat{c}_n}{\tau_n} \frac{m}{T} \\ \frac{\hat{c}_t}{\tau_t} \frac{m}{T} \end{bmatrix} \tag{5.76}$$

5.8.2 Multi – input autopilot scheme

Using the multi – input autopilot scheme shown in Figure 5.6, it can be seen that the guidance law issues a two dimensional vector given by:

$$\mathbf{u} = \begin{bmatrix} a^c_{P_n} & a^c_{P_t} \end{bmatrix}^T \tag{5.77}$$

Thus, the equivalent controller is obtained as follows:

$$\tilde{\mathbf{u}} = \begin{bmatrix} \hat{c}_n \frac{m}{T} & 0 \\ 0 & \hat{c}_t \frac{m}{T} \end{bmatrix} \mathbf{u} - \begin{bmatrix} \hat{c}_n & K_{\dot{\theta}_n} & \hat{c}_n & \hat{c}_n \\ \hat{c}_t & K_{\dot{\theta}_t} & \hat{c}_t & \hat{c}_t \end{bmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \\ \delta_n \\ \delta_t \end{bmatrix} \tag{5.78}$$

where the matrices $\mathbf{C_{Eq}}$ and $\mathbf{D_{Eq}}$ are given by:

$$\mathbf{C_{Eq}} = \begin{bmatrix} [\mathbf{0}] & \hat{c}_n & K_{\dot{\theta}_n} & \hat{c}_n & \hat{c}_n \\ [\mathbf{0}] & \hat{c}_t & K_{\dot{\theta}_t} & \hat{c}_t & \hat{c}_t \end{bmatrix} \tag{5.79}$$

$$\mathbf{D_{Eq}} = \begin{bmatrix} \hat{c}_n \frac{m}{T} & 0 \\ 0 & \hat{c}_t \frac{m}{T} \end{bmatrix} \tag{5.80}$$

In the multi – input autopilot scheme, the state space matrix $\mathbf{G_A}$ is also determined by the autopilot scheme and it is given by:

$$\mathbf{G_A} = \begin{bmatrix} [\mathbf{0}] & [\mathbf{0}] \\ \frac{\hat{c}_n}{\tau_n}\frac{m}{T} & 0 \\ 0 & \frac{\hat{c}_t}{\tau_t}\frac{m}{T} \end{bmatrix} \tag{5.81}$$

5.9 Conclusions

After starting with the general form of the guidance laws in Sections 4.2 – 4.4, the above sections provided the formulation of the separated two – loop autopilot – guidance law, the integrated single – loop guidance law and the integrated two – loop autopilot – guidance law applied to a dual – controlled exo – atmospheric missile.

In order to do so, the dual – controlled aerodynamic model and the kinematic equations based on the linearised end – game engagement scenario were also provided. Besides, the scenario parameters values were defined in Section 5.4.

Furthermore, it required the exploration of two possible schemes for the autopilot when studying the integrated two – loop autopilot – guidance law scheme. The two autopilot designs were based on the number of inputs to the autopilot block. In the single – input autopilot scheme, the guidance law issued the acceleration command; whereas in the multi – input autopilot scheme, the guidance law issued a two dimensional vector: the nose and tail controller commands. Finally, the design gains for the autopilot blocks were chosen in order to achieve a given damping ratio and gain margin.

Chapter 6

Simulation results

6.1 Introduction

The purpose of this chapter is to compare the separated two – loop autopilot – guidance law and the integrated two – loop autopilot – guidance law with the integrated single – loop law. In the integrated single – loop law, the airframe dynamics and kinematics are combined. As a result of this, the response of the missile is described in a more accurate way.

Therefore, the integrated single – loop is expected to achieve the best end – game performance and, consequently, it can be used as a benchmark to analyse the other guidance laws' performances. Furthermore, the results of the integrated two – loop autopilot – guidance case will be computed for both the single – input autopilot scheme (Figure 5.4) and the multi – input autopilot scheme (Figure 5.6).

The results of the simulations will be displayed with the help of a Pareto front. Sample runs will also be used in order to explain the simulations.

6.2 Pareto front

The Pareto front has been extensively used in multi – objective optimisation problems in order to compare different solutions.

The concept of Pareto can be defined as a set of non – dominated solutions, where none of the objectives can be modified or improved without worsening at least one objective among the chosen objectives.

The concept will be illustrated using the given cost function:

$$J \,=\, b^2\, y^2(t_f) + c^2\, \Delta\gamma_c^2(t_f) + \int_{t_0}^{t_f} \left\{ \alpha \delta_n^{c\,2}(t) + \delta_t^{c2}(t) \right\}\, dt \tag{6.1}$$

When imposing zero miss distance ($b \rightarrow \inf$) and zero impact angle error ($c \rightarrow \inf$), the cost function depends only on α. Therefore, there are two design criteria:

$$f_1 = \int_{t_0}^{t_f} \delta_n^{c\,2}(t)\,dt \tag{6.2}$$

$$f_2 = \int_{t_0}^{t_f} \delta_t^{c2}(t)\,dt \tag{6.3}$$

The Pareto curve shown in Figure 6.1 is obtained by changing the variable α of the cost function. It is clear that the curve **A** – **B** represents the Pareto curve. Thus, any point in the curve cannot be improved in one criteria without worsening the other criteria.

It has to be pointed out that the point **P**, which is above the curve **A** – **B**, is not an optimal solution since any point in the curve **A** – **B** has at least one of the two design criteria optimised.

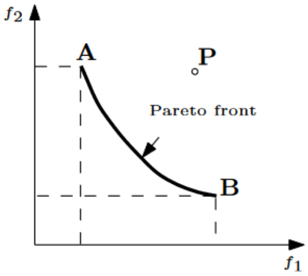


Figure 6.1: Example of a perfect interception Pareto front.

6.3 Separated two – loop autopilot – guidance law

The simulation results are shown in Figure 6.2 via a Pareto front for a perfect interception case, and for a separated guidance law and an integrated single – loop guidance law. Hence, the obtained curves present the optimal performance in terms of both controllers' effort and subject to the given guidance law approach.

Each point located in the Pareto front curve is a simulation result obtained for a fixed miss distance penalty (b), a fixed impact angle error penalty (c) and a different nose control effort penalty (α). Note that, in both schemes, when $\alpha \to \inf$, the nose control effort decreases; whereas when $\alpha \to 0$, the tail control effort does. The values of the parameters b and c are given in Table 6.1.

Analysing the Pareto front curves, the separated guidance law curve appears above the integrated one, which implies that the optimal combination of nose and tail control efforts is given by the integrated approach. Essentially, the separated two – loop scheme was expected to predict a less accurate missile's response since it has no feedback available of the target states to the control system, and first – order approximation of the autopilot dynamics are assumed. In addition, the integrated approach has a full feedback to the control system and, therefore, uses information on both states (kinematic and dynamic states); whereas the separated one only uses information on the kinematic states. Thus, Figure 6.2 demonstrates the superiority of the integrated approach.

Gain set	b	c
1	0.0001	1

Table 6.1: Separated two – loop design parameters.

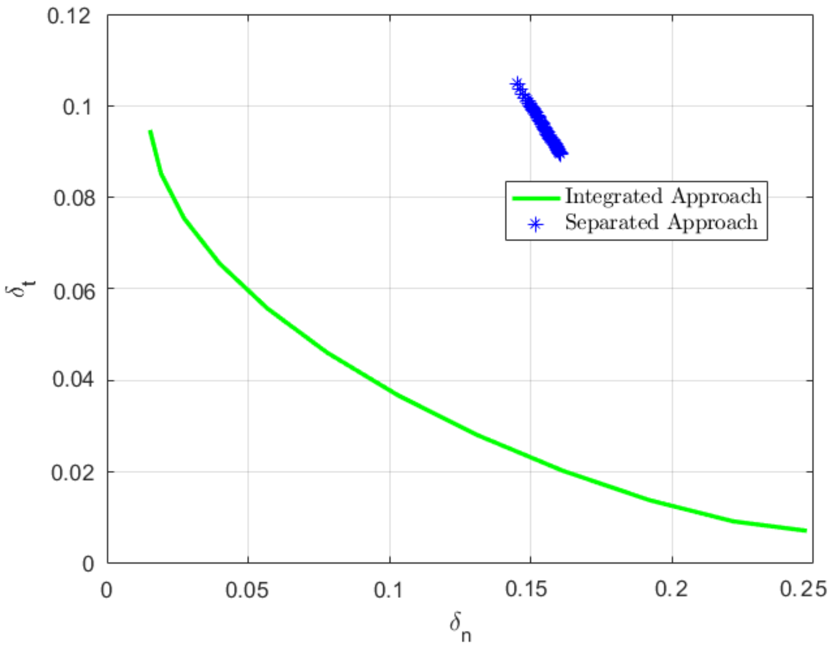


Figure 6.2: Pareto front curves for a perfect interception case: separated guidance law and integrated single – loop guidance law.

6.4 Integrated two – loop single – input autopilot – guidance law

Figure 6.3 presents the Pareto front curves for the integrated single – loop multi – input guidance law and the integrated two – loop single – input autopilot – guidance law. It is done for a perfect interception case. As expected, the control effort required for the integrated two – loop is greater than the control effort required for the single – loop multi – input case.

As a matter of fact, the single – input autopilot in the two – loop configuration cannot generate the appropriate commands to control each of the controllers separately since it loses one degree of freedom. Therefore, for a multi – input multi – output system, the guidance law is not precise when the number of controller commands is smaller than the number of controllers available in the missile's model.

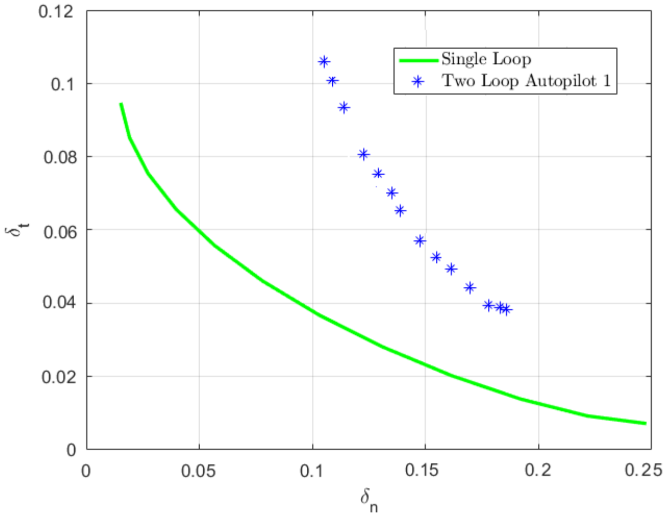


Figure 6.3: Pareto front curves for a perfect interception case: integrated two – loop single – input autopilot – guidance law and integrated single – loop guidance law.

Furthermore, the single – input two – loop autopilot scheme is exemplified using different design gains in order to study limiting cases. Table 6.2 summarise the 3 autopilot versions and the fixed values for the miss distance and impact angle error weights.

Gain set	$\mathbf{K}_{\dot{\theta}_n}$	$\mathbf{K}_{\dot{\theta}_t}$	$\hat{\mathbf{c}}_\mathbf{n}$	$\hat{\mathbf{c}}_\mathbf{t}$	b	c
1	0.04	0.04	0.26	-0.1	0.0001	1
2	0.035	0.035	0.1	-0.1	0.0001	1
3	0.04	0.04	20	-0.1	0.0001	1

Table 6.2: Integrated two – loop single – input design parameters.

Note that changing the value of α would not have any effect on the second gain set, since $M_{\delta_n} = M_{\delta_t}$ and the same control effort is enforced to both controllers.

Figures 6.4 – 6.7 present the results for the 3 different versions of the integrated two – loop single – input autopilot – guidance law and the integrated single – loop multi – input guidance law. The sample runs were obtained for the given fixed weights: $b = 0.0001$, $c = 1$ and $\alpha = 1$.

It can be appreciated that the miss distance tends to zero at the end of the engagement in all cases. Furthermore, Figure 6.6 illustrates the missile's body angle and its rate. It can be seen that for 'Autopilot 2' the values of θ and $\dot{\theta}$ remain zero throughout the interception. Thus, taking $|\hat{c}_n| = |\hat{c}_t|$ restrained the missile body angle.

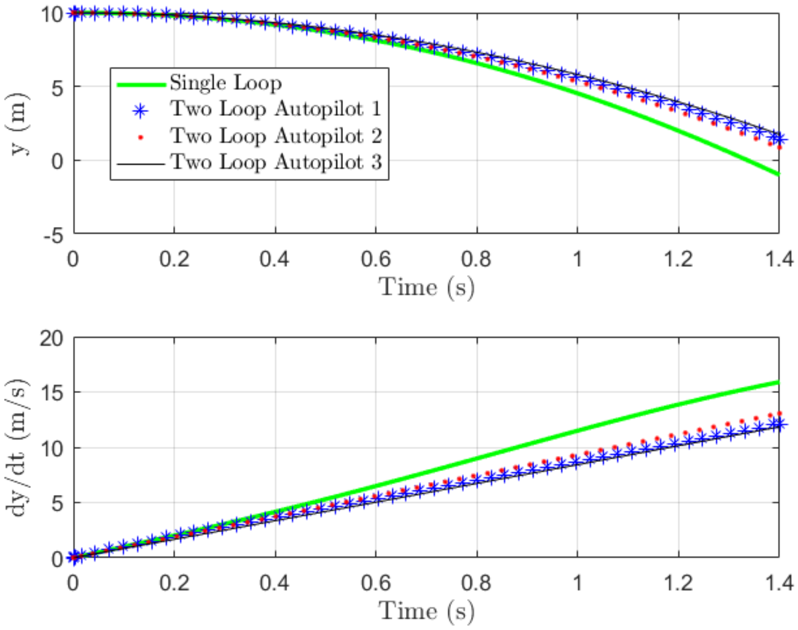


Figure 6.4: Kinematic states 1: Single – input design.

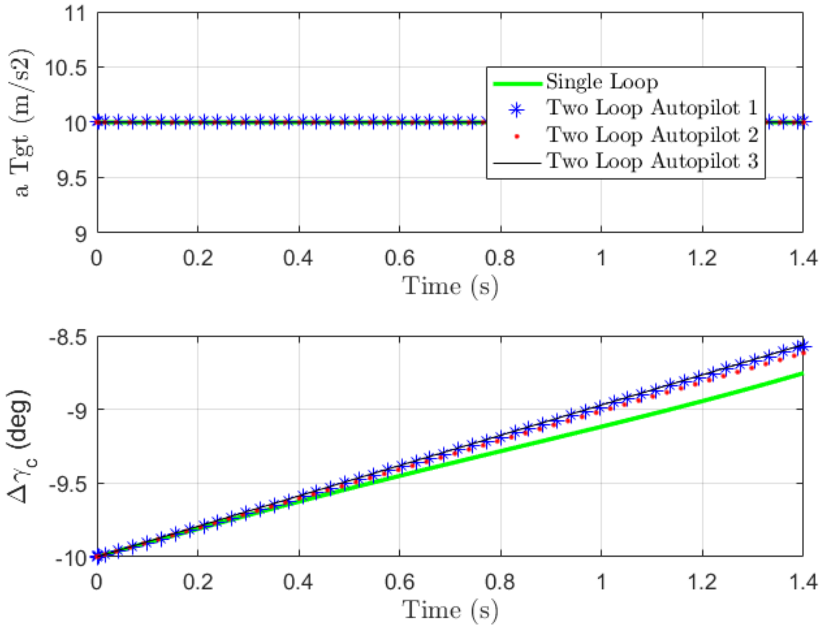


Figure 6.5: Kinematic states 2: Single – input design.

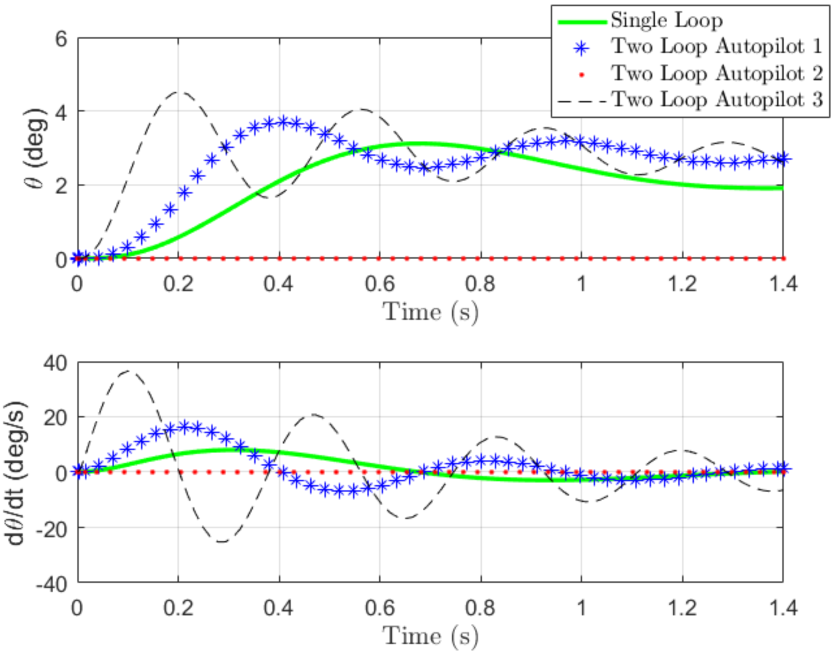


Figure 6.6: Dynamic states: Single – input design.

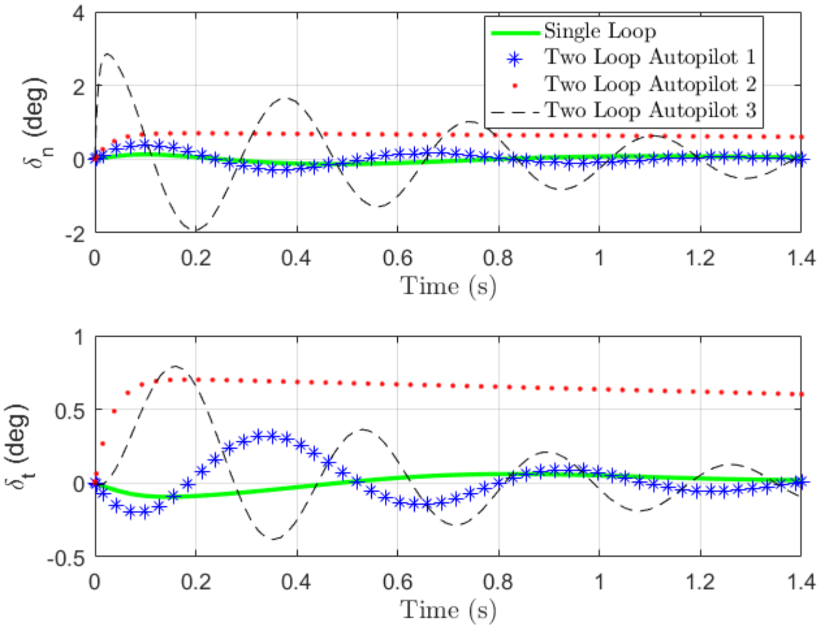


Figure 6.7: Servo states: Single – input design.

Figures 6.8 – 6.10 illustrate the equivalent acceleration commands and their components. The acceleration commands can be rewritten as follows:

$$\delta_i^c = \hat{c}_i \, (\phi_c - \phi) - K_{\dot{\theta}_i} \, \dot{\theta} \tag{6.4}$$

When $\hat{c}_n > \hat{c}_t$ (‘Autopilot 3’) δ_t^c is close to $-K_{\dot{\theta}_t} \, \dot{\theta}$. In the same manner, when $\hat{c}_n < \hat{c}_t$, δ_n^c would be closer to $-K_{\dot{\theta}_n} \, \dot{\theta}$. Hence, the guidance law is enforced to use both controllers even for higher or smaller angles of α.

Moreover, when $\dot{\theta} = 0$ (‘Autopilot 2’), the relation between δ_n^c and δ_t^c is inversely proportional to the ratio given in Equation (6.5). In other words, bigger controller deflections are required for smaller steering moments.

$$\frac{\delta_n^c}{\delta_t^c} = \frac{M_{\delta_t}}{M_{\delta_n}} \tag{6.5}$$

Furthermore, the missile acceleration command is proportional to the missile’s body angle and both controller deflections. Therefore, δ_n^c and δ_t^c in ’Autopilot 2’ must be increased so as to compensate the absence of rotation of the missile (the missile’s body angle is bounded to zero). Note that, in this version of the autopilot, the nose and tail controller deflections are identical since both provide the same moments about the center of gravity.

$$a_{PN} = \frac{T}{m} \, (\theta + \delta_n + \delta_t) \tag{6.6}$$

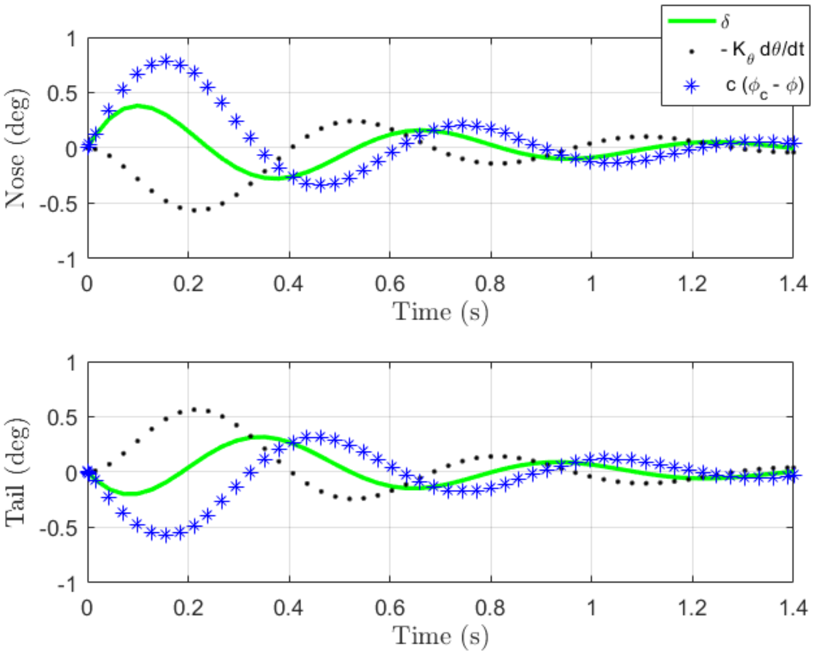


Figure 6.8: Equivalent acceleration commands: Single−input design (Autopilot 1).

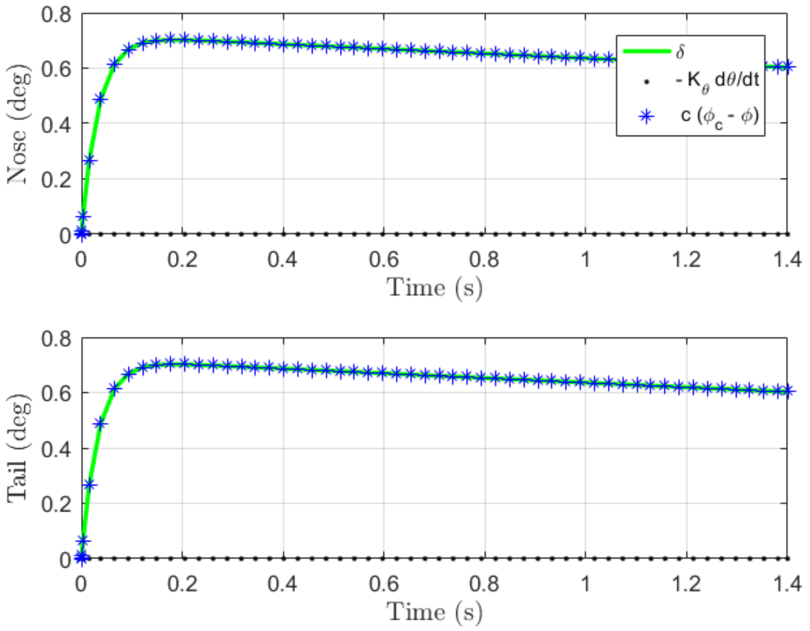


Figure 6.9: Equivalent acceleration commands: Single−input design (Autopilot 2).

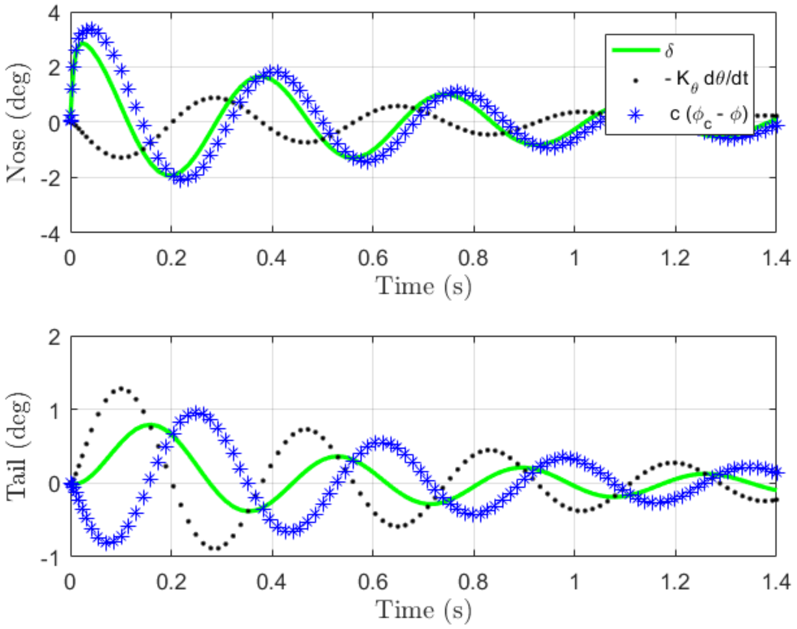


Figure 6.10: Equivalent acceleration commands: Single – input design (Autopilot 3).

6.5 Integrated two – loop multi – input autopilot – guidance law

Figure 6.11 illustrates the Pareto front curve for an integrated two – loop guidance law and an integrated single – loop guidance law with a multi – input block diagram. Again, the Pareto front curve is created for a fixed miss distance penalty (b), a fixed impact angle error penalty (c) and varying the nose control effort penalty (α). Therefore, each point of the curve is a simulation result in terms of the nose and the tail control effort. When $\alpha \to \infty$, the guidance laws are forced to use more tail control effort; and vice versa when $\alpha \to 0$.

It can be appreciated that the results of the integrated two – loop scheme are identical to the ones of the integrated single – scheme. In both cases, there is a full feedback on the guidance loop. However, in the two – loop configuration, the inner autopilot loop is designed individually from the outer guidance loop. Hence, Figure 6.11 demonstrates that the integrated two – loop approach takes into consideration the inner autopilot dynamics due to the full feedback.

Note that the integrated two – loop configuration is preferred over the single – loop one since it ensures the inner stability of the airframe, even if the outer guidance loop is not active.

Additionally, it should be noted that the guidance command issued by the integrated single – loop scheme is directly the deflection command. However, the integrated two – loop guidance law issues an equivalent command, which was given in Equation (5.58). As a result, in order to achieve identical scenario performance and obtain the same value for the quadratic cost function, this equivalent acceleration command must have the same values as the guidance deflection commands generated by the single – loop scheme.

Table 6.3 summarises the design parameters used in the integrated two – loop multi – input scheme.

Gain set	$\mathbf{K}_{\dot{\theta}_n}$	$\mathbf{K}_{\dot{\theta}_t}$	$\hat{\mathbf{c}}_\mathbf{n}$	$\hat{\mathbf{c}}_\mathbf{t}$	**b**	**c**
1	0.04	0.04	0.26	-0.1	0.0001	1

Table 6.3: Integrated two – loop multi – input design parameters.

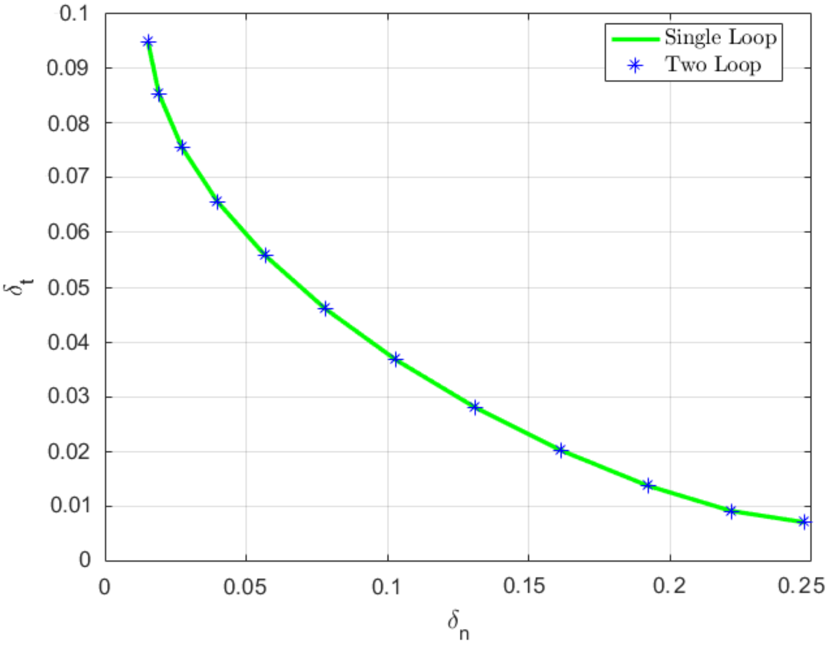


Figure 6.11: Pareto front curves for a perfect interception case: Integrated multi – input designs.

Furthermore, Figures 6.12 – 6.13 showcase the kinematic states. It can be appreciated that the miss distance tends to zero as the relative displacement between the adversaries decreases. The impact angle error also decreases. Recall that the initial values of the scenario parameters were given in Table 5.1. Finally, note that both guidance laws achieve identical results for the kinematic states.

On the other hand, Figures 6.14 – 6.16 illustrate the dynamic states, the servo states and the servo commands (or guidance commands) of both guidance laws.

Concerning the missile's body angle, it can be appreciated that the interceptor had a 2 deg angle of attack right before interception. As expected, it can be seen that the equivalent acceleration commands generated by the integrated two – loop scheme are identical to the deflection commands issued by the integrated single – loop scheme. Note that the result obtained for both servo commands is equal in module.

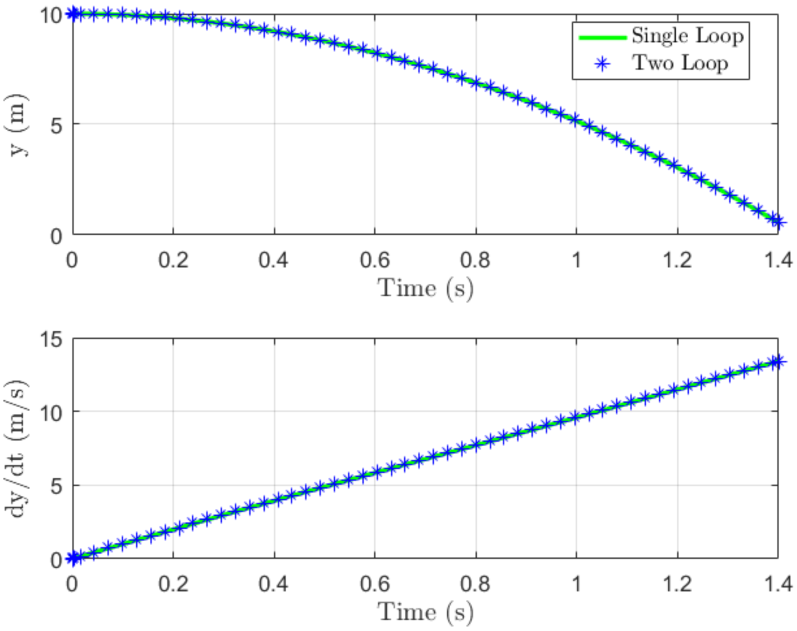


Figure 6.12: Kinematic states 1: Multi – input design.

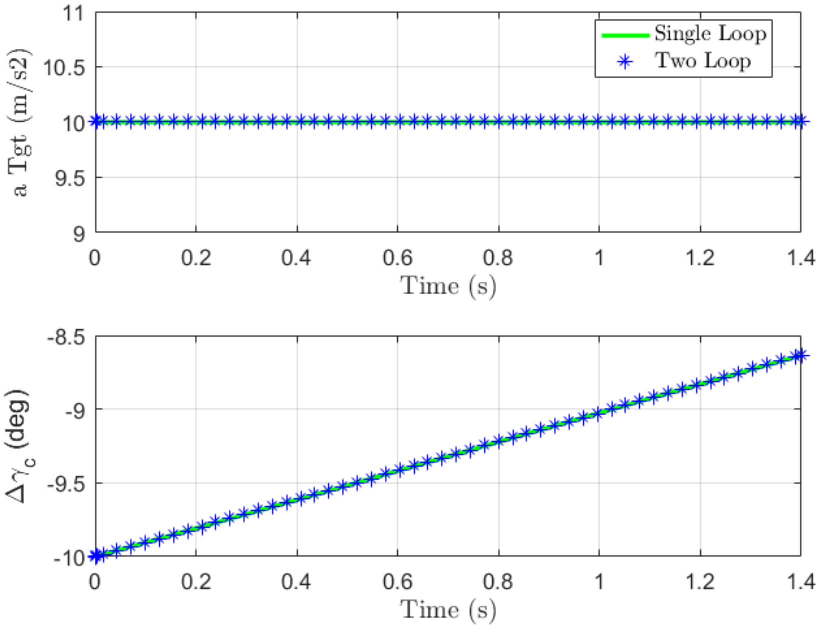


Figure 6.13: Kinematic states 2: Multi−input design.

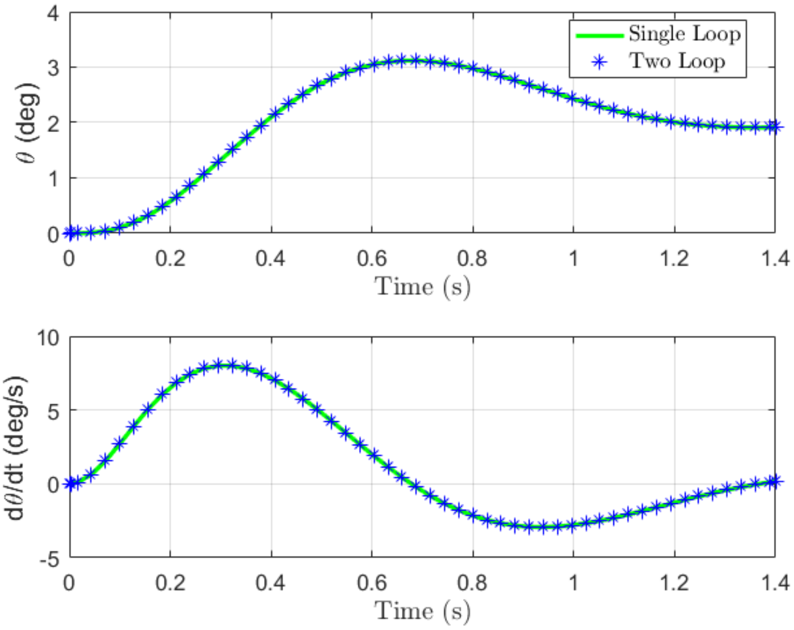


Figure 6.14: Dynamic states: Multi−input design.

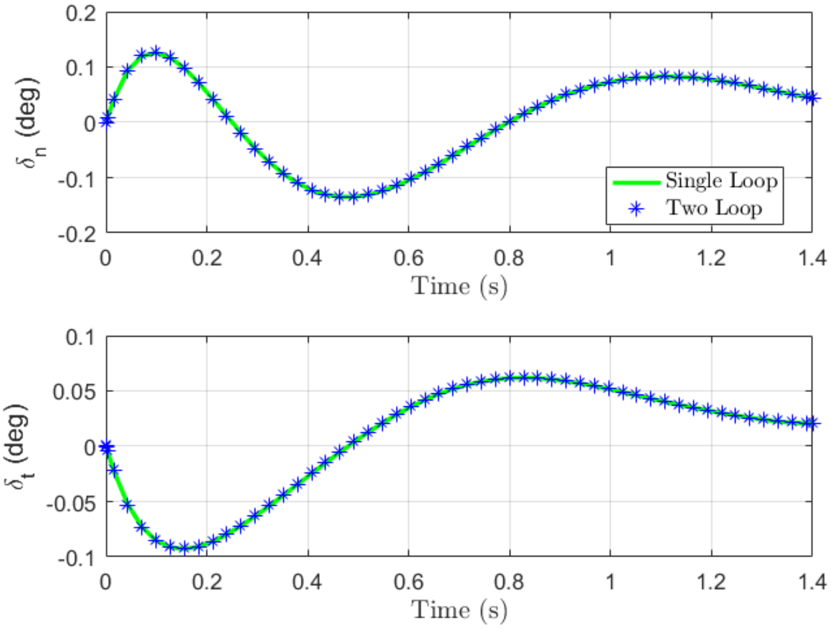


Figure 6.15: Servo states: Multi – input design.

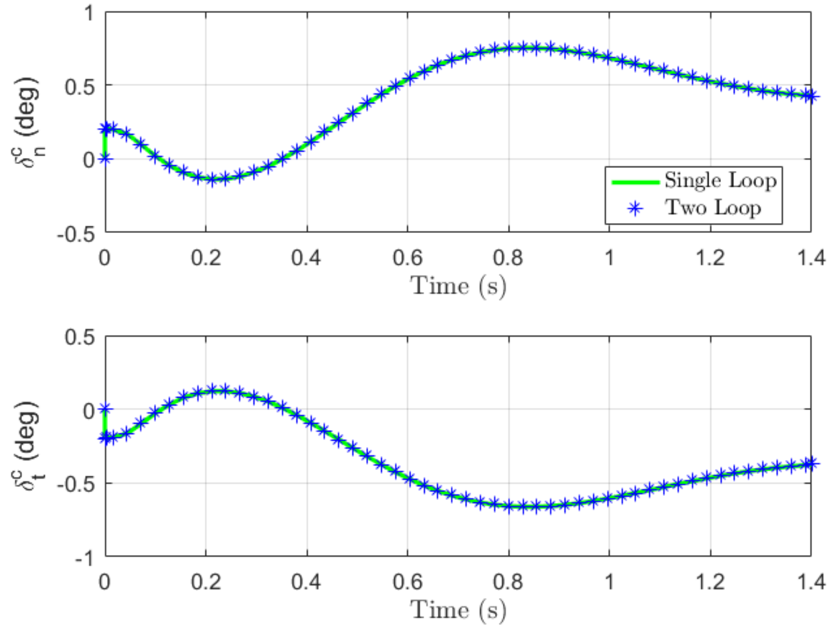


Figure 6.16: Deflection commands: Multi – input design.

Chapter 7

Conclusions and future work

7.1 Conclusions

The main contribution of this thesis was proving and exemplifying that under a linear quadratic formulation the single – loop and two – loop configurations achieve identical performance if and only if the number of guidance commands is equal to the number of controllers. In addition, it was also proven that the performance of the integrated approaches is higher than the one of the traditional separated approach.

In order to address these challenges, three different types of guidance and control schemes were designed under a linear quadratic formulation: a separated two – loop autopilot – guidance design, an integrated single – loop guidance design, and an integrated two – loop autopilot – guidance design. Moreover, in the latter scheme, two possible autopilot block diagrams were examined: a single – input scheme and a multi – input scheme. The guidance laws were illustrated using an exo – atmospheric dual control missile model.

In addition, given that the integrated single – loop guidance law was expected to achieve the best engagement scenario performance subject to the given missile's model, this scheme was used as a benchmark system to evaluate the other guidance laws' performance. Simulations were carried out using the Pareto front concept and sample runs.

On one hand, a traditional separated approach was designed using a decoupled architecture. In order to address this challenge, the analytic feedback solution in Equation (7.1) was used to generate the acceleration command:

$$u = a_P^c = \frac{\Lambda_G}{t_{Go}^2}\left[y + \dot{y}\,t_{Go} + \frac{1}{2}\,a_E\,t_{Go}^2 - a_E\,\tau^2\left(e^{-\left(\frac{t_{Go}}{\tau}\right)} + \left(\frac{t_{Go}}{\tau}\right) - 1\right)\right] \tag{7.1}$$

where Λ_G was the navigation ratio.

This scheme had no feedback available of the target states to the control system. As a result, the separated autopilot – guidance Pareto curve appeared above the integrated one, demonstrating the superiority of the integrated approach. However, a low – order approximation of the autopilot dynamics was assumed in the separated approach, which lead to a less accurate missile's response.

On the other hand, multiple – input single – output integrated guidance laws were designed while minimising the quadratic cost function given in Equation (7.2). These integrated schemes were two – input single – output systems, where the two inputs were the nose and tail control surfaces of the missile and the single output was the terminal cost on the miss distance.

$$\min_{u(t)} J\left(\mathbf{x}(\mathbf{t_0}),\, \mathbf{u}(\mathbf{t_0}),\, \mathbf{t_0}\right) \;=\; \mathbf{x}^T(\mathbf{t_f})\,\mathbf{Q_f}\,\mathbf{x}\,(\mathbf{t_f}) + \int_{t_0}^{t_f} \mathbf{u}^T(\mathbf{t})\,\mathbf{R}\,\mathbf{u}\,(\mathbf{t})\;dt \tag{7.2}$$

Analysing the integrated approach, two types of autopilot – guidance laws were explored: a single – loop and a two – loop configuration. In both cases, there was a full feedback on the guidance loop. However, in the two – loop configuration, the inner autopilot loop was designed individually from the outer guidance loop.

It was proven through simulations using the Pareto front concept that both integrated schemes achieve identical end – game scenario performance if and only if the number of guidance commands is equal to the number of controllers. However, the integrated two – loop configuration is preferred over the single – loop one since it ensures the inner stability of the airframe, whether the outer guidance loop is active or not.

In order to address this challenge, a single – input autopilot block was also implemented for the integrated two – loop configuration, proving that the multi – input autopilot scheme achieved better homing performance than the single – input autopilot configuration. Thereafter, for a MIMO guidance system, a single – input autopilot cannot generate the appropriate commands to control each of the controllers since it loses one degree of freedom. However, by using a multi – input autopilot design, the guidance law generates two accelerations commands, and the autopilot is able to control both nose and tail controllers individually.

7.2 Future work

Deeper research is necessary since this thesis focused on an ideal engagement scenario where the real effects were not taken into consideration. In particular, the guidance and the flight control subsystems were considered to be linear, the missile's speed remained constant, the missile's aerodynamic model was based on constant coefficients, and the aerodynamic saturation was not contemplated.

Therefore, a more realistic approach should be addressed where the engagement scenario is not confined into a plane and linearised about the initial line of sight. Besides, both the missile's body angle (θ) and the control deflections (δ_n and δ_t) were restrained.

The separated guidance law was based on first – order autopilot dynamics, which implied that the airframe response wasn't described accurately and there was a deterioration of the end – game performance. Therefore, it would be mandatory to design a guidance law based on a higher – order dynamics model in order to obtain a better end – game performance when studying the separated autopilot – guidance scheme [11, 33].

Moreover, the aerodynamic cross – coupling control shall be investigated in addition to the missile's drag and its effect on the guidance law. This thesis was carried out considering that the controls acted separately. However, if the aerodynamic coupling between the two channels is considered, the tail response will be affected by the canard downwash, which will also vary with the missile's body angle and the canard deflection.

A further extension could also be to define the weighting matrices ($\mathbf{R}$ and $\mathbf{Q}_f$) or the vectorial control law parameters ($\hat{c}_n$ and $\hat{c}_t$) as functions dependent upon the relative distance between the missile and the target.

On the other hand, no investigation has been carried out regarding the effect of noise, radome errors, and plant uncertainties. Therefore, further research shall be done without assuming perfect knowledge of the target's manoeuvre strategy and taking into consideration the non – linear aerodynamics effects.

Also, only constant manoeuvre has been explored and both outer guidance loops of the separated and integrated approaches depend upon the missile's manoeuvre. Hence, further analysis shall be carried out considering that the target performs a different manoeuvre strategy, for instance like a sinusoidal manoeuvre.

References

[1] Anderson B. D. and Moore, J. B., *Optimal control: Linear quadratic methods.* Dover, Mineola, NY, 2007.

[2] Balakrishnan, S. N. and Donald T. Stansbery, *Analytical guidance laws and integrated guidance autopilot for homing missiles.* IEEE Conference on Control Application, Vancouver, 1993.

[3] Ben – Asher, J. Z. and I. Yaesh, *Advances in missile guidance theory.* American Institute of Aeronautics and Astronautics, 1998.

[4] Bryson, A. E. and Ho, Y. C., *Applied optimal control; optimization, estimation and control.* Blaisdell, Waltham, MA, 1969.

[5] Dancer, M. W., Balakrishnan, S. N. and Ohlmeyer, E. J., *Discussion and analysis of missile igc design.* AIAA Guidance, Navigation and Control Conference, 2008.

[6] Gutman, S., *Superiority of canards in homing missiles.* IEEE Transactions on Aerospace and Electronic Systems, 2003.

[7] Gutman, S., *Applied min – max approach to missile guidance and control*, volume 209. American Institute of Aeronautics and Astronautics, 2005.

[8] Gutman, S. and Goldan, O., *Optimal guidance – A guaranteed miss approach.* AIAA Guidance, Navigation and Control Conference, 2009.

[9] Gutman, S., Goldan, O. and Rubinsky, S., *Guaranteed miss distance in guidance systems with bounded controls and bounded noise.* Journal of Guidance, Control and Dynamics, 2012.

[10] Hu, X. B., Wang, M. and Di Paolo, E., *Calculating complete and exact Pareto Front for multiobjective optimization: A new deterministic approach for discrete problems.* IEEE Transactions on Cybernetics, volume 43, 2013.

[11] Ibarrondo, F. and Sanz – Aranguez, P., *Integrated versus two – loop guidance – autopilot for a dual control missile with high – order aerodynamic model.* Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineers, 2015.

[12] Idan, M., Shima, T. and Golan, O. M., *Integrated sliding mode autopilot – guidance for dual control missiles.* Journal of Guidance, Control and Dynamics, 2005.

[13] Jialing, Z. and Jianying, Y., *Integrated missile guidance and control design with smooth adaptive sliding mode control.* Proceedings of the 34th Chinese Control Conference, 2015.

[14] Levy, M., Shima, T. and S. Gutman, *Integrated single versus two – loop autopilot – guidance design for dual – controlled missiles.* IEEE International Conference on Control and Automation, China, 2013.

[15] Levy, M., Shima, T. and S. Gutman, *Linear quadratic integrated versus separated autopilot – guidance design.* Journal of Guidance, Control and Dynamics, 2013.

[16] Levy, M., Shima, T. and S. Gutman, *Single versus two – loop full state multi – input multi – output missile guidance.* AIAA Guidance, Navigation and Control Conference, 2013.

[17] Menon, P. and Ohlmeyer, E. J., *Integrated design of agile missile guidance and autopilot systems.* Control Engineering Practice, 2001.

[18] Mracek, C. P. and Ridgely, D. B., *Optimal control solution for dual (tail and canard) control missiles.* AIAA Guidance, Navigation and Control Conference, 2006.

[19] Palumbo, N. F., Blauwkamp, R. A. and J. M. Lloyd, *Basic principles of homing guidance.* Johns Hopkins APL Technical Digest, 2010.

[20] Palumbo, N. F. and Jackson, T. D., *Integrated missile guidance and control: A state dependent Riccati differential equation approach.* Proceedings of IEEE International Conference on Control Applications, volume 1, Hawaii 1999.

[21] Palumbo, N. F., Reardon, B. E. and Blauwkamp, R. A., *Integrated guidance and control for homing missiles.* Johns Hopkins APL Technical Digest, 2004.

[22] Park, B. G., T. H. Kim and M. J. Tahk, *Time delay control for integrated missile guidance and control.* International Journal of Aeronautical and Space Science, 2011.

[23] Perlis, S., *Theory of matrices.* Dover Publications, Mineola, New York, 1991.

[24] Ryoo, C. K., Cho, H. and Tahk, M. J., *Closed form solutions of optimal guidance with terminal impact angle constraint.* Proceedings of the IEEE Conference on Control Applications, 2003.

[25] Shaferman, V. and Shima, T., *Linear quadratic guidance laws for imposing a terminal intercept angle.* Journal of Guidance, Control and Dynamics, 2008.

[26] Shima, T., *Intercept angle guidance.* Journal of Guidance, Control and Dynamics, 2011.

[27] Shima, T. and Golan, O. M., *End – game guidance laws for dual – control missiles.* Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineers, 2006.

[28] Shima, T. and Golan, O. M., *Bounded differential games guidance law for dual – controlled missiles.* IEEE Transactions on Control Systems Technology, 2006.

[29] Shima, T. and Golan, O. M., *Linear quadratic differential games guidance law for dual – controlled missiles.* IEEE Transactions on Aerospace and Electronic Systems, 2007.

[30] Shima, T., Idan, M. and Golan, O. M., *Sliding mode control for integrated missile autopilot guidance.* Journal of Guidance, Control and Dynamics, 2006.

[31] Shin, H. S., Hwang, T. W., Tsourdos, A., White, B. A. and Tahk, M. J., *Integrated intercept missile guidance and control with terminal angle constraint.* International congress of the aeronautical sciences, 2008.

[32] Shkolnikov, I., Shtessel, Y. and Lianos, D., *Integrated guidance control system of a homing interceptor: Sliding mode approach.* American Institute of Aeronautics and Astronautics, 2001.

[33] Shtessel, Y. B. and Tournes, C. H., *Integrated higher order sliding mode guidance and autopilot for dual control missiles.* Journal of Guidance, Control and Dynamics, 2009.

[34] Taub, I. and Shima, T., *Intercept angle missile guidance under time varying acceleration bounds.* AIAA Guidance, Navigation and Control Conference, 2012.

[35] Vaddi, S. S., Menon, P. K. and Ohlmeyer, E. J., *Numerical state dependent Riccati equation approach for missile integrated guidance control.* Journal of Guidance, Control and Dynamics, 2009.

[36] Weiss, M. and Shima, T., *Minimum effort pursuit/evasion guidance with specified miss distance.* Journal of Guidance, Control and Dynamics, 2016.

[37] Xin, M, Balakrishnan, S. N. and Ohlmeyer, E. J., *Integrated guidance and control of missile with theta – D method.* IEEE Transactions on Control Systems Technology, 2006.

[38] Zarchan, P., *Tactical and strategic missile guidance*, volume 176. American Institute of Aeronautics and Astronautics, Reston, VA, 3rd edition, 1997.

[39] Zarchan, P., *Tactical and strategic missile guidance*, volume 219. American Institute of Aeronautics and Astronautics, Reston, VA, 5th edition, 2007.

[40] Ibarrondo, F., *Optimization of the integrated guidance and control for a dual aerodynamic control missile.* Universidad Politecnica de Madrid, 2015.

Appendix A

Theorem proof

A.1 Equivalent problem formulation

Note that some results from the theory of matrices explained in [23] are used to proof Theorem 1.

Definition 2. *An n x s matrix* $\boldsymbol{A}$ *is said to have* $\boldsymbol{B}$ *as a left inverse if* $\boldsymbol{BA} = \boldsymbol{I}$*. In this case* $\boldsymbol{I}$ *must be s x s, and* $\boldsymbol{B}$ *then must be s x n. Similarly, if there is a matrix* $\boldsymbol{C}$ *such that* $\boldsymbol{AC} = \boldsymbol{I}$*, then* $\boldsymbol{C}$ *is called a right inverse of* $\boldsymbol{A}$*. The matrix* $\boldsymbol{I}$ *in this case must be n x n and* $\boldsymbol{C}$ *must be s x n.*

Lemma 1. *If* $\boldsymbol{A}$ *has both a left inverse* $\boldsymbol{B}$ *and a right inverse* $\boldsymbol{C}$*,* $\boldsymbol{A}$ *is non − singular and* $\boldsymbol{B} = \boldsymbol{C} = \boldsymbol{A}^{-1}$*.*

Recalling the single − loop and the two − loop optimisation problems:

$$J \;=\; \mathbf{x}^T(t_f)\,\mathbf{Q_f}\,\mathbf{x}(t_f) + \int_{t_0}^{t_f} \mathbf{u}^T\,\mathbf{R}\,\mathbf{u}\,dt \tag{A.1}$$

$$\dot{\mathbf{x}} \;=\; \mathbf{F}\,\mathbf{x} + \mathbf{G}\,\tilde{\mathbf{u}} \tag{A.2}$$

where

$$\mathbf{u} \;=\; \tilde{\mathbf{u}} \tag{A.3}$$

$$\mathbf{x} \;=\; \begin{bmatrix} \mathbf{x_G^T} & \mathbf{x_D^T} & \mathbf{x_S^T} \end{bmatrix}^{\mathbf{T}} \tag{A.4}$$

$$\mathbf{F} \;=\; \begin{bmatrix} \mathbf{A_{GG}} & \mathbf{A_{GD}} & \mathbf{A_{GS}} \\ [\mathbf{0}] & \mathbf{A_{DD}} & \mathbf{A_{DS}} \\ [\mathbf{0}] & [\mathbf{0}] & \mathbf{A_S} \end{bmatrix} \tag{A.5}$$

$$\mathbf{G} \;=\; \begin{bmatrix} [\mathbf{0}] \\ \mathbf{B_S} \end{bmatrix} \tag{A.6}$$

$$\begin{aligned} J &= \mathbf{x}^T(t_f)\,\mathbf{Q_f}\,\mathbf{x}(t_f) + \int_t^{t_f} \tilde{\mathbf{u}}^T(\tau)\,\mathbf{R}\,\tilde{\mathbf{u}}(\tau)\;d\tau \;=\; \mathbf{x}^T(t_f)\,\mathbf{Q_f}\,\mathbf{x}(t_f) \\ &+ \int_t^{t_f} \left(\mathbf{u}^T\,\mathbf{R_A}\,\mathbf{u} + 2\,\mathbf{x}^T\,\mathbf{S_A}\,\mathbf{u} + \mathbf{x}^T\,\mathbf{Q_A}\,\mathbf{x}\right)\,(\tau)\;d\tau \end{aligned} \tag{A.7}$$

$$\dot{\mathbf{x}} = \mathbf{F_A}\,\mathbf{x} + \mathbf{G_A}\,\mathbf{u} \tag{A.8}$$

where $\tilde{\mathbf{u}}$ has been previously defined in (5.58), (5.59) (5.60), (5.61) and (5.62).

Recalling the state space matrices $\mathbf{F_A}$ and $\mathbf{G_A}$:

$$\mathbf{F_A} = \begin{bmatrix} \mathbf{A_{GG}} & \mathbf{A_{GD}} & \mathbf{A_{GS}} \\ [\mathbf{0}] & \mathbf{A_{DD}} & \mathbf{A_{DS}} \\ [\mathbf{0}] & -\mathbf{B_S}\,\hat{\mathbf{C}}\,\mathbf{k_D} & \mathbf{A_S} - \mathbf{B_S}\,\hat{\mathbf{C}}\,\mathbf{k_S} \end{bmatrix} \tag{A.9}$$

$$\mathbf{G_A} = \begin{bmatrix} [\mathbf{0}] \\ \mathbf{B_S}\,\hat{\mathbf{C}} \end{bmatrix} \tag{A.10}$$

With no loss of generality, it is assumed $\tilde{\mathbf{R}} = \mathbf{R} = \mathbf{I_{m\,x\,m}}$. The integrand expression of the running cost in (4.11) can be rewritten as follows:

$$\begin{aligned} \tilde{\mathbf{u}}^\mathbf{T}\,\tilde{\mathbf{u}} &= \mathbf{u}^\mathbf{T}\,\hat{\mathbf{C}}^\mathbf{T}\,\hat{\mathbf{C}}\,\mathbf{u} + 2\left[\mathbf{x_G^T} \;\; \mathbf{x_D^T} \;\; \mathbf{x_S^T}\right] \begin{bmatrix} 0 \\ -\mathbf{k_D^T} \\ -\mathbf{k_S^T} \end{bmatrix} \hat{\mathbf{C}}^\mathbf{T}\,\hat{\mathbf{C}}\,\mathbf{u} \\ &+ \left[\mathbf{x_G^T} \;\; \mathbf{x_D^T} \;\; \mathbf{x_S^T}\right] \begin{bmatrix} 0 & 0 & 0 \\ 0 & \mathbf{k_D^T}\,\hat{\mathbf{C}}^\mathbf{T}\,\hat{\mathbf{C}}\,\mathbf{k_D} & \mathbf{k_D^T}\,\hat{\mathbf{C}}^\mathbf{T}\,\hat{\mathbf{C}}\,\mathbf{k_S} \\ 0 & \mathbf{k_S^T}\,\hat{\mathbf{C}}^\mathbf{T}\,\hat{\mathbf{C}}\,\mathbf{k_D} & \mathbf{k_S^T}\,\hat{\mathbf{C}}^\mathbf{T}\,\hat{\mathbf{C}}\,\mathbf{k_S} \end{bmatrix} \begin{bmatrix} \mathbf{x_G^T} \\ \mathbf{x_D^T} \\ \mathbf{x_S^T} \end{bmatrix} \end{aligned} \tag{A.11}$$

Thus:

$$\tilde{\mathbf{u}}^\mathbf{T}\,\tilde{\mathbf{u}} = \mathbf{u}^\mathbf{T}\,\hat{\mathbf{R}}\,\mathbf{u} + 2\,\mathbf{x}^\mathbf{T}\,\hat{\mathbf{S}}\,\mathbf{u} + \mathbf{x}^\mathbf{T}\,\hat{\mathbf{Q}}\,\mathbf{x} \tag{A.12}$$

Anderson, B. and Moore, J. [1] introduce the solution to the extended regulator problem considering that the performance index contains cross product terms. However, the aim is to reduce the problem to a standard regulator problem by defining:

$$\mathbf{u_0} = \mathbf{u} + \mathbf{R_A^{-1}}\,\mathbf{S_A^T}\,\mathbf{x} \tag{A.13}$$

Note that the inverse of $\mathbf{R_A}$ exists since $m_G \leq m$.

Using the following identity:

$$\begin{aligned} \mathbf{u^T}\,\mathbf{R}\,\mathbf{u} + 2\,\mathbf{x^T}\,\mathbf{S}\,\mathbf{u} + \mathbf{x^T}\,\mathbf{Q}\,\mathbf{x} &= \left(\mathbf{u} + \mathbf{R^{-1}}\,\mathbf{S^T}\,\mathbf{x}\right)^{\mathbf{T}}\,\mathbf{R}\,\left(\mathbf{u} + \mathbf{R^{-1}}\,\mathbf{S^T}\,\mathbf{x}\right) \\ &+ \ \mathbf{x^T}\,\left(\mathbf{Q} - \mathbf{S}\,\mathbf{R^{-1}}\,\mathbf{S^T}\right)\,\mathbf{x} \end{aligned} \tag{A.14}$$

Substituting (A.4) and (A.13) in (A.8):

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{F_A}\,\mathbf{x} + \mathbf{G_A}\,\mathbf{u} = \mathbf{F_A}\,\mathbf{x} + \mathbf{G_A}\,\left(\mathbf{u_0} - \mathbf{R_A^{-1}}\,\mathbf{S_A^T}\,\mathbf{x}\right) \\ &= \left(\mathbf{F_A} - \mathbf{G_A}\,\mathbf{R_A^{-1}}\,\mathbf{S_A^T}\right)\,\mathbf{x} + \mathbf{G_A}\,\mathbf{u_0} = \bar{\mathbf{F}}\,\mathbf{x} + \mathbf{G_A}\,\mathbf{u_0} \end{aligned} \tag{A.15}$$

And rearranging the cost function:

$$\begin{aligned} J &= \mathbf{x^T}(t_f)\,\mathbf{Q_f}\,\mathbf{x}(t_f) + \int_{t_0}^{t_f} \left\{\mathbf{u_0^T}\,\mathbf{R_A}\,\mathbf{u_0} + \mathbf{x^T}\,\left(\mathbf{Q_A} - \mathbf{S_A}\,\mathbf{R_A^{-1}}\,\mathbf{S_A^T}\right)\,\mathbf{x}\right\}\,d\tau \\ &= \mathbf{x^T}(t_f)\,\mathbf{Q_f}\,\mathbf{x}(t_f) + \int_{t_0}^{t_f} \left\{\mathbf{u_0^T}\,\mathbf{R_A}\,\mathbf{u_0} + \mathbf{x^T}\,\tilde{\mathbf{Q}}\,\mathbf{x}\right\}\,d\tau \end{aligned} \tag{A.16}$$

$$\tilde{\mathbf{Q}} = \mathbf{Q_A} - \mathbf{S_A}\,\mathbf{R_A^{-1}}\,\mathbf{S_A^T} \tag{A.17}$$

where

$$\mathbf{Q_A} = \begin{bmatrix} [\mathbf{0}] & [\mathbf{0}] \\ [\mathbf{0}] & \hat{\mathbf{Q}} \end{bmatrix} \tag{A.18}$$

$$\mathbf{S_A} = \begin{bmatrix} [\mathbf{0}] \\ \hat{\mathbf{S}} \end{bmatrix} \tag{A.19}$$

$$\mathbf{R_A} = \tilde{\mathbf{R}} = \hat{\mathbf{C}}^{\mathbf{T}}\,\hat{\mathbf{C}} \tag{A.20}$$

$$\hat{\mathbf{Q}} = \begin{bmatrix} \mathbf{k_D^T}\,\hat{\mathbf{C}}^{\mathbf{T}}\,\hat{\mathbf{C}}\,\mathbf{k_D} & \mathbf{k_D^T}\,\hat{\mathbf{C}}^{\mathbf{T}}\,\hat{\mathbf{C}}\,\mathbf{k_S} \\ \mathbf{k_S^T}\,\hat{\mathbf{C}}^{\mathbf{T}}\,\hat{\mathbf{C}}\,\mathbf{k_D} & \mathbf{k_S^T}\,\hat{\mathbf{C}}^{\mathbf{T}}\,\hat{\mathbf{C}}\,\mathbf{k_S} \end{bmatrix} \tag{A.21}$$

$$\hat{\mathbf{S}} = \begin{bmatrix} -\mathbf{k_D^T} \\ -\mathbf{k_S^T} \end{bmatrix}\,\hat{\mathbf{C}}^{\mathbf{T}}\,\hat{\mathbf{C}} \tag{A.22}$$

The equivalent problem (A.15) – (A.22) can be solved as a finite time regulator problem considering that $\mathbf{R_A}$ is positive definite and $\tilde{\mathbf{Q}}$ satisfies the non−negative condition:

$$\tilde{\mathbf{Q}} = \mathbf{Q_A} - \mathbf{S_A}\,\mathbf{R_A^{-1}}\,\mathbf{S_A^T} \geq 0 \tag{A.23}$$

The optimal controller is given by:

$$\mathbf{u_0^*}\left(\mathbf{t}\right) \;=\; -\mathbf{R_A^{-1}}\,\mathbf{G_A^T}\,\mathbf{P}\,\mathbf{x} \tag{A.24}$$

And the differential Ricatti equation:

$$\begin{aligned} -\dot{\mathbf{P}} \;&=\; \mathbf{P}\left(\mathbf{F_A} - \mathbf{G_A}\,\mathbf{R_A^{-1}}\,\mathbf{S_A^T}\right) + \left(\mathbf{F_A^T} - \mathbf{S_A}\,\mathbf{R_A^{-1}}\,\mathbf{G_A^T}\right)\mathbf{P} \\ &-\; \mathbf{P}\,\mathbf{G_A}\,\mathbf{R_A^{-1}}\,\mathbf{G_A^T}\,\mathbf{P} + \mathbf{Q_A} - \mathbf{S_A}\,\mathbf{R_A^{-1}}\,\mathbf{S_A^T} \end{aligned} \tag{A.25}$$

$$-\dot{\mathbf{P}} \;=\; \mathbf{P}\,\bar{\mathbf{F}} + \bar{\mathbf{F}}^{\mathbf{T}}\,\mathbf{P} - \mathbf{P}\,\mathbf{G_A}\,\mathbf{R_A^{-1}}\,\mathbf{G_A^T}\,\mathbf{P} + \tilde{\mathbf{Q}} \tag{A.26}$$

$$\mathbf{P}\left(\mathbf{t_f}\right) \;=\; \mathbf{Q_f} \tag{A.27}$$

The previously defined constraint (A.23) has to be verified:

$$\tilde{\mathbf{Q}} \;=\; \mathbf{Q_A} - \mathbf{S_A}\,\mathbf{R_A^{-1}}\,\mathbf{S_A^T} \;=\; [\mathbf{0}] \tag{A.28}$$

Recall that for any non−singular matrices $\mathbf{A}$, $\mathbf{B} \in R^{\,k\,x\,k}$:

$$\left(\mathbf{A}\,\mathbf{B}\,\right)^{-\mathbf{1}} \;=\; \mathbf{B}^{-\mathbf{1}}\,\mathbf{A}^{-\mathbf{1}} \tag{A.29}$$

Therefore, if $\hat{\mathbf{C}}$ is non−singular, it can be inverted.

The theorem is valid for SISO systems, meaning it has a scalar controller:

$$\tilde{u} \;=\; \hat{\mathbf{C}}\left(u - \begin{bmatrix}\mathbf{k_D} & \mathbf{k_S}\end{bmatrix}\begin{bmatrix}\mathbf{x_D}\\ \mathbf{x_S}\end{bmatrix}\right) \tag{A.30}$$

A.2 Proof of Theorem 1

Proof of Theorem 1. *Sufficient condition.* Substituting the terms $\mathbf{F_A}$, $\mathbf{G_A}$, $\mathbf{R_A}$ and $\mathbf{S_A}$ in $\tilde{\mathbf{F}}$ and $\mathbf{G_A}\,\mathbf{R_A^{-1}}\,\mathbf{G_A^T}$:

$$\begin{aligned} \tilde{\mathbf{F}} \;&=\; \mathbf{F_A} - \mathbf{G_A}\,\mathbf{R_A^{-1}}\,\mathbf{S_A^T} = \begin{bmatrix} \mathbf{A_{GG}} & \mathbf{A_{GD}} & \mathbf{A_{GS}} \\ [0] & \mathbf{A_{DD}} & \mathbf{A_{DS}} \\ [0] & -\mathbf{B_S}\,\hat{\mathbf{C}}\,\mathbf{k_D} & \mathbf{A_S} - \mathbf{B_S}\,\hat{\mathbf{C}}\,\mathbf{k_S} \end{bmatrix} \\ &-\; \begin{bmatrix} [0] \\ [0] \\ \mathbf{B_S}\,\hat{\mathbf{C}} \end{bmatrix}\left(\hat{\mathbf{C}}^{\mathbf{T}}\,\hat{\mathbf{C}}\right)^{-1}\left(\hat{\mathbf{C}}^{\mathbf{T}}\,\hat{\mathbf{C}}\right)\begin{bmatrix}[\mathbf{0}] & -\mathbf{k_D} & -\mathbf{k_S}\end{bmatrix} \;=\; \mathbf{F} \end{aligned} \tag{A.31}$$

where

$$\mathbf{F} = \begin{bmatrix} \mathbf{A_{GG}} & \mathbf{A_{GD}} & \mathbf{A_{GS}} \\ [\mathbf{0}] & \mathbf{A_{DD}} & \mathbf{A_{DS}} \\ [\mathbf{0}] & [\mathbf{0}] & \mathbf{A_S} \end{bmatrix} \tag{A.32}$$

$$\begin{aligned} \mathbf{G_A}\,\mathbf{R_A^{-1}}\,\mathbf{G_A^T} &= \begin{bmatrix} [\mathbf{0}] \\ \mathbf{B_S}\,\hat{\mathbf{C}} \end{bmatrix} \left(\hat{\mathbf{C}}^{\mathbf{T}}\,\hat{\mathbf{C}}\right)^{-1} \begin{bmatrix} [\mathbf{0}] & \hat{\mathbf{C}}^{\mathbf{T}}\mathbf{B_S^T} \end{bmatrix} && \text{(A.33)} \\ &= \begin{bmatrix} [\mathbf{0}] & [\mathbf{0}] \\ [\mathbf{0}] & \mathbf{B_S}\,\hat{\mathbf{C}}\,\hat{\mathbf{C}}^{-1}\,\hat{\mathbf{C}}^{-\mathbf{T}}\,\hat{\mathbf{C}}^{\mathbf{T}}\,\mathbf{B_S^T} \end{bmatrix} = \mathbf{G}\,\mathbf{G^T} && \text{(A.34)} \end{aligned}$$

since

$$\hat{\mathbf{C}}\,\hat{\mathbf{C}}^{-1}\,\hat{\mathbf{C}}^{-\mathbf{T}}\,\hat{\mathbf{C}}^{\mathbf{T}} = \mathbf{I} \tag{A.35}$$

Concluding, the Riccati equations solved in the single – loop case and in the two – loop case are identical. Therefore, the solution, $\mathbf{P}\,(\mathbf{t})$, will also be identical. The optimal controller of the two – loop autopilot – guidance law is obtained by substituting (A.24) in (A.13):

$$\mathbf{u}^{*} = -\mathbf{R_A^{-1}}\,\left[\,\mathbf{G_A^T}\,\mathbf{P} + \mathbf{S_A^T}\,\right]\mathbf{x} \tag{A.36}$$

When expressing $\tilde{\mathbf{u}}$ in terms of $\mathbf{R_A}$ and $\mathbf{S_A}$:

$$\tilde{\mathbf{u}} = \hat{\mathbf{C}}\,\left(\mathbf{u} + \mathbf{R_A^{-1}}\,\mathbf{S_A^T}\,\mathbf{x}\right) \tag{A.37}$$

And substituting (A.36) in (A.37):

$$\tilde{\mathbf{u}} = -\hat{\mathbf{C}}\,\mathbf{R_A^{-1}}\,\mathbf{G_A^T}\,\mathbf{P}\,\mathbf{x} \tag{A.38}$$

Substituting (A.38) in the running cost's integrand of (A.7) and the terms $\mathbf{G_A}$ and $\mathbf{R_A}$:

$$\begin{aligned} \tilde{\mathbf{u}}^{\mathbf{T}}\,\tilde{\mathbf{R}}\,\tilde{\mathbf{u}} &= \mathbf{x^T}\,\mathbf{P}\,\mathbf{G_A}\,\mathbf{R_A^{-1}}\,\hat{\mathbf{C}}^{\mathbf{T}}\,\hat{\mathbf{C}}\,\mathbf{R_A^{-1}}\,\mathbf{G_A^T}\,\mathbf{P}\,\mathbf{x} \\ &= \mathbf{x^T}\,\mathbf{P}\,\mathbf{G_A}\,\left(\hat{\mathbf{C}}^{\mathbf{T}}\,\hat{\mathbf{C}}\right)^{-1}\,\hat{\mathbf{C}}^{\mathbf{T}}\,\hat{\mathbf{C}}\,\left(\hat{\mathbf{C}}^{\mathbf{T}}\,\hat{\mathbf{C}}\right)^{-1}\,\mathbf{G_A^T}\,\mathbf{P}\,\mathbf{x} \\ &= \mathbf{x^T}\,\mathbf{P}\,\begin{bmatrix} [\mathbf{0}] \\ \mathbf{B_S} \end{bmatrix}\,\hat{\mathbf{C}}\,\left(\hat{\mathbf{C}}^{\mathbf{T}}\,\hat{\mathbf{C}}\right)^{-1}\,\hat{\mathbf{C}}^{\mathbf{T}}\,\begin{bmatrix} [\mathbf{0}] & \mathbf{B_S^T} \end{bmatrix}\,\mathbf{P}\,\mathbf{x} \\ &= \mathbf{x^T}\,\mathbf{P}\,\begin{bmatrix} [\mathbf{0}] \\ \mathbf{B_S} \end{bmatrix}\,\hat{\mathbf{C}}\,\hat{\mathbf{C}}^{-1}\,\hat{\mathbf{C}}^{-\mathbf{T}}\,\hat{\mathbf{C}}^{\mathbf{T}}\,\begin{bmatrix} [\mathbf{0}] & \mathbf{B_S^T} \end{bmatrix}\,\mathbf{P}\,\mathbf{x} \\ &= \mathbf{x^T}\,\mathbf{P}\,\begin{bmatrix} [\mathbf{0}] \\ \mathbf{B_S} \end{bmatrix}\,\begin{bmatrix} [\mathbf{0}] & \mathbf{B_S^T} \end{bmatrix}\,\mathbf{P}\,\mathbf{x} && \text{(A.39)} \end{aligned}$$

From (A.39) it has been proved that the single – loop guidance law is identical to the two – loop autopilot – guidance law. Consequently, the cost function of both guidance laws reaches the same minimum value.

Substituting (A.36) in (A.15):

$$\mathbf{\dot{x}} \,=\, \mathbf{F_A\,x + G_A\,u} \,=\, \left[\mathbf{F_A - G_A\,R_A^{-1}\,S_A^T - G_A\,R_A^{-1}\,G_A^T\,P}\right]\,\mathbf{x} \tag{A.40}$$

And substituting the terms $\mathbf{G_A}$, $\mathbf{R_A}$ and $\mathbf{\tilde{R}}$:

$$\mathbf{\dot{x}} \,=\, \mathbf{F_A\,x + G_A\,u} = \left[\mathbf{F - G\,G^T\,P}\right]\,\mathbf{x} \tag{A.41}$$

where

$$\mathbf{F_0} \,=\, \mathbf{F - G\,G^T\,P} \tag{A.42}$$

The differential set of equations (A.41) is identical to the differential set of equations obtained in the single loop case. Therefore, the states will be exactly identical for both cases.

Proof of Theorem 1. *Necessary condition.* It will be shown that if the single – loop and the two – loop designs are equivalent, $\mathbf{\hat{C}}$ has to be non – singular. If the single – loop and the two – loop designs are equivalent, the closed – loop equations and the cost function have to be identical.

The closed – loop equation of the single – loop case is:

$$\mathbf{\dot{x}} \,=\, \left[\mathbf{F - G\,G^T\,P_{1L}}\right]\,\mathbf{x} \tag{A.43}$$

where **1L** stands for the single – loop design.

$$\mathbf{F_{1L}} \,=\, \mathbf{F - G\,G^T\,P_{1L}} \tag{A.44}$$

The closed – loop equation of the two – loop case is obtained by substituting (A.24) in (A.15):

$$\mathbf{\dot{x}} \,=\, \left(\mathbf{F_A - G_A\,R_A^{-1}\,S_A^T}\right)\,\mathbf{x} \,+\, \mathbf{G_A\,u_0} \,=\, \left[\mathbf{\tilde{F} - G_A\,R_A^{-1}\,G_A^T\,P_{2L}}\right]\,\mathbf{x\,(t)} \tag{A.45}$$

where **2L** stands for the two – loop design.

$$\mathbf{F_{2L}} \,=\, \mathbf{\tilde{F} - G_A\,R_A^{-1}\,G_A^T\,P_{2L}} \tag{A.46}$$

The solutions of both differential set of equations (A.43) and (A.45) will be identical, considering the same initial conditions, if and only if $\mathbf{F_{1L}} \,=\, \mathbf{F_{2L}}$.

$\mathbf{\tilde{F}} \,=\, \mathbf{F}$ has been proven by substituting the terms $\mathbf{F_A}$, $\mathbf{G_A}$, $\mathbf{R_A}$ and $\mathbf{S_A}$ in $\mathbf{\tilde{F}}$ in

(A.31). Therefore, it is left to proof that the solutions of the Riccati equation are equivalent $\mathbf{P_{1L}} = \mathbf{P_{2L}}$ and that $\mathbf{G_A}\,\mathbf{R_A^{-1}}\,\mathbf{G_A^T} = \mathbf{G}\,\mathbf{G^T}$.

Observing the 2 Riccati equations, is it clear that $\mathbf{P_{1L}} = \mathbf{P_{2L}}$ if $\mathbf{G_A}\,\mathbf{R_A^{-1}}\,\mathbf{G_A^T} = \mathbf{G}\,\mathbf{G^T}$, $\tilde{\mathbf{F}} = \mathbf{F}$ and $\tilde{\mathbf{Q}} = [\mathbf{0}]$. Since $\tilde{\mathbf{Q}} = [\mathbf{0}]$ was settled in (A.28), it is left to proof that $\mathbf{G_A}\,\mathbf{R_A^{-1}}\,\mathbf{G_A^T} = \mathbf{G}\,\mathbf{G^T}$.

From (A.34) and aiming to satisfy $\mathbf{G_A}\,\mathbf{R_A^{-1}}\,\mathbf{G_A^T} = \mathbf{G}\,\mathbf{G^T}$, it is required that:

$$\hat{\mathbf{C}}\left(\hat{\mathbf{C}}^{\mathbf{T}}\,\hat{\mathbf{C}}\right)^{-1}\hat{\mathbf{C}}^{\mathbf{T}} = \mathbf{I} \tag{A.47}$$

Note that this condition has to be valid to obtain the cost functions in both designs.

If (A.47) is valid, $\hat{\mathbf{C}}\,\mathbf{A} = \mathbf{I}$ where $\mathbf{A}$ is the right inverse of $\hat{\mathbf{C}}$. On the other hand, the left inverse of $\hat{\mathbf{C}}$ exists since $m_G \leq m$. Therefore, $\hat{\mathbf{C}}$ has both a right and a left inverse. In consequence, from Lemma 1, $\hat{\mathbf{C}}$ is non−singular.

Appendix B

Matlab® code

The Appendix B.1 encloses a Matlab script with the scenario parameters values. Besides, Appendices B.2, B.3 and B.4 include the Matlab scripts used to implement the classical autopilot – guidance design, the integrated two – loop single – input autopilot – guidance design, and the integrated multi – input autopilot – guidance design, respectively. In the latter, both the single – loop and the two – loop schemes are included.

Additionally, Appendices B.5 and B.6 include the functions used to calculate the integrated autopilot – guidance law and the separated two – loop autopilot – guidance law, respectively. In the latter, the guidance law is presented in the form of a closed – solution. Finally, Appendices B.7 and B.8 enclose the functions used to obtain the solution to the differential Riccati equation, and the minimum effort law gains required to implement the separated autopilot – guidance law.

B.1 Scenario parameters

```
% Name: Maria Morante Soria
% File: Scenario_Parameters.m

% close all;
% clc;
% clear all;

% Scenario Parameters Values

tau_n = 0.1; % Units [sec]
tau_t = 0.1; % Units [sec]

y_0 = 10; % Units [m]

Gamma_c = 10; % Units [deg]

Gamma_Msl_0 = 0; % Units [deg]
Gamma_Tgt_0 = 0; % Units [deg]
Lambda_0    = 0; % Units [deg]

Tm = 120; % Units [m/sec^2]

a_Tgt   = 10; % Units [m/sec^2]
a_Tgt_N = a_Tgt*cos(Gamma_Tgt_0 + Lambda_0); % Units [m/sec^2]

V_Msl = 800; % Units [m/sec]
V_Tgt = 300; % Units [m/sec]

MDelta_n = 200; % Units [1/sec^2]
MDelta_t = 200; % Units [1/sec^2]

% Parameters Interception

alpha = 1;

a = 0.0003;
b = 1;

% Conversion Degrees to Radians

Deg2Rad = pi/180;
Rad2Deg = 180/pi;

Gamma_c_Rad = Gamma_c*Deg2Rad; % Units [rad]

% Initial conditions State Space model Simulink

x_Initial  = [y_0; 0; a_Tgt_N; -Gamma_c_Rad; 0; 0; 0; 0];
x_Initial2 = [0; 0; 0; 0];
```

B.2 Separated architecture

```
% Name: Maria Morante Soria
% File: MinimumEffort.m

% Simulation - Non Ideal Missile & Ideal Target
% Step in Target Acceleration
% Minimum Effort Law

close all;
clc; clear all;

% Parameters Value

g   = 9.81;
dt  = 0.01;
dtg = dt;

% Pursuit Evasion Parameters

t_Final   = 2;
V_Closing = 600;

g_Max   = 15; % [g's]
ncg_Max = g_Max*g; % [m/sec^2]

tau   = 0.1;
tau_D = 0.1;

% State Space Representation
% x = [y_Tgt2Msl; dy_Tgt2Msl; a_Msl]
% u = a_Msl
% w = a_Tgt

A = [0 1 0; 0 0 1; 0 0 -1/tau];
B = [0; 0; 1/tau];
D = [0; 1; 0];

% Guidance Law Parameteres

inv_b  = 1e-3; % Penalty on miss distance
Gamma  = Inf; % One-sided optimal control (w is assumed to be constant)
i_Feed = 1; % w is assumed to be measured

% Time to go measurement errors

t_Go_Error = 0; % Zero bias
t_Go_Scale = 1;

% Simulation - Initial Conditions

n   = length(A);
```

```
x   = [10; zeros(n-1,1)];
nt  = -1*g;
ntd = nt;

i   = 0;
J_u = 0;

for t = 0:dt:t_Final

    i = i + 1;

    t_GoTrue = t_Final - t; % True time to go
    t_GoMeas = t_GoTrue*t_Go_Scale + t_Go_Error; % Measured time to go

    k = MinEffort_Gains(t_GoMeas, tau_D, inv_b, Gamma, i_Feed);
    u = k*[x; ntd]; % Missile acceleration command

    if (u > ncg_Max)
        u = ncg_Max;
    elseif (u < -ncg_Max)
        u = -ncg_Max;
    end

    J_u = J_u + u^2*dt; % Integral control effort

    w  = ntd; % State Space Equations
    dx = A*x + B*u + D*w;
    x  = x + dx*dt;

    t_Vec(i)      = t; % Store data - Graphics
    y_Vec(i)      = x(1);
    dy_Vec(i)     = x(2);
    nL_Vec(i)     = x(3);
    ntd_Vec(i)    = ntd;
    nc_Vec(i)     = u;
    J_u_Vec(i)    = J_u;

end

% Plots

N     = i;
Miss  = x(1);
Vel   = x(2);
Angle = 180/pi*x(2)/V_Closing;
J     = J_u_Vec(N)/(g^2*t_Final);

Miss_Title  = sprintf('Miss Distance (m) = %g', Miss);
Angle_Title = sprintf('Interception Angle (deg) = %g', Angle);
J_Title     = sprintf('Integral of u^2/g^2*tf = %g', J);

figure
subplot(2,2,1)
```

```
plot(t_Vec,y_Vec);
xlabel('Time (s)');
ylabel('Relative separation (m)');
grid on;
title(Miss_Title);

subplot(2,2,2)
plot(t_Vec,dy_Vec);
xlabel('Time (s)');
ylabel('Relative velocity (m/s)');
grid on;
title(Angle_Title);

subplot(2,2,3)
plot(t_Vec,nc_Vec/g,t_Vec,nL_Vec/g);
xlabel('Time (s)');
ylabel('Interceptor acceleration (g)');
legend('nc','nL');
grid on;
title('Acceleration Profile');

subplot(2,2,4)
plot(t_Vec,J_u_Vec/(g^2*t_Final));
xlabel('Time (s)');
ylabel(J_Title);
grid on;
title('Control Effort');

% Autopilot

y_0     = 10;
a_Tgt_N = ntd_Vec(1)/g;
V_Msl   = 300;
V_Tgt   = 300;
Tm      = 120;
Gamma_c = 10*pi/180;

MDelta_n = 200;  % Units [1/sec^2]
MDelta_t = 200;

F11 = [0 1 0 0; 0 0 1 0; 0 0 0 0; 0 0 g/V_Tgt 0];
F12 = [0 0 0 0; -Tm 0 -Tm -Tm; 0 0 0 0; Tm/V_Msl 0 Tm/V_Msl Tm/V_Msl];
F21 = zeros(4,4);
F22 = [0 1 0 0; 0 0 MDelta_n -MDelta_t; 0 0 -1/tau 0; 0 0 0 -1/tau];

FA = [F11 F12; F21 F22];
GA = [0; 0; 0; 0; 0; 0; 1/(Tm*tau); -1/(Tm*tau)];
CA = eye(8);
DA = zeros(8,1);

x_Initial = [y_0; 0; a_Tgt_N; -Gamma_c; 0; 0; 0; 0];
```

B.3 Integrated architecture – SISO systems

```
% Name: Maria Morante Soria
% File: Missile_SISO.m

Scenario_Parameters; % Initialization Scenario Parameters Values

% Weight matrices

R  = 1;
Qf = diag([a^2 0 0 b^2 0 0 0 0]);

C = [0 0 0 0 Tm 0 Tm Tm]; % a_PN = C*x

% SISO Autopilot Scheme Design Gains

K_dTheta_n = 0.035;
K_dTheta_t = -0.035;
cHat_n     = 0.25;
cHat_t     = -0.15;

%% FULL STATE SINGLE LOOP GUIDANCE LAW - SEPARATED G&C

F = [0 1 0; 0 0 1; 0 0 0];
G = [0; 1; 0];

CSim = eye(3);
DSim = zeros(3,1);

M = [a 0 0];

% Simulation

sim('Missile_LQ_Separated');

%% FULL STATE TWO LOOP AUTOPILOT GUIDANCE LAW

F11 = [0 1 0 0; 0 0 1 0; 0 0 0 0; ...
    0 0 1/(V_Tgt*cos(Gamma_Tgt_0 + Lambda_0)) 0];

F12 = [0 0 0 0; -Tm 0 -Tm -Tm; 0 0 0 0; ...
    Tm/(V_Msl*cos(Gamma_Msl_0 - Lambda_0)) 0 ...
    Tm/(V_Msl*cos(Gamma_Msl_0 - Lambda_0)) ...
    Tm/(V_Msl*cos(Gamma_Msl_0 - Lambda_0))];

F21  = zeros(4,4);
FA22 = [0 1 0 0; 0 0 MDelta_n -MDelta_t; ...
    -cHat_n/tau_n -K_dTheta_n/tau_n -(1+cHat_n)/tau_n -cHat_n/tau_n; ...
    -cHat_t/tau_t -K_dTheta_t/tau_t -cHat_t/tau_t -(1+cHat_t)/tau_t];

FA = [F11 F12; F21 FA22];
GA = [0; 0; 0; 0; 0; 0; cHat_n/tau_n; cHat_t/tau_t];
```

```
CASim = eye(8);
DASim = zeros(8,1);

P = Riccati(FA, GA, R, Qf, 2, 0.001);

% Simulation

sim('Missile_SISO_TwoLoop_2');

%% FULL STATE TWO LOOP AUTOPILOT GUIDANCE LAW - SERVO & AIRFRAME

FA22 = [0 1 0 0; 0 0 MDelta_n -MDelta_t; ...
    -cHat_n/tau_n -K_dTheta_n/tau_n -(1+cHat_n)/tau_n -cHat_n/tau_n; ...
    -cHat_t/tau_t -K_dTheta_t/tau_t -cHat_t/tau_t -(1+cHat_t)/tau_t];

FA = [FA22];
GA = [0; 0; cHat_n/tau_n; cHat_t/tau_t];

CASim = eye(4);
DASim = zeros(4,1);

Qf2 = diag([a^2 0 0 0]);
P   = Riccati(FA, GA, R, Qf2, 2, 0.001);

C2 = [Tm 0 Tm Tm]; % a_PN = C*x

% Simulation

sim('Missile_SISO_TwoLoop_Servo_Airframe');
```

B.4 Integrated architecture – MIMO systems

```
% Name: Maria Morante Soria
% File: Missile_MIMO.m

Scenario_Parameters; % Initialization Scenario Parameters Values

% Weight matrices

R  = [alpha 0; 0 1];
Qf = diag([a^2 0 0 b^2 0 0 0 0]);

C = [0 0 0 0 Tm 0 Tm Tm]; % a_PN = C*x

% MIMO Autopilot Scheme Design Gains

K_dTheta_n = -0.035;
K_dTheta_t = 0.035;
cHat_n     = 0.25;
```

```
cHat_t      = -0.15;

% sim('MIMO_2Loop');

%% FULL STATE SINGLE LOOP GUIDANCE LAW

F11 = [0 1 0 0; 0 0 1 0; 0 0 0 0; ...
    0 0 1/(V_Tgt*cos(Gamma_Tgt_0 + Lambda_0)) 0];

F12 = [0 0 0 0; -Tm 0 -Tm -Tm; 0 0 0 0; ...
    Tm/(V_Msl*cos(Gamma_Msl_0 - Lambda_0)) 0 ...
    Tm/(V_Msl*cos(Gamma_Msl_0 - Lambda_0)) ...
    Tm/(V_Msl*cos(Gamma_Msl_0 - Lambda_0))];

F21 = zeros(4,4);
F22 = [0 1 0 0; 0 0 MDelta_n -MDelta_t; 0 0 -1/tau_n 0; 0 0 0 -1/tau_t];

F = [F11 F12; F21 F22];
G = [0 0; 0 0; 0 0; 0 0; 0 0; 0 0; 1/tau_n 0; 0 1/tau_t];

CSim = eye(8);
DSim = zeros(8,2);

P = Riccati(F, G, R, Qf, 2, 0.001);

% Simulation

sim('Missile_MIMO_SingleLoop');

%% FULL STATE TWO LOOP AUTOPILOT GUIDANCE LAW

F11 = [0 -1 0 0; 0 0 1 0; 0 0 0 0; ...
    0 0 1/(V_Tgt*cos(Gamma_Tgt_0 + Lambda_0)) 0];

F12 = [0 0 0 0; -Tm 0 -Tm -Tm; 0 0 0 0; ...
    Tm/(V_Msl*cos(Gamma_Msl_0 - Lambda_0)) 0 ...
    Tm/(V_Msl*cos(Gamma_Msl_0 - Lambda_0)) ...
    Tm/(V_Msl*cos(Gamma_Msl_0 - Lambda_0))];

F21  = zeros(4,4);
FA22 = [0 1 0 0; 0 0 MDelta_n -MDelta_t; ...
    -cHat_n/tau_n -K_dTheta_n/tau_n -(1+cHat_n)/tau_n -cHat_n/tau_n; ...
    -cHat_t/tau_t -K_dTheta_t/tau_t -cHat_t/tau_t -(1+cHat_t)/tau_t];

FA = [F11 F12; F21 FA22];
GA = [0 0; 0 0; 0 0; 0 0; 0 0; 0 0; cHat_n/(Tm*tau_n) 0; ...
    0 cHat_t/(Tm*tau_t)];

CASim = eye(8);
DASim = zeros(8,2);

P = Riccati(FA, GA, R, Qf, 2, 0.001);
```

```
% Simulation

% sim('Missile_MIMO_TwoLoop');
```

B.5 Guidance law

```
% Name: Maria Morante Soria
% File: Guidance.m

function u = Guidance(y, dy, a_TN, R, G, P, x)
%#codegen

u = -inv(R)*transpose(G)*P*x;

end
```

B.6 Guidance law: Closed form solution

```
% Name: Maria Morante Soria
% File: Guidance_ClosedFormSolution.m

function u = Guidance_ClosedFormSolution(y, dy, a_TN, a, M, F, Time)
%#codegen

t_Go = 3 - Time;
x    = [y; dy; a_TN];
u    = a*t_Go/(1 + a^2*(t_Go)^3/3)*transpose(M)*exp(F*t_Go)*x;

end
```

B.7 Riccati equation

```
% Name: Maria Morante Soria
% File: Riccati.m

function P = Riccati(A, B, R, Qf, tf, dt)

i = 0;

P    = Qf;
dP   = zeros(size(P));
invR = inv(R);

for t = 0:dt:tf
```

```
    i      = i + 1;
    M      = transpose(B)*P;
    dP_pre = dP;
    dP     = transpose(A)*P + P*A - transpose(M)*invR*M;
    P      = P + (dP + dP_pre)*dt/2;
    P      = (P + transpose(P))/2;

end
end
```

B.8 Minimum effort law gains

```
% Name: Maria Morante Soria
% File: MinEffort_Gains.m

function [k, NO_Out] = MinEffort_Gains(t, tau, inv_b, Gamma, i_Feed)

% Generalized Minimum Effort Law gains
% Inputs
    % t: Time to go
    % tau: Missile time constant
    % inv_b: Miss distance weighting
    % Gamma: Target maneuver weighting
    % i_Feed: 1 = Apply feedback from target acceleration
    %         0 = Zero gain on target acceleration

h   = t/tau;
h2  = h.*h;
h3  = h2.*h;
he  = exp(-h);
he2 = exp(-2*h);

tau2 = tau^2;
tau3 = tau^3;

if (Gamma ~= Inf)
    Gamma2  = 1/Gamma^2;
    f_Gamma = 1 - Gamma2;
else
   f_Gamma = 1;
end

eps1 = 1e-14;

NO_Num = 6.*h2*(he - 1 + h);
NO_Den = 2*h3*f_Gamma + 3 + 6*h - 6*h2 - 12*h*he - 3*he2;
NO_Den = NO_Den + eps1 + inv_b*6/tau3;
NT     = NO_Num/NO_Den;

NO1 = (6/tau2)*(he - 1 + h);
```

```
NO2 = (6*h/tau)*(he - 1 + h);

NO_Out = NO_Den - eps1;

KL = NT*(he - 1 + h)/(h2 + eps1);

if (i_Feed == 1)
    k = [-NO1/NO_Den, -NO2/NO_Den, -KL, -NT/2];
else
    k = [-NO1/NO_Den, -NO2/NO_Den, -KL, -0];
end

end
```

Appendix C

Simulink® model

The Appendices C.1 and C.2 include the two autopilot schemes studied based on the number of inputs. Appendix C.1 encloses the classical autopilot – guidance scheme where the guidance law issues a single command. Besides, Appendix C.2 presents the multi – input autopilot design, where there are two inputs to the autopilot (nose and tail deflection commands).

Additionally, Appendices C.3, C.4 and C.5 include the separated two – loop autopilot – guidance design, the integrated single – loop guidance design and the integrated two – loop autopilot – guidance design, respectively.

C.1 Single – input two – loop autopilot scheme

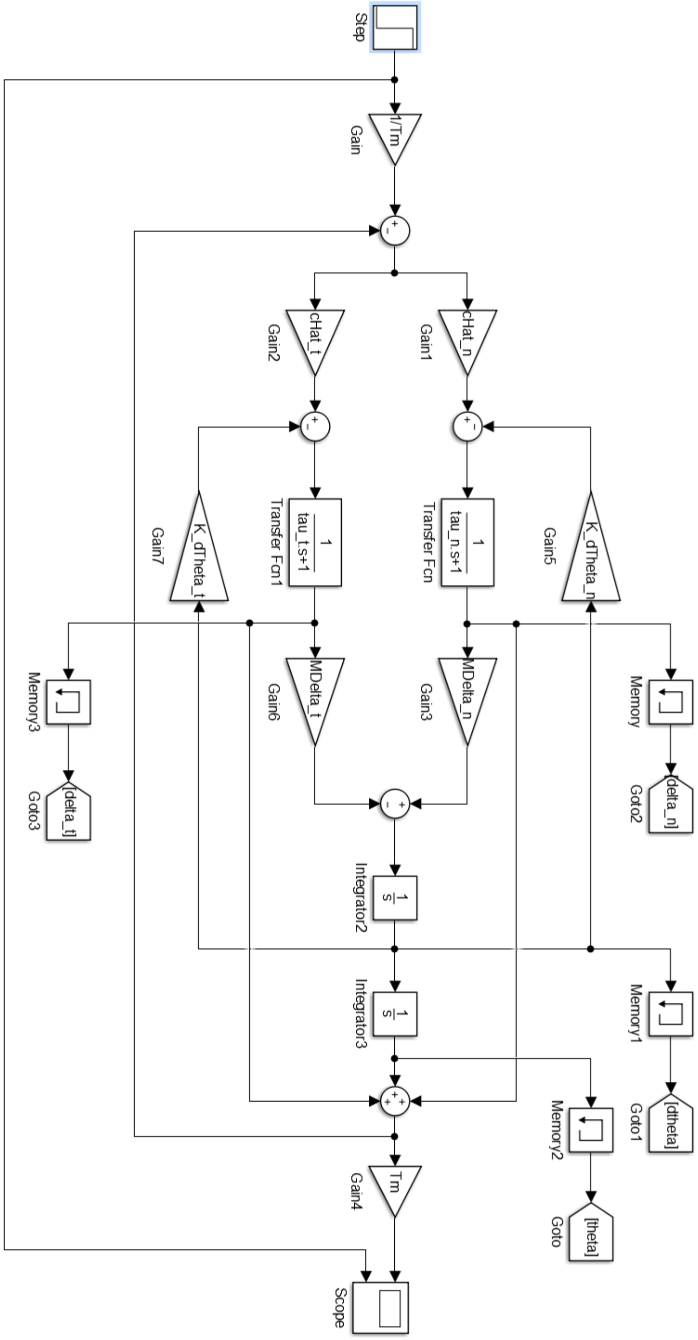


Figure C.1: Single – input two – loop autopilot design.

C.2 Multi – input two – loop autopilot scheme

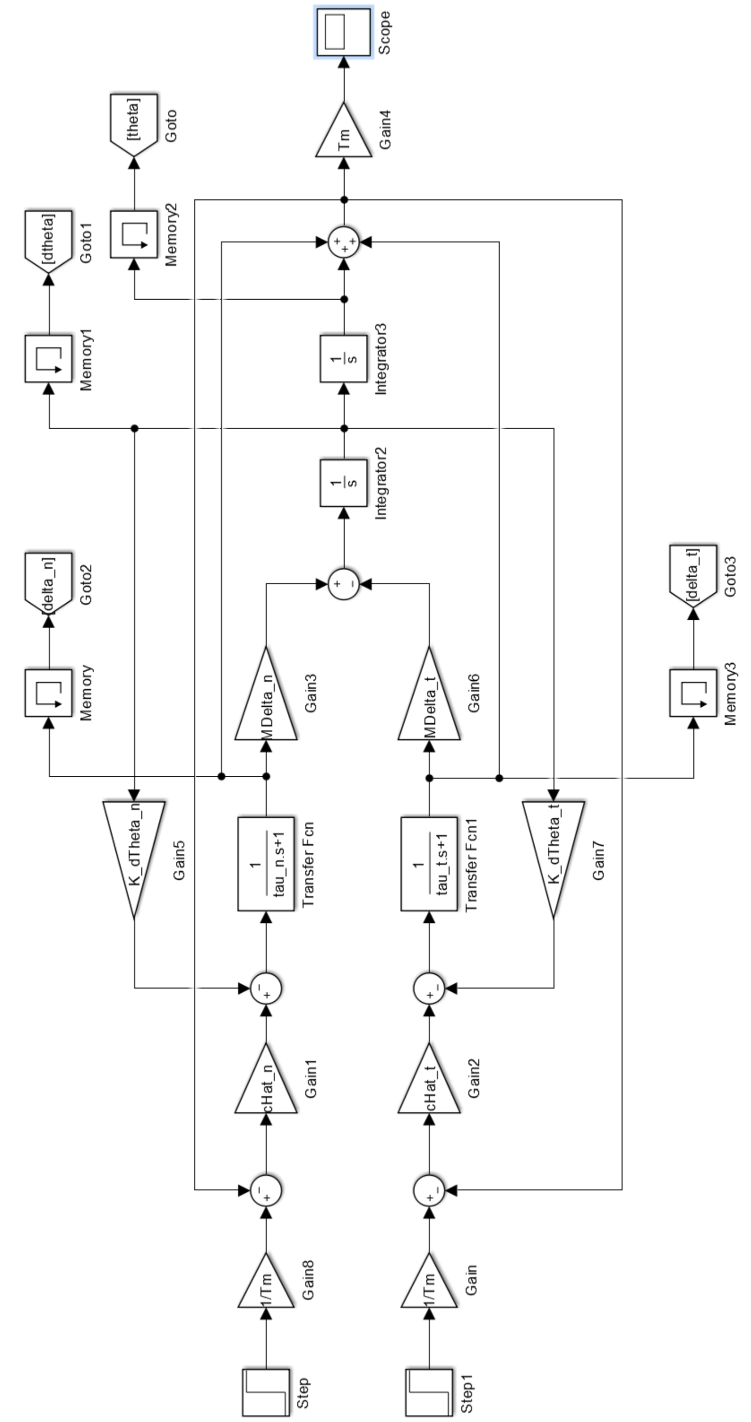


Figure C.2: Multi – input two – loop autopilot design.

C.3 Separated two – loop autopilot – guidance design

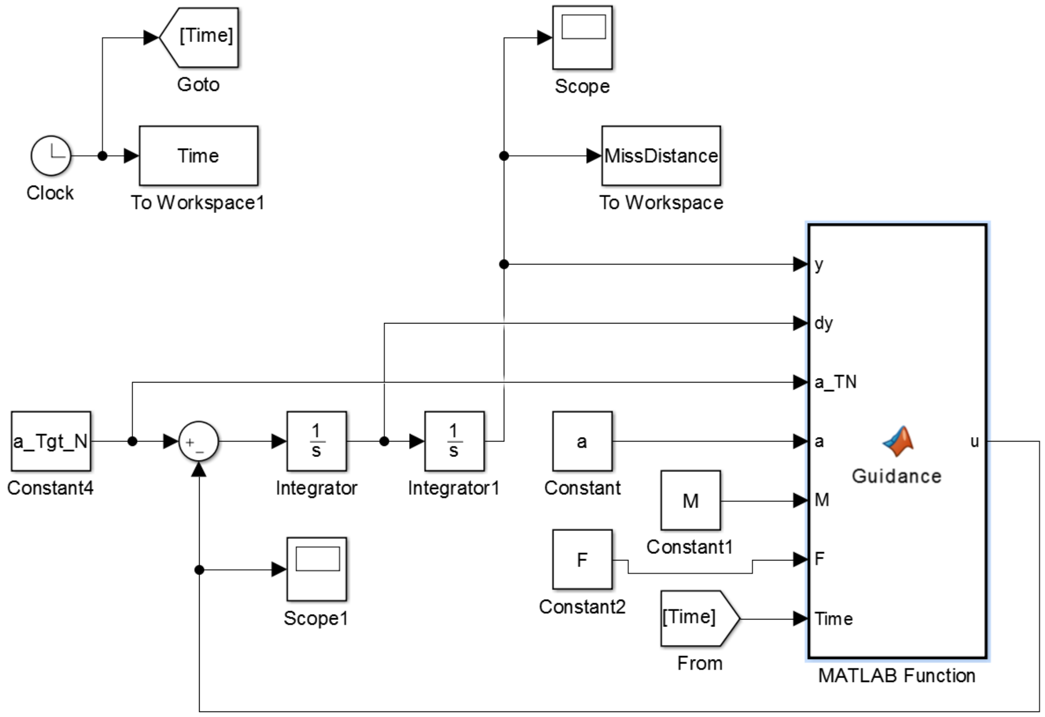


Figure C.3: Separated two – loop autopilot – guidance design.

C.4 Integrated single – loop autopilot – guidance design

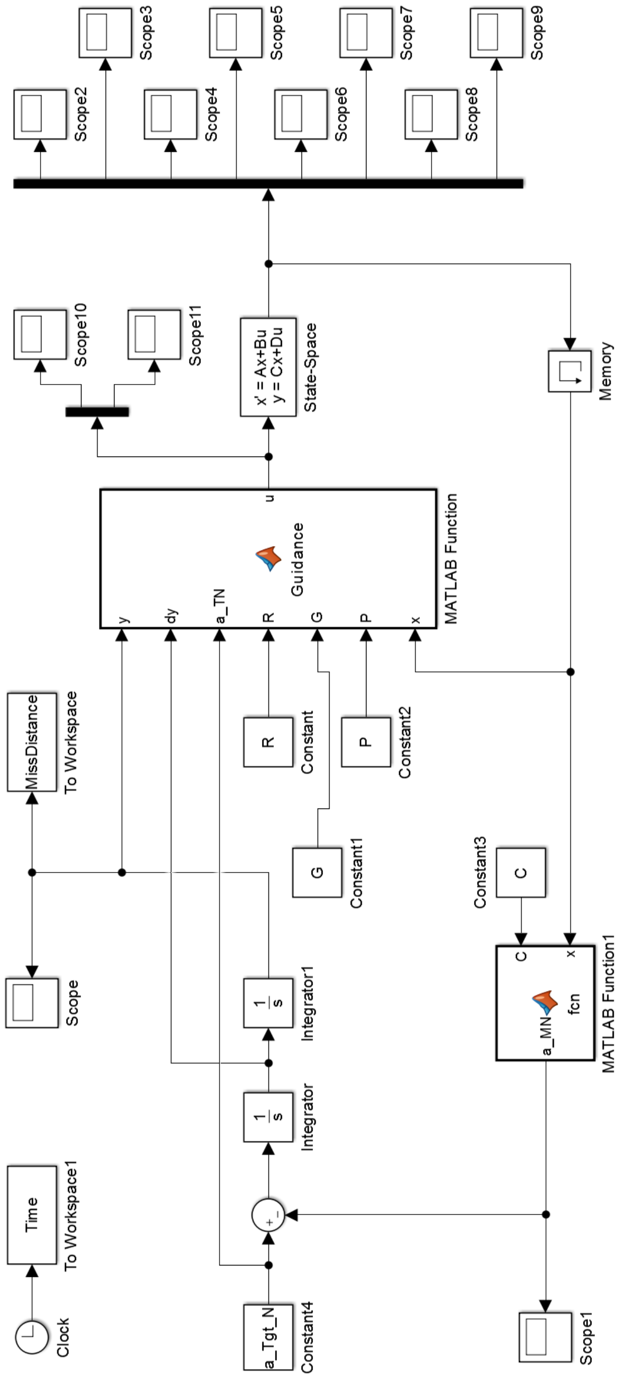


Figure C.4: Integrated multi – input single – loop autopilot – guidance design.

C.5 Integrated two – loop autopilot – guidance design

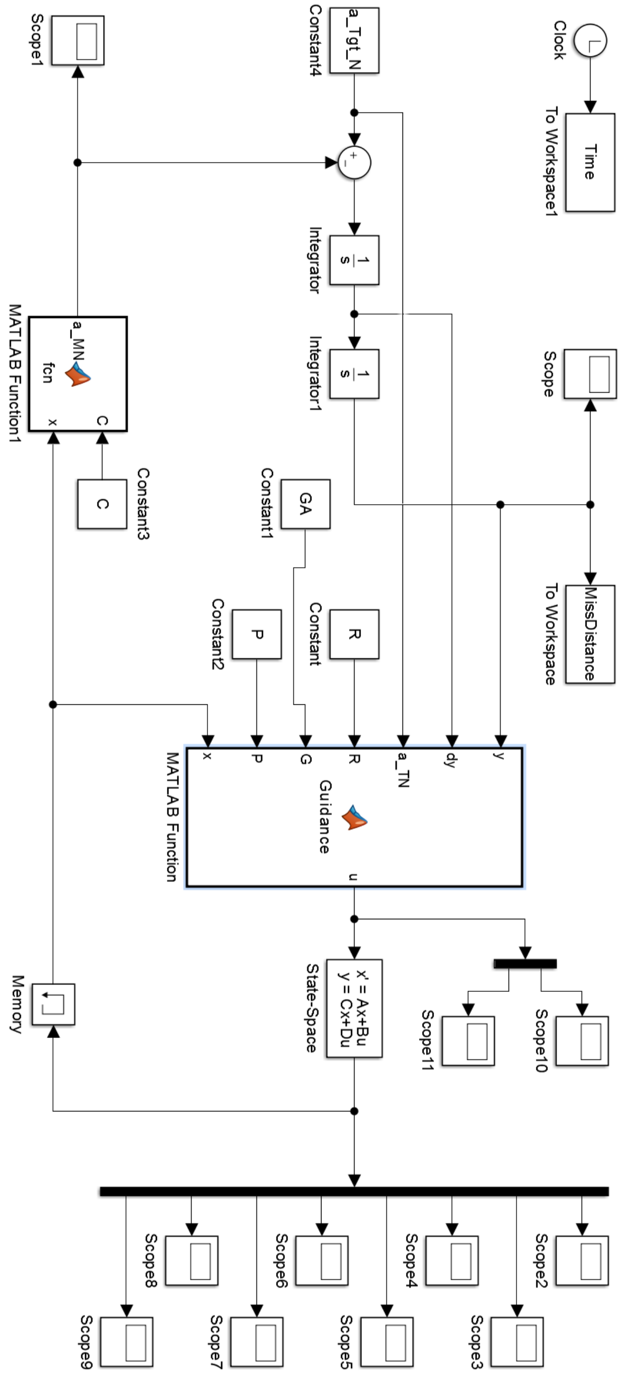


Figure C.5: Integrated multi – input two – loop autopilot – guidance design.