Optimal trajectory tracking control design

Emulating fighter aircraft flight test tracks with nonlinear model predictive control

Andres Jürisson

Master of Science Thesis

TU Delft Delft University of Technology

Delft Center for Systems and Control

Optimal trajectory tracking control design

Emulating fighter aircraft flight test tracks with nonlinear model predictive control

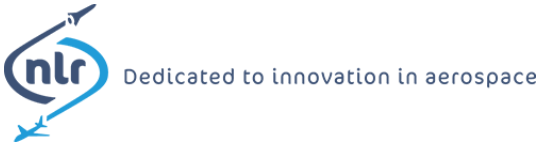
Master of Science Thesis

For the degree of Master of Science in Systems and Control at Delft University of Technology

Andres Jürisson

July 9, 2018

Faculty of Mechanical, Maritime and Materials Engineering (3mE) · Delft University of Technology



The work in this thesis is the result of a collaboration with Netherlands Aerospace Centre NLR.

Abstract

Aircraft lifespan can be extended by upgrading and modernizing the electrical subsystems and instruments. However, this often introduces increased power demands and heat generation that the aircraft was not originally designed for and can result in a reduced flight performance and increased wear. Netherlands Aerospace Centre (NLR) and the Netherlands Ministry of Defence have set up a project to investigate power and thermal management of different aircraft platforms and to optimise the operational capabilities of the F-16. As part of this project, flight tests with the F-16 fighter jet performing various challenging aerobatic maneuvers were conducted and also replicated in a simulation environment by the pilot to validate the simulator accuracy. In order to evaluate the effect of different flight configurations on the power and thermal loads, it is necessary to track the same trajectories numerous times while changing the aircraft settings. Having a pilot in the loop introduces undesired variability to the results and is infeasible. Instead, a controller is required that is able to use the recorded reference trajectories and closely reproduce these flights in a simulated environment.

In this thesis, a publicly available F-16 model was used and extended with a feedback linearization controller to generate an analogous set of aerobatic reference flight trajectories. To track the reference trajectories two controllers were developed. A nonlinear model predictive controller and a nonlinear model predictive controller combined with a feedback linearization controller in the form of incremental nonlinear dynamic inversion. Adding feedback linearization reduced computation time and improved tracking of engine dynamics but made the controller dependent on reference actuator angles while the nonlinear model predictive controller alone is slower, but is able to track accurately with and without thrust and actuator angle reference signals. Tracking accuracy was tested on a set of well known aerobatic maneuvers such as Barrel rolls, loops and Half Cuban Eights. The NMPC controller was able to track the aircraft position in these maneuvers with a mean error of 0.44 ft ($\sigma = 0.39$ ft) while the NMPC-INDI controller achieved a mean position tracking error of 0.20 ft ($\sigma = 0.10$ ft). Further tests included mismatches in initial fuel weight and position and finally both controllers were validated by tracking a reference generated by NLR F-16 simulator. Here both controllers were able to adjust the thrust level to counter the effects of the speed brakes that were present for the trajectory generation but not for tracking. The NMPC was able to track the aircraft position with a mean position error of 20.01 ft ($\sigma = 13.74$ ft) while the

NMPC-INDI achieved a mean position error of 8.70 ft ($\sigma = 2.38$ ft). These tests showed that both controllers are able to handle significant differences in the aircraft models and still keep the mean position tracking errors within one wingspan length (30 ft) which was the size of the reference tunnel for the pilot to track.

Table of Contents

List of Figures

List of Tables

Acknowledgements

First, I would like to thank Tamás Keviczky for his supervision and guidance throughout this thesis. I would also like to thank Bart Eussen, Mark Verveld and Kees Wijnberg for asking the right questions to improve my thesis and providing a wonderful working environment at NLR.

Finally, I would like to thank Pille, Alar and Eliis for supporting me during my studies and Egelyn and Karl for bringing me to Delft in the first place.

Delft, University of Technology
July 9, 2018

Andres Jürisson

Chapter 1

Introduction

It is in the interest of every operator to use an aircraft as long as it is safe and viable as this delays the need to replace the fleet and can lead to cost reductions. One option, to extend the aircraft lifespan, is to upgrade the electrical systems to more modern versions, which can add extra functionality that was not available originally. However, this upgrade often leads to higher electrical power demands and higher thermal loads that can lead to significant effects on the aircraft cooling capabilities and a reduction in performance. Furthermore, the increased power demands can exceed the designed power requirements and the operational capabilities of the aircraft can be impacted due to the higher temperatures and increased wear [1].

In order to gain insight to power and thermal management of different aircraft platforms and to optimise the operational capabilities of the F-16 Fighting Falcon (F-16), Netherlands Aerospace Centre (NLR) and the Netherlands Ministry of Defence have set up the Fighter Aircraft Robust Power Management (FARPM) project. As part of the project a power and thermal management simulator has been created which includes models of numerous sub-systems such as the engine, fuel/oil system, environmental control system, electrical power system and hydraulic power actuation system. With the simulator it is possible to estimate power demands and heat distribution during flight as can be seen in Figure 1-1.

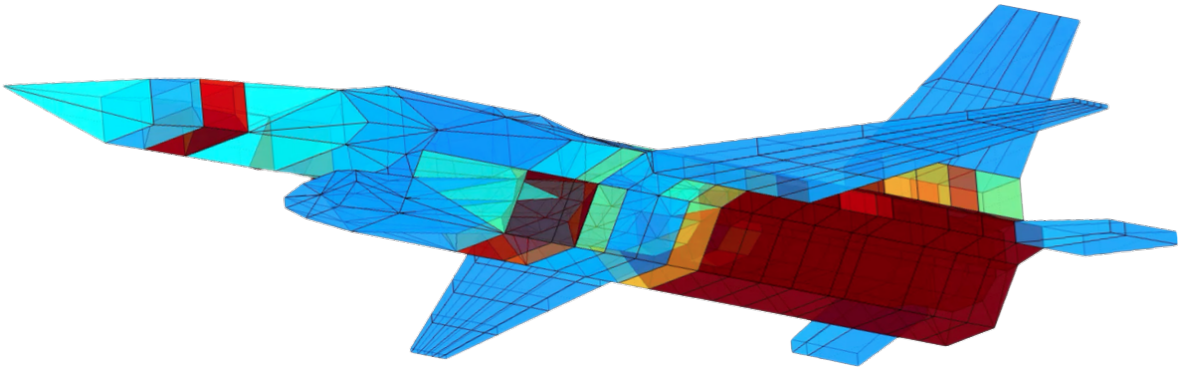


Figure 1-1: FARPM simulator heat distribution [1].

The FARPM simulator is driven using outputs from a Flight Dynamics Simulator (FDSIM) which include the control surface motions, atmospheric conditions, engine settings and more. Therefore, the accuracy of the FARPM simulator results is closely connected to the accuracy of the flight dynamics simulator. To validate the FDSIM simulator, a set of flight tests were conducted with a specially instrumented F-16 fighter jet measuring different parameters, such as the aircraft position, velocities, rotational rates and attitude, actuator angles, etc. Then the same flights were conducted in the simulation environment by creating a Tunnel-In-The-Sky from the reference data for the pilot to follow as seen in Figure 1-2. Observing that the control inputs to achieve both flights are comparable, gave validity to the FDSIM model.

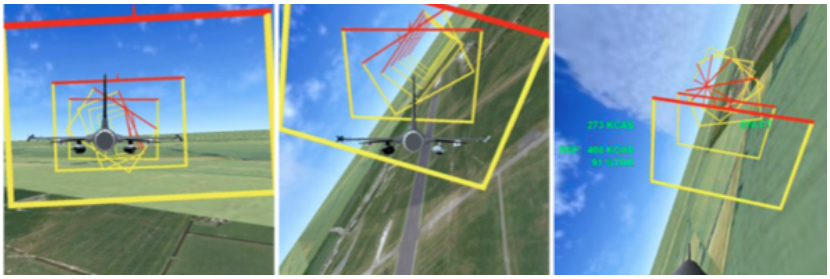


Figure 1-2: Tunnel-In-The-Sky simulation [1].

1-1 Motivation and problem formulation

The reference trajectories include, among others, aerobatic flight demonstrations which include very dynamic maneuvers, such as fast turns and rolls, loops, barrel rolls, spirals etc. Due to the complex nature of the reference trajectories it is challenging for the pilot to remain in the reference tunnel with two wingspans wide (60 ft) sides, when trying to reproduce these flights inside a simulation environment. It can easily take an experienced pilot up to a week of practice to accurately follow a reference flight that is 12 minutes long. Using a pilot to validate the accuracy of the FDSIM model by performing a single reference flight in the simulation was sufficient as it only needed to be done once. However, the FARPM simulator will be used as a tool to evaluate the effect of different flight configurations on the power and

thermal loads, which requires tracking the same trajectories numerous times while changing the aircraft settings. A pilot will not be able to carry out a large number of flight iterations while at the same time having minimal variability between the tests. Therefore a controller is needed to perform this tracking automatically and consistently in order to be able to assess the effects of changes in the aircraft configuration correctly.

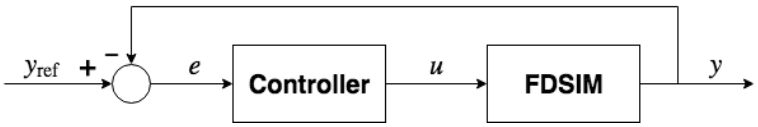


Figure 1-3: Controller and FDSIM system architecture.

The controller is required to generate commands for thrust, aileron, elevator and rudder angles for FDSIM such that the error between the reference signals presented in Table 1-1 and the simulation states is minimized as shown in Figure 1-3. The controller needs to be able to track the reference signals with and without using the actuator angle and thrust references shown separately in Table 1-1. Furthermore, the controller is required to handle mismatches between the aircraft model used to generate the reference signals and the model used for tracking as for example, the aircraft mass and moment of inertia could differ. However, there is no real time computation requirement for the controller as it is used for generating accurate inputs for the FARPM simulator that is also not real time.

Reference signal	Symbol
North position	x
East position	y
Altitude	z
North velocity	V_x
East velocity	V_y
Climb rate	V_z
Angle of attack	α
Angle of sideslip	β
Roll angle	ϕ
Pitch angle	θ
Yaw angle	ψ
Roll rate	p
Pitch rate	q
Yaw rate	r
.....	.....
Elevator angle	δ_e
Aileron angle	δ_a
Rudder angle	δ_r
Thrust level	δ_T

Table 1-1: Available reference signals.

1-2 Related literature

In literature many possible approaches can be found to perform trajectory or reference tracking on aircraft. Due to the nonlinear nature of flight dynamics and aerodynamic equations, nonlinear control laws such as Nonlinear Dynamic Inversion (NDI) also known as Feedback Linearization (FBL) are often used as an alternative to linear gain scheduled controllers, which require designing a separate controller valid within a small operational region around each linearization point across the flight envelope. With NDI, instead of using gain scheduling a single controller can be designed, which covers the entire flight envelope. In addition to reduced design effort, nonlinear controllers were shown to also have greater performance compared to gain scheduled laws [4]. Aircraft altitude and heading reference tracking was implemented in [5] using time-scale separation principle, where fast and slow system dynamics were split to separate guidance and attitude control loops while making a constant airspeed assumption. In [6] aircraft position tracking for the full F-16 model was achieved through the implementation of a backstepping controller with four separate feedback loops and tested using trajectories involving upwards spirals and turns with altitude changes. NDI methods require an accurate model in order to correctly calculate the system inverse which can lead to reduced performance in the presence of modelling errors. The robustness of NDI controllers to model mismatch can be increased by implementing Incremental Nonlinear Dynamic Inversion (INDI) control, which is a version of NDI less dependent on the system model. This method was demonstrated in [7] to control the aircraft rotational rates using measurements or estimates of the rotational accelerations.

Another common approach to implement trajectory tracking on aircraft is to use optimal control methods. In optimal control, the control law is found such that the optimality of a cost function is achieved which in turn is a function of the state variables and inputs. In [8] Linear Quadratic Regulator (LQR) is applied on a linearized F-16 longitudinal model to track pitch and flight path angle reference signals. Another approach is to use Model Predictive Control (MPC) which supports implementing system input and state constraints and performing optimization online. In [9] airspeed, roll and pitch angle references are tracked for a Boeing 747 that is linearized around an equilibrium point in straight and level flight. Linear Parameter Varying (LPV) models can be used to increase the set of flight conditions where good tracking performance can be achieved. This was shown in [10], where the longitudinal model of F-16 was linearized across the flight envelope and in [11] a LPV model was obtained for a F-16 model with both longitudinal and lateral modes. When linearizing the model an approximation of the system dynamics is obtained. However, in Nonlinear Model Predictive Control (NMPC) nonlinear system models can be used, but at the expense of increased computation times that grow unfeasible when large models or long horizons are used. In [12] a NMPC controller was used to perform dynamic maneuvers with an aircraft model involving a simple aerodynamic model. It is also possible to combine MPC with feedback linearization to convert the model into linear form in an inner loop that then can be used by MPC in an outer loop. This idea is explored in [13, 14] together with methods to map the constraints for the MPC when feedback linearization is performed. In situations, where the reference signals are known in advance, it is possible to incorporate that information into the MPC controller. Then, instead of trying to match predicted states to a constant reference value over the horizon, it is possible to use future reference values to achieve better performance [15].

Tracking performance can significantly suffer when there is uncertainty or mismatches in the

system. For aircraft this uncertainty can arise from inaccuracies in aerodynamic modelling, neglected dynamics, measurement noise or flexible body dynamics. Often some dynamics is knowingly neglected in order to simplify the models and additionally not all system parameters are accurately known or are varying in time. Robust controllers have been developed to handle these uncertainties and the approaches are similar to MPC where the nonlinear systems are mostly either linearized and LPV models are obtained or feedback linearization is applied to again acquire a linearized model. Altitude and velocity reference tracking is performed in [16, 17] while using LPV models obtained from aircraft nonlinear longitudinal models. The gain scheduled state feedback controllers were designed using H_∞ and μ synthesis respectively. In [18], FBL was first applied to the longitudinal model of a hypersonic vehicle and then a robust controller was designed to track the altitude and velocity reference signals.

1-3 Thesis contribution

In this thesis a Nonlinear Model Predictive Control method is developed to emulate acrobatic F-16 reference flights in a simulated environment by simultaneously tracking reference position, velocity, attitude and rotational rate signals. Two separate controllers have been developed - a NMPC controller that is capable of performing tracking without need for reference actuator signals and a NMPC-INDI controller that requires reference actuator signals, but has reduced simulation time.

The NMPC approach was chosen due to the following advantages:

1. Allows to prioritize the supplementary reference signals using weights in the cost function
2. Systematic treatment of constraints
3. Supports using future reference signals in the form of preview control
4. Performs control allocation
5. Using a nonlinear model in the optimization allows to accurately capture the dynamics of difficult reference maneuvers

NMPC can be computationally very demanding, which can make using it unfeasible in most situations, but since there is no real time requirement for the controller, this disadvantage is not critical. NMPC performance can suffer when the controller prediction model is inaccurate. However, the tracking is performed in a simulated environment and for this reason it is assumed that the aircraft model used for tracking is also available for the controller and there is no significant difference between them. Furthermore, full state information is assumed to be available for the controller from the simulator.

The control algorithm was developed in Matlab/Simulink using publicly available[1] F-16 model developed at University of Minnesota for reference trajectory generation and tracking.

[1] F-16 model available at: www.aem.umn.edu/people/faculty/balas/darpa_sec/software/

1-4 Thesis outline

This chapter gave an overview of the project background and introduced the requirements for the trajectory tracking controller. Furthermore, the chosen approach together with the advantages was presented. The remainder of the thesis is structured as follows:

- Chapter 2 gives an overview of the F-16 model and the rotational rate and sideslip angle controllers developed to be able to generate reference trajectories. In addition, the modifications made to improve the accuracy of the model are presented.
- Chapter 3 presents the tuning process of the NMPC and NMPC-INDI controller, including the selection of the cost function weights, sampling time and prediction horizon lengths.
- In Chapter 4 the tracking performance of the controllers following various maneuvers is presented. This is followed by tests including model mismatch and tracking of NLR reference trajectory to validate the controllers. Further, the performance of the controllers is discussed.
- Finally, Chapter 5 concludes the thesis and gives recommendations for future work.

Chapter 2

Reference trajectory generation

This chapter gives an overview of the F-16 model in Section 2-1 and presents the implementation of rotational rate and sideslip angle controllers in Section 2-2 and Section 2-3. Then the aircraft attitude representation using quaternions is presented in Section 2-4 followed by modifications done to the engine model in Section 2-5. Finally, an overview of the generated reference trajectories is given in Section 2-6.

2-1 University of Minnesota F16 model

The F-16 model used throughout this thesis is a publicly available flight dynamics model published by University of Minnesota [3]. This is a nonlinear model that is based on extensive wind tunnel tests conducted by NASA and published in a technical report in 1979 [2].

The nonlinear simulation model consists of thirteen states that are presented in Table 2-1 together with the state explanation and used units. The outputs of the model are the states and additional parameters like normalized accelerations and aerodynamic parameters as presented in Table 2-2.

Table 2-1: Model states [3].

State	Explanation	Units
x	North position	ft
y	East position	ft
z	Altitude	ft
ϕ	Roll angle	rad
θ	Pitch angle	rad
ψ	Yaw angle	rad
V_t	Total velocity	ft/s
α	Angle of attack	rad
β	Sideslip angle	rad
p	Roll rate	rad/s
q	Pitch rate	rad/s
r	Yaw rate	rad/s
δ_{lef}	Leading edge flap deflection angle	rad

Table 2-2: Extended outputs of the model [3].

State	Explanation	Units
a_{nx}	Normalized acceleration in x direction	g
a_{ny}	Normalized acceleration in y direction	g
a_{nz}	Normalized acceleration in z direction	g
M	Mach number	-
$\bar{q}$	Free-stream dynamic pressure	lb/ft^2
P_s	Static pressure	lb/ft^2

The control inputs of the model are thrust level and elevator, aileron and rudder angles. These are presented in Table 2-3 together with the actuator limits. The leading edge flap is an additional control surface which cannot be controlled by the pilot directly. Instead the deflection angle is calculated as a function of the angle of attack, static and dynamic pressure as shown in (2-1).

Table 2-3: Control inputs of the F-16 model [3].

Control	Explanation	Units	Min	Max	Min rate	Max rate
δ_T	Thrust	lbs	1000	19000	-10000	10000
δ_e	Elevator	deg	-25	25	-60	60
δ_a	Aileron	deg	-21.5	21.5	-80	80
δ_r	Rudder	deg	-30	30	-120	120
δ_{lef}	Leading Edge Flap	deg	0	25	-25	25

$$\delta_{\text{lef}} = 1.38\frac{2s+7.25}{s+7.25}\alpha - 9.05\frac{\bar{q}}{P_s} + 1.45 \tag{2-1}$$

The model also has restrictions on the allowable flight conditions. These are presented in Table 2-4.

Table 2-4: Model flight condition limitations [3].

Variable	Explanation	Units	Min	Max
z	Altitude	ft	5000	40000
α	Angle of attack	deg	-20	90
β	Angle of sideslip	deg	-30	30
V_t	Total velocity	ft/s	300	900

Finally, the actuator and engine dynamics are modelled as first order lags as in (2-2), where the corresponding gains are presented in Table 2-5.

$$\dot{\delta} = K(\delta_c - \delta) \tag{2-2}$$

Table 2-5: Actuator and engine dynamics gains.

Gain	Value
K_T	1.0
K_e	20.2
K_a	20.2
K_r	20.2

Equations of motion and aerodynamic equations used by the F-16 simulation model are presented in Appendix A.

2-2 INDI controller for rotational rates

The F-16 is an inherently unstable aircraft, which means that in order to generate a set of reference trajectories, it is necessary to implement a control system to stabilize the aircraft. For this an Incremental Nonlinear Dynamic Inversion (INDI) controller to control the aircraft rotational rates (p, q, r) was chosen. INDI is a special version of Nonlinear Dynamic Inversion (NDI) or Feedback Linearization (FBL) that is more robust to model uncertainty and requires less information about the system. The dependency on the model is reduced by making use of the actuator output and angular acceleration measurements [19].

The INDI method is demonstrated using a system, where the output is the state, as shown in (2-3).

$$\begin{aligned} \mathbf{y} &= \mathbf{x} \\ \dot{\mathbf{y}} &= \dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{G}(\mathbf{x})\mathbf{u} \end{aligned} \tag{2-3}$$

Then the state derivative can be approximated using a Taylor series expansion truncated at first derivative as shown in (2-4) [7, 19].

$$\mathbf{f}(\mathbf{x}) + \mathbf{G}(\mathbf{x})\mathbf{u} \approx \mathbf{f}(\mathbf{x}_0) + \mathbf{G}(\mathbf{x}_0)\mathbf{u}_0 + \frac{\partial}{\partial \mathbf{x}}[\mathbf{f}(\mathbf{x}_0) + \mathbf{G}(\mathbf{x}_0)\mathbf{u}_0](\mathbf{x} - \mathbf{x}_0) + \frac{\partial}{\partial \mathbf{u}}\mathbf{G}(\mathbf{x}_0)(\mathbf{u} - \mathbf{u}_0) \tag{2-4}$$

This can then be rewritten to (2-5).

$$\dot{\mathbf{y}} = \dot{\mathbf{x}} \approx \dot{\mathbf{x}_0} + \mathbf{A}_0(\mathbf{x} - \mathbf{x}_0) + \mathbf{B}_0\Delta\mathbf{u} \tag{2-5}$$

$$\mathbf{A}_0 = \frac{\partial}{\partial \mathbf{x}}[f(\mathbf{x}) + \mathbf{G}(\mathbf{x})\mathbf{u}]_{x=x_0,u=u_0} \tag{2-6}$$

$$\mathbf{B}_0 = \frac{\partial}{\partial \mathbf{u}}[\mathbf{G}(\mathbf{x})\mathbf{u}]_{x=x_0,u=u_0} \tag{2-7}$$

$$\Delta\mathbf{u} = \mathbf{u} - \mathbf{u}_0 \tag{2-8}$$

The controller that linearizes the system to the form in (2-10) is then given by (2-9), where ν is the desired output.

$$\Delta\mathbf{u} = \mathbf{B}_0^{-1}[\nu - (\dot{\mathbf{x}_0} + \mathbf{A}_0(\mathbf{x} - \mathbf{x}_0))] \tag{2-9}$$

$$\dot{\mathbf{y}} = \nu \tag{2-10}$$

When the control signal is updated at a high rate such that the states do not change significantly from the starting values, x is very close to x_0, the control law given by (2-11) can be used instead [7, 19].

$$\Delta\mathbf{u} = \mathbf{B}_0^{-1}[\nu - \dot{\mathbf{x}_0}] = \mathbf{G}(\mathbf{x}_0)^{-1}[\nu - \dot{\mathbf{x}_0}] \tag{2-11}$$

In the case of F-16 these equations then become:

$$\begin{bmatrix}\Delta\delta_a\\ \Delta\delta_e\\ \Delta\delta_r\end{bmatrix} = \frac{\mathbf{I}}{\frac{1}{2}\rho V_t^2 S}\begin{bmatrix}bC_{l_{\delta_a}} & bC_{l_{\delta_e}} & bC_{l_{\delta_r}}\\ \bar{c}C_{m_{\delta_a}} & \bar{c}C_{m_{\delta_e}} & \bar{c}C_{m_{\delta_r}}\\ bC_{n_{\delta_a}} & bC_{n_{\delta_e}} & bC_{n_{\delta_r}}\end{bmatrix}^{-1}\left\{\begin{bmatrix}\nu_p\\ \nu_q\\ \nu_r\end{bmatrix} - \begin{bmatrix}\dot{p}\\ \dot{q}\\ \dot{r}\end{bmatrix}\right\} \tag{2-12}$$

Here the aerodynamic coefficients are updated based on the flight conditions such that it works across the flight envelope. The virtual input ν is generated using a proportional controller based on the error between rotational rates commanded by the pilot and measurements from the aircraft where a gain of $K_{p_{pqr}} = 10$ is used [7]. The resulting control architecture is presented in Figure 2-1.

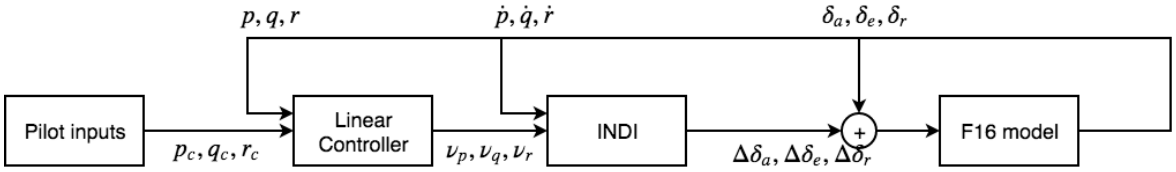


Figure 2-1: INDI controller architecture.

In the F-16 aerodynamic model rudder and ailerons are given in affine form, but the elevator is in non-affine form. To determine the coefficients C_{l_e}, C_{m_e}, C_{n_e} in affine form the aerodynamic moments generated by the elevator were plotted and the gradients were found that can be used by the INDI. From the NLR test flight data it was determined that the range of interest for angle of attack α is from -15 deg to 30 deg and for sideslip β ± 10 deg.

Pulse tests on the rotational rate channels confirmed that the controller is working as intended. As can be seen from Figure 2-2, other channels might be affected, when very challenging

commands are given. In addition, tracking of the commands will be lost in situations where there is no way to sustain a given rotational rate.

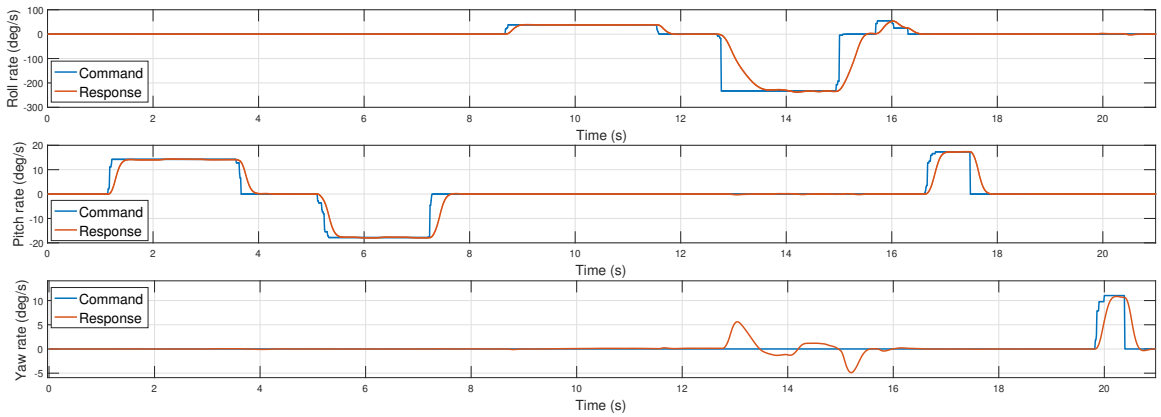


Figure 2-2: INDI pulse input commands.

2-3 Sideslip angle controller

When performing turns it is desirable to perform coordinated flight in order to minimize drag and increase passenger comfort. To this end the pilot or the control system needs to keep the sideslip angle at zero. To be able to create realistic reference trajectories, a sideslip angle controller was added to the system. This way, instead of controlling the yaw rate, it is possible to set a desired sideslip angle and the controller comes up with a yaw rate command for the INDI to maintain the sideslip angle. Now the pilot only needs to set the reference roll and pitch rates as can be seen in Figure 2-3.

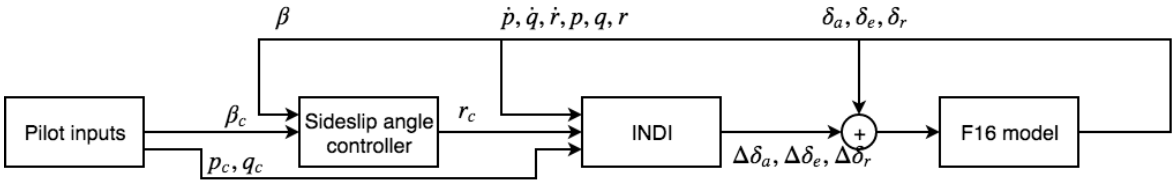


Figure 2-3: Sideslip angle control architecture.

A NDI control law for the sideslip angle is found by taking the derivative of the sideslip angle, shown in (2-14), and substituting in the accelerations from (2-15) [7].

$$\beta = \sin^{-1} \frac{v}{V_t} \tag{2-13}$$

$$\dot{\beta} = \frac{\dot{v}V_t - v\dot{V}_t}{V_t\sqrt{V_t^2 - v^2}} \tag{2-14}$$

$$\begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} = \frac{1}{m} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + \begin{bmatrix} 0 & -w & v \\ w & 0 & -u \\ -v & u & 0 \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} + \mathbf{T}_{bE} \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \tag{2-15}$$

After rearranging the equation, it is possible to come up with an equation for the yaw rate reference, as shown in (2-16), where a_x, a_y, a_z are the measured aircraft body accelerations. Furthermore, ν_β is the virtual input found from the error between the sideslip angle and the command with a PI controller using gains of $K_{P_\beta} = 2$ and $K_{I_\beta} = 0.2$ for the proportional and integral parts respectively.

$$r_{\text{ref}} = \left(\frac{-u}{\sqrt{V_t^2 - v^2}} \right)^{-1} \left[\nu_\beta - \frac{1}{\sqrt{V_t^2 - v^2}} \left(-\frac{uv}{V_t^2} a_x + \left(1 - \frac{v^2}{V_t^2} \right) a_y - \frac{vw}{V_t^2} a_z \right) - \frac{w}{\sqrt{V_t^2 - v^2}} p_{\text{ref}} \right] \tag{2-16}$$

Adding sideslip angle control to the rotational rate controller is based on time scale separation principle. In time scale separation faster and slower dynamics are distinguished and are placed in separate loops. In this case the outer loop sideslip controller generates reference rotational rates for the inner loop controller, which assumes the reference values to be constant. In turn the slow outer loop assumes that the commanded reference values are reached instantaneously [20].

2-4 Attitude representation using Quaternions

The University of Minnesota F-16 model uses Euler angles to represent the aircraft attitude, where the roll, pitch and yaw angle derivatives are found using (2-17). However, this attitude representation experiences a singularity when the aircraft is flying at a ±90 degree pitch angle which corresponds to the aircraft flying straight up or down. The aerobatic test flights that need to be tracked do include, among other maneuvers, loops where this singularity is encountered. Therefore a singularity free attitude representation needs to be implemented.

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \sin\phi \tan\theta & \cos\phi \tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \frac{\sin\phi}{\cos\theta} & \frac{\cos\phi}{\cos\theta} \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{2-17}$$

One such representation that is free of singularities is quaternions. With quaternions an additional redundant fourth parameter is added, to represent the attitude, that removes the singularity. All three parameter sets of attitude coordinates have kinematic differential equations that contain mathematical singularities and are nonlinear, but with quaternions the kinematic equation becomes (2-18) [21].

$$\begin{bmatrix} \dot{q}_0 \\ \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 0 & -p & -q & -r \\ p & 0 & r & -q \\ q & -r & 0 & p \\ r & q & -p & 0 \end{bmatrix} \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} \tag{2-18}$$

The transformation matrix to convert from Vehicle-carried normal Earth reference frame to body-fixed reference frame is then formed using (2-19).

$$\mathbf{T}_{bE} = \begin{bmatrix} q_0^2 - q_1^2 - q_2^2 + q_3^2 & 2(q_0q_1 + q_2q_3) & 2(q_0q_2 - q_1q_3) \\ 2(q_0q_1 - q_2q_3) & -q_0^2 + q_1^2 - q_2^2 + q_3^2 & 2(q_1q_2 + q_0q_3) \\ 2(q_0q_2 + q_1q_3) & 2(q_1q_2 - q_0q_3) & -q_0^2 - q_1^2 + q_2^2 + q_3^2 \end{bmatrix} \tag{2-19}$$

Replacing Euler angles with quaternions in the F-16 model allows flying the aircraft at any desired attitude. On the other hand, when plotting quaternion values it is not immediately clear to what attitude the values correspond. For this reason the quaternion values are then converted back to Euler angles using (2-20) that are much easier to interpret. When evaluating the tracking accuracy, error signals displaying difference in Euler angles are also shown instead of quaternions. With Euler angles a 180 deg error can occur even when the attitudes are close to each other due to the sign change at ± 180 deg. For example, when flying upside down the roll angle can oscillate between -180 deg and 180 deg. In order to evaluate attitude tracking performance of different controller settings a distance metric using quaternions was needed. Equation (2-21) was found as a suitable method and used for evaluation [22]. Here a value of 0 means the attitudes are perfectly aligned and a value of 1 corresponds to an attitude that is exactly in the opposite direction.

$$\begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} = \begin{bmatrix} \tan^{-1}(2(q_0q_1 + q_2q_3)/(q_0^2 - q_1^2 - q_2^2 + q_3^2)) \\ \sin^{-1}(2(q_0q_2 - q_1q_3)) \\ \tan^{-1}(2(q_0q_3 + q_1q_2)/(q_0^2 + q_1^2 - q_2^2 - q_3^2)) \end{bmatrix} \tag{2-20}$$

$$1 - |\mathbf{q}_{\text{ref}} \cdot \mathbf{q}_{\text{sim}}| \tag{2-21}$$

The new state vector where the Euler angles are replaced by quaternions is now presented in (2-22).

$$\mathbf{x} = [x, y, z, q_0, q_1, q_2, q_3, V_T, \alpha, \beta, p, q, r]^T \tag{2-22}$$

2-5 Improved engine model

The aerobatic test flights conducted by Netherlands Aerospace Centre (NLR) include sections and maneuvers that are performed with afterburners applied. The University of Minnesota F-16 model does not include afterburner dynamics, but instead the whole engine is modelled as a first order lag. Fortunately, the NASA report, based on which the aerodynamic model is constructed, also includes more accurate description of the engine dynamics. In Figure 2-4 the engine dynamics diagram is presented, where P_1 is engine power command based on the throttle position as shown in Figure 2-5, P_2 is the engine power command to the engine and P_3 is the engine power [2].

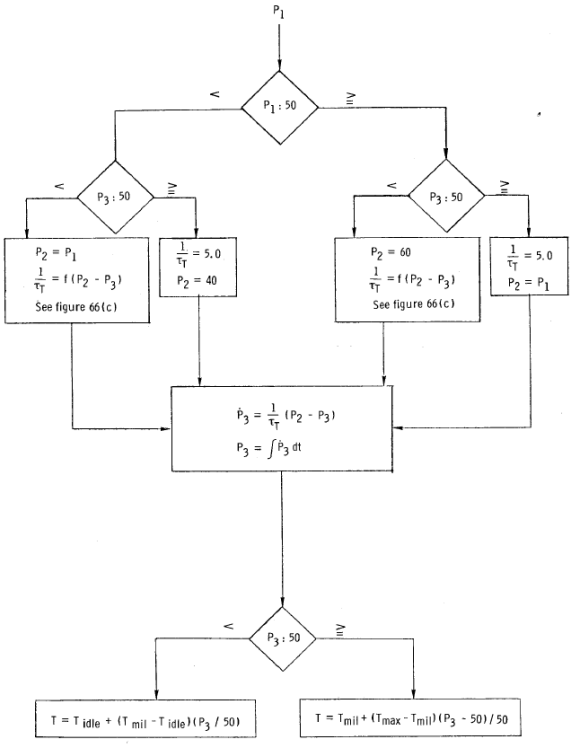


Figure 2-4: F-16 engine dynamics [2].

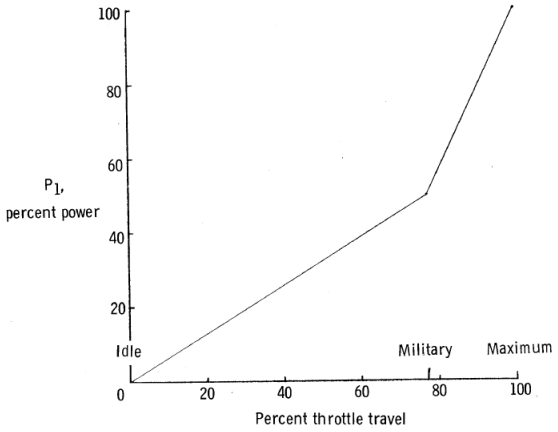


Figure 2-5: F-16 throttle level and power percentage relation [2].

Using these diagrams a more accurate model of engine dynamics was added to the simulation that includes afterburner dynamics and different engine response based on whether an increase or decrease in thrust is commanded.

Furthermore, the University of Minnesota F-16 model uses constant mass and moment of inertia values. However, in the 12 minute aerobatic flight test, where afterburners are used for extended periods, almost all of the fuel is consumed within that time. In this situation, using a constant mass and inertia can lead to large mismatch in the aircraft response and the tracking accuracy can suffer. For this reason, fuel flow was added to the engine model together with lookup tables for mass and inertia values based on data provided by NLR. The new extended state vector is presented in (2-23) with the mass state addition.

$$\mathbf{x} = [x, y, z, q_0, q_1, q_2, q_3, V_T, \alpha, \beta, p, q, r, m]^T \tag{2-23}$$

2-6 Trajectory generation

To test the performance of the controllers, a set of reference trajectories were generated with various levels of difficulty and covering the challenging flight conditions. These flights were performed by a F-16 test pilot using the previously described model and controllers implemented in Simulink. FlightGear flight simulator software was used to visualize the aircraft for the pilot using parameters computed in Simulink. This is shown in Figure 2-6.

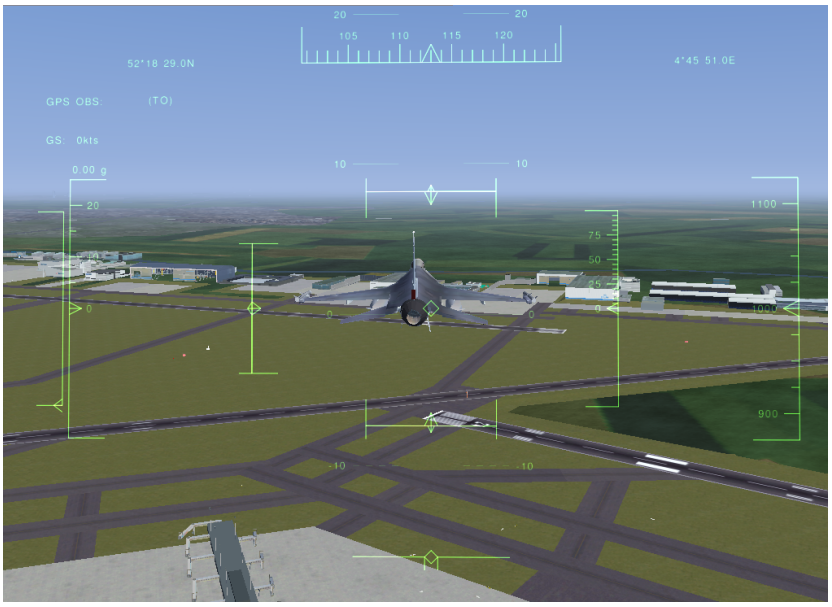


Figure 2-6: FlightGear simulation environment.

Various challenging maneuvers were recorded in order to be able to test the performance of the trajectory tracking controller. To start, a trajectory consisting of turns with high bank angles were generated as illustrated in Appendix B in Figure B-1. Next, fast and slow rate rolls were flown. First, using only ailerons as shown in Figure C-1 and then in the form of a barrel roll as shown in Figure D-1. Then a loop was performed that is presented in Figure E-1 and a Half Cuban Eight, as shown in Figure F-1, that combines loop and roll motions. Recovery was simulated, where the aircraft is climbing and loosing speed, followed by a turn and dive to regain speed and recover from stall. Finally, a trajectory, where numerous turns and rolls are combined, was generated and is presented in Figure H-1. These maneuvers were selected as they are well known aerobatic maneuvers that cover a wide range of challenging flight conditions. Furthermore, these elements are also present in NLR flight tests.

Summary of the recorded maneuvers is presented in the following list:

1. Turns
2. Aileron rolls
3. Barrel roll
4. Loop
5. Half Cuban Eights
6. Recovery
7. Combined maneuvers

Chapter 3

Model predictive control for trajectory tracking

In this chapter, first an overview of the NMPC controller is given in Section 3-1 together with the process for tuning the cost function weights and selection of the sampling time and prediction horizon. In Section 3-2 the way the NMPC and INDI were combined, to increase the tracking accuracy and reduce the computation time, is shown.

3-1 NMPC controller

Model Predictive Control (MPC) methods predict the future system states over a horizon using an internal model and determines the control inputs through the optimization of a cost function. This optimization results in a set of optimal open loop predicted control actions from which only the first term is applied to the system and the process is repeated again at the next time instance. During optimization system constraints can be enforced on the states and inputs to ensure that the physical limitations are not violated. In linear MPC, linear system models are used, which results in a convex optimization problem. For this fast solvers are available and global optimum can be guaranteed while in Nonlinear Model Predictive Control (NMPC), nonlinear system models are used, which can provide more accurate predictions, but global optimum solutions can no longer be guaranteed and more computational effort is required [23].

MPC algorithms require the system equations to be in discrete time form in order to be able to perform the optimization. The F-16 model system equations are constructed in continuous time format, so first, they needed to be discretized. Euler approximation, described in (3-1) to (3-3), was applied to perform the discretization, where h represents the sampling time.

$$\frac{\mathbf{x}(k+1) - \mathbf{x}(k)}{h} = \dot{\mathbf{x}} \tag{3-1}$$

$$\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u}) \tag{3-2}$$

$$\mathbf{x}(k+1) = \mathbf{x}(k) + h\ f(\mathbf{x}(k), \mathbf{u}(k)) \tag{3-3}$$

The optimization problem solved by NMPC is presented in (3-4). Here N_p and N_c represent the prediction horizon and control horizon length respectively and Q and R are the weight matrices for the states and inputs. The optimization is subject to constraints specified for the input increments, input magnitudes and state magnitudes and follows the system dynamics. Here $y_{\text{ref}}(k+i|k)$ represent the reference outputs that are desired to be reached with the MPC. These can be constant values over the horizon, if the reference is not known in advance, or can be updated for each step in the prediction horizon, when the reference signals are known, which is the situation with the F-16 reference tracks. Updating the reference signal each step in the prediction horizon can improve the tracking results as the controller is able to anticipate and prepare for maneuvers before they need to be executed.

$$\begin{aligned}
\underset{\Delta u}{\text{minimize}} \quad & J(k) = \sum_{i=1}^{N_p} \|\hat{\mathbf{y}}(k+i|k) - \mathbf{y}_{\text{ref}}(k+i|k)\|_{\mathbf{Q}} + \sum_{i=0}^{N_c-1} \|\Delta\mathbf{u}(k+i|k)\|_{\mathbf{R}} \\
\text{subject to} \quad & \\
& \mathbf{x}(k+1) = \mathbf{x}(k) + h\ f(\mathbf{x}(k), \mathbf{u}(k)) \\
& \mathbf{y}(k) = \mathbf{C}\mathbf{x}(k) \\
& \mathbf{u}(k) = \mathbf{u}(k-1) + \Delta\mathbf{u}(k) \\
& \Delta\mathbf{u}_{\min} \leq \Delta\mathbf{u} \leq \Delta\mathbf{u}_{\max} \\
& \mathbf{u}_{\min} \leq \mathbf{u} \leq \mathbf{u}_{\max} \\
& \mathbf{x}_{\min} \leq \mathbf{x} \leq \mathbf{x}_{\max}
\end{aligned} \tag{3-4}$$

The prediction horizon is illustrated in Figure 3-1[1], where it can be seen, that the controller is optimizing the inputs such that the predicted output matches the reference along the horizon.

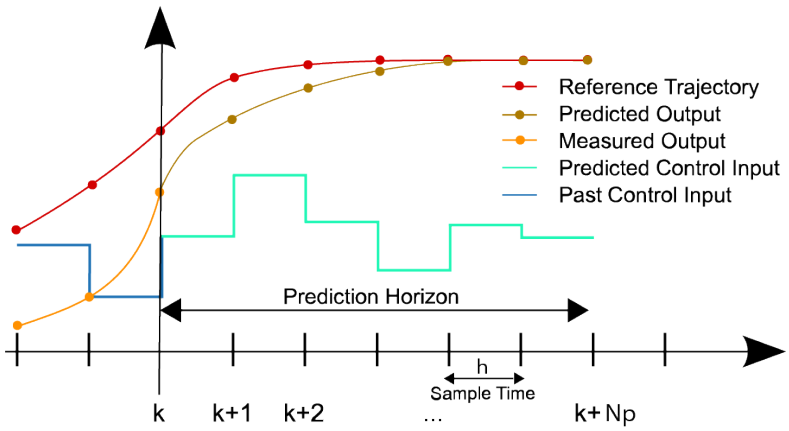


Figure 3-1: Model predictive control prediction horizon.

In the case of the F-16 model these constraints are the control surface actuator rate and angle limits, thrust magnitude limits and flight condition restrictions as specified in Section 2-1.

[1]Modified illustration based on: http://commons.wikimedia.org/wiki/File:MPC_scheme_basic.svg

Matlab function *fmincon* was used to solve the NMPC problem, which is an algorithm for optimizing nonlinear constrained multivariable functions. However, this algorithm requires combining the constraints as functions of input increments only, which required rewriting the input magnitude constraints. This is demonstrated in (3-5) to (3-9), where the resulting A_{cons} is a lower triangular matrix filled with ones.

$$\mathbf{u}(k) = \mathbf{u}(k-1) + \Delta\mathbf{u} \tag{3-5}$$

$$\begin{aligned} \mathbf{u}_{\min} \leq & \mathbf{u}(k) \leq \mathbf{u}_{\max} \\ \mathbf{u}_{\min} \leq & \mathbf{u}(k-1) + \Delta\mathbf{u}(k) \leq \mathbf{u}_{\max} \\ \mathbf{u}_{\min} - \mathbf{u}(k-1) \leq & \Delta\mathbf{u}(k) \leq \mathbf{u}_{\max} - \mathbf{u}(k-1) \end{aligned} \tag{3-6}$$

The input magnitude constraint needs to be satisfied at each step in the prediction horizon:

$$\begin{aligned} \mathbf{u}_{\min} - \mathbf{u}(k-1) \leq & \Delta\mathbf{u}(k) \leq \mathbf{u}_{\max} - \mathbf{u}(k-1) \\ \mathbf{u}_{\min} - \mathbf{u}(k-1) \leq & \Delta\mathbf{u}(k) + \Delta\mathbf{u}(k+1) \leq \mathbf{u}_{\max} - \mathbf{u}(k-1) \\ & \vdots \\ \mathbf{u}_{\min} - \mathbf{u}(k-1) \leq & \Delta\mathbf{u}(k) + \Delta\mathbf{u}(k+1) \ldots + \Delta\mathbf{u}(k+N_c) \leq \mathbf{u}_{\max} - \mathbf{u}(k-1) \end{aligned} \tag{3-7}$$

Using the property in (3-8) the constraints can be combined to matrix format shown in (3-9).

$$\mathbf{u}_{\min} \leq \mathbf{u}(k) \Leftrightarrow -\mathbf{u}(k) \leq -\mathbf{u}_{\min} \tag{3-8}$$

$$\begin{bmatrix} \mathbf{A}_{\text{cons}} \\ -\mathbf{A}_{\text{cons}} \end{bmatrix} \Delta\mathbf{u} \leq \begin{bmatrix} \mathbf{u}_{\max} - \mathbf{u}(k-1) \\ -(\mathbf{u}_{\min} - \mathbf{u}(k-1)) \end{bmatrix} \tag{3-9}$$

In the F-16 simulation model the actuator and engine model dynamics are separate from the flight dynamics model equations. In order to correctly predict the system output, it was necessary to modify the NMPC internal model to include these dynamics. This was done by changing the system inputs to actuator angle and thrust commands and by extending the system states with actuator angles and thrust level as shown in (3-10) and (3-11).

$$\mathbf{x} = [x, y, z, q_0, q_1, q_2, q_3, V_t, \alpha, \beta, p, q, r, m, \delta_T, \delta_a, \delta_e, \delta_r]^T \tag{3-10}$$

$$\mathbf{u} = [\delta_{T_c}, \delta_{a_c}, \delta_{e_c}, \delta_{r_c}]^T \tag{3-11}$$

The states, that are used for tracking, are presented in (3-12). Here it can be seen that velocity components V_x, V_y, V_z have been added which are determined from the model acceleration outputs, described in Table 2-2, and are used to guide the position tracking.

$$\mathbf{y} = [x, y, z, V_x, V_y, V_z, q_0, q_1, q_2, q_3, p, q, r, \delta_T, \delta_a, \delta_e, \delta_r]^T \tag{3-12}$$

It is desirable to have all the system inputs with similar magnitudes in order to avoid potential problems with the optimization algorithm. For this reason the actuator angle inputs were converted from degrees to radians and for the thrust command a percentage of maximum thrust was used. The NMPC controller system architecture is presented in Figure 3-2.

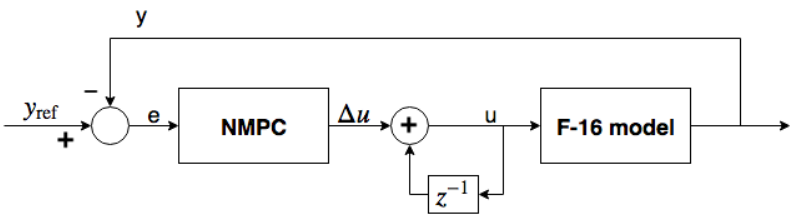


Figure 3-2: NMPC control architecture.

3-1-1 Sampling time and prediction horizon selection

The first controller parameter, that needs to be selected, is the sampling time h. Sampling time determines how often the controller checks the plant outputs and computes the control signals. Slower sampling times are preferred, as it leads to lower computational loads, but it still needs to be fast enough to capture the system dynamics. A general rule of thumb is to select a sampling time such that there is around 4-10 samples per rise time of the fastest system dynamic. The fastest dynamic in the F-16 model is the actuator dynamics for which step input tests were performed to determine the rise time. The sampling times and corresponding rounded samples per rise time are shown in Table 3-1.

Table 3-1: Sampling time selection

Sampling time (s)	Number of samples per rise time (-)
0.01	11
0.02	5
0.03	4
0.04	3

From this test a sampling time of 0.03s was chosen to keep the computational load low.

3-1-2 Weights and prediction horizon tuning

Tuning of the weights on the states in the cost function and the prediction horizon length was an iterative process, where tracking was implemented and tuned on the faster dynamics first while ignoring the slower states. When the tracking for fast states was tuned the next set of states was added to the cost function. To determine the horizon length tracking accuracies of tests with varying horizons were compared.

First, tracking of the reference actuator signals only was tested to ensure that the actuator dynamics modelling was implemented correctly. The reference signals generated for these initial tests involved flying straight and level in trim conditions to ensure that the NMPC is also capable of stabilizing the aircraft before moving further to dynamic maneuvers. For actuator angle tracking a sampling time of 0.03s as determined previously was used with a prediction horizon of $N_p = 5$ and the weights for actuator signals were set to 1 as seen in

(3-13) with other states currently not included in the tracking.

$$\mathbf{Q} = diag\left(\begin{bmatrix} x & y & z & V_x & V_y & V_z & q_0 & q_1 & q_2 & q_3 & p & q & r & \delta_T & \delta_a & \delta_e & \delta_r \\ - & - & - & - & - & - & - & - & - & - & - & - & - & - & 1 & 1 & 1 \end{bmatrix}\right) \tag{3-13}$$

The weights on the input increments were all set to 1 as there is no reason to prioritize one input over others.

$$\mathbf{R} = diag\left(\begin{bmatrix} \Delta\delta_{T_c} & \Delta\delta_{a_c} & \Delta\delta_{e_c} & \Delta\delta_{r_c} \\ 1 & 1 & 1 & 1 \end{bmatrix}\right) \tag{3-14}$$

From Figure 3-3 and Figure 3-4 it can be seen that the NMPC is able to track the reference actuator angles. This allowed to move to tracking other reference signals and start varying the prediction horizon lengths in order to determine the optimal.

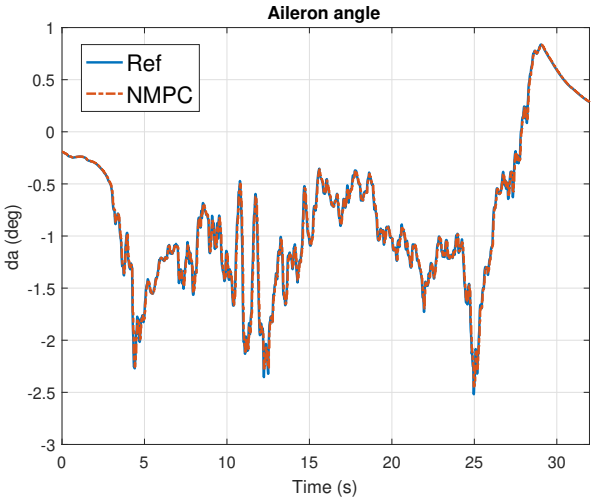


Figure 3-3: Aileron angle reference tracking.

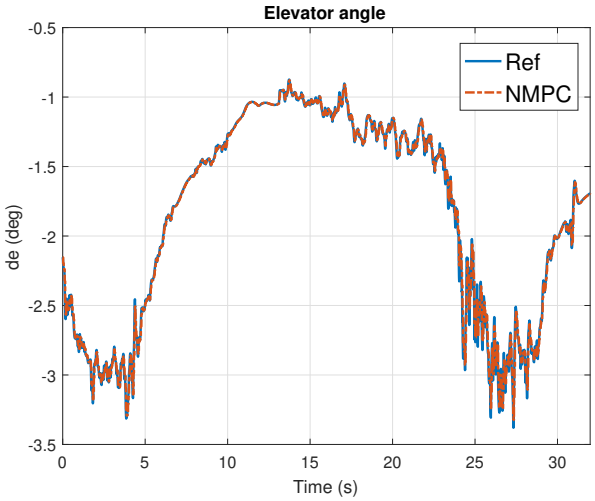


Figure 3-4: Elevator angle reference tracking.

Next, rotational rates were added for tracking with a weight of 10. The general approach for choosing the weights for the states was to have higher values for slower states. This would allow more deviation from the reference in the fast states in order to ensure that the tracking of slow states is accurate.

$$\mathbf{Q} = diag\left(\begin{bmatrix} x & y & z & V_x & V_y & V_z & q_0 & q_1 & q_2 & q_3 & p & q & r & \delta_T & \delta_a & \delta_e & \delta_r \\ - & - & - & - & - & - & - & - & - & - & 10 & 10 & 10 & - & 1 & 1 & 1 \end{bmatrix}\right) \tag{3-15}$$

The tracking was again first tested on a trim flight trajectory and then on a more difficult track with fast rolls and pitching motions with constant thrust, which are shown in Figure 3-5.

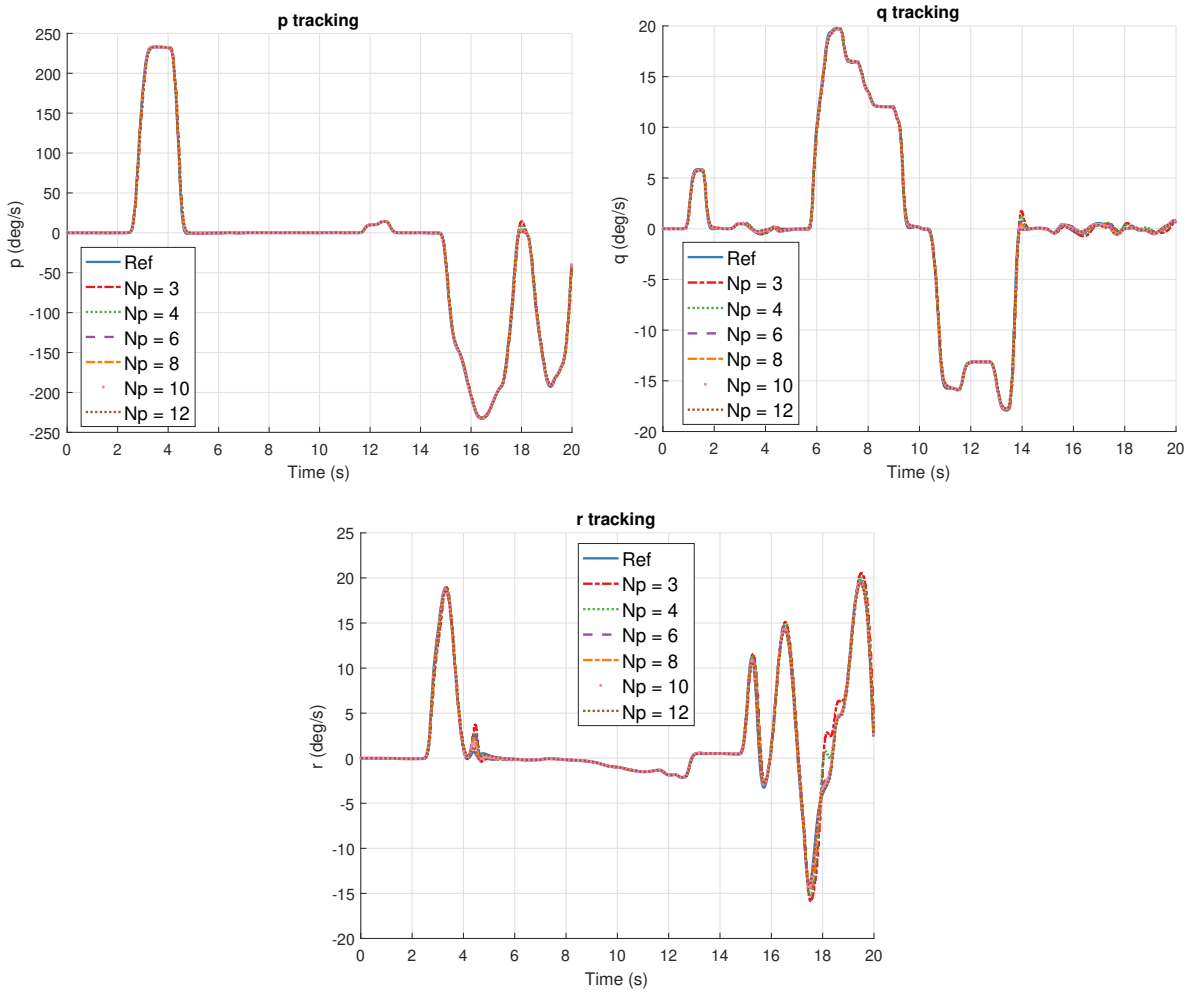


Figure 3-5: Rotational rate tracking with NMPC, varying N_p lengths.

For attitude tracking the weights were increased to 1000. The larger increase in weights was due to different units in quaternions compared to rotational rates and actuator angles. The quaternion values remain between -1 and 1 while rotational rates can range up to $\pm 2\pi$. Using a high weight would ensure that attitude tracking is strongly prioritized over rotational rate tracking. The attitude tracking for rolling and pitching motion can be seen in Figure 3-6.

$$\mathbf{Q} = diag\left(\begin{bmatrix} x & y & z & V_x & V_y & V_z & q_0 & q_1 & q_2 & q_3 & p & q & r & \delta_T & \delta_a & \delta_e & \delta_r \\ - & - & - & - & - & - & 10^3 & 10^3 & 10^3 & 10^3 & 10 & 10 & 10 & - & 1 & 1 & 1 \end{bmatrix}\right) \tag{3-16}$$

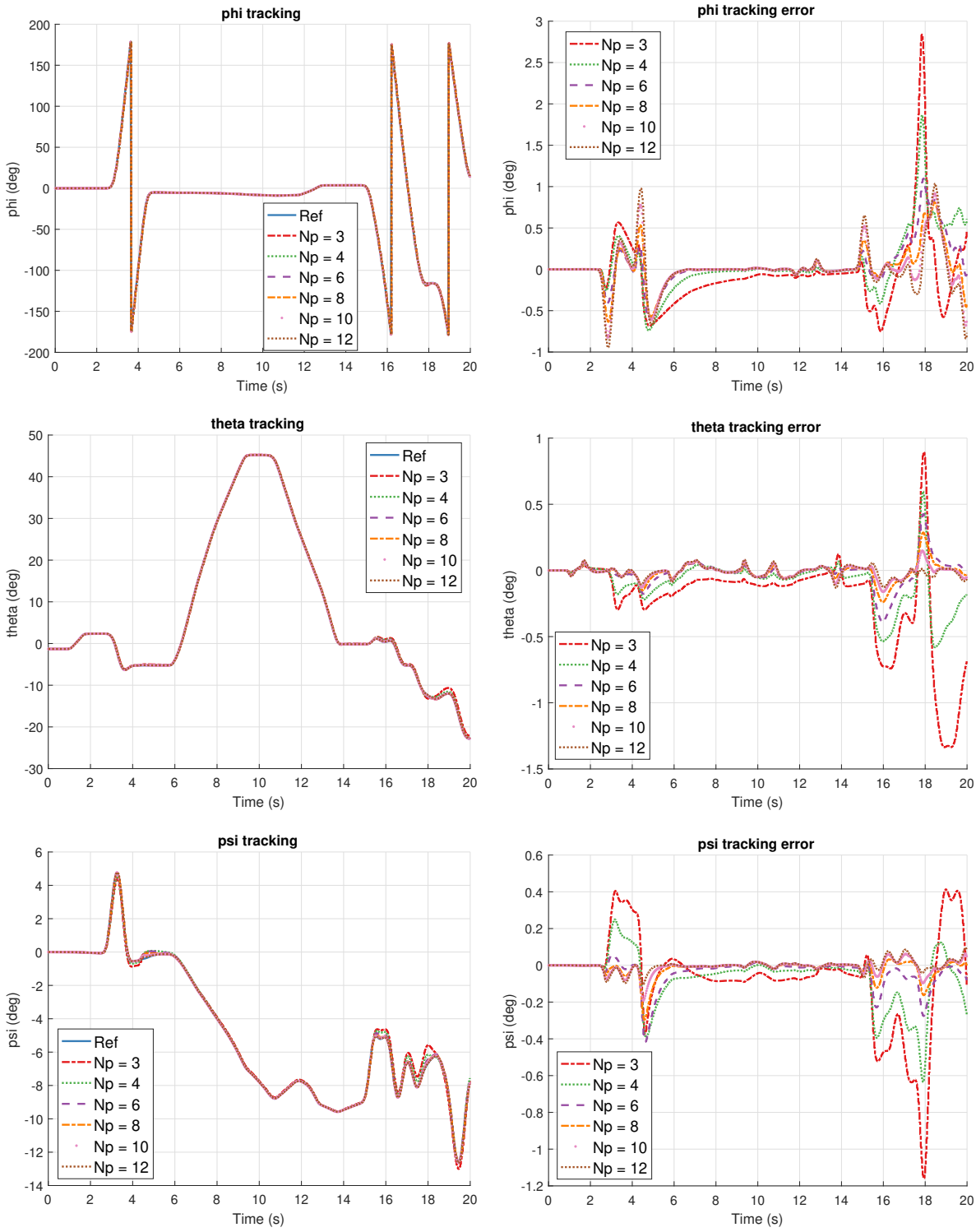


Figure 3-6: Attitude angle tracking with NMPC, varying N_p lengths.

Next, attitude tracking was tested for the situation, where actuator angle reference signals are unavailable. From Figure 3-7 it can be seen that even without actuator angle reference

signals the NMPC is able to follow attitude reference.

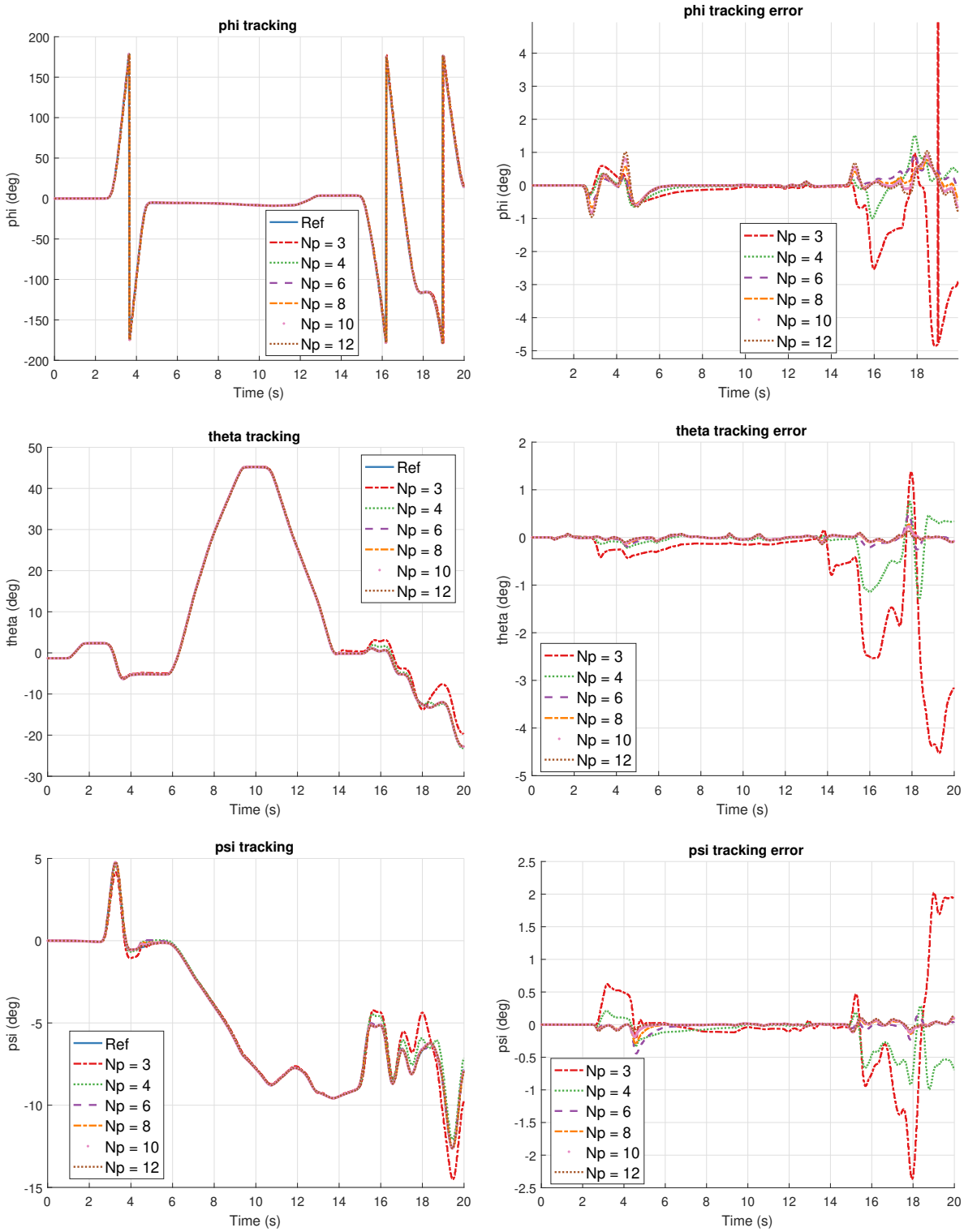


Figure 3-7: Attitude angle tracking with NMPC, varying N_p lengths, no actuator angle reference.

From Figure 3-8 and Figure 3-9, it can be seen that the tracking accuracy is almost identical for the cases where the actuator reference signals are available and when they are not. Further, it can be seen that for tracking the attitude, a prediction horizon of 8 samples is producing most accurate results.

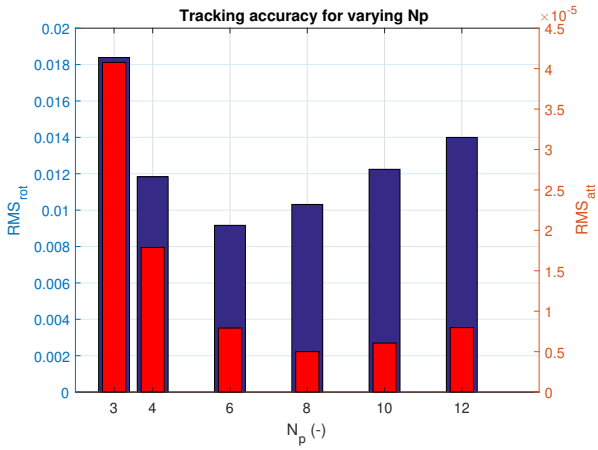


Figure 3-8: Tracking accuracy for varying N_p, rotational rate and attitude RMS comparison.

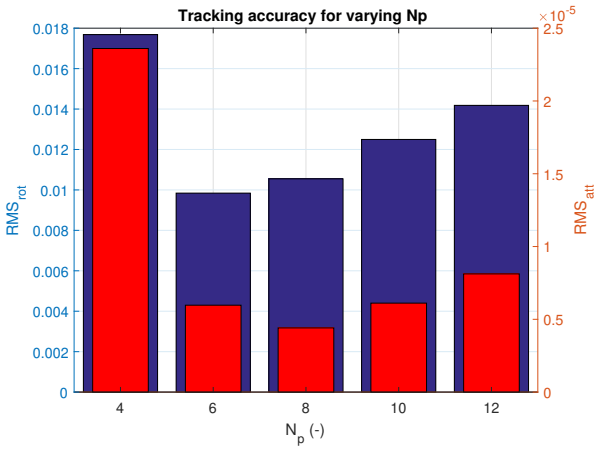


Figure 3-9: Tracking accuracy for varying N_p, rotational rate and attitude RMS comparison, no actuator reference.

With these tests concluded, it was possible to move towards adding position tracking in addition to the attitude tracking.

3-1-3 Move blocking implementation

When trying to achieve position tracking more difficulty was experienced. Initially only the position information and the total velocity was used, but as soon as the weights would be increased to a value where they would start influencing the tracking, the results would diverge and tracking would be lost. Then the idea was to add directional velocity information found from the acceleration outputs, which could guide the controller. This way tracking the velocities would help to decrease the errors in the position tracking. However, this addition still did not solve the diverging results. The root of the problem was found, when investigating the controller horizon length. The engine and position information are slow dynamics. For example with a first order lag constant of $1/5$ as shown in Section 2-5, the engine settling time to a step input is $0.8s$. With a sampling time of $0.03s$ and horizon of $N_p = 8$ the controller horizon covers only $0.24s$ which means that the controller can not correctly evaluate the thrust input effect on the position states and a longer horizon is required. However, each extra step in the horizon introduces 4 extra optimization variables for the algorithm to solve for. This can quickly increase the required computation time to infeasible lengths, which is also why this approach was not investigated in the first place.

The solution to this is to use a method called move blocking. In move blocking the control inputs are updated at every sample time in the beginning of the horizon, but are then kept constant over a number of instances at the end of the horizon. This is illustrated in Figure 3-10. Here N_p is the total horizon length and N_h is the amount of sampling instances the control inputs are kept constant for. Other variables introduced by this method are N_f and

N_s, which correspond to the number of fast control updates and number of slow control updates respectively. Move blocking allows to increase the prediction horizon, while keeping the number of optimization variables low and manageable, which then leads to a reduction in number of iterations performed by the algorithm and in turn reduced computation time [24].

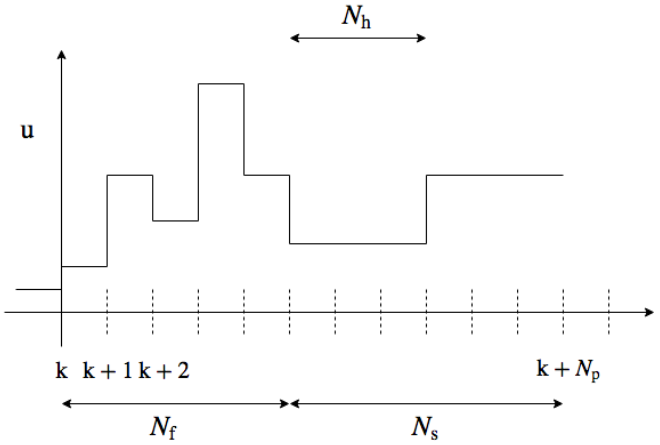


Figure 3-10: Move blocking illustration.

Move blocking can be implemented as a matrix ($\mathbf{T}$) that assigns the input increments to the corresponding inputs and keeps the values constant at the required locations. This is presented in (3-17) and (3-18).

$$\mathbf{u} = \mathbf{u}_0 + \mathbf{T}\Delta\mathbf{u} \tag{3-17}$$

$$\begin{bmatrix} \mathbf{u}(k) \\ \mathbf{u}(k+1) \\ \mathbf{u}(k+2) \\ \mathbf{u}(k+3) \\ \vdots \\ \mathbf{u}(k+N_p) \end{bmatrix} = \mathbf{u}(k-1) + \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 \\ & & & \vdots & \end{bmatrix} \begin{bmatrix} \Delta\mathbf{u}(k) \\ \Delta\mathbf{u}(k+1) \\ \Delta\mathbf{u}(k+2) \\ \Delta\mathbf{u}(k+3) \\ \Delta\mathbf{u}(k+4) \end{bmatrix} \tag{3-18}$$

After implementing move blocking, it was possible to increase the prediction horizon to greater lengths while still achieving feasible simulation times. The weights in the cost function followed the reasoning of attitude tracking, position states were assigned a weight of 3, velocity a weight of 1 and thrust level a weight of 1. Smaller magnitudes compared to the attitude state weights is again used due to different units - position and velocity errors are represented in ft and ft/s respectively.

$$\mathbf{Q} = diag\left(\begin{bmatrix} x & y & z & V_x & V_y & V_z & q_0 & q_1 & q_2 & q_3 & p & q & r & \delta_T & \delta_a & \delta_e & \delta_r \\ 3 & 3 & 3 & 1 & 1 & 1 & 10^3 & 10^3 & 10^3 & 10^3 & 10 & 10 & 10 & 1 & 1 & 1 & 1 \end{bmatrix}\right) \tag{3-19}$$

The reference track used to test position tracking is similar to one used for attitude tracking. However, this time the thrust level is not kept constant, but instead is covering the entire

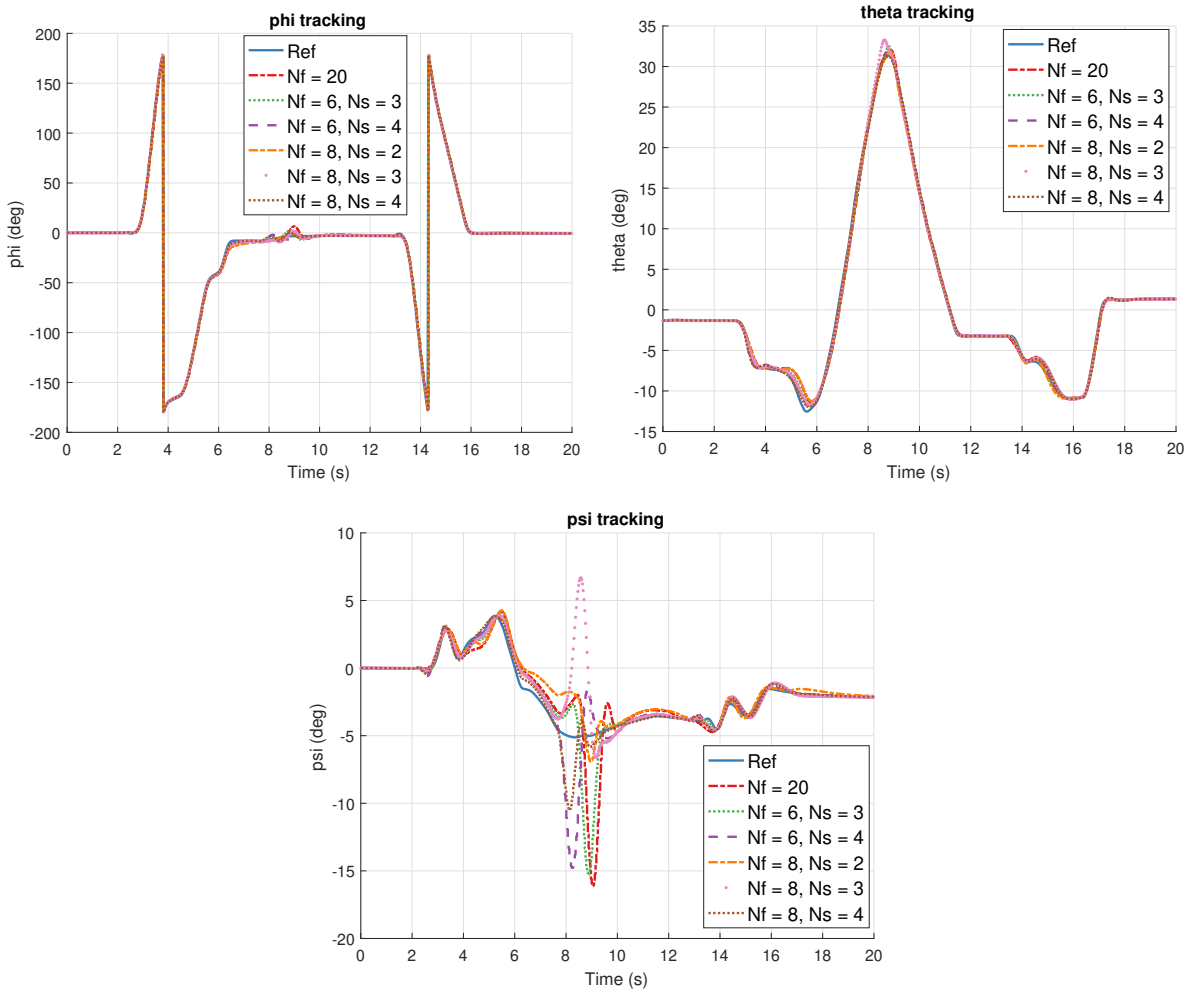


Figure 3-11: Attitude tracking with NMPC, varying N_p lengths.

range. Figure 3-11 presents the reference attitude, while Figure 3-12 demonstrates the position tracking accuracy.

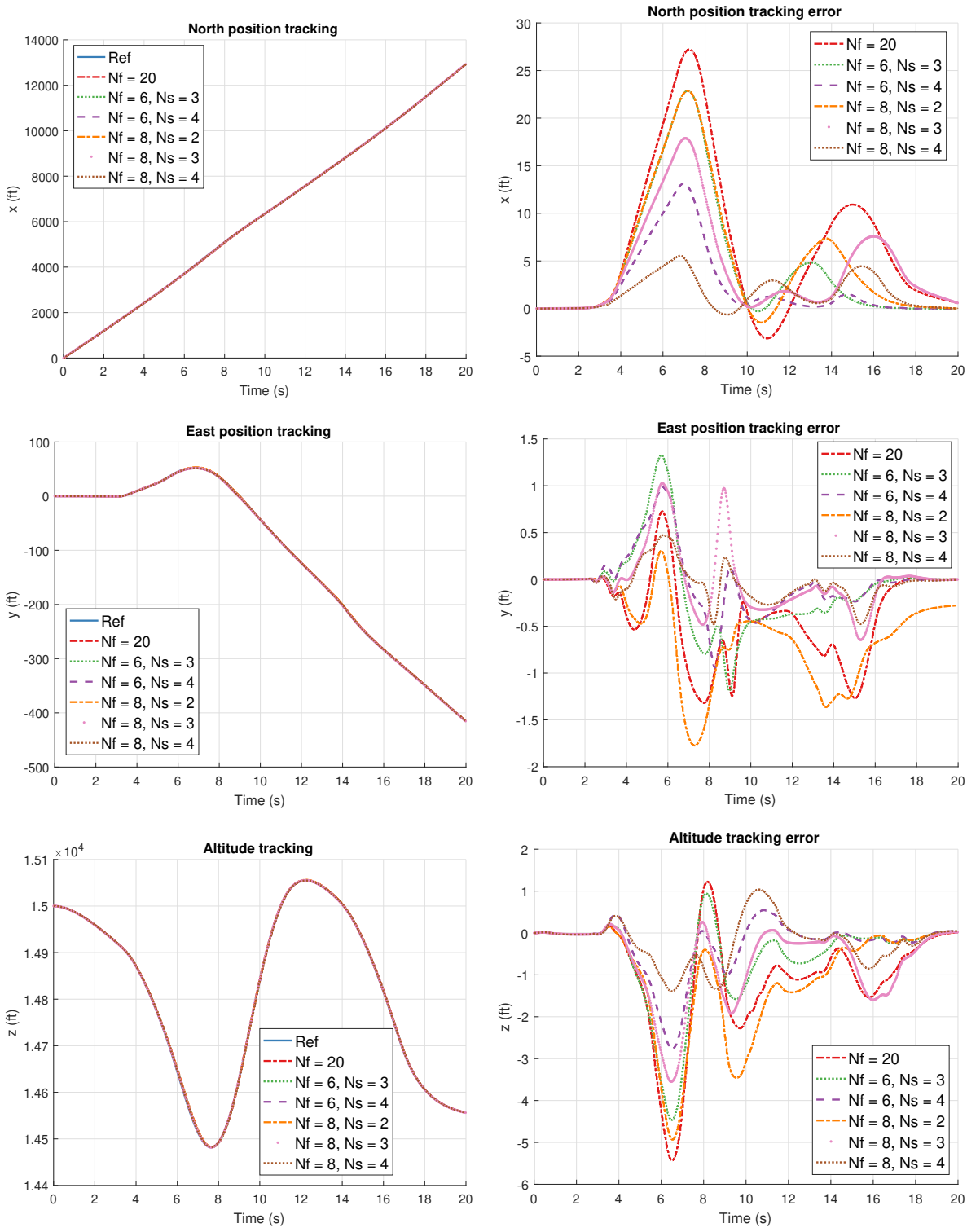


Figure 3-12: Position tracking with NMPC, varying N_p lengths.

In Figure 3-12 it can be seen that the position tracking error is largest in sections where rolling is performed which is expected since these are very dynamic maneuvers to track with rolling

rates close to 250 deg/s. A test where move blocking was not implemented is also included in the results with a horizon length of $N_p = 20$. From Table 3-2 it can be seen that this results in a simulation time that is 4x longer compared to a move blocking implementation with similar horizon length.

Table 3-2: NMPC horizon length and simulation time comparison.

Nf	Ns	Nh	Nc	Np	RMS att (-)	RMS pos (ft)	T_{sim} (min)
20	0	-	20	20	$1.4 \cdot 10^{-3}$	6.1	228
8	4	5	12	28	$2.0 \cdot 10^{-4}$	1.4	148
6	4	5	10	26	$6.0 \cdot 10^{-4}$	2.5	79
8	3	5	11	23	$9.0 \cdot 10^{-4}$	3.9	76
6	3	5	9	21	$9.0 \cdot 10^{-4}$	4.7	58
8	2	5	10	18	$3.0 \cdot 10^{-4}$	4.9	45

When comparing Figure 3-13 and Figure 3-14, it can be seen that the tracking accuracy can even increase when actuator angle and thrust reference signals are not used. However, this increase in accuracy results in larger differences in actuator angle signals, which in turn means decreased emulation performance.

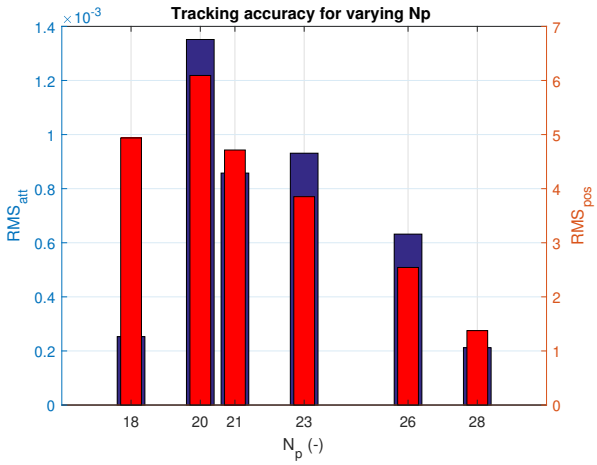


Figure 3-13: Tracking accuracy for varying N_p, position and attitude RMS comparison.

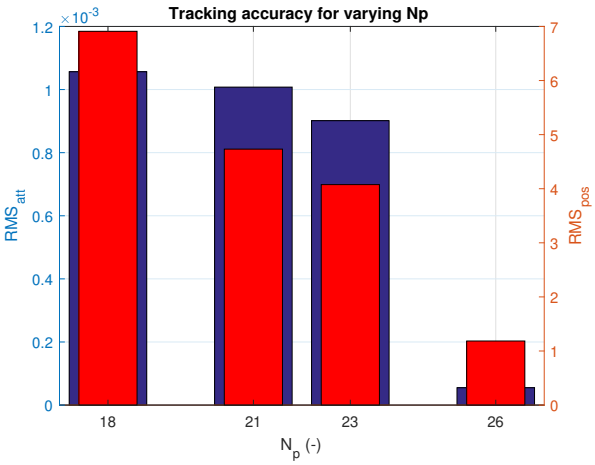


Figure 3-14: Tracking accuracy for varying N_p, position and attitude RMS comparison, no actuator reference.

For example, in Figure 3-15 it can be seen that there are extra oscillations present in the signals where thrust reference was not made available for the controller.

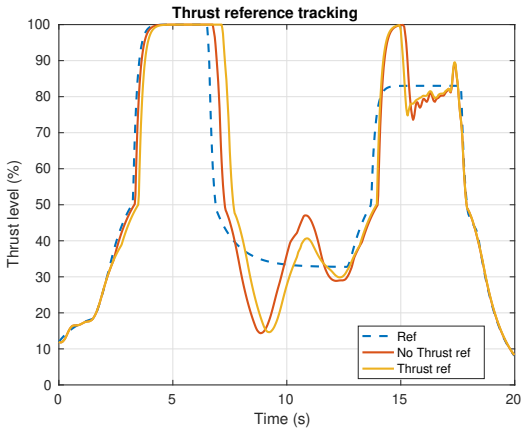


Figure 3-15: Thrust level tracking with NMPC, with and without reference comparison, $N_f = 6$, $N_s = 4$, $N_h = 5$.

From the set of prediction horizon tuning tests it was concluded, that the slow dynamics of the engine requires a large prediction horizon to completely capture the aircraft response. However, it was determined that acceptable level of accuracy can already be obtained with a shorter horizon, where the computation times are also still manageable. With the computation time and the tracking accuracy in mind, a move blocking structure of $N_f = 6$, $N_s = 4$, $N_h = 5$ was selected for the NMPC. The choice is a trade-off between accuracy and simulation time and therefore the user can always refine the horizon length to match their accuracy and time requirements.

3-1-4 Solver selection

The simulation speed is also highly dependent on the optimization algorithm. For *fmincon* function in Matlab there are three algorithms to choose from 'Interior point', 'Sequential Quadratic Programming (SQP)' and 'Active set'. In order to determine the fastest solver a set of timed tests was conducted for which the results are presented in Table 3-3. 'Active set' was chosen as the solver for the NMPC, because it is the fastest of the three. 'Active set' gains the speed advantage through the algorithm's ability to take large steps during the optimization process, which can lead to finding the optimum faster. There is no significant difference in the optimization accuracy between the algorithms.

Table 3-3: Optimization algorithm simulation time comparison.

Algorithm	$T_{\text{sim}}(min)$
Interior Point	120
SQP	38
Active set	32
Active set (no warm start)	32

Initially *fminsearch* function was also tested which is used for optimizing unconstrained nonlinear multivariable functions and uses 'Nelder-Mead' algorithm. However, this algorithm

showed much longer computation times compared to the constrained optimization methods and therefore was not considered when tests with NMPC were conducted.

An additional method that can be applied to speed up the simulation time includes using a 'warm start' for the optimization. Here the optimal solution for one time step is used as the starting point of the next optimization routine. This is based on the assumption that the optimization problem for current time instant is a slightly perturbed version of the problem solved at a previous instant and therefore the optimal solution is likely to also remain close to the previous solution [23]. The comparison between using 'warm start' and not using it is also shown in Table 3-3. However, here it can be seen that there is no difference between using 'warm start' and not using it. This is likely due to the incremental form used in the NMPC. The controller optimizes for the mostly small changes in inputs and therefore just using zeros, meaning no change in inputs, as the starting point of the optimization is also not far from the optimal solution.

3-1-5 Disturbance observer

The trajectory tracking is performed in a simulated environment, where the plant model is known and available to be used by the controller, which is why it can also be assumed that there is no significant mismatch between the plant and controller model. However, the plant model is in continuous time, while the controller model is in discrete time, which can introduce numerical inaccuracies due to Euler discretization. These numerical inaccuracies could accumulate to a constant offset in the tracking results. For this reason a disturbance observer was added to the controller as shown in (3-20) [25]. This is illustrated in Figure 3-16. Here $\hat{d}$ is the observer state and L is the observer gain.

$$\hat{\mathbf{d}}(k+1) = \hat{\mathbf{d}}(k) + \mathbf{L}(\mathbf{y}(k) - \mathbf{C}\mathbf{x}(k) - \hat{\mathbf{d}}(k)) \tag{3-20}$$

The disturbance observer was added to the position and attitude states, which are the slowest dynamics, so the disturbance observer is able to react and capture the offset errors. Tests were conducted to determine the gain of the observer. From Table 3-4 it can be seen that a gain of 0.1 performed the best.

Table 3-4: Disturbance observer gain tuning

Observer gain	RMS att (-)	RMS pos (ft)
0.1	$1.41 \cdot 10^{-7}$	0.144
0.5	$1.39 \cdot 10^{-7}$	0.146
0.9	$1.60 \cdot 10^{-7}$	0.167

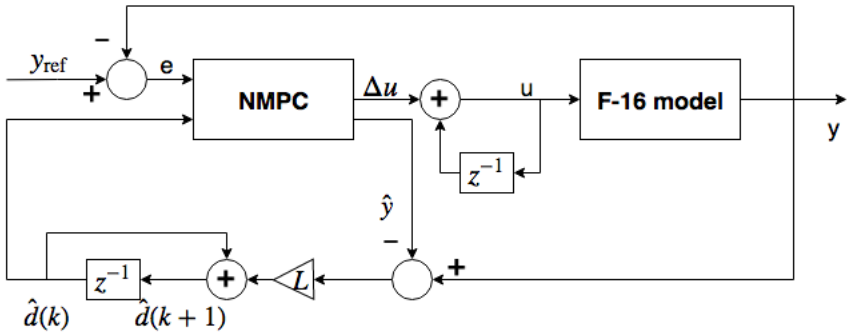


Figure 3-16: Disturbance observer and NMPC architecture.

3-2 NMPC-INDI controller

In order to improve the simulation time it was proposed to combine the NMPC controller together with Incremental Nonlinear Dynamic Inversion (INDI). The idea is to apply feedback linearization in the form of INDI that would stabilize the aircraft and control the fast aircraft dynamics while the NMPC would control the slower dynamics. This way a slower sampling time could be used for the NMPC resulting in a reduction in the number of optimizations performed and the fast control updates required to stabilize the aircraft are handled by the INDI. Instead of controlling the actuator angle commands the NMPC will control the rotational rate commands that are then used by the INDI as can be seen in Figure 3-17.

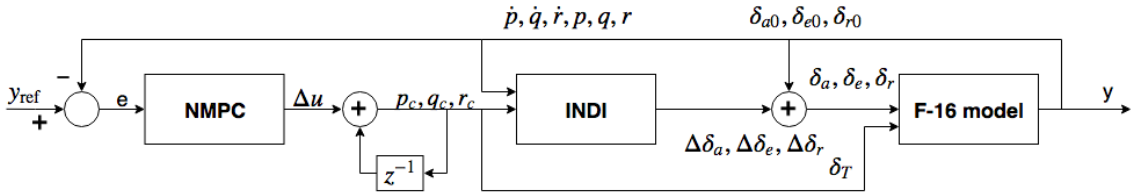


Figure 3-17: NMPC-INDI controller architecture.

Initially the objective was to completely remove actuator angles and associated aerodynamic coefficient lookup tables from the NMPC model. However, further inspection of the equations of motions revealed that even though with INDI the actuators can be replaced in the rotational dynamics, the actuator angles will still be present in the translation equations when determining the body accelerations as shown in (3-21).

$$\begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} = \frac{1}{m} \begin{bmatrix} X(\alpha, \beta, \delta_a, \delta_e, \delta_r ...) \\ Y(\alpha, \beta, \delta_a, \delta_e, \delta_r ...) \\ Z(\alpha, \beta, \delta_a, \delta_e, \delta_r ...) \end{bmatrix} + \begin{bmatrix} 0 & -w & v \\ w & 0 & -u \\ -v & u & 0 \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} + \mathbf{T}_{bE} \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \tag{3-21}$$

In the situation where actuator angle reference signals are available it is possible to use those reference values in the NMPC controller over the prediction horizon with the assumption that the actuator angles remain close to the reference. This way it is possible to combine NMPC and INDI as long as the actuator reference signals are available.

The new states, inputs and outputs of the NMPC-INDI controller model are presented in (3-22) to (3-24). Here it can be seen that the actuator reference signals are used when determining the system state in the prediction horizon.

$$\mathbf{x} = [x, y, z, q_0, q_1, q_2, q_3, V_t, \alpha, \beta, p, q, r, m, \delta_T, \delta_{a_{\text{ref}}}, \delta_{e_{\text{ref}}}, \delta_{r_{\text{ref}}}]^T \tag{3-22}$$

$$\mathbf{u} = [\delta_{T_c}, p_c, q_c, r_c]^T \tag{3-23}$$

$$\mathbf{y} = [x, y, z, V_x, V_y, V_z, q_0, q_1, q_2, q_3, p, q, r, \delta_T]^T \tag{3-24}$$

New control inputs for the NMPC are now the thrust command and rotational rate commands. For this it was necessary to determine the relation between the rotational rate commands and rotational rates resulting from the INDI feedback linearization. Step inputs were given to the INDI and the plant rotational rates were measured as seen in Figure 3-18. This system response was then approximated using a second order transfer function as in (3-25) and the coefficients were determined to be $\zeta = 0.86$ and $\omega_n = 15.8$.

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \begin{bmatrix} p_c \\ q_c \\ r_c \end{bmatrix} \tag{3-25}$$

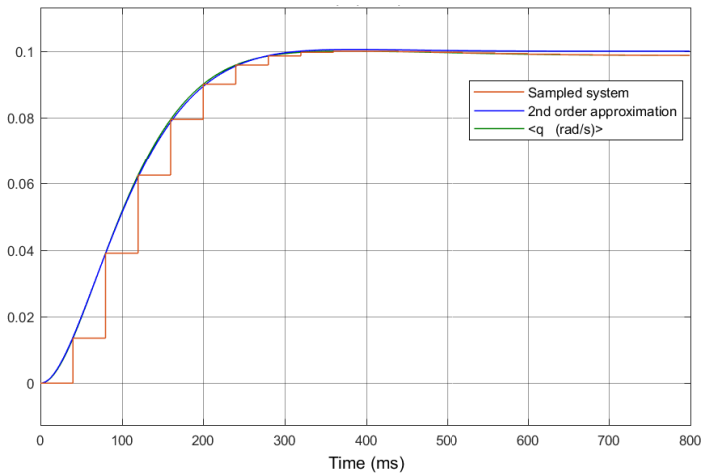


Figure 3-18: INDI step input approximation.

3-2-1 Sampling time selection

As before, the sampling time of the NMPC was determined through the number of samples per rise time of the fastest dynamic. The fastest dynamic that the NMPC is controlling is now

the rotational rates instead of the actuator dynamics. From the step response in Figure 3-18 the number of samples per rise time for varying sampling times were determined and are presented in Table 3-5. This shows that with the addition of INDI controller it is possible to reduce the sampling time from previous 0.03s to 0.04s. If all other parameters are kept the same this should lead to a 25% improvement in computation time.

Table 3-5: Sampling time selection with INDI

Sampling time (s)	Number of samples per rise time (-)
0.02	9
0.03	6
0.04	4
0.05	3

3-2-2 Prediction horizon tuning

Again for NMPC-INDI tests with various prediction horizon lengths were conducted to compare tracking accuracy and simulation time. The reference track from Section 3-1-3 was reused. When combining NMPC and INDI it is important that the approximation obtained in Section 3-2 is not violated as that would lead to inaccuracies in the predictions made by the NMPC over the horizon and tracking could be lost. For this reason the weights for rotational rates were increased to 100.

$$\mathbf{Q} = diag\left(\begin{bmatrix} x & y & z & V_x & V_y & V_z & q_0 & q_1 & q_2 & q_3 & p & q & r & \delta_T \\ 3 & 3 & 3 & 1 & 1 & 1 & 10^3 & 10^3 & 10^3 & 10^3 & 100 & 100 & 100 & 1 \end{bmatrix}\right) \tag{3-26}$$

Figure 3-19 displays the position tracking accuracy for various controller horizon lengths where value in brackets shows the length of N_h. Using a slower sampling time means that a longer horizon can be covered time-wise when using the same amount of prediction instances. As can be seen, this has led to a reduction in position errors.

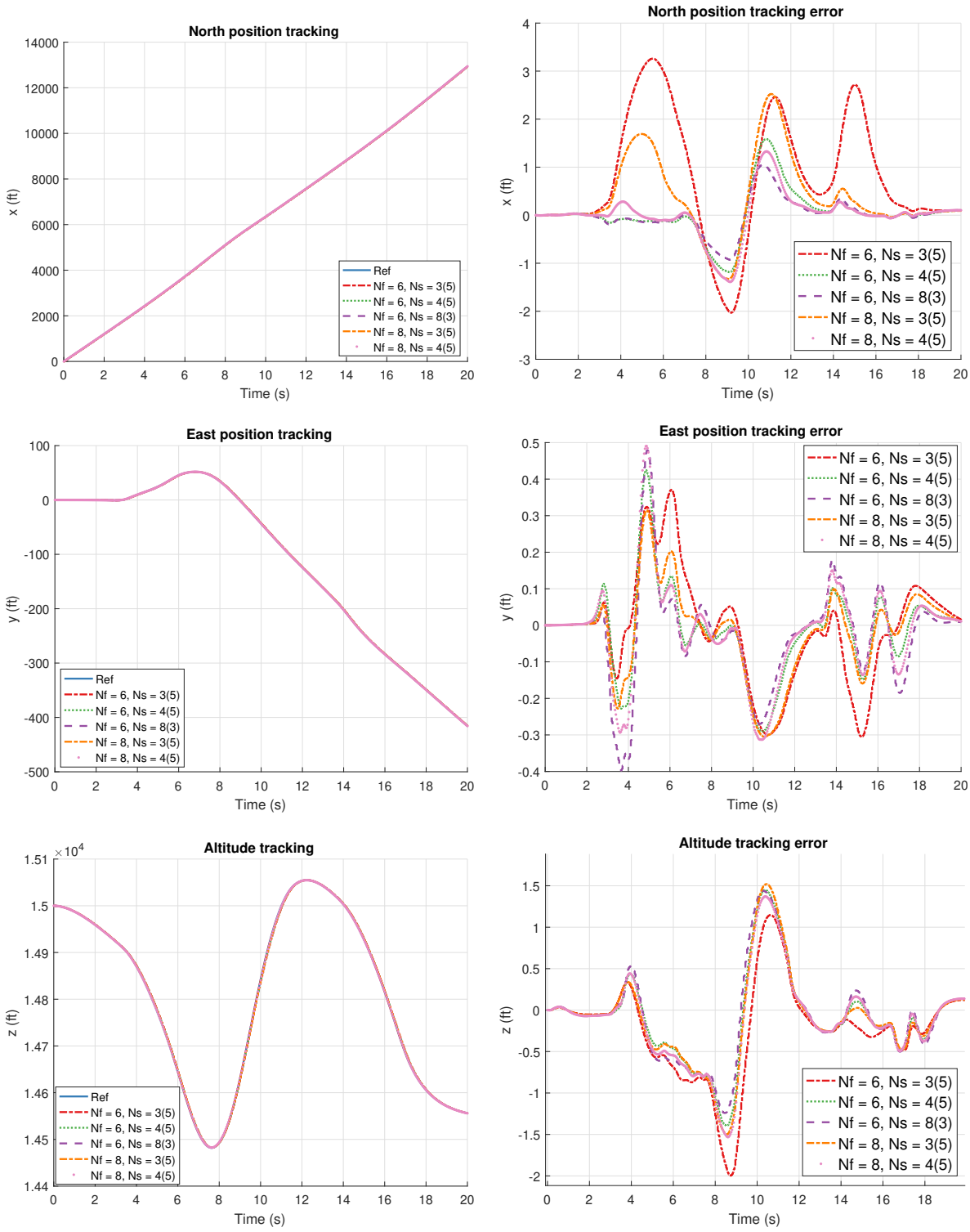


Figure 3-19: Position tracking error, NMPC-INDI with varying horizon length.

From Figure 3-20 it can be seen that position tracking improves very little when using prediction horizons longer than $N_p = 26$ and at the same time attitude tracking accuracy starts

to decrease faster. From this a move blocking structure of $N_f = 6$, $N_s = 4$, $N_h = 5$ was also chosen for the NMPC-INDI controller.

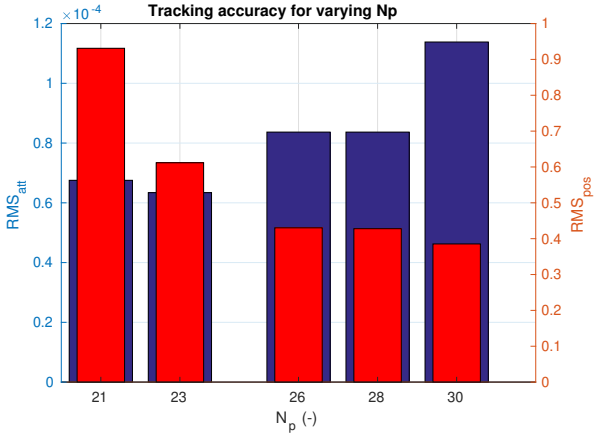


Figure 3-20: Tracking accuracy for varying N_p, position and attitude RMS comparison, NMPC-INDI.

In Table 3-6 the comparison between the NMPC controller and NMPC-INDI controller position tracking accuracy and simulation time taken is presented. It can be seen that the position signal Root Mean Square (RMS) error is approximately $70-80\%$ smaller when using NMPC-INDI compared to NMPC with equivalent horizon length while the simulation time is around 40% faster.

Table 3-6: NMPC-INDI horizon length and simulation time comparison and change with respect to NMPC.

Nf	Ns	Nh	Nc	Np	RMS att	RMS pos	Change (%)	T (min)	Change (%)
6	8	3	14	30	$1.14 \cdot 10^{-4}$	0.385	-	111	-
8	4	5	12	28	$8.37 \cdot 10^{-5}$	0.428	-68.9	74	-50.0
6	4	5	10	26	$8.37 \cdot 10^{-5}$	0.430	-83.1	48	-39.2
8	3	5	11	23	$6.34 \cdot 10^{-5}$	0.612	-84.1	44	-42.1
6	3	5	9	21	$6.75 \cdot 10^{-5}$	0.931	-80.3	32	-44.8

However, this improved performance comes at the cost of controller robustness. Once the feedback linearization differs too much from the second order approximation the incorrect predictions made by NMPC can lead to increasing oscillations in rotational rates which eventually crash the simulation. This can be seen in Figure 3-21. Due to incorrect choice of cost function weights and initialization, the initial pitch rate command can not be achieved by the INDI which leads to inaccurate predictions by the NMPC and leads to growing overshoots that eventually affect all rotational rates.

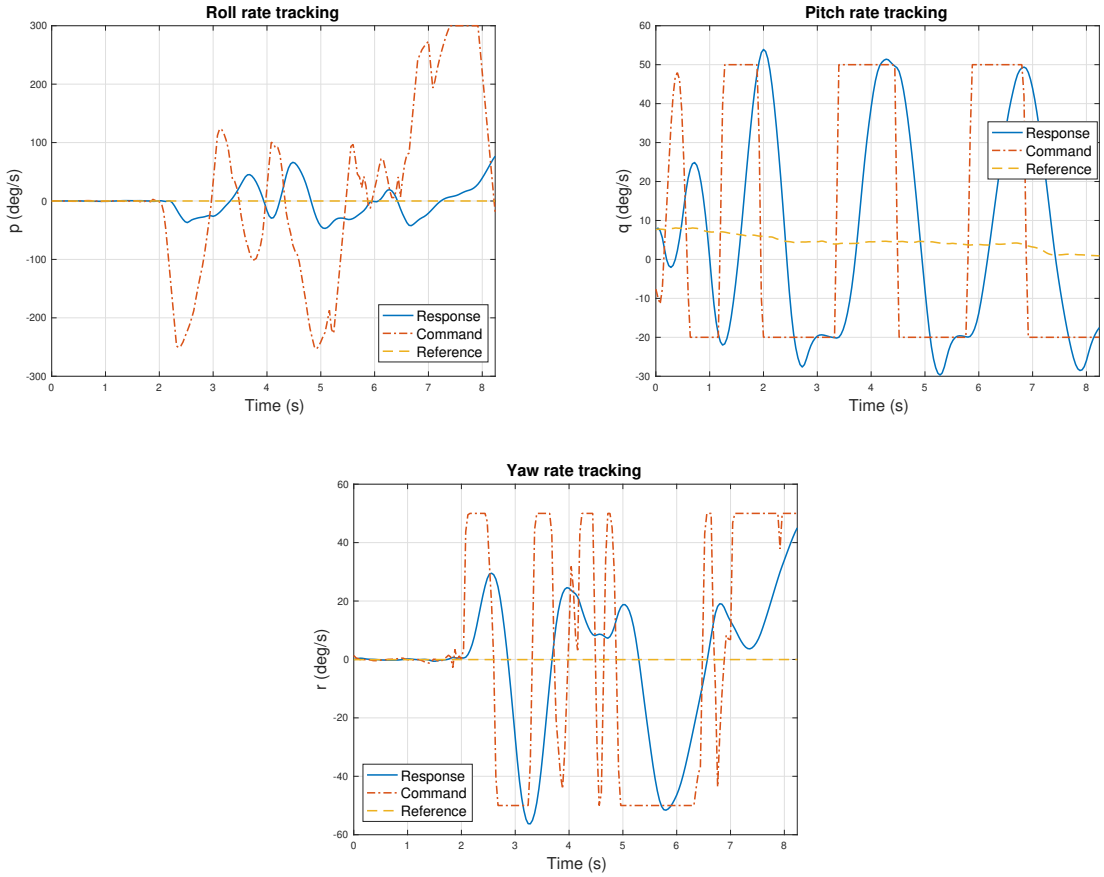


Figure 3-21: INDI linearization instability example.

Chapter 4

Results

In this chapter the controller performance is evaluated by tracking multiple reference maneuvers in Section 4-1. Then the tracking performance is tested in situations where model mismatch is present in Section 4-2. The controllers are validated by following NLR reference trajectory in Section 4-3 followed by a discussion of the controller performance in Section 4-4.

4-1 NMPC and NMPC-INDI controller performance

After tuning the controller prediction horizon lengths and cost function weights it was possible to test the tracking performance on the generated reference trajectories. For each trajectory three tests were conducted: using NMPC with reference actuator and thrust signals, using NMPC without reference actuator and thrust signals and finally using the NMPC-INDI controller. The tracking accuracy of the three cases were then compared. Furthermore, tracking the reference trajectories allowed to evaluate the way these controllers respond to various situations and flight conditions. In these tests there is no mismatch between the aircraft model used to generate the reference tracks, model in the simulation and model used by the controller to predict the response over the horizon.

4-1-1 Track: Turns

For the first trajectory tracking test a reference with the least challenging maneuvers was selected. This track consists of three high bank angle turns and is illustrated in Figure B-1. The trajectory was tracked with previously determined tuning weights and prediction horizons. Main tracking results are presented in Figure 4-1 and Figure 4-2, while a full set of tracked signals is presented in Appendix B.

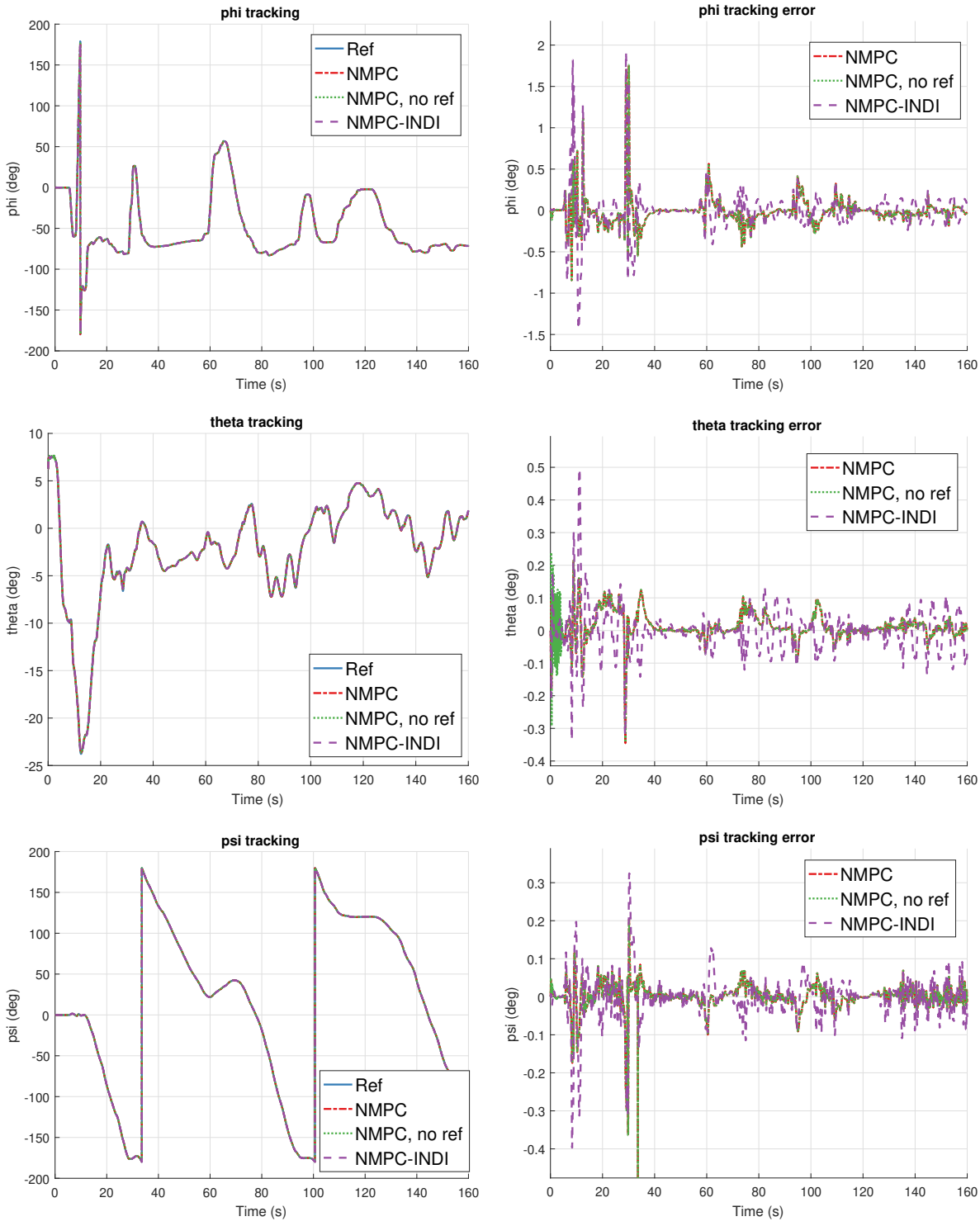


Figure 4-1: Attitude angle tracking, track: Turns.

As can be seen, the attitude is tracked very closely in all cases and error signals remaining mostly below 1 deg. When looking at the graph displaying psi (ψ) tracking error signals, there are occasions where the signal extends outside of the figure. In these locations the error

signal displays ±360 degrees due to the attitude signal changing sign. Similar effect can also be seen for other attitude signals in the upcoming graphs. The position errors remain below ±1 ft while flying at a total velocity close to 900 ft/s. The reference track was generated without using much variation in the thrust, as seen in Figure 4-3, which allowed to obtain such an accurate tracking for the position.

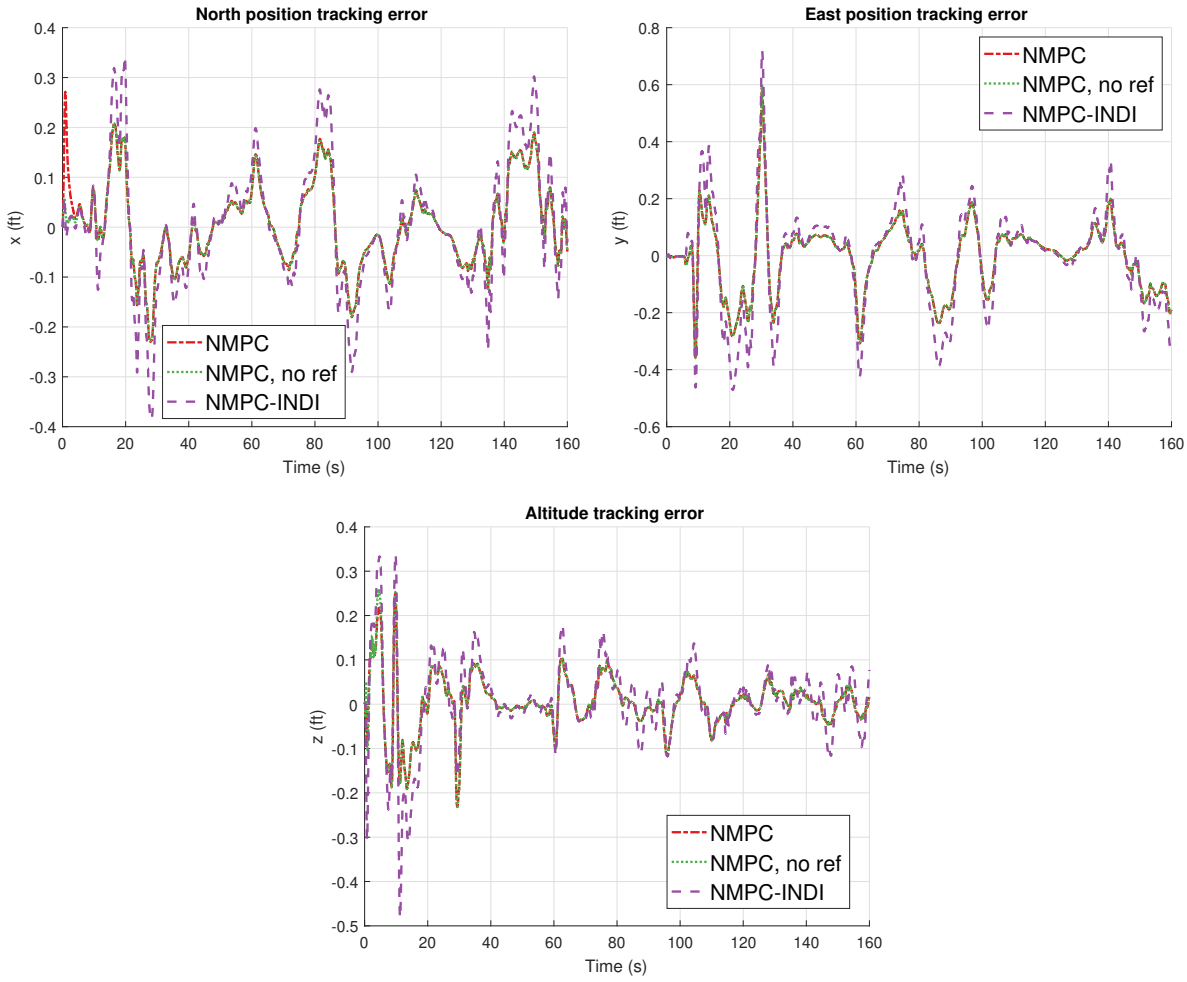


Figure 4-2: Position tracking error, track: Turns.

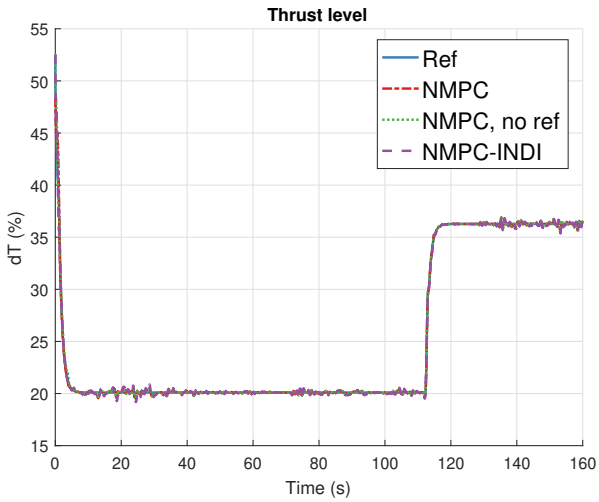


Figure 4-3: Thrust level tracking, track: Turns.

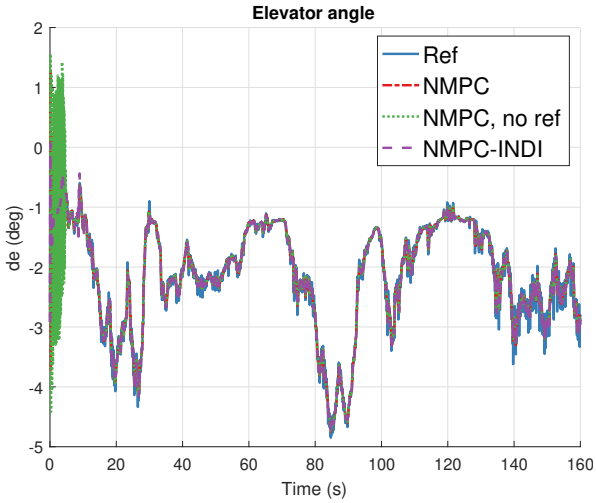


Figure 4-4: Elevator angle tracking, track: Turns.

The test also revealed an issue, which can occur when either the thrust level is initialized incorrectly or when the aircraft can not reduce the velocity enough by only using the thrust. Then the aircraft starts to pitch up and down in order to reduce the velocity further and match with the reference position. This can be seen in Figure 4-4, where the aileron angle oscillations are present at the beginning of the simulation for the NMPC without input references. For the NMPC with input references, the thrust was initialized at a lower value which eliminated this effect. Otherwise, attitude and position tracking errors show that the accuracy is nearly identical for the NMPC with and without reference input signals.

4-1-2 Track: Aileron rolls

Next, the tracking accuracy was tested while performing rolls at high rotational rates. Again, the attitude and position tracking results are shown in Figure 4-5 and Figure 4-6 while a complete overview of the results is presented in Appendix C. Four rolls were performed at rotational rates close to 200 deg/s. From Figure 4-5 it can be seen that even at these high rotational rates the aircraft attitude angles remain within 2 deg from the reference values with the highest peaks at the locations where rolls were performed which is as expected.

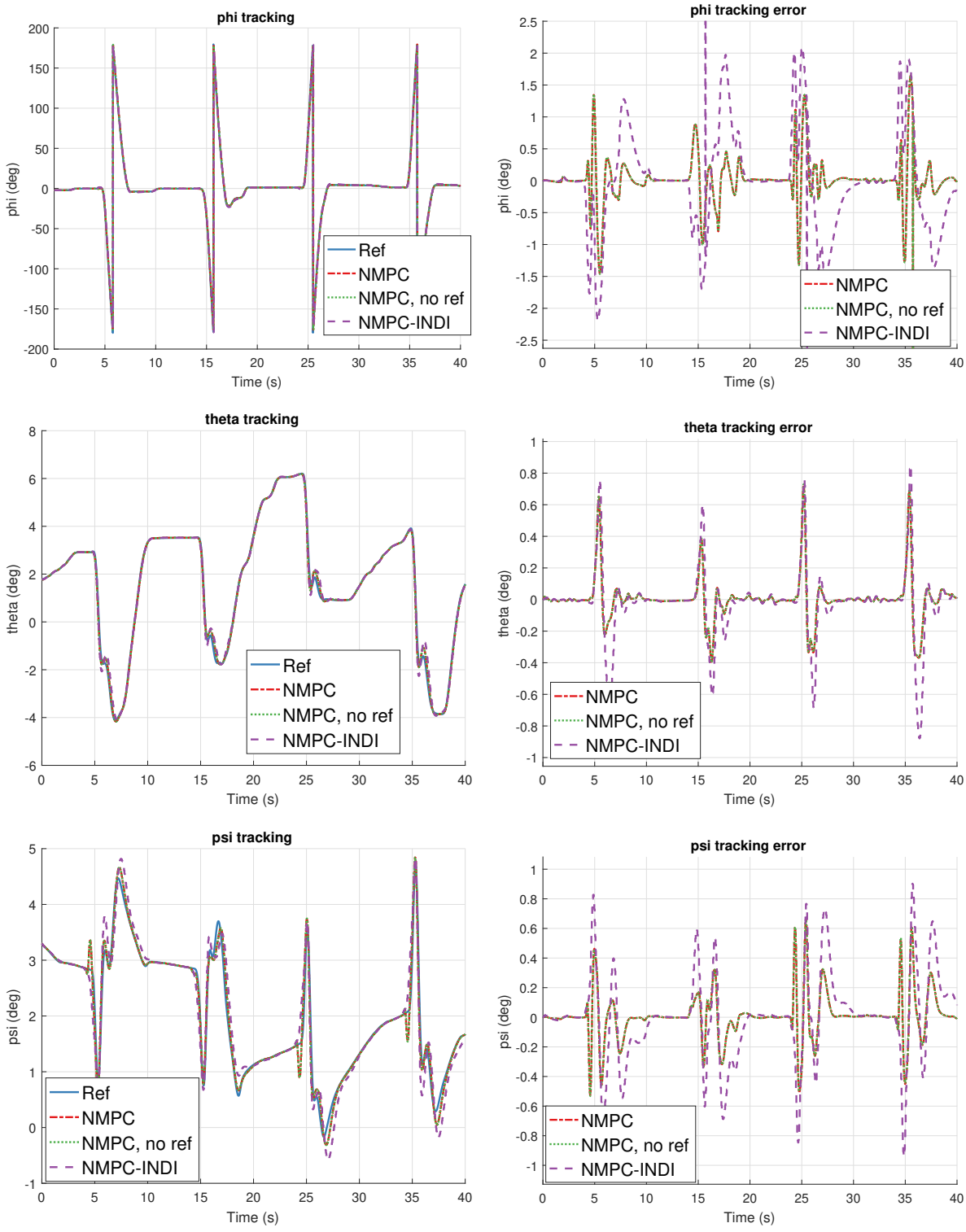


Figure 4-5: Attitude angle tracking, track: Aileron rolls.

When looking at the position tracking, again the values remain within 0.5 ft from the reference. Larger errors are in the east and altitude directions while the errors in the north direction

are smaller, meaning that when performing rolls the position tracking is affected more in the plane orthogonal to the total velocity.

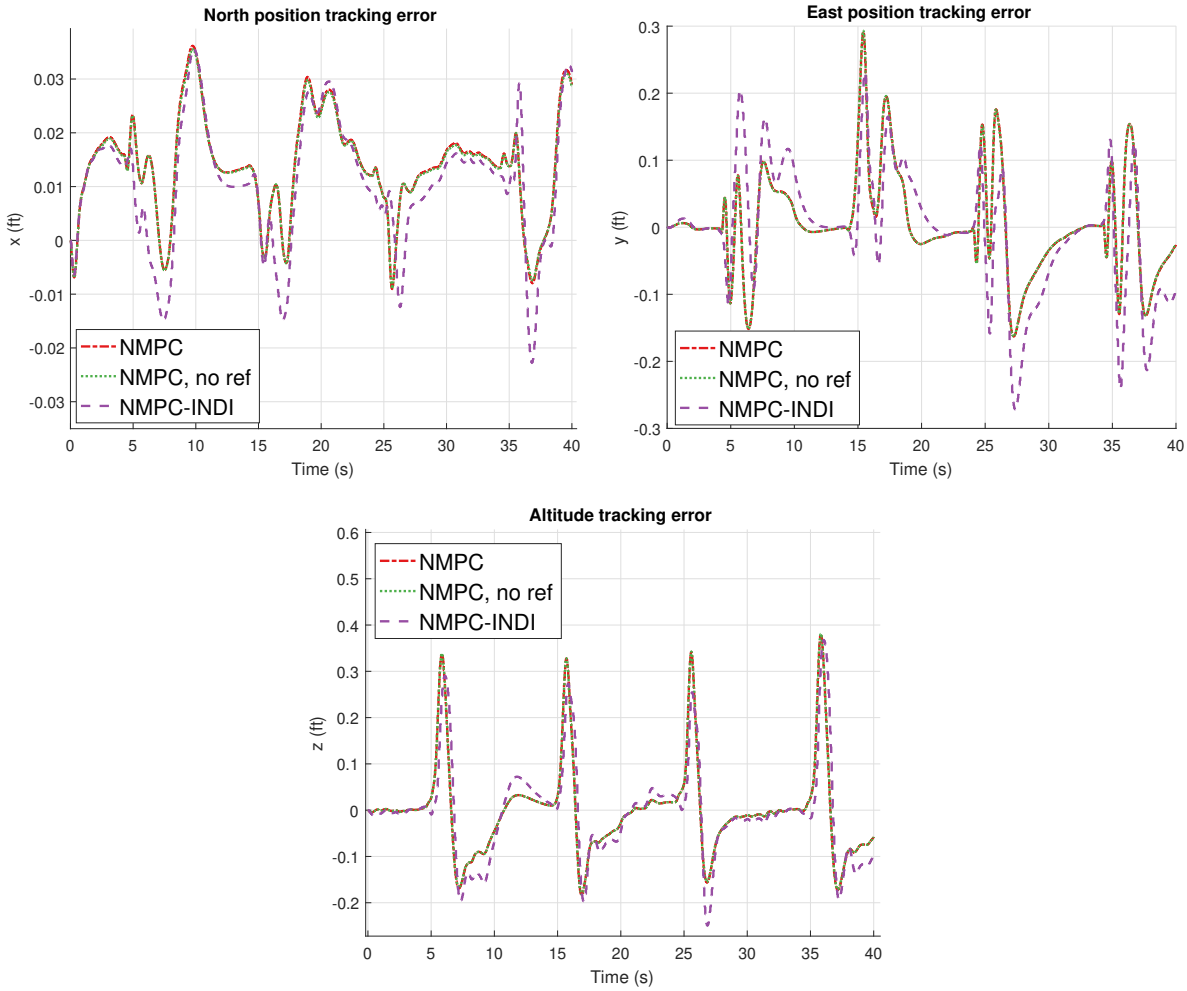


Figure 4-6: Position tracking error, track: Aileron rolls.

This test shows that very accurate tracking can be maintained even at rotational rates close to performance limits.

4-1-3 Track: Barrel rolls

Barrel rolls were conducted as the next test, where the rotational rates are slower but the maneuver is less isolated to just rolling. From the results in Figure 4-7 it can be seen that less challenging rotational rates allow to track attitude even more accurately and the difference from the reference values is below half a degree.

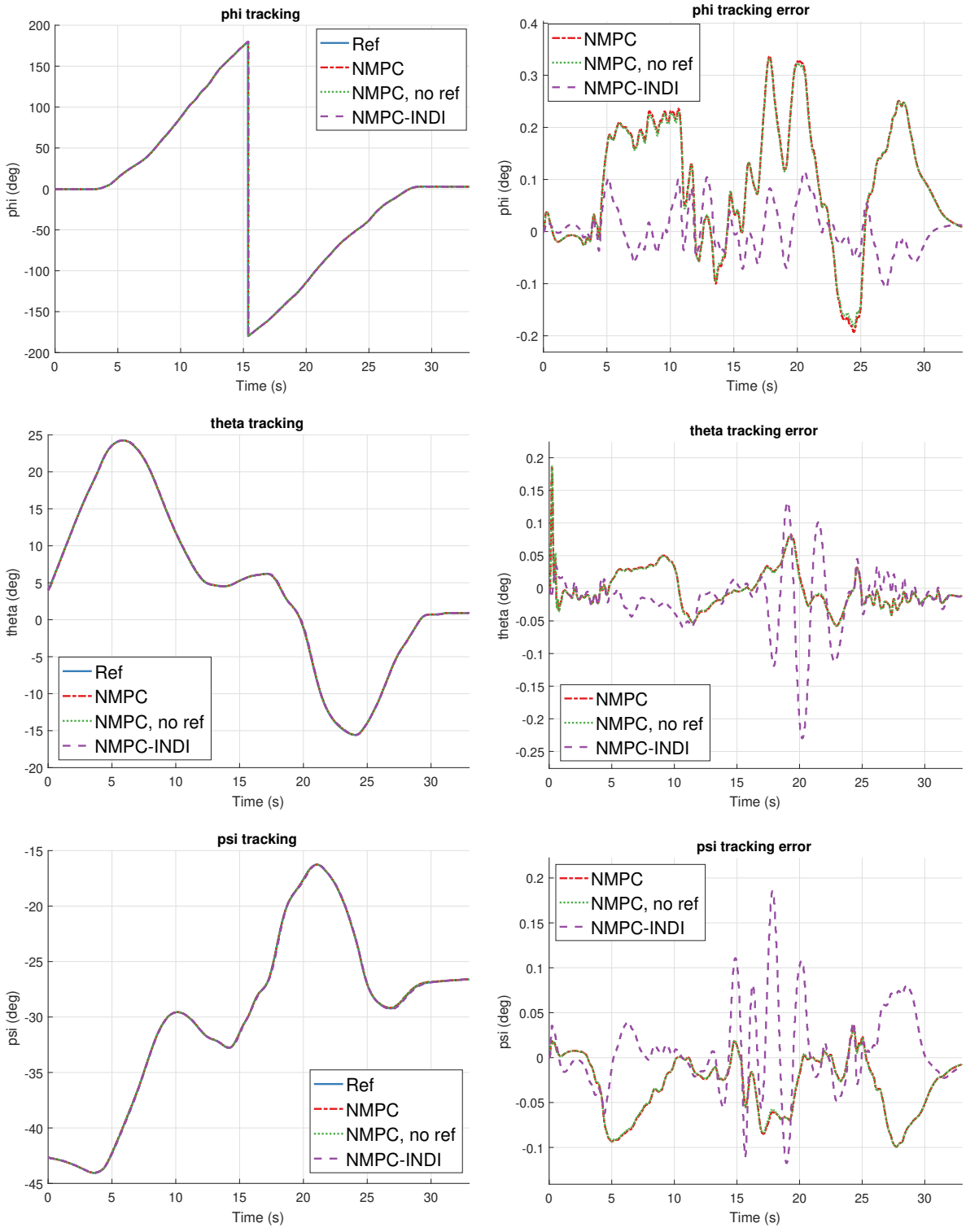


Figure 4-7: Attitude angle tracking, track: Barrel roll.

For position tracking also slightly more accurate results can be seen in Figure 4-8 but overall very similar values to the previous two tests.

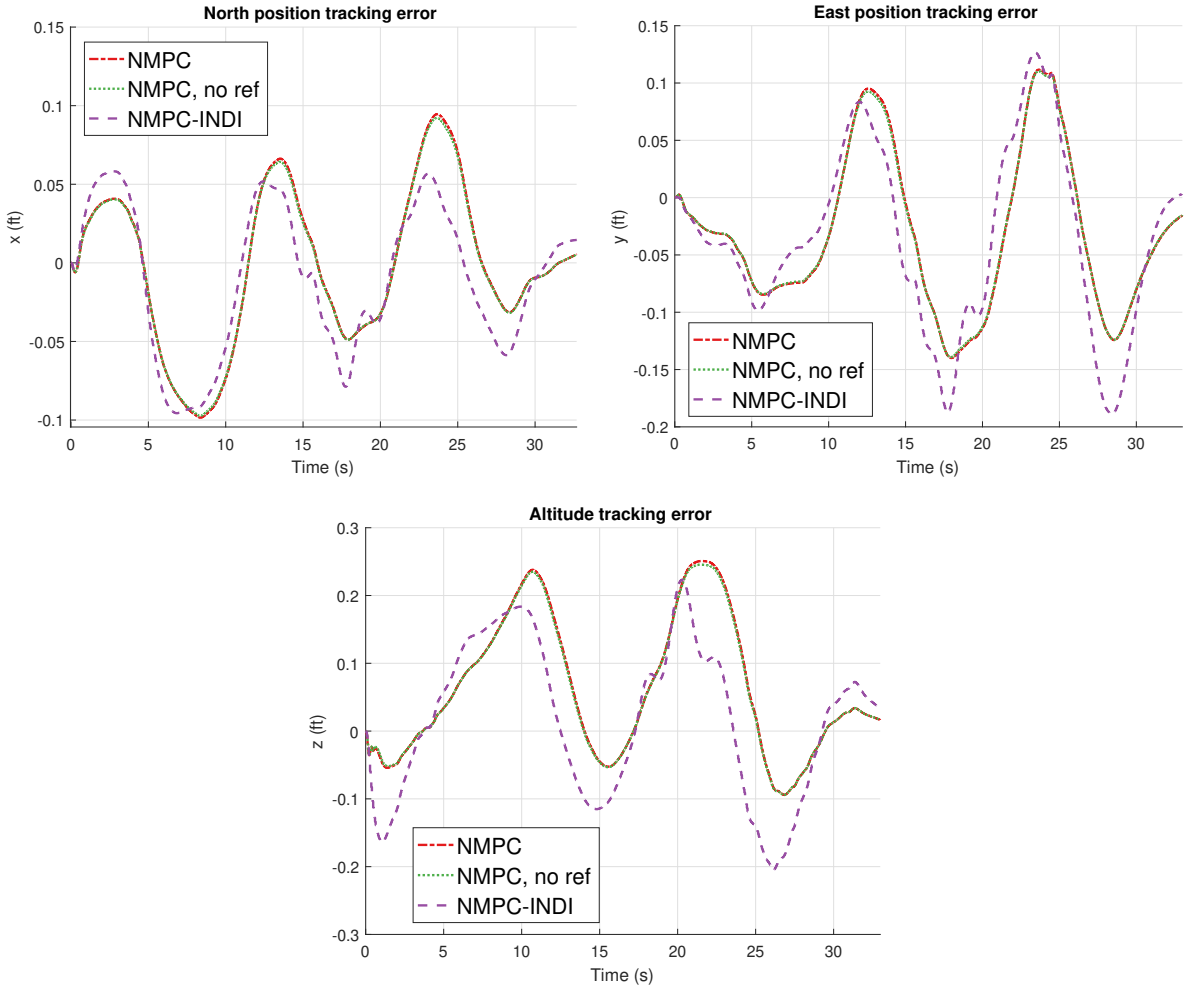


Figure 4-8: Position tracking error, track: Barrel roll.

Full set of tracking results for the barrel roll is presented in Appendix D.

4-1-4 Track: Loop

Tracking a loop is challenging since it covers the singularity when using Euler angles and includes low speed region at the top of the loop. During the initial test the NMPC lost tracking of the reference. This was likely due to the quaternion reference values oscillating around zero as seen in Figure 4-9, which can cause problems when computing the tracking error and evaluating the cost function. The oscillations occur at the low speed section of the loop where the sideslip angle controller is having hard time to keep the sideslip angle at zero and as a result introduces fast oscillations to the rudder and the yaw rate. By increasing the rotational rate weights from 10 to 100, to assist the attitude tracking, it was possible to successfully track the reference.

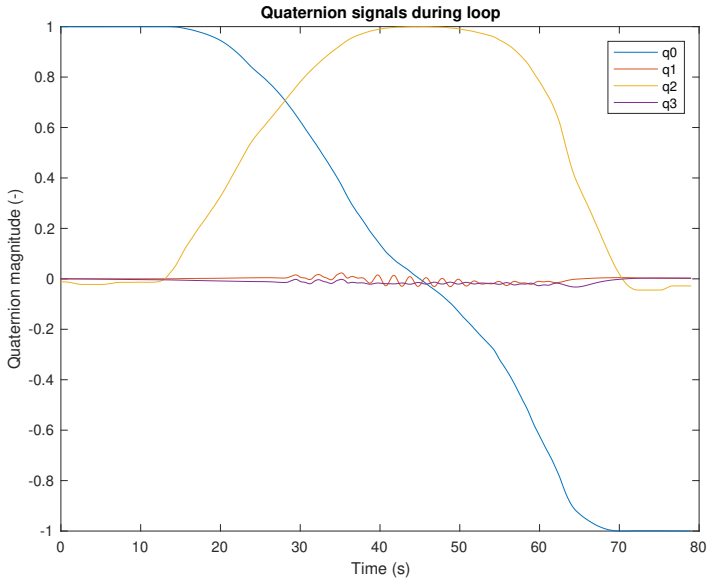


Figure 4-9: Quaternion reference signals during loop.

In the presented results, the NMPC has the increased rotational rate weights, while the NMPC without reference input signals uses original weights to show the difference changing the weights makes. With the increased weights on rotational rates the attitude tracking remained again within a few degrees from the reference compared to the initial weights, where the attitude errors are much larger as can be seen from the NMPC without input reference results. At around 55s mark, where the quaternions oscillate around 0 the fastest, a higher peak in the attitude errors for NMPC-INDI can also be seen for which the tuning weights were not changed. This was also the location where the tracking was completely lost for the NMPC with original weights for rotational rates. Previously, the attitude tracking errors were displayed in Euler angles as these are more intuitive to interpret. However, as can be seen from Figure 4-10, with this reference track there were many sign changes in the ψ angle tracking which also resulted in rather cluttered error signal graphs. For this reason the attitude tracking was presented as a quaternion orientation distance instead. This metric was introduced in Section 2-4 and here a value of 0 means that the attitude is perfectly aligned with the reference and a value of 1 represents an attitude completely opposite to the reference.

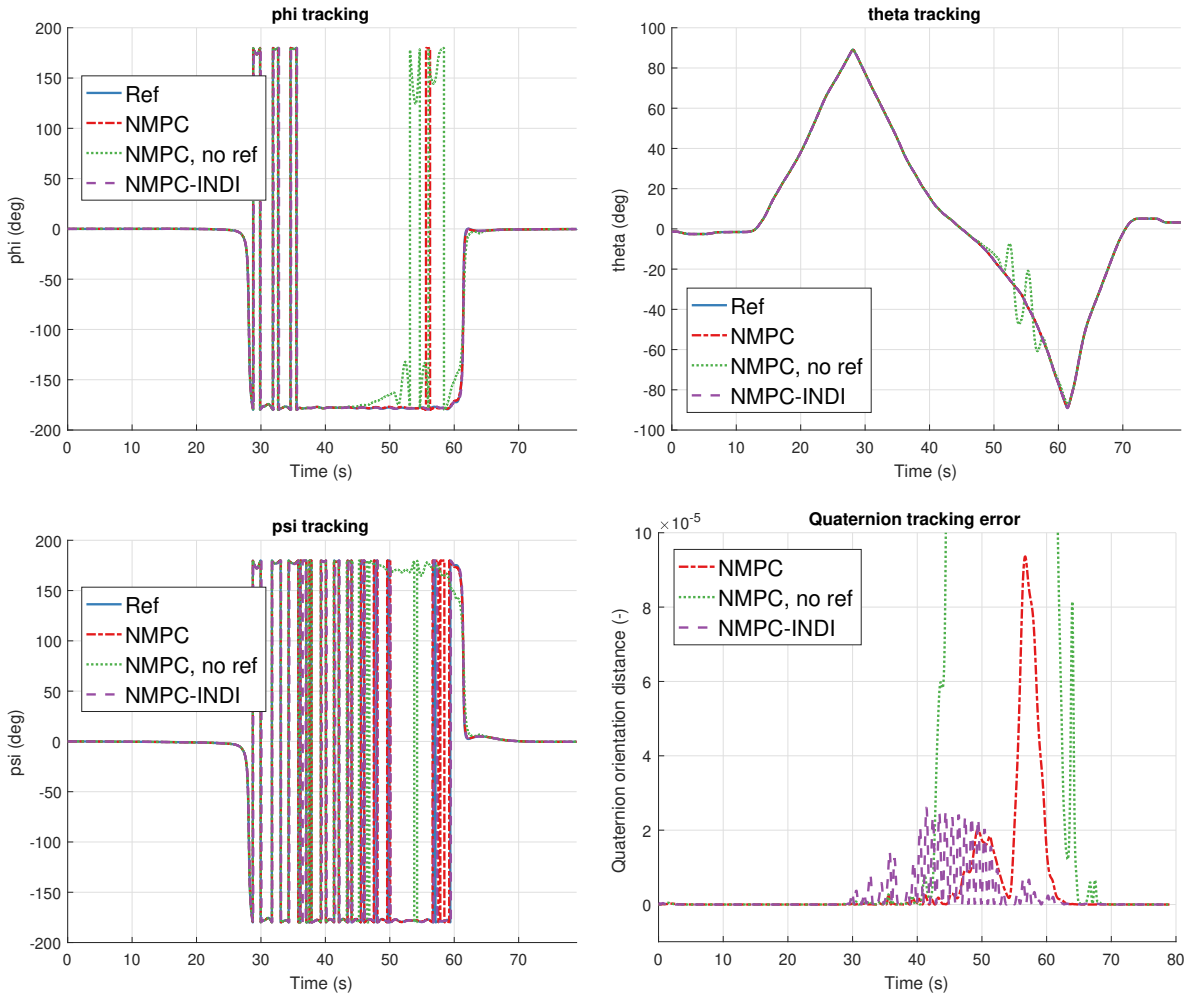


Figure 4-10: Attitude angle tracking, track: Loop.

When looking at the position tracking, again the accuracy is quite similar to previous results, except for the NMPC around the region with quaternion oscillations and for all the controllers at the beginning of the simulation. The peak in error for position tracking that occurs at the beginning arises from thrust initialization. When recording the reference trajectories the simulation is started at trim conditions for the aircraft. However, the thrust command can be at a different level, which results in a quick change in thrust immediately after the start of the simulation. In such a situation the trim condition does not hold anymore which makes the initialization more difficult for the controller compared to trim conditions.

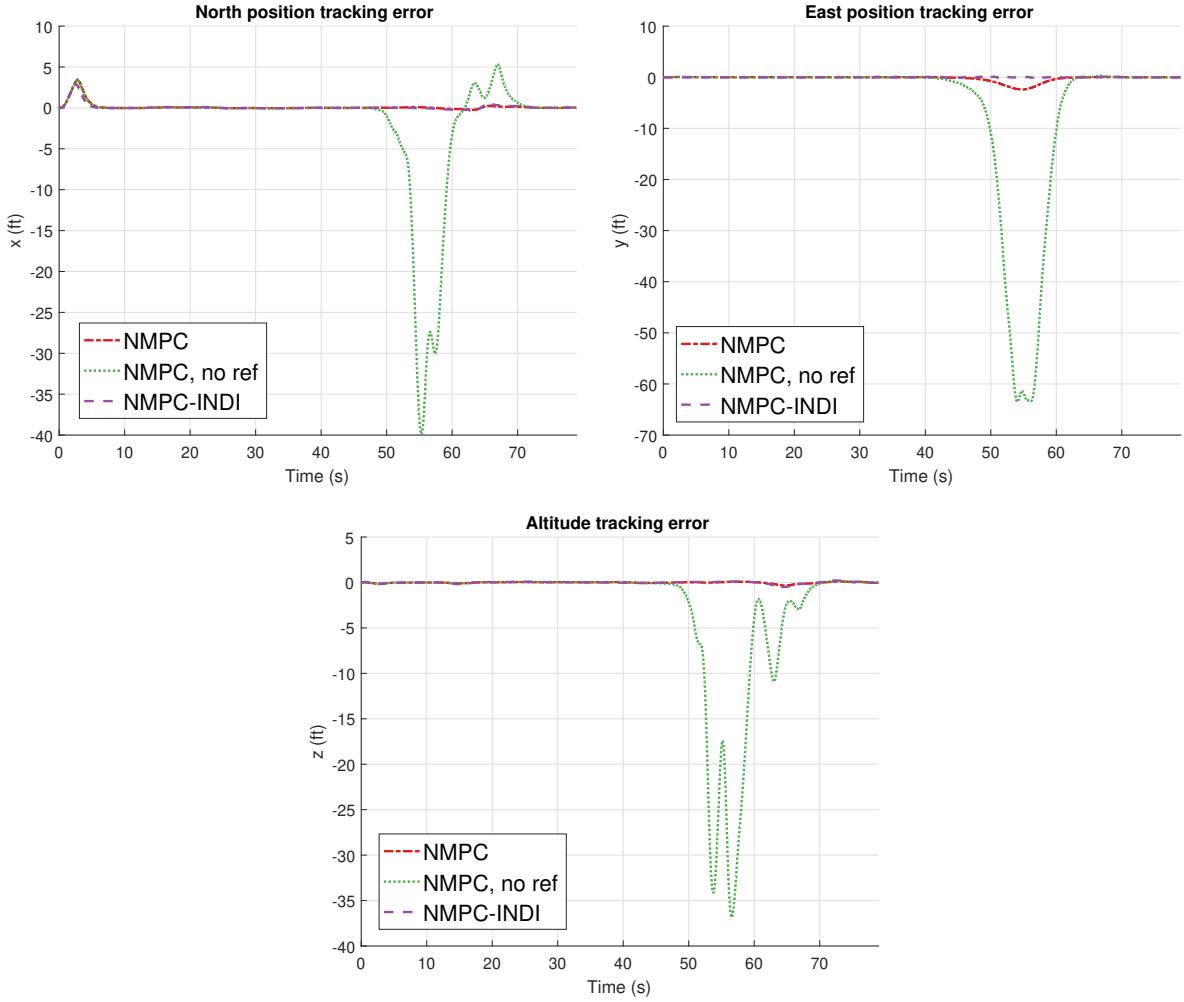


Figure 4-11: Position tracking error, track: Loop.

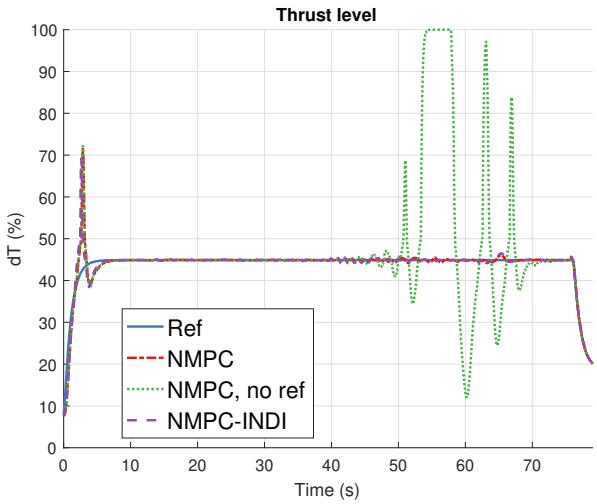


Figure 4-12: Thrust tracking, track: Loop.

By tracking the loop, it became clear that special care has to be taken when there are reference signals that are rapidly oscillating around zero as the evaluation of the cost function might lead to poor tracking performance. One method to combat that is to increase the weights and priority on other signals or reduce the weights on the signal displaying the oscillations. Again the full set of tracking results for the loop are presented in Appendix E.

4-1-5 Track: Half Cuban Eight

In a Half Cuban Eight both roll and loop motions are present. This time no oscillations around zero in the quaternion references were observed. However, fast oscillations in the yaw rate stemming again from the sideslip angle controller showed reason to use higher weights for the rotational rates as was also done with the loop reference track.

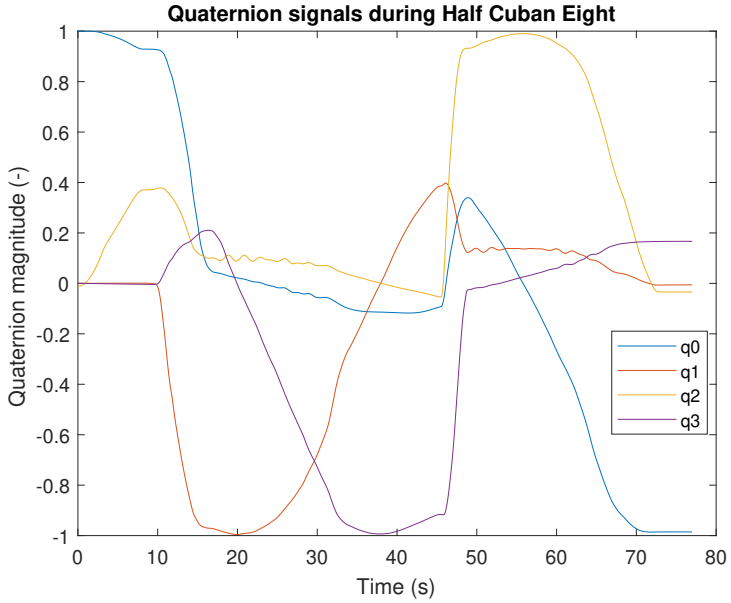


Figure 4-13: Quaternion reference signals during Half Cuban Eight.

In the following results the NMPC uses weights with increased rotational rates while NMPC without reference actuator and thrust signals has the original weights for comparison. When using the same set of weights, the responses for NMPC with and without reference actuator and thrust were nearly identical.

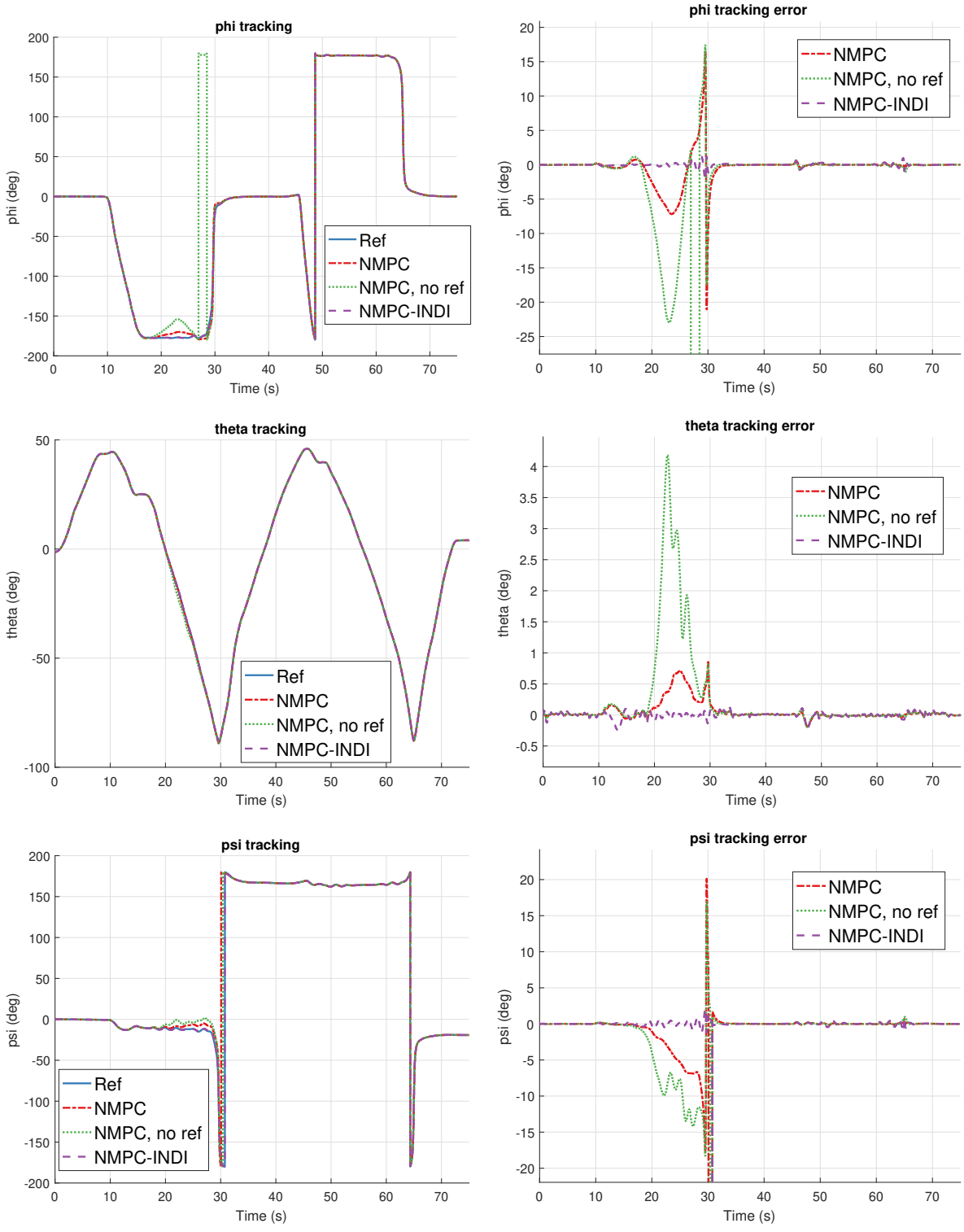


Figure 4-14: Attitude angle tracking, track: Half Cuban Eight.

From attitude tracking, it can be seen that the NMPC controller is having difficulty at the 20-30s mark where the attitude angle errors reach up to 20 deg. This is the region where the

largest oscillations in the rudder angle are encountered as seen in Figure 4-15. Difficulty in matching the yaw rates also results in higher peaks in the position tracking where the errors reach close to 18 ft. During the second loop, where there is less oscillations in the rudder, all controllers are able to accurately track the reference. From the results it can be seen that the NMPC-INDI controller performed much better and had no difficulty in tracking this maneuver.

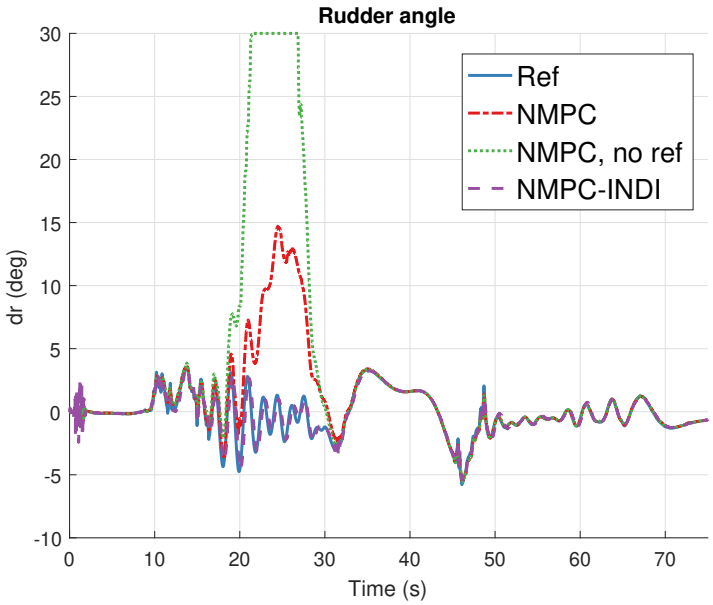


Figure 4-15: Input reference tracking, track: Half Cuban Eight.

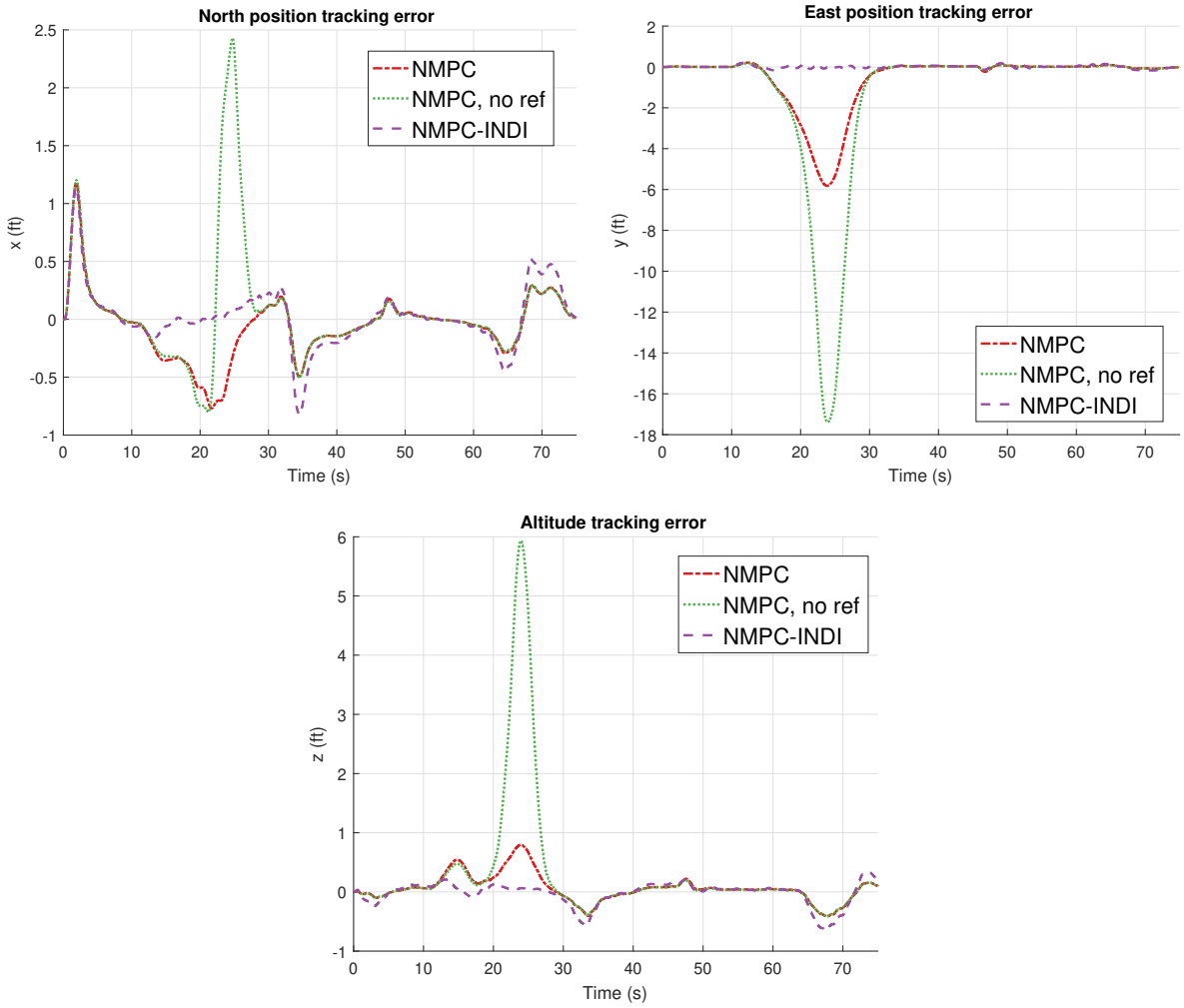


Figure 4-16: Position tracking error, track: Half Cuban Eight.

Appendix F presents the full set of results for the Half Cuban Eight maneuver.

4-1-6 Track: Recovery

The recovery reference trajectory covers a situation where an aircraft flying at a high angle of attack starts losing speed rapidly and in order to recover from stall, needs to turn and dive to regain speed. When generating reference trajectories, a 5s margin was left to the beginning of the recording before the maneuver was initiated. This was done to give some time for transients from the initialization to settle before starting the maneuver. However, for the recovery trajectory this was forgotten which resulted in oscillations in thrust level and elevator angle. Due to these oscillations tracking was lost. It was decided to relax the demands on position and velocity tracking by reducing the weights from 3 and 1 to 0.3 and 0.1 respectively but still using original weights for the rotational rates. This allowed to successfully track the reference. From attitude tracking in Figure 4-17, it can be seen that the errors again remain below 1 degree.

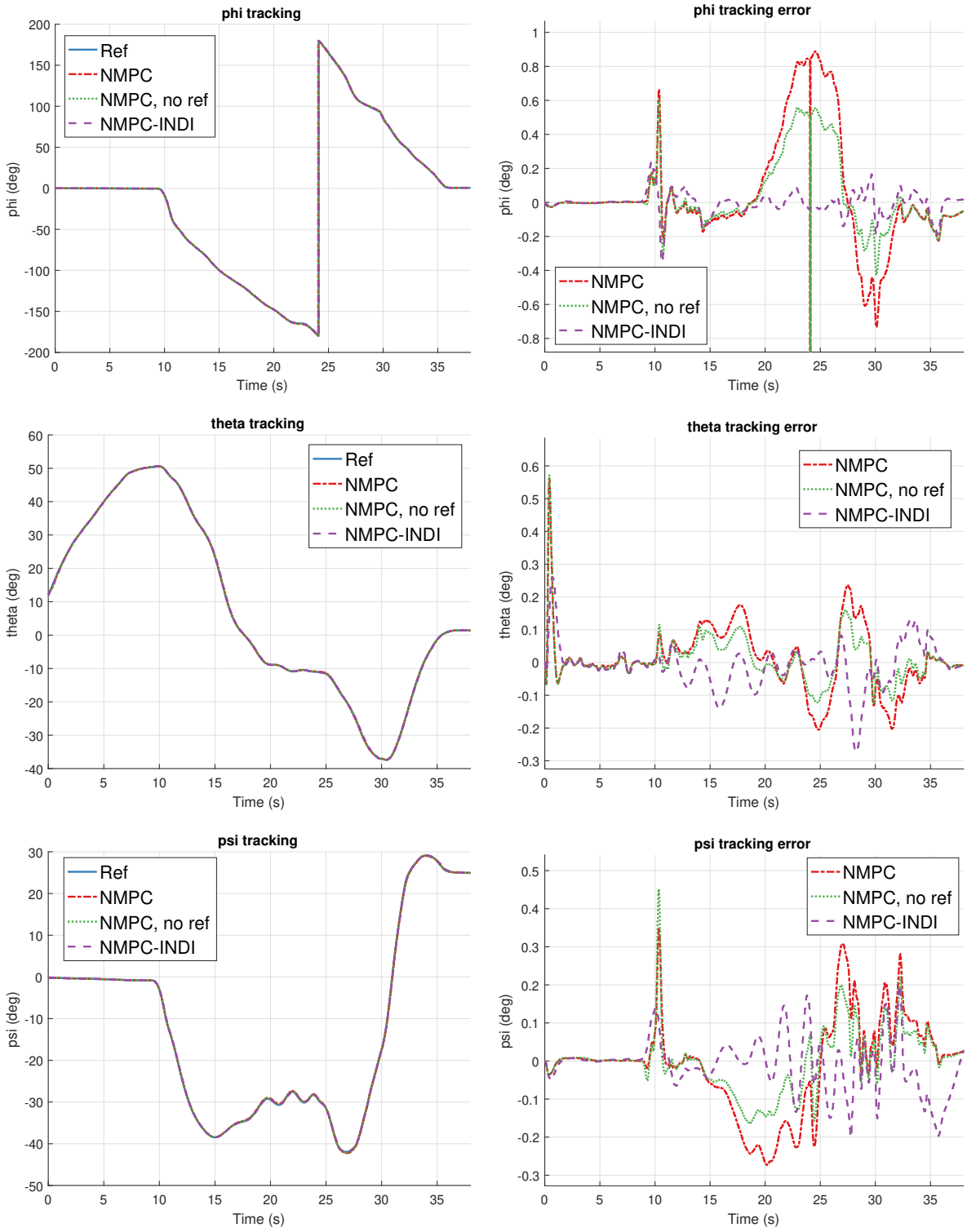


Figure 4-17: Attitude angle tracking, track: Recovery.

Even though the weights on the position and velocity were reduced, the tracking accuracy was still below 3.5 ft with the peak occurring during a roll. In the previous tests, when using

the same set of weights, the responses of NMPC with and without reference actuator and thrust signals were always nearly identical. In this test a clear difference in the results can be seen when comparing the NMPC controllers. However, there is no clear difference in the input reference tracking other than the slightly smaller oscillations in thrust tracking for the NMPC without input reference. NMPC-INDI performed the best as the longer horizon length again allowed to track the changes in the thrust level more accurately.

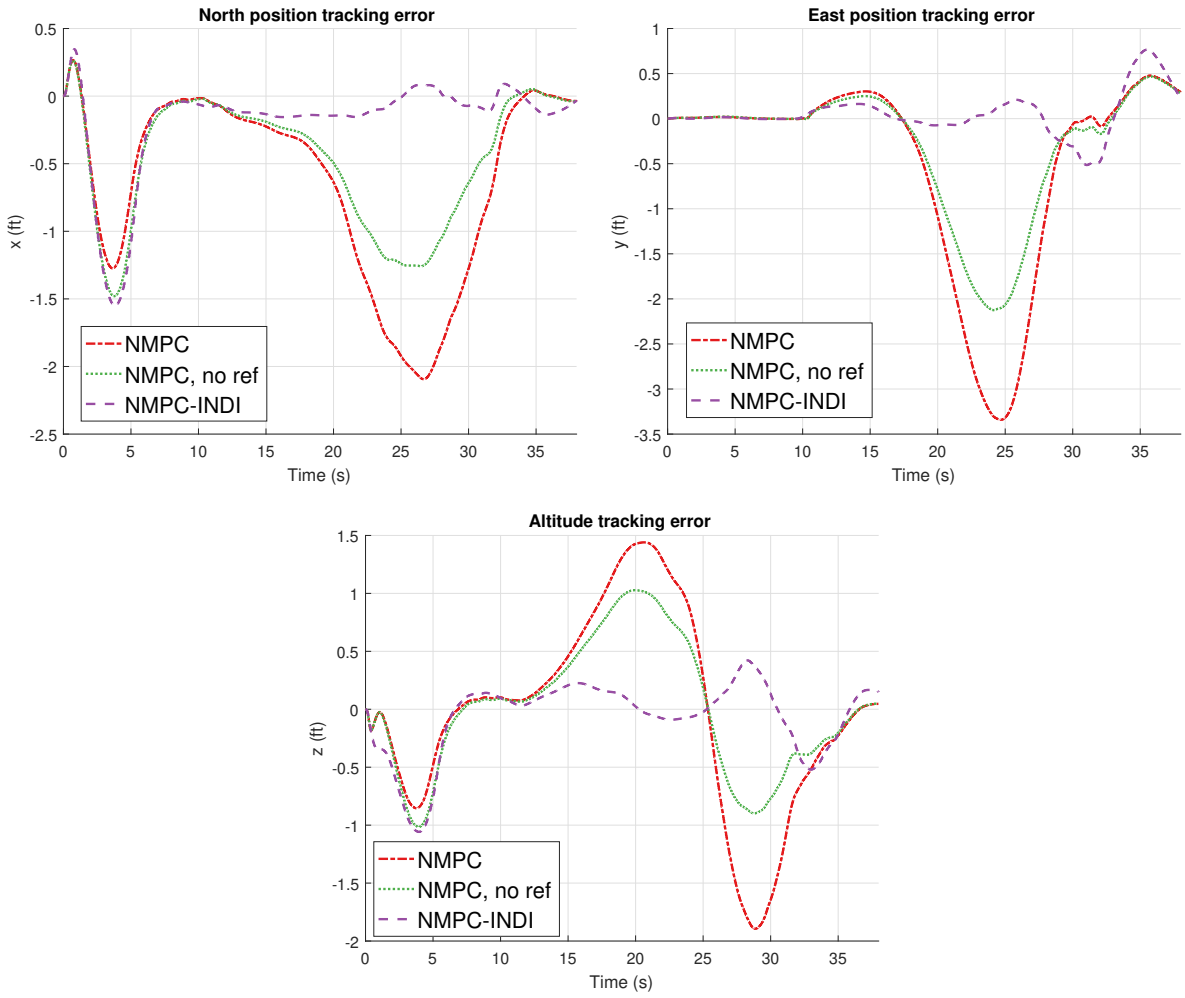


Figure 4-18: Position tracking error, track: Recovery.

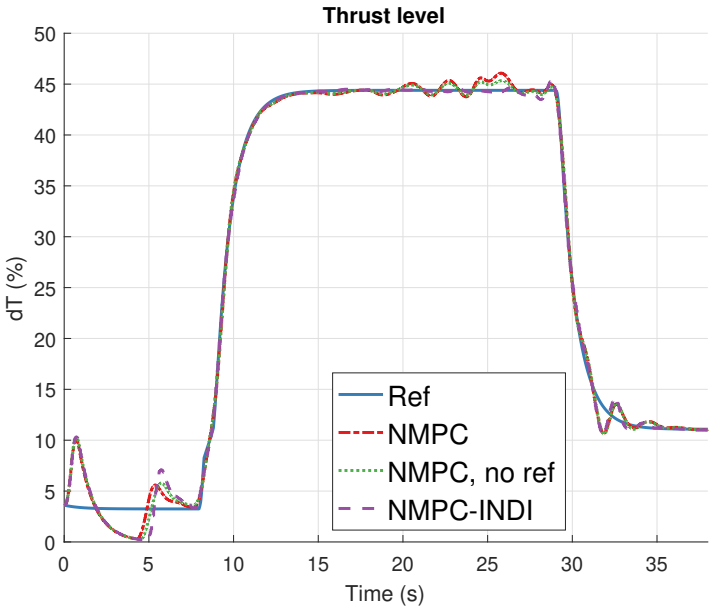


Figure 4-19: Thrust reference tracking, track: Recovery.

4-1-7 Track: Combined maneuvers

Finally, a two minute track was recorded where the pilot performed a series of rolls and turns at various rotational rates, speeds and angles of attack. This is a challenging track to follow since there are regions where the reference aileron and rudder angles get saturated as can be seen in Figure 4-20. This restricts the allowed commands the controller can set and can result in the aircraft flying in conditions outside the simulation limits, which is what exactly happened during the first tests. Then it was decided to reduce the position and velocity weights from 3 and 1 to 0.3 and 0.1 respectively for all controllers which resulted in a successful completion of the reference tracking as the position tracking requirements became less strict.

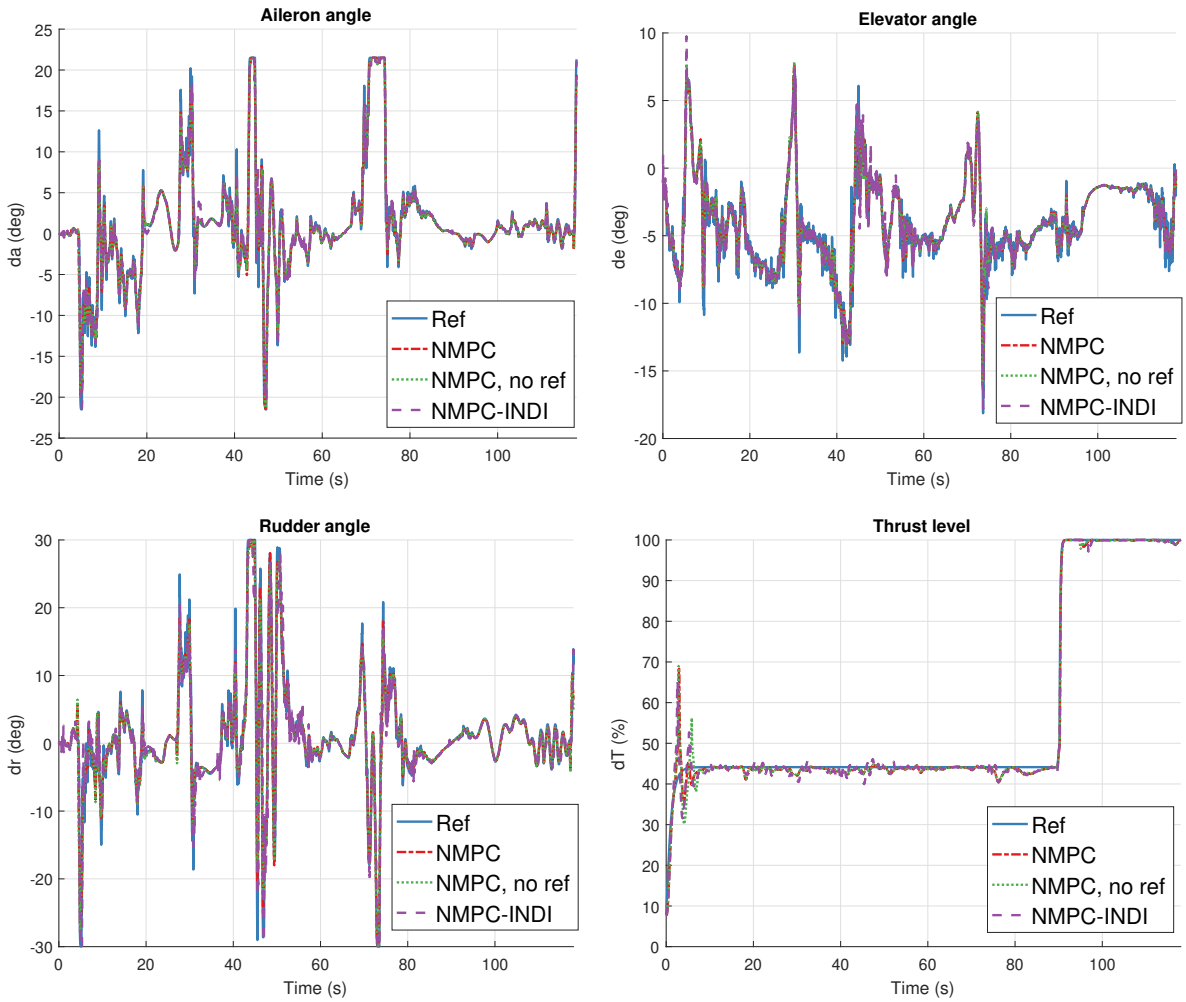


Figure 4-20: Input reference tracking, track: Combined maneuvers.

Even though the aileron and rudder angles reach the saturation limits, the attitude errors remain below 2 degrees from the reference as can be seen from Figure 4-21.

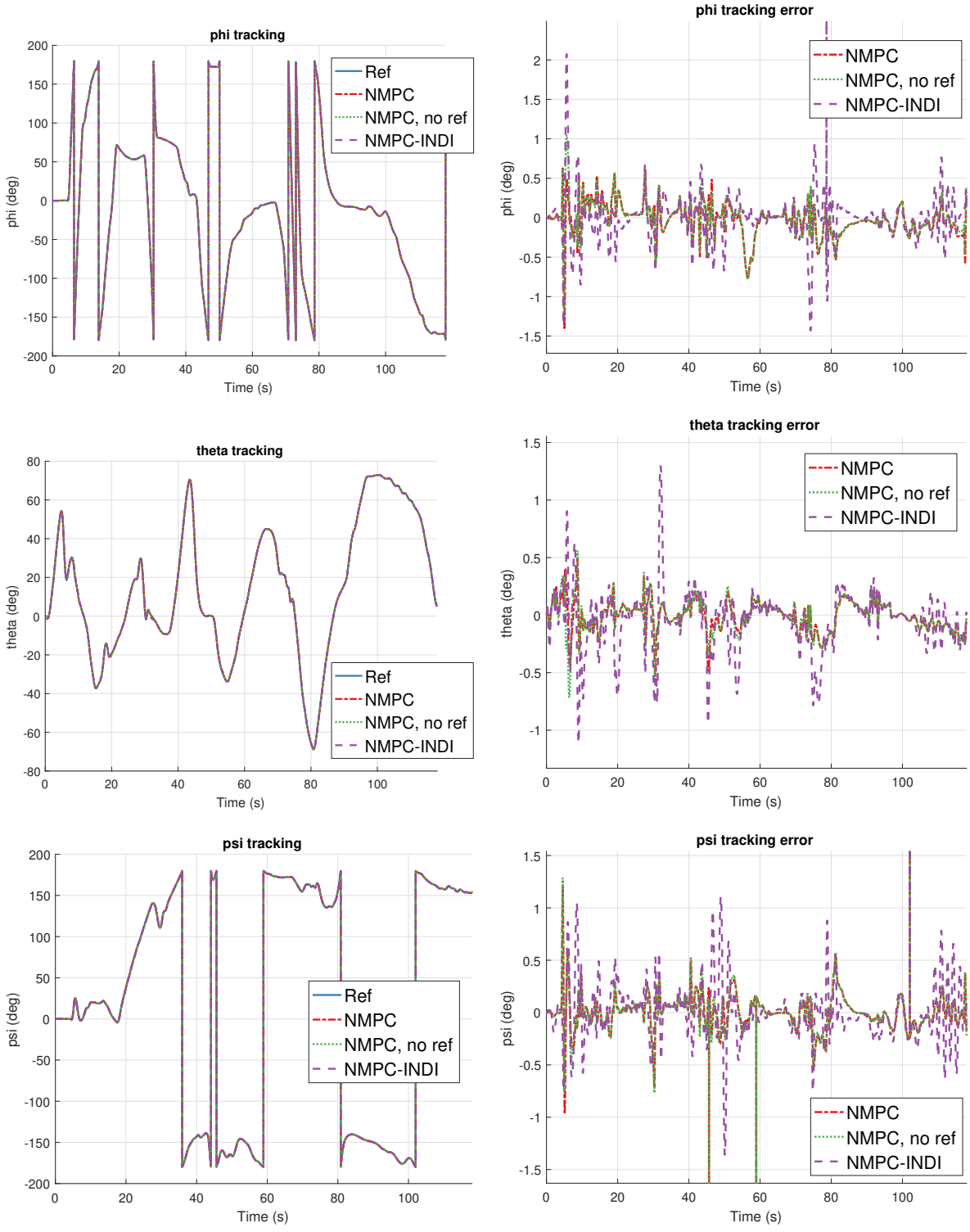


Figure 4-21: Attitude angle tracking, track: Combined maneuvers.

When looking at the position tracking error in Figure 4-22, peaks at the actuator saturation locations can be seen.

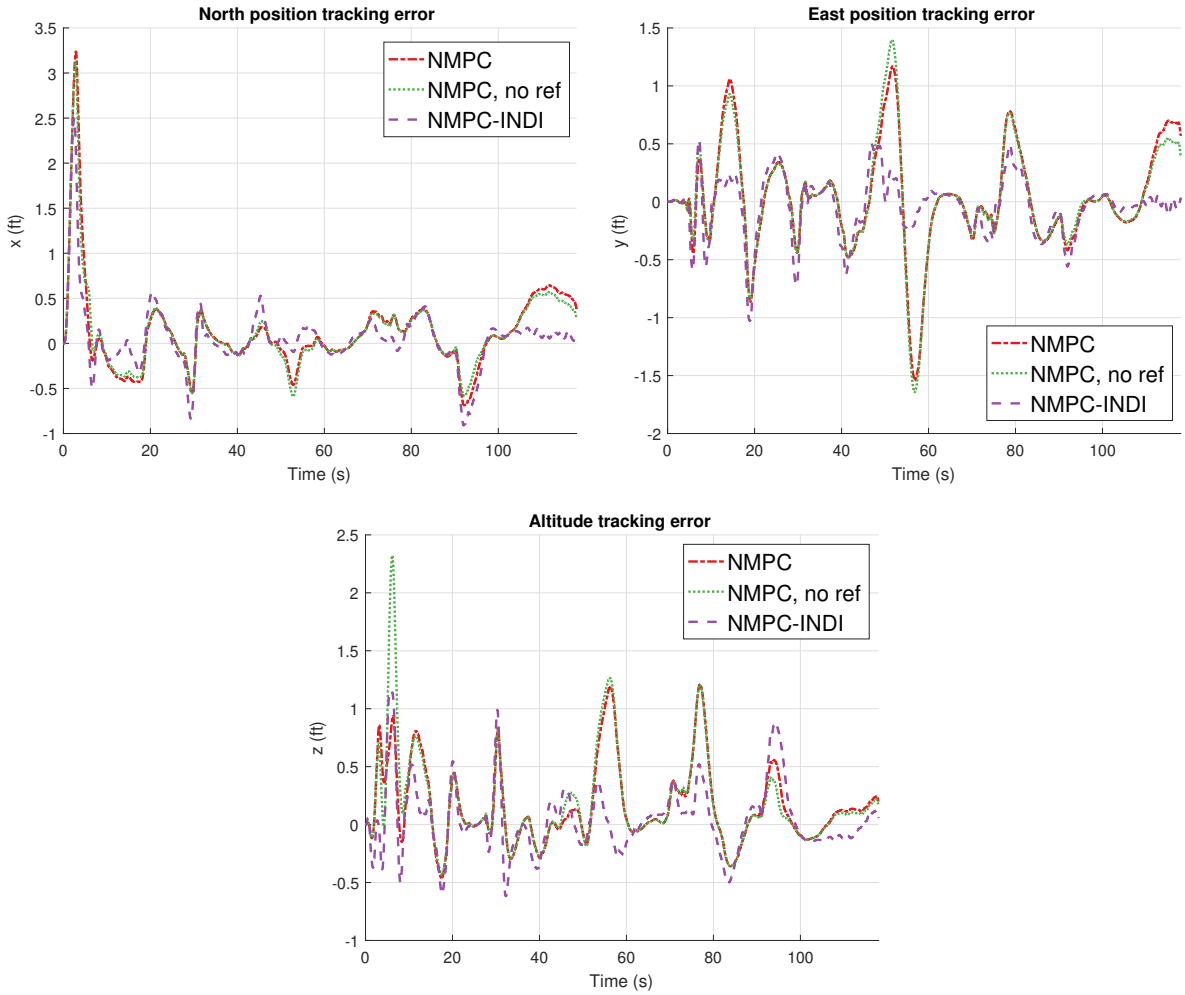


Figure 4-22: Position tracking error, track: Combined maneuvers.

A peak in position tracking error can also be seen at around 90s due to a sharp increase in the thrust level as seen in Figure 4-20. However, all the controllers are able to handle the thrust input saturation and keep the position errors bounded. Overview of all the tracking signals is presented in Appendix H.

4-1-8 Conclusion

Testing the controllers on such a variety of maneuvers gave more insight into the controllers and revealed issues not observed during the initial tuning process. For example, oscillations in the yaw rates stemming from the sideslip angle controller demonstrated the need to increase the weights for the rotational rates to be able to complete the tracking when performing loops. Furthermore, it can be beneficial to have less strict requirements for the position and velocity tracking in situations where actuators can get saturated or where the interaction between thrust and elevator can lead to increasing oscillations. In order to increase the robustness of the controllers an improvement to the state weights are proposed in (4-1) based on the adjustments done in the maneuver tracking test. This new set of weights will result

in a reduced accuracy in the position and velocity tracking but will allow the controllers to perform well in a wider range of situations.

$$\mathbf{Q} = diag\left(\begin{bmatrix} x & y & z & V_x & V_y & V_z & q_0 & q_1 & q_2 & q_3 & p & q & r & \delta_T & \delta_a & \delta_e & \delta_r \\ 0.3 & 0.3 & 0.3 & 0.1 & 0.1 & 0.1 & 10^3 & 10^3 & 10^3 & 10^3 & 100 & 100 & 100 & 1 & 1 & 1 & 1 \end{bmatrix}\right) \tag{4-1}$$

Figure 4-23 to Figure 4-29 present the Root Mean Square (RMS) errors for position and attitude and provide a summary of the tracking tests. The plots compare the tracking accuracy of NMPC, NMPC without input references and the NMPC-INDI controllers. As mentioned previously, the cost function weights were adjusted for some of the maneuvers in order to be able to successfully complete the tracking. In Figure 4-26 and Figure 4-27 the NMPC without reference input signals results represent the original weights, while NMPC has the updated weights.

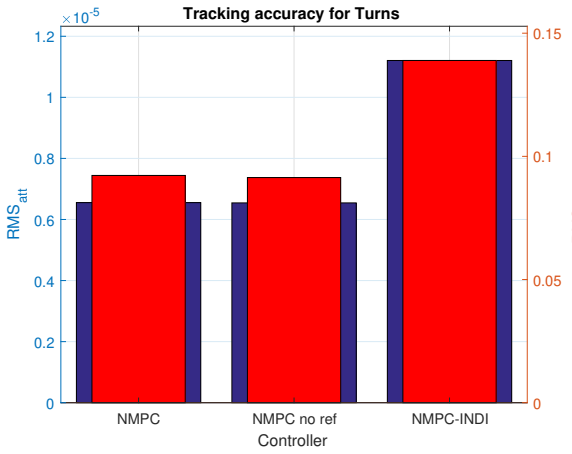


Figure 4-23: Tracking accuracy summary, track: Turns.

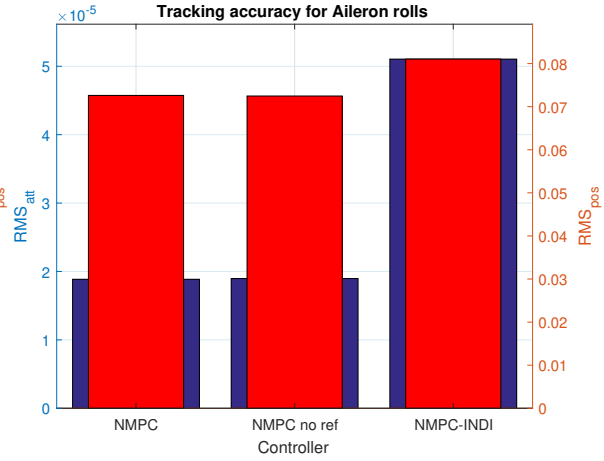


Figure 4-24: Tracking accuracy summary, track: Aileron rolls.

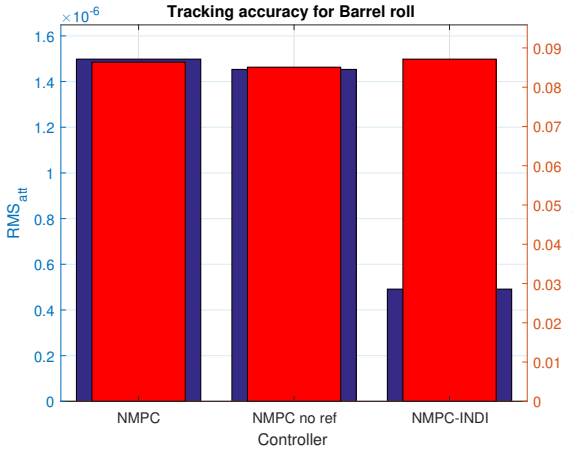


Figure 4-25: Tracking accuracy summary, track: Barrel roll.

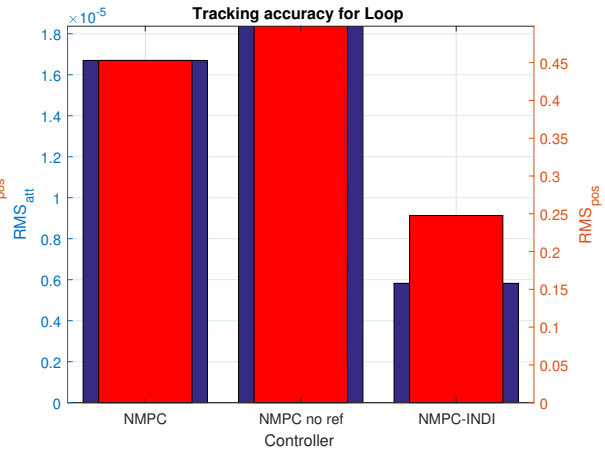


Figure 4-26: Tracking accuracy summary, track: Loop.

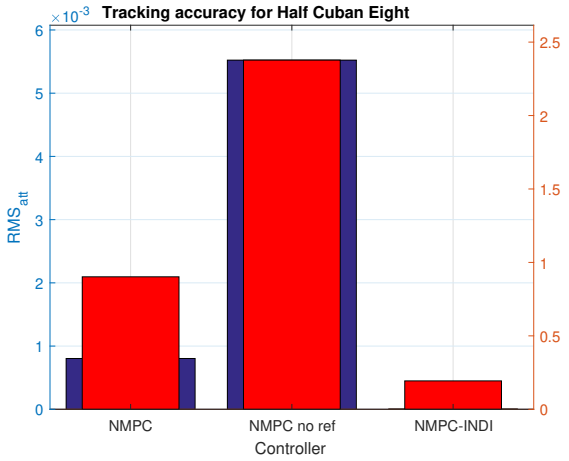


Figure 4-27: Tracking accuracy summary, track: Half Cuban Eight.

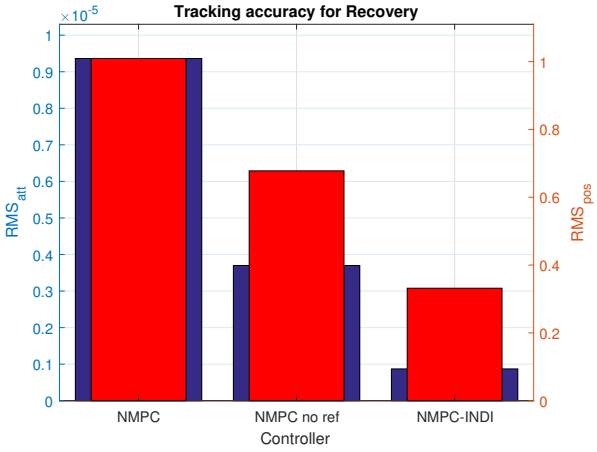


Figure 4-28: Tracking accuracy summary, track: Recovery.

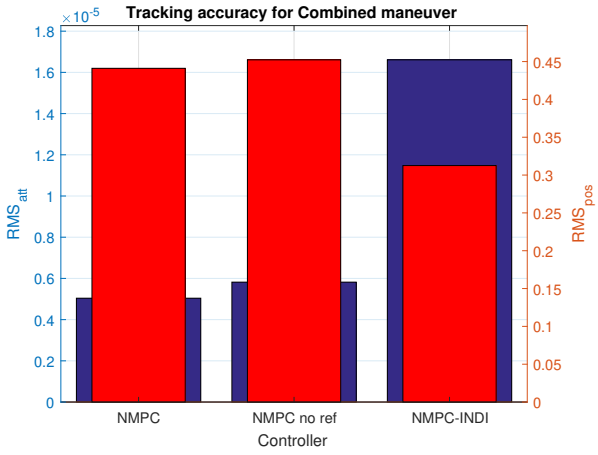


Figure 4-29: Tracking accuracy summary, track: Combined maneuver.

When comparing the tracking performance of the controllers, it can be seen that the results for NMPC with and without actuator and thrust reference are nearly identical, when using the same set of weights. This shows that the controller is able to achieve the same level of tracking accuracy even when the availability of some of the reference signals is removed. The NMPC-INDI controller is able to track the position better than NMPC, when there is more variation in the thrust level. This is due to the longer horizon of the NMPC-INDI, which makes it possible to capture the slow dynamics of the engine better.

The NMPC controller was able to track the aircraft position with a mean RMS error of 0.44 ft ($\sigma = 0.39$ ft), while the NMPC-INDI controller achieved an RMS error of 0.20 ft ($\sigma = 0.10$ ft), when averaging the results across all the maneuvers. This means that a 54.4% improvement in position tracking was achieved by adding INDI to the NMPC controller, which is also comparable to the results obtained during the tuning process. Such a high accuracy was also expected from both controllers, as there is no mismatch between the aircraft model used to generate the reference tracks, model in the simulation and model used by the controller to predict the response over the horizon.

4-2 Model mismatch tests

The controllers will be used to analyze the effects of different aircraft configurations to the performance. For this reason it is important to test the controllers in situations, where the model used to generate reference tracks is different from the model used for tracking and in the controller.

4-2-1 Weight and inertia

The first scenario, that was considered, involved a mismatch in the aircraft weight and inertia. For the tests, the fuel weight was changed ±50% which also leads to different values for the moment of inertia. The fuel weight and distribution in the tanks is difficulty to measure and model making this an important test of robustness. For the test the Barrel roll reference trajectory was used which was generated with an initial fuel weight of 3000 lbs. The tracking was performed with 1500 and 4500 lbs. As can be seen from Figure 4-30 and Figure 4-31, the controller is able to adjust the thrust to counter the effect of different weight and also change the elevator angle to fly at a slightly different pitch angle. The position tracking accuracy did decrease as a result of the weight mismatch but remained in the same order as the tracking results without weight mismatch.

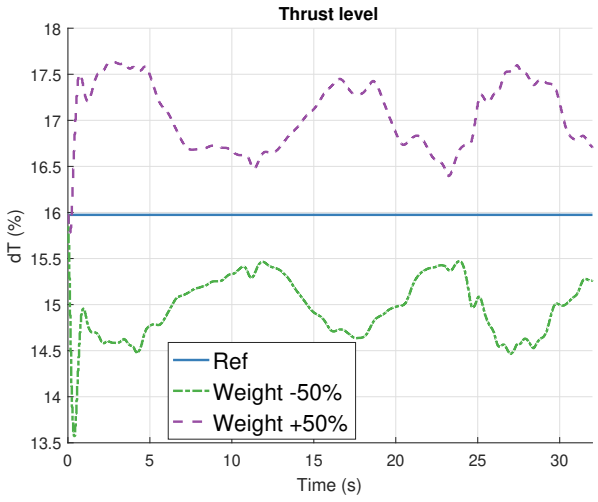


Figure 4-30: Thrust level tracking, weight mismatch.

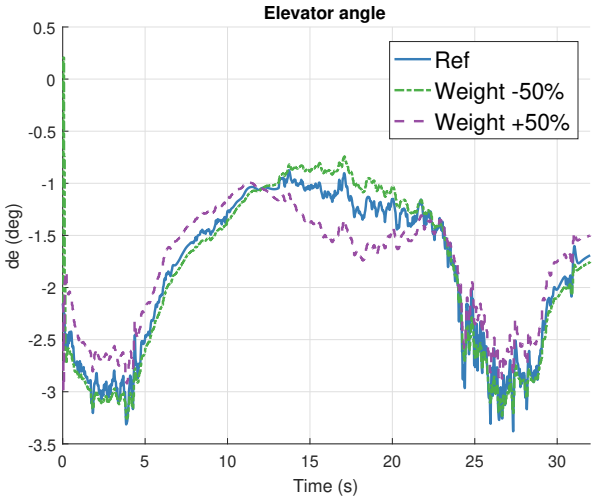


Figure 4-31: Elevator angle tracking, weight mismatch.

4-2-2 Initial position offset

Finally, an offset to the initialization position was added to observe how the controller returns to the track. Such a situation can occur as a result of a poor simulation initialization or for example, inaccuracies in the GPS signal measurements. For the tests, an offset of ±50 ft was added to the initial position of the Barrel roll reference trajectory. As can be seen in Figure 4-32, the controller is able to return to the track from a significant offset but can experience overshoot in the response. The time it takes to return to the track and oscillations

in the response are also dependent on the reference maneuver due to the controller trying to match the reference attitude at the same time. For more dynamic maneuvers, the controller can be more focused on the attitude and have less options for adjustments in position leading to a slower return to the track.

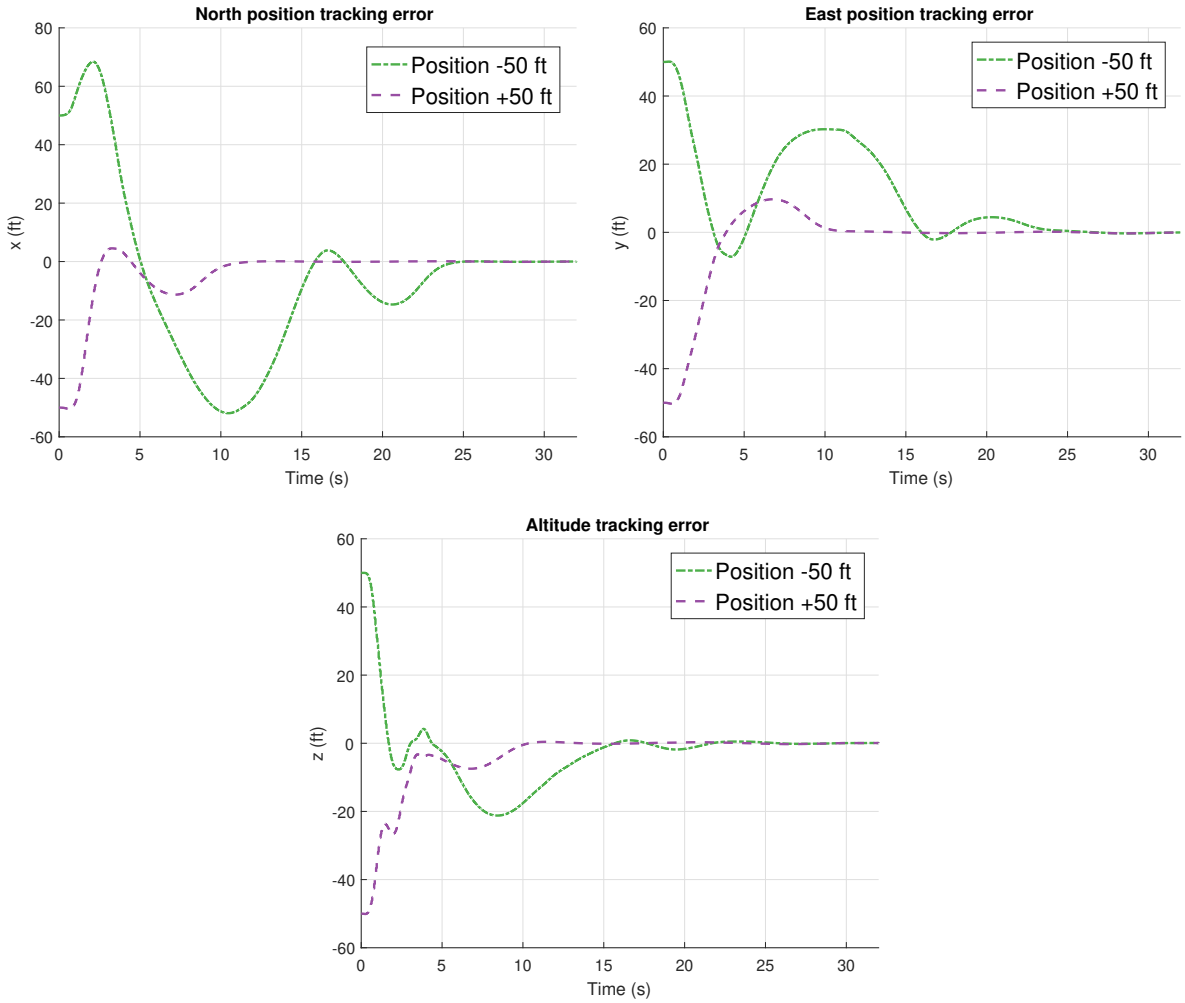


Figure 4-32: Position tracking, position offset.

4-3 Validation

Previously, all the reference tracks were generated by flying with the University of Minnesota F-16 model. However, now a reference trajectory generated using NLR F-16 simulator was tracked. For this tracking, the actuator and thrust reference signals were not used, but instead, were compared afterwards to the simulation results. This allows to validate that the controllers are able to emulate the reference flights accurately. For the tests, new set of weights was used as proposed in Section 4-1-8.

Two approximately 60s trajectories were used for the tracking. During the initial tracking tests, it was noticed, that the velocity errors had a constant offset. Furthermore, the distur-

bance observer estimates also converged to these offset values. These effects could only be explained by additional constant wind disturbance acting on the reference trajectory. This was confirmed by plotting the mismatch between reference position and velocity measurements presented in Figure 4-33.

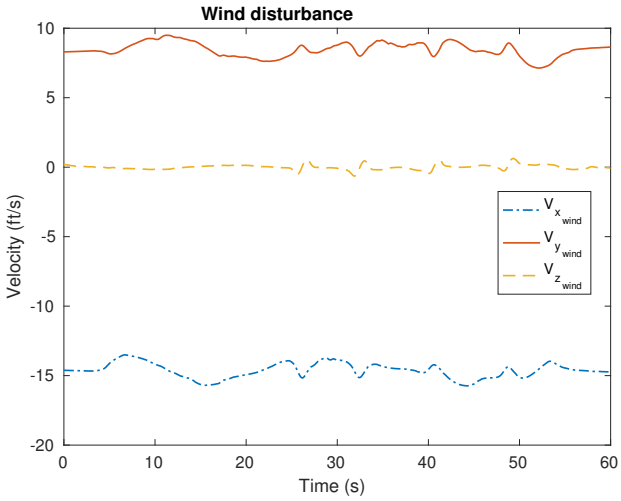


Figure 4-33: Wind disturbance on the reference track.

After including the wind disturbance to the simulation, it was possible to perform the tracking again. The first reference trajectory consists of turns with four fast rolls for which the tracking results are presented in Figure 4-34 to Figure 4-36. As can be seen, the attitude angle tracking errors remain within 8 deg for the NMPC-INDI and reach 14 deg for a short instance with the NMPC controller. At the same time, the position tracking errors remain within 40 ft for the NMPC-INDI and reach 120 ft for the NMPC controller.

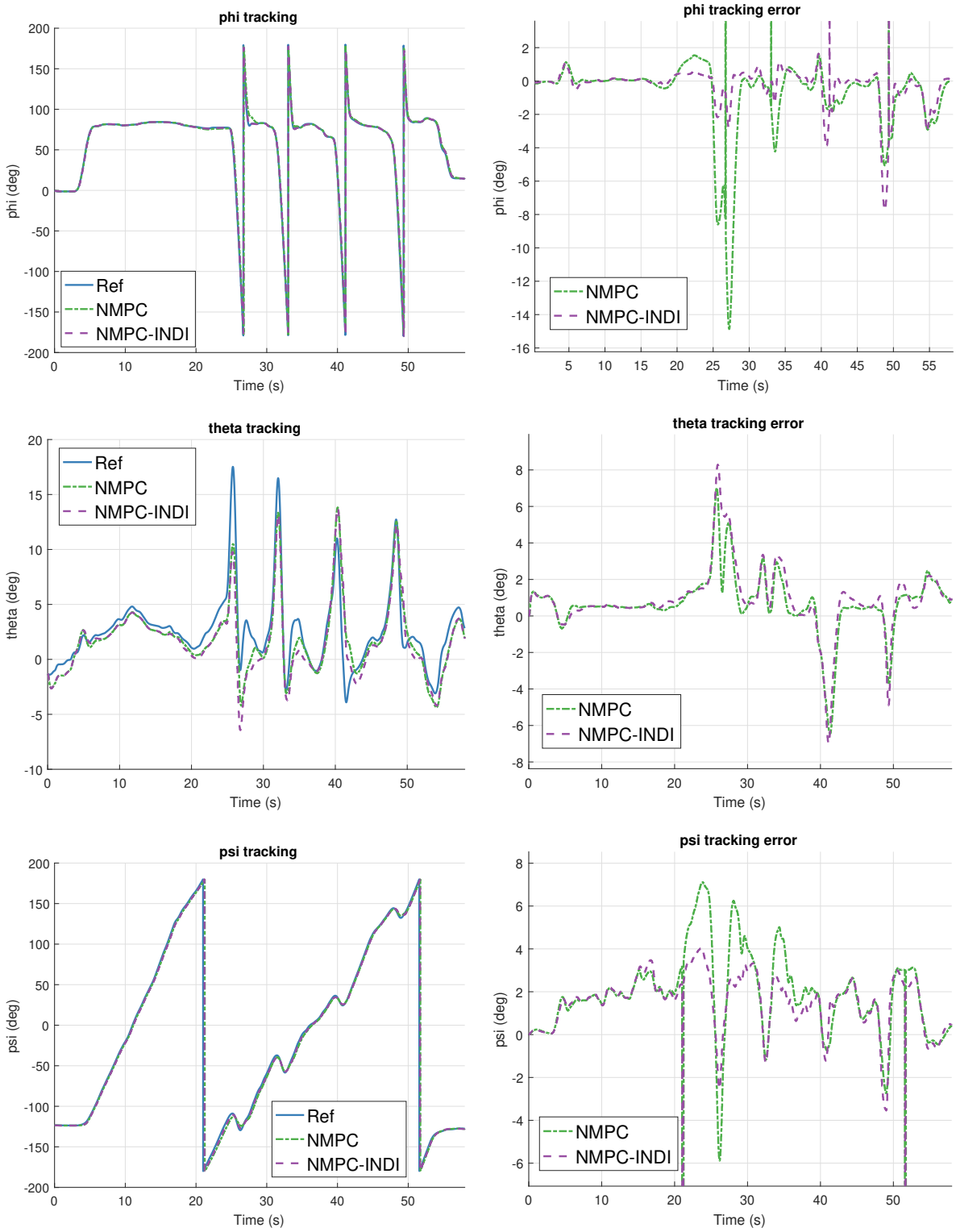


Figure 4-34: Attitude tracking, NLR track 1.

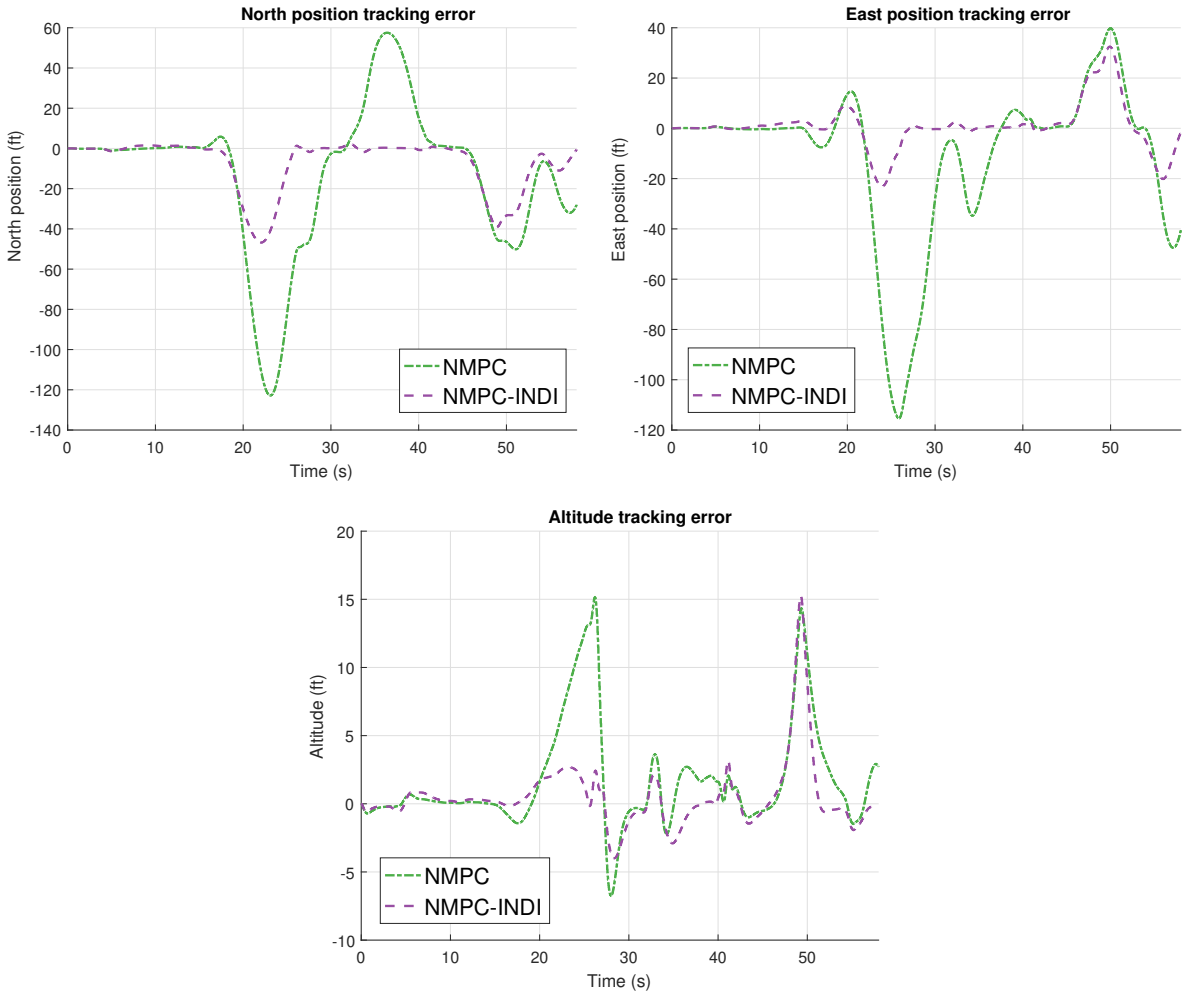


Figure 4-35: Position tracking, NLR track 1.

As mentioned previously, the actuator and thrust reference signals were not used for the tracking but instead can now be used for comparison. However, it should be noted that in the University of Minnesota model the aircraft actuators are mirrored for the ailerons and elevators, while in reality the F-16 can control all the control surfaces separately which gives more freedom over the combinations of rotational rates the aircraft can achieve. For example, higher rolling rates can be achieved when the tail is also incorporated.

When comparing the actuator signals to the reference values, it can be seen that the controllers are able to track the trajectory while matching the actuator signals quite closely. However, when comparing the thrust level, a large difference between the reference and results can be seen. Instead of keeping a constant thrust level, the controllers are actively varying the thrust. This can be explained by the fact that speed brakes were used during the generation of the reference trajectory while no speed brakes were included in the University of Minnesota model. There is no reference data about where exactly speed brakes were applied, but from the thrust response it can be estimated that the large reductions in thrust at 18s, 45s and 55s are likely due to speed brakes. This then also explains the large peaks in position tracking. As already seen in previous tests, the NMPC-INDI can handle variations in thrust much better

and is able to respond to the speed brakes better and with smaller errors in position tracking.

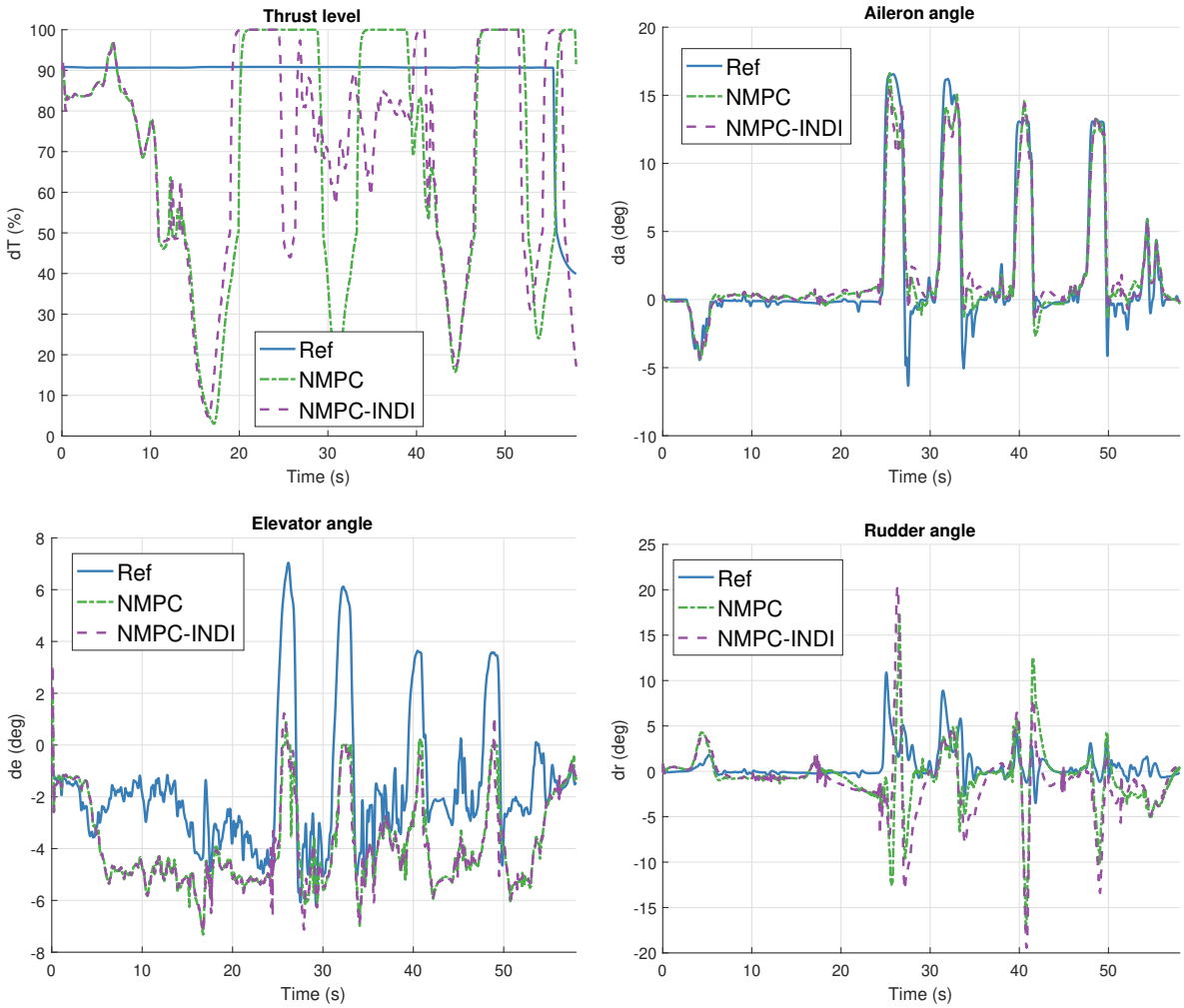


Figure 4-36: Thrust level and actuator angle comparison, NLR track 1.

The second reference trajectory recorded with the NLR simulator involves turns and a loop. The tracking results are presented in Figure 4-37 to Figure 4-39. This time both controllers are able to track the attitude very closely with the attitude angles remaining within 5 deg from the reference. Only at the locations of the singularity the error signals peak.

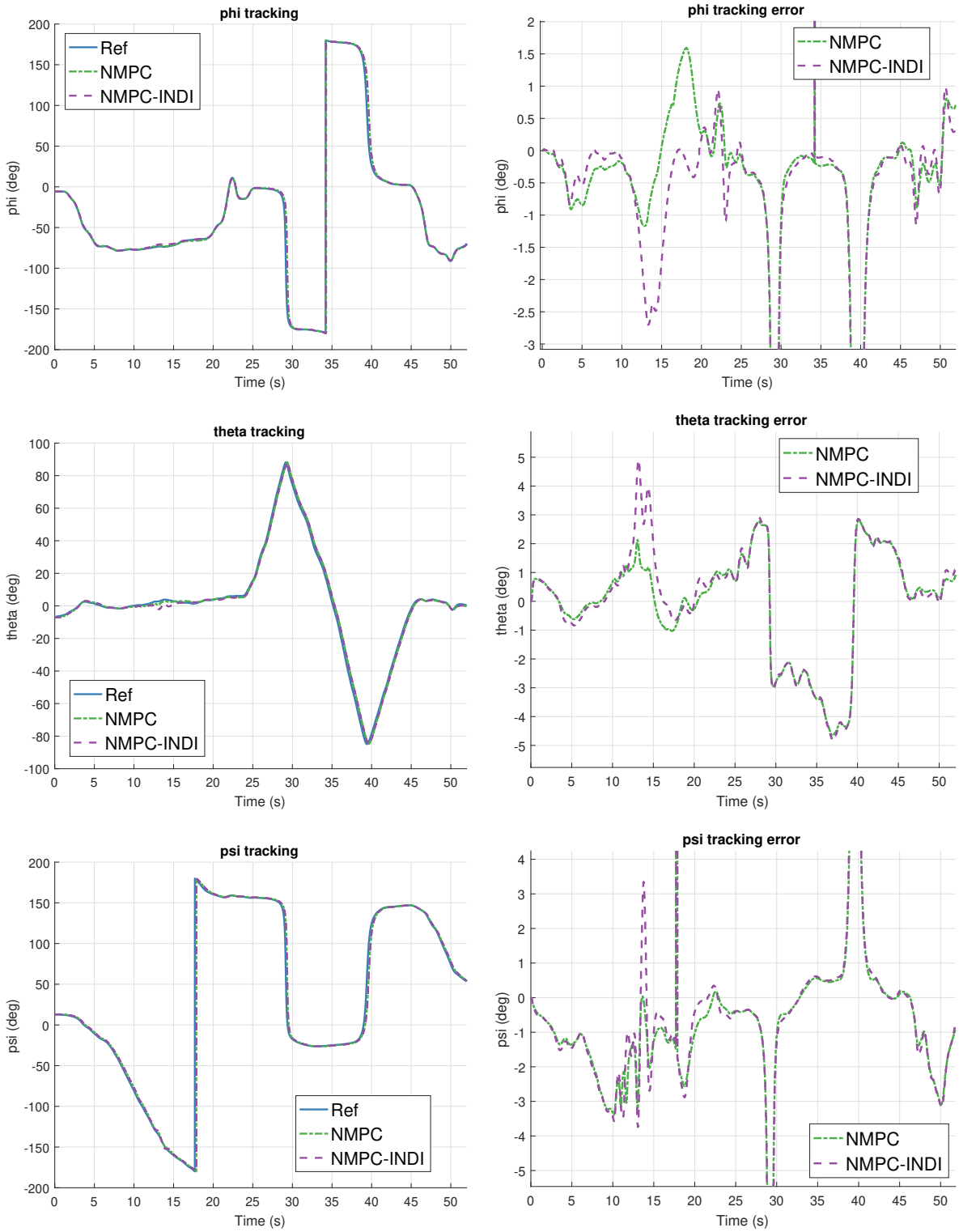


Figure 4-37: Attitude tracking, NLR track 2.

Even better performance was achieved by both controllers compared to previous trajectory

when looking at the position tracking. The highest peak for the errors is 40 ft for NMPC-INDI and 60 ft for the NMPC controller as seen in Figure 4-38. These peaks are likely caused again by the application of speed brakes as the thrust is suddenly decreased as see in Figure 4-38. The actuator signals at the same time are able to quite closely match the reference values.

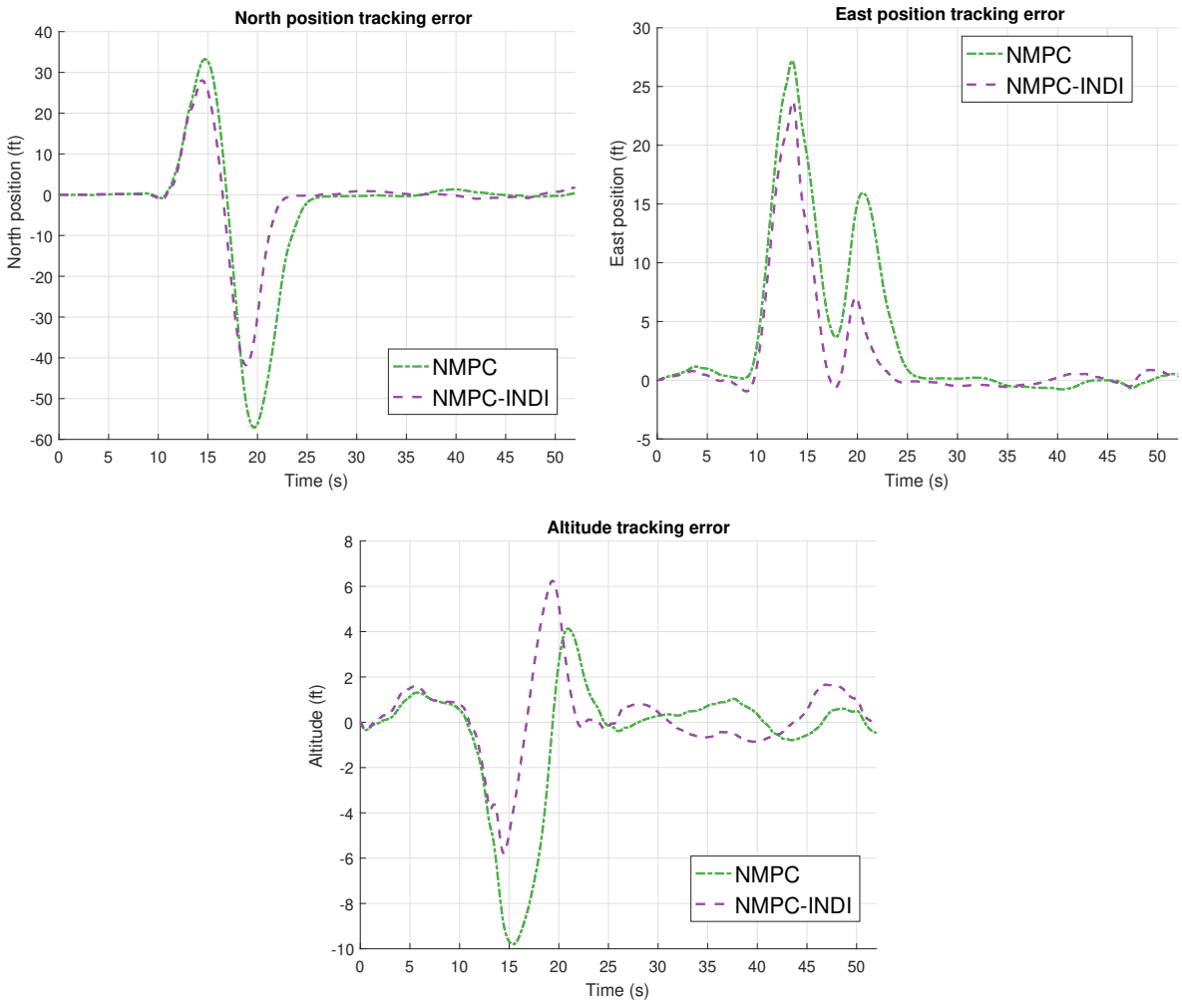


Figure 4-38: Position tracking, NLR track 2.

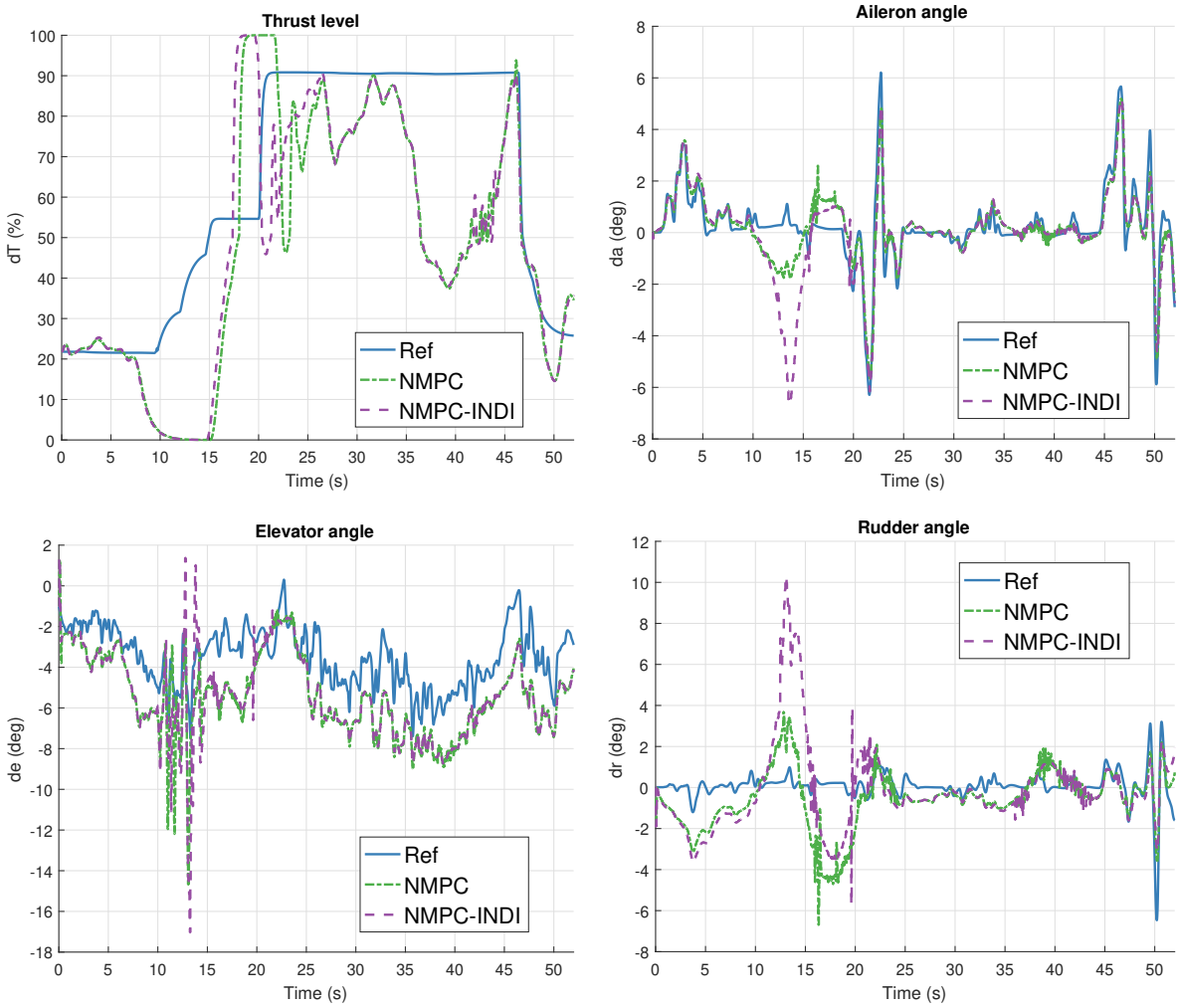


Figure 4-39: Thrust level and actuator angle comparison, NLR track 2.

Overall, both of the controllers were able to perform surprisingly well, when tracking the reference trajectories generated by the NLR simulator. Especially since there are significant differences in the models, namely the speed brakes and extra degrees of freedom in the actuator movements. The position and attitude tracking RMS errors are presented in Figure 4-40 and Figure 4-41. As can be seen, again the NMPC-INDI controller achieved better tracking accuracy than the NMPC controller even though it uses the actuator references from a different model to estimate the response over the prediction horizon. The NMPC controller was able to track the reference position with a mean RMS error of 20.01 ft ($\sigma = 13.74$ ft), while the NMPC-INDI controller achieved a mean RMS error of 8.70 ft ($\sigma = 2.38$ ft) over the two tests.

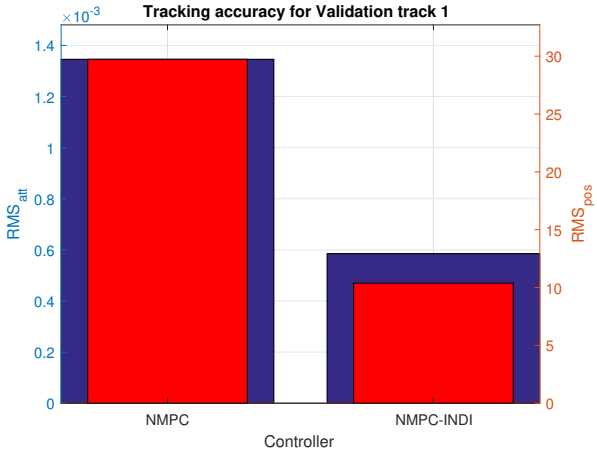


Figure 4-40: Tracking accuracy summary, NLR track 1.

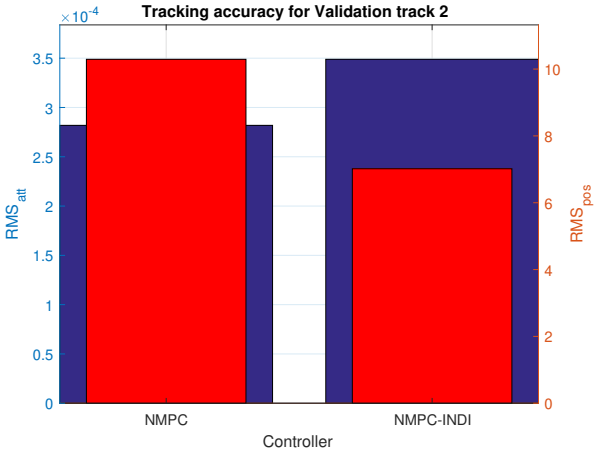


Figure 4-41: Tracking accuracy summary, NLR track 2.

4-4 Discussion

The controller parameter tuning, together with the set of reference maneuver tracking tests, gave a lot of insight to the NMPC and NMPC-INDI controllers. During the tuning tests, it was noticed that implementing move blocking did not only reduce the computation time, but also resulted in a more accurate tracking. This was surprising, since in move blocking, the number of control updates is reduced, which should result in a less accurate tracking, compared to a situation, where the control inputs can be updated after every sample. A possible explanation for this is that the extra optimization variables make finding the optimal solution more difficult or more likely to reach the maximum iteration limit and result in a worse performance.

The tuning process showed, that the NMPC-INDI controller is able to track the reference trajectories more accurately than the NMPC controller. Tracking of the various maneuvers showed that this is correct only, when there is variation in the reference thrust signals. Then the advantage of having a longer prediction horizon leads to a better tracking performance. Otherwise, the faster sampling time of NMPC leads to slightly better performance as the control inputs are updated more often, but the difference then is minimal. On average, across all the maneuvers, the NMPC-INDI controller achieved 54.4% smaller position tracking errors compared to the NMPC with the mean errors being 0.20 ft ($\sigma = 0.10$ ft) and 0.44 ft ($\sigma = 0.39$ ft) respectively. Higher accuracy together with reduced computation time makes the NMPC-INDI controller a clear favorite, as long as the reference actuator signals are available.

With the NMPC-INDI the aircraft rotational rates are controlled by the INDI controller. This allowed to use a slower sampling time for the NMPC controlling the slower states, which led to shorter computation times. In addition to rotational rates, assigning also the attitude angles to a NDI controller could lead to further improvements in computation time. However, due to time limitations, this was not investigated further in this thesis.

When using the NMPC-INDI controller it is important to ensure that the second order approximation for rotational rates and commands is not violated. From the tests with the various

maneuvers it was seen that the robustness of the controllers can be increased by choosing the cost function weights correctly. It was observed, that the priority should be given to attitude tracking over position tracking. With higher priority for the position the tracking can be better in short term but might not lead to a correct attitude, which is required to perform the upcoming maneuver.

The high tracking accuracy that was achieved for the set of various maneuvers was expected since the reference maneuvers were generated with the same model as used for tracking. However, the accuracy achieved when tracking the NLR reference exceeded expectations. Both controllers were able to handle the differences related to the speed brake and coupling of the control surfaces. Furthermore, before adding the wind disturbance, the effects of it could be clearly seen as an offset in velocity tracking errors, which allowed to easily identify the underlying mismatch source.

Chapter 5

Conclusions and Recommendations

Upgrading aircraft subsystems to more modern versions and adding additional instruments can provide extra functionality and extend the aircraft lifespan. However, this often introduces increased power demands and heat generation, which can expose the aircraft to conditions it was not originally designed for and can lead to increased wear and reduction in performance. To be able to analyze the effects of different aircraft configurations on the flight performance it is necessary to execute flight maneuvers while changing the aircraft parameters. Having a pilot perform these maneuvers introduces undesirable variability to the results. So instead, it is necessary to develop a controller that is able to use previously recorded reference trajectories and closely reproduce these in a simulated environment. The reference trajectories of interest were flown with the F-16 Fighting Falcon (F-16) fighter jet and consist of highly dynamic aerobatic maneuvers. The test flights were conducted with a specially instrumented aircraft, which allows to measure and record the aerodynamic angles together with the control surface angles in addition to the traditional position and attitude signals. The level of difficulty of the reference trajectories and the availability of the extra reference signals makes this a very unique problem.

In this thesis, a publicly available F-16 model was used and extended with a feedback linearization controller to generate an analogous set of aerobatic reference flight trajectories. To track the reference trajectories two controllers were developed. A Nonlinear Model Predictive Control (NMPC) controller and a NMPC controller combined with a feedback linearization controller in the form of Incremental Nonlinear Dynamic Inversion (INDI). Adding feedback linearization reduced computation time and improved tracking of engine dynamics but made the controller dependent on reference actuator angles while a nonlinear model predictive controller alone is slower, but is able to track accurately with and without thrust and actuator angle reference signals. Tracking accuracy was tested on a set of well known aerobatic maneuvers such as Barrel rolls, loops and Half Cuban Eights. The NMPC controller was able to track the aircraft position in these maneuvers with a mean error of 0.44 ft ($\sigma = 0.39$ ft) while the NMPC-INDI controller achieved a mean position tracking error of 0.20 ft ($\sigma = 0.10$ ft). Further tests included mismatches in initial fuel weight and position. Here both controllers showed the ability to adjust the thrust level to counter the change in mass and the

ability to return to the desired track from an offset. Finally, both controllers were validated by tracking a reference generated by NLR F-16 simulator. These tests included significant differences between the aircraft model used to generate the reference trajectory and the one used for tracking. One major difference was the lack of speed brakes for the aircraft tracking the reference. However, both controllers were able to adjust the thrust level to counter the effects of the speed brakes. The NMPC was able to track the aircraft position with a mean position error of 20.01 ft ($\sigma = 13.74$ ft) while the NMPC-INDI achieved a mean position error of 8.70 ft ($\sigma = 2.38$ ft). These tests showed that both controllers are able to handle significant differences in the aircraft models and still keep the mean position tracking errors within one wingspan length (30 ft) which was the size of the reference tunnel for the pilot to track.

5-1 Recommendations

The NMPC and NMPC-INDI controllers are able to achieve good tracking performance but there are numerous aspects that could be improved or investigated further. The recommendations for future work are presented in the following list:

- NMPC controller was chosen as it allows to use nonlinear models that can accurately capture the aircraft dynamics during loops, rolls etc. to predict the system response. As an alternative approach, a Linear Parameter Varying (LPV) model could be developed and used together with a linear MPC controller for comparison of the tracking performance and computation time.
- Combining NMPC with INDI led to a reduction in computation time. To potentially reduce the computation time further, controlling also the attitude angles in addition to the rotational rates with a NDI controller should be investigated.
- Using a dedicated NMPC solver should be considered as it can lead to improvements in computation time.
- In order to improve tracking accuracy, the F-16 model should be extended with speed brakes. Furthermore, allowing control surfaces on either side of the aircraft move separately would make the model closer to the actual F-16. However, control allocation should be then introduced in order not to increase the number of inputs and computation time for the NMPC.

Appendix A

Equations of Motion used by F-16 model

In this appendix an overview of the equations of motion used by the F-16 model is given.

The navigation equations to determine the position of the aircraft are presented in (A-1), where the transformation matrix is given by (A-2).

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = \mathbf{T}_{Eb} \begin{bmatrix} u \\ v \\ w \end{bmatrix} \tag{A-1}$$

$$\mathbf{T}_{Eb} = \begin{bmatrix} \cos\theta\cos\psi & \sin\phi\sin\theta\cos\psi - \cos\phi\sin\psi & \cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi \\ \cos\theta\sin\psi & \sin\phi\sin\theta\sin\psi + \cos\phi\cos\psi & \cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi \\ -\sin\theta & \sin\phi\cos\theta & \cos\phi\cos\theta \end{bmatrix} \tag{A-2}$$

Next the body frame accelerations are found using (A-3), where the body forces are found using (A-4).

$$\begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} = \frac{1}{m} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + \begin{bmatrix} 0 & -w & v \\ w & 0 & -u \\ -v & u & 0 \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} + \mathbf{T}_{bE} \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \tag{A-3}$$

$$\begin{aligned} X &= \frac{1}{2}\rho V_t^2 S C_X \\ Y &= \frac{1}{2}\rho V_t^2 S C_Y \\ Z &= \frac{1}{2}\rho V_t^2 S C_Z \end{aligned} \tag{A-4}$$

Rotational dynamics equations are presented in (A-5) where the moments acting on the aircraft are found using (A-6).

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} I_x & 0 & -I_{xz} \\ 0 & I_y & 0 \\ -I_{zx} & 0 & I_z \end{bmatrix}^{-1} \begin{bmatrix} L \\ M \\ N \end{bmatrix} - \begin{bmatrix} I_x & 0 & -I_{xz} \\ 0 & I_y & 0 \\ -I_{zx} & 0 & I_z \end{bmatrix}^{-1} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times \begin{bmatrix} I_x & 0 & -I_{xz} \\ 0 & I_y & 0 \\ -I_{zx} & 0 & I_z \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{A-5}$$

$$\begin{aligned} L &= \frac{1}{2}\rho V_t^2 S b C_l \\ M &= \frac{1}{2}\rho V_t^2 S \bar{c} C_m \\ N &= \frac{1}{2}\rho V_t^2 S b C_n \end{aligned} \tag{A-6}$$

The kinematic equations are presented in (A-7).

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \frac{\sin\phi}{\cos\theta} & \frac{\cos\phi}{\cos\theta} \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{A-7}$$

Finally the derivatives for velocity, angle of attack and sideslip are found using (A-8).

$$\begin{aligned} \dot{V}_t &= \frac{u\dot{u} + v\dot{v} + w\dot{w}}{V_t} \\ \dot{\alpha} &= \frac{u\dot{w} - w\dot{u}}{u^2 + w^2} \\ \dot{\beta} &= \frac{\dot{v}V_t - v\dot{V}_t}{V_t^2 \cos\beta} \end{aligned} \tag{A-8}$$

Appendix B

Tracking results of track: Turns

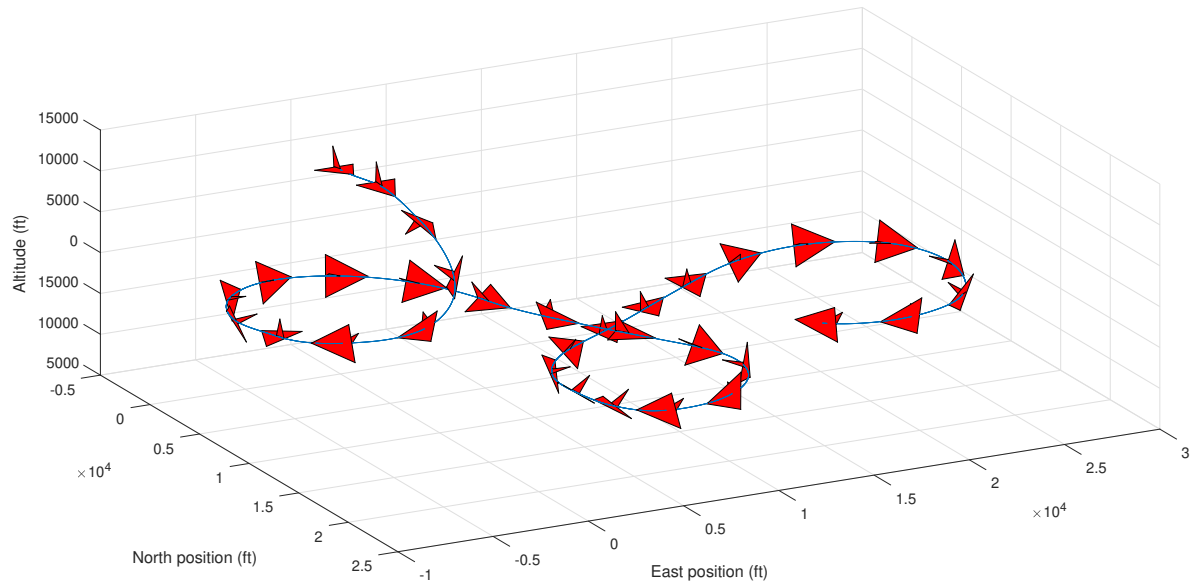


Figure B-1: Reference trajectory - Turns.

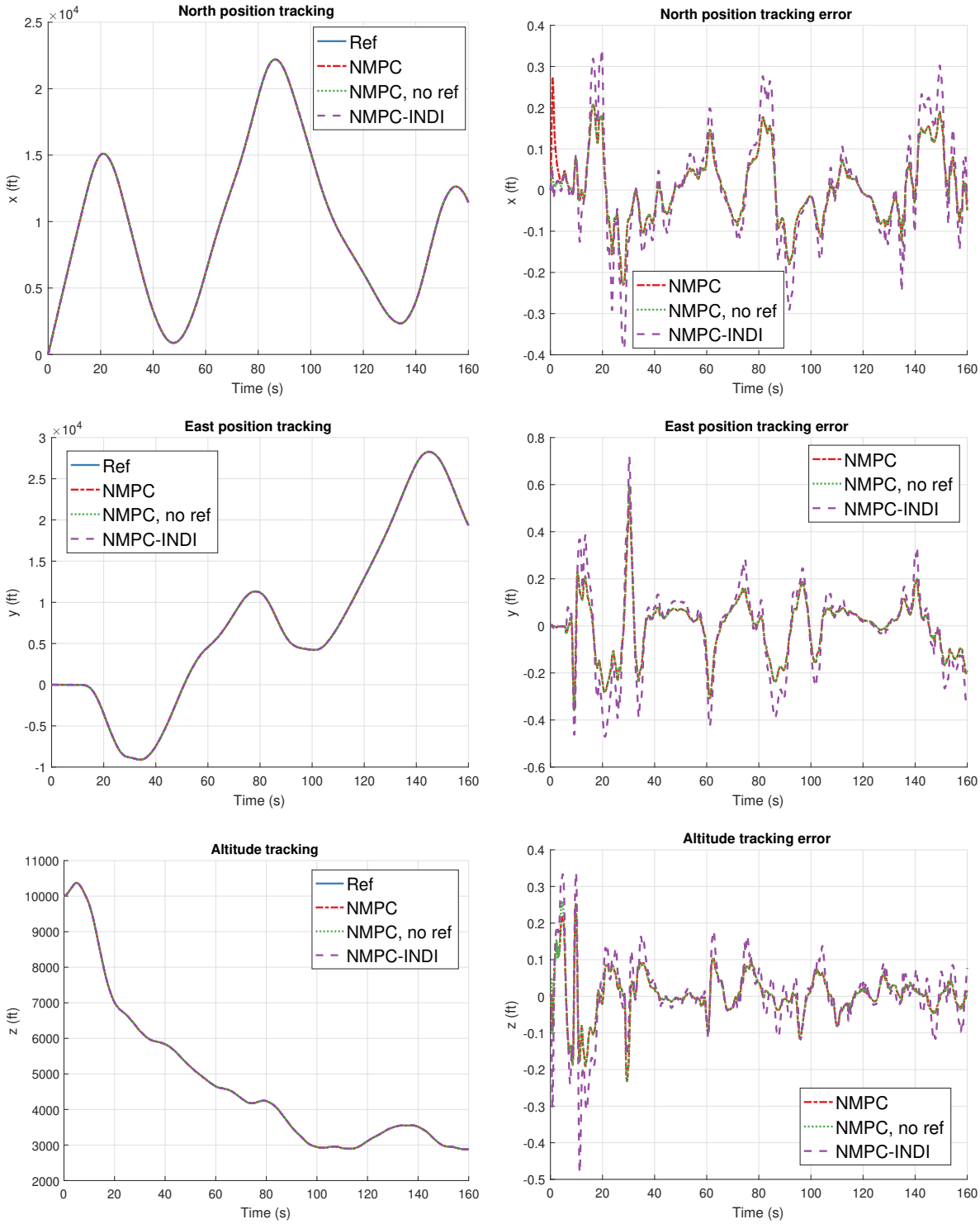


Figure B-2: Position tracking, track: Turns.

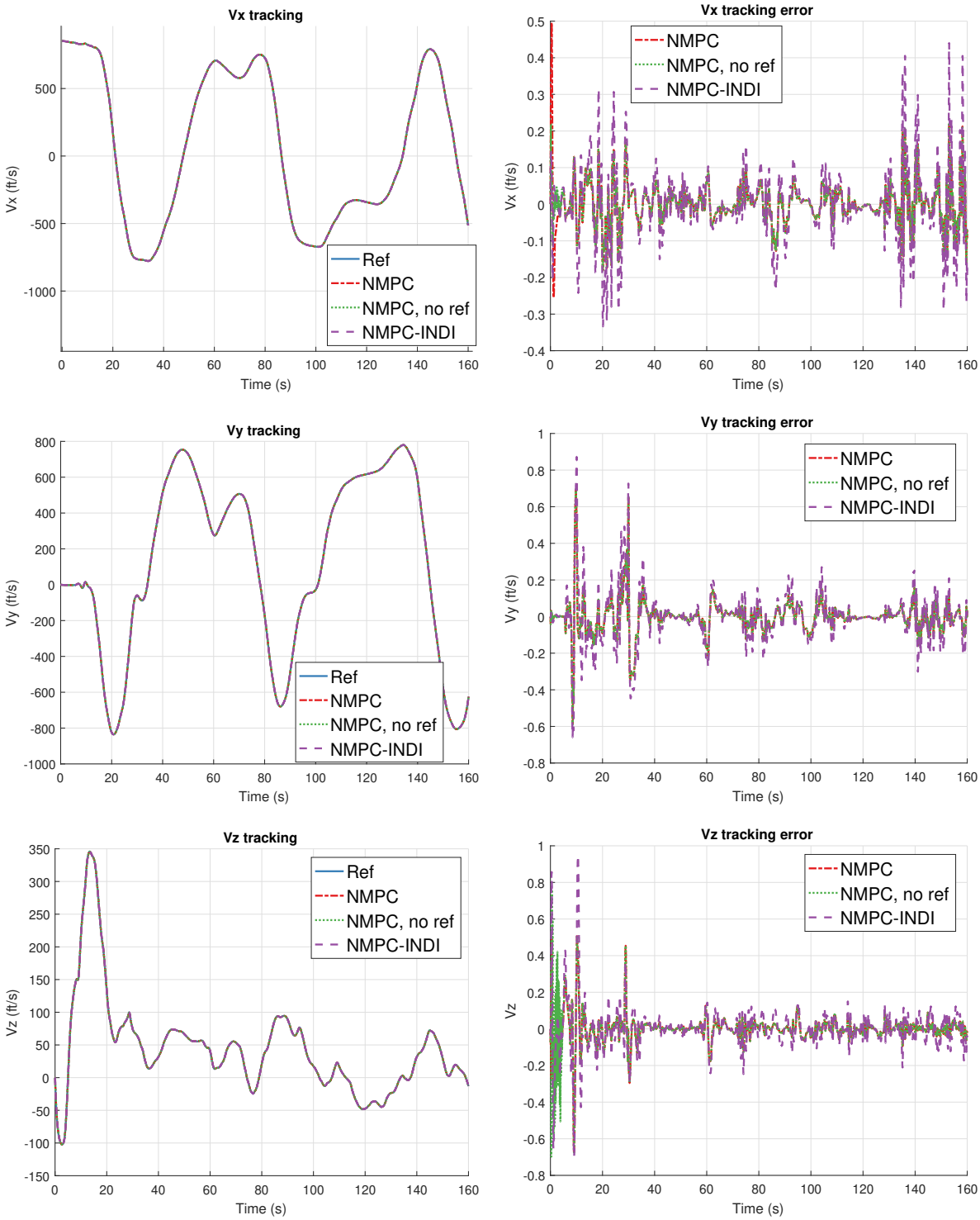


Figure B-3: Velocity tracking, track: Turns.

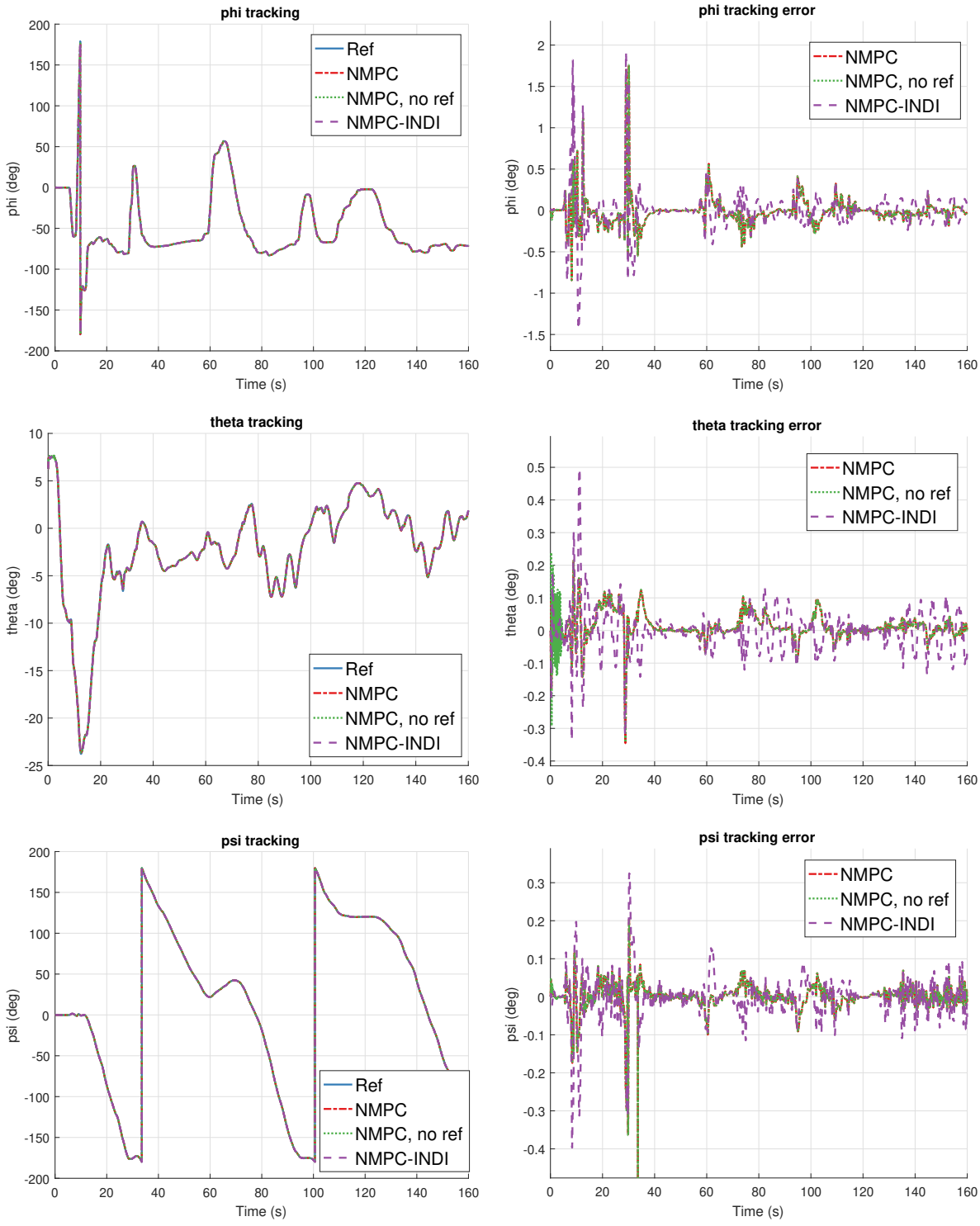


Figure B-4: Attitude angle tracking, track: Turns.

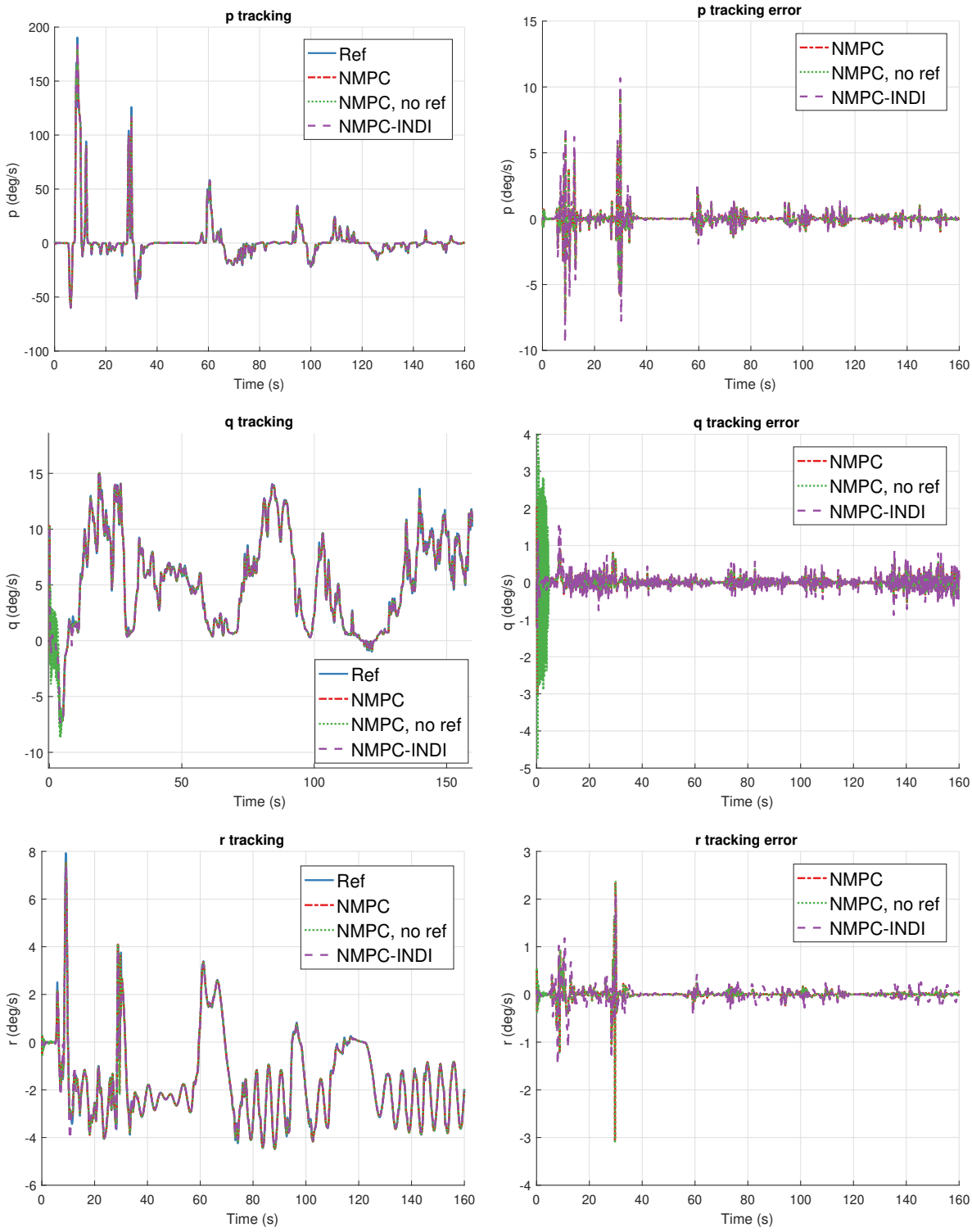


Figure B-5: Rotational rate tracking, track: Turns.

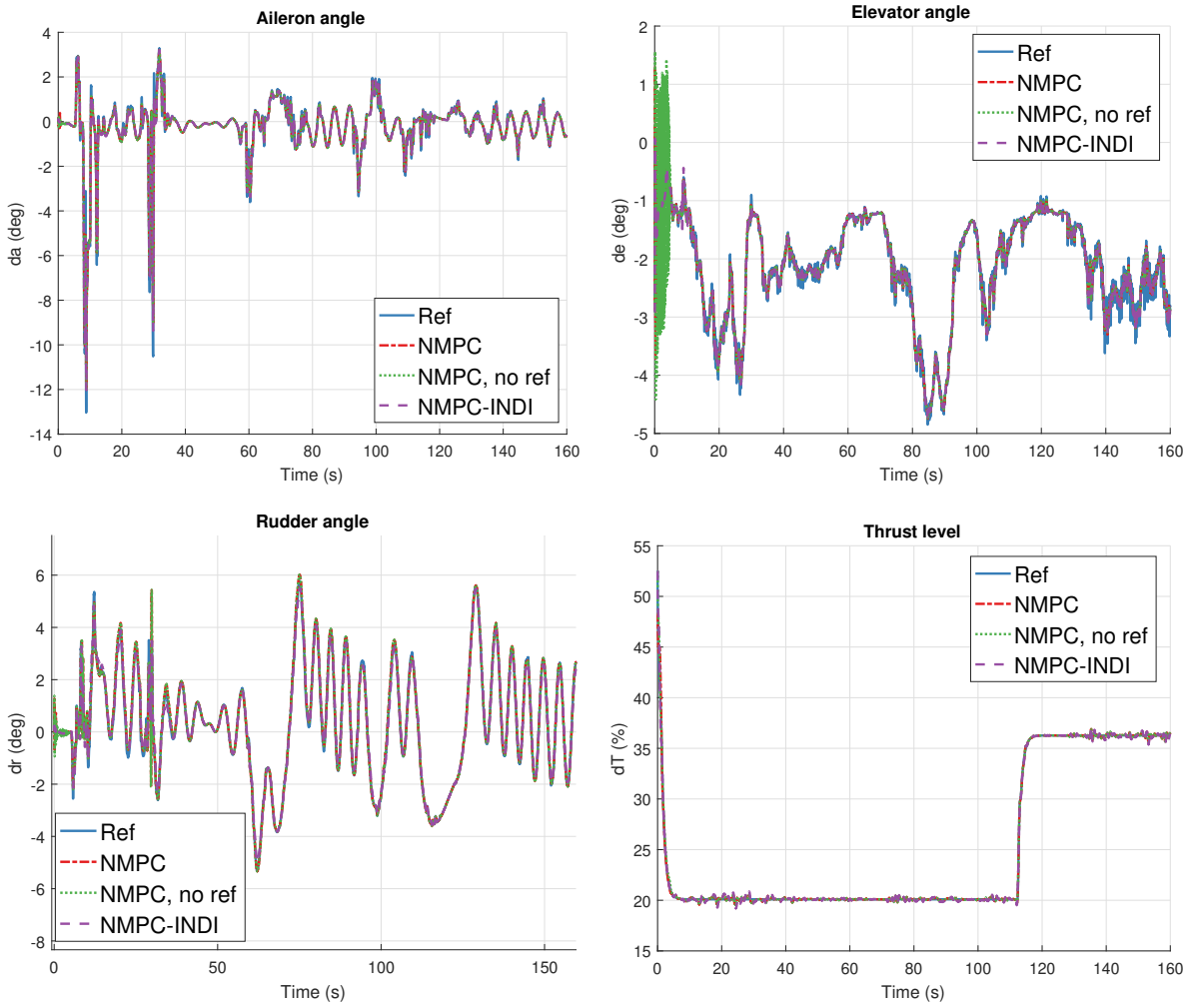


Figure B-6: Input reference tracking, track: Turns.

Appendix C

Tracking results of track: Aileron rolls

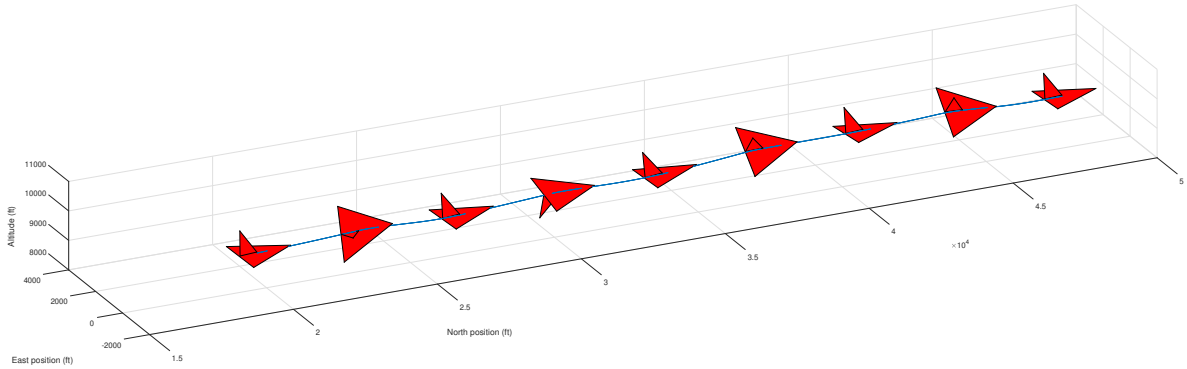


Figure C-1: Reference trajectory - Aileron rolls.

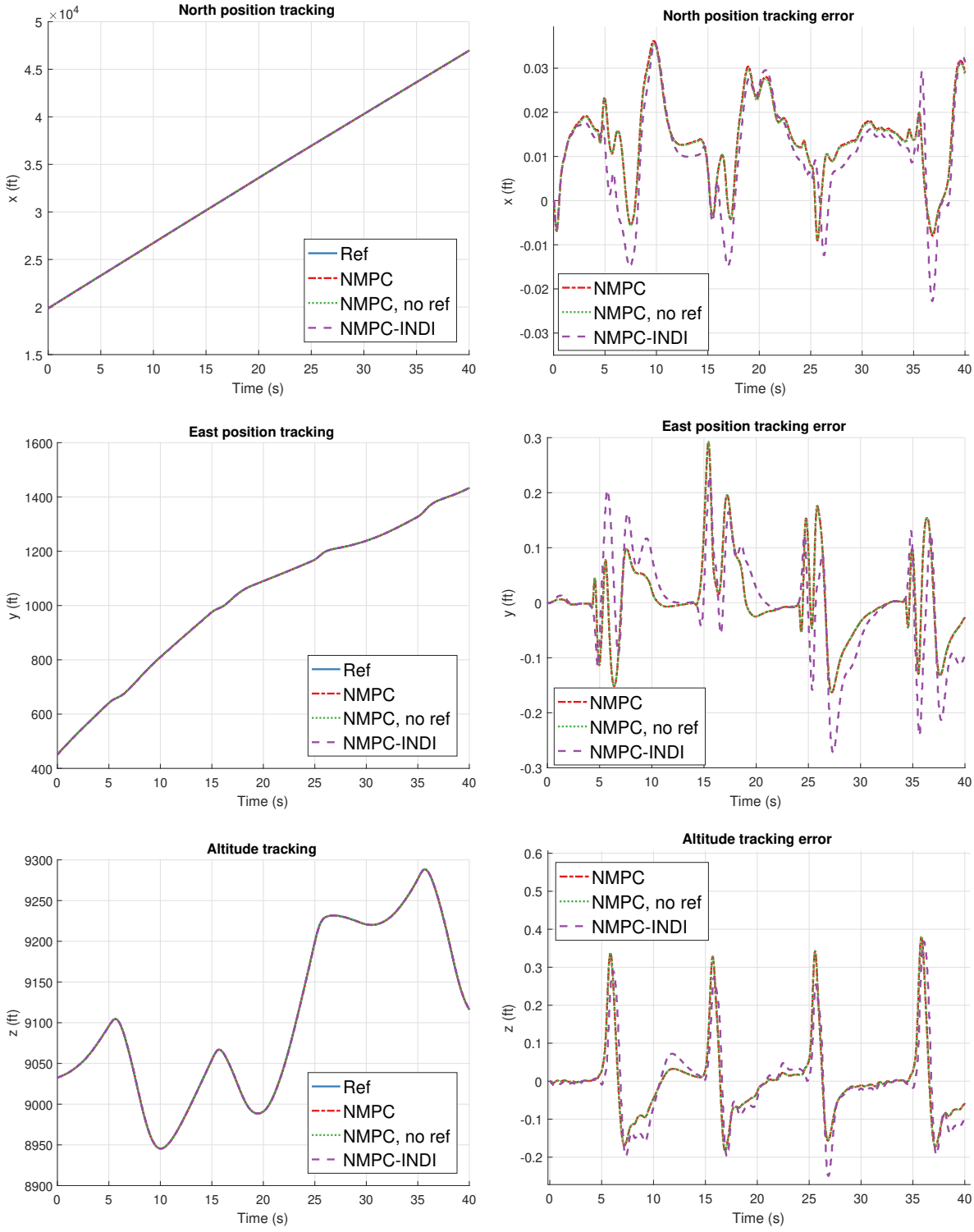


Figure C-2: Position tracking, track: Aileron rolls.

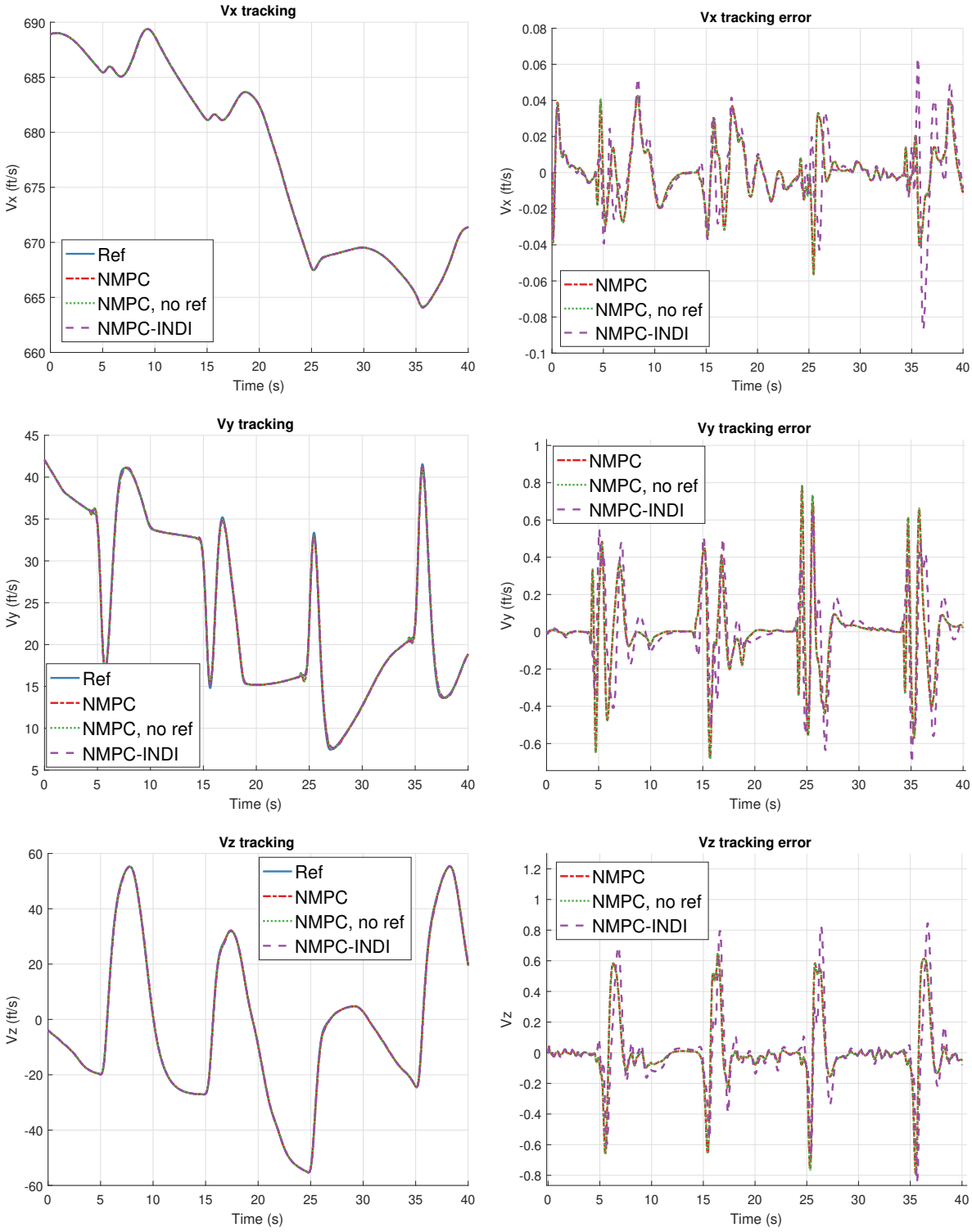


Figure C-3: Velocity tracking, track: Aileron rolls.

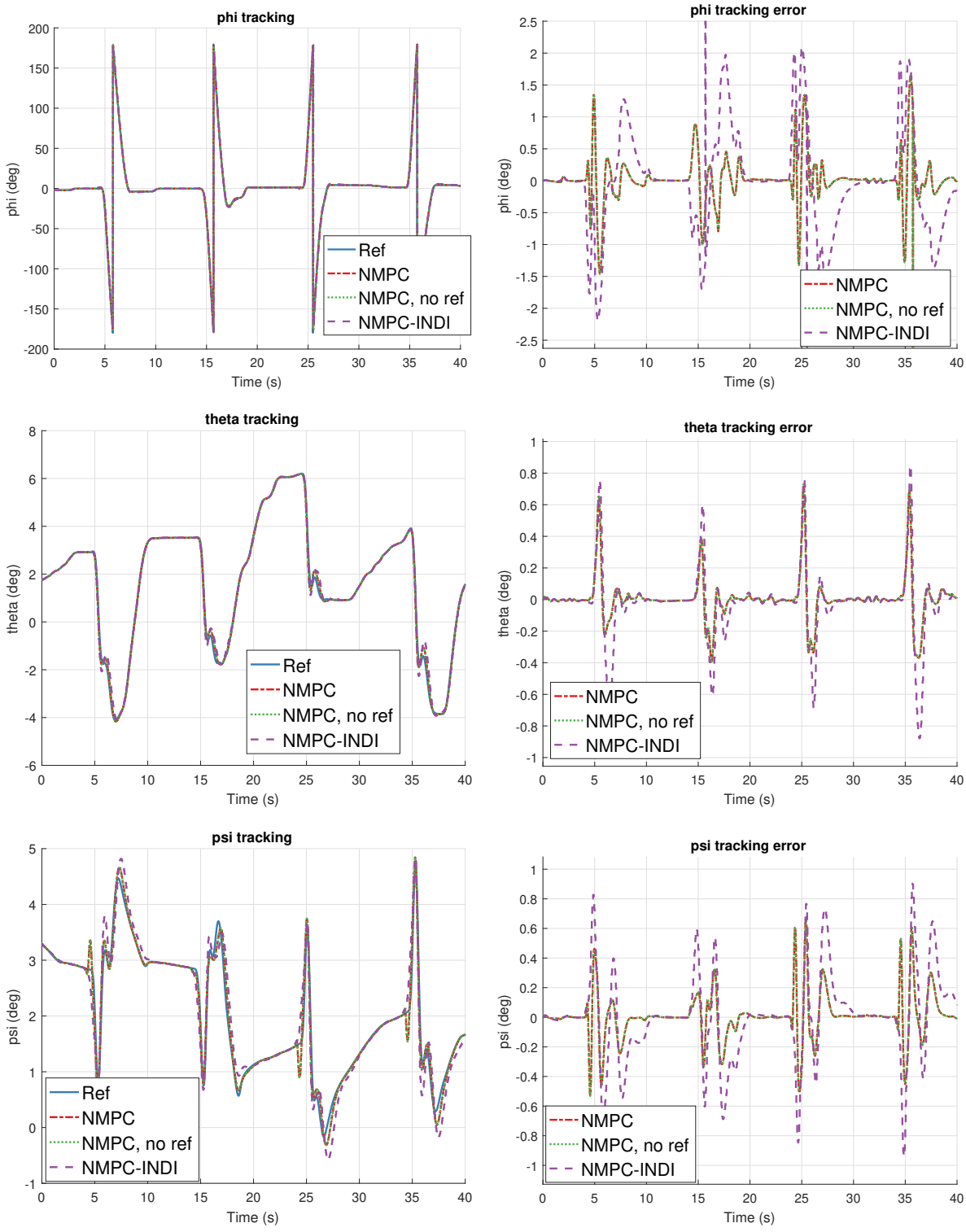


Figure C-4: Attitude angle tracking, track: Aileron rolls.

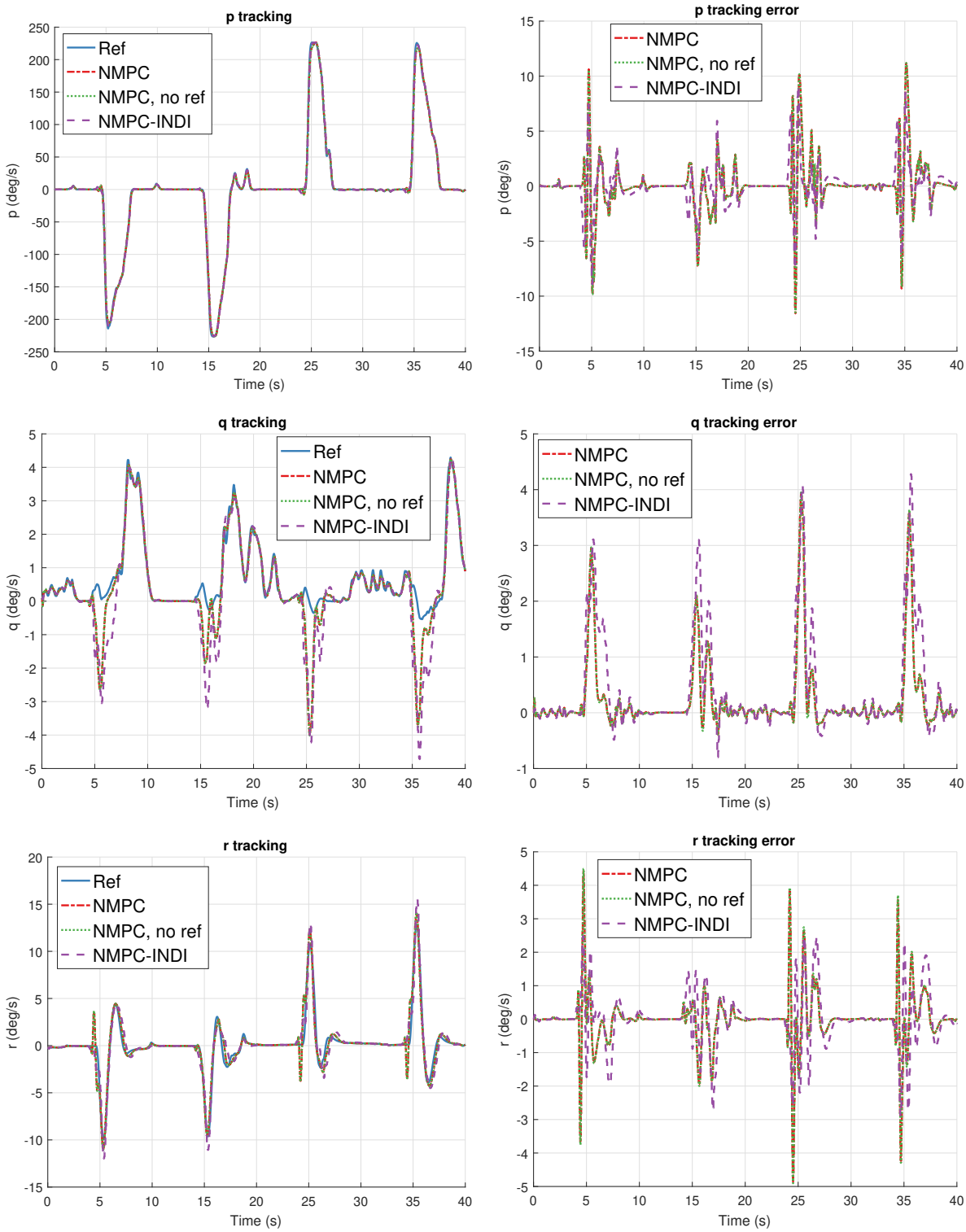


Figure C-5: Rotational rate tracking, track: Aileron rolls.

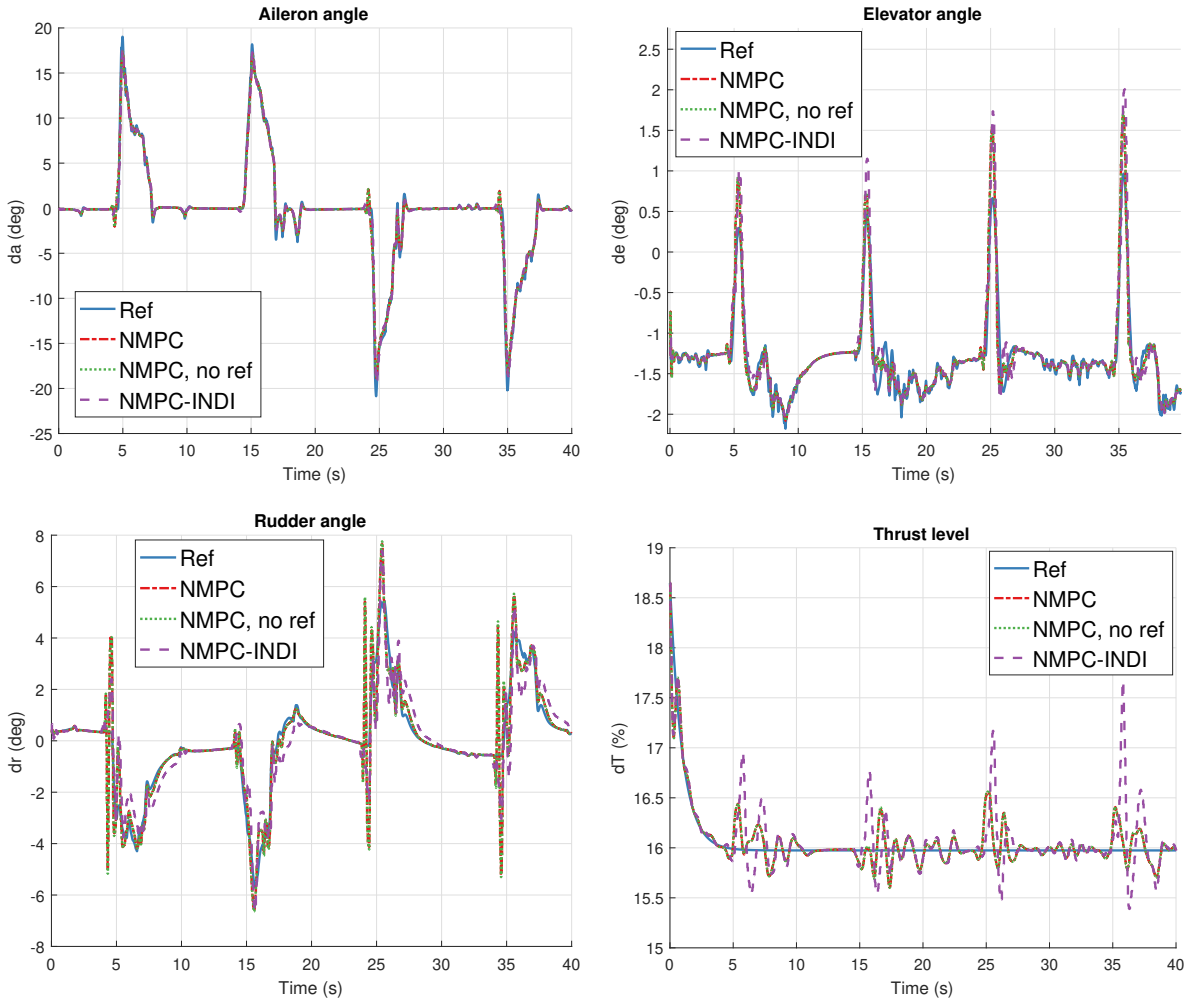


Figure C-6: Input reference tracking, track: Aileron rolls.

Appendix D

Tracking results of track: Barrel roll

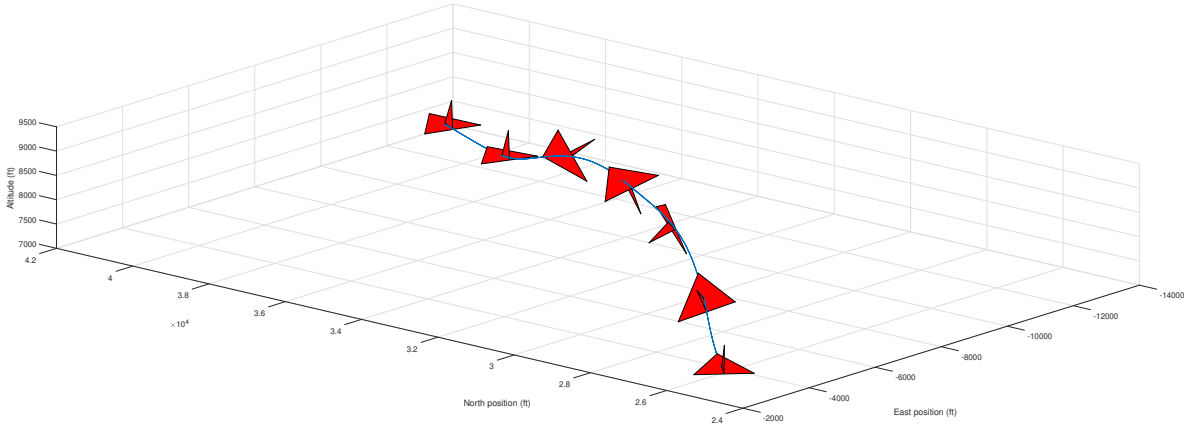


Figure D-1: Reference trajectory - Barrel roll.

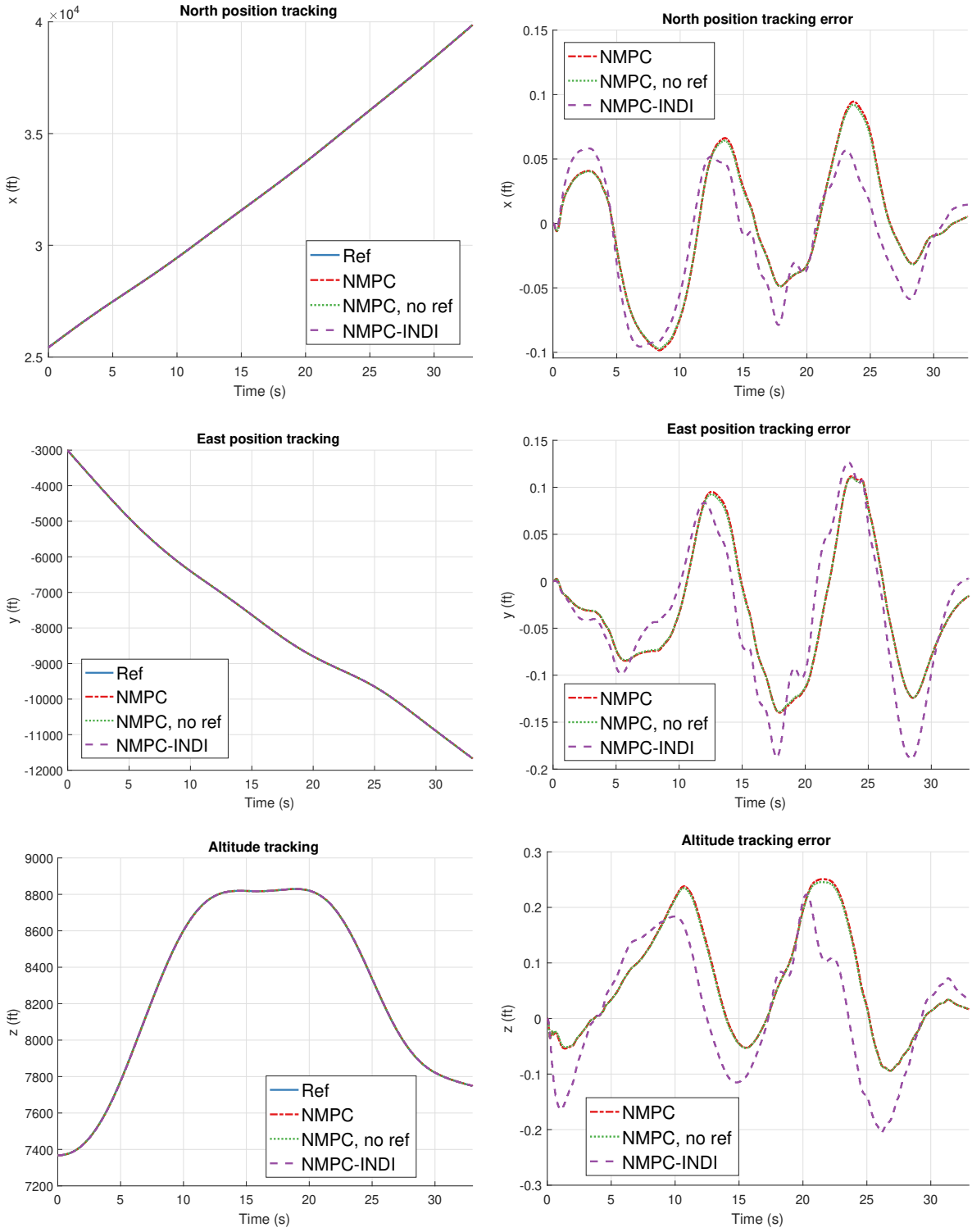


Figure D-2: Position tracking, track: Barrel roll.

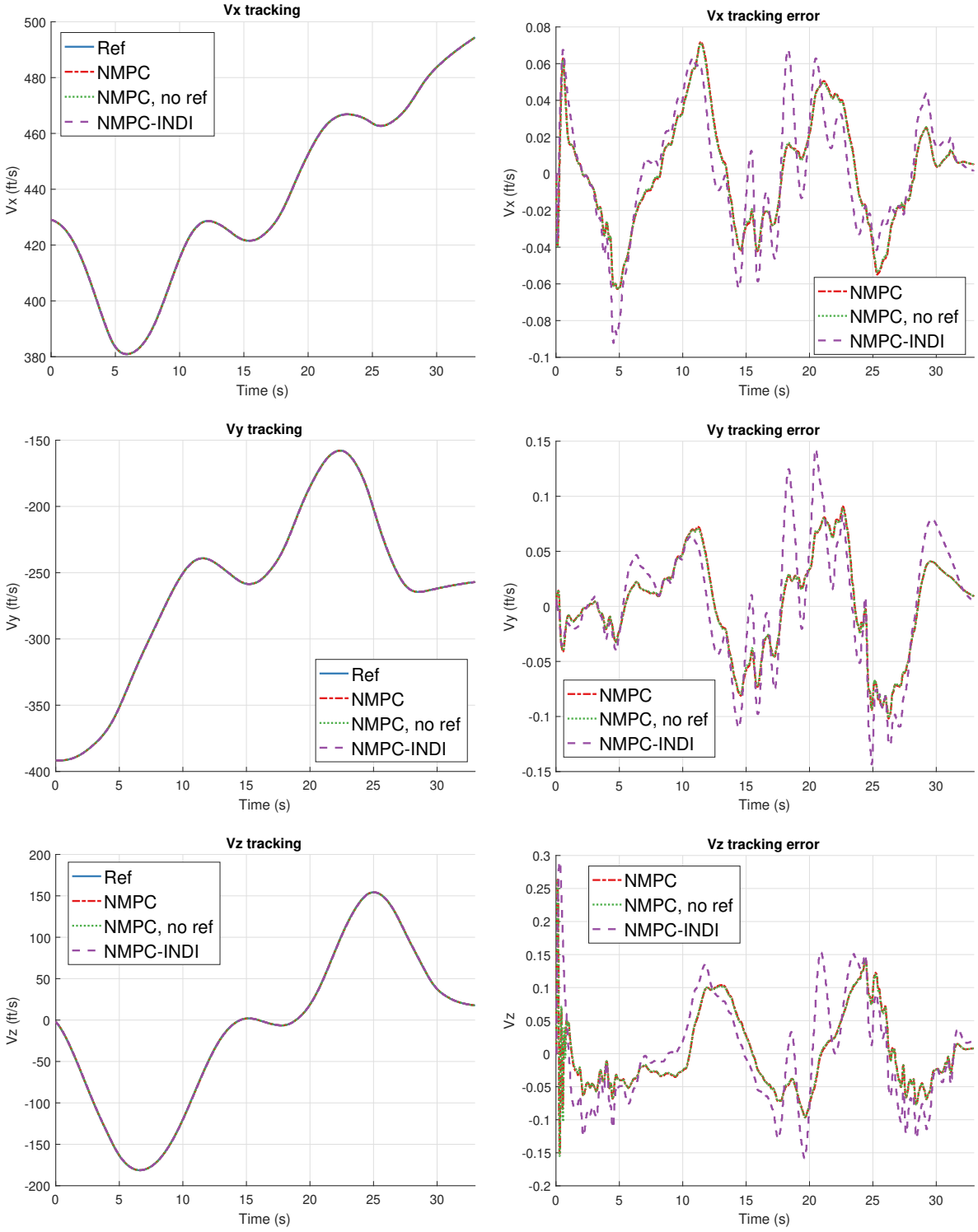


Figure D-3: Velocity tracking, track: Barrel roll.

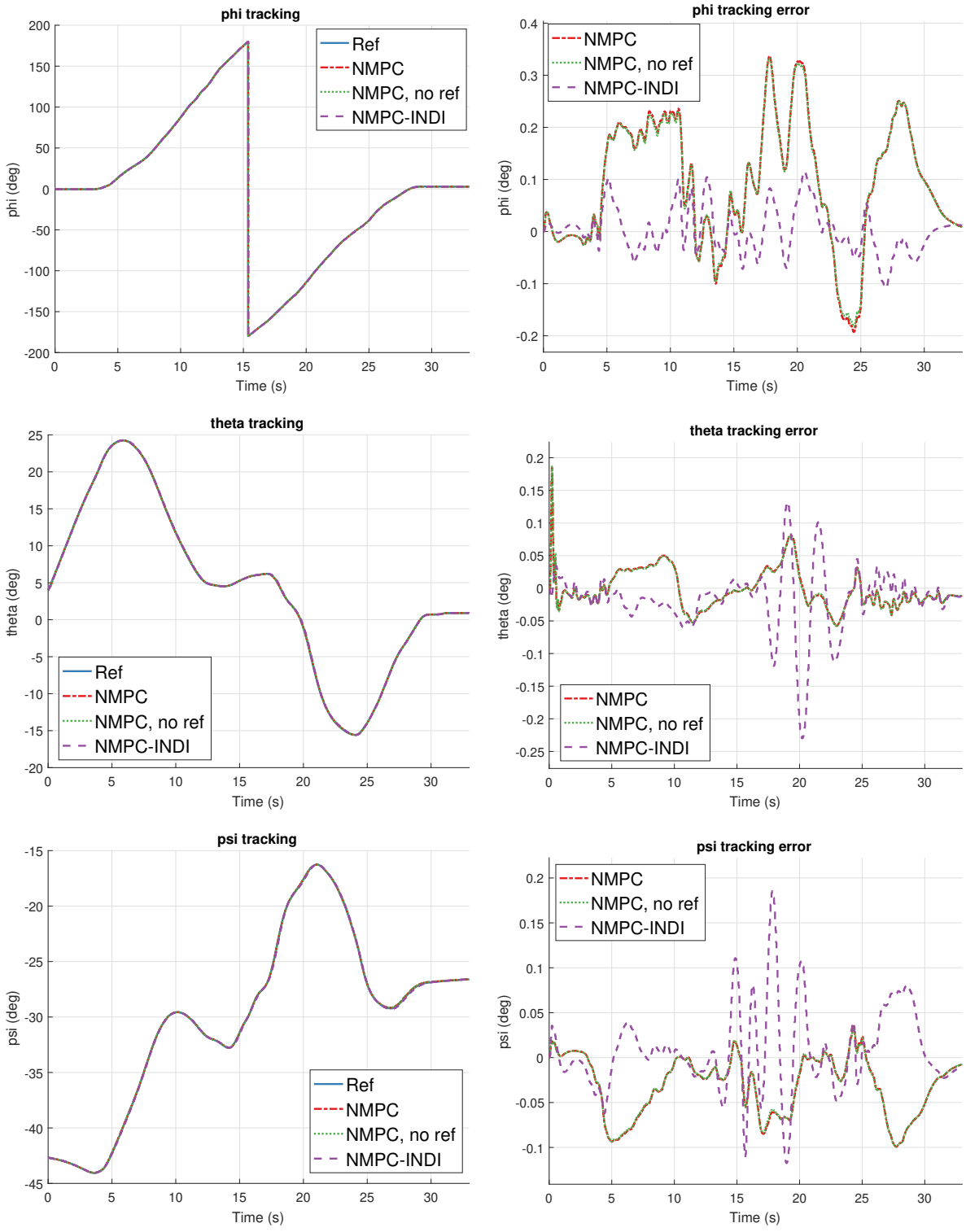


Figure D-4: Attitude angle tracking, track: Barrel roll.

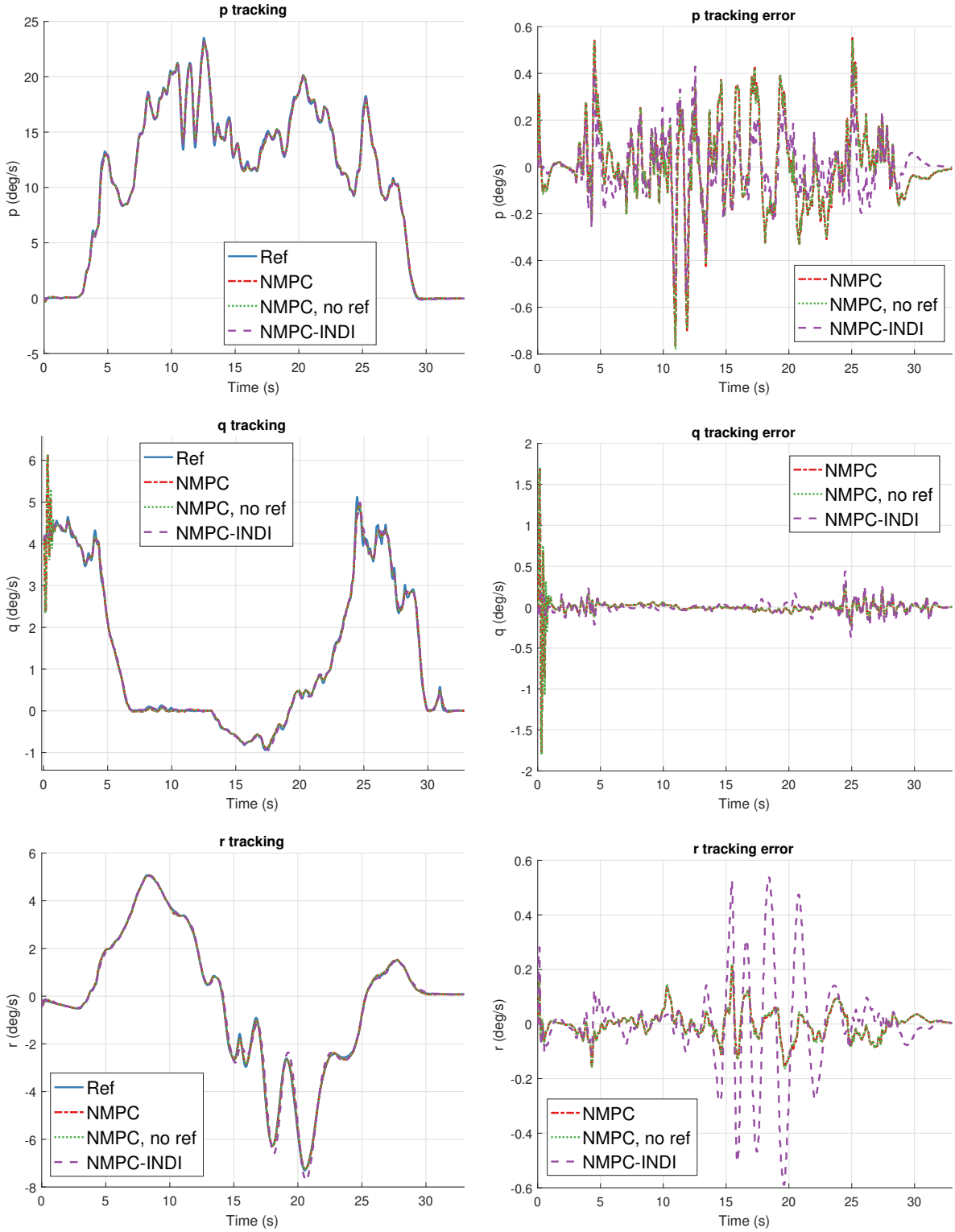


Figure D-5: Rotational rate tracking, track: Barrel roll.

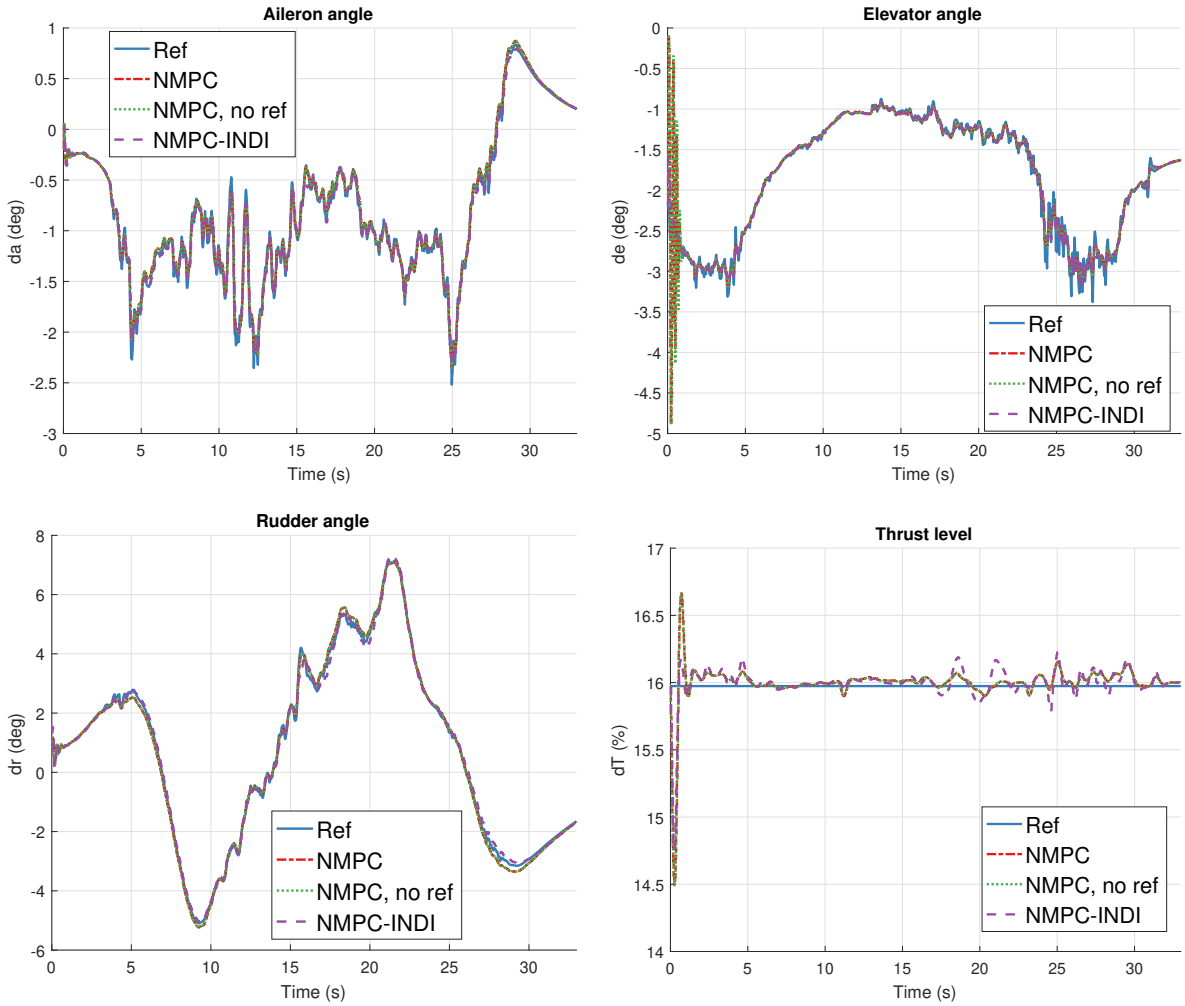


Figure D-6: Input reference tracking, track: Barrel roll.

Appendix E

Tracking results of track: Loop

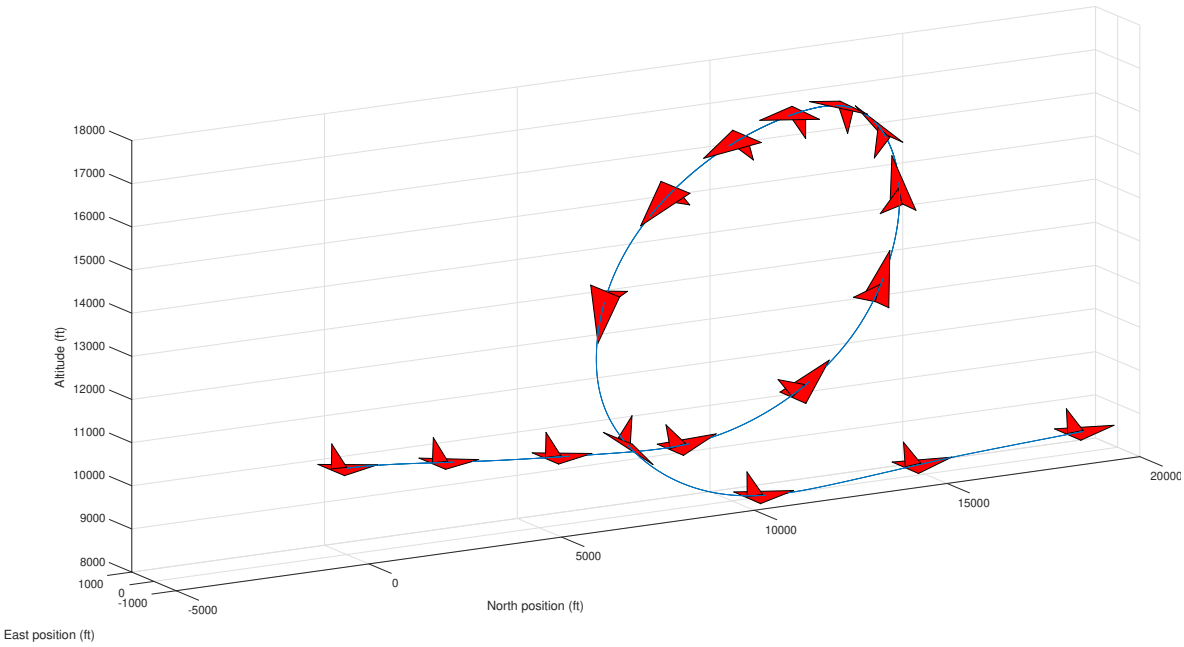


Figure E-1: Reference trajectory - Loop.

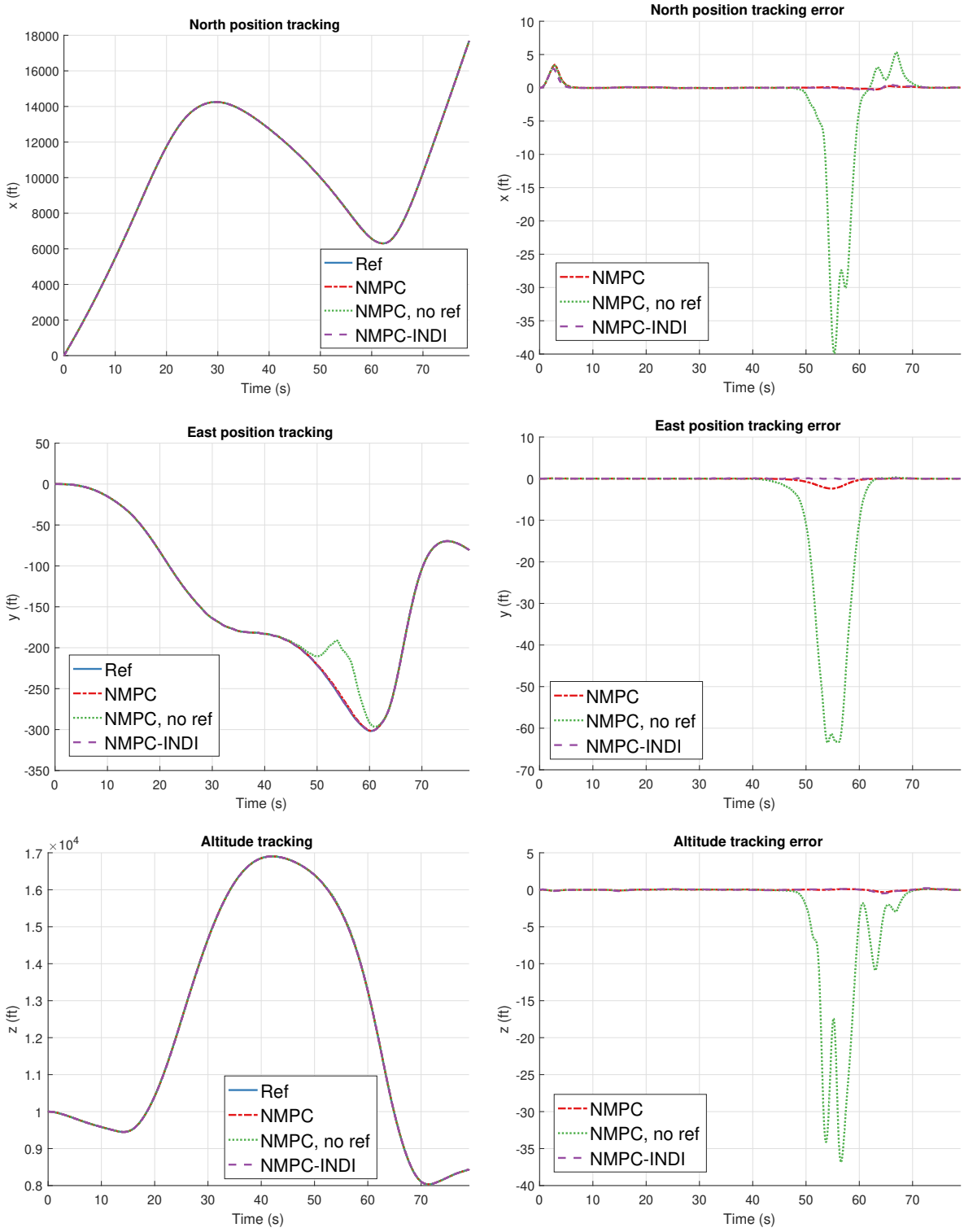


Figure E-2: Position tracking, track: Loop.

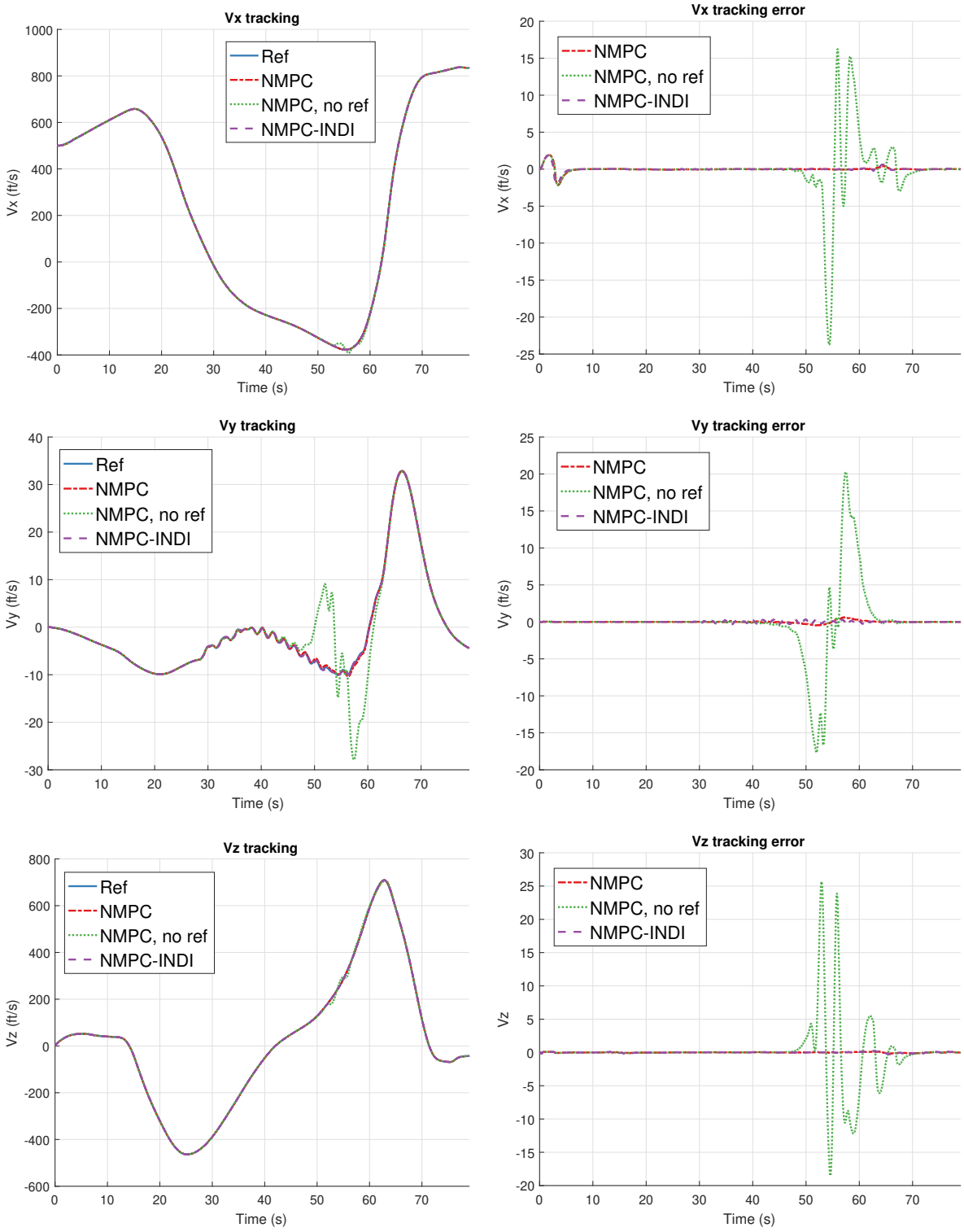


Figure E-3: Velocity tracking, track: Loop.

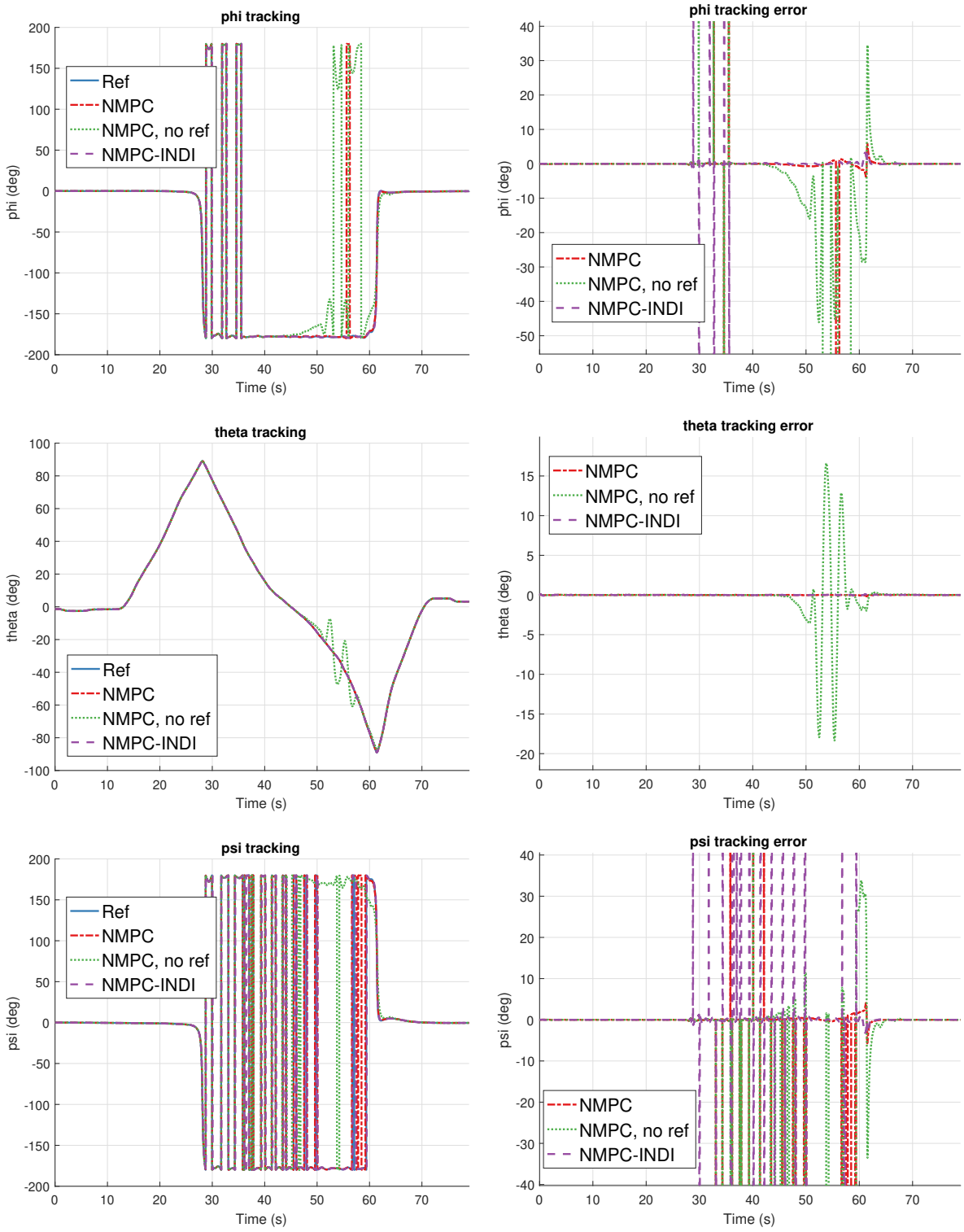


Figure E-4: Attitude angle tracking, track: Loop.

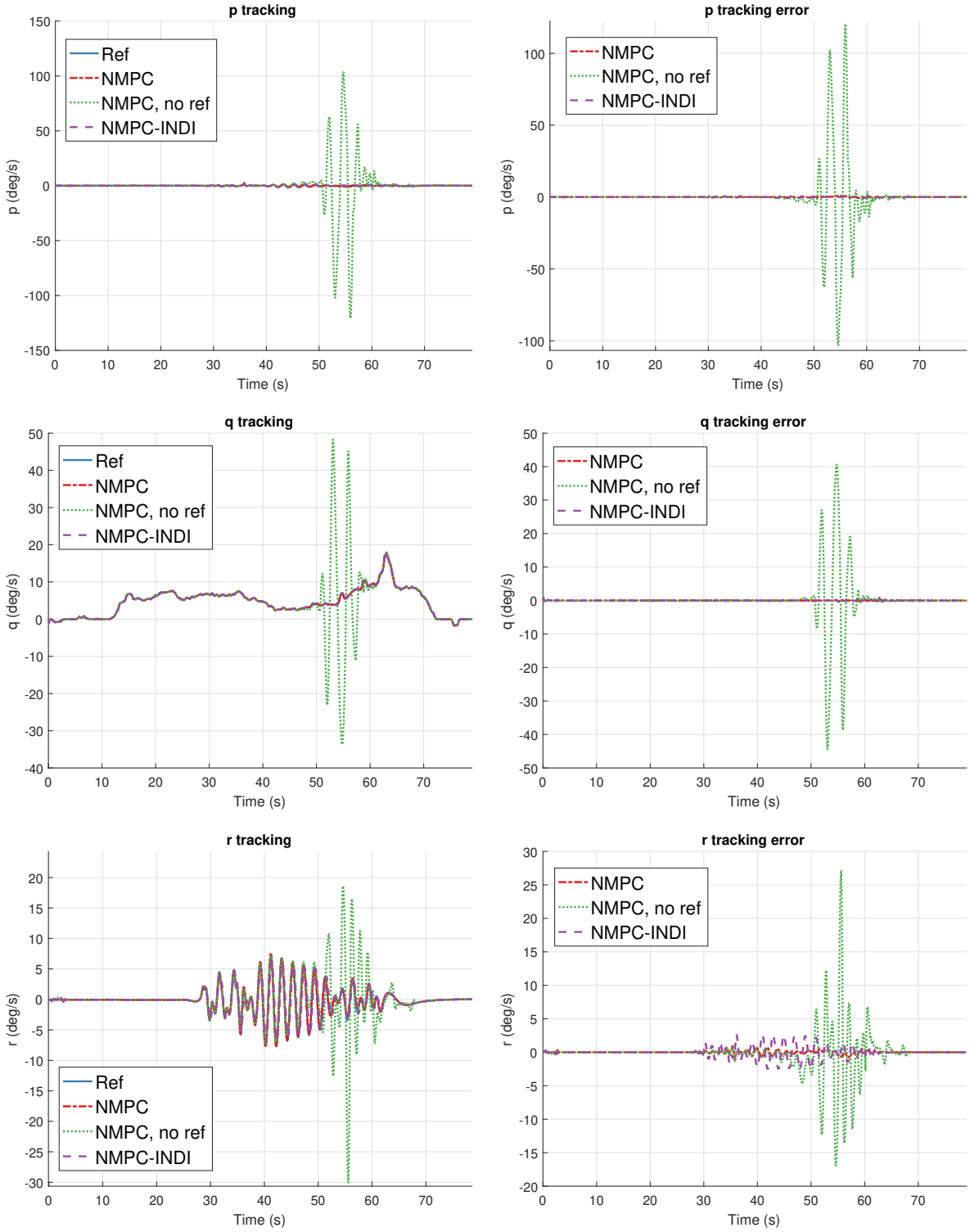


Figure E-5: Rotational rate tracking, track: Loop.

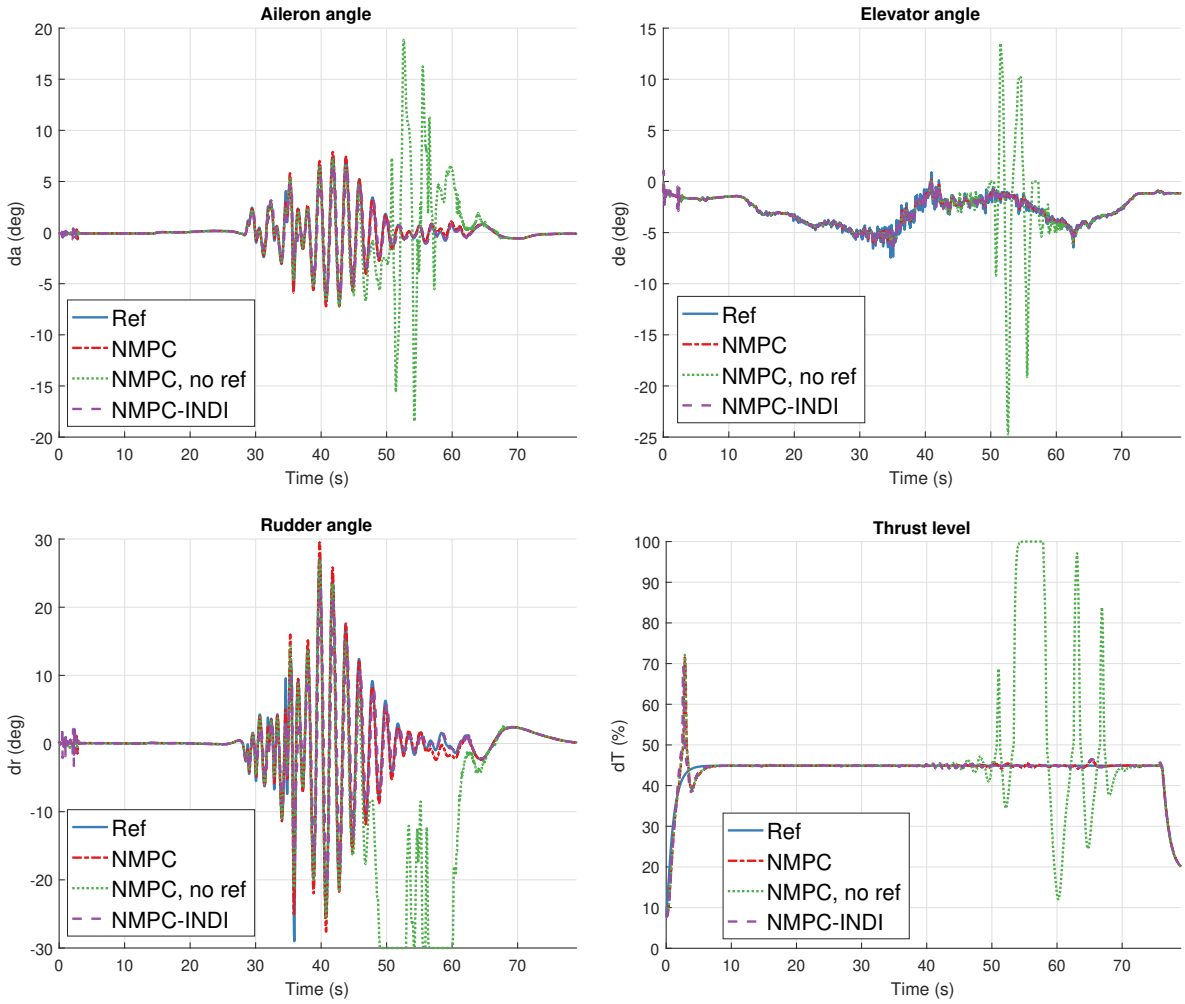


Figure E-6: Input reference tracking, track: Loop.

Appendix F

Tracking results of track: Cuban

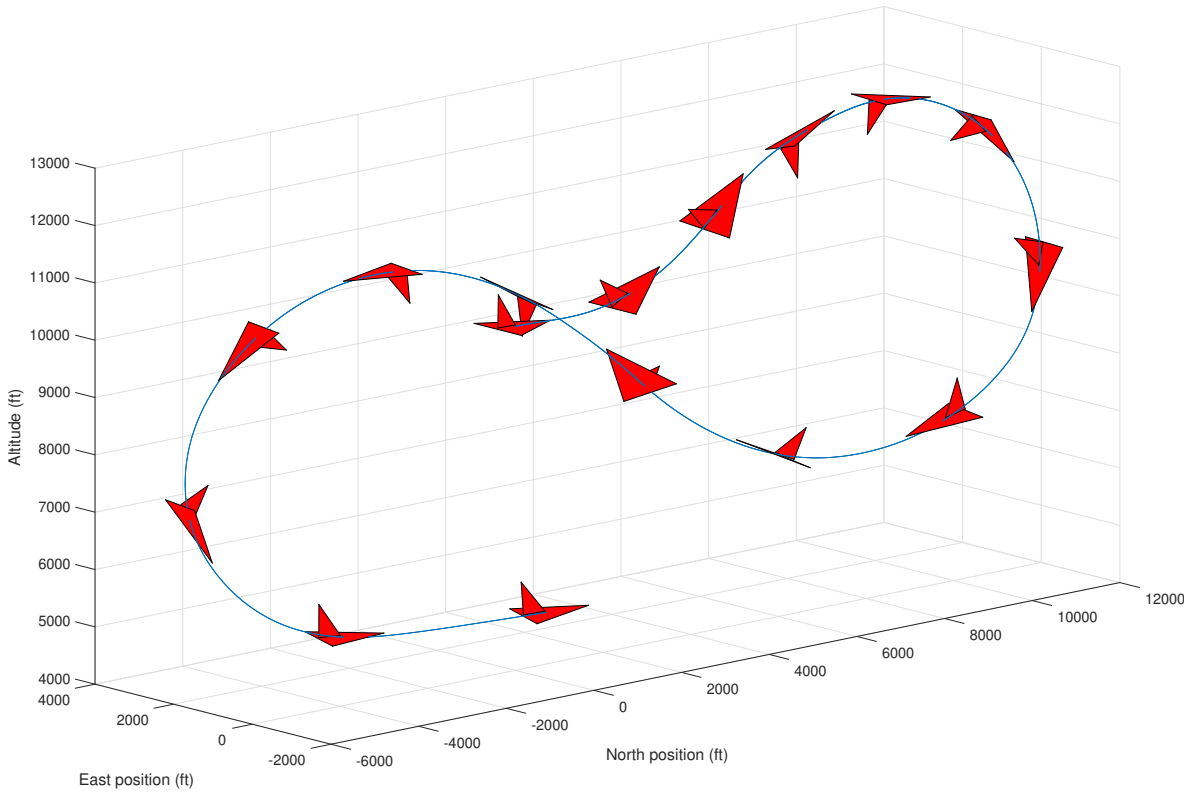


Figure F-1: Reference trajectory - Half Cuban Eight.

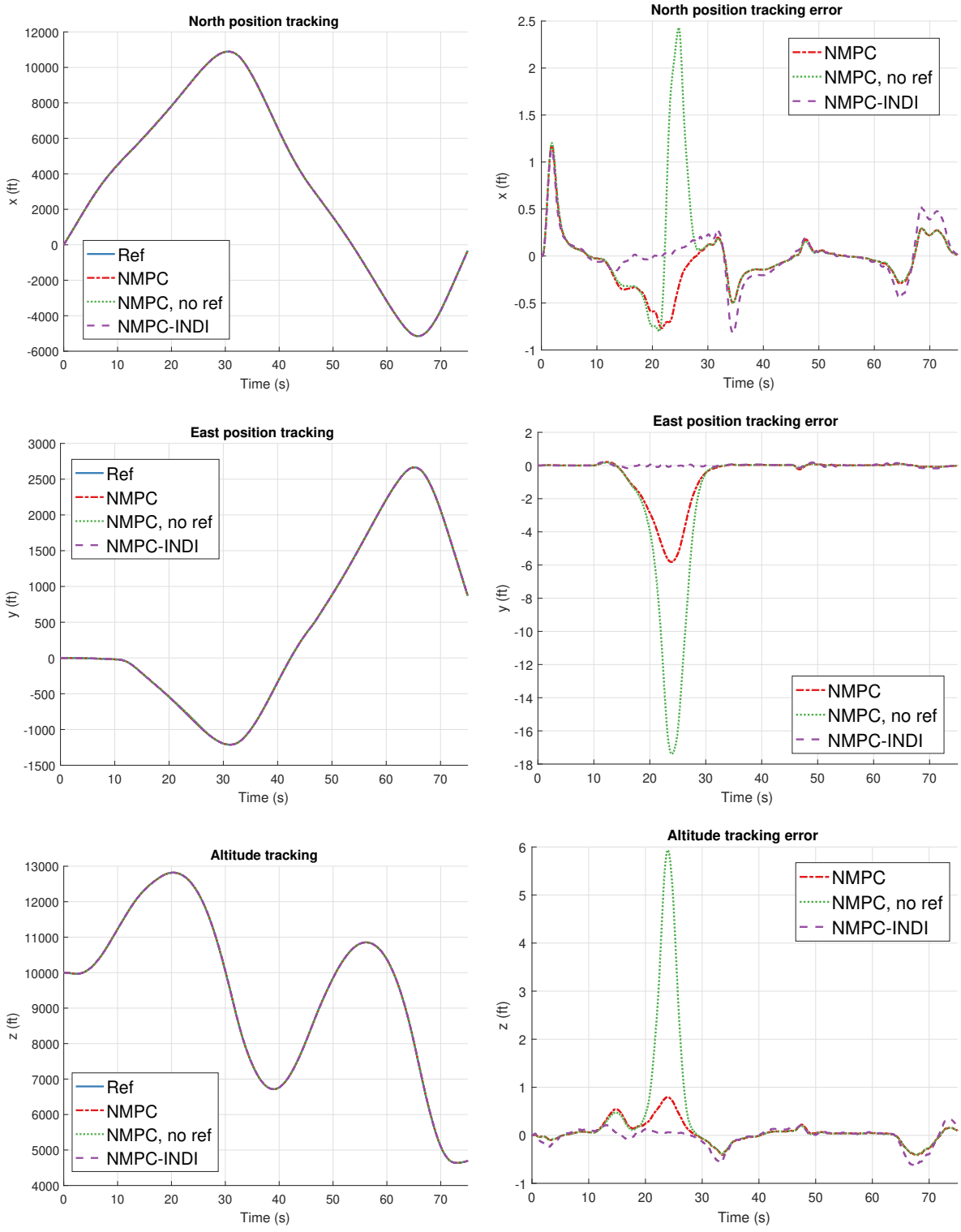


Figure F-2: Position tracking, track: Half Cuban Eight.

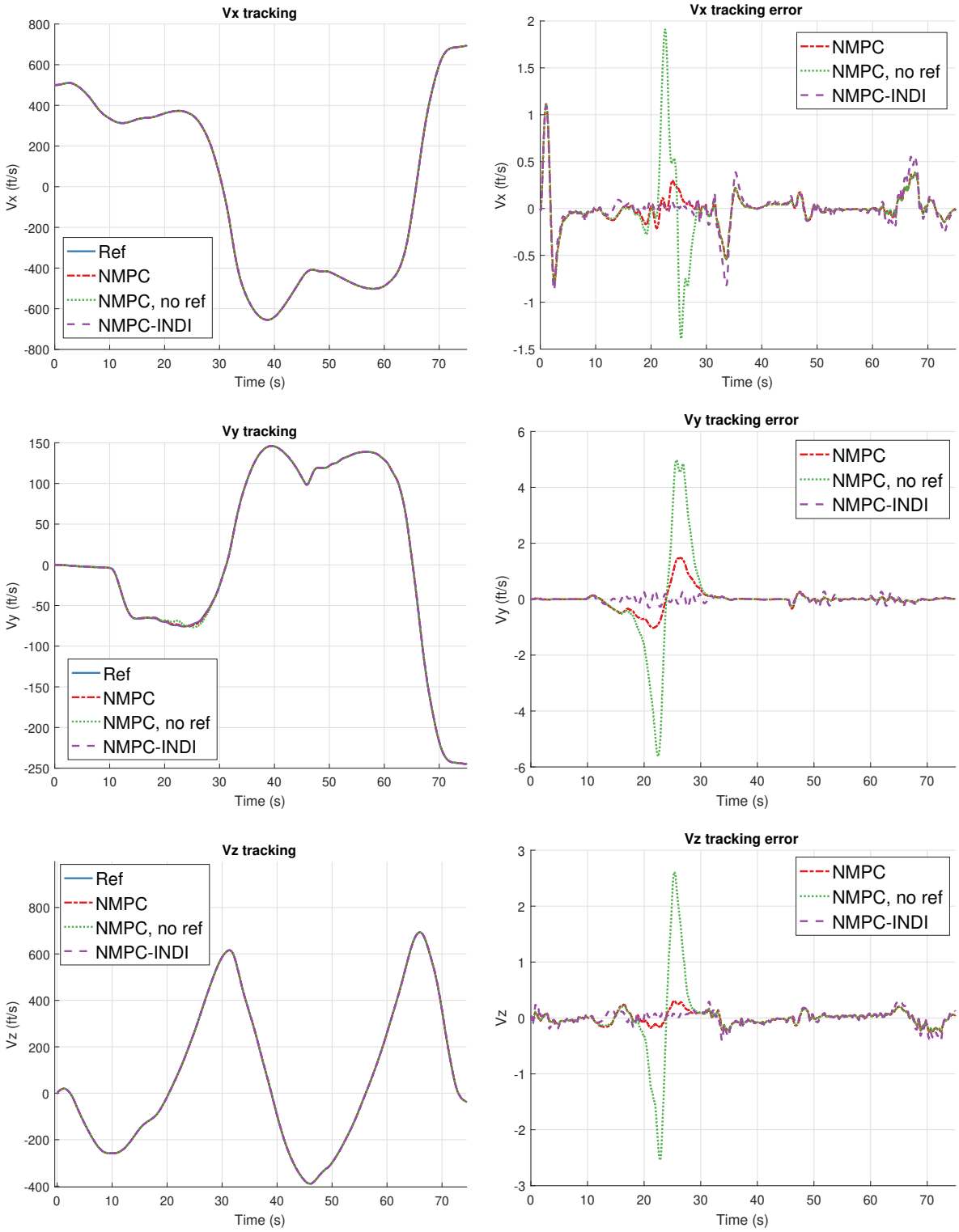


Figure F-3: Velocity tracking, track: Half Cuban Eight.

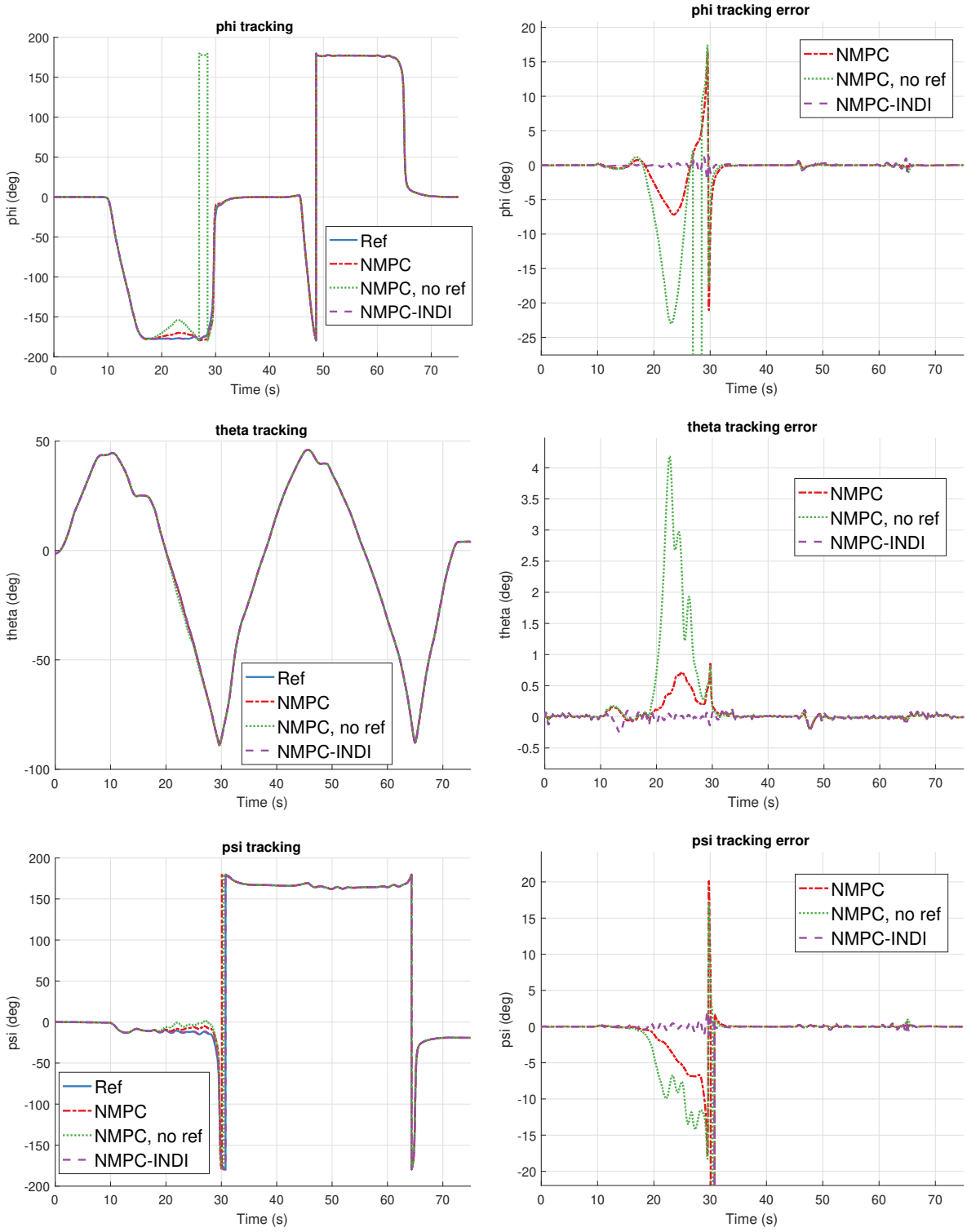


Figure F-4: Attitude angle tracking, track: Half Cuban Eight.

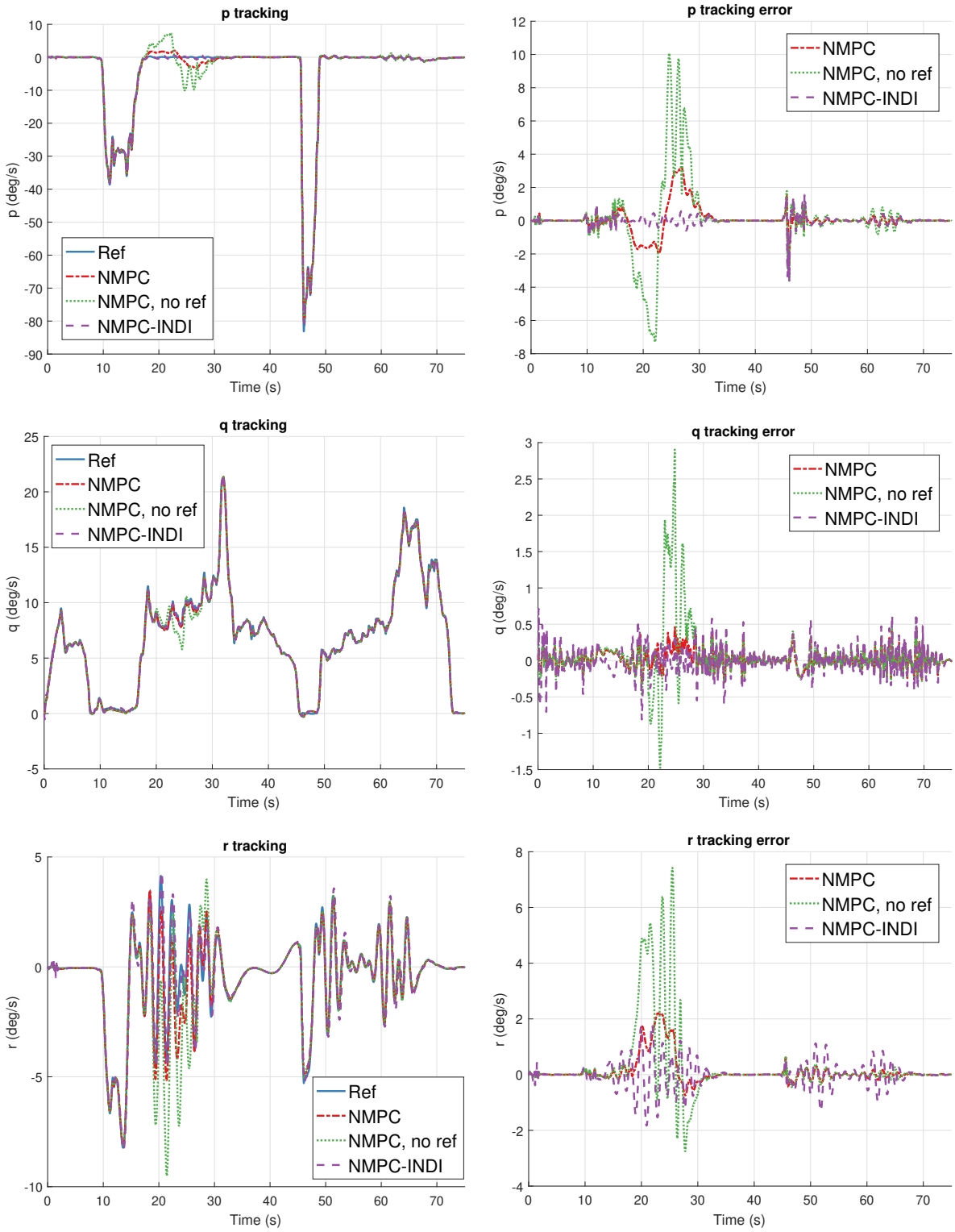


Figure F-5: Rotational rate tracking, track: Half Cuban Eight.

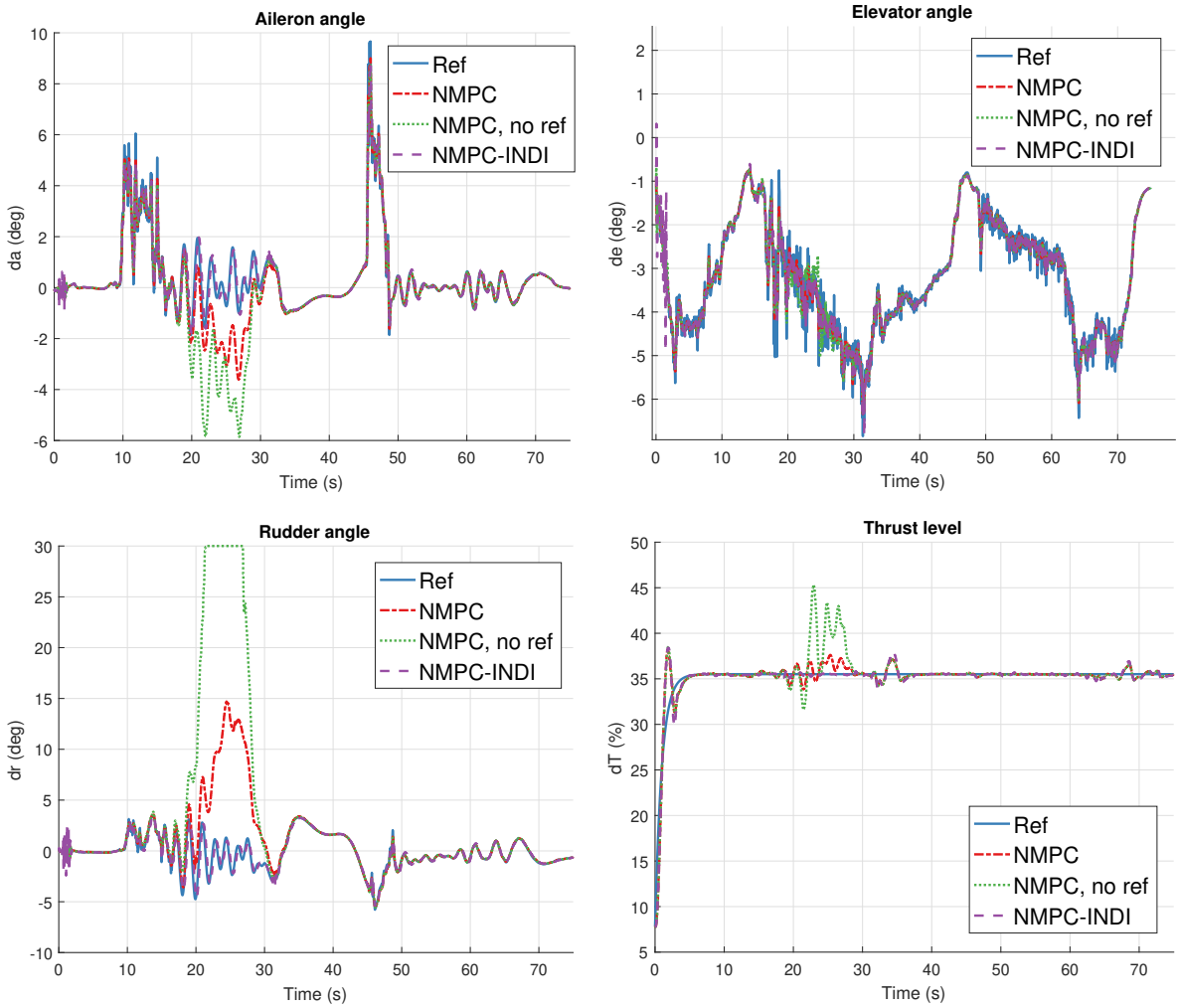


Figure F-6: Input reference tracking, track: Half Cuban Eight.

Appendix G

Tracking results of track: Recovery

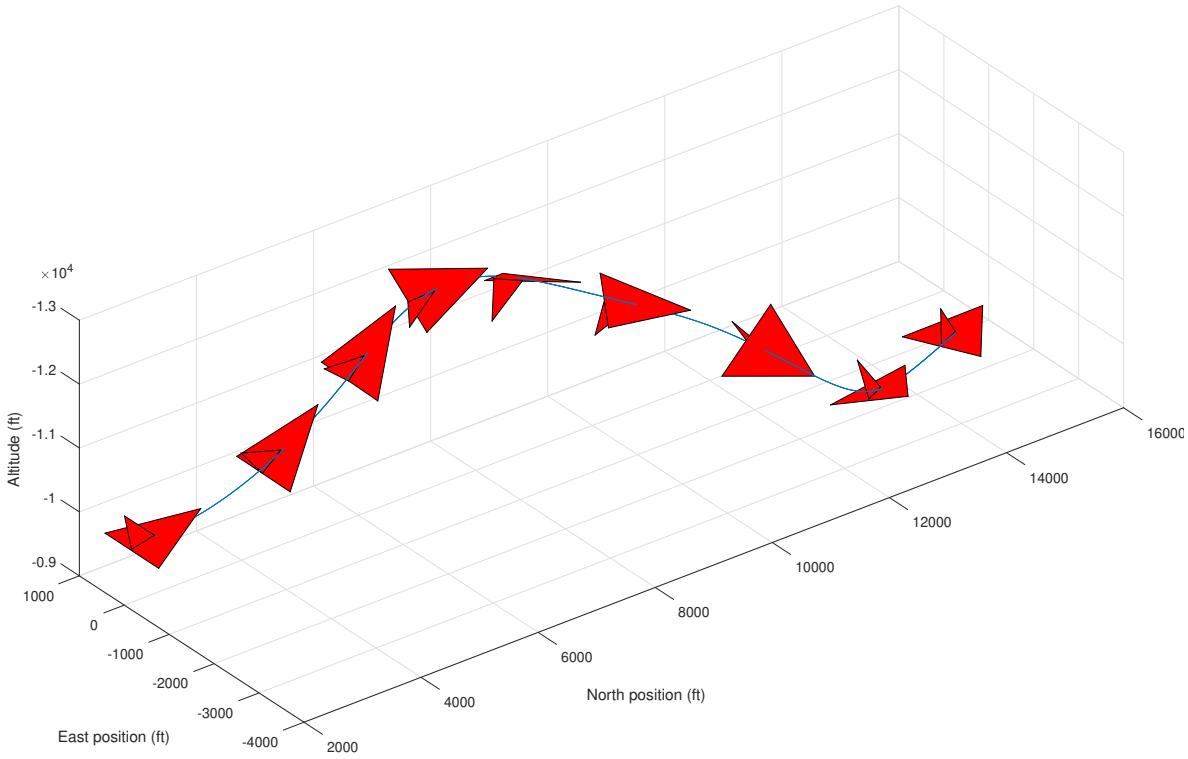


Figure G-1: Reference trajectory - Recovery.

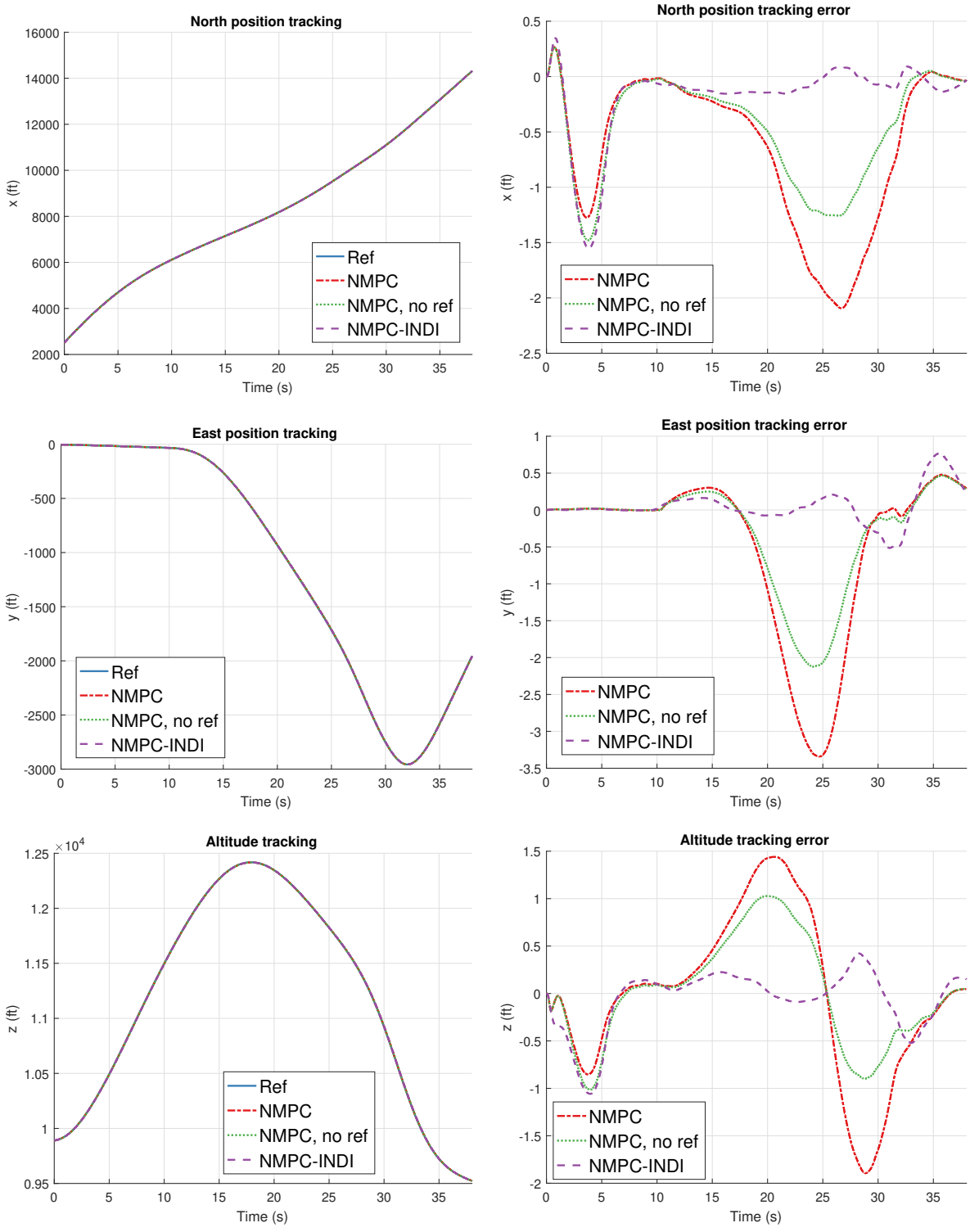


Figure G-2: Position tracking, track: Recovery.

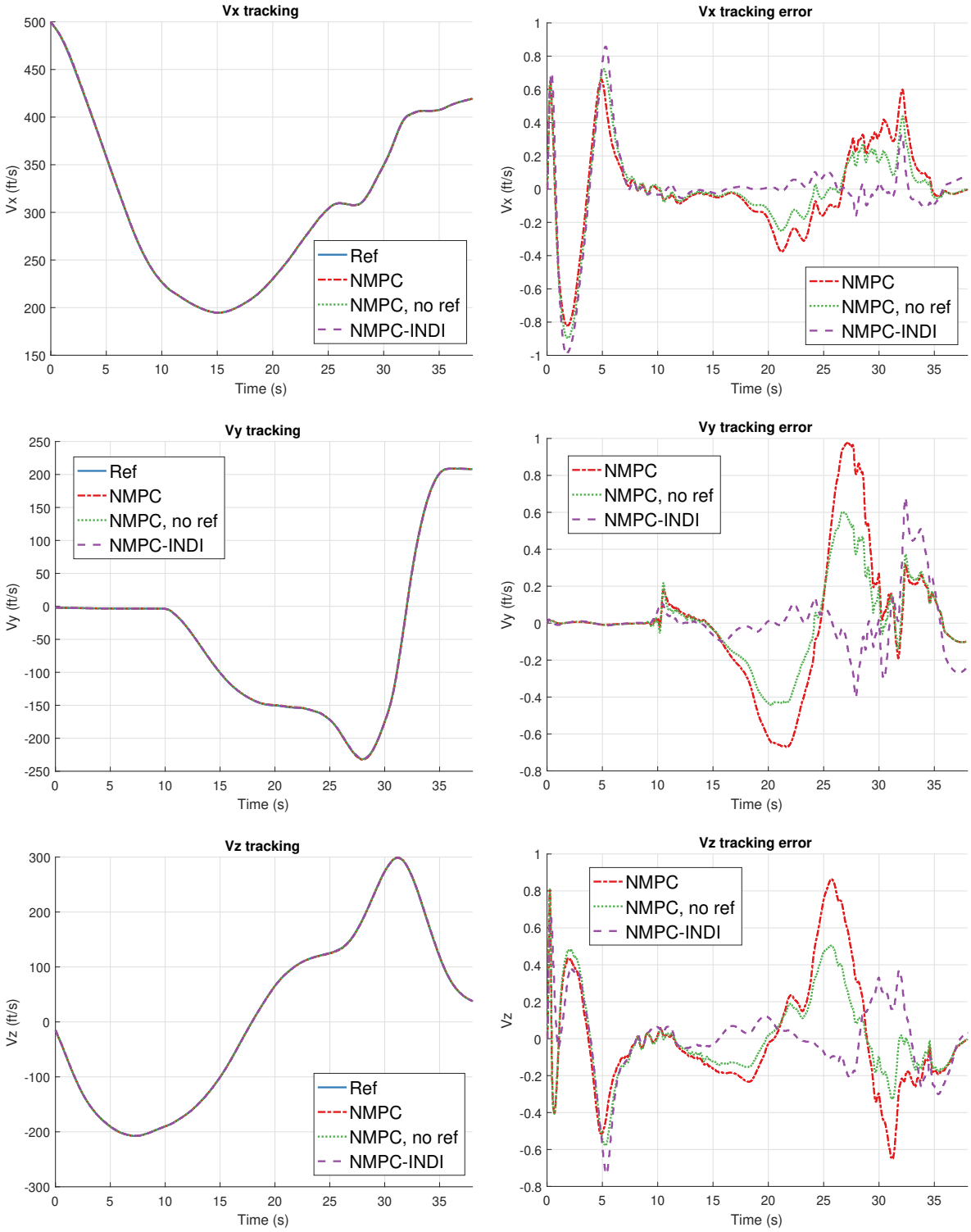


Figure G-3: Velocity tracking, track: Recovery.

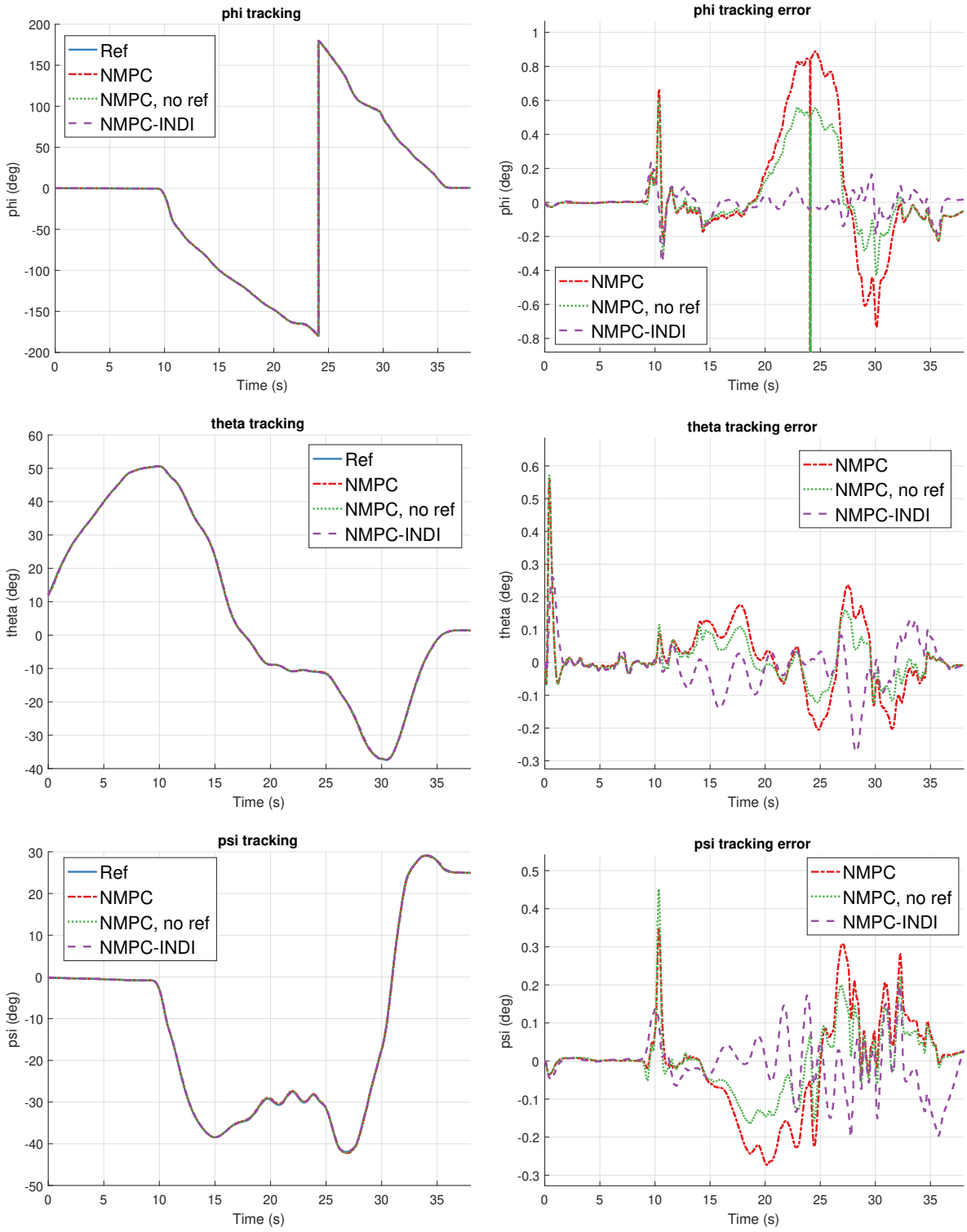


Figure G-4: Attitude angle tracking, track: Recovery.

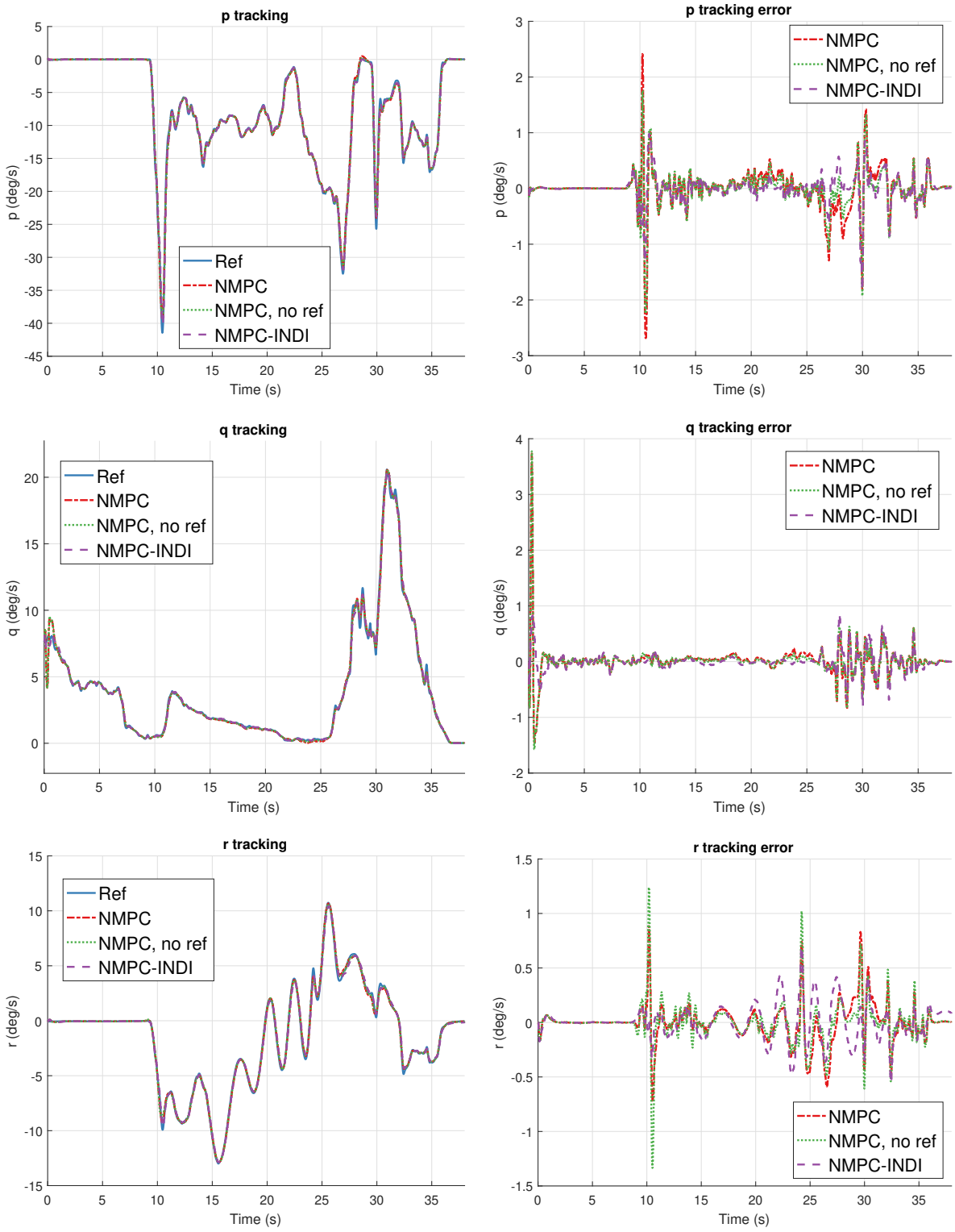


Figure G-5: Rotational rate tracking, track: Recovery.

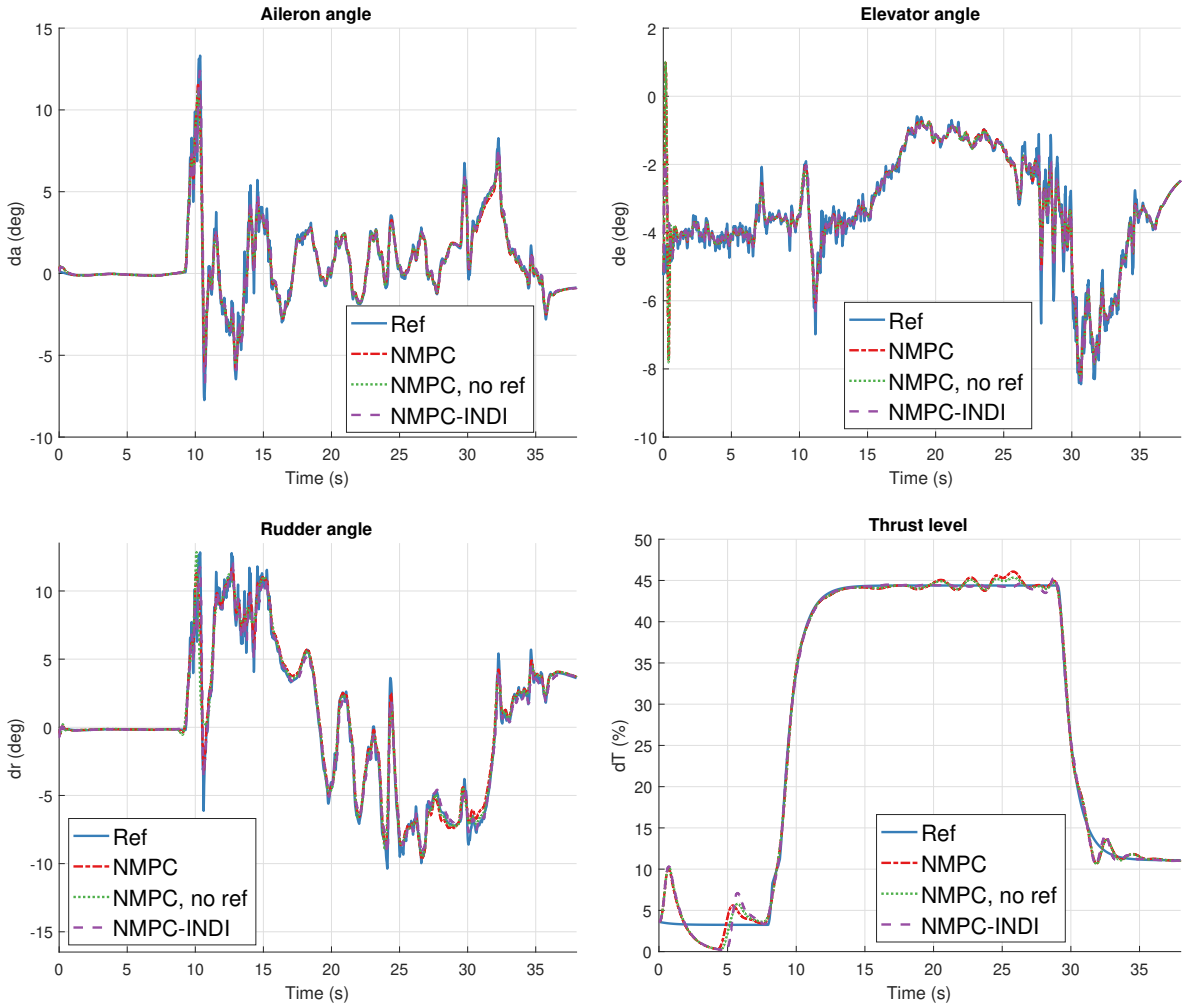


Figure G-6: Input reference tracking, track: Recovery.

Appendix H

Tracking results of track: Combined maneuvers

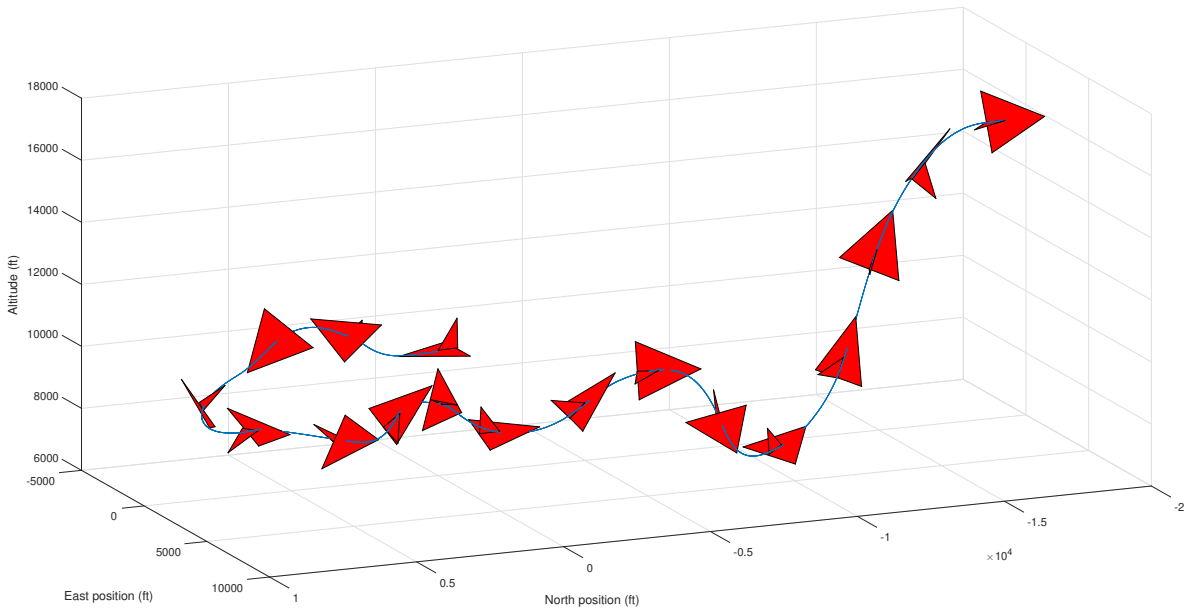


Figure H-1: Reference trajectory - Combined maneuvers.

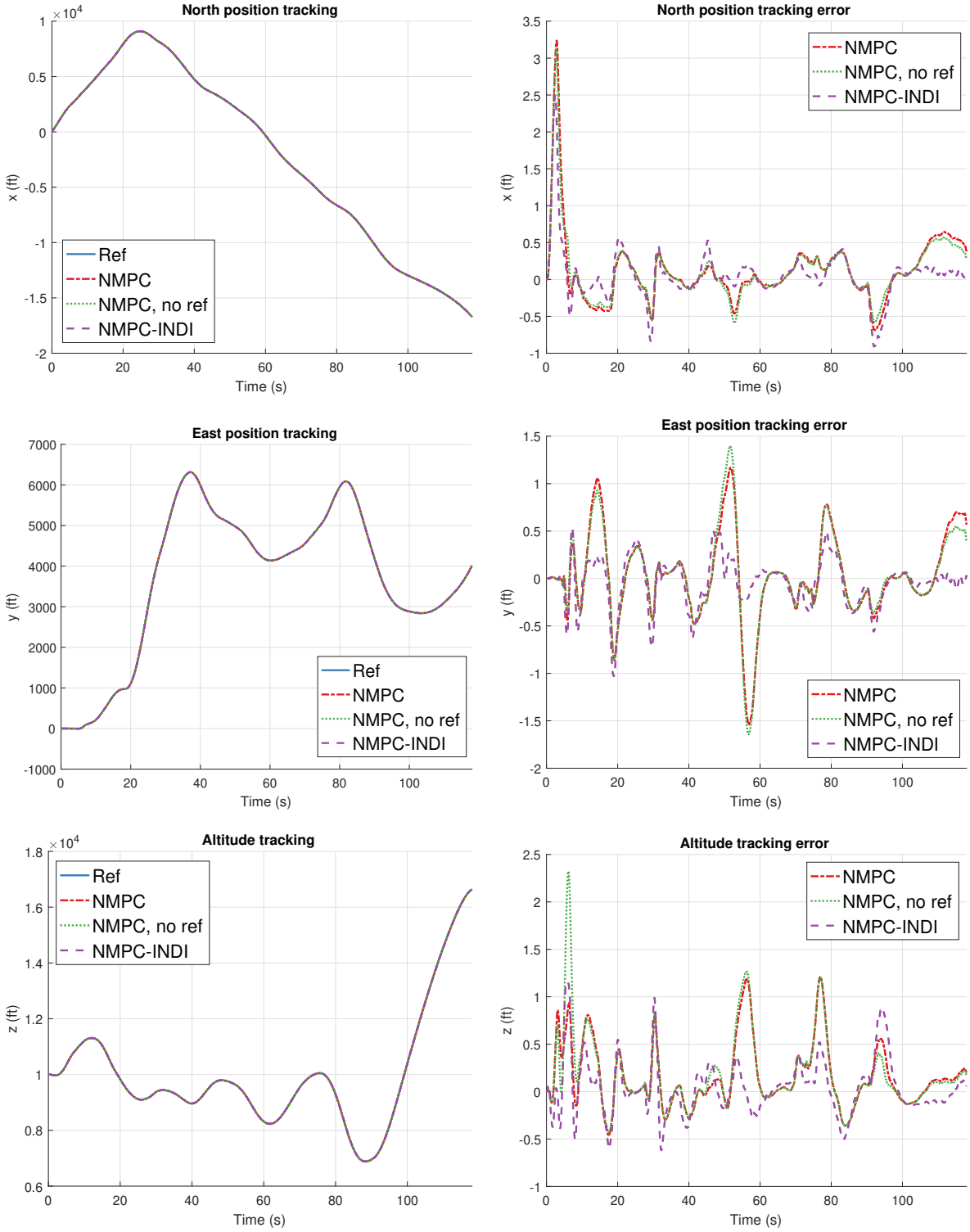


Figure H-2: Position tracking, track: Combined maneuvers.

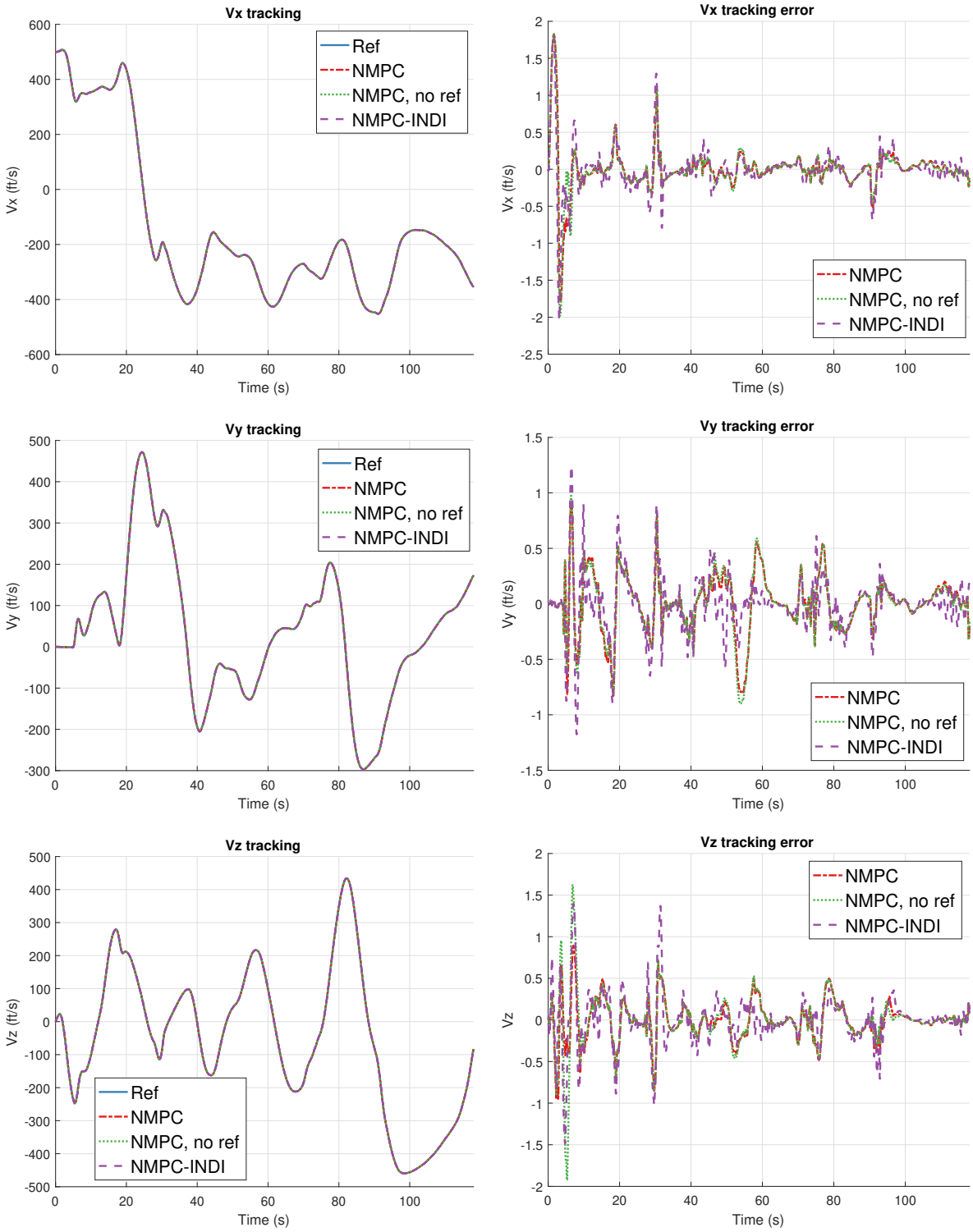


Figure H-3: Velocity tracking, track: Combined maneuvers.

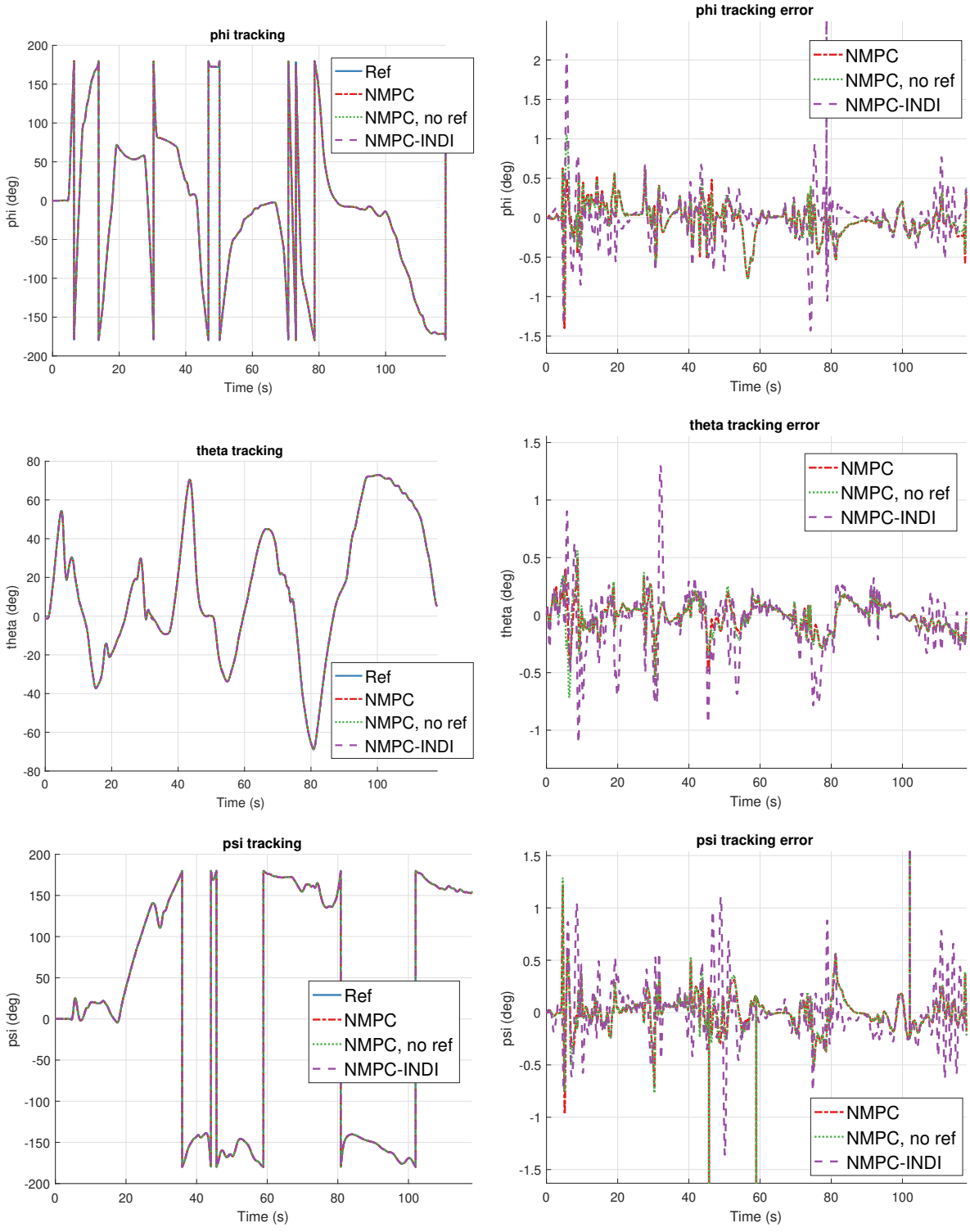


Figure H-4: Attitude angle tracking, track: Combined maneuvers.

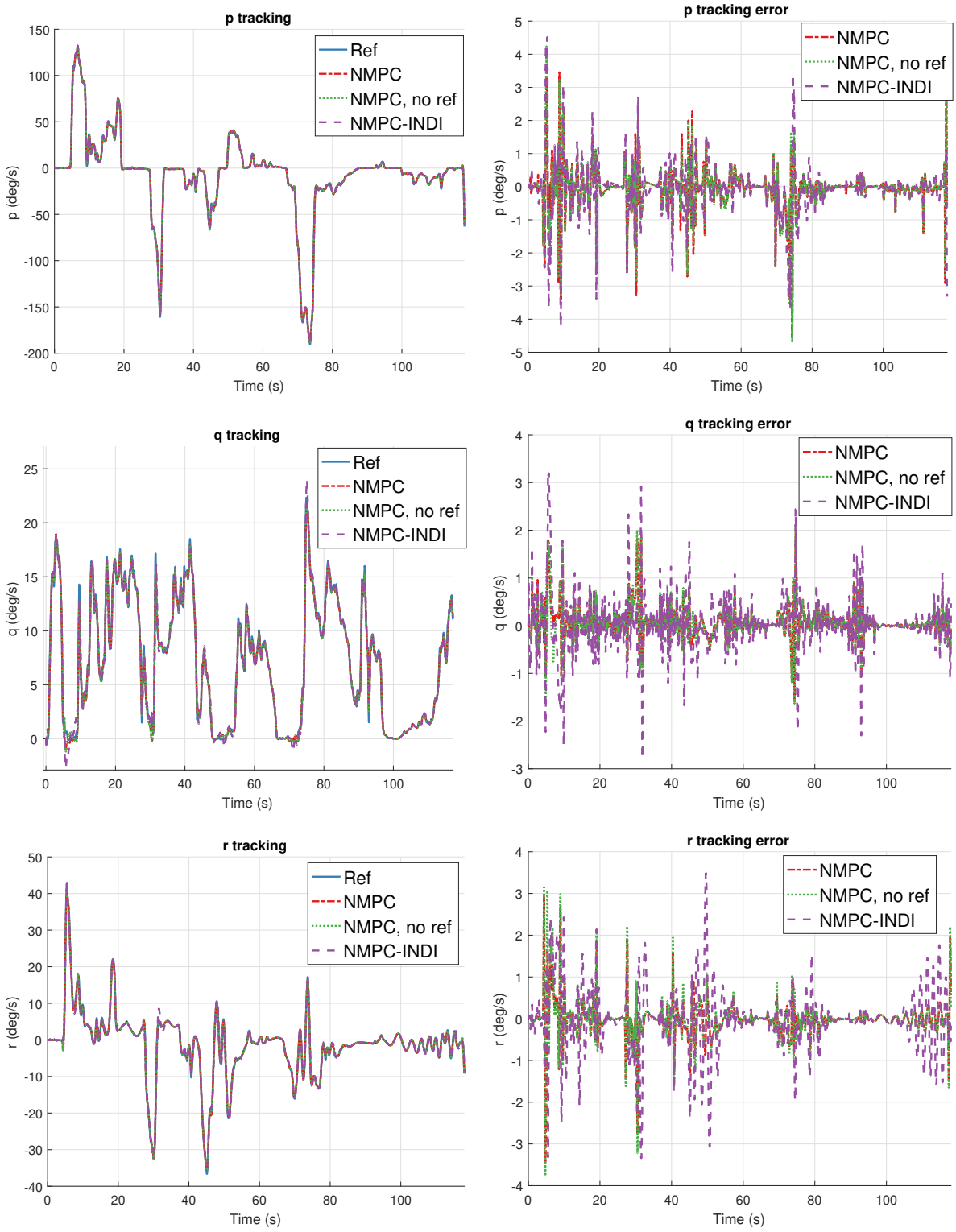


Figure H-5: Rotational rate tracking, track: Combined maneuvers.

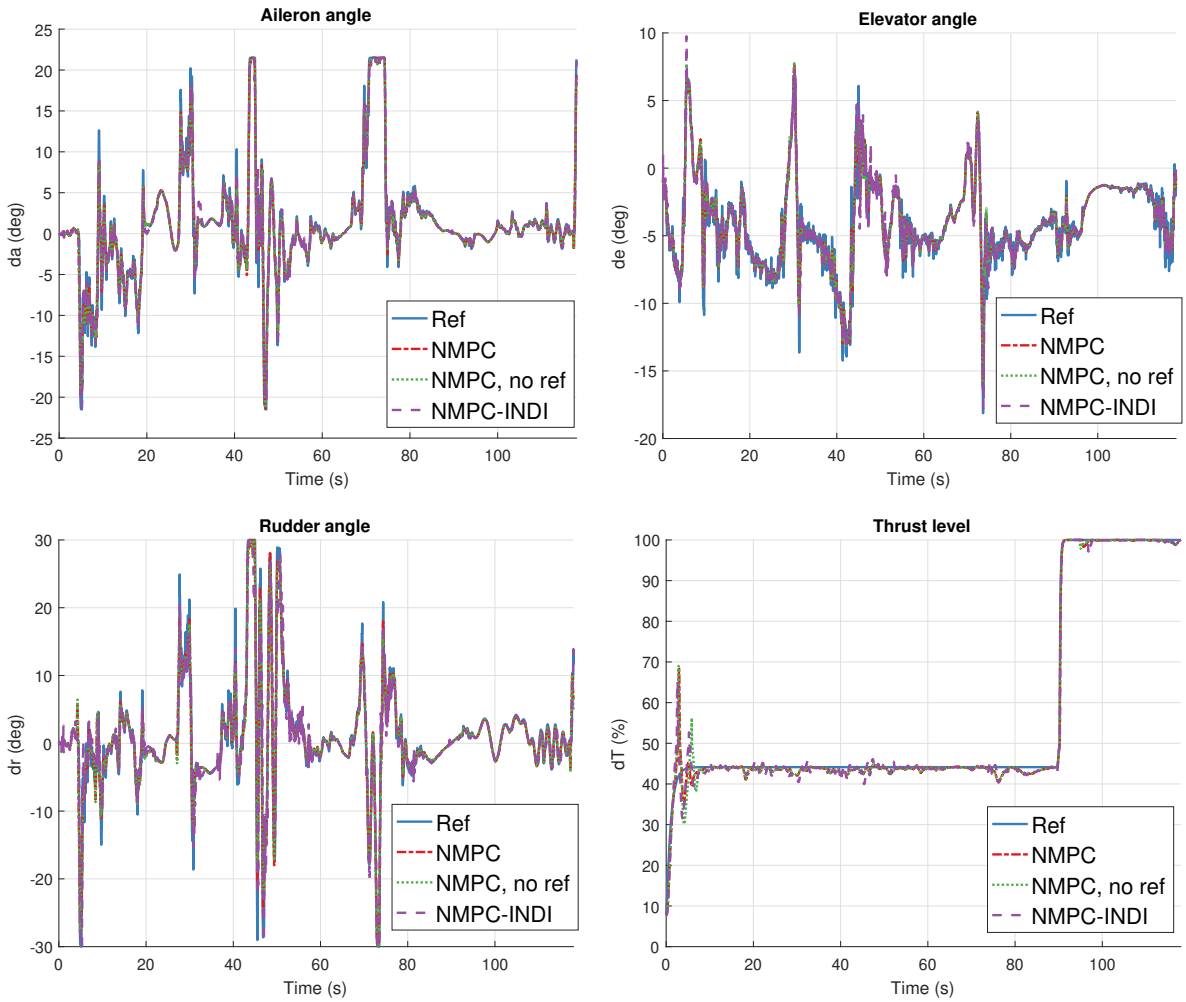


Figure H-6: Input reference tracking, track: Combined maneuvers.

Bibliography

[1] Netherlands Aerospace Centre, Anthony Fokkerweg 2, Amsterdam, The Netherlands, *Fighter Aircraft Robust Power Management*, 1 ed., 2017.

[2] L. T. Nguyen, M. E. Ogburn, W. P. Gilbert, K. S. Kibler, P. W. Brown, and P. L. Deal, "Simulator study of stall/poststall characteristics of a fighter airplane with relaxed longitudinal static stability," tech. rep., NASA, 1979.

[3] R. S. Russell, "Non-linear F16 Simulation using Simulink and Matlab," tech. rep., University of Minnesota, 2003.

[4] S. A. Snell, W. L. Garrard, and D. F. Enns, "Nonlinear inversion flight control for a supermaneuverable aircraft," in *Guidance, Navigation and Control Conference*, American Institute of Aeronautics and Astronautics, Aug. 1990.

[5] D. Boukraa, Y. Bestaoui, and N. Azouz, "3D trajectory tracking for a fixed wing unmanned aerial vehicle using dynamic inversion," *IFAC Proceedings Volumes*, vol. 40, no. 7, pp. 443–449, 2007.

[6] L. Sonneveldt, Q. Chu, and J. Mulder, "Adaptive backstepping flight control for modern fighter aircraft," in *Advances in Flight Control Systems*, InTech, apr 2011.

[7] S. Sieberling, Q. P. Chu, and J. A. Mulder, "Robust flight control using incremental nonlinear dynamic inversion and angular acceleration prediction," *Journal of Guidance, Control, and Dynamics*, vol. 33, pp. 1732–1742, nov 2010.

[8] M. R. Mortazavi and A. Naghash, "Pitch and flight path controller design for F-16 aircraft using combination of LQR and EA techniques," *Journal of Aerospace Engineering*, p. 13, apr 2017.

[9] E. N. Hartley, J. L. Jerez, A. Suardi, J. M. Maciejowski, E. C. Kerrigan, and G. A. Constantinides, "Predictive control using an FPGA with application to aircraft control," *IEEE Transactions on Control Systems Technology*, vol. 22, pp. 1006–1017, may 2014.

[10] T. Keviczky and G. Balas, "Software enabled flight control using receding horizon techniques," in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, American Institute of Aeronautics and Astronautics, aug 2003.

[11] R. Bhattacharya, G. J. Balas, M. A. Kaya, and A. Packard, "Nonlinear receding horizon control of an F-16 aircraft," *Journal of Guidance, Control, and Dynamics*, vol. 25, pp. 924–931, sep 2002.

[12] S. Gros, R. Quirynen, and M. Diehl, "Aircraft control based on fast non-linear MPC & multiple-shooting," in *51st IEEE Conference on Decision and Control (CDC)*, IEEE, dec 2012.

[13] D. Simon, *Model Predictive Control in Flight Control Design Stability and Reference Tracking*. PhD thesis, Linkoping University, 2014.

[14] D. Joosten, *Constrained and Reconfigurable Flight Control*. PhD thesis, Delft University of Technology, 2017.

[15] N. Birla and A. Swarup, "Optimal preview control: A review," *Optimal Control Applications and Methods*, vol. 36, pp. 241–268, jan 2014.

[16] Q. Wang, Y. Zhang, C. Dong, and M. Ni, "Robust trajectory tracking of unstable aircraft with measurement delay," *Journal of Systems and Control Engineering*, vol. 226, pp. 1220–1230, aug 2012.

[17] K. Natesan, D.-W. Gu, I. Postlethwaite, and J. Chen, "Design of flight controllers based on simplified LPV model of a UAV," in *Proceedings of the 45th IEEE Conference on Decision and Control*, IEEE, 2006.

[18] Z. Li, W. Zhou, and H. Liu, "Nonlinear robust control of hypersonic aircrafts with interactions between flight dynamics and propulsion systems," *ISA Transactions*, vol. 64, pp. 1–11, sep 2016.

[19] P. Acquatella, W. Falkena, E.-J. van Kampen, and Q. P. Chu, "Robust nonlinear spacecraft attitude control using incremental nonlinear dynamic inversion.," in *AIAA Guidance, Navigation, and Control Conference*, American Institute of Aeronautics and Astronautics, aug 2012.

[20] R. van't Veld, "Nonlinear dynamic inversion control for generic simulation models," tech. rep., Netherlands Aerospace Centre, 2015.

[21] H. Schaub and J. L. Junkins, *Analytical Mechanics of Space Systems*. AIAA Education Series, 2003.

[22] D. Q. Huynh, "Metrics for 3D rotations: Comparison and analysis," *Journal of Mathematical Imaging and Vision*, vol. 35, pp. 155–164, jun 2009.

[23] U. Eren, A. Prach, B. B. Koçer, S. V. Raković, E. Kayacan, and B. Açıkmeşe, "Model predictive control in aerospace systems: Current state and opportunities," *Journal of Guidance, Control, and Dynamics*, vol. 40, pp. 1541–1566, jul 2017.

[24] R. Cagienard, P. Grieder, E. Kerrigan, and M. Morari, "Move blocking strategies in receding horizon control," *Journal of Process Control*, vol. 17, pp. 563–570, jul 2007.

[25] D. Limon and T. Alamo, "Tracking model predictive control," in *Encyclopedia of Systems and Control*, pp. 1475–1483, Springer London, 2013.

Glossary

List of Acronyms

FARPM	Fighter Aircraft Robust Power Management
NLR	Netherlands Aerospace Centre
FDSIM	Flight Dynamics Simulator
F-16	F-16 Fighting Falcon
LQR	Linear Quadratic Regulator
MPC	Model Predictive Control
NMPC	Nonlinear Model Predictive Control
NDI	Nonlinear Dynamic Inversion
INDI	Incremental Nonlinear Dynamic Inversion
FBL	Feedback Linearization
LPV	Linear Parameter Varying
RMS	Root Mean Square

List of Symbols

α	Angle of attack
$\bar{c}$	Mean chord
$\bar{q}$	Free-stream dynamic pressure
β	Angle of sideslip
δ_a	Aileron angle

δ_e	Elevator angle
δ_r	Rudder angle
δ_T	Thrust level
δ_{lef}	Leading edge flap deflection angle
$\hat{\mathbf{d}}$	Observer state vector
ν	Virtual input
ω_n	Natural frequency
ϕ	Roll angle
ψ	Yaw angle
ρ	Air density
σ	Standard deviation
$\mathbf{A}$	State matrix
$\mathbf{A}_{\text{cons}}$	Constraint matrix
$\mathbf{C}$	Output matrix
$\mathbf{I}$	Inertia matrix
$\mathbf{L}$	Observer gain matrix
$\mathbf{Q}$	Weight matrix for states
$\mathbf{q}$	Quaternion vector
$\mathbf{R}$	Weight matrix for input increments
$\mathbf{T}$	Move blocking matrix for assigning input increments to corresponding inputs
$\mathbf{T}_{bE}$	Transformation matrix from Vehicle-carried normal Earth reference frame to body-fixed reference frame
$\mathbf{u}$	Input vector
$\mathbf{x}$	State vector
$\mathbf{y}$	Output vector
θ	Pitch angle
ζ	Damping ratio
a_{nx}	Normalized acceleration in x direction
a_{ny}	Normalized acceleration in y direction
a_{nz}	Normalized acceleration in z direction
b	Aircraft span
C_l	Aerodynamic roll moment coefficient
C_m	Aerodynamic pitch moment coefficient
C_n	Aerodynamic yaw moment coefficient
C_X	Aerodynamic body force coefficient in the x-direction in the body-fixed reference frame
C_Y	Aerodynamic body force coefficient in the x-direction in the body-fixed reference frame
C_Z	Aerodynamic body force coefficient in the x-direction in the body-fixed reference frame
g	Gravitational acceleration

h	Sampling time
I_i	Inertia component
J	Cost function
K	Gain
L	Roll moment
M	Mach number
M	Pitch moment
m	Aircraft mass
N	Yaw moment
N_c	Control horizon length
N_f	Number of fast control signal updates
N_h	Length of the slow control signal updates
N_p	Prediction horizon length
N_s	Number of slow control signal updates
p	Roll rate
P_1	Engine power command based on throttle position
P_2	Engine power command to the engine
P_3	Engine power level
P_s	Static pressure
q	Pitch rate
q_i	Quaternion parameter
r	Yaw rate
S	Wing area
s	Complex variable in Laplace transform
u	Velocity component in the x-direction in body-fixed reference frame
v	Velocity component in the y-direction in body-fixed reference frame
V_t	Total velocity
V_x	North velocity
V_y	East velocity
V_z	Climb rate
w	Velocity component in the z-direction in body-fixed reference frame
X	Body force in the x-direction in body-fixed reference frame
x	North position
Y	Body force in the y-direction in body-fixed reference frame
y	East position
Z	Body force in the z-direction in body-fixed reference frame
z	Altitude