VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

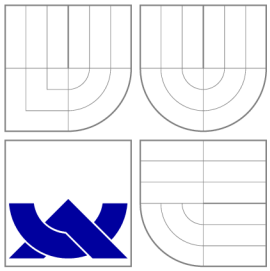
AUTOTAXI SYSTEM DESIGN FOR AIRCRAFT

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

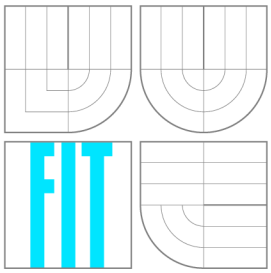
AUTOR PRÁCE
AUTHOR

JURAJ KARDOŠ

BRNO 2015



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

NÁVRH SYSTÉMU AUTOTAXI PRO LETOUN

AUTOTAXI SYSTEM DESIGN FOR AIRCRAFT

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE
AUTHOR

JURAJ KARDOŠ

VEDOUCÍ PRÁCE
SUPERVISOR

Ing. PETER CHUDÝ, Ph.D. MBA

BRNO 2015

Abstrakt

Nedávné studie předpovídají nárůst pasažérů využívajících leteckou dopravu. Tento trend bude vyžadovat zavedení nových leteckých linek, důsledkem čeho bude zhuštěn letový provoz s dopadem hlavně na nápor letišť v metropolitních oblastech. Automatizované řízení pojíždení letounu umožní menší rozestupy mezi jednotlivými linkami a zvýšení příletové a odletové kapacity letišť. Tato práce se zabývá návrhem modelu pohybu dopravního letounu po zemi s ohledem na různé provozní podmínky jako např.: stav povrchu vzletové a přistávací dráhy za různého počasí a lišící se provozní parametry letounu (tlak v pneumatikách, zatížení podvozků a pod.). Validace modelu byla založena na sledování poloměru zatáčky pro různe uhly natočení přední podvozkové nohy. Výsledky simulace byly validovany vzhledem k analytickému modelu Ackermanovy geometrie a na specifikační dokument od Boeingu určený pro plánovaní pohybu letounu na letišti [6]. Výsledky prokázaly přesnost modelu a potvrdily jeho možné nasazení pro simulace v reálnem čase.

Abstract

Recent studies focused on the global airline industry predict a continuous growth of passenger numbers, which will stimulate an increased demand for modern sophisticated aircraft capable of precise operations at reduced separation minima. Automation systems, such as AutoTaxi, will allow for decreased ground separation standards and a subsequent increase of throughput at airports in metropolitan areas. This thesis deals with an AutoTaxi control system for a single-aisle passenger aircraft, such as Boeing 737 series, under different operational conditions. The implemented model considers varying runway characteristics due to the atmospheric conditions and different aircraft configurations. Detailed force and momentum equilibria analysis are presented in a form of equations of motion, which is essential in order to achieve high-precision simulation. The validation of the model was based on the turn radii comparison for multiple steering angles. Simulation results were subjected to a comparison with the analytical solution of the Ackerman drive for a tricycle vehicle and with turn radii specified in Airplane Characteristics for Airport Planning [6] issued by Boeing. Obtained results confirm high-precision real-time simulation.

Klíčová slova

Dynamika letounu na zemi, model pohybu po zemi, pohybové rovnice, ODR, AutoTaxi, civilní letectví, model letiště.

Keywords

Aircraft Ground Dynamics, Ground Motion Model, Equations of Motion, ODE, AutoTaxi, Civil Avionics, Airport Model.

Citace

Juraj Kardoš: AutoTaxi System Design for Aircraft, diplomová práce, Brno, FIT VUT v Brně, 2015

AutoTaxi System Design for Aircraft

Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně pod vedením Ing. Petra Chudého, Ph.D. Další informace mi poskytl Ing. Petr Dittrich. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Juraj Kardoš
May 19, 2015

© Juraj Kardoš, 2015.
Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů.

Contents

1 Introduction **3**
- 1.1 State of the Art . . . 3
- 1.2 Thesis Overview . . . 5

2 Aircraft Ground Operations Simulation **6**
- 2.1 Reference Frames . . . 6
  - 2.1.1 Geometry of the Earth . . . 6
  - 2.1.2 ECEF Frame (E) . . . 8
  - 2.1.3 WGS 84 System . . . 8
  - 2.1.4 Navigational Frame (N) . . . 8
  - 2.1.5 NED Frame (O) . . . 8
  - 2.1.6 Body Fixed Frame (B) . . . 9
  - 2.1.7 Transformation between NED Frame (O) and ECEF (E) . . . 10
  - 2.1.8 Transformation between BFF (B) and NED Frame (O) . . . 10
- 2.2 Ground Motion Model . . . 12
  - 2.2.1 General Assumptions . . . 12
  - 2.2.2 Equations of Motion . . . 13
  - 2.2.3 Load Balancing . . . 15
  - 2.2.4 Tire-Ground Interaction Model . . . 16
  - 2.2.5 Engine Model . . . 22

3 Design and Implementation of Aircraft Taxi Simulation Model **24**
- 3.1 Numerical Methods . . . 25
  - 3.1.1 Euler Method . . . 26
  - 3.1.2 Runge-Kutta Methods . . . 26
  - 3.1.3 Multistep Adams' Methods . . . 27
  - 3.1.4 Variable Step Solvers . . . 28
- 3.2 Simulink Modeling Tool . . . 29
  - 3.2.1 Fixed Step Solvers . . . 30
  - 3.2.2 Variable Step Solvers . . . 30
- 3.3 Ground Motion Simulation Model . . . 31
  - 3.3.1 Equations of Motion . . . 32
  - 3.3.2 Tire Forces Model . . . 33
  - 3.3.3 Propulsion Model . . . 37
  - 3.3.4 Environment Model . . . 38

4 AutoTaxi Controller Design **39**
4.1 System Definition 39
4.2 Control Theory 40
4.2.1 PID Controller 40
4.3 Controller Implementation 42
4.3.1 Directional Control 42
4.3.2 Velocity Control 47
4.3.3 Brakes Control 48

5 Experiments and Model Verification **52**
5.1 Analytical Verification of the Turn Radius 52
5.2 Boeing's Specification of Turn Radius 53
5.3 AutoTaxi Experiments 55
5.3.1 Dry Runway Conditions 55
5.3.2 Wet Runway Conditions 56
5.3.3 Snow/Ice Contaminated Runway 56

6 Conclusion **60**

A Table of Symbols **65**

1. Introduction

Fascination by flying has long lasting roots in the history of the human-kind. The origin of mankind's desire to fly is lost in the distant past. We only know from the earliest legends there have been men strapping birdlike wings or other devices to themselves and attempting to fly. The kite may have been the first form of man-made aircraft [12]. It was invented in China possibly as far back as the 5th century BC and is thought to be first man-made flight. It was used for navigation tasks and thus enabling people to discover new horizons. As the improvements of technology continued through the centuries, we were able to rise from the Earth's surface and become dominant species not only on the ground but also in the air.

The first human walk on the Moon was a major demonstration of the possibilities that mastering of the flight can bring to us. If we consider that the resources of our planet are very limited and the humans' demand in order to survive is exponentially growing it is true that no civilization can last long in this Universe if it stays confined to a single planet. And that's not the only problem we need to face. Our mighty star, the ball of fusing hydrogen that anchors Earth and powers all of its life, will one day grow so large that its outer atmosphere will singe and sterilise our planet, and maybe even engulf it [3]. In this sense, the question if we ever reach the distant stars and colonise new planets, has been already answered. If we want to survive as a species we have no other choice.

We will need to find the courage and spirit that moved our own species to step into new continents, so that our recent ancestors could reach out to islands and archipelagos, before crossing whole oceans, on their way to the distant corners of this Earth. We will need to set out for new planets and eventually, new stars. In order to reach these goals we need to master elementary tasks first.

1.1 State of the Art

From the human nature we are highly unreliable and error-prone controllers. Human ability to control various flight tasks depends on multitude of factors and only the slightest misinterpretation of the situation by the pilot may result into a catastrophic scenario. Over 80% of pilot-caused runway incursions occur during taxi to the departure runway [15]. The automation of the flight and the taxi phase is therefore needed.

Automation tasks become even more pertinent when one looks at the *General Market Forecast* for aircraft published periodically by Airbus. It shows that the passenger numbers will double within next 15 years, with a consequent increased demand for new airframes [32]. NASA has also published a similar document in the form of the *National Plan for Aeronautics Research and Development and Related Infrastructure* [23]. Both reports highlight similar challenges and identify the automation of aircraft movements, on the ground and in the air, as means of meeting the objectives such as the quality and affordability of

aircraft, the effect on the environment, safety, security and the efficiency of the air transport system. Reduced aircraft separation due to increased demand will require a transition to the trajectory-based operations, novel approaches in navigation, and a paradigm shift in control with new allocation of responsibilities between humans and automation [9].

The need for maximising arrivals and departures capacities at airports in metropolitan areas will become critical. Procedures designed around now-antiquated technology lead to inefficient use of terminal area airspace. Airlines and airport ramp operators typically only track gate occupancy based on predicted gate-in and out times reflected by the schedule and updates to flight times. The decision of which aircraft is entering or exiting the ramp area, and at what time, is done manually and oftentimes with limited information that inadvertently leads to delays. The efficiency of technologies to reduce separations and improve flight paths for high-density arrival and departure traffic flows, will be highly dependent on automation and precision positioning, navigation, and timing. Automation will allow for decreased separation standards, taxi times reduction and subsequent increased throughput for single and multiple runways [23].

Aircraft manufacturers are also constantly striving to improve the efficiency of all aspects surrounding the operation of their aircraft. Fuel savings can be made by decreasing the drag of the aircraft during the cruise phase. However, less obvious savings can be made by improving the way aircraft are operated on the ground. Using aircraft's main engines during taxiing phase results in a huge consumption of fuel. It was forecasted to cost around 5.5bn euros by 2012, produce huge amounts of CO_2 (approximately 18 million tons per year) and is significant source of foreign object damage (costing around 280 million Euro annually) [31].

New operational procedures are directed towards the reduction of noise and developing environmentally friendly systems. Taxing thus impacts both domains, in terms of noise and air pollution. Projects such as CleanSky [10] and NextGen [2] have set their challenges to protect the environment and energy supply by improving air operations and traffic management. Their goal is the optimisation of each flight phase of the entire flight at a unitary aircraft level. NextGen proposes to transform air traffic control system from a ground-based system to a satellite-based system. GPS technology will be used to shorten routes, save time and fuel, reduce traffic delays, increase capacity, and permit controllers to monitor and manage aircraft with greater safety margins. Planes will be able to fly closer together, take more direct routes and avoid delays caused by airport stacking as planes wait for an open runway [2, 10]. CleanSky focuses its effort into development of breakthrough technologies integrated to significantly increase the environmental performances of airplanes and air transport, resulting in quieter and more fuel efficient aircraft.

An alternative approach to aircraft taxiing phase has been proposed by the research project TaxiBot [31]. This concept aims to remove the need to start the aircraft's main engines during taxi phase. Aircraft are moved on the ground by a partially automated tug while pilots are still in control. Taxibot lifts the aircraft's nose-wheel, which then rests on a rotating platform translating nose-gear deflections into directional changes for the tug. Thus, pilots steer the tractor via the nose-gear tiller in the cockpit. The aircraft wheel brakes are used for deceleration. Forward speed is also controlled by braking. The tractor starts to move as the flight crew releases the brakes. The maximum speed is digitally limited depending on the aircraft's position at the airport. Service entry of the TaxiBot is scheduled for 2016 as it is currently going through the test phase.

1.2 Thesis Overview

This thesis covers the task of aircraft ground motion simulation. Chapter 2 provides definitions of the reference frames used for description of the aircraft position, orientation and motion during the flight and ground maneuvers. The frames used to define force and moment components acting on the aircraft body are also provided. It is often required to use different frames for the description of different aspects of the simulation, therefore we also discuss the transformations between the reference frames.

Chapter 2 presents the ground motion model of a single aisle passenger aircraft, similar to a Boeing 737-400. The described model contains individual modules - propulsion characteristics, tire forces, brakes efficiency, aerodynamics and more. The partial models are then assembled together using the Newtonian physics, building the equations of motion. Implemented equations of motion provide a tool to model the aircraft ground motion.

Chapter 3 describes the implementation of the model in the Matlab/Simulink® environment. This chapter covers the aspects of the numeric simulation and provides the description of the designed simulation model. Simulink® supports block-modular system development, therefore this thesis provides description of the blocks corresponding to the individual equations of motion and its composition into blocks with higher level of abstraction. On the most abstract level, the simulation model is a single block with its defined interface.

In chapter 4, a single-box abstraction of ground motion model is introduced. On top of this model, controller module is developed. It is designed to be able to automatically steer the aircraft along a pre-defined trajectory. This thesis provides the basis of control theory and presents the design of individual controller modules, namely steering control, propulsion and braking control.

The last chapter 5 contains the description of a simulation model verification and experiments to verify the ability of the controller to automatically steer the aircraft along the predefined trajectory. The simulation model verification is based on an analytical model of a Ackerman tri-cycle drive and the specification of Boeing turn radii for given aircraft model. The auto-taxiing trials are simulated at major airports at Czech Republic - Praha, Brno and Ostrava.

2. Aircraft Ground Operations Simulation

In order to describe the nature of aircraft's ground maneuvers and to deploy its automation, mathematical model needs to be developed. The model will predict the behavior of the aircraft based on the input forces resulting from the engine thrust, landing gear brake forces or wheels and runway characteristics. The model is therefore derived via the force and momentum equilibrium considering the Newtonian physical model of the classical mechanics. A general description of the runway surface conditions influencing the tire-ground friction performance is also provided in this chapter, as well as the forces resulting from the interaction of the aircraft wheel and the runway surface. Likewise the model for engine forces, braking and aerodynamics is introduced. The forces are composed into the equations of motion, describing the behavior of the aircraft's ground motion.

2.1 Reference Frames

To describe the position and the behavior of an aircraft, a reference frame in contex of which one defines the positional and behavioral aspects (force and momentum components) is needed. On-board sensors measure force components with respect to an inertial frame which is resolved in the host body frame. These are the forces and moments acting on the aircraft during the ground motion, expressed using Body Fixed Frame (B) notation. However, to describe the trajectory of the ground motion, variables such as velocity need to be transformed to an appropriate Earth-fixed frame, e.g. coordinate system used by the GPS standard - WGS84. This allows us to successfuly manage the navigation tasks, such as specifying location of a moving aircraft with respect to the Earth's surface. Most of the widely-used coordinate frames relevant to the definition of acting forces on an aircraft and navigation purposes are discussed in this chapter, likewise their mutual transformations.

2.1.1 Geometry of the Earth

First, the abstraction of the Earth's shape is explained as it is used in the discussed navigation frames. The Earth's surface is extremely irregular with a very complex shape, it is therefore approximated by an ellipsoid for the computational convenience. The ellipsoid and various surfaces that are useful for understanding the geometry of the Earth's shape are shown in Figure 2.1.

Terrain is defined as an interface between the solid and fluid masses of the Earth and its atmosphere [24].

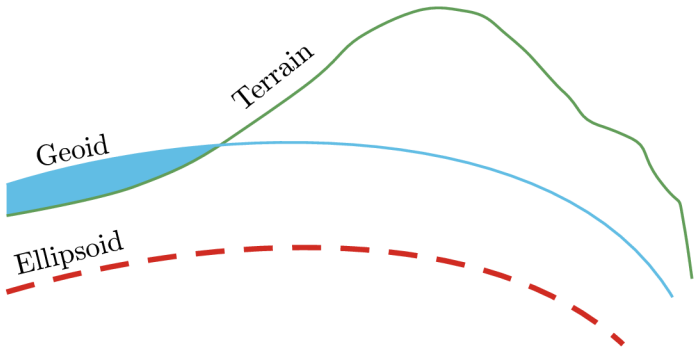


Figure 2.1: A depiction of various surfaces of the Earth

Geoid is an equipotential surface (surface of constant gravity) best fitting the average sea level in the least squares sense (ignoring tides and other dynamical effects in the oceans). It can be thought of as an idealized mean sea level extended over the land portion of the globe. The geoid is a smooth surface but its shape is irregular and it does not provide the simple analytic expression needed for navigational computations [24].

Reference Ellipsoid is a mathematically defined surface approximating the geoid. Ellipsoid is made by rotating an ellipse about its minor axis, which is coincident with the mean rotational axis of the Earth. The center of the ellipsoid is coincident with the Earth's center of mass. The ellipsoid is the most analytically convenient surface to work with for the navigational purposes. Its shape is defined by two geometric parameters called the semimajor axis and the semiminor axis. These are typically represented by a and b respectively (see Figure 2.2). The geoid height N is the distance along the ellipsoidal normal from the surface of the ellipsoid to the geoid. The orthometric height H is the distance from the geoid to the point of interest P. The geodetic height (also known as altitude) is the sum of the geoid and the orthometric heights $h = N + H$ [24].

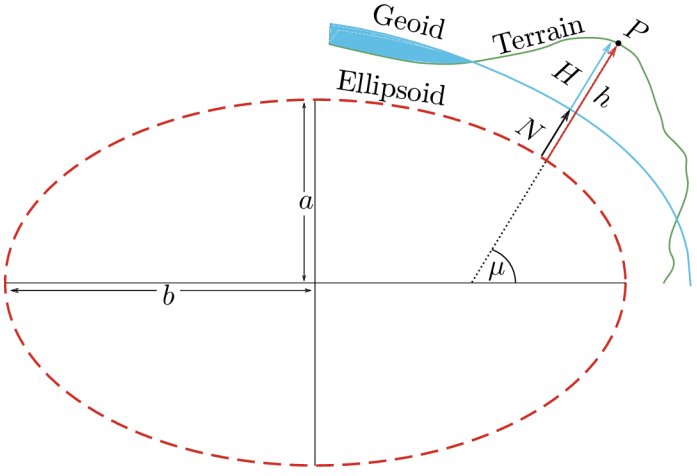


Figure 2.2: The relationship between various Earth surfaces (highly exaggerated).

2.1.2 ECEF Frame (E)

The ECEF frame (Earth Centered Earth Fixed, Figure 2.3) is used to specify position of the aircraft (or any point on Earth in general) for navigational purposes. The coordinates are given with respect to the reference point, origin of the frame, which is fixed to the mass center of the Earth. x_E axis lies in equatorial plane at intersection with Greenwich Meridian. z_E axis is aligned with Earth's rotational axis and positive direction is oriented to Earth's North Pole. Axis y_E is oriented so that it creates a right-hand system. As a consequence of such definition, ECEF frame rotates and moves together with Earth so that the axes are oriented as defined above at all moments and the coordinates of a point fixed on the surface of the Earth do not change.

2.1.3 WGS 84 System

WGS 84 is an alternative definition of ECEF. Axes are defined with the same origin and orientation, however the position coordinates are specified in a different way. While in ECEF the position of any given point is denoted by the cartesian coordinates $[x, y, z]$, WGS 84 gives this information in a form of two angles, longitude λ and latitude μ and the height above the ellipsoid h (Figure 2.5).

Geodetic longitude λ is the angle measured in the equatorial plane between zero meridian plane (xz-plane of ECEF system) and the meridian plane of the point. Range of longitude is defined as: $-\pi \leq \lambda \leq \pi$.

Geodetic latitude μ is the angle measured in the meridian plane of the point P between the equatorial plane and (xy-plane of ECEF) and the surface normal of the point P. Surface is not an ideal sphere so the normal does not necessarily have to pass through the Earth's center (as shown on Figure 2.5). Latitude range is $-\pi/2 \leq \mu \leq \pi/2$.

Geodetic height h is the height above the WGS-84 ellipsoid measured along the surface normal. The GPS system uses this geodetic height as a reference altitude information. However, for practical purposes ellipsoid approximation of the Earth surface is too coarse-grained and needs to be refined using local data of the Geoid height stored in the lookup tables. The refined GPS height is the approximation of the height above the mean sea level at a given location [33].

2.1.4 Navigational Frame (N)

In order to describe the position of the aircraft as it moves over the ground surface, it is necessary to do so with reference to a point with a fixed position and orientation in space. Navigational reference frame (Figure 2.3) is defined with respect to a given fixed point on Earth's surface, specified either in ECEF or WGS 84 system. As the reference point is fixed with respect to the Earth as well as the reference axes are, the navigational frame moves and rotates together with the Earth, however it remains stationary with respect to any given point on its surface. Axis x_N is parallel to the local geoid surface and is oriented to North. Axis z_N points downwards and is perpendicular to local geoid surface (does not necessary pass through the Earth's center as can be seen in Figure 2.5).

2.1.5 NED Frame (O)

NED Frame (North-East-Down System) purpose is to specify the attitude of the plane by the means of the rotation angles of BFF about the NED frame, which does not rotate

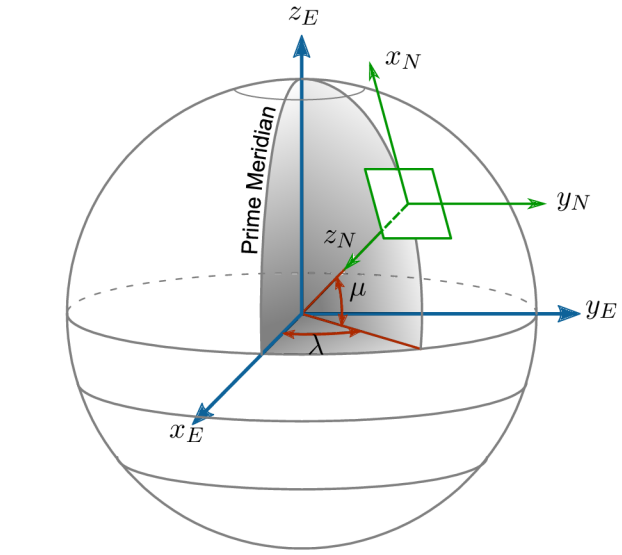


Figure 2.3: Navigational frame (denoted with N subscript) and its relation to Earth Centered Earth Fixed frame (indexed with E). Geodetic angles λ, μ are highlighted to illustrate alternative notation of WGS 84 system.

together with the aircraft. NED origin is a reference point of the aircraft, usually the center of gravity (C.G.). The system moves with the aircraft together with its reference point. The orientation of axes is the same as in the Navigational Frame; x_O is parallel to the local geoid surface pointing to the geographic north pole, y_O is parallel to the local geoid surface pointing east to form a right hand system with x-axis and z-axis. z_O is pointing downwards and is perpendicular to the local geoid surface.

2.1.6 Body Fixed Frame (B)

Body Fixed Frame (see Figure 2.4) is used to define forces and corresponding moments acting on the aircraft. The origin of BFF is defined as a reference point of the aircraft, usually at the C.G. BFF frame moves and rotates with the rigid body aircraft. Axis x_B points in the direction of aircraft's nose, y_B points to the right from pilot's view and z_b axis points downwards to form right-hand orthogonal system.

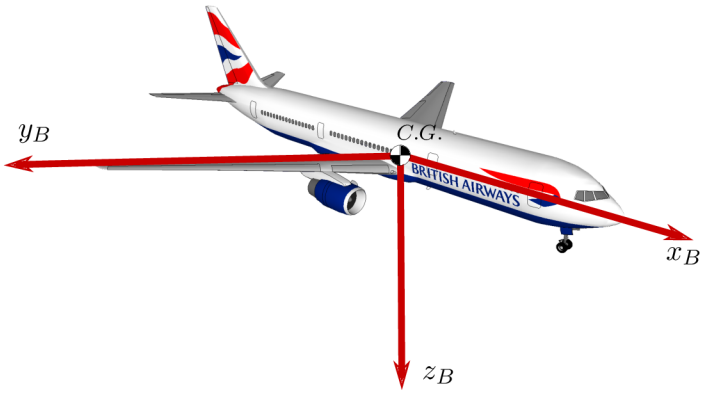


Figure 2.4: Body Fixed Frame.

2.1.7 Transformation between NED Frame (O) and ECEF (E)

To align the ECEF Frame with the NED, several transformations need to be performed. First, rotate the ECEF system around the z_E axis with the angle of the geodetic length λ. For illustration see Figure 2.5.

$$M_{OE_1}(\lambda) = \begin{bmatrix} \cos(\lambda) & \sin(\lambda) & 0 \\ -\sin(\lambda) & \cos(\lambda) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.1}$$

Second, rotate ECEF system around y_O with the negative angle of geodetic latitude $(-\mu - \pi/2)$.

$$M_{OE_2}(\mu) = \begin{bmatrix} \cos(-\mu - \pi/2) & 0 & \sin(-\mu - \pi/2) \\ 0 & 1 & 0 \\ -\sin(-\mu - \pi/2) & 0 & \cos(-\mu - \pi/2) \end{bmatrix} \tag{2.2}$$

The transformation matrix ECEF to NED has a form [7]:

$$M_{OE} = M_{OE_2}(\mu) M_{OE_1}(\lambda). \tag{2.3}$$

$$M_{OE} = \begin{bmatrix} -\sin(\mu)\cos(\lambda) & -\sin(\mu)\sin(\lambda) & \cos(\mu) \\ -\sin(\lambda) & \cos(\lambda) & 0 \\ -\cos(\mu)\cos(\lambda) & -\cos(\mu)\sin(\lambda) & -\sin(\mu) \end{bmatrix} \tag{2.4}$$

Reverse transformation matrix can be expressed as:

$$M_{EO} = M_{OE}^T. \tag{2.5}$$

2.1.8 Transformation between BFF (B) and NED Frame (O)

Apart from an aircraft's position that is described by the navigational frames, we are also interested in its orientation. It is described by heading and tilt angles. For this reason one needs to specify its position of the BFF axes with respect to NED. These angles are referred to as Euler angles, that are also used to describe the orientation of a reference frame relative to another frame.

Euler angles represent sequence of basic rotations about the axes of reference frame [17]. Any orientation of a rigid body can be expressed by series of rotations. The rotations can be performed about the axes of fixed coordinate system or about the axes of a rotating coordinate system, initially aligned with the fixed system (ECEF), and modifying its orientation with each rotation. The rotating reference frame can be represented as a rigidly attached to a rigid body, it is also referred to as a local system (e.g. BFF). Euler angles between NED and BFF are (also see Figure 2.6):

- **Pitch** angle Θ specifies the tilt of the aircraft's nose from the $x_O y_O$ plane.
- **Roll** angle Φ specifies rotation about the x_B axis of the BFF frame.
- **Yaw** angle Ψ describes the rotation about the vertical axis z_N of the NED system [17].

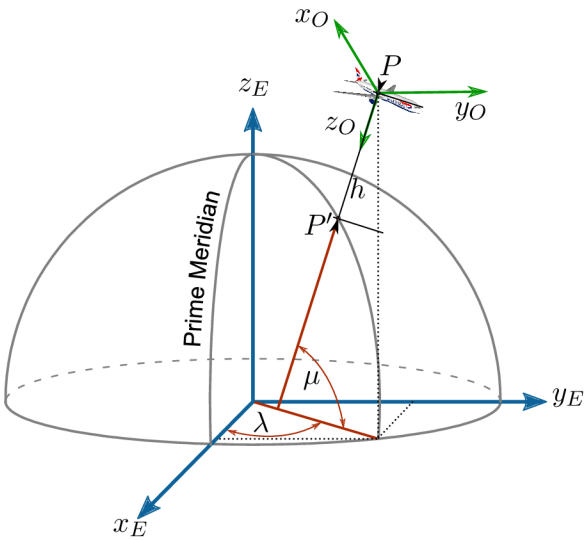


Figure 2.5: Relation between ECEF (E) and NED frame (O). The line z_O lies in the meridian plane of the point P, which is deflected from Prime meridian plane by an angle λ. Axis z_O is also normal to the local meridian, so as a consequence it does not have to pass through Earth's center. P' is the intersection of z_O axis with the ellipsoid, defining geodetic height $h = |PP'|$.

To transform NED (O) system to BFF (B), three rotations need to be performed in specific order [7]. First rotation compensates azimuth angle Ψ about rotation axis z_O:

$$M_{BO_1} = \begin{bmatrix} \cos(\Psi) & \sin(\Psi) & 0 \\ -\sin(\Psi) & \cos(\Psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.6}$$

Second, we compensate pitch angle Θ about auxiliary axis k_2

$$M_{BO_2} = \begin{bmatrix} \cos(\Theta) & 0 & -\sin(\Theta) \\ 0 & 1 & 0 \\ \sin(\Theta) & 0 & \cos(\Theta) \end{bmatrix} \tag{2.7}$$

Finally, roll (bank) angle Φ is compensated by rotation about axis x_B:

$$M_{BO_3} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\Phi) & \sin(\Phi) \\ 0 & -\sin(\Phi) & \cos(\Phi) \end{bmatrix} \tag{2.8}$$

Whole transformation of NED to BFF is expressed by series of the three rotations:

$$M_{BO} = M_{BO_3} M_{BO_2} M_{BO_1}. \tag{2.9}$$

$$M_{BO} = \begin{bmatrix} \cos(\Psi)\cos(\Theta) & \sin(\Psi)\cos(\Theta) & -\sin(\Theta) \\ \cos(\Psi)\sin(\Theta)\sin(\Phi) - \sin(\Psi)\cos(\Phi) & \sin(\Psi)\sin(\Theta)\sin(\Phi) + \cos(\Psi)\cos(\Phi) & \cos(\Theta)\sin(\Phi) \\ \cos(\Psi)\sin(\Theta)\cos(\Phi) + \sin(\Psi)\sin(\Phi) & \sin(\Psi)\sin(\Theta)\cos(\Phi) - \cos(\Psi)\sin(\Phi) & \cos(\Theta)\cos(\Phi) \end{bmatrix} \tag{2.10}$$

Reverse transformation matrix can be expressed as:

$$M_{OB} = M_{BO}^T. \tag{2.11}$$

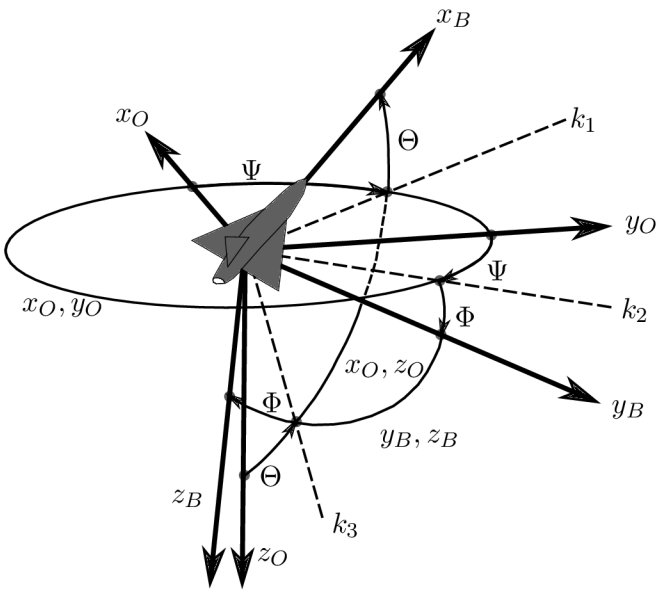


Figure 2.6: Relation between BFF (B) and NED frame (O) and Euler angles (Φ, Θ, Ψ).

2.2 Ground Motion Model

Assuming the reference frames discussed in previous section, force and momentum components acting on the aircraft can be defined. These forces and moments are put together to form the mathematical model in a form of the equations of motion. The model is designed for a use in dynamic analysis and real-time simulation. It therefore uses mathematical expressions for real-time computation, and is an alternative to earlier methods, which require storage of large amounts of data in look-up tables containing discrete values, which are then retrieved and interpolated. Ground motion model is based on previous work of [5], [28], [29] and the engine model characteristics are based on [19].

2.2.1 General Assumptions

The aircraft is modeled as a single rigid body with 3 degrees of freedom (DOF). It assumes two translational DOF (motion on the plane) and rotational motion about the vertical axis. The infrastructure of the all major airports is designed so that all the runways and the taxiways are flat so that danger in ground maneuvers is minimized. This allows us to abstract the surface to a flat plane and motion of the aircraft over it in two dimensions (no vertical motion). Since the aircraft is moving over almost ideal flat surface, one can assume there is no rotational motion about the x and y axis of the BFF reference frame (no roll and pitch).

The landing gear is a tricycle configuration (Figure 2.7) in which nose gear is used for steering. Pilot controls the deflection of the nose wheel via the rudder pedals that are primarily used for the control of the rudder influencing the aircraft's yaw while in the air. However, during the ground motion, the velocity remains relatively small therefore the rudder does not provide adequate response of direction change. The front wheel could be deflected up to 70° in both directions to provide sufficient maneuverability. To achieve even

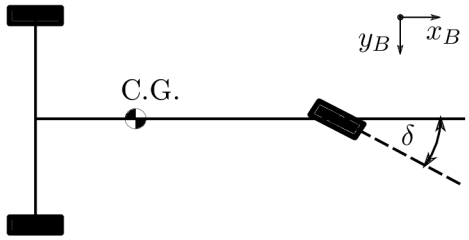


Figure 2.7: Tricycle configuration of the aircraft landing gear.

greater maneuverability the differential braking and engine thrust could be used. These techniques can be especially useful during U-turns on narrow taxiways.

There are usually two wheels per landing gear on an aircraft due to high loads that landing gear has to withstand. Since the spatial separation of the tires is very small the implemented model is simplified in a sense which assumes the two tires acting in unison.

The speed during the taxiing phase is not strictly limited by FAA, but in the document Runway Safety - Pilots Best Practices [15] it is stated: „maintain an appropriate taxi speed“. We might understand taxi speed limited to approximately 30 knots during motion on long straight line with no close obstacles, 20 knots with obstacles (other aircraft, ground vehicles, ground stands) and 10 knots during turns and entering ramp area [11]. A roll caused by a rapid cornering can be therefore neglected. Rapid braking and acceleration can result in a change of the pitch angle, but since we assume the aircraft to be a rigid body, the pitching effect (motion of the aircraft's nose up and down) can be neglected. However, the tire load will be influenced in such situations, affecting the performance of the brakes and steering abilities - during rapid braking more load is shifted to the nose wheel, reducing the maximum effective load of the main gear that can be used for braking and reversed effect during rapid acceleration when load is shifted from the nose to the main gear and the ability to steer is reduced. The load on each tire can dynamically change in the model depending on the acting forces, so that the simulation reflects the reality in high detail.

2.2.2 Equations of Motion

The forces acting on an aircraft are defined in Body Fixed reference frame, a conventionally accepted coordinate system in avionics. Forces are illustrated in Figure 2.8. The equations of motion for the velocities in the body coordinate system of the aircraft are given in a form of ordinary differential equations (ODEs) [28]:

$$m(\dot{V}_x - V_y\omega_z) = F_{xTL} + F_{xTR} - F_{xR} - F_{xL} - F_{xN}\cos(\delta) - F_{yN}\sin(\delta) - F_{xA}, \tag{2.12}$$

$$m(\dot{V}_y + V_x\omega_z) = F_{yR} + F_{yL} + F_{yN}\cos(\delta) - F_{xN}\sin(\delta), \tag{2.13}$$

$$\begin{aligned} I_{zz}\dot{\omega}_z = {} & l_{yR}F_{xR} - l_{yL}F_{xL} - l_{xR}F_{yR} - l_{xL}F_{yL} + \\ & + l_{xN}F_{yN}\cos(\delta) - l_{xN}F_{xN}\sin(\delta). \end{aligned} \tag{2.14}$$

Equations (2.12) and (2.13) are derived from the Newtonian law of motion, that says the vector sum of the forces acting on an object is equal to the mass of that object multiplied by the acceleration vector $\mathbf{a} = [\dot{V}_x, \dot{V}_y]$ of that object [17]. The mass of the aircraft is denoted by m and the components of the acceleration vector in x_B and y_B axis expressed via the velocity derivate are denoted as $\dot{V}_x$ and $\dot{V}_y$ respectively. Rotational motion of the aircraft is depicted by the projection of the perpendicular component of the acceleration vector

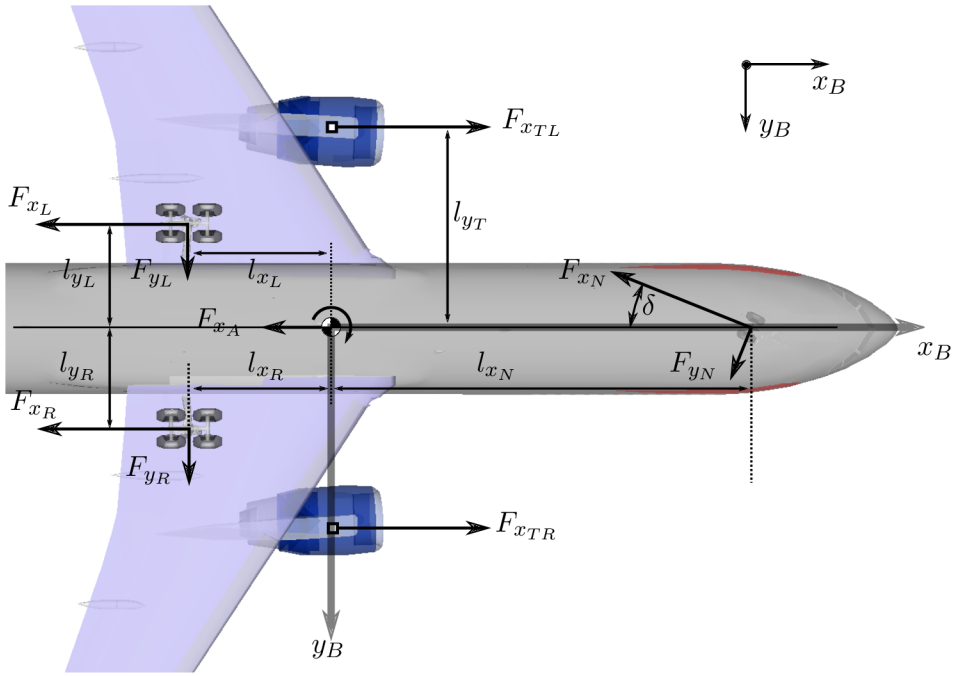


Figure 2.8: Schematic diagram illustrating force components acting on rigid aircraft body.

proportional to the angular velocity expressed via the $V_*\omega_z$ term (where $*$ represents either x or y component).

Equation (2.14) is formed by balancing the moments about the vertical axis z_B. The torque arms l_* are defined with respect to the the center of gravity. The dimensions l_* and principal moments of inertia are given in table 2.1. Parameter δ denotes the steer angle, i.e. the deflection of the nose wheel from axis x_B as illustrated in Figure 2.7.

The force components consists of longitudinal F_{x*} and lateral components F_{y*} for each tire ($*$ represents either right (R), left (L) or nose wheel (N)) resulting from interaction of tire and runway surface. The tire-ground interaction model is described in section 2.2.4. The thrust vector is represented by F_{xT} (section 2.2.5) and the aerodynamics is modeled as F_{xA} force, discussed in more detail at the end of this chapter.

Transformation to Navigational Frame To transform the effects of forces acting on aircraft into motion over spatial coordinates, the velocity vector needs to be transformed from BFF into navigational frame and then integrated over time domain. The transformation into world coordinates is performed using 2D transformation matrix as defined in equation (2.6).

$$\begin{bmatrix} V_x \\ V_y \end{bmatrix}_N = \begin{bmatrix} \cos(\Psi) & \sin(\Psi) \\ -\sin(\Psi) & \cos(\Psi) \end{bmatrix} \begin{bmatrix} V_x \\ V_y \end{bmatrix}_B \tag{2.15}$$

The rate of rotation is specified by the heading angle derivate $\dot{\Psi}$, that is angular velocity ω_z of the aircraft body around z_B axis of the BFF. Heading is a term used in navigation that refers to the direction where aircraft is pointing. If we neglect the effects of the wind, heading matches the direction that the vehicle actually travels, which is known as course. Heading is then determined by integrating angular velocity ω_z for given time interval and

Table 2.1: Aircraft parameters and their values [28].

Parameter	Description	Value
l_{x_N}	Nose gear x-distance relative to C.G.	11.235 m
$l_{x_{R,L}}$	Main gear x-distance relative to C.G.	1.450 m
l_{x_A}	Aerodynamic force x-distance relative to C.G.	0 m
l_{x_T}	Engine x-distance relative to C.G.	0 m
$l_{y_{R,L}}$	Main gear y-distance relative to C.G.	3.795 m
$l_{y_{T_{R,L}}}$	Engines y-distance relative to C.G.	5.755 m
l_{z_N}	Nose gear z-distance relative to C.G.	2.932 m
$l_{z_{R,L}}$	Main gear z-distance relative to C.G.	2.932 m
l_{z_A}	Aerodynamic force z-distance relative to C.G.	0.988 m
l_{z_T}	Engine z-distance relative to C.G.	1.229 m
m	Mass of the aircraft	45420 kg
I_{zz}	Moment of inertia about z-axis	3335000 kgm^2

initial condition:

$$\Psi = \int_{t_1}^{t_2} \omega_z = \int_{t_1}^{t_2} \dot{\Psi}. \tag{2.16}$$

The actual world coordinates of the aircraft are given by integrating the velocity components over entire time domain.

$$\begin{bmatrix} \dot{X} \\ \dot{Y} \end{bmatrix} = \begin{bmatrix} V_x \\ V_y \end{bmatrix}_N \tag{2.17}$$

2.2.3 Load Balancing

Longitudinal F_{x*} and lateral F_{y*} forces are both function of the vertical force F_{z*}. The mass of the aircraft engenders gravitational force F_g acting at the center of gravity, which in consequence produces the vertical forces F_{z*} acting against the gravitational force at the point of contact between the tire and the runway surface (see Figure 2.9). According to the Newton's third law of motion, the body exerts a force equal in magnitude and opposite in the direction on the another body when such body exerts a force on it [17]. Therefore we can derive following relation between vertical and gravitational forces:

$$F_{zN} + F_{zR} + F_{zL} = F_g. \tag{2.18}$$

By balancing the moments about axis y_B, while considering the rolling friction, braking and thrust force we obtain:

$$(l_{xN} - \mu_R l_{zN})F_{zN} + (l_{xR} - (\mu_R + k_B \mu_{B_{eff}})l_{zR})2F_{zR} = -l_{zT}F_{xT}. \tag{2.19}$$

When assuming that pitch and roll angle will remain small, since the taxiway speed of the aircraft is limited, we can assume that $F_{zR} = F_{zL}$. The relation for the vertical forces defining the load balance can be expressed as:

$$\begin{bmatrix} 1 & 1 \\ l_{xN} - \mu_R l_{zN} & l_{xR} - (\mu_R + k_B \mu_{B_{eff}})l_{zR} \end{bmatrix} \begin{bmatrix} F_{zN} \\ 2F_{zR} \end{bmatrix} = \begin{bmatrix} F_g \\ -l_{zT}F_{xT} \end{bmatrix} \tag{2.20}$$

Implications resulting from such definition of load balancing imply that the balance on a given tire will dynamically change depending on the amount of braking or thrust being

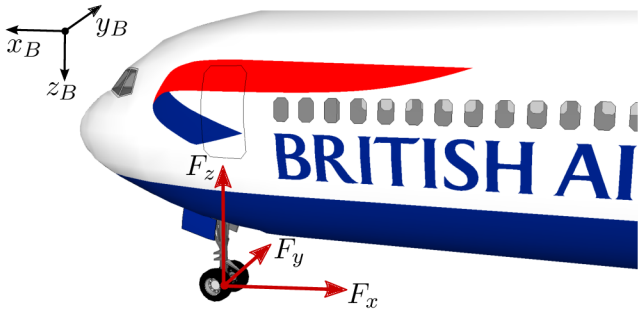


Figure 2.9: Forces generated at a nose tire strut.

applied. More load is shifted to nose wheel in case of braking and more load is shifted to main gear in case of acceleration, resulting in a slightly different motion characteristics of the aircraft.

2.2.4 Tire-Ground Interaction Model

The tire-ground interaction model has an essential impact on the ground motion model. Most of the forces influencing the behavior of an aircraft on ground originate at this contact point as a result of interaction between the wheel and the runway surface. These forces are a product of the vertical load (weight) on the tire F_z, and a coefficient of friction μ. The value of the friction coefficient is dependent upon many factors, including: type, texture and roughness of the runway surface; type and amount of pavement contaminant, e.g. snow, ice, water; tire construction, tread design and inflation pressure; type and efficiency of Automatic Brake System (ABS) and aircraft ground speed.

The ground forces may be decomposed into longitudinal and lateral components. Longitudinal forces act as a retarding forces incurred by the friction or by application of the wheel brakes. Longitudinal force acts in the direction opposite to the wheel motion. Lateral force is generated when the plane of a rolling wheel is yawed relative to its direction of a motion by an angle ψ. Lateral force acts in the direction perpendicular to the longitudinal force. The definition of tire-ground interface model is based on the source [5].

Longitudinal Force

Longitudinal force F_x acting on wheel is composed of two components - rolling friction F_{x_R} and braking F_{x_B}.

$$F_x = F_{x_R} + k_b F_{x_b}, \tag{2.21}$$

$$F_x = F_z \mu_R + k_b F_z \mu_{B_{eff}}. \tag{2.22}$$

where k_b is the proportion of brakes being used $k_b \in (0, 1)$; μ_R is rolling resistance constant and $\mu_{B_{eff}}$ is a braking effectiveness coefficient.

The rolling friction F_{x_R} is the resistance which one body offers to another when rolling along its surface [27]. It arises from the inequalities of the interfering surfaces. On microscopic level there are other processes at work, including chemical bonding and electrical interactions. However, the dominant factor of rolling friction are non-elastic effects. That is, not all energy used for deformation is recovered after the pressure is removed. Two forms of this energy dissipation are hysteresis losses and permanent (plastic) deformation of the

surface (e.g. soil). It is the rubber compound in a tire that exhibits hysteresis. As the tire rotates under the weight of the aircraft, it experiences repeated cycles of deformation and recovery, and it dissipates the hysteresis energy loss known as the heat. Suggested value for the rolling resistance coefficient is $\mu_R = 0.02$ [28] for an typical mid-sized single aisle passenger aircraft such as Boeing 737-400. The coefficient of the rolling resistance is generally much smaller than the coefficient of sliding friction, which determines the braking performance.

The retarding force produced by the tire on a braked wheel, F_{x_B}, is a product of the vertical load on the tire F_z, and a coefficient of friction $\mu_{B_{eff}}$, which varies upon many factors. A definition of the friction coefficients for all operating conditions is needed not only for braking performance prediction, but also to assess directional stability and control on the ground.

Two friction coefficients influence the braking performance. They are the maximum braking coefficient $\mu_{b_{max}}$, which is available just before slipping of the rolling wheel occurs, and the tire skid coefficient μ_{skid}, the friction coefficient of a locked wheel (fully developed skid). In a fully developed skid, the available retarding force is greatly reduced, and brakes should be operated to avoid this condition. Manual operation in critical conditions requires skill; consequently, most aircraft are fitted with an ABS, to prevent skidding. An ABS reduces the maximum available deceleration by about 10% on dry and wet surfaces and by about 20% on flooded, icy, and snow-covered runways [5]. A third braking coefficient, the braking effectiveness coefficient, $\mu_{B_{eff}}$, is used to take account of the ABS efficiency. These braking coefficients are used to calculate the forces acting on the tire, in the plane of the wheel rotation.

Dry Runway Surface Conditions Following relationships between the friction coefficients, tire pressure $p[Pa]$ and velocity $v[m/s]$ are assuming dry surface of a wire-brushed concrete runway. Equations are valid for $v < 50m/s$ [5].

$$\mu_{b_{max}} = 0.912(1 - 7.5842p) - 4.0641 \cdot 10^{-4}v, \tag{2.23}$$

$$\mu_{B_{eff}} = -0.03 + 0.94\mu_{b_{max}}, \tag{2.24}$$

$$\mu_{skid} = \frac{48.1}{50.2 + 0.5144v}\mu_{b_{max}}. \tag{2.25}$$

Wet Runway Surface Conditions Following equations are assuming wet surface of a wire-brushed concrete runway. Equations are valid for $v < 70m/s$ [5].

$$\mu_{b_{max}} = (0.91 - 6.8947p)(1 - 2.6751 * 106{-}3v), \tag{2.26}$$

$$\mu_{B_{eff}} = -0.03 + 0.94\mu_{b_{max}}, \tag{2.27}$$

$$\mu_{skid} = \frac{23.2 - 213.737p}{26.5 + 0.5144v}. \tag{2.28}$$

Snow Covered Runway Surface When the runway is severely contaminated, the influences of the type of runway surface and tire pressure are small, and can be neglected for this type of model. The relationships below assume approximately $15cm$ snow layer on the runway [5].

$$\mu_{b_{max}} = 0.185 + 5.144 \cdot 10{-}4v, \tag{2.29}$$

$$\mu_{B_{eff}} = 0.8\mu_{b_{max}}, \tag{2.30}$$

$$\mu_{skid} = \mu_{b_{max}}(0.8 - 2.0576 \cdot 10^{-4}v), \qquad v < 25m/s, \tag{2.31}$$

$$\mu_{skid} = 0.6\mu_{b_{max}}, \qquad v \geq 25m/s. \tag{2.32}$$

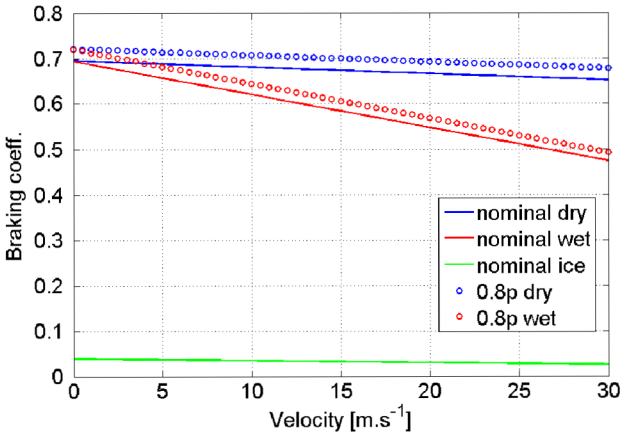


Figure 2.10: Braking effectiveness coefficient, $\mu_{B_{eff}}$ in various conditions.

Lateral Force

The lateral side-force F_y is created when the plane of a rolling wheel is yawed relative to the direction of the motion by an angle ψ_* ($*$ could be either R, L or N) according to a given tire of an aircraft. Two additional friction coefficients are needed to describe the lateral forces. The maximum lateral friction coefficient $\mu_{\psi_{max}}$, and the limiting lateral friction coefficient, $\mu_{\psi_{lim}}$. Coefficients differ in dependence on whether the brakes are applied. First mentioned characterizes the unbraked yawed rolling tire, the later considers the application of brakes. The model must include the effect of braking, because it can considerably reduce the maximum side-force generated by a yawed wheel. In such situations, total friction is shared between the side-force generation and the longitudinal deceleration.

Same conditions as mentioned above apply for various surface conditions. The k_b represents proportion of brakes being applied relative to maximal braking force$k_b \in (0, 1)$.

Dry Runway Surface Conditions Following equations are assuming dry surface of a wire-brushed concrete runway [5].

$$\mu_{\psi_{max}} = \mu_{b_{max}}, \tag{2.33}$$

$$\mu_{\psi_{lim}} = \mu_{b_{max}} \sqrt{1 - \left(k_b \cdot \frac{\mu_{B_{eff}}}{\mu_{b_{max}}} \right)^2}. \tag{2.34}$$

Wet and Snow Covered Runway Surface Conditions Equations for the maximum and limiting lateral friction coefficients are defined in the same way for wet and snow covered runway conditions. The difference is in the value of $\mu_{b_{max}}$ and μ_{eff} that differs with altering runway surface conditions (see relations for wet surface in equations (2.26) and (2.27), for snow cowered runway see equations (2.29) and (2.30)) [5].

$$\mu_{\psi_{max}} = 0.64\mu_{b_{max}} + 0.15\mu_{b_{max}}^2, \tag{2.35}$$

$$\mu_{\psi_{lim}} = \mu_{b_{max}} \sqrt{1 - \left(k_b \cdot \frac{\mu_{B_{eff}}}{\mu_{b_{max}}} \right)^2}. \tag{2.36}$$

The lateral force model represents the contact conditions between the tire and the ground as speed, tire yaw angle and load on the tire vary. It takes into account variation in ground surface conditions (contamination) and is valid over a large range of speeds and tire yaw angles. An adaptive braking system (ABS) is assumed (to avoid the complications of modeling wheel slip when braking).

To avoid an over-complex model, the effects of a non-vertical wheel, of self-aligning torque, and of pneumatic castor are neglected in the following cornering force model. In general, these effects have little influence on the performance and the dynamic stability of a modern aircraft during ground roll.

From definition [5], lateral force F_y is:

$$F_y = F_z \mu_\psi, \tag{2.37}$$

where lateral friction coefficient μ_ψ is

$$\begin{aligned} \mu_\psi &= \mu_{sya} && |\psi| < \psi_{lim}, \\ \mu_\psi &= \mu_{lya} && |\psi| \geq \psi_{lim}, \end{aligned} \tag{2.38}$$

where $\psi_{lim} \approx 20^\circ$ is the tire yaw angle beyond which the lateral friction coefficient decreases.

A tire yaw angle ψ is defined as an angle between the longitudinal tire axis and the velocity vector of a given tire (see Figure 2.11). When the plane of a rolling wheel is yawed relative to the direction of motion, lateral force F_y acting perpendicular to tire axis, is produced as a result of the tire deformation.

For small yaw angles (sya), coefficient of lateral friction μ_{sya} is a function of tire yaw angle ψ, tire pressure p (including maximal attainable tire pressure, rated pressure p_r) and its parameters: undeflected tire diameter d and width w. The tire parameters are assumed to be constant in the model, which does not fully corresponds with the reality. These parameters might dynamically change under different aircraft settings and ground roll situations but since the effect of such changes has minor effect on final model, one can assume theses numbers to be constant without any noticeable loss of information.

Maximum $\mu_{\psi_{max}}$ and limiting $\mu_{\psi_{lim}}$ lateral friction coefficients are influencing lateral friction coefficient depending on whether the brakes are applied. In case no brakes are applied the maximum lateral friction coefficient $\mu_{\psi_{max}}$ is used, otherwise the limiting lateral

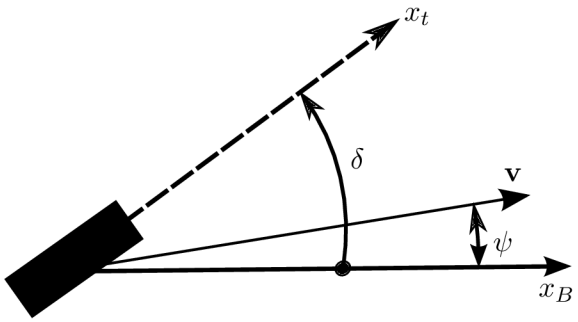


Figure 2.11: Tire yaw angle ψ. Axis of local tire coordinate system is denoted as x_t.

friction coefficient $\mu_{\psi_{lim}}$ is used. Formulas in following sections are presented assuming no brakes being applied [5].

$$\begin{aligned} \mu_{sya} &= \mu_{\psi_{max}} |\phi - \frac{4}{27}\phi^3|, && \text{for } |\phi| < 1.5, \\ \mu_{sya} &= \mu_{\psi_{max}}, && \text{for } |\phi| \geq 1.5. \end{aligned} \tag{2.39}$$

Function ϕ of tire yaw angle ψ is defined as:

$$\phi = \frac{N\psi}{\mu_{\psi_{max}} F_z}, \tag{2.40}$$

and function N of tire characteristics is defined as:

$$N = 31.3w^2(p + 0.44p_r)(1 - 3.17x)x, \tag{2.41}$$

$$x = \frac{F_z}{pd\sqrt{wd}}. \tag{2.42}$$

For large yaw angles (lya) beyond the ψ_{lim}, lateral friction coefficient decreases. Good approximation for the ψ_{lim} is given by

$$\psi_{lim} = \frac{2\mu_{\psi_{max}}}{\left(\frac{\partial \mu_\psi}{\partial \psi}\right)_{\psi=0}} \approx 2\mu_{\psi_{max}} \frac{F_z}{N}. \tag{2.43}$$

It is the value of ψ at the intersection of the μ_ψ vs. ψ curve with a line from the origin at half the initial slope of the curve.

Lateral friction coefficient μ_{lya} at large yaw angles is then defined as:

$$\begin{aligned} \mu_{lya} &= \mu_{skid} + j(\mu_{\psi_{max}} - \mu_{skid}), && \text{for } \mu_{\psi_{max}} > \mu_{skid}, \\ \mu_{lya} &= \mu_{\psi_{max}}, && \text{for } \mu_{\psi_{max}} \leq \mu_{skid}. \end{aligned} \tag{2.44}$$

Subsidiary function j used to define the lateral friction coefficient at large yaw angles is defined as:

$$\begin{aligned} j &= 1 - 1.93i, && \text{for } i < 0.3, \\ j &= 0.58 - 0.575i, && \text{for } i \geq 0.3. \end{aligned} \tag{2.45}$$

Subsidiary function i is defined as:

$$i = \frac{\psi - \psi_{lim}}{\pi/2 - \psi_{lim}}. \tag{2.46}$$

Allowing for positive and negative values of ψ and allowing values of ψ up to 180° we obtain following definition:

$$i = 0, \qquad \text{for } 0 \leq |\psi| < \psi_{lim}, \tag{2.47}$$

$$i = \frac{|\psi| - \psi_{lim}}{\pi/2 - \psi_{lim}}, \qquad \text{for } \psi_{lim} \leq |\psi| < \pi/2, \tag{2.48}$$

$$i = 2 + \frac{\psi_{lim} - |\psi|}{pi/2 - \psi_{lim}}, \qquad \text{for } \pi/2 \leq |\psi| < \pi - \psi_{lim}, \tag{2.49}$$

$$i = 0, \qquad \text{for } \pi - \psi_{lim} \leq |\psi|. \tag{2.50}$$

Because of the definition of function i we need to adjust formula (2.37) in order to allow for negative tire yaw angles ψ:

$$F_y = F_z \mu_\psi, \qquad \text{for } \psi > 0, \tag{2.51}$$

$$F_y = -F_z \mu_\psi, \qquad \text{for } \psi < 0. \tag{2.52}$$

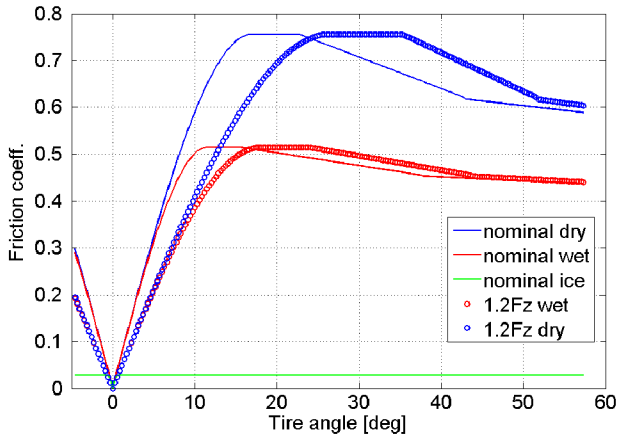


Figure 2.12: Lateral friction coefficient μ_ψ as a function of the tire yaw angle ψ.

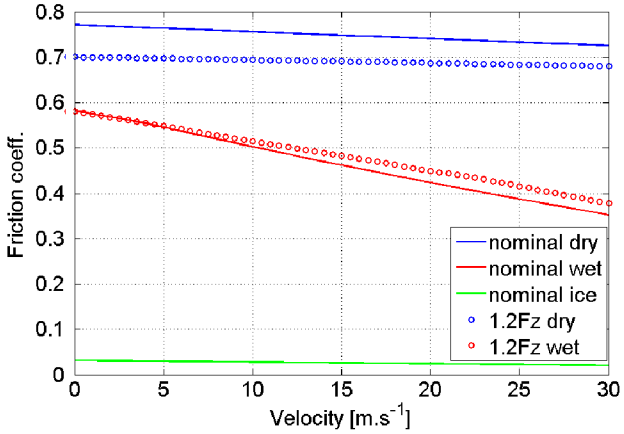


Figure 2.13: Lateral friction coefficient μ_ψ as a function of the tire velocity.

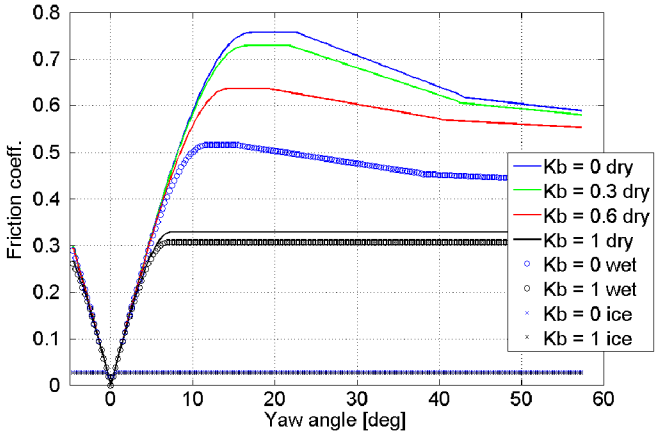


Figure 2.14: Lateral friction coefficient μ_ψ as a function of braking force ratio k_B applied.

2.2.5 Engine Model

A single aisle passenger aircraft (Boeing 737-400) equipped with two two high-bypass turbofan engines providing maximum thrust power of approximately 110 kN each is considered in this study.

The principle of how such engine works is that the incoming air is captured by the engine inlet and is accelerated inside the engine. Some of the incoming air passes through the fan, which is basically a large diameter propeller and continues into the core compressor with multiple stages gradually increasing the air pressure. Then the air gets into the burner, where it is mixed with jet fuel and the combustion occurs. The hot exhaust gases (reaching temperatures up to 1700°) pass through the core and multiple-stage turbines, where the high pressure gas is reduced as it drives the compressor and the fan. Finally, the air mixed with exhaust gases then leaves the nozzle. The rest of the incoming air passes through the fan and bypasses, or goes around the engine, just like the air through a propeller. The air that goes through the fan has a velocity that is slightly increased from free stream. So a turbofan gets some of its thrust from the core and some of its thrust from the fan. The ratio of the air that goes around the engine to the air that goes through the core is called the bypass ratio. In case of the high-bypass turbofan engine considered in this thesis, most of the incoming air bypasses the core ($\approx 80\%$) providing most of the thrust power [19].

The propulsion efficiency is a function of the relative airspeed of the exhaust to the surrounding air, therefore classical propeller engines are most efficient at low speed, pure jet engines at high (supersonic) speed and turbofan are most efficient at some point between these two at speeds reaching 500-1000 km/h which is the operational speed of most commercial aircrafts.

The model of a typical high by-pass ratio two-shaft gas turbine for civil aircraft engine contains high dimensional parameter space. Multiple input signal are identified, including fuel flow, inlet guide vanes position and bleed flow. Most relevant output signals include low pressure rotor (fan) and high pressure rotor (compressor) speeds, low pressure turbine pressure and temperature of gas at high pressure turbine [19]. During normal operation, an aircraft turbofan engine experiences large changes in ambient temperature, pressure, Mach number, and power output level. Consequently, the engine dynamics change in a significant nonlinear manner, therefore for authentic simulation these factors need to be considered.

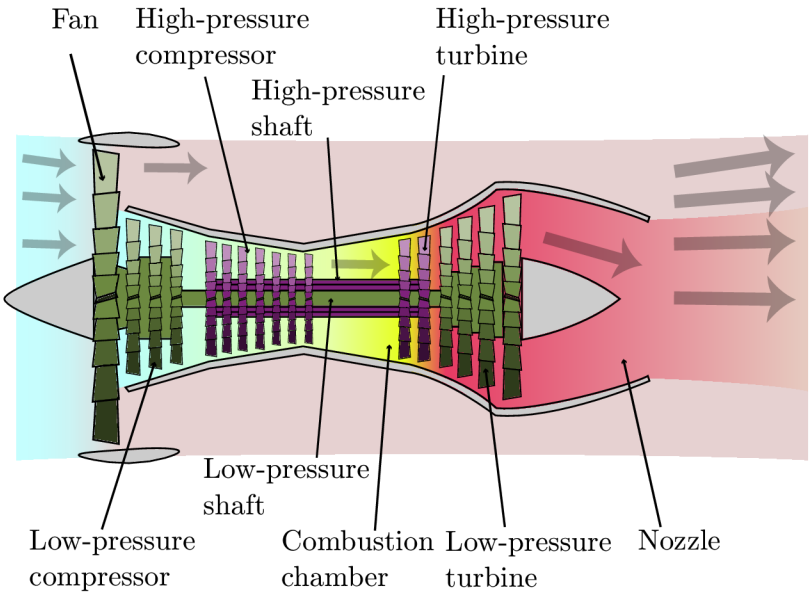


Figure 2.15: A schematic diagram illustrating the operation of a 2-spool, high-bypass turbofan engine [1].

However, in the case of ground motion model a lot of parameters are fixed or limited to very small operational range so that one can avoid an over-complex model definition. Under these assumptions, equivalent second-order model can be easily derived from the original high-dimensional description. The main control variables can be simplified to two signals that are fuel flow and fan speed. The dynamics of the engine are then defined by a second order transfer function fuel of flow to fan speed, which is of a key relevance in the engine control. Fuel flow is controlled by a throttle and engine output thrust is directly proportional to the fan speed.

$$H(s) = \frac{F_{xTmax}}{s^2 + 2s + 1} + F_{xTconst}. \tag{2.53}$$

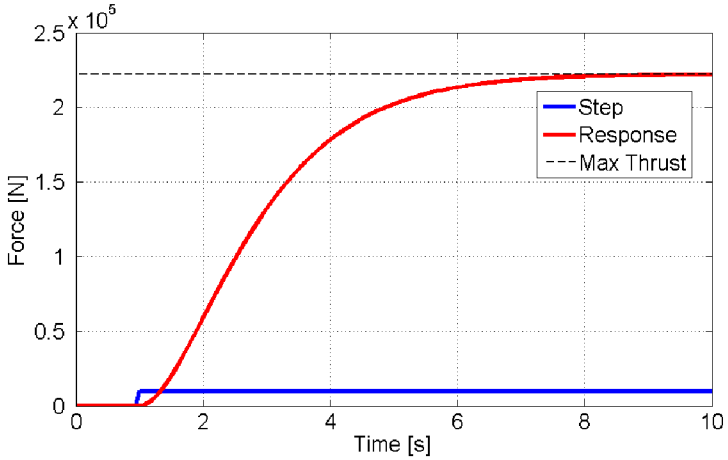


Figure 2.16: The unit step response of the engine model. Unit step signal is amplified by a factor of 10^4 for better readibility.

3. Design and Implementation of Aircraft Taxi Simulation Model

In the previous chapter, a mathematical model of the aircraft ground motion was defined. The mathematical model is an abstraction of reality, making various assumptions and limiting the boundaries of acceptable values of system variables. However the most important aspects of reality are present in a mathematical model in a form of a mathematical relations approximating the observed reality. The mathematical relations describe the behavior of an aircraft in ground operations under various environmental conditions so that the simulation result is as close as possible to real behavior of the system. The evolution of the system strictly obeys the physical laws that govern the real physical processes in the simulation region. Then the result of such simulation can have a good representation of the real environment.

Next step is transforming the mathematical model into a simulation model in form of a software product that is executable on appropriate computer platform. With such model one can perform simulations of the aircraft movement. Simulation allows us to experiment with the model and evaluate experiments that would be too expensive to perform in reality or not even possible to perform at all. From the result of such simulation one can safely draw conclusions and have a better understanding of the system and reality. Based on observation and experimentation with modeled system, knowledge can be obtained. This knowledge is then described in a form of mathematical equations and physical laws that form the mathematical model (also known as abstract model). Representing the abstract model by programming language makes executable simulation model that one can use to learn more about reality and modeled environment (see Figure 3.1).

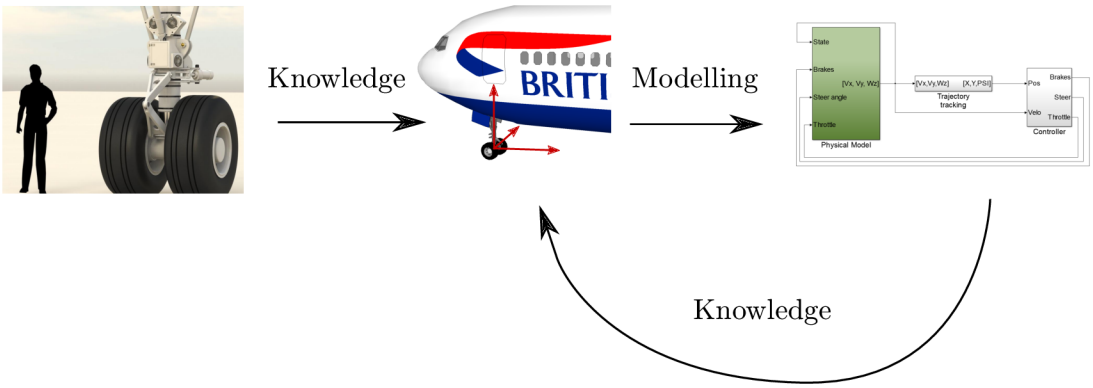


Figure 3.1: The principle of modelling and simulation.

Numerical simulations need vast computer resources. Simulating complex problems

requires usage of supercomputers and a large amount of computer resources, such as compute time or compute cores. It is more expensive than theoretical study but it is still much cheaper than experiments using real aircrafts. The rapid development of the computer industry is providing more and more powerful computing resources for a numerical simulation, which also makes numerical simulation more applicable to a wider area and more complex problems. Unlike observation which has time and space limitations, as long as there is enough observation basis, numerical techniques can simulate a very large scale regions and run almost unlimited long periods of time. Unlike theoretical study, which can not treat complex problems, numerical simulation can deal with many physical processes at the same time. Even nonlinear processes pose no difficulty to numerical simulation [30].

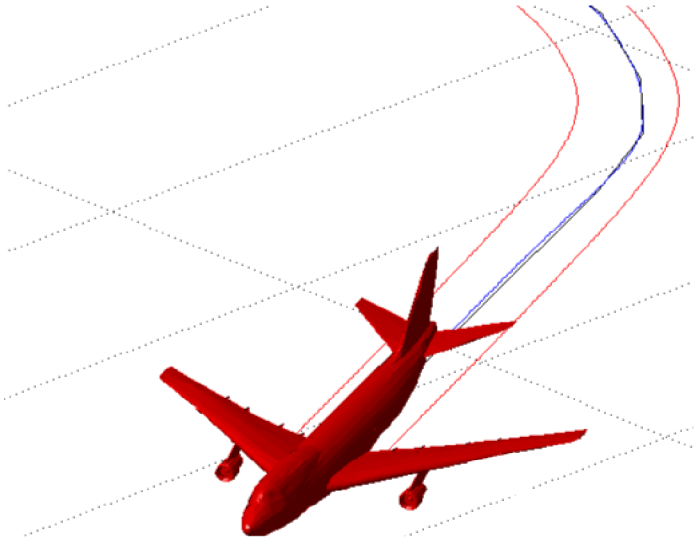


Figure 3.2: Aircraft ground motion simulation.

3.1 Numerical Methods

Presented system is modeled by a set of ordinary differential equations (ODE) - equations of motion ((2.12), (2.13) and (2.14)) based on Newtonian physics. The overwhelming majority of ODE do not have exact solution, which can be expressed in terms of simple functions. For this reason, one must rely on numerical methods that produce approximations to the desired solutions [18]. A first-order differential equation is an Initial value problem (IVP) of the form:

$$y'(t) = f(t, y(t)), \qquad y(t_0) = y_0, \tag{3.1}$$

where f is a function that maps $< t_0, \infty) \times R^d$ to R^d, and the initial condition $y_0 \in R^d$ is a given vector. First-order means that only the first derivative of y appears in the equation, and higher order derivatives are not included. In fact, first order equations are present only in case where the velocity derivate is expressed as sum of forces and moments. In order to obtain position of the aircraft, the second derivate appears in the equations. Without loss of generality, higher-order ODE can be converted into a larger system of first-order equations by introducing extra variables [18].

3.1.1 Euler Method

Euler method is a numerical procedure for solving ODE with a given initial value. From any point on a curve, you can find an approximation of a nearby point on the curve by moving a short distance along a line tangent to the curve. Starting with the differential equation (3.1), the derivative y' is replaced by the finite difference approximation

$$y'(t) \approx \frac{y(t+h) - y(t)}{h}, \tag{3.2}$$

which when re-arranged yields the following formula

$$y(t+h) \approx y(t) + hy'(t). \tag{3.3}$$

Substituting y' from (3.1) gives:

$$y(t+h) \approx y(t) + hf(t, y(t)). \tag{3.4}$$

This formula is usually applied in a way that reasonable step size h is chosen to compute the sequence $t_0,\ t_1 = t_0 + h,\ t_2 = t_0 + 2h$ and so on. We denote by y_n a numerical estimate of the exact solution $y(t_n)$. Motivated by (3.4), we compute these estimates by following recursive scheme:

$$y_{n+1} = y_n + hf(t_n, y_n). \tag{3.5}$$

The Euler method is an an explicit method, which means that the new value y_{n+1} is defined in terms of values that are already known, like y_n. The backward Euler method is an implicit method, meaning that an equation must be solved in order to find y_{n+1}:

$$y_{n+1} = y_n + hf(t_{n+1}, y_{n+1}). \tag{3.6}$$

It costs more time to solve this equation than explicit method and this cost must be taken into consideration when one selects the method to use. The advantage of implicit methods such as backward Euler is that they are usually more stable for solving a stiff equation, meaning that a larger step size h can be used. Another advantage of implicit methods is the increased stability and accuracy, which is limited in standard Euler method [14].

Stiffness is often caused by the presence of different time scales in the underlying problem. Stiff problems are ubiquitous in control theory that deals with the behavior of dynamical systems with inputs, and describes how their behavior is modified by feedback. Control theory for the presented model will be described in later sections of this text.

3.1.2 Runge-Kutta Methods

Runge-Kutta methods are procedures of numerically integrating ordinary differential equations by using a trial step. They are known to be very accurate and well-behaved for a wide range of problems.
The second order RK2 takes a form:

$$k_1 = hf(t, y(t)), \tag{3.7a}$$

$$k_2 = hf(t + \frac{h}{2}, y(t) + \frac{k_1}{2}), \tag{3.7b}$$

$$y(t+h) = y(t) + k_2. \tag{3.7c}$$

Fourth order RK4 takes a form:

$$k_1 = hf(t, y(t)), \tag{3.8a}$$

$$k_2 = hf(t + \frac{h}{2}, y(t) + \frac{k_1}{2}), \tag{3.8b}$$

$$k_3 = hf(t + \frac{h}{2}, y(t) + \frac{k_2}{2}), \tag{3.8c}$$

$$k_4 = hf(t + h, y(t) + k_3), \tag{3.8d}$$

$$y(t + h) = y(t) + \frac{k_1}{6} + \frac{k_2}{3} + \frac{k_3}{3} + \frac{k_4}{6}. \tag{3.8e}$$

This method is reasonably simple and robust and is a good general candidate for numerical solution of differential equations when combined with an intelligent adaptive step-size routine.

3.1.3 Multistep Adams' Methods

Up to now, all studied methods were single step methods. As opposed to single-step methods, which only utilize one previous value of the numerical solution to approximate the subsequent value, multistep methods approximate numerical values of the solution by referring to more than one previous value. Accordingly, multistep methods may often achieve greater accuracy than one-step methods that use the same number of function evaluations, since they utilize more information about the known portion of the solution than one-step methods do [13].

Consider the first order ODE as defined in (3.1). If we want to integrate from t_{n+1} to t_{n+2} we have:

$$y(t_{n+2}) - y(t_{n+1}) = \int_{t_{n+1}}^{t_{n+2}} f(t, y(t))dt. \tag{3.9}$$

This motivates the second-order Adams-Bashforth method:

$$y_{n+2} = y_{n+1} + \frac{h}{2}\left[3f(t_{n+1}, y_{n+1}) - f(t_n, y_n)\right]. \tag{3.10}$$

Since formula (3.10) involves two previously computed solution values, this method is known as a two-step method. Note that this method requires two initial conditions. Since the IVP will give us only one initial condition in practice one often precedes the Adams-Bashforth method by one step of, e.g., a second-order Runge-Kutta method to obtain remaining starting values of the multi-step method.

In general, a nth-order Adams method is obtained by replacing the integrand f in (3.9) by a polynomial of degree $p-1$. However, the Adams-Bashforth (AB) method is an explicit method that uses the most recent information as well as $p-1$ previous points to fit the polynomial. The pth-order Adams-Moulton method (AM) is an implicit method that fits the polynomial to the point to be determined next, the current point, and $p-2$ previous points [14]. Adams-Moulton methods have smaller error constants, use less steps, and have larger stability regions than their Adams-Bashforth counterparts (of the same order). However, AM methods using more than one step tend to have smaller regions of absolute stability than other implicit methods such as Runge-Kutta methods [13].

Combining Adams-Bashforth and Adams-Moulton methods we get predictor-corrector methods. First, the prediction step calculates a rough approximation of the desired quantity, typically using an explicit AB method. Second, the corrector step refines the initial approximation using another means, typically an implicit AM method.

3.1.4 Variable Step Solvers

All of the above methods involved fixed step size implementations of various algorithms. That is, step size h have to be specified explicitly. It is possible to have the algorithm itself select, for each step, the step size that it thinks will most efficiently give a specified accuracy. That means step size varies from step to step, depending on model dynamics [16]. Step size is reduced when model states change rapidly, to maintain accuracy and on the other hand, step size is increased when model states change slowly, to avoid unnecessary steps which consume precious compute time and resources. Variable-step is recommended for models in which states change rapidly or that contain discontinuities. In these cases, a variable-step solver requires fewer time steps than a fixed-step solver to achieve a comparable level of accuracy. This can significantly shorten simulation time.

Suppose we want to generate an approximation to the initial value problem as defined in (3.1) for some range of t's and we want the error introduced per unit increase of t to be no more than about ε. Suppose further that we have already produced the approximate solution as far as t_n. The rough strategy is as follows. We do the step from t_n to $t_n + h$ twice using two different algorithms, giving two different answers, that we call A_1 and A_2. The two algorithms are chosen so that:

- we can use $A_1 - A_2$ to compute an approximate local truncation error
- the two algorithms use almost the same evaluations of f.

In case that the local truncation error, divided by h, (i.e. the error per unit increase of t) is smaller than ε, we set $t_{n+1} = t_n + h$, accept A_2 as the approximate value for $y(t_{n+1})$ and move on to the next step. Otherwise we pick, using what we have learned from $A_1 - A_2$, a new trial step size h and start over again at t_n.

Let's assume that $\phi(t)$ is the exact solution to IVP $y' = f(t, y(t))$ that satisfies the initial condition $\phi(t_n) = y_n$. If we apply one step of Euler with step size h, giving

$$A_1 = y_n + hf(t_n, y_n), \tag{3.11}$$

we know that

$$A_1 = \phi(t_n + h) + Kh^2 + O(h^3). \tag{3.12}$$

The problem of course is that we do not know what the error is, even approximately, because we do not know what the constant K is. But we can determine K simply by redoing the step from t_n to t_{n+h} using a judiciously chosen second algorithm. There are a number of different second algorithms that will work. We will use two step Euler. One step of Euler-2step with step size h just consists of doing two steps of Euler of size $h/2$:

$$A_2 = y_n + \frac{h}{2} f(t_n, y_n) + \frac{h}{2} f\left(t_n + \frac{h}{2}, y_n + \frac{h}{2} f(t_n, y_n)\right). \tag{3.13}$$

The local truncation error introduced in the first half-step is $K(h/2)^2 + O(h^3)$. That for the second half-step is $K(h/2)^2 + O(h^3)$ with the same K, though a different $O(h^3)$. All together

$$A_2 = \phi(t_n + h) + \frac{1}{2} Kh^2 + O(h^3). \tag{3.14}$$

The difference is

$$A_1 - A_2 = \phi(t_n + h) + Kh^2 + O(h^3) - \phi(t_n + h) - \frac{1}{2} Kh^2 - O(h^3) \tag{3.15}$$

$$= \frac{1}{2} Kh^2 + O(h^3). \tag{3.16}$$

So if we do one step of both Euler and Euler-2step, we can estimate

$$\frac{1}{2}Kh^2 = A_1 - A_2 + O(h^3) \tag{3.17}$$

We now know that in the step just completed Euler-2step introduced an error of about $\frac{1}{2}Kh^2 \approx A_1 - A_2$. That is, the current error rate is about $r = \frac{|A_1 - A_2|}{h} \approx \frac{1}{2}Kh$ per unit increase of t. If $r > \varepsilon$, we reject A_2 and repeat the current step with a new trial step size h' chosen so that $\frac{1}{2}|K|h' \approx \frac{r}{h}h' < \varepsilon$.

If $|A_1 - A_2|/h < \varepsilon$ we could accept A_2 as an approximate value for $y(t_{n+1} = t_n + h)$ and move on to the next step. But $\phi(t_n + h) = A_2 - \frac{1}{2}Kh^2 + O(h^3) = 2A_2 - A_1 + O(h^3)$, so we do better by setting

$$y_{n+1} = 2A_2 - A_1 \tag{3.18}$$

For the next step, we would repeat the whole process, starting with a trial step size $h' = .9\frac{\varepsilon}{r}h$ to give ourselves a small safety margin [16].

3.2 Simulink Modeling Tool

Simulink® is a graphical modelling tool for modeling, simulating and analyzing multidomain dynamic systems by signal flow graphs. Its primary interface is a graphical block diagramming tool and a customizable set of block libraries. Simulink® is widely used in control theory and digital signal processing for multidomain simulation. It can automatically generate C source code for real-time implementation of systems. As the efficiency and flexibility of the code improves, this is becoming more widely adopted for production systems, in addition to being a popular tool for embedded system design work because of its flexibility and capacity for quick iteration [8].

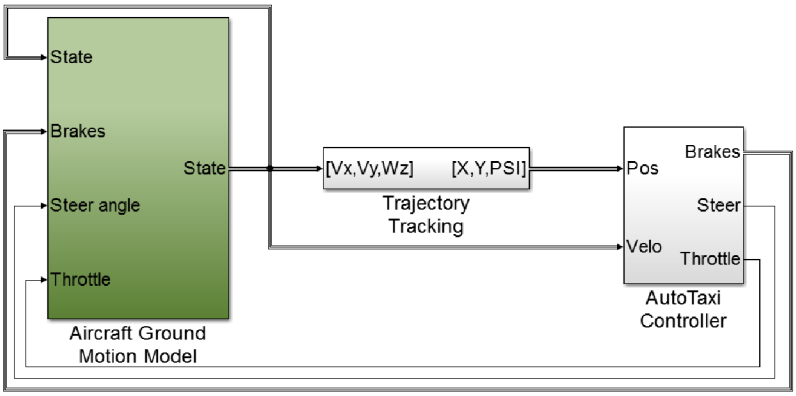


Figure 3.3: Simulink® model of the aircraft ground motion model and AutoTaxi controller.

Model in simulink consists of basic elements - blocks, that are characterized by input, output and its functionality encapsulated inside the block. Various blocks are provided by simulink block libraries, such as *Math* library that provides mathematical and trigonometric functions and arithmetic, logical and relational operators. System programmer can use these block as basic building elements for his model. The signal routes enable to interconnect the blocks and feed the output of one block to input of another or feed-back the signal back to input port in case of closed loop systems. A set of blocks can be grouped into subsystem, creating new block with its inputs and outputs. Such scheme allows modular development

by combining blocks of similar function and create hierarchic structures of blocks allowing to create complex models.

Simulink® provides rich scale of numerical methods, including those listed above. System programmer can select numerical solver, that computes a dynamic system's states at successive time steps over a specified time span and has option of configuring the numerical simulation, such as specifying the simulation start and stop time, specifying step size in case of fixed-step solvers or maximum/minimum step size and error tolerance in case of variable-step solvers [8].

3.2.1 Fixed Step Solvers

Fixed step solvers include [20]:

- **ode1 (Euler)** - Uses the Euler integration method to compute the model state at the next time step as an explicit function of the current value of the state and the state derivatives.
- **ode3 (Bogacki-Shampine)** - Computes the model's state at the next time step as an explicit function of the current value of the state and the state derivatives, using the Bogacki-Shampine Formula integration technique to compute the state derivatives.
- **ode4 (Runge-Kutta)** - Uses the fourth-order Runge-Kutta (RK4) formula to compute the model state at the next time step as an explicit function of the current value of the state and the state derivatives.
- **ode8 (Dormand-Prince RK8(7))** - Uses the eighth-order Dormand-Prince formula to compute the model state at the next time step as an explicit function of the current value of the state and the state derivatives approximated at intermediate points.

3.2.2 Variable Step Solvers

Variable step solvers include [20]:

- **ode45 (Dormand-Prince)** - Computes the model's state at the next time step using an explicit Runge-Kutta (4,5) formula (the Dormand-Prince pair) for numerical integration.
- **ode23 (Bogacki-Shampine)** - Computes the model's state at the next time step using an explicit Runge-Kutta (2,3) formula (the Bogacki-Shampine pair) for numerical integration. `ode23` is a one-step solver, and therefore only needs the solution at the preceding time point. `ode23` is more efficient than ode45 at crude tolerances and in the presence of mild stiffness.
- **ode113 (Adams)** - Computes the model's state at the next time step using a variable-order Adams-Bashforth-Moulton predictor/corrector numerical integration technique. `ode113` is a multistep solver, and thus generally needs the solutions at several preceding time points to compute the current solution. `ode113` can be more efficient than ode45 at stringent tolerances.

- **ode15s (stiff/NDF)** - Computes the model's state at the next time step using variable-order numerical differentiation formulas. These are related to, but more efficient than the backward differentiation formulas, also known as Gear's method. `ode15s` is a multistep solver, and thus generally needs the solutions at several preceding time points to compute the current solution. `ode15s` is efficient for stiff problems. This solver should be considered if `ode45` fails or is inefficient.

- **ode23s (stiff/Mod. Rosenbrock)** - Computes the model's state at the next time step using a modified Rosenbrock formula of order 2. `ode23s` is a one-step solver, and therefore only needs the solution at the preceding time point. `ode23s` is more efficient than `ode15s` at crude tolerances, and can solve stiff problems for which `ode15s` is ineffective.

- **ode23tb (stiff/TR-BDF2)** - Computes the model's state at the next time step using a multistep implementation of TR-BDF2, an implicit Runge-Kutta formula with a trapezoidal rule first stage, and a second stage consisting of a backward differentiation formula of order two. By construction, the same iteration matrix is used in evaluating both stages. `ode23tb` is more efficient than `ode15s` at crude tolerances, and can solve stiff problems for which `ode15s` is ineffective.

Identifying the optimal solver for a model requires experimentation. The optimal solver should balance acceptable accuracy with the shortest simulation time. General rules for selecting solver properties are that a smaller step size increases accuracy, but also increases simulation time and the degree of computational complexity increases for `ode`n, as n increases. As a result of increasing the computational complexity, one obtains increased accuracy of the results [20].

For presented model, fixed step Bogacki-Shampine solver is used. It computes the model's state at the next time step as an explicit function of the current value of the state and the state derivatives, using the Bogacki-Shampine Formula integration technique. Variable step solver are not efficient for presented model, since there is a large number of zero crossings. The variable step solvers are trying to hit the exact zero-crossing occurrence in time. Therefore the step size is reduced to overly fine-grained size. Simulation is then consuming vast compute resources and compute time is unnecessary long.

3.3 Ground Motion Simulation Model

The core of the presented system is modeled by a system of ordinary differential equations (ODE) - equations of motion (eqation (2.12), (2.13) and (2.14)) based on the Newtonian physics. The overwhelming majority of ODE do not have exact solution that can be expressed in terms of simple functions. For this reason, one must rely on numerical methods that produce approximations to the desired solutions as described in previous sections.

The numeric integration in Simulink® is implemented in the Integrator block. The Integrator block outputs the value of the integral of its input signal with respect to time. Simulink® uses numerical approximation methods to evaluate ODE with finite precision. Simulink® can use a number of different numerical integration methods to compute the Integrator block's output, each with advantages in particular applications. The selected numerical solver computes the output of the Integrator block at the current time step, using the current input value and the value of the state at the previous time step. To support this computational model, the Integrator block saves its output at the current time

step for use by the solver to compute its output at the next time step. The Integrator block also provides the solver with an initial condition for use in computing the block's initial state at the beginning of a simulation [8].

3.3.1 Equations of Motion

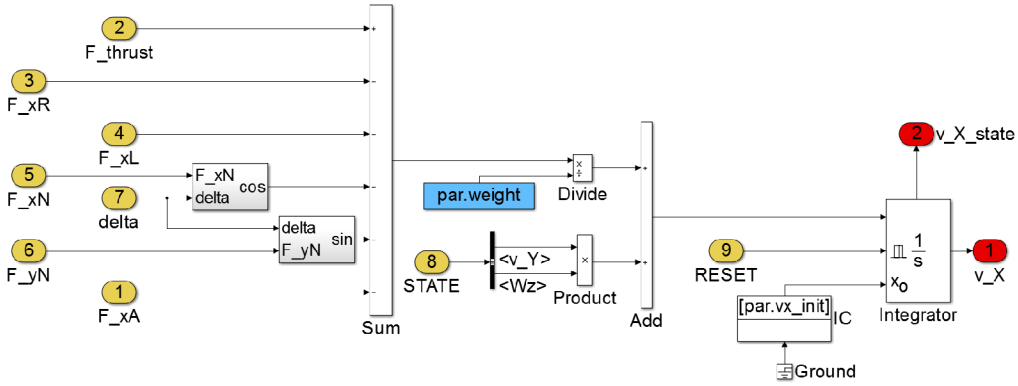


Figure 3.4: Simulink® model of v_x velocity component representing the first Equation of Motion.

The Figure 3.4 represents implementation of the first Equation of Motion (2.12) with the Integrator block in the rightmost part of the figure. Input to the Integrator is the sum of force components in x_B axis (BFF reference frame) acting on the aircraft. These force contributions are integrated over time where the output is component of velocity in x_B axis of BFF reference frame. There is also input port that resets the integrator and the port with initial condition that is used at the beginning of simulation when no previous value of integrator state is available. Initial value is defined in Matlab script, that is executed before the simulation to initialize the whole simulation model by providing values of parameters such as initial conditions for integrators (position, heading, velocity), characteristics of aircraft (dimensions, load, tire pressure) and environment conditions influencing the surface contamination of the runway surface.

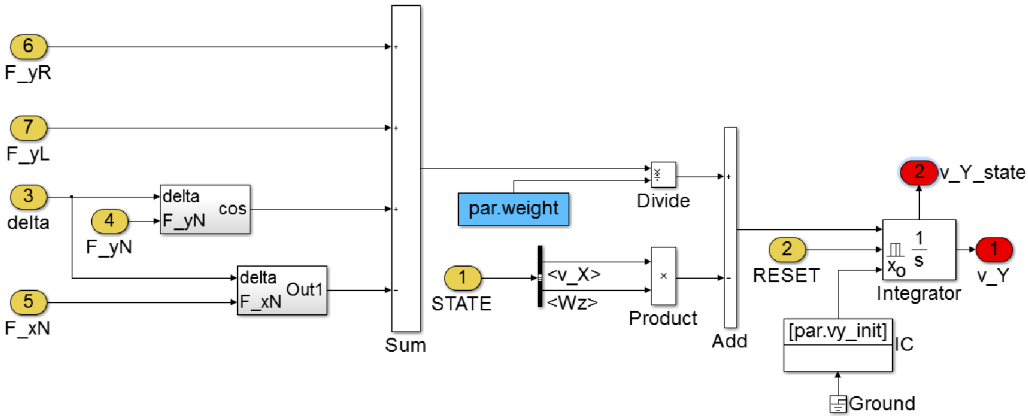


Figure 3.5: Simulink® model of v_y velocity component in terms of numerical integration.

The Integrator block's state port, labeled as output port `v_X_state` (in Figure 3.4), enables us to avoid an algebraic loop when creating an integrator that resets itself based on the value of its output. The output of the state port is the same as the output of the block's standard output port except for the case when the block is reset in the current time step. The output of the state port is the value that would have appeared at the block's standard output if the block had not been reset. This feature is introduced to the presented model in order to suppress the velocity of negative amplitudes. The longitudinal forces are defined as a function of vertical forces that are present at all times, meaning that aircraft would start moving even if no thrust is being applied. Self-reseting integrator is used in order to model the static friction, that allows aircraft motion only if forward force reaches certain threshold. After this point, regular kinetic friction coefficients are used. These coefficients are described in chapter 2.

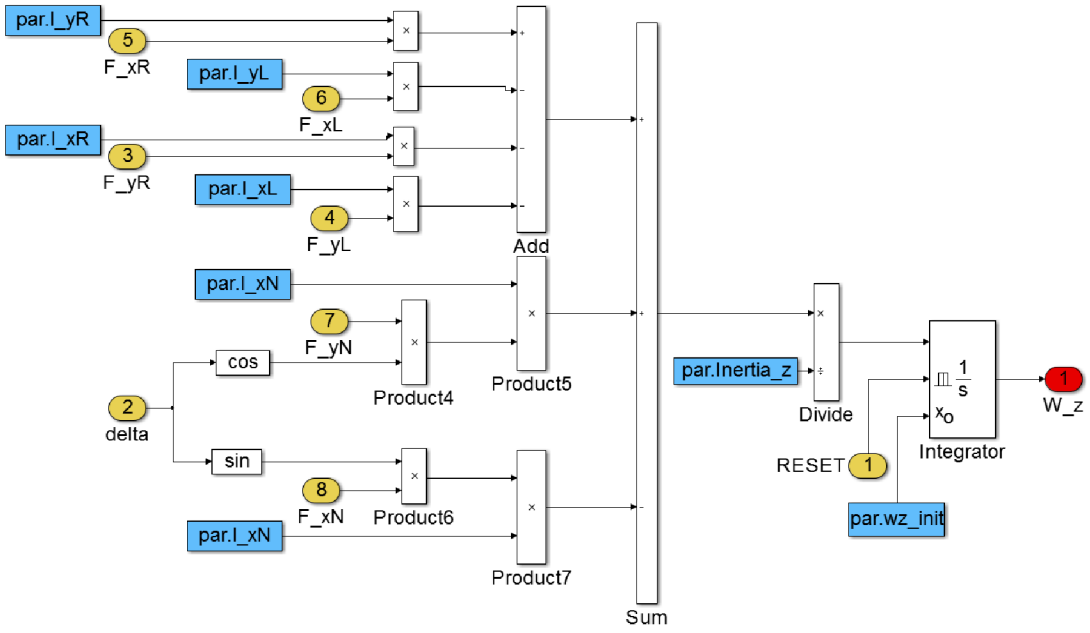


Figure 3.6: Simulink® model of the angular velocity ω_z around z_B axis.

The remaining two equations of motion are modeled in the same manner, as can be seen in Figure 3.5 representing integration of force components in in y_B axis into the component v_y of velocity vector. Figure 3.6 represents third equation of motion, which models the angular velocity ω_z about vertical axis z_B of the BFF reference frame based on the force moments acting on the aircraft.

Individual force components are connected to the blocks encapsulating Equations of Motion (see Figure 3.7). The Equations of Motion are implemented in blocks *Velocity*, where velocity vector components (v_x, v_y) are computed and block *Angular Velocity* that is computing ω_z.

3.3.2 Tire Forces Model

Simulink® model of the tire forces is decomposed into three sub-models including model of vertical forces, longitudinal forces and lateral forces. Vertical forces directly influence the amplitude of longitudinal and lateral forces. After longitudinal and lateral forces are com-

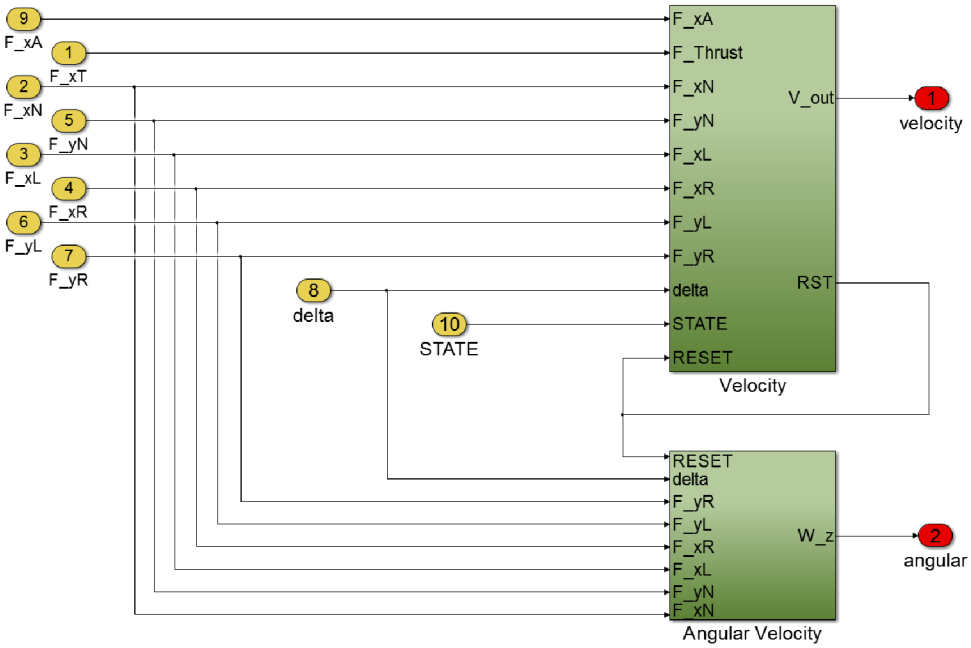


Figure 3.7: Overall Simulink® design scheme of the Equations of Motion.

puted at the current time-step, the signals representing forces are then connected to input ports of blocks modeling previously described equations of motion (Figure 3.7). In other words, the force interactions are transformed into motion characteristics of the airplane.

Vertical forces are computed at first. Vertical forces are formed as a response to gravitational force and represents the load distribution of aircraft weight to individual struts and corresponding wheels and its tires. The distribution of aircraft load is represented by equation (2.20) and its model is shown in Figure 3.8. The *Solver* block computes the algebraic equation derivable from equation (2.20) resulting into the vertical force acting on the main strut of the aircraft, which is equal for both left and right strut. This force is than used to compute vertical force acting on aircraft's front nose-wheel strut.

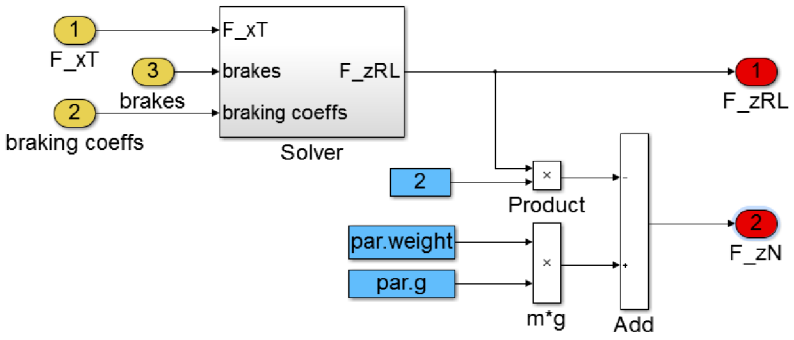


Figure 3.8: Simulink® model of load balancing defining the vertical forces.

Model of longitudinal forces consists of the rolling friction and the effective braking friction in case the brakes are being applied. Braking, of course, applies only to main wheels. Both rolling friction and effective braking coefficients depend on various factors, including the aircraft velocity and the environmental conditions. Therefore these coefficients cannot be static constants defined in the model but rather dynamically changing variables. These varying coefficients are computed in a different block, as described in section 3.3.4. According to the modular design concept, one can abstract these coefficients as constants provided by signal bus *coeffs*. In fact, the signal bus just encapsulates output of the block that dynamically computes the varying friction and braking coefficients.

The definitive longitudinal forces are computed by summing the retarding rolling friction efforts and braking forces (Figure 3.9) as defined by Equation (2.22).

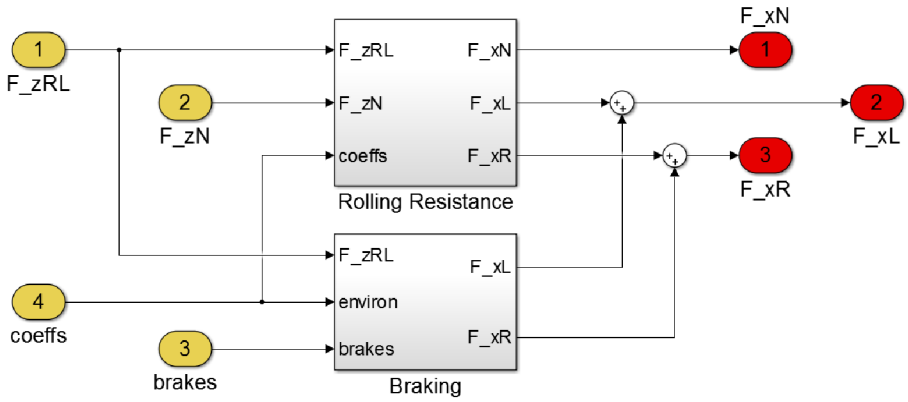


Figure 3.9: Simulink® model of the longitudinal tire forces F_{x*}.

Model of lateral forces with emphasis on high-detailed modeling is an essential requirement for a high-precision simulation. It represents the maneuvering abilities and characteristics of an aircraft in various environmental and operational conditions. It determines the forces formed in the tire in situations when the tire is yawed relative to its velocity vector, which occurs during the aircraft turns.

The model of lateral forces considers and includes various factors, namely the environmental conditions of the runway surface, operational characteristics of an aircraft including tire load and velocity and reduced lateral friction when wheel brakes are being applied. All these factors constitute the final value of the lateral friction coefficient μ_ψ, which determines the lateral force in a given situation. Simulink® model of computing the lateral friction coefficient μ_ψ and consecutively lateral force F_y, is shown in Figure 3.10. The underlying mathematical model is described in section 2.2.4.

The lateral friction coefficients has different characteristics in case of a small and large tire yaw angle. The model is designed so that it computes lateral friction coefficient for both situations and selects appropriate alternative based on the actual tire yaw angle according to equation (2.38). This structure is implemented in Simulink® by a block evaluating the equation (2.43) labeled as `PSI_limit`, condition block and the Multiport Switch block that chooses among two inputs - small yaw angle and large yaw angle. The first input to Multiport Switch block is the control input, while the others are data inputs. The value of the control input determines which input data passes to the output port.

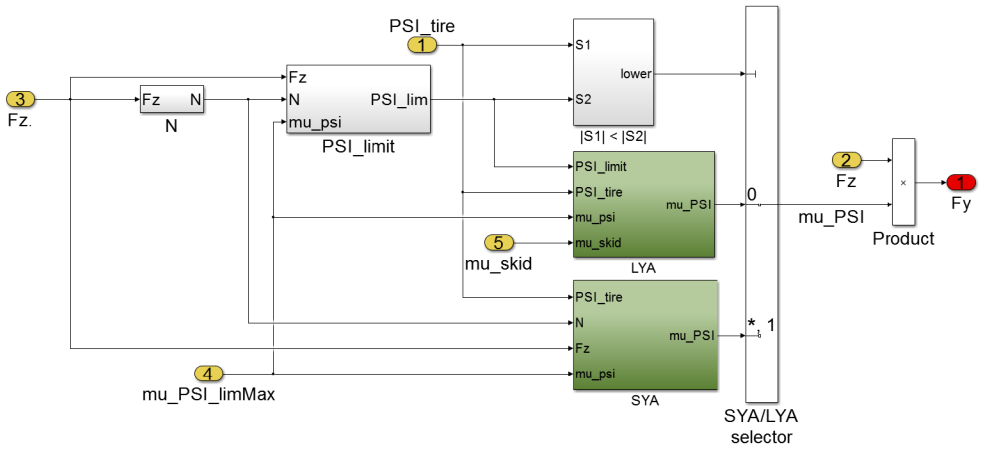


Figure 3.10: Simulink® model of the lateral friction coefficient μ_ψ and the lateral tire force F_y.

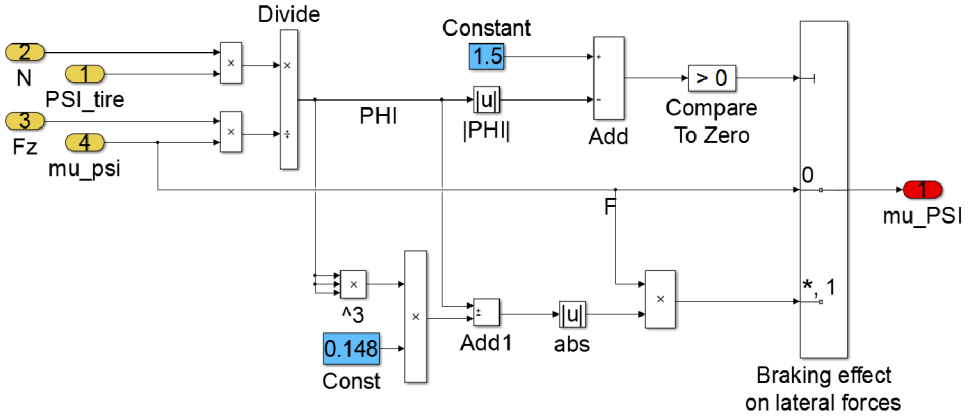


Figure 3.11: Simulink® model of the lateral friction coefficient for a small tire yaw angle μ_{sya}.

Lateral friction coefficient in case of a small tire yaw angle is a function of tire characteristics N, vertical tire load F_z, maximum lateral friction $\mu_{\psi_{max}}$ and actual tire yaw angle μ. There is an introduced nonlinearity based on the value of ϕ (equation 2.40) which switch between two values of lateral friction coefficient μ_{sya} according to the equation (2.39). The Simulink® block scheme is in Figure 3.11.

The value of the lateral friction coefficient is increasing with the increasing tire yaw angle, up to a certain tipping point ψ_{lim} defined by the equation (2.43). After reaching this point, lateral friction coefficient starts decreasing with an increasing tire yaw angle. Subsidiary functions i and j (equation 2.46 and 2.45 respectively) are used to model the lateral friction coefficient for large tire yaw angle. The value of lateral friction coefficient further depends on relation between $\mu_{\psi_{max}}$ and μ_{skid} as defined in equation (2.44). To provide an correct alternative according to aforementioned equation, we incorporate *If Action Subsystem* in the Simulink® model (Figure 3.12). Action subsystems are subsystems that execute in response to a conditional output from an *If* block. In essence, they are subsystems with an *Action port*, which allow for block execution based on conditional inputs from an *If* block. If action block is not active its output is zero. Therefore, simple

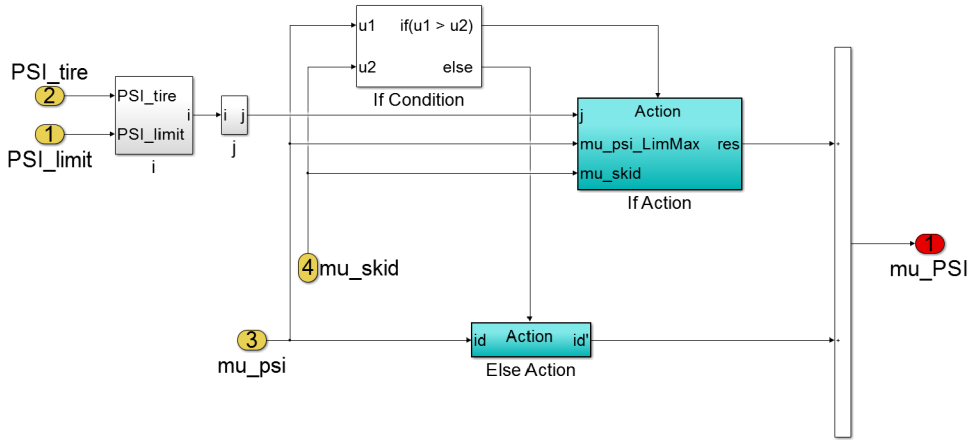


Figure 3.12: Simulink® model of the lateral friction coefficient for a large tire yaw angle μ_{lya}.

addition block is used in order to merge two signals from *If* and *Else* action blocks into single continuous signal.

3.3.3 Propulsion Model

Model of an engine is composed of an abstraction of multiple control signals thus allowing simple second order model. Complex propulsion model is not required since most of the engine control signals have very limited or fixed operational range in case of a ground motion. Under these assumptions, one can easily derive an equivalent second-order model, considering throttle as input signal and engine generated force as an output.

The behavior of the engine is modeled by a second order transfer function (equation (2.53)) implemented by the `Transfer Fcn` block in Simulink®. The `Transfer Fcn` block models a linear system by a transfer function of the Laplace-domain variable s. The numerator coefficient represents the maximum force generated by an engine at fully opened throttle (throttle actuator is at maximal position) and denominator specifies the response of a engine to change in the throttle position, that is simulating behavior of the engine. Small constant is added, which represents thrust of the engine generated when the throttle actuator is in minimal position. The output of transfer function would stabilize at zero after some settle time, however in reality engine generates some small force even if throttle is in this minimal position.

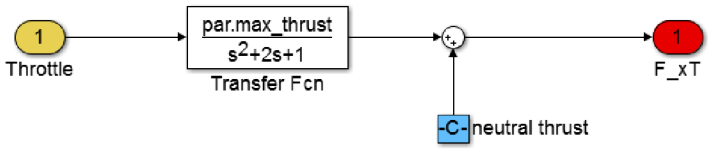


Figure 3.13: Simulink® model of the jet engine.

3.3.4 Environment Model

The characteristics of the aircraft ground motion are greatly influenced by runway surface conditions. The tire interacts differently with runway surface in wet, dry or snow contaminated conditions. In order to provide model with high-precision modeling capabilities one needs to consider these environment factors.

In the presented model, three different environment conditions of concrete runway are considered. Namely: standard dry conditions, wet runway surface and snow/ice contaminated runway surface. These conditions provide a wide operational scale for the simulation and suffice most of the scenarios required for an AutoTaxi simulations. The specific environment is selected in a configuration file and does not change throughout the simulation. The simulation is intended to be used for an individual taxi trials simulations that does not span long temporal horizons, therefore it is highly probable that the environmental conditions will remain constant. The scheme of environment model is shown in Figure 3.14. Modular design of the model allows easy implementation with more environmental models, such as adding natural surfaces including grass, dirt or gravel.

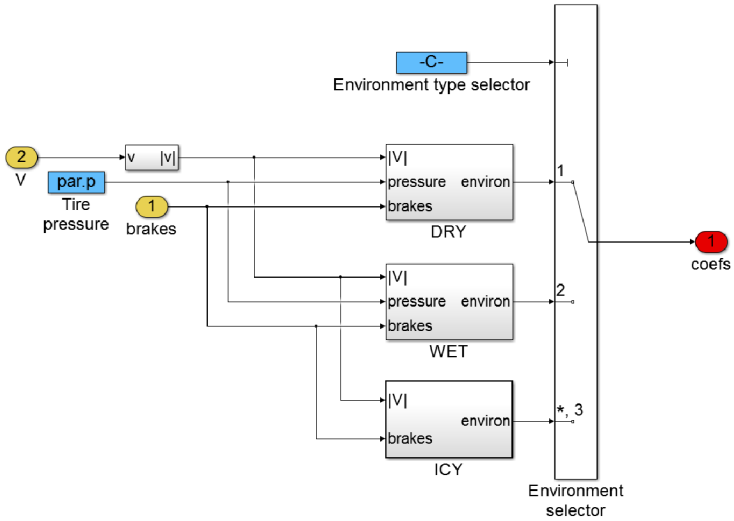


Figure 3.14: Simulink® model of the environment.

Each environment is implemented as an individual module where the output of each module is a bus consisting of the environment specific coefficients. Some coefficients are also influenced by the actual state of the aircraft, such as its speed, tire pressure or application of brakes. These factors are input signals to the corresponding blocks. The structure of block representing standard dry environment contains sub-blocks that compute the rolling friction coefficient; coefficients associated with the braking performance - maximum braking coefficient and the efficient braking coefficient considering performance of ABS; sliding friction in case of fully skid wheel; and finally the coefficients associated with the lateral forces with or without braking.

4. AutoTaxi Controller Design

In previous chapters we have defined the mathematical model of an aircraft ground motion and described its transformation to a simulation model implemented using MATLAB Simulink® environment with a focus on various aspect of the numerical analysis. This chapter will consider the aircraft ground motion model as a dynamic system (in control theory often referred to as plant) with its inputs, outputs and feedback loop that will control the aircraft's behavior. The controller module will provide steering signals to the aircraft based on some reference values that describe the desired behavior of the aircraft, namely the desired velocity and target trajectory and the final destination.

4.1 System Definition

Behavior of the aircraft is modeled as dynamical system. In general, system is any set of elements connected together by the signal links with defined spatial and temporal boundaries. System S is formally defined as a pair:

$$S = (U, R), \tag{4.1}$$

where:

- universum U is finite set of system elements $U = \{u_1, u_2, \ldots, u_n\}$,
- System element u is a pair (X, Y) where:
  - X is set of all input ports (variables),
  - Y is set of all output ports (variables),
- system characteristic R is set of all interconnections

$$R = \bigcup_{i,j=1}^{n} R_{ij},$$

  where $R_{ij} \subseteq Y_i \times X_j$ denotes interconnection of element u_i with element u_j [26].

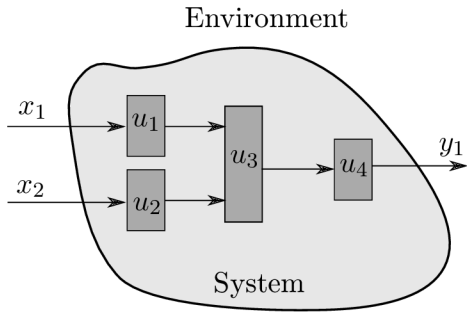


Figure 4.1: Illustration of an generic system consisting of four interconnected elements $U = \{u_1, u_2, u_3, u_4\}$ with two global system inputs x_1, x_2 and one output y_1.

4.2 Control Theory

The control theory deals with the analysis of the dynamic systems and methodologies to construct controllers. The main objective of the control theory is to control a system, so its output follows a desired control signal, called the reference signal, which may be fixed or changing in nature. The controller monitors the output and compares it with the reference signal. The difference between the actual and the desired output, called the error signal, is applied as a feedback to the input of the system, in order to bring the actual output closer to the reference [21].

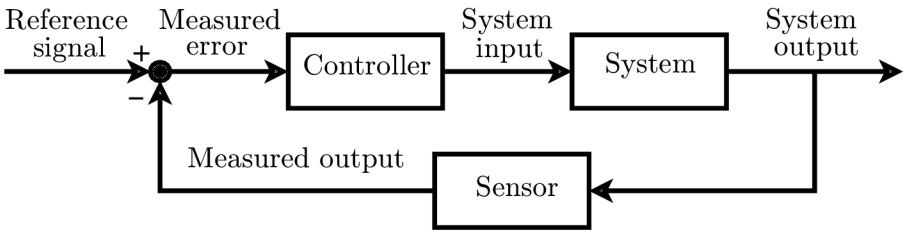


Figure 4.2: The concept of the feedback loop to control the behavior of the dynamic system.

A dynamic system is a system that changes its states over time. A mathematical model that describes the swinging of a clock pendulum or the ground motion model of an aircraft are both examples of a dynamic systems. Secondly, control theory is most interested and very powerful when dealing with linear systems. A linear system is a model of a system based on the utlization of a linear operator, that always maps linear subspace onto another linear subspace, which basically means that the output is proportional to the input. A major subclass of linear systems are linear time invariant (LTI) systems, which in addition have parameters which do not change over time [21].

4.2.1 PID Controller

A proportional-integral-derivative controller is a control loop feedback mechanism widely used in the industrial control systems. A PID controller calculates an error value as a difference between the measured system output and the desired system output. The controller attempts to minimize the error by adjusting the system behavior through the use of a system input variable.

The PID control scheme is named after its three parameters: the proportional, integral, and derivative terms. These terms are summed to calculate the output of the PID controller. Defining $u(t)$ as the controller output and $e(t)$ as the input error signal, the final form of the PID algorithm is:

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau)d\tau + K_d \frac{d}{dt} e(t). \tag{4.2}$$

With its three-term functionality covering treatment to both transient and steady-state responses, PID control offers the simplest and yet most efficient solution to many real-world control problems. With advances in digital technology, the science of automatic control now offers a wide spectrum of choices for control schemes. However, more than 90% of industrial controllers are still implemented based around PID algorithms, particularly at lowest levels [4].

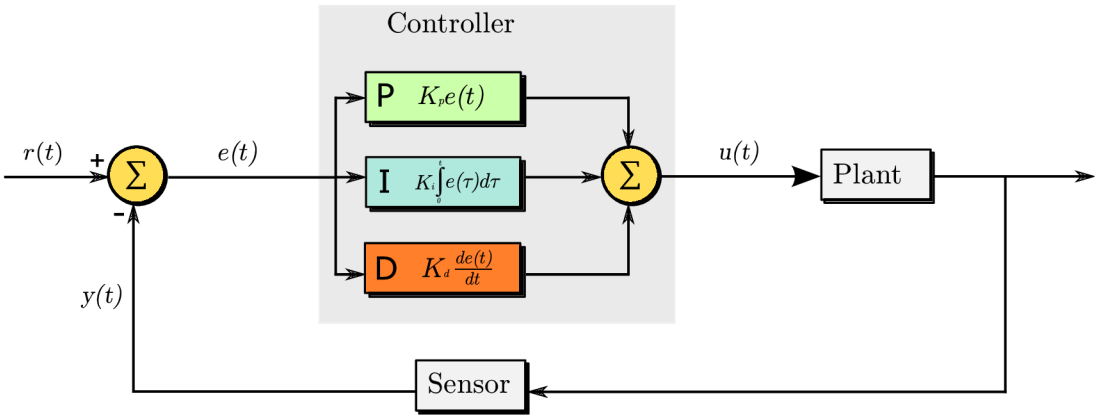


Figure 4.3: PID feedback loop controller.

Proportional term produces an output value that is proportional to the current error value. The proportional response can be adjusted by multiplying the error by a proportional gain K_p. A high proportional gain results in a large change in the output for a given change in the error. If the proportional gain is too high, the system can become unstable, which basically means that output of the controlled process will diverge. In contrast, a small gain results in a small output response to a large input error, and a less responsive or less sensitive controller. If the proportional gain is too low, the control action may be too small when responding to system disturbances.

Integral term contribution to controller output is proportional to both the magnitude of the error and the duration of the error. The integral term in a PID controller is the sum of the instantaneous error over time and gives the accumulated offset that should have been corrected previously. The accumulated error is then multiplied by the integral gain K_i and added to the controller output. The integral term accelerates the motion of the process towards desired target value and eliminates the residual steady-state error that occurs with a pure proportional controller. However, since the integral term responds to accumulated errors from the past, it can cause the present value to overshoot the target value.

Derivate term of the system error is calculated by determining the slope of the error over time and multiplying this rate of change by the derivative gain K_d. Derivative action predicts system behavior and thus improves settling time and stability of the system. An ideal derivative is not causal, so that implementations of PID controllers include an additional low pass filtering for the derivative term, to limit the high frequency gain and noise. In fact, derivative action is seldom used in practice because of its variable impact on system stability in real-world applications [4].

4.3 Controller Implementation

The design and the implementation of the aircraft ground motion was discussed in the previous chapters. Now, we can abstract the aircraft model as a dynamic system (in control theory referred to as plant) with its inputs, outputs and create a feedback loop that will control the aircraft behavior. The controller module will provide steering signals to the aircraft based on the reference values that describe the desired behavior of the aircraft, namely desired velocity and a target trajectory.

The general scheme of the controller module is shown in Figure 4.4. The input signals to the controller module are the current position and velocity vector. Position vector $Pos = [X, Y, \Psi]$ contains coordinates of the aircraft on two-dimensional plane and also the heading angle Ψ. Velocity vector $Velo = [v_x, v_y]$ contains velocity components in the main axes of the BFF coordinate system. Velocity vector is specified in BFF reference frame, while position vector is defined in earth-fixed Navigational reference frame.

The controller module further contains three submodules, where each of theses submodules provides control of the specific aspects of the aircraft ground motion. The *Direction control* provides steering signal the for nose-wheel, *Velocity control* provides control signal for the throttle and finally *Brakes control* provides braking signals for each wheel of the main landing gear.

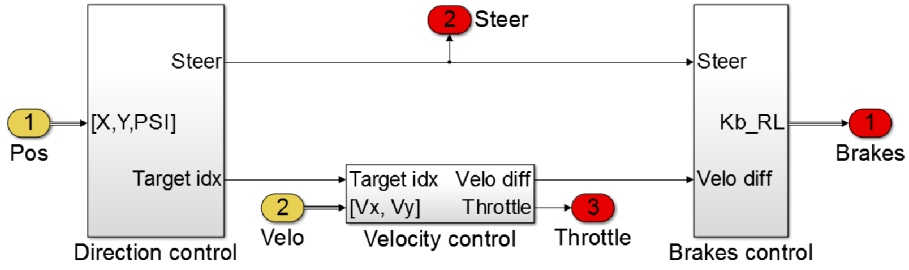


Figure 4.4: Scheme of the AutoTaxi controller.

4.3.1 Directional Control

The objective for the directional control is to move the aircraft in a direction of the desired heading. The control variable is a steering angle and the controlled variable is the heading (yaw), where the control signal is provided by two PID controllers that modify steering angle based on the error of current heading with respect to desired heading and displacement of the aircraft from the optimal trajectory. The controllers design is discussed below.

Taxiways on the airports use special marking for the centerlines (Figure 4.5), which are used as a reference trajectory for the nose-wheel steering by pilots during manual taxiing

Figure 4.5: Taxi-lines marking on the Vaclav Havel International Airport in Prague.

maneuvers. We use these centerlines as a reference trajectory for the evaluation of the controller abilities to maneuver the aircraft and to follow the predefined trajectory. We approximate the taxi-lines by series of GPS waypoints with a spatial separation up to 10 meters. Straight segments are sufficiently approximated by two waypoints, marking the beginning and the end of the segment, while turns need more fine-grained approximation (see Figure 4.6). A GPS waypoint is an absolute location of a point on the Earth's surface, defined by the latitude and longitude in degrees which, when combined with height specify its spherical coordinates. In order to simplify geometric calculations of the distances and angles in the waypoint following algorithm, the waypoints are converted to a Cartesian system.

The desired heading Ψ_{des} indicates heading to a target waypoint. It is computed relative to the current aircraft position as:

$$\Psi_{des} = \tan^{-1}\left(\frac{Y_{wpt} - Y}{X_{wpt} - X}\right). \tag{4.3}$$

The heading error is then defined as a difference between the desired heading and a current heading of the aircraft:

$$\Delta_{\Psi} = \Psi_{des} - \Psi. \tag{4.4}$$

Under certain circumstances, the magnitude of the heading error can be larger than 180°. This means that the aircraft will turn the „long way“ towards the desired waypoint which is behind it. To avoid this, 360°is subtracted from the heading error if it is greater than 180°, and 360°is added to the heading error if it is less than -180°. A negative heading error requires a left turn to compensate, and a positive heading error requires a right turn to compensate.

The direction controller can be easily used to implement the waypoint following algorithm. The goal of the waypoint following algorithm is to control and steer the aircraft along the trajectory approximated by a series of individual GPS waypoints. As the aircraft

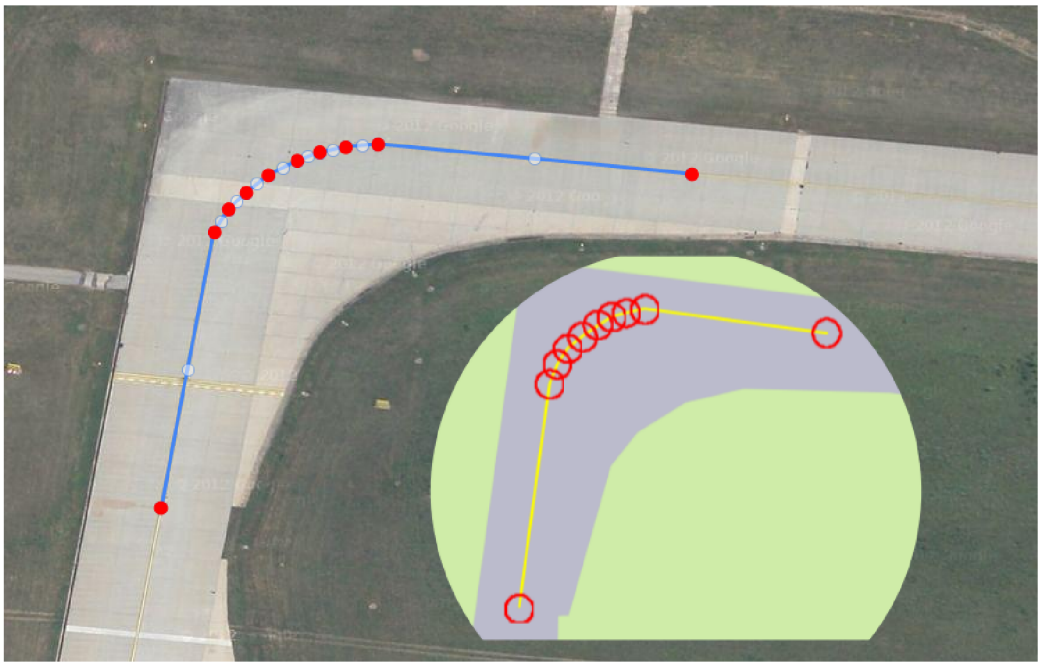


Figure 4.6: Approximation of the taxi-way centerline at Brno International Airport with visualized safety margins around waypoints.

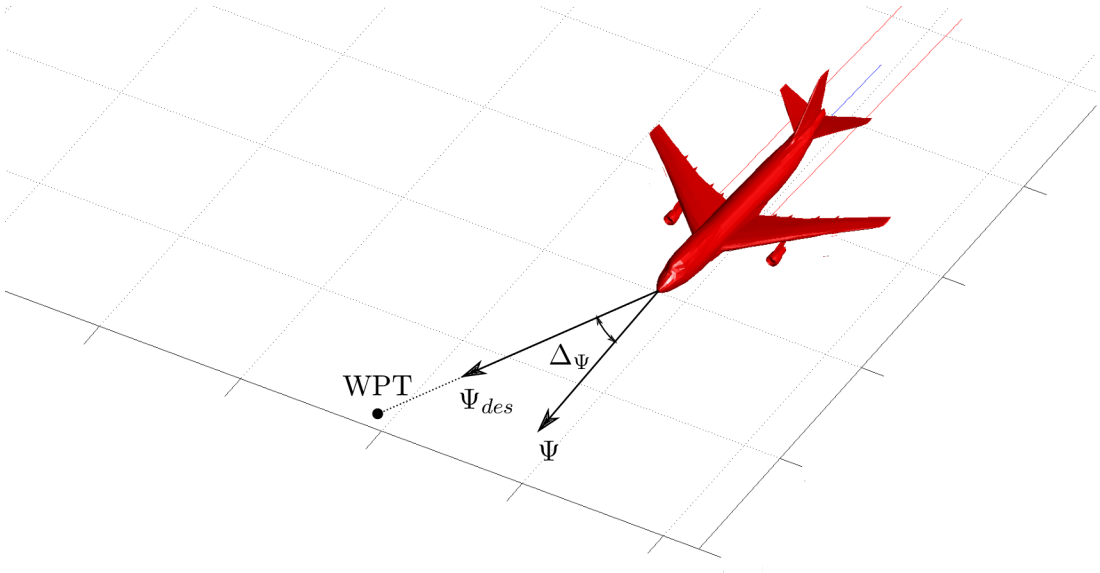


Figure 4.7: Desired heading to the target Ψ_{des} and heading error Δ_Ψ.

reaches certain waypoint, the waypoint following algorithm chooses the next waypoint from the sequence as a destination to reach next.

This approach works well provided that the aircraft always reaches the target with a perfect precision. However, if the aircraft approaches the waypoint at a slight offset, the desired heading can quickly change from couple of degrees to 80°and more. This causes the aircraft to lurch to the side before reaching the waypoint. This is due primarily to error in

the GPS receiver. A simple and robust approach to correct this problem involves defining a circle of specified radius around the waypoint. As soon as the aircraft enters the circle, the waypoint is considered to be achieved. Determined through experimentation, the radius of this circle was set to 3 meters (see Figure 4.6).

Lateral displacement can also be minimized by a direction controller. The lateral displacement of the aircraft is defined as an offset e between the aircraft position P and the straight line path between two consecutive waypoints P_1 and P_2 (Figure 4.8). The control algorithm uses lateral displacement from the optimal path to guide the aircraft back to the path where the trajectory is assumed to be most desirable. The lateral displacement occurs mainly because of the reduced capabilities of the aircraft to develop rapid steering response in case of sharp turns. As a result, the aircraft overshoots the turn. The lateral displacement is then corrected by navigating the aircraft back to desired centerline trajectory.

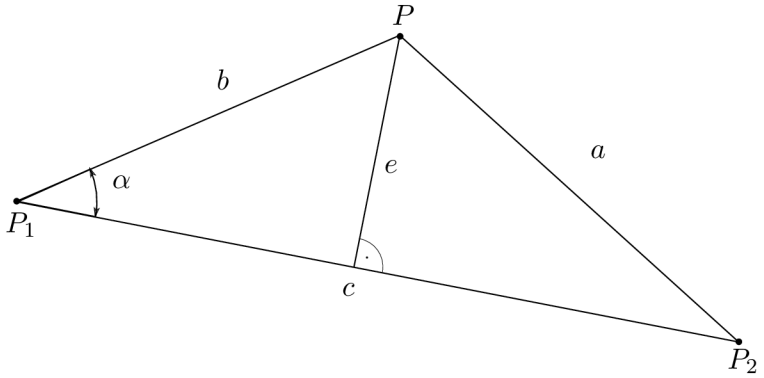


Figure 4.8: Geometry of the lateral displacement calculations.

The coordinates of the waypoints are well-known, as is the position of the aircraft. Therefore one can easily estimate the distances between these points forming a triangle in Figure 4.8. From law of cosines we can compute α and consecutively the lateral displacement e:

$$\alpha = \cos^{-1}\left(\frac{b^2 + c^2 - a^2}{2bc}\right). \tag{4.5}$$

$$e = b\sin(\alpha). \tag{4.6}$$

Equations (4.5) and (4.6) gives us error magnitude but no information about the relative position of aircraft and optimal trajectory - whether the aircraft is on left side or right side. We use cross product of two vector to determine this information. The vectors are vector from P_1 to P_2 and the vector from P_1 to aircraft position P. The cross product is given by equation (4.7).

$$k = sign((x_{P_2} - x_{P_1})(y_P - y_{P_1}) - (x_P - x_{P_1})(y_{P_2} - y_{P_1})). \tag{4.7}$$

If the result is positive the aircraft is to the left of the optimal trajectory, the lateral displacement is positive and the aircraft must turn right in order to compensate for displacement error. The inverse rules applies for negative result of the cross product.

If we want to fuse heading and lateral displacement control signals, we need to address special situation when the heading error is greater than 90 °. If the aircraft is pointed away from the desired waypoint (heading error is greater than 90°) the heading error can be of

the opposite sign than the lateral displacement error. If the errors are similar in magnitude, they will cancel out and the aircraft will not reach the target because it will continue driving straight. This situation occurs, for example, when aircraft reaches a waypoint with the next desire waypoint behind reached one.

To correct this situation, the lateral displacement feedback is multiplied by a scaling factor to limit its effect when the heading error magnitude is large. The scaling factor is specified in equation (4.8) [22].

$$\begin{aligned} multiplier &= 0.72 * \Delta_\Psi - 1.13, && 10^\circ < \Delta_\Psi < 80^\circ, \\ multiplier &= 0, && \Delta_\Psi > 80^\circ, \\ multiplier &= 1, && \Delta_\Psi < 10^\circ. \end{aligned} \tag{4.8}$$

Fusing heading and lateral displacement control is achieved by multiplying heading error Δ_Ψ and lateral displacement error e by respective PID gains and adding these two together to determine the steering angle of the aircraft. Because the heading error is in radians, its maximum value is π. The lateral displacement error value is in meters, and is typically much higher, thus it requires much lower gains to scale properly. The formula is expressed by equation (4.9) and the Simulink® scheme is in Figure 4.9.

$$\begin{aligned} Steerangle = {} & \Delta_\Psi \cdot K_{P_H} + ke \cdot K_{P_P} \\ & + \frac{d}{dt}\Delta_\Psi \cdot K_{D_H} + \frac{d}{dt}ke \cdot K_{D_P} \\ & + \int \Delta_\Psi \, dt \cdot K_{I_H} + \int ke \, dt \cdot K_{I_P}. \end{aligned} \tag{4.9}$$

The gains are:

$$\begin{aligned} K_{P_H} &= 1.000 & K_{P_P} &= 0.030 \\ K_{D_H} &= 0.050 & K_{D_P} &= 0.008 \\ K_{I_H} &= 0.001 & K_{I_P} &= 0.001. \end{aligned}$$

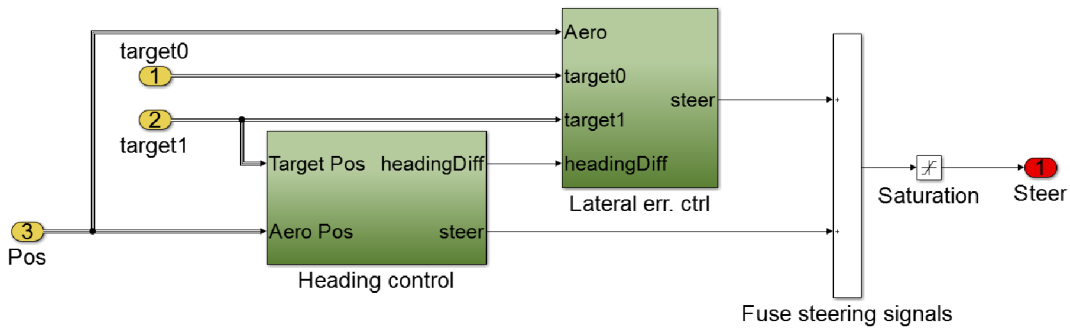


Figure 4.9: Simulink® scheme of the direction control.

Controller response determines the behavior of the aircraft motion. Regarding the comfort and safety of passengers we want out controller to perform a smooth steering

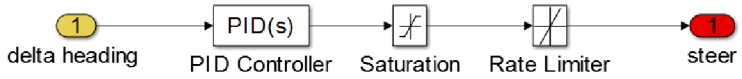


Figure 4.10: Simulink® scheme of heading sub-controller module.

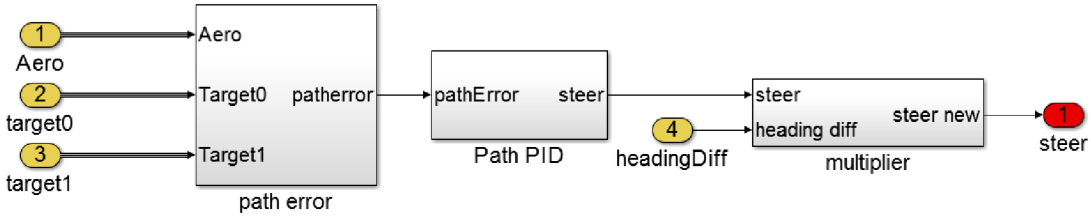


Figure 4.11: Simulink® scheme of lateral displacement control.

changes but still vigorous enough in order to perform certain rapid steering maneuvers, i.e. small radius turns on narrow taxiways. The response of heading controller is tuned to inhibit any oscillations of an actual heading with respect to the desired heading. Such behavior would cause the aircraft to actually never go straight but would perform alternate left and right turning pattern, which is fairly undesirable. This behavioral pattern is mostly caused by the integral term of the PID controller. Therefore this term needs to be minimized to such extent that this behavior does not occur, but controller is still able to eliminate constant factor errors. The PID controller response basically follows the input error signal (Figure 4.12) except for the discrete step changes of a desired heading which happens when destination waypoint is switched to the next one as previous waypoint was reached. Heading controller in this case reacts with a smooth continuous change of a steering signal until desired value is reached.

The response of the controller based on the lateral displacement error should be even smoother. We want to minimize and correct for lateral error over longer time periods. Immediate correction of the lateral error would introduce oscillatory pattern along the optimal aircraft trajectory, which is undesirable. Short term lateral error is also introduced by the waypoint following algorithm. As the waypoints are switched discretely and we introduce safety offset around individual waypoints, aircraft finds itself off-track from the approximated trajectory by waypoints in a case when waypoints are not perfectly aligned. This short term error gets corrected as the aircraft continues its motion, no additional actions are needed. The response of the lateral displacement controller is therefore tuned to minimize the displacement error over longer time periods as can be seen on Figure 4.13b and 4.13a.

4.3.2 Velocity Control

The control system must also modulate the throttle and the brake to achieve a desired speed. In this case, overshoot is more important than rise time, because the vehicle cannot be permitted to overshoot a speed limit. We focus on a well damped response in case of the velocity controller. As seen in Figure 4.15 we achieve almost constant response for the velocity controller for a constant target velocity. The velocity error oscillates in case of sharper turns due to the imposed lateral friction and the application of differential braking

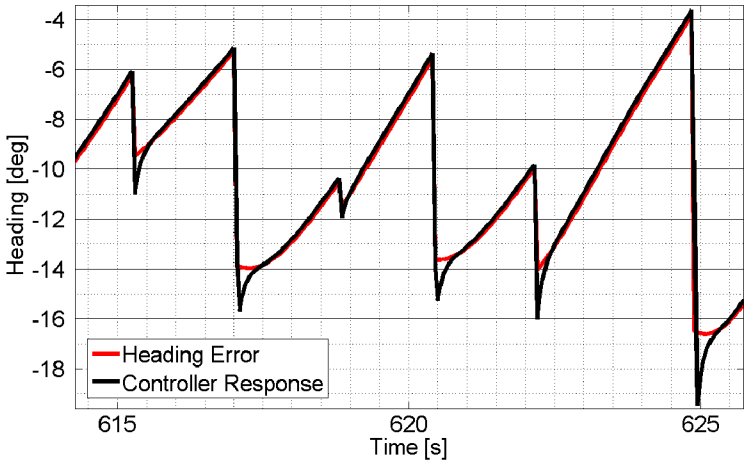


Figure 4.12: Heading controller response.

in case of sharper turns.

The PID scheme of the velocity control is expressed by the equation (4.10). The velocity error signal Δv is computed as a difference between the actual and the target velocity.

$$Throtlle = \Delta v \cdot K_{P_V} + \frac{d}{dt}\Delta v \cdot K_{D_V} + \int \Delta v \; dt \cdot K_{I_V}. \tag{4.10}$$

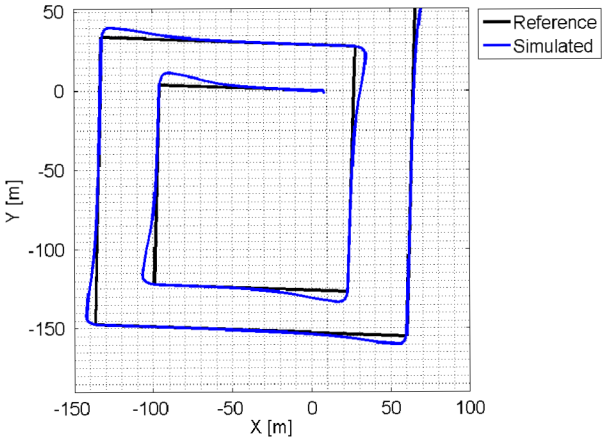
The gains are:

$$\begin{aligned} K_{P_V} &= 0.030 \\ K_{D_V} &= 0.002 \\ K_{I_V} &= 0.030. \end{aligned}$$

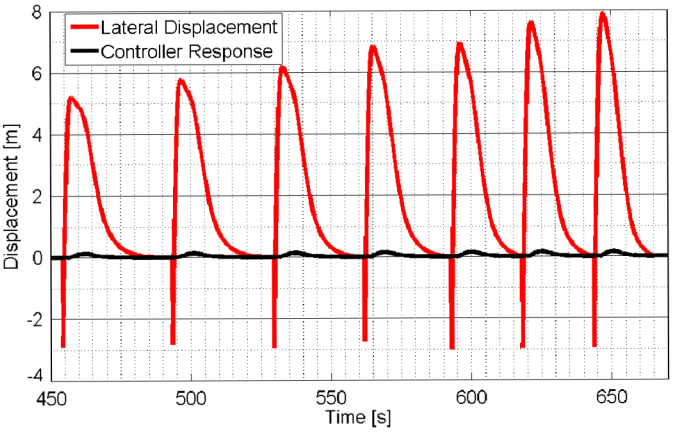
In the experiment in Figure 4.15 the aircraft starts taxiing from stationary position and accelerates to 5 m/s. During the first 150 seconds the aircraft performs a series of rapid turns using differential braking therefore we observe an increased error amplitude. Despite this behavior of the error signal, the response of the controller is rather smooth during the whole taxiing trial, which is the desired behavior of the velocity control loop.

4.3.3 Brakes Control

The last submodule of the controller module is the brakes control. Brakes are used to reduce velocity in case the aircraft needs to decelerate, usually in situations when the aircraft is about to stop. We also use brakes as means of controlling the aircraft steering. We deploy differential braking in certain situations, thus enhancing the maneuvering capabilities of the aircraft in sharper turns. Differential braking basically means activating brakes only on one wheel of the main landing gear. This slows down corresponding wheel while the opposite wheel continues moving at faster speed, which results in a rotational motion of the aircraft around the braked wheel. The general scheme of the brakes control is in Figure 4.16. We use two input signals, first of which is the steering angle, which determines whether the differential braking will be activated. The second signal is the difference of the velocity from the target velocity. In case of the actual velocity exceeding desired velocity, we activate brakes proportionally to the error magnitude. After computation of brake control signals,



(a) Aircraft's trajectory for the spiral test case.



(b) Controller response for lateral offset error.

Figure 4.13: Behavior of the controller for lateral displacement correction in a spiral benchmark.

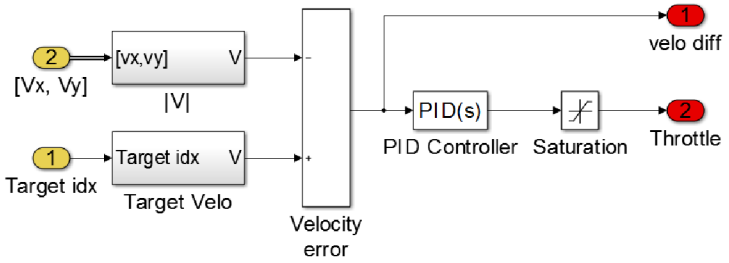


Figure 4.14: Simulink® model of the velocity controller.

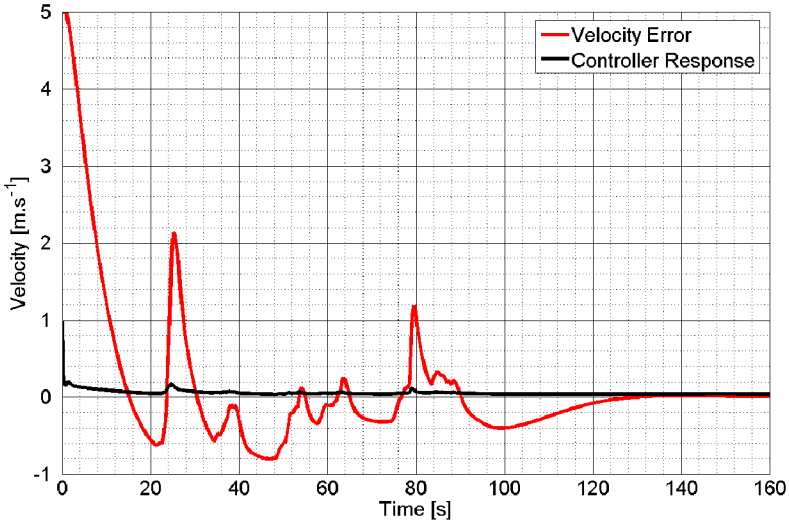


Figure 4.15: Controller response to the velocity error signal.

signals resulting from differential braking and velocity control are fused to form a single brake control signal for each wheel.

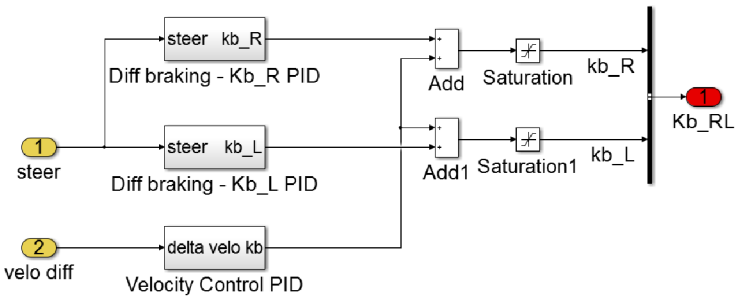


Figure 4.16: Simulink® model of the brake controller.

Differential braking and corresponding scheme is shown in Figure 4.17. As the AutoTaxi is designed to support differential braking, allowing for sharper turns in narrow areas, the steering angle is also fed to the brake controller. Differential braking is applied after steering angle reaches certain limit value, experimentally set to $\delta_{lim} = 0.8rad$. After reaching this point, the differential brakes are applied proportionally. The PID uses only

proportional term $K_{P_{kb}} = 10$. Figure 4.18 shows behavior of the differential braking during taxiing trial. As the steering angle reaches predefined limit value, the brakes are activated proportionally until steering angle falls below the threshold value.

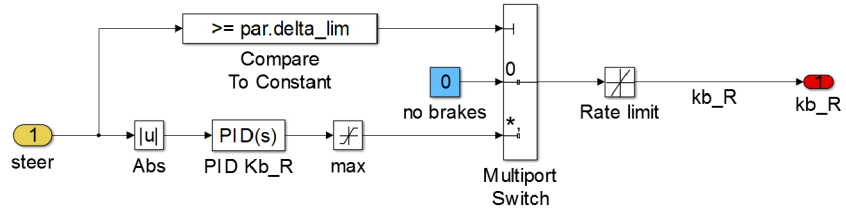


Figure 4.17: Simulink® model of differential brakes controller for the right wheel of the main landing gear.

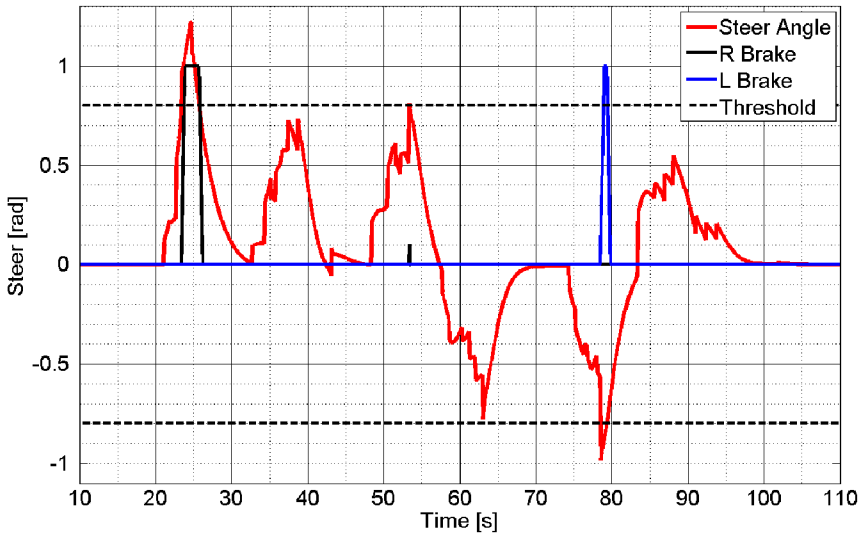


Figure 4.18: Response of the differential brake controller.

Velocity correction with activated brakes occurs in cases when actual aircraft velocity exceeds desired velocity over $1m \cdot s^{-1}$. Both main wheel brakes are applied proportionally to the velocity overshoot in order to decelerate the aircraft. The corresponding PID term is $K_{P_{kb}=0.2}$. The controller scheme is shown inFigure 4.19.

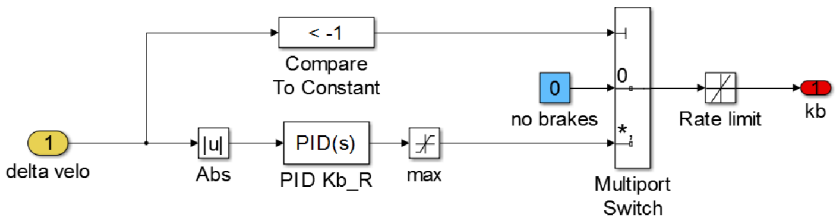


Figure 4.19: Simulink® model of brake controller for the velocity correction.

5. Experiments and Model Verification

This chapter provides the verification of the simulation model presented in chapter 3 based on the comparison with the analytical model of Ackerman steering. Experiments are also performed in order to evaluate the AutoTaxi controller module by navigating the aircraft through the simulated taxi-ways on multiple international airports.

5.1 Analytical Verification of the Turn Radius

The presented model is validated with respect to Ackermann tricycle analytical model. Ackermann geometry avoids the need for tires to slip sideways when following the path around a curve. However, in presented model, the tire slip is of primary focus. Therefore it cannot be expected that the results of the simulation match the exact analytical solution of Ackermann tricycle drive. Nevertheless, It can still provide nominal guidelines to the simulation correctness.

Ackermann model assumes that lines perpendicular to the wheel axles meet at one point, denoted as ICC (instantaneous center of curvature) when the vehicle is turning. As the rear wheels are fixed and cannot rotate as the fron steering wheel, this center point must be on a line extended from the rear wheel axles (Figure 5.2). Consequently, if the steering angle δ is fixed wheels are moving over circular trajectory with common center at ICC but with different radius. This radius can be analytically evaluated for each wheel, as well as for the C.G. of the vehicle. The turn radius for nose wheel with given steer angle δ is:

$$R = \frac{l_{xR} + l_{xN}}{\sin(\delta)} \tag{5.1}$$

The comparison is given in table 5.1 and error evaluation is given in table 5.2. Results are as expected, that is close to analytical solution wit relative error under 5%. The model assumes various non-linear characteristics of the ground motion and tire-runway interaction. Analytical model is solely based on geometrical analysis and does not consider such complex behavior. The exact match of the turn radii is therefore not expected. The model assumes various parameters influencing the behavior of aircraft on the ground resulting into highly non-linear model of the tire lateral forces. Among others, the model variables influencing the aircraft ground motion are: tire inflation pressure, runway surface contamination and tire velocity that directly influences the maximum lateral friction coefficient responsible for lateral force and thus the radius of the simulated turn. The experiment was performed considering dry concrete runway surface, aircraft velocity 5 $m \cdot s^{-1}$, tire inflated to recommended pressure 140 psi and 75% of maximum aircraft load.

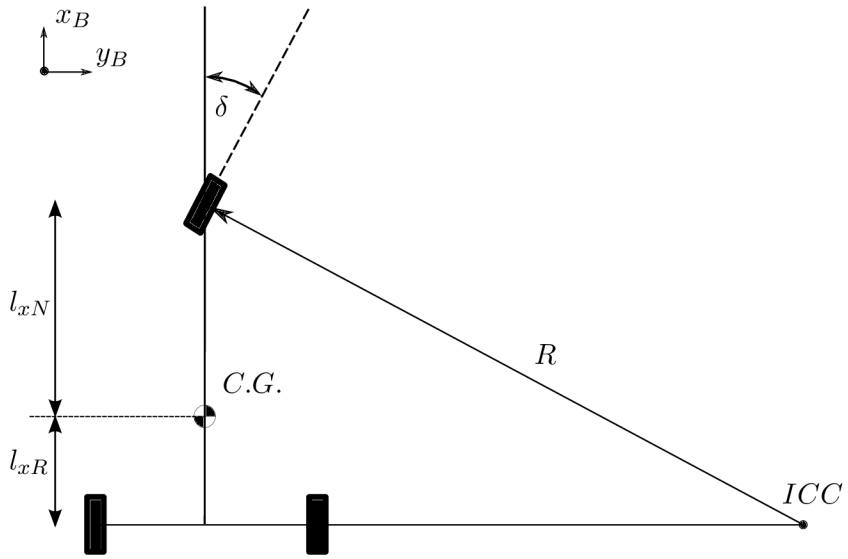


Figure 5.1: Ackermann tricycle steering model.

5.2 Boeing's Specification of Turn Radius

Simulation results and analytical solution of the nose wheel turn radii are also subjected to Boeing's guideline *Airplane Characteristics for Airport Planning* document and shown in Figure 5.2. Constant offset 0.3 m of turn radii between analytical solution and Boeing's specifications can be observed in Table 5.1. This suggests Boeing's analysis is based on the analytical solution of the Ackermann steering model with some safety margin included, considering linear tire slip. The presented model is able to dynamically compute the tire slip based on various environmental conditions and operational characteristics of the aircraft.

The simulation results suggest that the divergence of the turn radii from the Boeing's official document [6] is also below a 5% error, similar as comparison to the analytical solution. These results partially confirm the ability of the model to perform high-precision simulations. In order to evaluate the realism of presented simulation model it would be needed to compare simulation results with real data from the actual taxiing trials of the real aircraft. Unfortunately, such data are not available at the time of this publication.

Table 5.1: Comparison of simulated turn radius R with analytical solution of Ackermann steering and Boeing specifications radii.

Steer angle δ [deg]	Analytical R [m]	Boeing R [m] [6]	Simulation R [m]
30	28.5	28.8	30.0
35	24.9	25.2	25.8
40	22.2	22.5	22.6
45	20.2	20.5	20.3
50	18.6	18.9	18.4
55	17.4	17.7	16.9
60	16.5	16.8	16.1
65	15.7	16.1	15.8

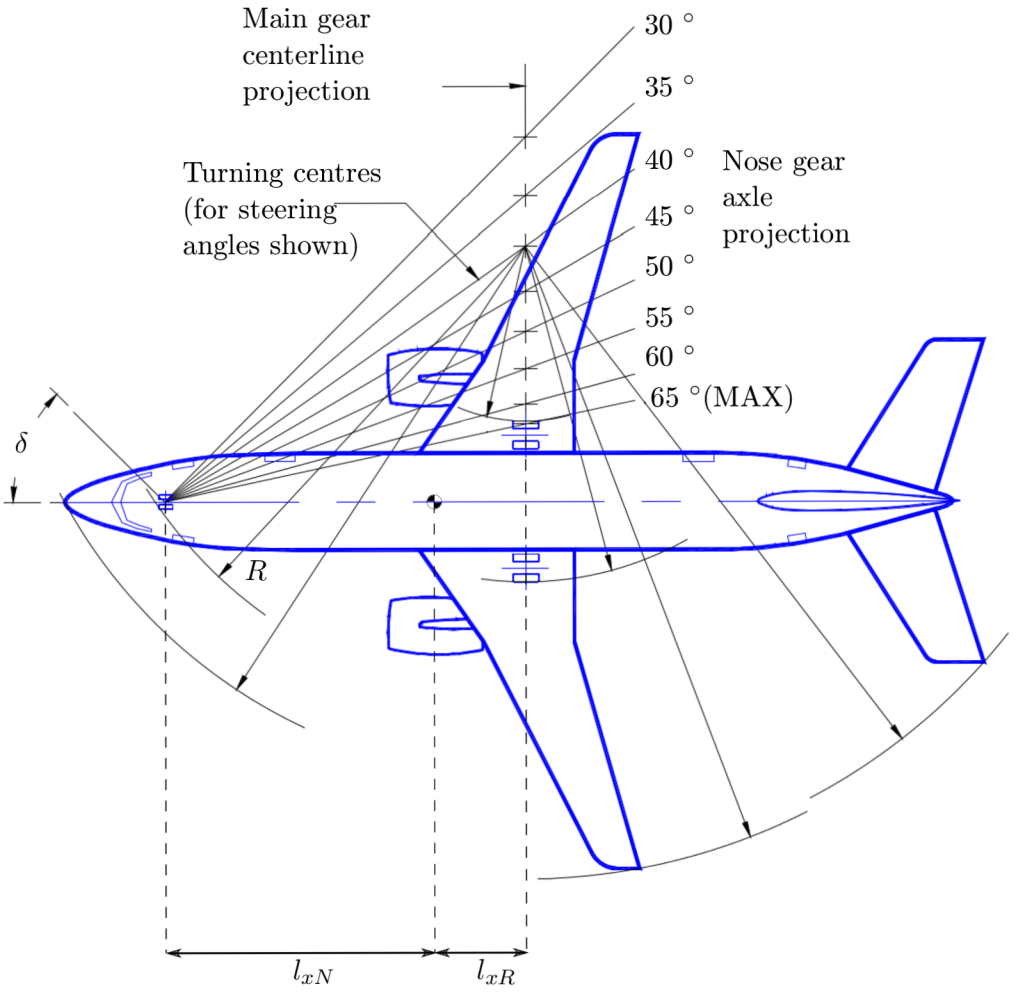


Figure 5.2: Ackermann tricycle steering model and the turn radii for Boeing 737-400. Nose-wheel radius is denoted by R [6].

Table 5.2: Error evaluation of simulated turn radii R relative to analytical solution of Ackermann steering and Boeing specifications.

	Analytical solution		Boeing spec. [6]	
Steer angle δ [deg]	Absolute Error [m]	Relative Error [%]	Absolute Error [m]	Relative Error [%]
30	1.50	5.00	1.20	4.00
35	0.90	3.49	0.60	2.33
40	0.48	2.12	0.18	0.79
45	0.10	0.49	-0.20	0.99
50	-0.19	1.03	-0.49	2.66
55	-0.47	2.78	-0.77	4.55
60	-0.43	2.68	-0.73	4.54
65	0.12	0.76	-0.28	1.77

5.3 AutoTaxi Experiments

A further verification should be based on a comparison of the simulated data to the real data in order to confirm that the model is valid and represents a real aircraft behavior. The simulation model is assumed to be validated using an actual real aircraft measured data from the taxiing trials at designated international airport. However, at this stage, no such data are available. Therefore, subsidiary experiment has been performed. In the experiment, taxiway guidelines on the airport's runway were approximated by series of GPX waypoints. These guidelines mark an ideal trajectory for the aircraft nose wheel during the taxiing and as such are a good reference for evaluation of the ability of aircraft to steer and follow predefined trajectory. The density of taxilines approximation depends on its curvature. Straight segments are approximated by two waypoints, at the beginning and at the end; turns are approximated with higher waypoint density - up to 10 meter spacial separation between consecutive marks.

Simulated taxiing trials were performed at international airports in Czech Republic, namely airports in Brno, Ostrava and Prague. The observed behavior of the aircraft during the experiments is shown in Figure 5.3, 5.5 and 5.6 respectively. The trajectories were chosen so that there are sharp turns (small radius turns), regular turns (large radius turns) and straight segments in order to evaluate the fidelity of the simulation model under various scenarios. The varying atmospheric conditions are also assumed, as it influences the surface characteristics of the taxiways and runways. Standard dry conditions of the runway surface are considered, as well as wet runway surface during rainfalls and snow/ice contaminated runway surface during freezing temperatures. The standing water pools on the runway in wet conditions are not assumed since it would require significant changes of the simulation model and the definition of new friction coefficients in these scenarios. The snow/ice contamination assumes approximately 15 cm snow layer on the runway with underlaying ice layer [5].

5.3.1 Dry Runway Conditions

In standard operational conditions with a dry runway surface, the aircraft successfully covered the test trajectory for all three aforementioned international airports. The maximum trajectory offset from predefined routes was below 1.6 m and a velocity deviation from the target velocity in negative direction (overshoot) did not exceed 0.6 $m \cdot s^{-1}$. Positive velocity deviation (undershoot) in sharper turns reached deviations with greater amplitude because of activated differential braking, which slows the aircraft down. In large radii turns, the absolute error amplitude stayed below 0.2 $m \cdot s^{-1}$ and 1 m spatial offset from the desired trajectory.

The offset from predefined trajectory (path-error) reaches its maximal value in sharp turn segments of the test taxiing trials. This behavior is partially expected, since the taxiway is approximated only by straight lines represented by series of discrete points but the aircraft is moving on smooth curve. As the angle between lines approximating the turn increases (which happens in small radii turns), the path-error increases as well. The deviation is caused by introducing circle with radius $r = 3$ m around individual waypoints (see Figure 4.6). Inside the circle the waypoint is considered reached and next waypoint is selected as target. However, under such situation the aircraft is still 3 m away from the given waypoint. And since path-error is evaluated as offset of aircraft's position between line connecting target waypoint and previous waypoint (from which aircraft is 3 m away),

this necessarily introduces path-error in case waypoints are not aligned. Obviously, this error is most evident if waypoints are approximating perpendicular lines. In the taxiing experiments, all taxiways are approximated smoothly so that there are no such situations as perpendicular lines. But as turn radius decreases (turn is sharper), the angles between individual waypoints increase. As a result, the overshoot error becomes more noticeable.

5.3.2 Wet Runway Conditions

The simulation considering wet runway surface confirmed the ability of AutoTaxi controller to steer the aircraft along the predefined trajectory with required precision under varying atmospheric conditions. The AutoTaxi controller steered the aircraft with precision up to 2.7 m trajectory offset from the desired trajectory, which is within the maximal allowed error. Velocity control restrained overshoots greater than 0.7 $m \cdot s^{-1}$, which is also in acceptable safety margins. Similar results are observed for all selected airports (see Figure 5.3d and 5.3e for the Brno airport; Figure 5.5d and 5.5e for the Ostrava airport; Figure 5.6d and 5.6e for the Prague airport).

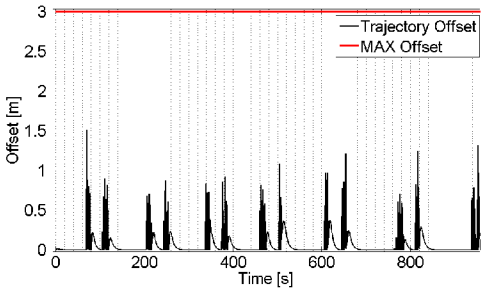
5.3.3 Snow/Ice Contaminated Runway

The harsh environmental conditions impose several restrictions for the AutoTaxi controller. First, the taxiing velocity of the aircraft needs to be significantly decreased, up to 1.5 $m \cdot s^{-1}$, due to very limited friction of the wheels on the snow and ice contaminated runway. The tire friction is further decreased with the increased velocity, therefore the velocity needs to be limited.

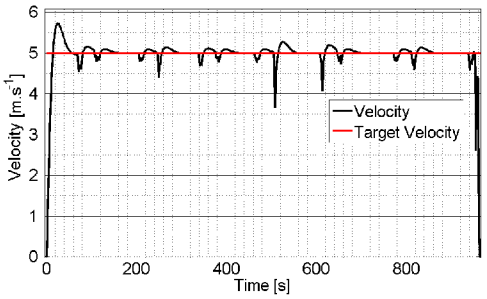
Second, the aircraft did not successfully pass the test trajectory in neither of the prepared taxiing experiments. The safety margin of 3 m was violated in sharp turns, as shown in Figure 5.4, even with the limited aircraft velocity. The AutoTaxi controller is designed to primarily utilize the nose-wheel steering of the aircraft. However, the efficiency of nose-wheel steering method is considerably reduced in freezing conditions as the nose-wheel very easily skids on the slippery runway surface. The approach to the aircraft control in such conditions requires to use smaller nose-wheel steering and utilize the differential engine thrust in order to assists in maintaining the aircraft momentum through a turn [25]. Differential braking may also be more effective than nose-wheel steering on slippery or contaminated surfaces.

For these reasons, the AutoTaxi controller requires different design approach for the aircraft control during winter freezing temperatures and associated runway conditions. The resulting controller would include two operational regimes - one for standard dry conditions and wet runways and second for snow/ice covered runways. The appropriate operational regime would be selected prior to the aircraft departure from the gates to taxiway, either by pilot or by submodule responsible for analyzing the weather conditions.

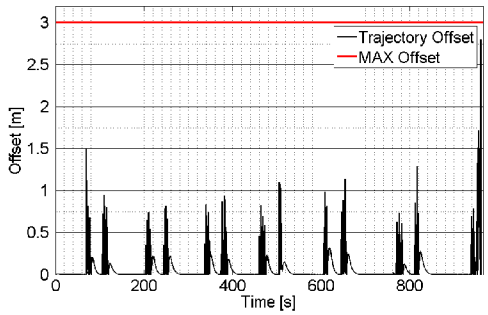
(a) Trajectory of the aircraft during experiment.



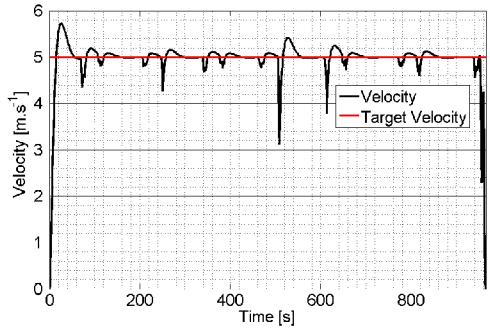
(b) Trajectory offset (dry surface).



(c) Velocity characteristic (dry surface).



(d) Trajectory offset (wet surface).



(e) Velocity characteristic (wet surface).

Figure 5.3: Evaluation of AutoTaxi on Brno International Airport.

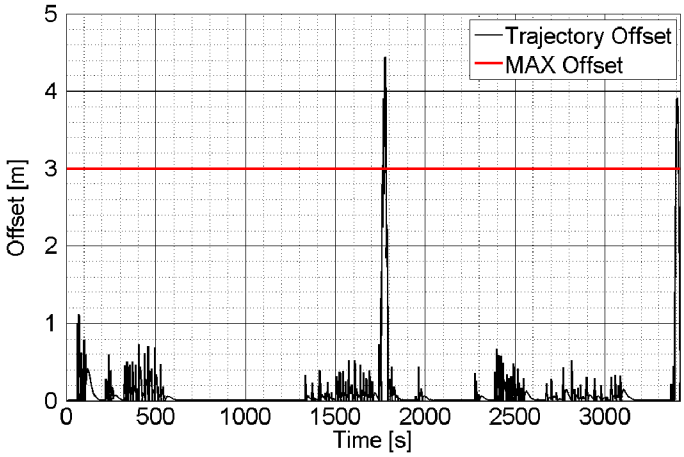
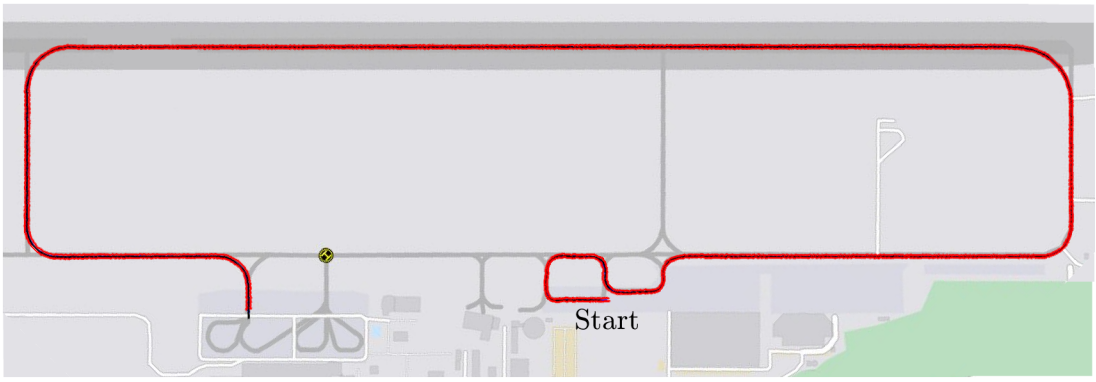
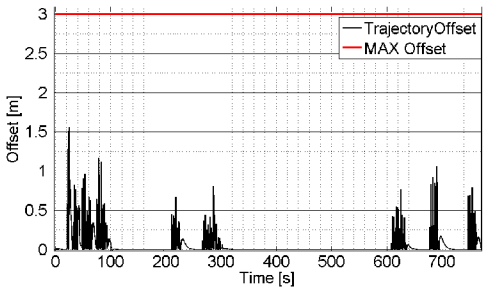


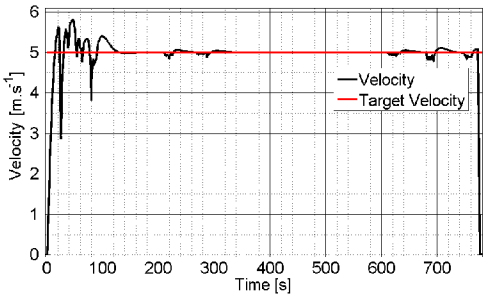
Figure 5.4: Trajectory offset on snow/ice covered runway at Prague airport.



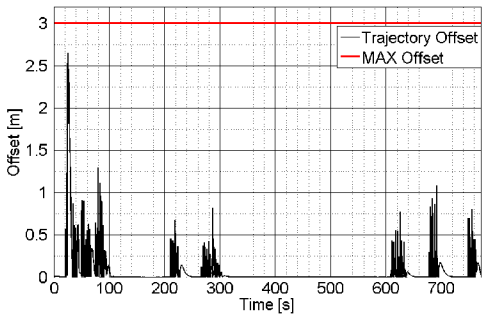
(a) Trajectory of the aircraft during experiment.



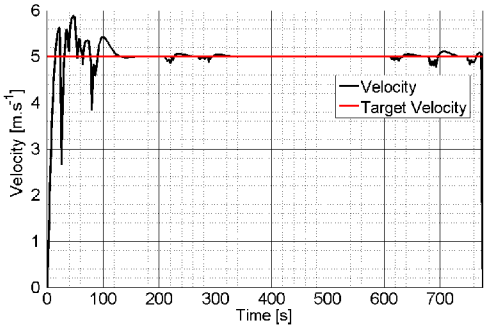
(b) Trajectory offset (dry surface).



(c) Velocity characteristic (dry surface).

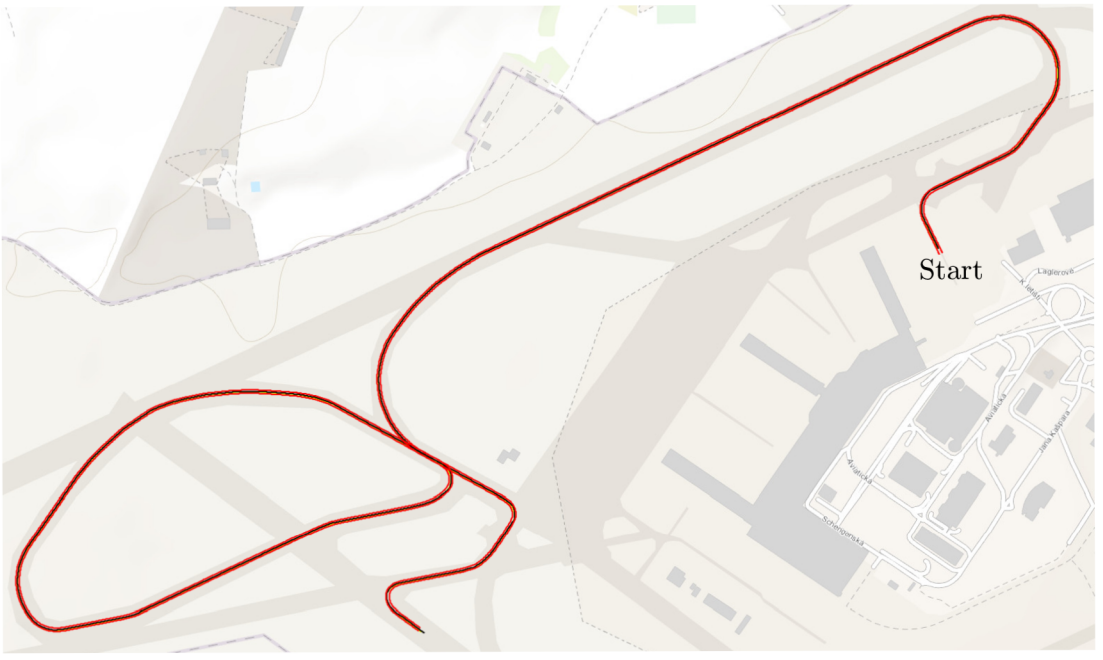


(d) Trajectory offset (wet surface).

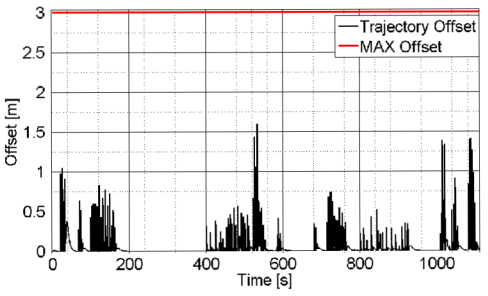


(e) Velocity characteristic (wet surface).

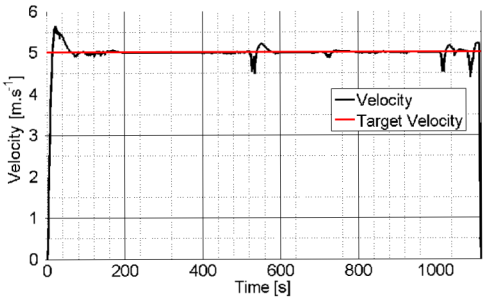
Figure 5.5: Evaluation of AutoTaxi on Ostrava-Mošnov International Airport.



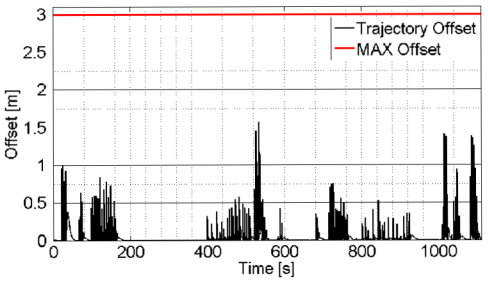
(a) Trajectory of the aircraft during experiment.



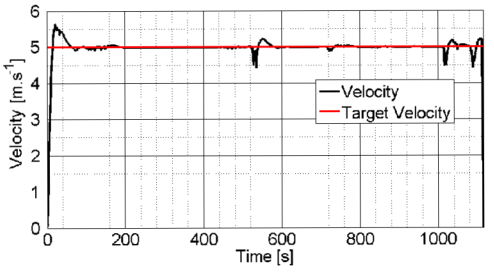
(b) Trajectory offset (dry surface).



(c) Velocity characteristic (dry surface).



(d) Trajectory offset (wet surface).



(e) Velocity characteristic (wet surface).

Figure 5.6: Evaluation of AutoTaxi on Prague International Airport.

6. Conclusion

This thesis presents a ground motion model of single-aisle passenger aircraft and a AutoTaxi control algorithm, which is required in order to successfully manage the automation tasks. Automation will play essential role in increasing throughput of airports in metropolitan areas that will need to service more flights since passenger numbers continually grow and such trend is expected to continue also in near future. Presented model is valid under different operational conditions, such as varying runway characteristics due to the state of the atmosphere at a particular place and time regarding the temperature and precipitation. Model is able to represent standard concrete runway surface, water contamination during rainfalls or icy and snow-covered runway conditions during freezing temperatures. The simulation model also assumes varying aircraft parameters such as aircraft load and tire pressure. These parameters influence the interaction of the landing gear with the runway surface, therefore are essential for a high-precision ground motion modeling.

The MATLAB/Simulink® implementation of the model is also presented. Furthemore, ground motion model is used as an underlying component for the AutoTaxi controller module, which is able to automatically steer aircraft to the predefined target destination based on following the trajectory defined by the taxi-lines on the airport runways. AutoTaxi controller establishes feedback loop to control input signals in order to achieve desired steering, throttle and brakes control of the aircraft.

The model is designed to be used in a dynamic analysis and real-time simulation. Obtained results suggest high-precision simulation, that can be utilized for the automation of the ground operations at the airports in metropolitan areas. The automation will lead to decreased separation minima and increased airport throughput, reduced CO_2 emissions and increased safety levels.

The ability of the AutoTaxi to perform realistic simulations was verified based on comparison with analytical solution of Ackermann tri-cycle model and turn radii specification by Boeing. The error of simulated turn radii is below 4.5% for whole operational range. Evaluation of the AutoTaxi control algorithm at multiple international airports showed deviation from predefined trajectory below 60% of the maximal allowed error in standard dry operational conditions and wet runway conditions. The safety margin was violated during the simulated snow/ice contaminated runway taxiing trials. The AutoTaxi controller primarily uses nose-wheel steering as a mean of the aircraft control. However, in such conditions, the nose wheel steering is no longer efficient as the nose wheel tire skids easily on the slippery runway surface. Different operational regime is therefore required for this scenario.

Developed simulation model can be used as a simulation tool for aircraft taxi routines. Incorporating model into airport planning framework would result into high-precision simulation of aircraft trajectory with a possibility to adapt to airport's environmental conditions. Such framework might be deployed to plan optimal trajectories of airport service

vehicles and aircrafts. Optimal airport planning would increase airport throughput and minimize runway incidents and collisions. Future work will be focused on developing model of selected airports and designing planning algorithm that will select optimal trajectory of aircraft to its destination within given airport. In addition to the planning algorithm on individual aircraft level, control algorithm on global level is required, that will avoid spacial and temporal crossing of trajectories from multiple aircrafts.

Bibliography

[1] K. Aainsqatsi. Schematic diagram of a 2-spool, high-bypass turbofan engine. http://commons.wikimedia.org/wiki/File%3ATurbofan_operation.svg, 2004.

[2] Federal Aviation Administration. Next generation air transportation system. FAA, 2007-02 [cit. 2014-10-25]. https://www.faa.gov/nextgen/.

[3] R. Andersen. Exodus. Aeon, 2014-09 [cit. 2014-10-20]. http://aeon.co/magazine/technology/the-elon-musk-interview-on-mars/.

[4] K.H. Ang et al. Pid control system analysis, design, and technology. *IEEE Trans Control Systems Tech*, 13(4):559–576, 2005.

[5] A.G. Barnes and T.J. Yager. Enhancement of aircraft ground handling simulation capability. 1998. ISBN 92-836-1066-0.

[6] Boeing. 737 airplane characteristics for airport planning. Boeing Commercial Airplanes, September 2013. http://www.boeing.com/assets/pdf/commercial/airports/acaps/737.pdf.

[7] G. Cai et al. *Unmanned Rotocraft Systems*. Springer, 2011. ISBN: 978-0-85729-634-4.

[8] D. Chaturvedi. *Modeling and Simulation of Systems Using MATLAB and Simulink*. CRC Press, 2009. ISBN-978-1-4398-0673-9.

[9] E. Coetzee. *Modelling and nonlinear analysis of aircraft ground manoeuvres*. PhD thesis, University of Bristol, May 2011.

[10] European Comission. Clean sky. http://www.cleansky.eu/, 2002-10 [cit. 2014-10-25].

[11] The Boeing Company. 737 - 600/700/800/900 flight crew training manual, 2005. http://www.smartcockpit.com/.

[12] T. Crouch. *Wings: A History of Aviation from Kites to the Space Age*. New York: W.W. Norton & Co, 2004. ISBN 0-393-32620-9.

[13] N. Dattani. Linear multistep numerical methods for ordinary differential equations. 2008. Department of Applied Mathematics, University of Waterloo.

[14] G. Fasshauer. Numerical methods for differential equations. 2007. Illinois Institute of Technology.

[15] Federal Aviation Administration. Runway safety - pilots best practices. FAA, 2011-03 [cit. 2014-12-27]. http://www.faa.gov/airports/runway_safety/pilots/best_practices/.

[16] J. Feldman. Variable step size methods. 1999. University of British Columbia.

[17] H. Goldstein, CH. Poole, et al. *Classical Mechanics*. Pearson, 2010. ISBN-10:1-292-02655-3.

[18] D. Griffiths and J. Higham. *Numerical Methods for Ordinary Differential Equations*. Springer Berlin Heidelberg, 2010. ISBN-978-0-85729-148-6.

[19] D. Henrion, L. Reberga, et al. Linearization and identification of aircraft turbofan engine models. 2004.

[20] The MathWorks Inc. Solver pane documentation page. http://www.mathworks.com/help/simulink/gui/solver-pane.html, 2015. [cit. 2015-3-23].

[21] J.R. Leigh. *Control Theory*. Institution of Engineering and Technology London, 2004. ISBN-978-0-86341-339-1.

[22] J.P. Massey. Control and waypoint navigation of an autonomous ground vehicle. 2006. Texas A&M University.

[23] NASA. National plan for aeronautics research and development and related infrastructure. http://www.whitehouse.gov/sites/default/files/microsites/ostp/aero-rdplan-2010.pdf, 2010-02 [cit. 2014-10-24].

[24] A. Noureldin et al. *Fundamentals of Inertial Navigation, Satellite-based Positioning and their Integration*. Springer Berlin Heidelberg, 2013. ISBN-978-3-642-30465-1.

[25] H. Oda et al. Safe winter operations. Boeing, October 2010. http://www.boeing.com/commercial/aeromagazine.

[26] P. Peringer. Modelování a simulace. FIT VUT v Brně, 2010. http://www.fit.vutbr.cz/study/courses/IMS/public/.

[27] A. Pytel and J. Kiusalaas. *Engineering Mechanics: Statics*. 2010. ISBN-10:0-495-29559-0.

[28] J. Rankin and B. Krauskopf. Operational parameter study of aircraft dynamics on the ground. 2010. J. Comput. Nonlinear Dynam. 5.

[29] J. Rankin, M.Desroches, et al. Canard cycles in aircraft ground dynamics. *Nonlinear Dynamics 66*, 2011.

[30] C. T. Russell. Numerical simulation. University of California, Los Angeles, 2002-10 [cit. 2015-04-10]. http://www-ssc.igpp.ucla.edu/personnel/russell/ESS265/Ch10/ylwang/node8.html.

[31] Airbus S.A.S. Airbus mou with iai to explore eco-efficient 'engines-off' taxiing. Airbus, 2009-06-17 [cit. 2014-10-24]. http://www.airbus.com/presscentre.

[32] Airbus S.A.S. Global market forecast for 2014-2033. Airbus, 2014-10 [cit. 2014-10-24]. `http://www.airbus.com/company/market/forecast/`.

[33] J.M. Zogg and U-Blox. *GPS: Essentials of Satellite Navigation : Compendium : Theorie and Principles of Satellite Navigation, Overview of GPS/GNSS Systems and Applications.* U-Blox, 2009.

A. Table of Symbols

Symbol	Description	Unit
F_{xN}	Nose gear longitudinal force	N
$F_{xR,L}$	Main gear (right, left) longitudinal force	N
F_{xT}	Thrust force from engines	N
F_{xA}	Aerodynamic drag force in x dimension	N
F_{yN}	Nose gear lateral force	N
$F_{yR,L}$	Main gear (right, left) lateral force	N
F_{zN}	Nose gear vertical force	N
$F_{zR,L}$	Main gear (right, left) vertical force	N
l_{xN}	Nose gear x-distance relative to C.G.	m
$l_{xR,L}$	Main gear x-distance relative to C.G.	m
l_{xT}	Engine x-distance relative to C.G.	m
$l_{yR,L}$	Main gear y-distance relative to C.G.	m
l_{zN}	Nose gear z-distance relative to C.G.	m
$l_{zR,L}$	Main gear z-distance relative to C.G.	m
l_{zT}	Engine z-distance relative to C.G.	m
δ	Nose wheel steer angle	rad
ψ	Tire yaw angle	rad
μ_R	Rolling resistance coefficient	1
k_b	Proportion of brakes being applied	1
$\mu_{b_{max}}$	Maximal braking coefficient	1
$\mu_{b_{skid}}$	Tire skid coefficient	1
$\mu_{b_{eff}}$	Braking effectiveness coefficient	1
μ_ψ	Lateral friction coefficient	1
$\mu_{\psi_{max}}$	Maximal lateral friction coefficient (no braking)	1
$\mu_{\psi_{lim}}$	Limiting lateral friction coefficient (braking)	1
V_x	x-axis component of velocity vector	$m \cdot s^{-1}$
V_y	y-axis component of velocity vector	$m \cdot s^{-1}$
ω_z	Angular velocity about z-axis	rad^{-1}
X	X coordinate of the aircraft position	m
Y	Y coordinate of the aircraft position	m
Ψ	Heading angle	rad