

Bachelor's Thesis



Czech
Technical
University
in Prague

F3

Faculty of Electrical Engineering
Department of Electromagnetic Field

User trajectory prediction based on a-priori knowledge of position and velocity

Vanja Neretljak

May 2016

Supervisor: Ing. Pavel Puričer (Department of Radio engineering)

Acknowledgement / Declaration

First I would like to thank my thesis supervisor Ing. Pavel Puričer from the Faculty of Electrical Engineering at Czech Technical University in Prague. The door to Ing. Puričer office was always opened whenever I ran into a trouble spot or had a question about my research or writing. He consistently allowed this thesis to be my own work, but steered me in the right direction whenever he thought I needed it. I must express my very profound gratitude to my parents and to my friend Vladimir Perić for providing me by unfailing support throughout my years of study and through the process of researching and writing this thesis. This project would not have been made without them.

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

V Praze dne 27.5.2016

A handwritten signature in black ink, appearing to be a stylized 'B' or similar character.

Abstrakt / Abstract

Tento projekt se zabývá předpovídáním trajektorií letadel. Popisuje různé metody a ilustruje je na reálných datech. Trajektorie lze předpovědět z informací o letu, přičemž za použití údajů jako rychlost, zrychlení, dosaovací body průletu apod. lze budoucí trajektorii předpovědět s poměrně vysokou přesností ve smyslu procentuální chyby. Tento projekt je koncipován jako prvotní návrh implementace programu vytvořeného v softwaru Matlab pro předpovědi v procesech využívajících TCAS systémy. Výsledkem projektu jsou různé modely pro rychlou předpověď trajektorie letadla, které jsou využity k odesílání informace do kokpitu ve velmi krátkém časovém úseku, což má za následek aktivaci systému TCAS, který by zabránil nebezpečnému vzájemnému přiblížení se dvou letadel ve vzduchu.

Klíčová slova: Letadlo, trajektorie, TCAS, Matlab.

Příklad titulu: Predikce trajektorie pohybu uživatele ze znalosti apriorních polohových a rychlostních dat

The prediction of an airplane trajectory, using different methods and different real-time available data, is described in this thesis. According to the information about the airplane flight, it is possible to predict its route. Using data such as speed or velocity, future points of its trajectory can be predicted in short time period with high accuracy. Moreover, using data such as values of points of its position that have already been sent to us, done prediction of the next points can be done, also with small error percent, too. This project is conceived to be the starting idea of implementing Matlab prediction in process of work of TCAS systems. The result of the project is a set of different models of prediction of an airplane route designed to send information in a very short time period to airplane, and in that way prevent turning on TCAS system which can prevent nearmiss of two airplanes on the sky.

Keywords: Airplane, trajectory, TCAS, Matlab.

Contents /

1 Introduction	1	References	37
1.1 Flow diagram	4	A CD	39
2 Defining position using GNSS system	5		
2.1 Navigation Principles	5		
2.2 Global Positioning System - GPS	6		
2.3 Geodetic system	8		
2.3.1 Transformation between Geodetic and ECEF coordinates	8		
3 Description of The Traffic Collision Avoidance System (TCAS)	10		
3.1 Method of TCAS	11		
4 Description GPX	14		
4.1 Matlab and GPX files	15		
5 List of airplanes used in scrutinizing	16		
5.1 Technical information about airplanes	17		
5.2 Description of the considered airport: Vaclav Havel (Ruzyne) Prague	17		
6 Methods for determining position of airplanes in Matlab	19		
6.1 Linear extrapolation	19		
6.1.1 Test of linear extrapolation - Matlab	22		
6.2 Exponential extrapolation	23		
6.3 Conclusion of evaluation two different methods	24		
7 Altitude prediction of airplane trajectory	25		
8 Prediction of Longitude and Latitude of airplane trajectory based on previous points	29		
9 Application of the method in TCAS	32		
9.1 Sensitivity Levels and tau	32		
9.2 <i>time-to-go process</i>	33		
9.3 Implementation of our method	33		
10 Conclusion	35		
10.1 Future work	35		
10.2 Airport Vaclav Havel	36		

Tables /

6.1. Point values	22
8.1. 2D trajectory point values	31
9.1. Sensitivity Levels	33

Chapter 1

Introduction

*"The prediction is very difficult, especially if it's about the future."*¹⁾ This sentence was said by Niels Henrik David Bohr, Danish physicist. Thinking about it just for a moment, one has to completely agree with it. People would like to predict at least one step into the future, and many times they fail. They want to predict many small things which are important for them in some moments of their lives, but this thesis is going to predict something that is important to many of them in large number of moments in everyday life. I am not talking about things such as the prediction of the final result in an ice hockey game or about the weather forecast for tomorrow. In this project, I want to predict as correct as I can, something that can save many lives and that can save money to companies that take care about their airplanes and passengers. I know that it may be a big piece of cake for me, but my wish for the reduction of airplane incidents and accidents is really strong, and thus my thesis will try to fight this problem.

According to many sources and information, airplane accidents happen mostly because of the human error, and often computers have advantage in making orders to pilots because it is shown that they had less errors than human. Therefore I decided to make a computer program to determine the airplane trajectory in the future. Different ways to do it are found, according to different data that is available. Someone may ask, whether the prediction is correct, and the answer is very simple. This project will test all the ways of predictions by comparing our predictions to real data. Therefore it will be known in which time period, it predicted trajectory of one airplane with the smallest error. It should help airplanes in the sky to determine real and predicted trajectories of airplanes close to them or their predicted route of flight, and consequently to make needed decisions.

In 21st century, the planet Earth is no longer a big place for living as it was before, when people were not using as modern technology as today. Everything is improving so fast, and it affects our planet, some things in a good way, the others in a very bad one. Motivation for this project was to do for this planet and people who still believe that it will survive together with them, at least one good thing. A man can do many things to improve his society and his neighborhood but always has to start with solving the first problem.

In the previous years, our society (and globally our species) had many problems in everyday transport. Every day, accidents happen all around the world. In this work I do not want to discuss whether train transport is safer than travelling by car, or ship transport is safer than airplane. It is important to make all of them safe as much as possible.

The passion for airplanes and the sky runs through veins of many people, and I am one of them. At the beginning of this work, I believed that my bachelor thesis would be connected to this topic. Another thing that led me through this work was love for navigation. Radars, maps and orientation in space gave me an idea to make a **goal of this project which is to help to reduce aircraft accidents when knowing their**

¹⁾ Niels Bohr, 1885-1962.

position in every moment and predicting it with a very small error if something is wrong in their electrical system.

This project is conceived to show analytical view of few different methods finding correct position of an airplane. Of course, it would not be interesting to make something that is already in use, therefore of that all three methods that are presented in the project are made knowing only few airplane characteristics. The speed and the velocity, the points of position before and prediction of the next one, are considered in calculations. As mentioned in the previous paragraph, I want to show that there is chance to estimate position of the airplane in the air without any or small amount of error. Another motivation for this project was found in EUROCONTROL[1] center and in their Experimental Flight Management System (denoted as EFM System in the sequel). It was made to test different situations of flight from a gate to another gate focusing on airplane trajectory. The main purpose of this system was to check trajectories against each other to identify any conflicts between them. If the test shows that everything is all right, then airplane can be assured a trouble free and unperturbed flight. I found this way of solution very interesting but since it is a test variant I decided to make it little simpler and easier to use in every airplane. My project is conceived to show analytical view of few different methods finding correct position of airplane. The last point of the motivation was found in Traffic Collision Avoidance system (TCAS) and the idea is to implement my results and methods to improve TCAS. The plan is to make more sensitive TCAS and to reduce the amount of situations by turning on its alarm system. Many engineers strive to make project that will not be from economic perspective so required. This means that plan is to make this idea at the end, as cheap as possible, expensive. Matlab, a high-performance language for technical computing, that has already been in wide use, will be used for work with observed data in this thesis. I hope that aircraft companies may find this project interesting because it is in their interest to make the airplanes safer and to offer to their costumers the best and the safest way of transport.

This thesis is organized as follows. Chapter 2 concerns Global Navigation Satellite System (GNSS) system together with Global Positioning System (GPS). Description of how it works and how it should be used can be found in this chapter. In Chapter 3 there is theoretical explanation of the Traffic Alert and Collision System. The following chapter describes GPX files and how they are get together with their role in this work. Reaching our goal would not be possible without them. GPX files give access to point values of airplane trajectory. As it was created at the beginning, the goal is to predict aircraft trajectory, and this gives GPX files huge importance in finding solution for prediction of trajectory. The following chapters will describe how many points will be taken in observation and in which way it will be done. Matlab can process GPX files in different ways, which is helpful for methods that will be created for prediction of trajectory. Connection between Matlab and GPX files is described in Chapter 5 together with explanation and description of GPX files. Methods of prediction will be different but GPX files do not affect it in any way so they will be the same. Speaking about methods, it is important to write how many of them will be. In Chapter 6, linear and extrapolation methods of prediction are explained theoretically and it is shown how they work in Matlab. Further, in this chapter there is described a prediction of a point that is done using mathematics only and then it is coded in Matlab to check percentage of correction.

In real life situations, different airplanes fly with different trajectories. Because of that, this projects requires few different airplanes to be taken in observation. After

GPX files and methods of prediction, in Chapter 7 there is technical description of different airplanes with all data that has been available. Different characteristics of airplanes give different solutions for prediction of their trajectory. According to their speed or velocity, different methods can be used.

Chapters 8 and 9 concern different predictions. In the first one, there is described altitude prediction. It is done in Matlab using GPX file of one aircraft flight. Also in this chapter, there are shown graphs following airplane trajectory and showing difference in real and predicted flight. Details about how it is done are explained in this chapter. Chapter 9 describes the main part of this thesis which is a prediction of longitude and latitude of airplane trajectory based on knowing few points before. In this chapter, there is in detail explained in which way it needs to be done.

At the end, there is described how this idea will be implemented in TCAS system and how it will help reducing accidents and incidents at the sky. The last chapter concerns summary and gives ideas for the future work.

On the following page, there is shown flow diagram that represents in one algorithm step by step reaching the goal.

1.1 Flow diagram

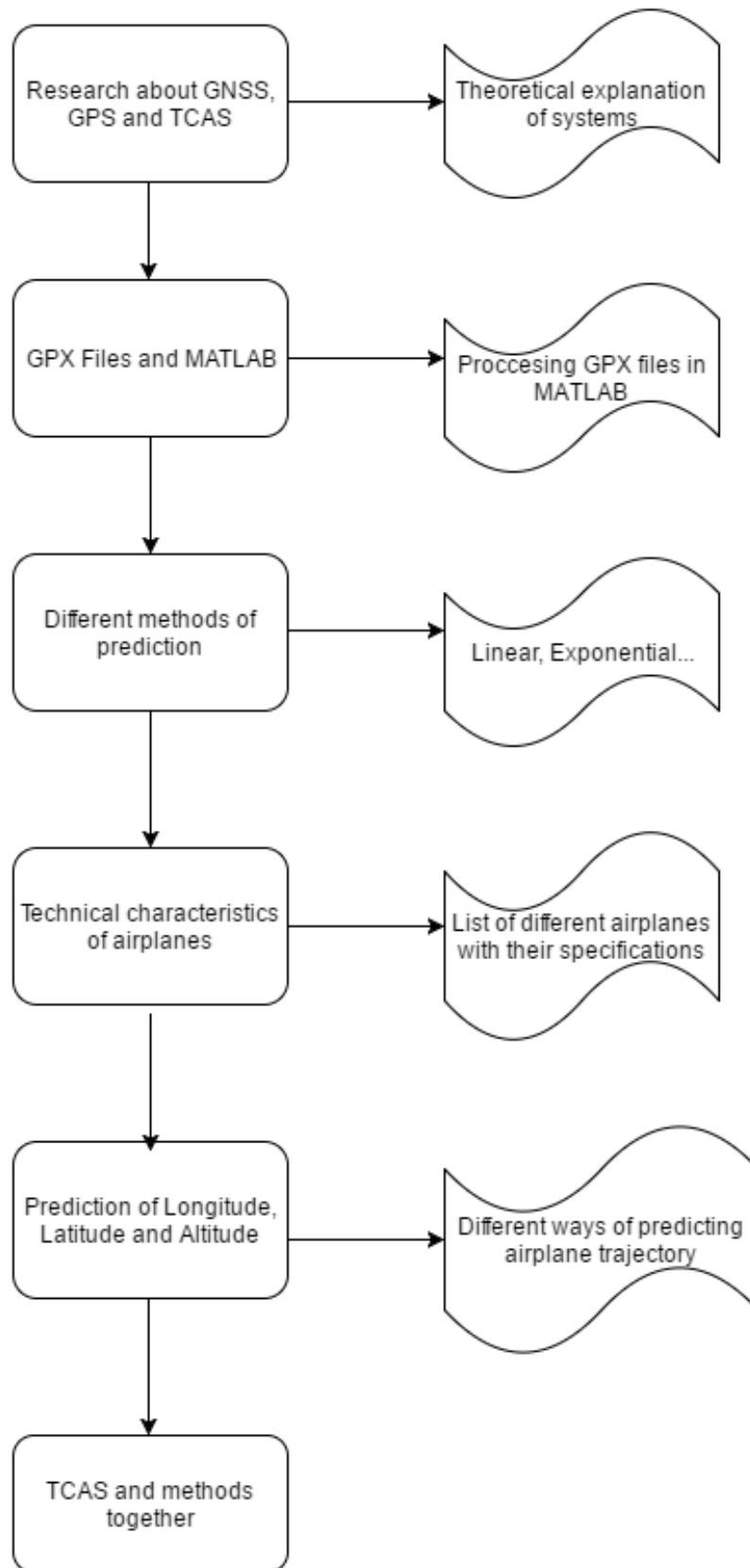


Figure 1.1. Step by step algorithm of realization.

Chapter 2

Defining position using GNSS system

2.1 Navigation Principles

*"Navigation is science of determining position, location, distance traveled, and course to a known destination."*¹⁾ This means that the navigation is used by people in everyday life. People need to know their position in almost every moment, as when they want to come to work on time or when they need to use some shortcuts, when a child gets lost after staying late in school and it needs safe road home etc. But when we speak about something which is more important for people everywhere in the world, something used in getting directions and positions of ships, trains and the fastest way of transport airplanes, then we speak about very serious and much more complicated navigation systems. In this project, I will use principles of aircraft navigation system. By definition of International Civil Aviation Organization²⁾, *GNSS is a world-wide position and time determination system that includes one or more satellite constellations, aircraft receivers, system integrity monitoring augmented as necessary to support the required navigation performance for the intended operation*[2]

In the past, pilots looked out their open cockpits for roads, rail lines and airports to find their way in daytime flight. The pilots were looking the horizon to make sure they were flying with the aircraft's nose and wings in the proper position relative to the ground, called altitude. As the pilots began to fly at night and in all kinds of weather in the early 1920s, a new equipment helped them to navigate and maintain aircraft attitude when they could not see the ground. Navigation aids were developed for use in the inside of the aircraft and also to guide the pilots from the ground.

Navigation is the art and science of getting from point A to point B in appropriate time without losing your way.[3] In the early days of aviation, navigation was mostly an art. The simplest instruments of flight had not been invented, so pilots flew by the seat of their pants. Today, navigation is a science with sophisticated equipment being standard on the most of the aircrafts. The type of the navigation used by the pilots depends on many factors. The navigation used method depends on where the pilot is going, how long the flight will take, when the flight is to take off, the type of aircraft being flown, the on-board navigation equipment, the ratings and currency of the pilot and especially on the expected weather.

Today, there are many different navigation principles and rules that are used by the pilots. Pilotage for a non-instrumental rated, Deductive Reckoning or better known as Dead Reckoning navigation or Radio navigation (that is used almost in all aircrafts today) are some of the ways of navigating the airplanes. But the best friend of all the pilots in aircrafts is compass. It is not always the truest friend because the moving of an airplane can make big effects on compass, so big that sometimes the pilot does not know if compass works for him, or against him.

¹⁾ Bussines Dictionary, 2016.

²⁾ ICAO 2005.

If we talk about the most useful and the most precise system for determining the position and getting almost always true and correct real time direction we have to talk about Global Positioning System (GPS) which is described in the following section.

2.2 Global Positioning System - GPS

Global Positioning System (GPS)[4] is fully operational and meets the criteria established in the 1960s for an optimum positioning system. The system provides accurate, continuous, worldwide, three-dimensional position and a velocity information to users with an appropriate receiving equipment. GPS also disseminates a form of Coordinated Universal Time (UTC). The satellite constellation nominally consists of 24 satellites¹⁾ arranged in 6 orbital planes with 4 satellites per plane. A worldwide ground control/monitoring network monitors the health and status of the satellites. This network also uploads navigation and other data to the satellites.

GPS can provide service to an unlimited number of users since the user receivers operate passively. Also this system is dual-use system, which means it provides separate services for civil and military users. Standard Positioning Service (SPS) is designated for the civil community. Another system is Precise Positioning Service (PPS) and it is used for military purposes.

In aviation, the use of GPS system led to the use of GNSS system. Global Navigation Satellite System is defined as the worldwide set of satellite navigation systems. It is used in airplane navigation to provide guidance for the route through precision approach phases of flight. The continuous global coverage of this system permits aircraft to fly directly from one location to another provided factor such as obstacle clearance and required procedures are adhered to. GPS is comprised of three segments: satellite constellation, ground-control/monitoring network and user receiving equipment. Formal GPS JPO programmatic terms for these components are space, control and user equipment segments, respectively. The satellite constellation is the set of satellites in orbit that provide the ranging signals and data message to the user equipment. The control segment tracks and maintains the satellites in space. And at the end the user receiver equipment performs the navigation, timing or other related functions. When we talk about the errors that can happen during the flight using GPS system it depends on many different things. The accuracy with which an user receiver can determine its position or velocity or synchronize to GPS system time, depends on a complicated interaction of various factors. In other words, GPS accuracy performance depends on the quality of the pseudo range and carrier phase measurements as well as the broadcast navigation data. If we put this in the form of a mathematical formula it would look like this:

$$(\text{error in GPS solution}) = (\text{geometry factor}) \times (\text{pseudo range error factor})$$

GPS uses the concept of Time of Arrival (TOA) ranging to determine user position. This concept entails measuring time it takes for a signal transmitted by an emitter at a known location to reach an user receiver. This time interval referred to as the signal propagation time, is then multiplied by the speed of signal to obtain the emitter to receiver distance. By measuring the propagation time of the signal broadcast from multiple emitters at known location positioning is provided next.

¹⁾ Note that today there are active 31 satellites.

If we want to describe mathematical way of the determine position using GPS system we have to first determine which coordinate system we will use. Cartesian coordinate system is worldwide spread system used in many spheres of our lives. This coordinate system is an inertial and rotating system and it is made of few other systems. Earth-Centered Inertial coordinate system is one of them. It is used for purposes of measuring and determining the orbits of GPS satellites. Its origin is at the center of the mass of the Earth and its axes are pointing in fixed directions with respect to the stars. In this coordinate systems, the xy -plane is taken to coincide with the Earths equatorial plane, the $+x$ -axis is permanently fixed in a particular direction relative to the celestial sphere, the $+z$ -axis is taken normal to the xy -plane in the direction of the north pole, and the $+y$ -axis is chosen so as to form a right-handed coordinate system.

Another system is Earth-Centered Earth-Fixed coordinate system and it used for purpose of computing position of a GPS receiver. In this coordinate system it is easier to compute the latitude, longitude and height parameters that the receiver displays. The $+x$ -axis of this system points in the direction of 0 longitude and the $+y$ -axis points in the direction of 90 E longitude. The x -, y - and z -axes therefore rotate with the Earth and no longer describe fixed directions in inertial space. z -axis in this system is chosen to be normal to the equatorial plane in the direction of the geographical North Pole, thereby completing the right-handed coordinate system.[4]

WGS known as World Geodetic system is also one of the useful systems determining position in a way of getting information for latitude, longitude and height for GPS receiver. This system provides an ellipsoidal model of the Earth shape. That represent cross-section of the Earth parallel to the equatorial plane. The definition of pseudorange to SV i , where i is the PRN number defined by:

$$i(n) = c[TR(n) - TTi(n)](m)$$

where c is the speed of light (299.792.458 m/s), $TR(n)$ denotes the receive time corresponding to epoch n of the GPS receivers clock (seconds) 6, and $TTi(n)$ is the transit time based on the SV i clock (seconds).

It can be concluded that if the receiver baseband process can extract the SV transmit time from the code tracking loop, then it can provide a pseudorange measurement. The precise transmit time measurement for SV i is equivalent to its code phase offset with respect to the beginning of the GPS week. There is a one-to-one relationship between the SV i replica code phase and the GPS time. The replica code state corresponds to the receiver's best estimate of the SV transmit time. The receiver baseband process knows the code state because it sets the initial states during the search process and keeps track of the changes in the code state thereafter. The receiver baseband code tracking loop process keeps track of the GPS transmit time corresponding to the phase state of the code NCO and the replica PRN code generator state after each code NCO update.

It does this by discrete integration of every code phase increment over the interval of time since the last NCO update and add this umber to the code accumulator. The combination of the replica code generator state and the code NCO state is the replica code state. Since the code phase states of the PRN code generator are pseudorandom, it would be impractical to read the code phase state of the PRN code generator and then attempt to convert this nonlinear code state into a linear GPS time state by a table lookup.[1]

2.3 Geodetic system

The most used system in navigation is the geodetic system. There is of course difference between Cartesian and geodetic system. The main difference is that geodetic system defines points above the earth surface in terms of longitude, latitude and altitude. The first one, longitude represents rotational angle between prime Meridian and the measure point in range $-180^\circ - 180^\circ$. The role of latitude is to measure the angle between the equatorial plane and the normal reference ellipsoid that passes through point. The last one is altitude or somewhere marked also as height. This is the vertical distance between the point we are looking and the reference ellipsoid. Important thing to write in this part is that adopted geodetic latitude differs from the geocentric latitude. Geocentric latitude represents angle between the equatorial plane and a line from the mass center of the earth.

The spherical coordinates of a point in the Longitude/Latitude/Height convention can be obtained from its Cartesian coordinates (x, y, z) by the formulae:

$$\begin{aligned} x &= r \sin \theta \cos \varphi, \\ y &= r \sin \theta \sin \varphi, \\ z &= r \cos \theta \end{aligned}$$

where r denotes radius, ϑ is inclination and φ is azimuth.

2.3.1 Transformation between Geodetic and ECEF coordinates

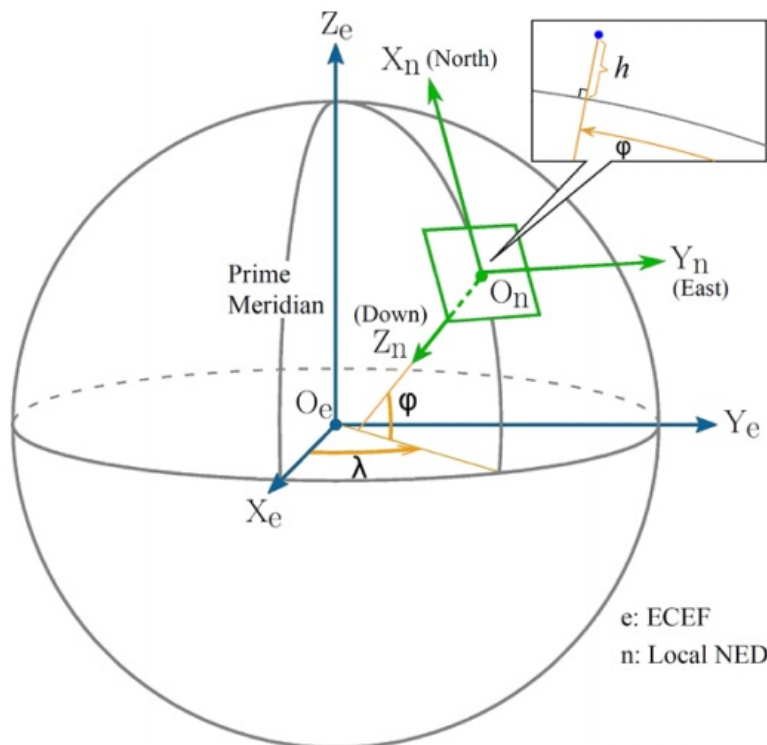


Figure 2.1. Geodetic, ECEF and NED coordinate systems.

The ECEF or Earth-Centred Earth-Fixed system is specific coordinate system because of its rotation with the earth around its spin axis.

- The origin (denoted by O_e) is located at the center of the Earth.
- The z-axis (denoted by Z_e) is along the spin axis of the Earth, pointing to the north pole.
- The x-axis (denoted by X_e) intersects the sphere of the Earth at 0 latitude and 0 longitude.
- The y-axis (denoted by Y_e) is orthogonal to the Z- and X-axes with the usual right-hand rule.

If we want to change geodetic coordinate points to ECEF[5] points we can do it by the following equations:

$$P_e = \begin{pmatrix} x_e \\ y_e \\ z_e \end{pmatrix} = \begin{pmatrix} (N_E + h) \cos \varphi \cos \lambda \\ (N_E + h) \cos \varphi \sin \lambda \\ [N_E(1 - e^2) + h] \sin \varphi \end{pmatrix},$$

where e denotes the first eccentricity and N_e is the prime vertical radius of curvature given by:

$$e = \frac{\sqrt{R_{Ea}^2 - R_{Eb}^2}}{R_{Ea}} = 0.08181919,$$

$$N_E = \frac{R_{Ea}}{\sqrt{1 - e^2 \sin^2 \varphi}}$$

- $R_{Ea} = 6.378.137$ [m] - the semi major axis
- $f = 298.257223563^{-1}$ [-] - the flattening factor
- $R_{Eb} = R_{Ea}(1 - f)$ [m] - the semi minor axis

Chapter 3

Description of The Traffic Collision Avoidance System (TCAS)

Today, people use aircrafts every day in a way of transport from one point to another one. We know that number of airplane connections is very huge and it is growing day by day. People want to feel safe when flying to their destinations.

Exapt the comfort that companies offer to their passengers, they also want to be sure that they will safely reach their destination. The sky mainly above Europe or America is full of airplanes transporting passengers to different cities. To be sure that all of them will reach their destination, people on the ground who work for Air Traffic Control (ATC)¹), follow them every day on their radars giving them instructions how to fly safely. ATC is, of course, one of the most important links in the chain of airplane transport either civil or freight transport, but it is not only one system that works on aircraft safety. Another well-known system is so called Traffic Alert and Collision Avoidance System (TCAS in the sequel).

“The Traffic Alert and Collision Avoidance System (TCAS), also known as the Airborne Collision Avoidance System (ACAS) is an airborne system designed to increase cockpit awareness of nearby aircraft and service as a last defense against mid-air collisions.“ [6] This system installed in an airplane works in the way to search for potential risks around the airplane that may collide with it and gives information to the pilot. TCAS works on its own, which means it is not connected to or in any way controlled by a pilot or ATC from a ground. As I mentioned at the beginning of this thesis, aviation travel is really safe despite its huge usage in everyday life. It is estimated that who would fly continuously, that the average value of waiting for an accident for a single passenger is around 11,000 years. TCAS and ATC are layers in the defense of safe air transport. But they do not have the same tasks to deal with. ATC works more on organizing and managing aircrafts in their flights while TCAS works independently and most of the time in shadow being sure that two airplanes will not be closer than three miles apart laterally or 1000ft vertically. TCAS is system that not only detects potential threat but it works also on communication between two airplanes and tries to assist in the resolution of it. Assistance of TCAS is a way how to recommend both pilots what they have to do to deal with situation. Even if pilots in their cabin are not aware of close contact with another airplane, TCAS will interrupt them by alarming them and literally giving them instructions what to do. Thanks to that TCAS is very usable in situations that even pilots do not see on time, and time in this situations can be of vital significance to save many lives. It is also important to say that TCAS stops working when airplane is lower than 1000ft above the ground because in this case ATC takes control to manage airplane to land safely. This is because on this altitude there are lots of other aircrafts close to each other, but they are not at risk for collision, and

¹) According to CollegeGrad: Air traffic controllers coordinate the movement of air traffic, to ensure that aircraft stay safe distances apart.

TCAS does not know how to regulate and recognize which airplane is at risk and which is not.

3.1 Method of TCAS

We already mentioned that time is very important factor in working of TCAS system. It is important because it can save lives but it is also important in realizing how TCAS works. By sending interrogation to other aircrafts transponders and waiting its response, in a way how radar replies, TCAS calculates time difference between the interrogation and the reply. Knowing time difference, distance between two aircrafts is calculated.

Additionally, the reply from the second airplane contains its altitude so the first airplane gets important information about the flight of the second airplane. Three different steps are included in this process. The first step in this link is of course observation and searching for a potential risk around the airplane. It is called surveillance, and it works in a way of collecting state information about intruder airplane. There is a lot of sensors on both the airplanes that are searching for an airplane that is close to them. After those sensors detect potential intruder they pass the information to a set of algorithms to get clear picture if that aircraft is really at risk of collision or not. This process is based on air-to-air interrogation broadcast every second from antennae on TCAS aircraft using frequency 1030 MHz and waveform as ground-based air traffic control sensors. The reply of transponder that gets this info is on another frequency 1090 MHz. From the fact that TCAS and ATC use and share the same frequencies we can say that significant problem can occur. In aircrafts that fly today, there are two types of transponders that can be installed. Mode S transponders and older Air Traffic Control Radar Beacon System (ATCRBS in the sequel). Difference between these two transponders is that Mode S contains unique 24-bit identifier and ATCRBS do not have unique addressing capability. As the number of airplanes grows it is logically that number of TCAS systems with them also increases. The problem can be rise of TCAS units that can interrogate transponders in airplanes. To deal with this problem, transponders now spend most of their time to respond to TCAS and they decrease their response to the ground service. Frequencies that are shared between TCAS and ATC can split and make a fallacy about some problem TCAS uses interference-limiting algorithms to reduce competition between them. It works in the way that TCAS determines the number and distribution of other system unites in its nearness. Knowing this, TCAS reduce its maximum transmit power and on that way decrease the impact on the victim transponders and ground services.

There are also other different things that can affect TCAS to follow intruders. It is possible that older transponders do not have a system that sends with replay information about their altitude when they are asked by other TCAS system. But today it should not be such often problem, because large aircrafts and airplanes that are flying above 10,000ft must have the newest systems that replays with information about altitude. Another problem that can occur is that the second aircraft does not have transponder at all or it does not work. In this situation, it cannot be detected or tracked by TCAS system at all. Then the pilot of airplane which does not have transponder is required to be responsible and to avoid collision by visual observation. Next step in this process how TCAS works, is Threat Detection and Display. The beginning of this step works in a way that TCAS algorithms starts by classifying intruders into one of four discrete levels. Main task here is to predict and to project airplane position in the

next few seconds. This is done by a simple linear extrapolation based on the aircraft current velocity. After this is done, algorithm uses different key metrics including the vertical and slant range distance between two airplanes. Very important parameter used in this calculation is tau parameter. It describes the time until the closest point of approach between aircrafts. In the cockpit, the pilot gets information in a form of arrow him shows them what another airplane is going to do or is doing just in the moment. If the arrow is turned down, it indicates that another aircraft is lowering its altitude. Climbing of another aircraft is shown so that the arrow is turned up. This can help the pilots to take visual observation of another airplane. “Distant, non-threatening aircraft appears as hollow diamond icons.”[6] If the intruder is close but not still a threat solid diamond is shown by TCAS in cockpit. In the situation where collision is probably to occur in the time interval between 20 to 48 seconds, TCAS provides traffic advisory in the cockpit. This means that system alert pilots by sending a voice message to them that sounds “traffic, traffic”. Change of traffic icon into a solid yellow circle represents the same message. This provides possibility to pilots to start communication with ATC on the ground requesting more information. Display also shows to the pilots a target vertical rate and command “climb, climb”. Next step for the pilot is to turn off autopilot and reach the recommended vertical height manually.

The last step in this TCAS work process is called Threat Resolution. As its name says, in this step TCAS finds the way how to deal with possible collision. Here algorithms that are in TCAS system are very important, because they are the main link in searching and maneuvering airplane to avoid problems with another airplane. According to information that they already received, the algorithms must decide whether aircrafts have to climb or descend. TCAS shows only these two ways for solving the problem. Next step is that the algorithm tries to find out how fast the pilot has to react. “Each template assumes a 5 sec delay before a response begins, followed by a 0.25 g vertical acceleration until reaching a target vertical rate of 1500 ft/min.” ¹⁾ This works on a prediction that another airplane will not change its course or altitude. If we talk about two airplanes that are both equipped with TCAS system, the system will regulate through Mode S link that two of them will not choose the same instruction. For example, it will ensure that one will go down and another will climb up. But if such situation happens, aircraft with the lower numerical valued Mode S address has priority and will continue to display for example to climb. Another airplane will get the information to descend because of its higher Mode S address. It can also happen that the aircraft gets inappropriate information about what to do. This is possible because TCAS has 1 Hz update rate and filtering lags. This is solved in the way that if there are big uncertainties TCAS refrains from issuing an RA. If after some time period TCAS recognizes that the situation is becoming worse, it can reverse the sense from climb to descend or the other way around. Every information is going through Mode S link. It is important to say that this reversal messages from TCAS cannot be so safe and they could just create problems if the pilots do not react fast enough. The block diagram shows step by step how usual TCAS system should work:

¹⁾ See References: The Traffic Alert and Collision Avoidance System, 2007.

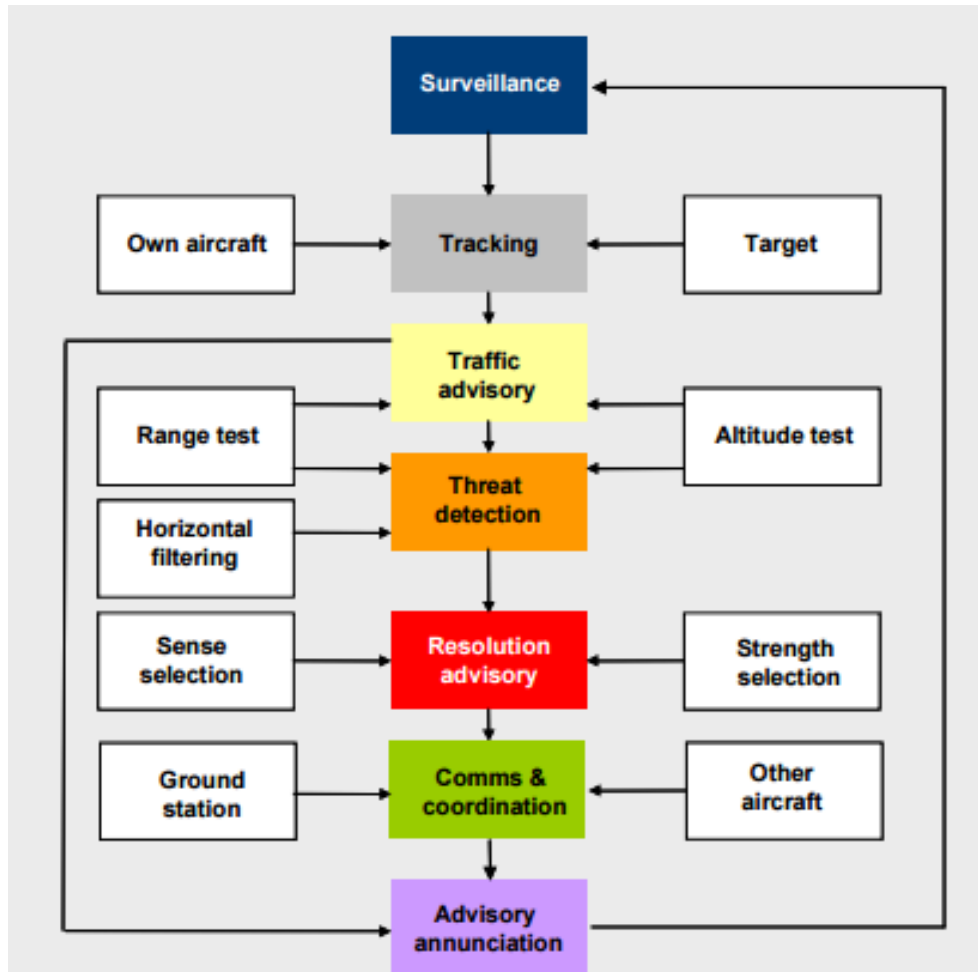


Figure 3.1. TCAS logic function

At the end of this chapter, we can say that TCAS operates in a very complex environment. Even though it is proved that TCAS gets faster and much more correct data about altitudes of airplanes around them, they cannot take in consideration other facts that also affect the flights. ATC and pilots are people that make mistakes, TCAS is system which causes mistakes that are put on the minimum, but works in ATC and the pilots can see much more further information about some flight than TCAS. “TCAS is extremely successful in providing a last-resort safety net, and does not necessarily need to operate perfectly to be effective. Still, it is important to identify situations where TCAS may have difficulty and, if possible, to modify the logic to better handle such circumstances.”[6]

Chapter 4

Description GPX

GPX (the GPS eXchange Format) [7] is a data format for exchanging GPS data between programs and for sharing GPS data with other users. Unlike other data files, which can only be understood by the programs that created them, GPX files actually contain a description of what is inside them, allowing anyone to create a program that can read the data within. GPX is a light-weight XML data format for the interchange of GPS data between applications and Web services on the Internet. It can be used to describe waypoints, tracks and routes. The format is open and can be used without the need to pay license fees. Its tags store location, elevation and time, and can in this way be used to interchange data between GPS devices and software packages. Such computer programs allow users, for example, to view their tracks, project their tracks on satellite images or other maps, annotate maps, and tag photographs with the geolocation in the Exif metadata.

Example of GPX code:

```
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>

<gpx xmlns="http://www.topografix.com/GPX/1/1"
xmlns:gpvx="http://www.garmin.com/xmlschemas/GpxExtensions/v3"
xmlns:gpvtpx="http://www.garmin.com/xmlschemas/TrackPointExtension/v1"
creator="Oregon 400t"
version="1.1"xmlns:xsi="http://www.w3.org/2001/XMLSchemainstance"
xsi:schemaLocation="http://www.topografix.com/GPX/1/1
http://www.topografix.com/GPX/1/1/gpx.xsd
http://www.garmin.com/xmlschemas/GpxExtensions/v3
http://www.garmin.com/xmlschemas/GpxExtensionsv3.xsd
http://www.garmin.com/xmlschemas/TrackPointExtension/v1
http://www.garmin.com/xmlschemas/TrackPointExtensionv1.xsd">
<metadata>
<link href="http://www.garmin.com">
<text>Garmin International</text>
</link>
<time>2009-10-17T22:58:43Z</time>
</metadata>
<trk>
<name>Example GPX Document</name>
<trkseg>
<trkpt lat="47.644548" lon="-122.326897">
<ele>4.46</ele>
<time>2009-10-17T18:37:26Z</time>
</trkpt>
</trkseg>
</trk>
</gpx>
```

[ex: route]

```

<gpx>
<rte>
<name> xsd:string </name> [0..1] ?
<cmt> xsd:string </cmt> [0..1] ?
<desc> xsd:string </desc> [0..1] ?
<src> xsd:string </src> [0..1] ?
<link> linkType </link> [0..*] ?
<number> xsd:nonNegativeInteger
</number> [0..1] ? <type> xsd:string </type> [0..1] ?
<extensions> extensionsType </extensions> [0..1] ?
<rtept lat="47.644548" lon="-122.326897">
<name>rtename</name>
</rtept>
</rte>
</gpx>

```

GPX data used in this project are real and they were created by different airplanes.[8] Because our focus is on proving that we can predict correct route, it does not depend on the airplanes. It is the same for us to take either Cessna 172 in consideration or Boeing 737. Their points provided by GPX files are important. It is very hard to find real GPX data for huge airplanes and airplanes that flies for companies, but we managed to find it for smaller airplanes.¹⁾

4.1 Matlab and GPX files

It is also possible to open GPX format[9] file in the program that we will use in this project, called Matlab. There is an example of opening one GPX file in this program: Read waypoints and track log from the sample.mixed.gpx file:

```

wpt = gpxread('sample_mixed');
trk = gpxread('sample_mixed', 'FeatureType', 'track');

```

Display the waypoints and the track log on a webmap:

```

webmap('mapquestopenaerialmap')
wmline(trk, 'OverlayName', 'Track Logs');

```

Important fact to mention is that in this project, we are going to use data from GPX and according to them try to predict next points and compare it with real one[10]. It means that we will not use in consideration any other elements of flight such as weather or wind. These factors are very important for flight and they can affect it very badly, but we are not able to change them so we will ignore them.

¹⁾ All GPX data were taken from the Internet page. See References.

Chapter 5

List of airplanes used in scrutinizing

Because the main aim of this project is to present real life situations of navigating different airplanes from one point to another one, represented through simulations in some computer program, we have to divide airplanes into different groups according to their purpose as being used in the real life.

List of different groups is:

- Single-engine propeller airplanes – Cessna 172 and Piper-PA28
- Business jets – Learjet 45 and Gulfstream G100
- Medium transport aircrafts – Airbus A320 and Boeing 737
- Big transport aircrafts – Airbus A380 and Antonov An-225
- Military aircrafts – MIG 35 and A-10 Thunderbolt

In each group, there are two different airplanes that will be taken in observation for this project. Now, we have to precisely determine which information we need from these airplanes to make this project as good as we can. This is the list of technical performances of each of these airplanes that we will need:

- Take-off speed
- Maximum speed
- Normal (Travelling - **Cruising**) speed
- Landing speed
- Grade ability
- Turning radius
- Acceleration

Note that almost everything is in some way connected to speed of the airplanes. This collected information will be presented in a manner of table where each of the airplanes will have its own info. Another important thing is that the speed will be measured in kilometers per hour. Table with data is below.

5.1 Technical information about airplanes

Tech. Info	Single engine propeller airplanes		Bussines jets		Medium transport aircrafts		Big transport aircrafts		Military aircrafts	
	Cessna 172	Piper PA28	Learjet 45	Gulfstream G100	Airbus A320	Boeing 737	Airbus A380	An-225	Mig 35	A-10
Take off speed	120	122	370	350	275	250	280	320	300	292
Maximum speed	257	235	857	895	903	912	1180	850	2400	833
"Crusing" speed	210	195	820	560	840	813	1078	800	200	560
Landing speed	83	86	321	290	185	250	250	250	250	220
Grade ability	-	60[s] - 200 [m]	-	-	-	-	-	-	[s]-330[m]	-
Turning radius	160/100[m]	195/95[m]	11.5 [m]	-	-	180[s]/360°	53 [m]	-	-	-
Acceleration	20[s]/120	20[s]/120		20[s]/220					14[s]/500	-

Figure 5.1. Technical information about airplanes. All data in which is not a mark, are introduced in kilometers per hour [km/h].

We can see that there are missing information in this table[11],[12]. Many airplane companies secretly keep their technical information about airplanes, and it is very hard to get them. We have to point out that this will not affect our further work in any way.

5.2 Description of the considered airport: Vaclav Havel (Ruzyne) Prague

Information written in the following text concerning Vaclav Havel airport in Prague[13] are important in the way that they give technical support of the project. For example if there is needed, to make solution that is suitable for this airport these data has to be taken in consideration. Even though the project is focused more on airplanes, it cannot be made in correct way without any connection with ground. Here, the focus is given on ATC and all the people that are in charge for safety on, near and around airport. It is not possible to make any changes in airplane system if air traffic control centers do not approve it. Coordinates of airport are written together with number of runways and their characteristics because it can be useful for airplanes using systems for prediction and knowing that there is possibility to get feedback from another aiplane from different runway for example. In the last section, *Future work*, there is placed instrument approach chart map of the Vaclav Havel airport. This map consists of many important infomation for pilots in order to land or take off safety. It contains data for closing to the airport and instructions how to do it. Also it shows angles and speed that has to be used by airplanes. Different airplanes need to prepare for that on time, but more about those things will be described in last section.

- Number of runways: 2
 - RWY 06/24
 - RWY 12/30

- Number of terminals: 3
 - Non-Schengen area flights
 - Schengen area flights
 - General aviation
- The number of terminals for handling cargo: 2

Obstacles in the area of the airport :

- Obstacles at the airport
 - the radar on the building of the technical block - TWR
 - Coordinates: 050° 06' 22,33"N; 014° 16' 01,30"E,
 - Height above sea level: 1352 ft/412 m
 - Marking: LGTD
- Obstacles in the airport's surroundings:
 - None

Geographical coordinates of the reference point of the Airport and its location:

- Coordinates:¹⁾
 - 050° 06' 03" N
 - 014° 15' 36" E
 - Located on the axis of RWY 12/30 : 1936 m from the threshold of runway 30
- Distance and Direction of the Airport from the City:
 - 10 km in the direction 277° from Prague castle
- The airport's height above sea level:
 - 380 m / 1247 ft
- Reference Temperature: - 23.6°C July
- The Area of the Airport:
 - total area of the airport: 9 200 000 m²
- The area of movement and road surfaces:
 - approx. 2 000 000 m²
- The circumference of the area of the airport (the length of the enclosure):
 - 25 km

Runways: ²⁾

- RWY 06:
 - 3715 x 45 m equipped for CAT I
- RWY 24:
 - 3715 x 45 m CAT III B operations up to visibility 50 meters
- RWY 12/30:
 - 3250 x 45 m equipped for CAT I operations

¹⁾ According to WGS-84 system

²⁾ The take-off and landing of jet aircraft with the maximum take-off weight of more than 9 ton must customarily be performed from RWY 06/24.

Chapter 6

Methods for determining position of airplanes in Matlab

6.1 Linear extrapolation

The first method that we will explain how it works is a linear extrapolation. The best way to show it is to use screen shots of program and describe them. Important thing in this part is that we have to have further information about the airplane and its flight. Here, it is not enough just to know points of its trajectory, but we have to know either its speed or velocity in the moment when we lose contact with the airplane. For converting points from degrees to Cartesian coordinate system we will use simple code in Matlab:

```
for (i = 1:length (Latitude(:,1)))

[xyzpos(i,1),xyzpos(i,2),xyzpos(i,3)] =
  lla2ecef (Latitude(i,1),Latitude(i,2),Latitude(i,3));

end
```

Latitude is only the name of workspace script where we will put data for observation. The following code will show program with comments:

```
function [Vx, Vy, Vz] =racunanjeBrzina(x, y, z)

% copping values of x in temporary variable vector x1
x1 = x;
% deleting first member of vector x1
x1(1) = [];

% copping values of x in temporary variable vector x2
x2 = x;
a = length(x);
% deleting last member of vector x2
x2(a) = [];

% calculating velocity projection on x axis as difference on x1 and x2
% this is possible because vectors have same length (length(x) - 1)
% and first member of x1 is x(2) and first member of x2 is x(1)
Vx = x1 - x2;

% repeating process for y axis
y1 = y;
y1(1) = [];
y2 = y;
a = length(y);
```

```

y2(a) = [];

Vy = y1 - y2;

% repeating process for z axis
z1 = z;
z1(1) = [];
z2 = z;
a = length(z);
z2(a) = [];

Vz = z1 - z2;

% plotting coordinates
plot3(x,y,z);

return

```

The Code above shows that we can calculate speed of the aircraft. This will help us later to get its position. From the comment above you can see how and what each line of code does. The following code shows the calculation of an acceleration. The acceleration is one of imported needs in the sequel. It is done similarly to calculation of the speed.

```

function [Ax, Ay, Az] =racunanjeUbrzanja(Vx, Vy, Vz)

% copping values of Vx in temporary variable vector Vx1
Vx1 = Vx;
% deleting first member of vector Vx1
Vx1(1) = [];
% copping values of Vx in temporary variable vector Vx2
Vx2 = Vx;
% deleting last member of vector Vx2
a = length(Vx);
Vx2(a) = [];

% calculating acceleration projection on x axis
%as difference on Vx1 and Vx2
Ax = Vx1 - Vx2;

% repeating process for y axis
Vy1 = Vy;
Vy1(1) = [];
Vy2 = Vy;
a = length(Vy);
Vy2(a) = [];

Ay = Vy1 - Vy2;

% repeating process for z axis
Vz1 = Vz;
Vz1(1) = [];
Vz2 = Vz;
a = length(Vz);
Vz2(a) = [];

```

```
Az = Vz1 - Vz2;

return
```

Now, when we have the information about the flight, we are able to calculate its position. Of course, the position will be in Cartesian coordinate system in a 3D view. It means we will have all three values x, y and z.

```
function [xi, yi, zi] = racunanjePozicije(Vx, Vy, Vz, Ax, Ay, Az, x,y,z)

% calculating velocity projection on x axis based on
% known acceleration and known velocity
Vxi = Vx(length(Vx)) + Ax(length(Ax));

% calculating position projection on x axis based on know
% position and calculated velocity
xi = x(length(x)) + Vxi;

% calculating velocity projection on y axis based on known
% acceleration and known velocity
Vyi = Vy(length(Vy)) + Ay(length(Ay));

% calculating position projection on y axis based on know
% position and calculated velocity
yi = y(length(y)) + Vyi;

% calculating velocity projection on y axis based on known
% acceleration and known velocity
Vzi = Vz(length(Vz)) + Az(length(Az));

% calculating position projection on z axis based on know
% position and calculated velocity
zi = z(length(z)) + Vzi;

return
```

Last rows of code program shows only command window and how it “calls” everything from other skripts.

```
Coordinate

% extracting coordinates
x = xyzpos(:, 1);
y = xyzpos(:, 2);
z = xyzpos(:, 3);

% calculating velocity projections on x, y and z axis based
% on know positions
[Vx , Vy, Vz ] = racunanjeBrzina(x,y,z);

% calculating acceleration projections on x, y and z axis based
% on calculated velocities
[Ax , Ay, Az ] = racunanjeUbrzanja(Vx,Vy,Vz);
```

```
% calculating future position based on know position,
% velocity and acceleration
[xi, yi, zi] = racunanjePozicije(Vx, Vy, Vz, Ax, Ay, Az, x, y, z);
```

6.1.1 Test of linear extrapolation - Matlab

After we had shown the code of the program and described each line what it should do, we have to test it and see whether it is correct and on which percentage of correction it works. To do it in the best way, we will show it one more time through print screens. Data that will be used in observation:

No.	alpha	beta	gamma
1	43.922657	-88.726177	259.1
2	43.922388	-88.726898	258.2
3	43.922140	-88.726406	257.6
4	43.921883	-88.726516	257.2
5	43.921626	-88.726625	258.0
6	43.921372	-88.726734	259.7
7	43.921099	-88.726886	262.4
8	43.920800	-88.727042	266.0

Table 6.1. Point values. All values are presented in kilometers [km].

In this table, we have some of the points of a trajectory Cessna airplane. Note that this data is in a degree way of represent. Matlab will make Cartesian system from it and it will try to find its next point. To test the program, we are going to use the first 7 points. According to what we will get in the end of process, we will be able to compare it with the last 8. point.

```
xi = 4.6037e+06
yi = -4.3974e+06
zi = -3.8934e+05
```

Using this program, point value that I got can be seen above, and from the figure 6.1, the 8.point of the trajectory has values:

	1	2	3
1	4.6082e+06	-4.3938e+06	-3.7764e+05
2	4.6076e+06	-4.3942e+06	-3.7934e+05
3	4.6070e+06	-4.3947e+06	-3.8091e+05
4	4.6065e+06	-4.3951e+06	-3.8254e+05
5	4.6059e+06	-4.3955e+06	-3.8416e+05
6	4.6054e+06	-4.3960e+06	-3.8577e+05
7	4.6046e+06	-4.3966e+06	-3.8750e+05
8	4.6039e+06	-4.3972e+06	-3.8939e+05
9			

Figure 6.1. Point values of airplane trajectory in Cartesian system

As seen from Figure 6.1 I got almost the same points. Now I can calculate how precisely I was using this program. As it has been shown, the difference between the real points (row 8. at the figure 6.1) and the point that I got is really small. In other words, I can say that this program of prediction of the next point of the airplane trajectory works 99.99% correctly. It is important to write that I use time space of only 1 second. The reason for this is that GPX system saves point values in this time scale.

6.2 Exponential extrapolation

Here we will show how this method works. It is also written in Matlab. This method is not as complicated as the linear extrapolation was. An exponential extrapolation works according to the data we have, and functions in Matlab, which will make a trend curve that fits the best. Here, we will not be able to find correct point, but we will be able to find the part of missing trajectory. Of course it will not be as precise as the first method but it will help us to find out faster the longer trajectory than only one point. For better comparison with the first method, we will use the same data as we used for the linear extrapolation. Also we will have to transfer it to Cartesian coordinate system and then to extrapolate it with Matlab function.

Again, we will use print screens of the program code to show how it works. The first figure here shows the code of the program and the next figure shows its output.

```
plot(Latitude(:,1), Latitude(:,2), 'bo')
hold on
plot(data, 'r')
plot(data, 'predobs')
data

data =
Linear model Poly2:
data(x) =p_1x^2 + p_2x + p_3
Coefficients (with 95% confidence bounds):
p_1 = -35.26 (-61.56, -8.966)
p_2 = 3098 (788.11 5408)
p_3 = -6.0814e+04 (-1.189e+05, -1.741e+04)
```

Because Matlab has a possibility to calculate what way trend curve will fit in our prediction best, we use Poly2 function to check it (it was done in a consultation with the mentor). As seen from Figure 6.2. Matlab has implemented function that is used to draw a curve that will predict flight trajectory. Also, using this program I am able to construct the range of possible mistakes using function predobs. The output of this program looks like this:

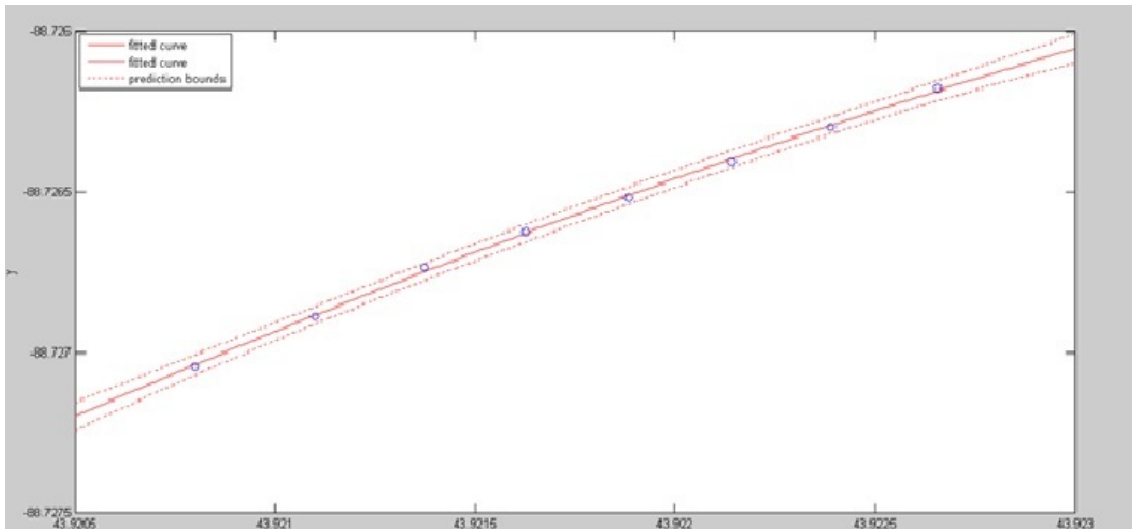


Figure 6.2. Extrapolation method in Matlab

Red thick curve shows best fit trajectory according to the data we have. Points that were used here are marked with blue circles. If we change the range of this window we could get longer predicted trajectory. Two red thin dotted lines show prediction area where other points can be possibly shown up. As we can notice, area between those two lines is much narrower in the case of data points, than in the area with predicted points.

6.3 Conclusion of evaluation two different methods

As the conclusion we can say that both the programs work well. User first have to decide what he/she is looking for. If person wants to calculate the correct point of position of airplane in some correct time it will use the first program. On the other hand, if user wants to know predicted trajectory of flight the second program is better. Both the programs can help in different situations. They are not complicated, especially the second one. As I wrote at the beginning this thesis, it is only one of four parts that will be in bachelor work.

Chapter 7

Altitude prediction of airplane trajectory

In this chapter, I am going to show, how we can predict on which height the airplane is cruising.

This method is different from the previous ones, because I will show trajectory that I know in 3-dimension and predict altitude of the next points of the flight. For the first time, this project is going to have not only x,y coordinates but also the third one z . This coordinate presents altitude or height of the point where the airplane was or is going to be. Using the real flight trajectory with all three values of coordinates I am going to predict the next z points using again Matlab program. Because I need a sample of the flight trajectory where airplane will descent its altitude I decided to take 100 points from the beginning of landing. The following picture shows how that sample with all x,y and z points look like:

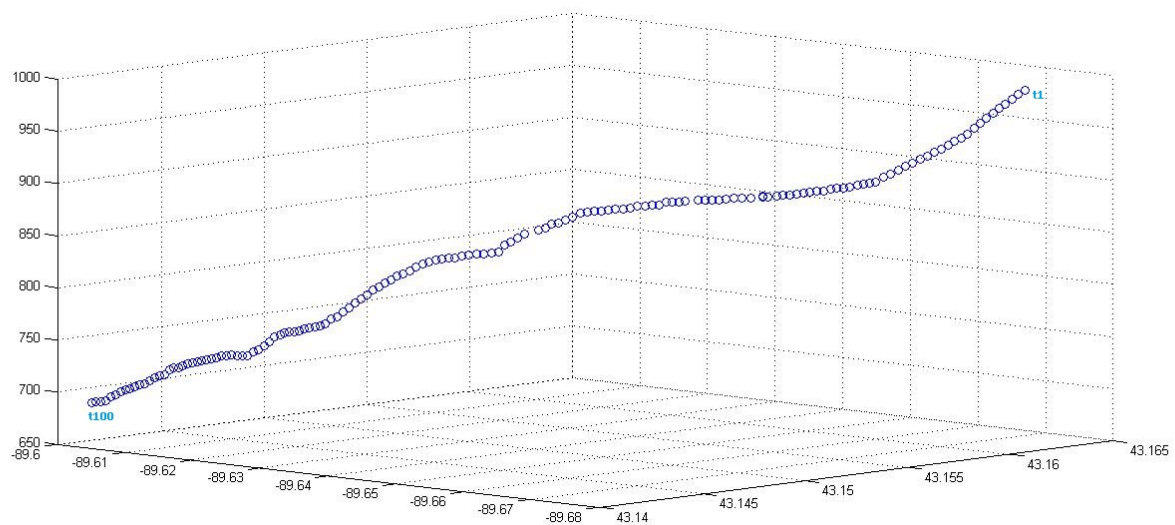


Figure 7.1. 3D airplane trajectory - landing. It starts at $t1$ and it is observed till $t100$ where the trajectory ends.

Next step is to write a code in Matlab that will use the values of x and y , and according to them calculate prediction of z in the future. Note that I have to write code which will be able to calculate not only using few first points of trajectory, but also one which will be able to do that for any points from a flight route. The first thing which had to be done was to transfer the points from longitude, latitude and altitude values to the x,y and z coordinates of Cartesian system. All points on the graph above are in Cartesian values. For the next move and making prediction from some points, the Matlab function called *scatteredInterpolant* will be used.

There are few different methods for the function *scatteredInterpolant*, such as *linear*, *nearest* and *natural*. After testing all of them it is concluded that the best fit method

for our problem is *linear* one. This method works in the way that it uses the already known points x and y , following them and according to the previous z values, calculate and show the predicted points of z .

Part of the program code looks as follows:

```
scatter3(x1,y1,z1,'MarkerEdgeColor','b');
hold on;
for k=[10]
    F = scatteredInterpolant(x1(1:k),y1(1:k),z1(1:k), 'linear');
    zt = F([x1(k+1:k+10),y1(k+1:k+10)]);
    scatter3(x1(k+1:k+10),y1(k+1:k+10),zt, 'r','fill');
end
hold on;
```

Output of this program will be the points z , but because we are following x and y they will be the same. The aim of this part of the project is to predict altitude of airplane if it changes faster than for example when airplane just cruises through the air.

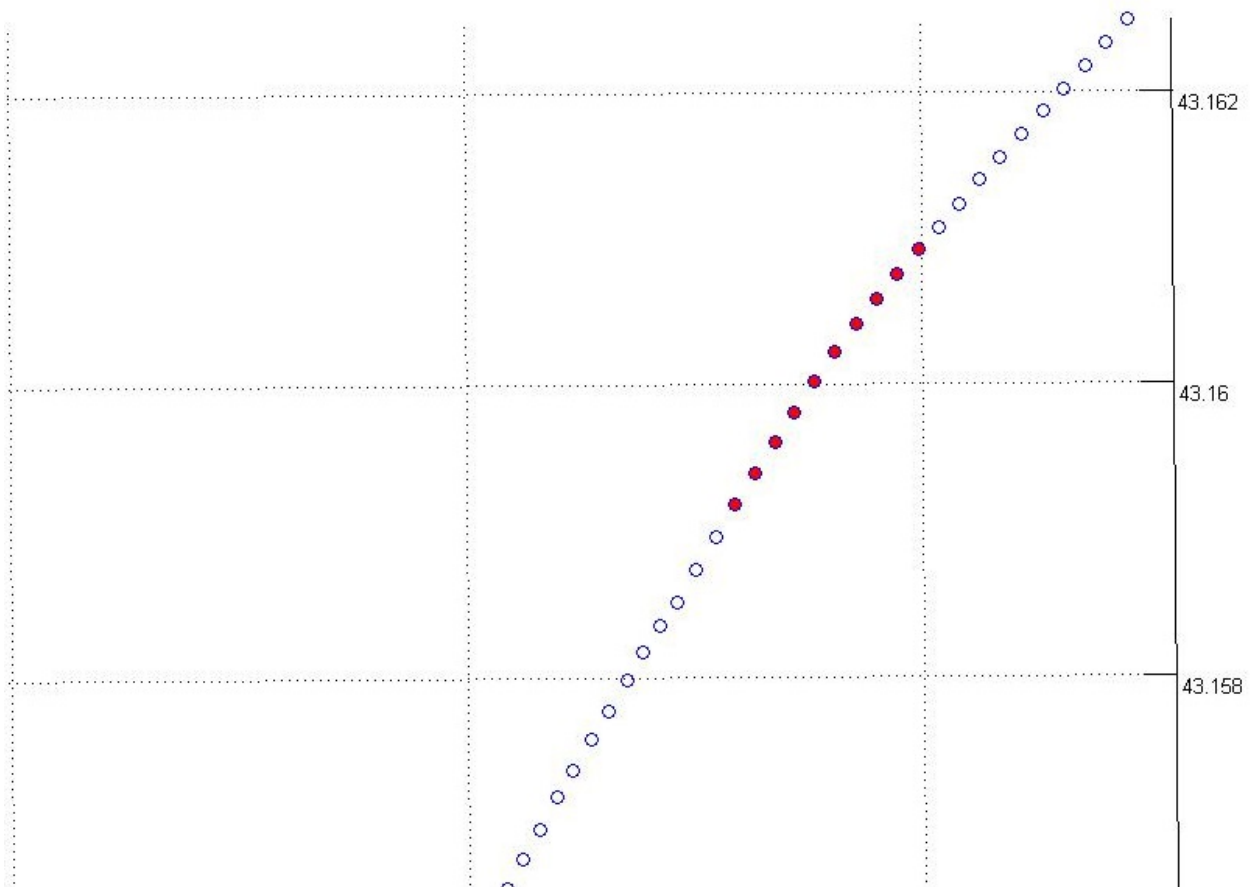


Figure 7.2. Bird-view of 10 points of airplane trajectory.

The graph shows the trajectory from the 2-dimensional view and it confirms what is already written, that the predicted points (marked with red filled circles), follows x and y coordinates. In this case using the previous 10 points of the real trajectory, the next 10 are predicted. In this case the prediction of 10 points of z , or the prediction of the next 10 points of altitude will be done.

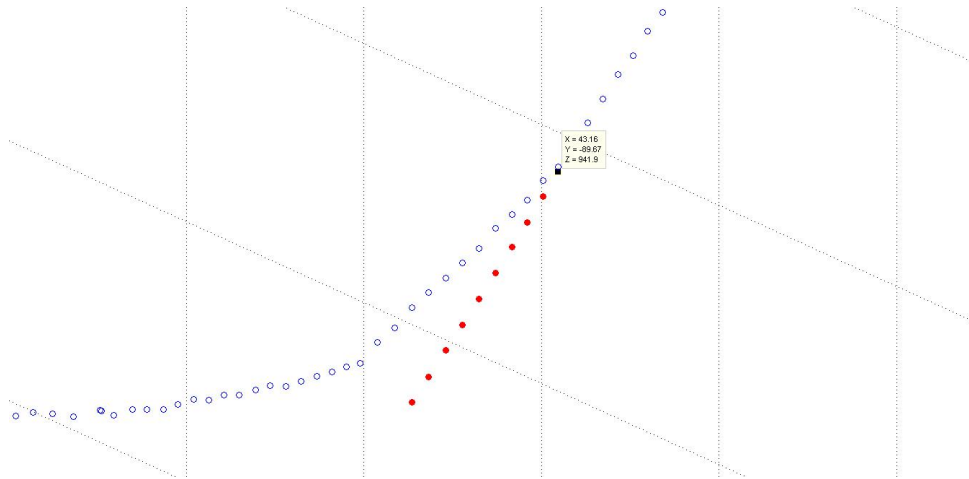


Figure 7.3. 3D-view 10 points.

As seen from the picture above, the values of z that were predicted are different than the real ones. First predicted point has the value of $z=941.9$. But then the value of real z (also drawn on the graph) is $z=943$. It can be said that the first point of prediction has an error but not as big as the next one and the last one. I could include only this point in some calculation but the others have huge difference from the real ones. The biggest problem of this part is that Matlab calculates and predicts points z , on a behave of the all previous points. This means that if there is a trajectory made of 1 000 x,y and z points and if I want to make prediction of z for example after 500 points, this function will consider all 500 previous and make the prediction from them.

Further thing that can be shown here is that if there are more continous points of trajectory the posibility that Matlab will calculate better prediction is greater. For example I took the same flight route with the same points, and I made prediction after 60 points. The result is shown on the following picture:

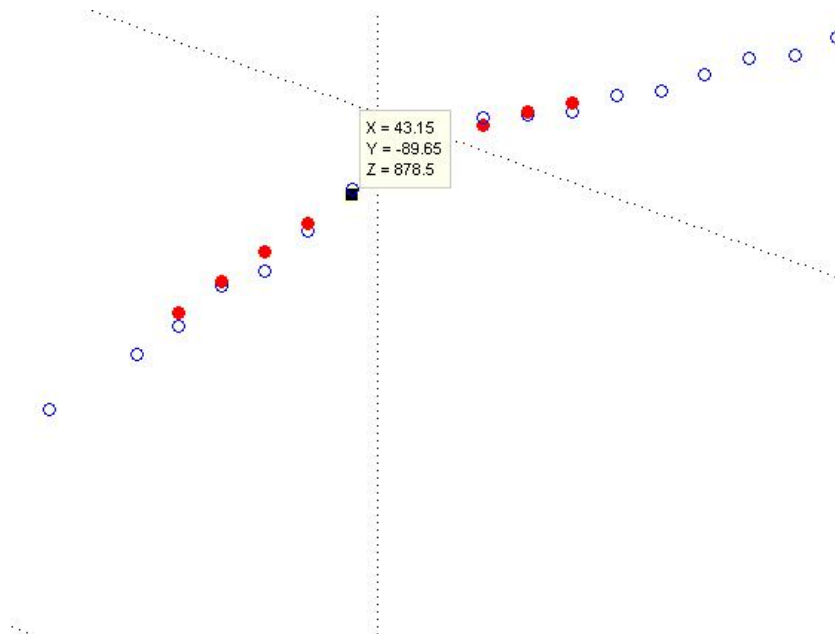


Figure 7.4. 3D-view 10 points airplane trajectory.

Here the difference between predicted value of z and the real one (through all 10 points) is small and I could take them in some calculations. But, I do not want to limit

myself and to use this prediction only when I know that flight route has not changed for some period of time.

This way of solving the problem is not what I need, and it does not solve my problem with the highest percent of correction. The solution that I am looking for is prediction of route with last for example 5 points of all three dimensions. It means that the previous points of the trajectory will not affect my prediction. Here is an example for this situation: If an airplane flies from Dubai to Belgrade, the route will definitely not be a straight line, and if I want to calculate the prediction in the middle of the flight I do not want to consider the points from taking off. In the following chapter, I am going to propose solution for this problem.

Chapter 8

Prediction of Longitude and Latitude of airplane trajectory based on previous points

In this chapter, I am going to show the main part of this project. The previous predictions were small preparations for the following one. In this prediction I am going to implement all previous predictions. In the same ways as before, I will use real airplane trajectory points that are going to be used and then compare them with predicted ones. This part of the project is the most important one, because here I will prove that if the problem I gave to myself has such solution which I expect. If I prove that I can predict with the closest possible error to 0 value, few next points of trajectory I can say that the prediction can be useful. Of course, I am going to compare the first points and the next one with the real and according to the results I will be able to write conclusion. Another important thing is to write that I will be implementing 2D points, which means that I will use only x and y values. Because in the previous chapter, I showed the calculation and prediction of altitude, here I will focus only on these two characteristics. On next graph the trajectory of aiplane considering around 400 x and y points is seen. While there are marked 10 points serving for making the prediction of the next points of the trajectory using code in Matlab program.

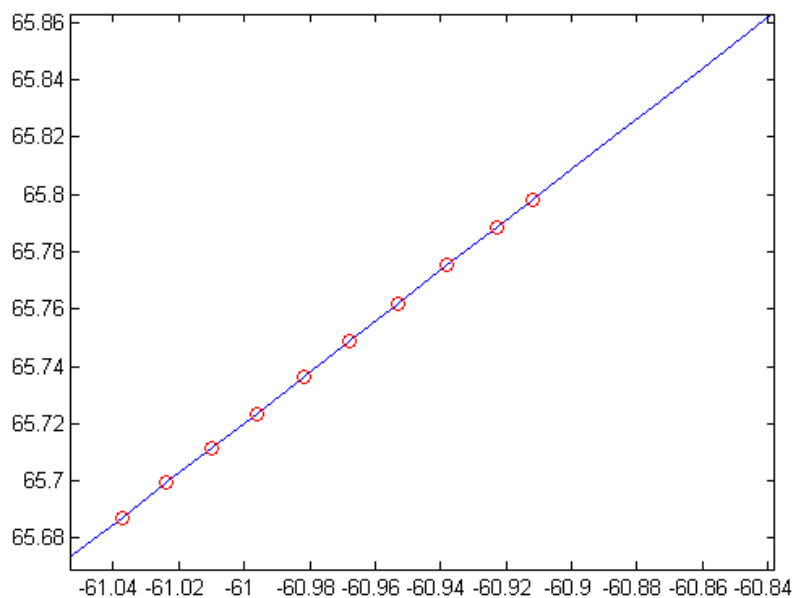


Figure 8.1. 2D-view of 10 points of airplane trajectory

The blue line represents the real trajectory that is led by x and y coordinates. The small red circles are values of x and y that will be used to predict next points. The points which were imported into the program were in the Geodetic system values, and

the first step that Matlab did was transferring them to Cartesian x and y values. If succeed with x and y values, the third dimension will not be problem to add. Because some of the detectors in airplane detect movement of another airplane after more than few seconds they can not be so reliable and can not be taken for true source of its position everytime. Matlab offers many different methods of prediction of the next points. Because in Matlab, predictions of something based on some data are done trough extrapolation, methods that can be used are *previous*, *linear*, *spline* and *pchip* method. The following graph shows the predicted points based on those red points marked on the trajectory:

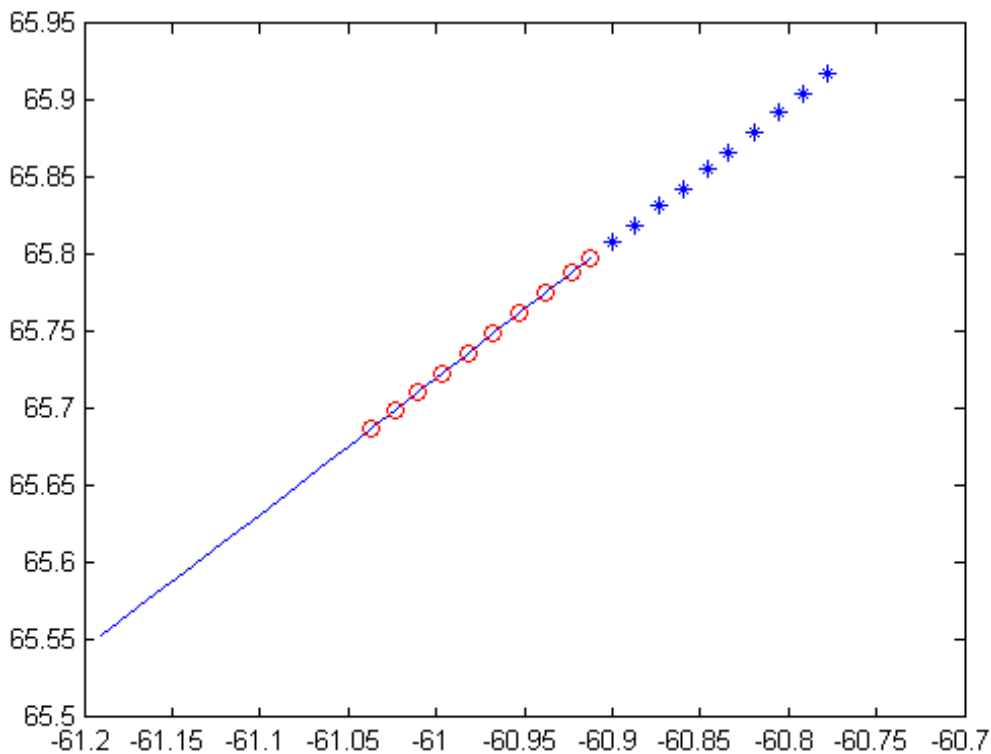


Figure 8.2. 2D-view predicted points

Blue stars represent 10 points that Matlab predicted as the next airplane positions. Here on this graph it is shown how the trajectory would look like without knowing about the real one. But what has to be done here is to import next 10 real points into the graph and see difference between them and the blue stars representing the predicted ones. Because the points and the flight in this range does not contain curves, small difference can be predicted in it and the real points. Lets see how the trajectory and the next real 10 points look like comparing to the predicted ones.

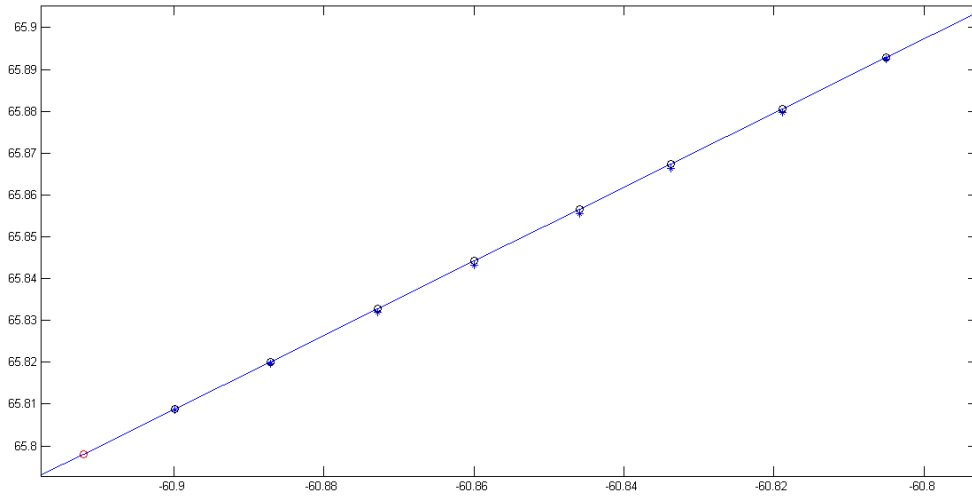


Figure 8.3. 2D-view comparison

The small black circles represent real points to be predict. As seen from the graph difference is really small. Here are the numerical values:

No.	Real value	Predicted value
1	x = -60.8998; y = 65.8088;	x = -60.8996; y = 65.8087;
2	x = -60.8865; y = 65.8203;	x = -60.8870; y = 65.8196;
3	x = -60.8729; y = 65.8320;	x = -60.8732; y = 65.8310;
4	x = -60.8605; y = 65.8447;	x = -60.8580; y = 65.8435;
5	x = -60.8500; y = 65.8550;	x = -60.8580; y = 65.8435;
6	x = -60.8330; y = 65.8662;	x = -60.8351; y = 65.8651;
7	x = -60.8195; y = 65.8858;	x = -60.8188; y = 65.8797;
8	x = -60.8052; y = 65.8929;	x = -60.8059; y = 65.8924;
9	x = -60.7916; y = 65.9047;	x = -60.7913; y = 65.9055;
10	x = -60.7782; y = 65.9167;	x = -60.7779; y = 65.9190;

Table 8.1. Point values.All values are presented in kilometers [km]

As seen from the list above, there is really small difference between the real and the predicted points. It can be also noticed that sometimes there is greater difference in x values, and sometimes the difference is bigger in y values. This is determined by the points before and by changing airplane trajectory in the future. I can write that this solution best fits in solving the problem and that we can use it for some calculation in the future.

Chapter 9

Application of the method in TCAS

In this last chapter, I am going to write more about implementation of the solution in real life situations. As I wrote about TCAS I will try to describe how my method can be useful for preventing potential alarms and collisions of airplane. As TCAS is the system that takes care about safety of airplanes in the sky, I will try to prove that my method can stop situations where TCAS is alarming pilots. It means I will try to implement our calculation and see whether there is a possibility to prevent any situations even before dangerous area of TCAS and is there a faster way to detect the position of the airplane. I am not saying that method I was doing and calculating is safer and better than TCAS. I just want to try to reduce the number of alarms that are made by TCAS systems and in that way help to increase safety on the sky. The main point here is not to allow TCAS to come at that point where it will signalize so called CPA alarm. CPA means Closest Point of Approach and in this situation TCAS makes the worst scenario and alarms everyone in the cockpit about it. Our method is connected with TCAS characteristics called Sensitivity Level (SL).

9.1 Sensitivity Levels and tau

There are two different ways of SL and those are: horizontal protection and vertical protection. Sensitivity Level is divided into 7 different parts. The greater the number of SL is the better protection can be provide.

Graphically it looks like this:

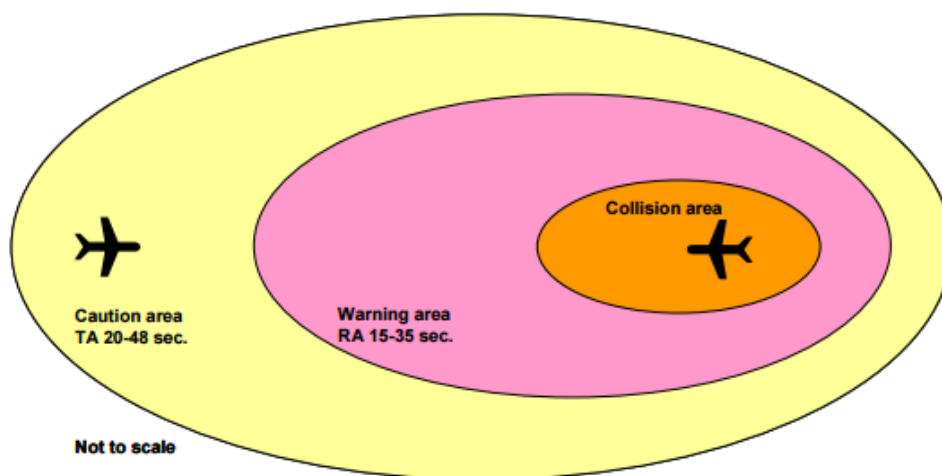


Figure 9.1. SL - Horizontal protection

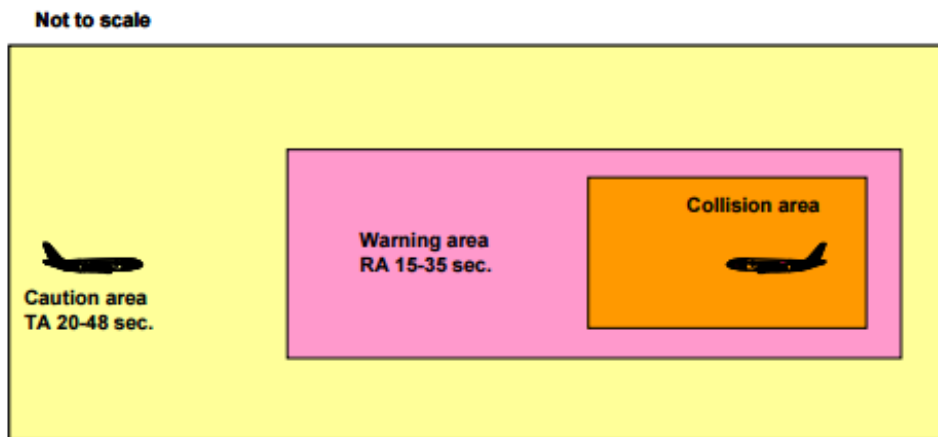


Figure 9.2. SL- Vertical protection

The table representing *SL* levels:

Own Altitude	Sensitivity levels (SL)
Stand-by mode	1
0 – 1000 ft AGL.	2
1000 – 2350 ft AGL.	3
2350 ft AGL – FL50.	4
FL50 – FL100	5
FL100 – FL200	6
Above FL200	7

Table 9.1. Sensitivity Levels

9.2 *time-to-go* process

Another important characteristic in this part of TCAS system is so called *time-to-go* process. This process determines the time that will turn on target aircraft (TA) equipment. This TA is in range from 20 to 48 sec. The range tau and vertical tau are calculated in different ways. The range tau can be get by dividing slant range with closing speed of airplane and then multiplying with 3600. The vertical tau can be calculated by dividing altitude separation by the vertical closing speed multiplying by 60. Because of these calculations and because of the fact that where the rate of closure is very low, an intruder airplane can come very close in range without crossing tau boundaries and leave TA turned off.

9.3 Implementation of our method

Some methods for improving this problem were done, before there, I show how to implement my method of prediction of trajectory.[1] According to the results, I got in the prediction it is possible to make system that will work in shadow of TCAS system. It can work all the time by sending prediction points to TCAS excluding even to notify pilots about it. Only when according to prediction two trajectories would cross each other, then through TCAS, before it gets to dangerous zone can send some information to cockpit. This method can be used before zones where TCAS starts to count time

of intersect of trajectories of two airplanes. Of course, the method I made is not so reliable to be trusted more than anything else but the structure of it can be improved in following work and it can become important thing in the airplane industry. It is very simple, but calculations that I can get from it are very important in terms of airplane position on sky and it can be used for other things in this sphere of life, too. Predictions of where the airplane could go when the electric system in it goes down, is also one of the situations where I can use some of my methods for finding its correct position. If we look on the Internet and study about disappearing of airplanes we can see that their number is not small at all. Using this process we can maybe predict the area where the airplane could fall by knowing some data about its flight.

Chapter 10

Conclusion

In the conclusion, I can say that the goal of this project, set at the beginning of the thesis, was successfully achieved. Idea of making prediction of airplane trajectory based on knowing some data of flight is implemented in practice. Using flight characteristics such as speed or velocity, the next correct point can be calculated with the error of 0.001 percent, as shown in the first part of this thesis. Further, the methods for finding altitude of the airplane knowing its longitude and latitude was made successfully. If the user needs to know predicted height on which airplane will fly in following few seconds he/she can use it. The method provided in this work, finds and predicts values of x and y characteristics of aircraft trajectory with the smallest possible error. In real world it means prediction of longitude and latitude values that can be used for recognizing whether two trajectories of two different airplanes will cross each other in near future and way cause accidents on the sky.

I also have to write that except variable advantages, there are some disadvantages that has to be improved. Probably the biggest lack here is that we can not use all methods in one, and in this way to get 3D projection of all x,y and z characteristics. It would help us to implement that straight to calculations of everything that is needed for this problem. However now we are satisfied with the existing solutions but it does not mean we should stop working on spreading this idea of solving trajectory prediction problems.

10.1 Future work

In this project, some good frames were set for the beginning of work on topics that are related with this project. Of course, it can be later improved in many different ways such as making whole program with all algorithms and methods in it, or it can be implemented separately where those methods are needed. In the next steps, and I hope that this project will continue in my diploma thesis and at the end it will get more complex nature. The main idea for the future work is to implement this project not only for TCAS systems at the airplanes but also to make it possible for using at the airports, and the most important one for is Vaclav Havel airport in Prague. This principle of solving and preventing airplane trajectory interception near ground can be used for the cases of landing and taking off at the airports where time gap between these situations is very small. Below, there is instrument approach chart map for the Prague airport.

10.2 Airport Vaclav Havel

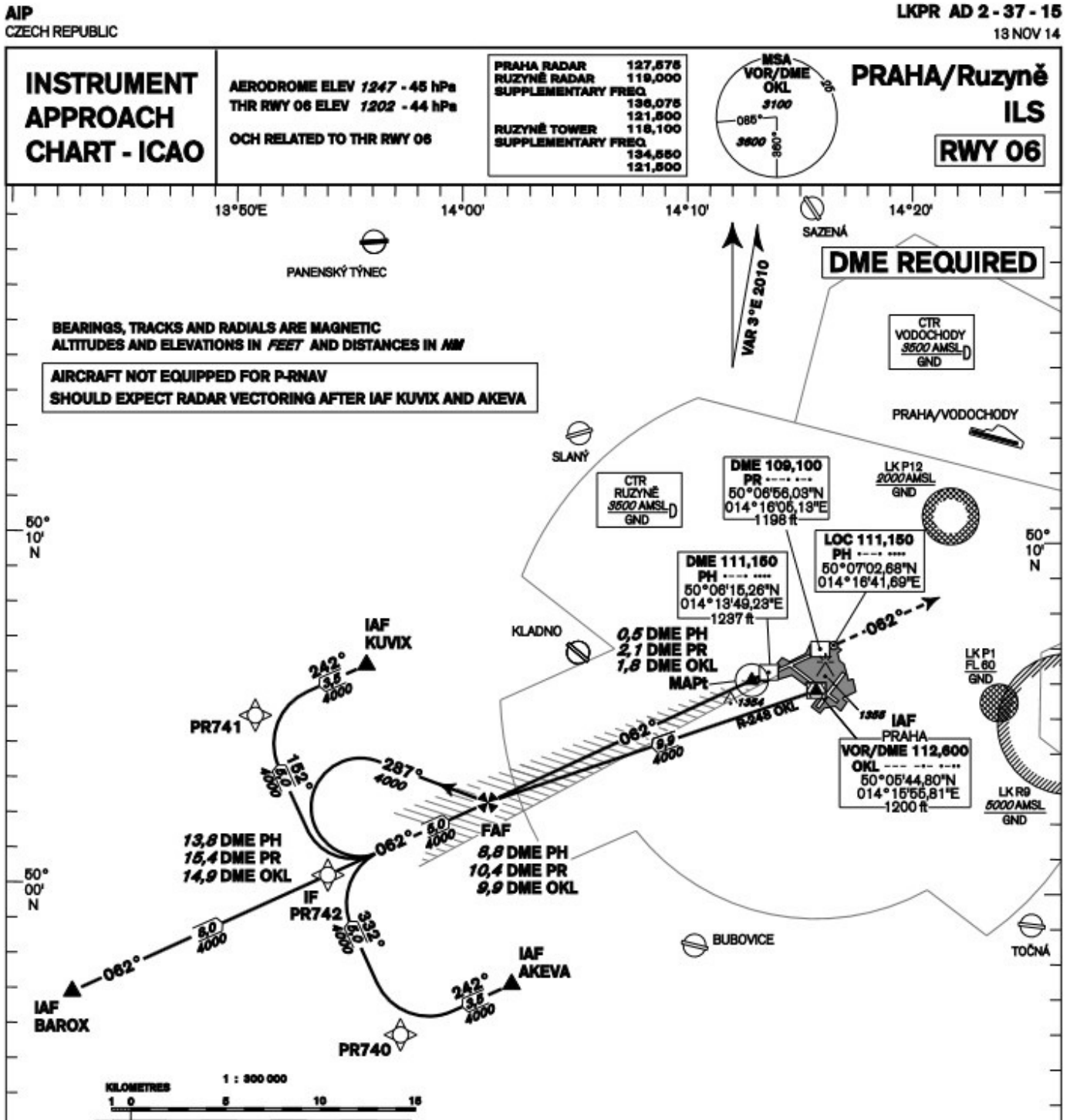


Figure 10.1. Instrument approach map of airport Vaclav Havel [14]

References

- [1] Eurocontrol *Research, and Development*.
https://www.eurocontrol.int/phare/public/standard_page/EFMS.html, URL accessed on 2016-03-20.
- [2] M. Wieser B. Hofmann-WellenhoL K. Legat. *Navigation - Principles of Positioning and Guidance*. Wien, AU: Novographie Druck G.m.b.H, 2003. ISBN 10713249.
- [3] E.D. Kaplan, and Ch. Hegarty. *Understanding GPS: Principles and Applications. 2nd ed.* London, UK: Artech House,, 2005. ISBN 1580538940.
- [4] B. Bhatta. *Global Navigation Satellite Systems*. 1st edition. BS Publications, An Unit of BSP Books Pvt.,Ltd., 2009. ISBN 978-0-415-66560-5.
- [5] J. Sanz Subirana.
<http://www.navipedia.net/index.php/>, URL accessed on 2015-05-15.
- [6] Ann C. Drumm James K. Kuchar. *The Traffic Alert and Collision Avoidance System*. Lincoln Laboratory Jurnal, 2007.
- [7] *GPX-Files* [software].
https://en.wikipedia.org/wiki/GPS_Exchange_Format, URL accessed on 2015-12-19.
- [8] GPX Files Values of point trajectory.
<http://openflights.org/about>, URL accessed on 2014-10-20.
- [9] Values ofpointtrajectory [GPXFiles].
<http://www.gpslogbooks.com/>, URL accessed on 2014-10-20.
- [10] Dan Foster.
http://www.topografix.com/gpx_for_users.asp, URL accessed on 2015-12-09.
- [11] Technical infomration about airplanes AirBus.
<http://www.airbus.com/aircraftfamilies/passengeraircraft/a380family/performance/1>, URL accessed on 2015-12-07.
- [12] Chris Brady The Boing technical information.
<http://www.b737.org.uk/>, URL accessed on 2015-12-03.
- [13] *Technical information of Vaclav Havel Airport, Prague*.
<http://www.prg.aero/en/business-section/aviation-business/airport-facts-prague/technical-information/1>, URL accessed on 2015-09-15.
- [14] Safety information-management document.
<http://www.icao.int/safety/information-management/documents>, URL accessed on 2015-11-03.



Appendix A

CD

As an attachment for this work is CD with:

- A Bachelor Thesis in PDF format [BT- VanjaNeretljak]
- A source code-programmed application:
- [Genije.m] - containing basic code for one of the calculation methods and calling methods: [racunanjePozicije.m], [racunanjeBrzine.m] and [racunanjeUbrzanja.m].
- [koordinata.m] - transforming GPX points
- [fitFunkcija.m]
- [data.m] and [skript.m] - used for another calculation methods
- All .tex files needed for thesis and excel data used in calculations

Czech Technical University in Prague
Faculty of Electrical Engineering

Department of Electromagnetic Field

BACHELOR PROJECT ASSIGNMENT

Student: **Vanja Neretljak**

Study programme: Communications, Multimedia, Electronics
Specialisation: Communication Technology

Title of Bachelor Project: **User trajectory prediction based on apriori knowledge of position and velocity**

Guidelines:

Study the position determination methods based on GNSS data and the usage of these data for modelling and predicting the movement of one or several users with focus on area of air transport.

Study the typical principles and approaches used in collision avoidance systems. Design the algorithms for user movement prediction and determination of possible collision with predicted movement of other users. Analyze the error in movement modelling for the used algorithms. The modelling and evaluation of collision possibility realize in Matlab.

Bibliography/Sources:

Kaplan, E.D. - Hegarty, Ch.: Understanding GPS: Principles and Applications. 2nd ed., 2005. London: Artech House, (c)2005. ISBN 1580538940.

Hayes, M.H.: Statistical Digital Signal Processing and Modeling. 1996. Wiley & Sons, (c) 1996. ISBN 0-471-59431-8.

Bachelor Project Supervisor: Ing. Pavel Puričar

Valid until: S 2016/2017



prof. Ing. Pavel Pečač, Ph.D.
Head of Department

prof. Ing. Pavel Ripka, CSc.
Dean

Prague, February 17, 2016

České vysoké učení technické v Praze
Fakulta elektrotechnická

katedra elektromagnetického pole

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Student: **Vanja Neretljak**

Studijní program: Komunikace, multimédia a elektronika
Obor: Komunikační technika

Název tématu: **Predikce trajektorie pohybu uživatele ze znalosti apriorních polohových a rychlostních dat**

Pokyny pro vypracování:

Prostudujte metody určení polohy na základě dat získaných z GNSS systémů a využití těchto apriorních dat pro modelování a predikci pohybu jednoho či více uživatelů se zaměřením na oblast letecké dopravy. Prostudujte typické postupy a principy antikolizních systémů. Navrhněte algoritmy pro predikci pohybu uživatele a určení možné kolize s predikovaným pohybem jiného uživatele. Analyzujte přesnost modelování při použití různých algoritmů. Proveďte simulaci takového modelování a vyhodnocování pravděpodobnosti kolize prostřednictvím systému Matlab.

Seznam odborné literatury:

[1] Kaplan, E.D. - Hegarty, Ch.: Understanding GPS: Principles and Applications. 2nd ed., 2005. London: Artech House, (c)2005. ISBN 1580538940.

[2] Hayes, M.H.: Statistical Digital Signal Processing and Modeling. 1996. Wiley & Sons, (c) 1996. ISBN 0-471-59431-8.

Vedoucí: Ing. Pavel Puričer

Platnost zadání: LS 2016/2017

Prof. Ing. Pavel Pechač, Ph.D.
vedoucí katedry

L.S.

prof. Ing. Pavel Ripka, CSc.
děkan

V Praze dne 17. 2. 2016