



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

**LETECKÝ SIMULÁTOR S PODPOROU
VIRTUÁLNÍ REALITY**

VIRTUAL REALITY ENABLED FLIGHT SIMULATOR

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

DOMINIK RUTA

VEDOUCÍ PRÁCE

SUPERVISOR

doc. Ing. PETER CHUDÝ, Ph.D., MBA

BRNO 2019

Zadání bakalářské práce



21725

Student: **Ruta Dominik**
Program: Informační technologie
Název: **Letecký simulátor s podporou virtuální reality**
Virtual Reality Enabled Flight Simulator
Kategorie: Modelování a simulace

Zadání:

1. Nastudujte matematický popis dynamiky letu.
2. Seznamte se s principy vizualizace palubních letových dat.
3. Vytvořte, případně zajistěte z vhodných zdrojů, virtuální 3D kokpit letounu a funkční simulační jádro, které umožní využití technologií pro virtuální realitu.
4. Váš návrh implementujte do simulačního prostředí.
5. Zhodnoťte dosažené výsledky a diskutujte možné směry pokračování projektu.

Literatura:

- dle pokynů vedoucího

Pro udělení zápočtu za první semestr je požadováno:

- Body 1, 2 a část 3.

Podrobné závazné pokyny pro vypracování práce viz <http://www.fit.vutbr.cz/info/szz/>

Vedoucí práce: **Chudý Peter, doc. Ing., Ph.D. MBA**

Vedoucí ústavu: Černocký Jan, doc. Dr. Ing.

Datum zadání: 1. listopadu 2018

Datum odevzdání: 15. května 2019

Datum schválení: 9. května 2019

Abstrakt

Tato práce se zabývá vývojem leteckého simulátoru, který vytváří dynamický model letounu pomocí vlastního simulačního jádra. Simulační rámec se skládá z modelu pro výpočet pohybových rovnic, modelu atmosféry Země, aerodynamických a hmotově setrvačných vlastností letounu a modelu pohonu letounu. Při vývoji bylo využito vývojové prostředí Unity 3D, kde byla vytvořena scéna simulace a uživatelské rozhraní aplikace. 3D Modely letounu a kokpitu byly vytvořeny pomocí studentské verze 3DS Max. Simulátor poskytuje i podporu virtuální reality pro zvýšení autenticity a pocitu realističnosti.

Abstract

This bachelor thesis is focused on the development of a flight simulator, which uses its own simulation core to run an aircraft dynamic model. The simulation framework consists of a motion model, the Earth atmosphere model, aerodynamics and inertial characteristics of the aircraft and an aircraft propulsion model. The Unity 3D software was used for the simulation environment and user interface development. The 3D models of the cockpit and the aircraft were modeled in a student version of the 3DS Max. The presented simulator also supports the virtual reality for higher simulation authenticity and an improved sense of realism.

Klíčová slova

simulace, letectví, letecký simulátor, pohybové rovnice, Unity 3D, 3DS Max, aerodynamika, virtuální realita

Keywords

simulation, aviation, flight simulator, equations of motion, Unity 3D, 3DS Max, aerodynamics, virtual reality

Citace

RUTA, Dominik. *Letecký simulátor s podporou virtuální reality*. Brno, 2019. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce doc. Ing. Peter Chudý, Ph.D., MBA

Letecký simulátor s podporou virtuální reality

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana doc. Ing. Petra Chudého, Ph.D., MBA. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Dominik Ruta
14. května 2019

Poděkování

Mé obrovské díky patří vedoucímu této bakalářské práce, panu doc. Ing. Petru Chudému, Ph.D., MBA, za velkou podporu, cenné rady a čas, který mi při vývoji simulátoru věnoval.

Obsah

1	Úvod	8
2	Matematický popis dynamiky letu	9
2.1	Souřadné systémy	9
2.2	Transformace mezi souřadnými systémy	12
2.3	Pohybové rovnice tuhého tělesa	14
2.4	Mezinárodní standardní atmosféra	18
2.5	Řešení obyčejných diferenciálních rovnic	19
2.6	Řídicí systém letounu	19
3	Principy vizualizace palubních letových dat	23
3.1	Mechanická éra	23
3.2	Elektro-mechanická éra	24
3.3	Elektro-optická éra	28
4	Návrh 3D kokpitu a simulačního jádra	31
4.1	Technologie použité pro realizaci návrhu	32
4.2	Simulační jádro	33
4.3	3D model letounu a kokpitu	36
4.4	Letový model	40
5	Implementace	41
5.1	Uživatelské rozhraní	41
5.2	Scéna simulace	44
5.3	Implementace letounu v Unity	46
5.4	Simulační jádro	51
5.5	Podpora virtuální reality	53
6	Zhodnocení dosažených výsledků a možná vylepšení	55
6.1	Praktické testování	55
6.2	Uživatelské testování	55
6.3	Vyhodnocení	56
6.4	Možnosti budoucího vývoje	58
7	Závěr	59
	Literatura	60
A	Doprovodný dotazník	63

Seznam obrázků

2.1	Souřadný systém WGS 84	9
2.2	Souřadný systém Země	10
2.3	Lokální souřadný systém NED	11
2.4	Letadlový souřadný systém	11
2.5	Aerodynamický souřadný systém ve vztahu k letadlovému souřadnému systému	12
2.6	Eulerovy úhly	13
2.7	Osy otáčení letounu	14
2.8	Šest stupňů volnosti	15
2.9	Křídélka	20
2.10	Výškové kormidlo	21
2.11	Směrové kormidlo	21
3.1	Přístrojová deska Consolidated NY-2 [21]	24
3.2	Základní šestice	25
3.3	Uspořádání do T	25
3.4	Pitot-statická soustava	26
3.5	Umělý horizont ADI [5]	27
3.6	Indikátor horizontální situace [10]	28
3.7	Primární letový displej [29]	29
3.8	Průhledový displej [3]	29
4.1	Schéma simulátoru	31
4.2	Schéma simulačního jádra	33
4.3	Koeficient odporu křídla letounu XP-51 v závislosti na koeficientu vztlaku [1].	34
4.4	Koeficient vztlaku křídla letounu XP-51 v závislosti na úhlu náběhu [1]. . .	34
4.5	Model pohonu letounu	35
4.6	P-51D Mustang [9]	37
4.7	Křídla a trup se zpřesněnou geometrií	38
4.8	Model podvozkového kola před aplikováním Turbosmooth	38
4.9	Model před aplikováním Turbosmooth	38
4.10	Model po aplikování Turbosmooth	38
4.11	Konečný model kokpitu	39
4.12	Textura rychloměru	39
4.13	Přístrojová deska s otexturovanými přístroji	39
5.1	Stavový automat programu	42
5.2	Rozložení textového výstup simulačních dat na obrazovce	44
5.3	Profilování terénu	45
5.4	Výsledná podoba terénu	45

5.5	Vojenská základna ve scéně	45
5.6	Pohled na výslednou krajinu	45
5.7	Konečný model letounu v Unity	46
5.8	Konečný model kokpitu letounu v Unity	46
5.9	Jednoduchý kolizní model vytvořený z několika colliderů	47
5.10	Pozice vnější kamery vůči letounu	48
5.11	Vizualizace posunu řídicí páky	48
5.12	Vizualizace rotace pravého křídélka	48
5.13	Kompas s pamatováčkem	49
5.14	Plynová páka	50
5.15	Iterace výpočtu	51
5.16	Souřadný systém v Unity [15]	52
5.17	Navržené ovládání pomocí HTC Vive	53
5.18	Grafické rozhraní na vytváření akcí pro tlačítka VR ovládačů	54
6.1	Genderové a věkové rozložení testovací skupiny	56
6.2	Zkušenosti se simulátory v testovací skupině	56

Seznam zkratk

ADI	Umělý horizont
BFF	Letadlový souřadný systém
CG	Těžiště
CRT	Katodová trubice
ECEF	Souřadný systém Země
GPS	Globální polohový systém
HSI	Indikátor horizontální situace
HVAR	Vysokorychlostní letecká raketa
IIS	Systém integrovaných přístrojů
ILS	Integrovaný přistávací systém
ISA	Model mezinárodní standardní atmosféry
LCD	Displej z tekutých krystalů
NASA	Národní úřad pro letectví a kosmonautiku
NED	Lokální souřadný systém North, East, Down
RAF	Britské královské letectvo
SI	Mezinárodní systém jednotek
UI	Uživatelské rozhraní
USA	Spojené státy americké
VR	Virtuální realita
WGS 84	Světový geodetický systém 1984

Seznam veličin a jednotek

a	$[m.s^{-1}]$	Rychlost zvuku
b	$[m]$	Rozpětí křidel
\bar{c}	$[m]$	Délka střední aerodynamické tětiny
C_T	$[1]$	Tahový koeficient
C_P	$[1]$	Výkonový koeficient
D	$[m]$	Průměr vrtule
F_T	$[N]$	Tahová síla
g	$[m.s^{-2}]$	Tíhové zrychlení
h	$[m]$	Aktuální nadmořská výška letounu
I_{XX}	$[kg.m^2]$	Kvadratický moment setrvačnosti k ose X
I_{YY}	$[kg.m^2]$	Kvadratický moment setrvačnosti k ose Y
I_{ZZ}	$[kg.m^2]$	Kvadratický moment setrvačnosti k ose Z
I_{XY}	$[kg.m^2]$	Deviační moment setrvačnosti
I_{YZ}	$[kg.m^2]$	Deviační moment setrvačnosti
I_{XZ}	$[kg.m^2]$	Deviační moment setrvačnosti
J	$[1]$	Postupný poměr
m	$[kg]$	Celková hmotnost letounu
n	$[ot.s^{-1}]$	Otáčky vrtule
P	$[W]$	Celkový výkon motoru
P_{vrtule}	$[W]$	Požadovaný výkon na vrtuli
P_{motor}	$[W]$	Dostupný výkon motoru
P_h	$[W]$	Výkon motoru ve výšce
P_{vyb}	$[W]$	Spotřební výkon vybavení
P_{dostup}	$[W]$	Dostupný výkon
\bar{q}	$[Pa]$	Dynamický tlak
S	$[m^2]$	Celková plocha křidel
T	$[K]$	Termodynamická teplota okolí
V_T	$[m.s^{-1}]$	Pravá vzdušná rychlost
α	$[rad]$	Úhel náběhu
β	$[rad]$	Úhel vybočení
ρ	$[kg.m^3]$	Hustota vzduchu
η	$[1]$	Účinnost vrtule
λ	$[^\circ]$	Zeměpisná délka
φ	$[^\circ]$	Zeměpisná šířka
F_X	$[N]$	Celková síla v ose X letadlového souřadného systému
F_Y	$[N]$	Celková síla v ose Y letadlového souřadného systému
F_Z	$[N]$	Celková síla v ose Z letadlového souřadného systému
\bar{X}	$[N]$	Celková odporová síla v BFF
\bar{Y}	$[N]$	Celková boční síla v BFF
\bar{Z}	$[N]$	Celková vztlaková síla v BFF
D	$[N]$	Celková odporová síla v aerodynamickém souřadném systému
Y	$[N]$	Celková boční síla v aerodynamickém souřadném systému
L	$[N]$	Celková vztlaková síla v aerodynamickém souřadném systému
\bar{L}	$[N.m]$	Celkový moment klonění
\bar{M}	$[N.m]$	Celkový moment klopení
\bar{N}	$[N.m]$	Celkový moment vybočení

C_X	[1]	Celkový koeficient síly v ose X v BFF
C_Y	[1]	Celkový koeficient síly v ose Y v BFF
C_Z	[1]	Celkový koeficient síly v ose Z v BFF
C_D	[1]	Celkový koeficient odporu v aerodynamickém souřadném systému
$C_{D\alpha}$	[1]	Dílčí koeficient odporu vzhledem k úhlu náběhu
C_{DMach}	[1]	Dílčí koeficient odporu vzhledem k Machově číslu
$C_{D\beta}$	[1]	Dílčí koeficient odporu vzhledem k úhlu vybočení
C_{DFlaps}	[1]	Dílčí koeficient odporu vzhledem k výchylce vztlakových klapků
C_{DGear}	[1]	Dílčí koeficient odporu vzhledem k pozici podvozku
$C_{D\eta}$	[1]	Dílčí koeficient odporu vzhledem k výchylce výškového kormidla
$C_{D\xi}$	[1]	Dílčí koeficient odporu vzhledem k výchylce křidélek
C_Y	[1]	Celkový koeficient boční síly v aerodynamickém souřadném systému
$C_{Y\beta}$	[1]	Dílčí koeficient boční síly vzhledem k úhlu vybočení
C_L	[1]	Celkový koeficient vztlaku v aerodynamickém souřadném systému
$C_{L\alpha}$	[1]	Dílčí koeficient vztlaku vzhledem k úhlu náběhu
C_{LFlaps}	[1]	Dílčí koeficient vztlaku vzhledem k výchylce vztlakových klapků
$C_{L\eta}$	[1]	Dílčí koeficient vztlaku vzhledem k výchylce výškového kormidla
C_l	[1]	Celkový koeficient momentu klonění
$C_{l\beta}$	[1]	Dílčí koeficient momentu klonění vzhledem k úhlu vybočení
C_{lp^*}	[1]	Dílčí koeficient momentu klonění vzhledem k bezrozměrné úhlové rychlosti klonění
C_{lr^*}	[1]	Dílčí koeficient momentu klonění vzhledem k bezrozměrné úhlové rychlosti bočení
$C_{l\xi}$	[1]	Dílčí koeficient momentu klonění vzhledem k výchylce křidélek
$C_{l\zeta}$	[1]	Dílčí koeficient momentu klonění vzhledem k výchylce směrového kormidla
$C_{l\alpha}$	[1]	Dílčí koeficient momentu klonění vzhledem k úhlu náběhu
C_m	[1]	Celkový koeficient momentu klopení
$C_{m\alpha}$	[1]	Dílčí koeficient momentu klopení vzhledem k úhlu náběhu
C_{mMach}	[1]	Dílčí koeficient momentu klopení vzhledem k Machově číslu
C_{mFlaps}	[1]	Dílčí koeficient momentu klopení vzhledem k výchylce vztlakových klapků
C_{mGear}	[1]	Dílčí koeficient momentu klopení vzhledem k pozici podvozku
$C_{m\eta}$	[1]	Dílčí koeficient momentu klopení vzhledem k výchylce výškového kormidla
C_{mq^*}	[1]	Dílčí koeficient momentu klopení vzhledem k bezrozměrné úhlové rychlosti klopení
C_n	[1]	Celkový koeficient momentu vybočení

C_{n_β}	[1]	Dílčí koeficient momentu bočení vzhledem k úhlu vybočení
$C_{n_{r^*}}$	[1]	Dílčí koeficient momentu klonění vzhledem k bezrozměrné úhlové rychlosti bočení
C_{n_ζ}	[1]	Dílčí koeficient momentu klonění vzhledem k výchylce směrového kormidla
u	$[m.s^{-1}]$	Složka rychlosti podél osy X v BFF
v	$[m.s^{-1}]$	Složka rychlosti podél osy Y v BFF
w	$[m.s^{-1}]$	Složka rychlosti podél osy Z v BFF
p	$[rad.s^{-1}]$	Úhlová rychlost klonění
q	$[rad.s^{-1}]$	Úhlová rychlost klopení
r	$[rad.s^{-1}]$	Úhlová rychlost bočení
p^*	[1]	Bezrozměrná úhlová rychlost klonění
q^*	[1]	Bezrozměrná úhlová rychlost klopení
r^*	[1]	Bezrozměrná úhlová rychlost bočení
ϕ	$[rad]$	Úhel klonění
θ	$[rad]$	Úhel klopení
ψ	$[rad]$	Úhel bočení
ξ	$[rad]$	Výchylka křídélek
η	$[rad]$	Výchylka výškového kormidla
ζ	$[rad]$	Výchylka směrového kormidla
δ_F	$[rad]$	Výchylka vztlakových klapek
X_N	$[m]$	Pozice letounu na ose X v NED
Y_N	$[m]$	Pozice letounu na ose Y v NED
Z_N	$[m]$	Pozice letounu na ose Z v NED
\mathbf{M}	[-]	Transformační matice
\mathbf{T}	[-]	Stavový vektor
$\vec{\mathbf{V}}$	[-]	Vektor rychlosti letounu
$\vec{\omega}$	[-]	Vektor úhlových rychlostí letounu
$\vec{\mathbf{P}}_N$	[-]	Vektor pozice letounu
$\vec{\Phi}$	[-]	Vektor polohy letounu

Kapitola 1

Úvod

Letecká simulace je jedno z nejkompexnějších, avšak velmi zajímavých inženýrských odvětví na světě. Simulace závisí na obrovském množství veličin, které je potřeba v průběhu samotné simulace počítat. Základem jsou pohybové rovnice, pomocí kterých jsme schopni popsat pohyb letadla v prostoru. Tyto rovnice jsou však jen špičkou ledovce. Je nutné správně vytvořit aerodynamický model, který obyčejné těleso promění v letoun s reálnými vlastnostmi. Dále je nutné implementovat reálný model pohonné jednotky, kolizní modely, reakci se zemí a mnoho dalšího. Ačkoliv se zdá, že vytvoření leteckého simulátoru je velmi náročná činnost pro velké týmy lidí, lze vytvořit letecký simulátor, poskytující důležité vlastnosti, v relativně akceptovatelné době.

V této práci je popsáno, jak na základě znalostí o popisu pohybu letadla v prostoru, lze vytvořit jednoduchý simulátor pomocí herního jádra Unity. Navzdory tomu, že Unity poskytuje svůj vlastní fyzikální model, je kladen důraz na to, aby výsledný simulátor tohoto fyzikálního modelu nevyužíval a na letoun se naopak aplikovalo vlastní simulační jádro. Aby byla simulace více autentická a uživatel měl z letu pocit větší realističnosti, simulátor implementuje i podporu virtuální reality.

Celá práce se dělí do 7 kapitol. V druhé kapitole je uveden matematicko-fyzikální popis pohybu letounu v prostoru a s tím spojený způsob jeho řešení. Na konci kapitoly je popsáno, jak se letoun za letu ovládá. Druhá kapitola tedy popisuje principy, na základě kterých je postavena většina simulace. Ve třetí kapitole jsou popsány principy vizualizace letových dat. Následuje čtvrtá kapitola, ve které je nejdříve popsána architektura simulačního jádra zahrnující i návrh modelu pohonu letounu. Následuje popis tvorby 3D modelu letounu a jeho kokpitu. Na konci kapitoly je pak popsán návrh letového modelu letounu. Návrh zahrnuje jakým způsobem je model v simulátoru realizován a jak jsou počítány aerodynamické koeficienty. Pátá kapitola je pak převážně věnována implementaci simulátoru. Nejdříve je popsán návrh a implementace uživatelského rozhraní. Dále pak tvorba scény simulace, implementace letounu v Unity a implementace simulačního jádra. Konec kapitoly je věnován implementaci virtuální reality včetně návrhu ovládání pomocí ovládačů pro virtuální realitu. V šesté kapitole je popsán testovací proces a jeho výsledky. Konec kapitoly je věnován možnostem budoucího vývoje. V poslední, sedmé kapitole, jsou pak shrnuty dosažené výsledky této práce.

Kapitola 2

Matematický popis dynamiky letu

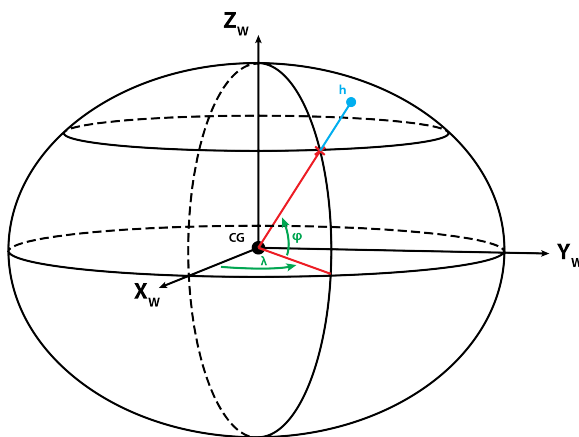
Aby bylo možné vytvořit kvalitní simulátor, který bude věrohodně napodobovat chování simulovaného systému či objektu, je nejdříve potřeba umět tato chování popsat z matematicko-fyzikálního hlediska. Získané znalosti je následně potřeba aplikovat na dané objekty, od kterých se očekává, že budou co nejvěrohodněji napodobovat chování svých reálných předloh.

V případě leteckých simulátoru je důležité popsat a realizovat prostorový pohyb letounu. Pohyb letounu se popisuje pomocí soustavy dvanácti pohybových rovnic.

2.1 Souřadné systémy

Jak již bylo zmíněno výše, pohyb letounu v prostoru je popsán pomocí soustavy dvanácti diferenciálních pohybových rovnic. Než však začneme s jejich vyjadřováním, musíme si nejdříve určit souřadné systémy, ve kterých se budou dané rovnice odvozovat.

2.1.1 Geodetický souřadný systém



Obrázek 2.1: Souřadný systém WGS 84

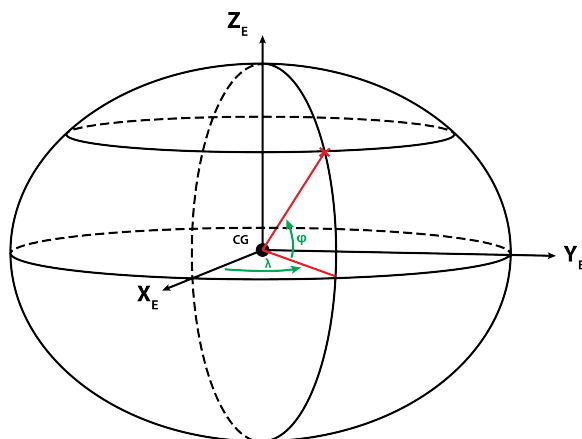
Geodetický souřadný systém WGS 84 (World Geodetic System 1984), navržený roku 1984 a naposled upraven v roce 2004, je pravotočivý, inerciální souřadnicový systém se středem v těžišti Země. Dnes je tento systém využíván v oblasti kartografie, geodezie a navigace, kde jej mimo jiné využívá i systém GPS. Model definuje souřadný systém jako referenční elipsoid. Poloha bodu je určena pomocí zeměpisné šířky, délky a výšky. Osa X_G

je kladná směrem k průsečíku rovníku s nultým poledníkem, kladná osa Z_G směřuje směrem k severnímu pólu a osa Y_G je kolmá na předešlé dvě osy, přičemž kladná je ve směru podle pravidla pravé ruky [4].

Zeměpisná šířka je určena velikostí úhlu λ v rozmezí od -180 do 180 mezi měřeným bodem a poledníkem, zeměpisná šířka velikostí úhlu μ v rozmezí od -90 do 90 mezi měřeným bodem a rovníkem, a zeměpisná šířka jako vzdálenost měřeného bodu od povrchu elipsoidu viz obrázek 2.1 [4].

2.1.2 Souřadný systém Země

Souřadný systém Země ECEF (Earth-Centered Earth-Fixed) je pravotočivý, neinerciální, kartézský souřadnicový systém se středem v těžišti Země. Tento systém umožňuje popsat pozici tělesa vůči Zemi. Dnes se používá v oblasti geodzie a navigace. Stejně jako u souřadnicového systému WGS 84, osa X_E je kladná směrem k průsečíku rovníku s nultým poledníkem, osa Z_E je kladná směrem k severnímu pólu Země a osa Y_E je opět kolmá na předešlé osy, přičemž kladná je rovněž ve směru podle pravidla pravé ruky. Jelikož se Země otáčí kolem osy Z_E , kterou označujeme jako rotační, je tento systém neinerciální. Systém ECEF je zobrazen na obrázku 2.2 [4].



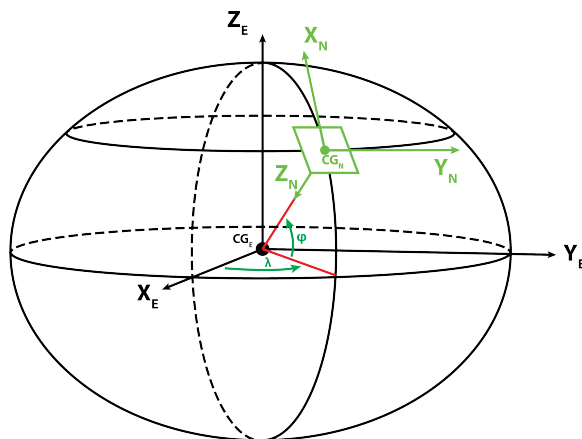
Obrázek 2.2: Souřadný systém Země

2.1.3 Lokální souřadný systém

Lokální souřadný systém NED (North-East-Down) je pravotočivý, inerciální, kartézský souřadnicový systém, který se běžně využívá v letectví. Vztahuje se vůči elipsoidu, v našem případě Zemi. Na rozdíl od dvou výše zmíněných soustav, je zde počátek volen podle určité potřeby. V letectví je to typicky těžiště letadla. Osa X_N je vždy kladná směrem k severnímu pólu, osa Y_N k východu a třetí osa Z_N je orientovaná směrem dolů, k povrchu elipsoidu. Soustava se tak skládá ze tří souřadnic. První udává pozici po ose směrem na sever, druhá určuje pozici po východní ose a třetí definuje vertikální pozici. Systém NED je možné vidět na obrázku 2.3 [4].

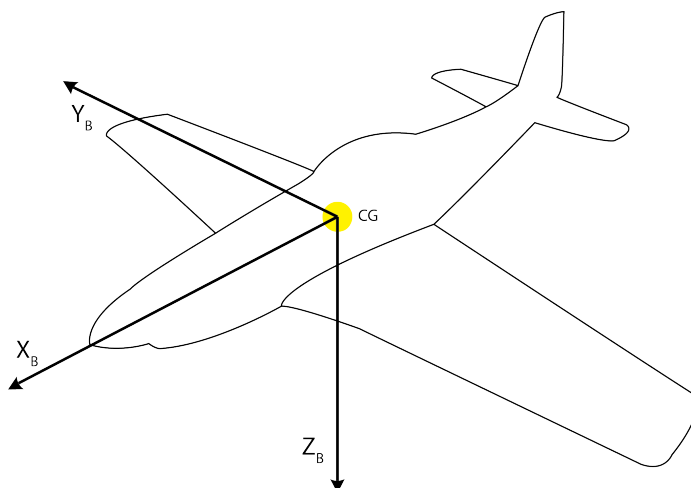
2.1.4 Letadlový souřadný systém

Letadlový souřadný systém (Body-Fixed Frame, Body Axis System), zobrazený na obrázku 2.4, je pravotočivý souřadnicový systém, který se vztahuje k samotnému letadlu, kdy po-



Obrázek 2.3: Lokální souřadný systém NED

čátek souřadnicového systému se nachází v jeho těžišti. Osa X_B je kladná vždy směrem dopředu, osa Y_B směřuje do pravého křídla a osa Z_B je orientovaná podle pravidla pravé ruky pod letoun směrem dolů [4].



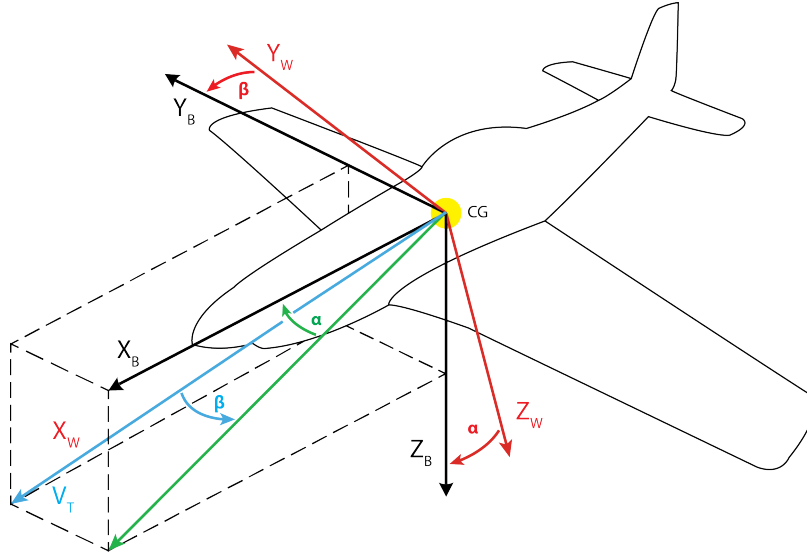
Obrázek 2.4: Letadlový souřadný systém

2.1.5 Aerodynamický souřadný systém

Aerodynamický souřadný systém (Wind Axis System), je podobný tomu letadlovému, kdy počátek soustavy opět představuje těžiště letounu. V aerodynamickém modelu je však zohledněn vektor rychlosti nabíhajícího proudu vzduchu, svírající s původní osou X_B úhel α , který je v aerodynamice označován jako úhel náběhu [23].

Tento úhel má velký vliv na výsledný součinitel vztlaku a tudíž i na samotnou vztlakovou sílu letadla, kdy do určitého úhlu platí, že čím vyšší je úhel α , tím vyšší je i součinitel vztlaku. Při překročení maximálního úhlu náběhu a jeho neustálém zvyšování, začne součinitel vztlaku letounu naopak klesat, což má za následek také pokles samotné vztlakové síly.

Rotací o úhel α vznikne stabilní souřadný systém (Stability Axis System) s osami $X_S Y_S Z_S$. Při následné rotaci systému o úhel β , označovaný jako úhel vybočení, kolem osy Z_S , vznikne zmíněný aerodynamický souřadný systém. Osy aerodynamického souřadného systému označíme jako $X_W Y_W Z_W$. Vztah mezi aerodynamickým a letadlovým souřadným systémem je znázorněn na obrázku 2.5 [23].



Obrázek 2.5: Aerodynamický souřadný systém ve vztahu k letadlovému souřadnému systému

2.2 Transformace mezi souřadnými systémy

Mezi jednotlivými souřadnými systémy je možno provádět vzájemné transformace. Níže jsou popsány dvě nejdůležitější, které jsou nezbytné pro popis pohybových rovnic letadla.

2.2.1 Transformace z aerodynamického systému do letadlového

Pro transformaci z aerodynamického systému do letadlového, je potřeba nejdříve aerodynamickou soustavu rotovat kolem osy Z_W o úhel vybočení β . Tím dostaneme souřadný systém $X_S Y_S Z_S$, kdy transformační matice vypadá následovně [23]:

$$\mathbf{M}_{SW} = \begin{bmatrix} \cos \beta & \sin \beta & 0 \\ -\sin \beta & \cos \beta & 0 \\ 0 & 0 & 1 \end{bmatrix}. \quad (2.1)$$

Následně se systém $X_S Y_S Z_S$ otočí o úhel náběhu α kolem osy Y_S pomocí matice [23]:

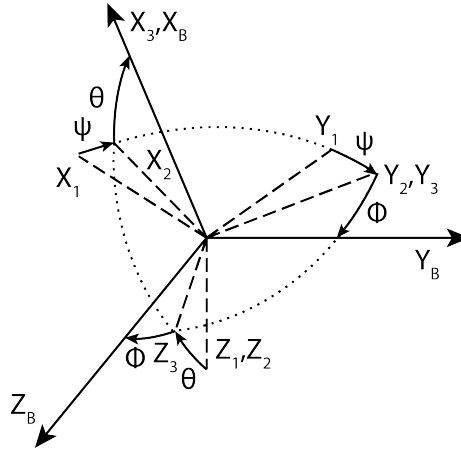
$$\mathbf{M}_{BS} = \begin{bmatrix} \cos \alpha & 0 & \sin \alpha \\ 0 & 1 & 0 \\ \sin \alpha & 0 & \cos \alpha \end{bmatrix}. \quad (2.2)$$

Celkově tedy platí [23]:

$$\mathbf{T}_B = \begin{bmatrix} \cos \alpha \cos \beta & -\cos \alpha \sin \beta & -\sin \alpha \\ \sin \beta & \cos \beta & 0 \\ \sin \alpha \cos \beta & -\sin \alpha \sin \beta & \cos \alpha \end{bmatrix} \mathbf{T}_W . \quad (2.3)$$

2.2.2 Transformace z letadlového do lokálního souřadného systému NED

Pro transformaci z letadlového do lokálního souřadného systému NED využijeme tzv. Eulerových úhlů, kdy letadlová soustava je rotována přes tři úhly kolem tří os [26].



Obrázek 2.6: Eulerovy úhly

Nejdříve vezmeme letadlový souřadný systém s osami $X_B Y_B Z_B$, u kterého přejmenujeme osy na $X_1 Y_1 Z_1$. Počátek tohoto systému si pro přehlednost posuneme do těžiště letadla. Rotace jsou komutativní, čili pořadí rotací je striktně dané. Pokud by došlo k rotacím ve špatném pořadí, výsledkem by byla špatná prostorová poloha letadla [26].

Prvním krokem transformace je rotace souřadného systému $X_1 Y_1 Z_1$ kolem osy Z_B o úhel Ψ , který je označován jako úhel bočení. Tím vznikne nový souřadný systém $X_2 Y_2 Z_2$. Vztah je popsán následovně [26]:

$$\mathbf{T}_2 = \begin{bmatrix} \cos \Psi & \sin \Psi & 0 \\ -\sin \Psi & \cos \Psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \mathbf{T}_1 . \quad (2.4)$$

V druhém kroku je potřeba systém $X_2 Y_2 Z_2$ rotovat kolem osy Y_2 o úhel Θ . Tento úhel je znám jako úhel klopení. Nově vzniklý systém značíme $X_3 Y_3 Z_3$. Vztah dvou souřadných systémů je vyjádřen jako [26]:

$$\mathbf{T}_3 = \begin{bmatrix} \cos \Theta & 0 & -\sin \Theta \\ 0 & 1 & 0 \\ \sin \Theta & 0 & \cos \Theta \end{bmatrix} \mathbf{T}_2 . \quad (2.5)$$

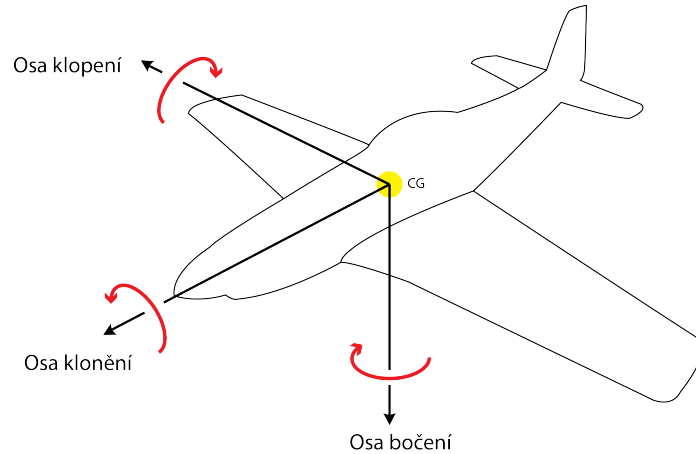
Posledním krokem je pootočení systému $X_3Y_3Z_3$ kolem osy X_3 o úhel Φ , který je definován jako úhel klonění. Vztah mezi dvěma souřadnými systémy popisuje následující rovnice [26]:

$$\mathbf{T}_N = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \Phi & \sin \Phi \\ 0 & -\sin \Phi & \cos \Phi \end{bmatrix} \mathbf{T}_3 . \quad (2.6)$$

Spojením rovnic 2.4, 2.5 a 2.6 a následným vynásobením matic, získáme výsledný vztah pro transformaci vektoru mezi letadlovým souřadným systémem a systémem NED [26]:

$$\mathbf{T}_N = \begin{bmatrix} \cos \Theta \cos \Psi & \sin \Phi \sin \Theta \cos \Psi - \cos \Phi \sin \Psi & \cos \Phi \sin \Theta \cos \Psi + \sin \Phi \sin \Psi \\ \cos \Theta \sin \Psi & \sin \Phi \sin \Theta \sin \Psi + \cos \Phi \cos \Psi & \cos \Phi \sin \Theta \sin \Psi - \sin \Phi \cos \Psi \\ -\sin \Theta & \sin \Phi \cos \Theta & \cos \Phi \cos \Theta \end{bmatrix} \mathbf{T}_B \quad (2.7)$$

Na obrázku 2.7 jsou znázorněny všechny tři osy otáčení letounu. Je zde také vidět, jakým směrem dané úhly nabývají kladných hodnot.



Obrázek 2.7: Osy otáčení letounu

2.3 Pohybové rovnice tuhého tělesa

Před samotným vyjadřováním pohybových rovnic si musíme určit několik předpokladů [23]:

1. *Letoun je tuhé těleso*, což znamená, že jakékoliv dva body uvnitř, či na povrchu letounu, zůstávají vůči sobě ve stejné poloze, neboli při působení různě velkých sil zůstane tvar a objem letadla nezměněn.
2. *Země je plochá, nerotující a souřadný systém Země je inerciální.*
3. *Hmotnost letounu je konstantní.* U tohoto předpokladu se počítá s tím, že zanedbáme spotřebu paliva díky čemuž neovlivňujeme hmotnost letounu. Tento předpoklad je důležitý pro použití Newtonových pohybových zákonů.
4. *Letoun je symetrický vůči rovině X_BZ_B .*

Vlivem výše zmíněných předpokladů má let v prostoru tzv. šest stupňů volnosti, znázorněných na obrázku 2.8, kdy se může pohybovat třemi směry posunu a otáčet třemi směry otáčení.

Pohyb letounu v prostoru je popsán pomocí změny jeho pozice, polohy, rychlosti a rychlosti změny úhlů polohy v čase. Tyto změny v čase jsou reprezentovány jako vektory, které mají následující tvar [22]:

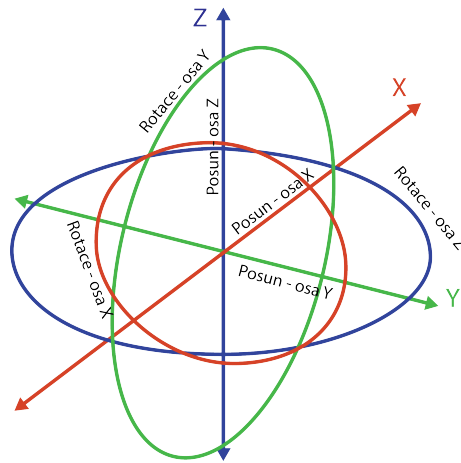
$$\vec{\mathbf{P}}_{\mathbf{N}} = \begin{bmatrix} X_N \\ Y_N \\ Z_N \end{bmatrix} \quad (2.8)$$

$$\vec{\Phi} = \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} \quad (2.9)$$

$$\vec{\mathbf{V}} = \begin{bmatrix} u \\ v \\ w \end{bmatrix} \quad (2.10)$$

$$\vec{\omega} = \begin{bmatrix} p \\ q \\ r \end{bmatrix} \quad (2.11)$$

kde $\vec{\mathbf{P}}_{\mathbf{N}}$ je vektor pozice vyjádřený v soustavě NED. Vektor $\vec{\Phi}$ slouží k určení polohy letounu, kde ϕ je úhel klonění, θ úhel klopení a ψ úhel vybočení. $\vec{\mathbf{V}}$ značí vektor rychlosti se složkami u, v, w určující postupně rychlost podélnou, boční a vertikální. Posledním vektorem je $\vec{\omega}$, kde složky p, q a r jsou úhlové rychlosti klonění, klopení a bočení [23].



Obrázek 2.8: Šest stupňů volnosti

2.3.1 Diferenciální rovnice síly

Při výpočtu diferenciálních rovnic sil působících na letoun během letu, využijeme Newtonova druhého zákona který říká, že součet vnějších sil působících na těleso musí být roven časové změně jeho hybnosti.

Vztah, ze kterého odvodíme jednotlivé rychlosti po všech osách, bude [23]:

$$\vec{\mathbf{F}} = \left. \frac{d(m\vec{\mathbf{V}})}{dt} \right]_B + \vec{\omega} \times m\vec{\mathbf{V}} \quad (2.12)$$

kde $\vec{\omega}$ a $\vec{\mathbf{V}}$ jsou vektory úhlové rychlosti, resp. rychlosti letadla. Vypočtením rovnice 2.12 získáme vztahy pro výpočet působících sil po jednotlivých osách:

$$\begin{aligned} F_X &= m(\dot{u} + qw - rv) \\ F_Y &= m(\dot{v} + ru - pw) \\ F_Z &= m(\dot{w} + pv - qu) \end{aligned} \quad (2.13)$$

kde F_X , F_Y a F_Z závisí na hmotnosti, prostorové orientaci letounu, aerodynamických silách a vektoru tahu. Tyto závislosti jsou vyjádřeny:

$$\begin{aligned} F_X &= \bar{X} + F_T - mg \sin \theta \\ F_Y &= \bar{Y} + mg \sin \phi \cos \theta \\ F_Z &= \bar{Z} + mg \cos \phi \sin \theta \end{aligned} \quad (2.14)$$

Konstanta m představuje hmotnost letounu, g je tíhové zrychlení Země a F_T je tahová síla produkovaná motorem letounu, která působí podél osy X_B ve směru letu. V ostatních směrech je tah nulový. Vektor aerodynamické síly se skládá ze tří složek \bar{X} , \bar{Y} a \bar{Z} . Jsou jimi odpor, vztlak a boční síly a jsou vypočítány následovně [23]:

$$\begin{aligned} \bar{X} &= \bar{q}SC_X(V_T, \alpha, \beta, p, q, r, \delta, \dots) \\ \bar{Y} &= \bar{q}SC_Y(V_T, \alpha, \beta, p, q, r, \delta, \dots) \\ \bar{Z} &= \bar{q}SC_Z(V_T, \alpha, \beta, p, q, r, \delta, \dots) \end{aligned} \quad (2.15)$$

kde \bar{q} je dynamický tlak, S je plocha křídel a C_X , C_Y a C_Z jsou aerodynamické koeficienty získané z aerodynamického tunelu. Dynamický tlak je vypočítán následovně [23]:

$$\bar{q} = \frac{1}{2}\rho V_T^2 \quad (2.16)$$

Kombinací rovnic 2.13 a 2.14, a následném vyjádření \dot{u} , \dot{v} a \dot{w} dostaneme rovnice pro výpočet rychlostí po jednotlivých osách [23]:

$$\begin{aligned} \dot{u} &= rv - qw - g \sin \theta + \frac{1}{m}(\bar{X} + F_T) \\ \dot{v} &= pw - ru + g \sin \phi \cos \theta + \frac{1}{m}\bar{Y} \\ \dot{w} &= qu - pv + g \cos \phi \cos \theta + \frac{1}{m}\bar{Z} \end{aligned} \quad (2.17)$$

Ná základě rychlostí dílčích prvků vektoru rychlosti $\vec{\mathbf{V}}$, můžeme následně vypočítat pravou vzdušnou rychlost letounu V_T , úhel náběhu α a úhel vybočení β . Vztahy pro výpočet jsou následující [23]:

$$\begin{aligned} V_T &= \sqrt{u^2 + v^2 + w^2} \\ \alpha &= \arctan \frac{w}{u} \\ \beta &= \arcsin \frac{v}{V_T} \end{aligned} \quad (2.18)$$

2.3.2 Momentové diferenciální rovnice

Momentové diferenciální rovnice popisují rotaci letounu v čase. Opět je využit Newtonův druhý zákon říkající také, že výsledný moment vnějších sil působící na těleso, je roven časové změně momentu hybnosti. Využijeme vztahu [23]:

$$\vec{\mathbf{M}} = \left. \frac{d(\vec{\mathbf{H}})}{dt} \right]_B + \vec{\omega} \times \vec{\mathbf{H}} \quad (2.19)$$

kde vektor hybnosti $\vec{\mathbf{H}}$ můžeme v letadlovém souřadném systému při použití 1. a 3. předpokladu z části kapitoly 2.3 definovat jako [23]:

$$\vec{\mathbf{H}} = I\vec{\omega} \quad (2.20)$$

Matice momentů setrvačnosti kolem tří os s ohledem na předpoklad symetričnosti letounu má následující tvar [23]:

$$I = \begin{bmatrix} I_{XX} & 0 & -I_{XZ} \\ 0 & I_{YY} & 0 \\ -I_{XZ} & 0 & I_{ZZ} \end{bmatrix} \quad (2.21)$$

Pokud tedy vypočteme rovnici 2.19, dostaneme:

$$\begin{aligned} M_X &= \dot{p}I_{XX} - \dot{r}I_{XZ} + qr(I_{ZZ} - I_{YY}) - pqI_{XZ} \\ M_Y &= \dot{q}I_{YY} + pq(I_{XX} - I_{ZZ}) + (p^2 - r^2)I_{XZ} \\ M_Z &= \dot{r}I_{ZZ} - \dot{p}I_{XZ} + pq(I_{YY} - I_{XX}) + qrI_{XZ} \end{aligned} \quad (2.22)$$

kde M_X , M_Y a M_Z jsou vnější momenty. Ty se rovnají aerodynamickým momentům \bar{L} , \bar{M} , \bar{N} , které jsou vyjádřeny podobně jako aerodynamické síly [23]:

$$\begin{aligned} M_X &= L = \bar{q}SbC_l(V_T, \alpha, \beta, p, q, r, \delta, \dots) \\ M_Y &= M = \bar{q}S\bar{c}C_m(V_T, \alpha, \beta, p, q, r, \delta, \dots) \\ M_Z &= N = \bar{q}SbC_n(V_T, \alpha, \beta, p, q, r, \delta, \dots) \end{aligned} \quad (2.23)$$

kde b je rozpětí křídel a \bar{c} představuje střední aerodynamickou těživu. Kombinací rovnic 2.22 a 2.23 a následném vyjádření úhlových rychlostí dostaneme [23]:

$$\begin{aligned} \dot{p} &= (c_1r + c_2p)q + c_3\bar{L} + c_4\bar{N} \\ \dot{q} &= c_5pr - c_6(p^2 - r^2) + c_7\bar{M} \\ \dot{r} &= (c_8p + c_2r)q + c_4\bar{L} + c_9\bar{N} \end{aligned} \quad (2.24)$$

Koeficienty momentů setrvačnosti jsou definovány následovně [23]:

$$\begin{aligned} \Gamma c_1 &= (I_{YY} - I_{ZZ})I_{ZZ} - I_{XZ}^2 & \Gamma c_2 &= (I_{XX} - I_{YY} + I_{ZZ})I_{XZ} \\ \Gamma c_3 &= I_{ZZ} & \Gamma c_4 &= I_{XZ} \\ c_5 &= \frac{I_{ZZ} - I_{XX}}{I_{YY}} & c_6 &= \frac{I_{XZ}}{I_{YY}} \\ c_7 &= \frac{1}{I_{YY}} & \Gamma c_8 &= I_{XX}(I_{XX} - I_{YY}) + I_{XZ}^2 \\ \Gamma c_9 &= I_{XX} \end{aligned}$$

kdy $\Gamma = I_{XX}I_{ZZ} - I_{XZ}^2$.

2.3.3 Diferenciální rovnice polohy

Polohu letounu určujeme v letadlovém souřadném systému, kdy vyjadřujeme změnu Eulerových úhlů vektoru $\vec{\Phi}$ v čase, na základě vektoru úhlových rychlostí $\vec{\omega}$. Transformace vektoru $\vec{\omega}$ na vektor polohy letounu je dána vztahem [23]:

$$\dot{\vec{\Phi}} = \begin{bmatrix} 1 & \sin \phi \tan \theta & \cos \phi \tan \theta \\ 0 & \cos \phi & -\sin \phi \\ 0 & \frac{\sin \phi}{\cos \theta} & \frac{\cos \phi}{\cos \theta} \end{bmatrix} \vec{\omega} \quad (2.25)$$

Po roznásobení matice s vektorem $\vec{\omega}$ nám vyjdou rovnice pro výpočet změny Eulerových úhlů v čase [23]:

$$\begin{aligned} \dot{\phi} &= p + \tan \theta (q \sin \phi + r \cos \phi) \\ \dot{\theta} &= q + \cos \phi - r \sin \phi \\ \dot{\psi} &= \frac{q \sin \phi + r \cos \phi}{\cos \theta} \end{aligned} \quad (2.26)$$

2.3.4 Diferenciální rovnice pozice

Pozici letounu budeme určovat vůči Zemi pomocí souřadnic X_N , Y_N a Z_N . K tomuto výpočtu využijeme výše zmíněného předpokladu, kdy Země je plochá a nerotující, čili souřadný systém ve kterém se budeme pohybovat, bude inerciální. Pro výpočet pozice využijeme Eulerových úhlů a vektoru rychlosti určující rychlost pohybu letadla v souřadném systému [23].

Souřadnice tedy určíme následovně [23]:

$$\begin{aligned} \dot{X}_N &= (\cos \theta \cos \psi)u + (-\cos \phi \sin \psi + \sin \phi \sin \theta \cos \theta)v + (\sin \phi \sin \psi + \cos \phi \sin \theta \cos \psi)w \\ \dot{Y}_N &= (\cos \theta \sin \psi)u + (\cos \phi \sin \psi + \sin \phi \sin \theta \sin \theta)v + (-\sin \phi \sin \psi + \cos \phi \sin \theta \sin \psi)w \\ \dot{Z}_N &= (-\sin \theta)u + (\sin \phi \cos \psi)v + (\cos \phi \cos \theta)w \end{aligned} \quad (2.27)$$

kdy jsme využili transformační matici z rovnice 2.7 a tu vynásobili vektorem \vec{V} .

2.4 Mezinárodní standardní atmosféra

Při výpočtu aerodynamických sil a momentů využíváme znalosti hodnoty dynamického tlaku. Ten závisí na pravé vzdušné rychlosti letounu V_T a hustoty vzduchu ρ . Hustota vzduchu je na základě modelu mezinárodní standardní atmosféry (ISA) vypočítána jako [16]:

$$\begin{aligned} \rho &= \rho_0 e^{-\frac{g}{287.05T}h} \\ T &= T_0 - 0.0065h \\ a &= \sqrt{1.4 \times 287.05 \times T} \end{aligned} \quad (2.28)$$

kde $\rho_0 = 1.225 \text{ kg.m}^3$ je hustota vzduchu na úrovni střední hladiny moře a $T_0 = 288.15 \text{ K}$ je termodynamická teplota na úrovni moře. Rychlost zvuku a se využívá při výpočtu Machova čísla [16].

2.5 Řešení obyčejných diferenciálních rovnic

Obyčejné diferenciální rovnice popisují závislosti změny veličiny, to znamená, že se v nich vyskytují derivace proměnných. Takovým příkladem jsou i pohybové rovnice popsány v kapitole 2.3. Při hledání řešení obyčejných diferenciálních rovnic se aproximuje toto řešení v bodech úlohy, kdy intervalu dělíme jednotlivé body říkáme integrační krok h [18].

Pro výpočet řešení obyčejných diferenciálních rovnic prvního řádu, kdy je řešena úloha s počátečními podmínkami, existuje několik numerických integračních metod, které dělíme na jednokrokové a vícekrokové. Rozdíl mezi těmito dvěma typy je v přístupu, podle kterého je vypočítána aproximace dalšího bodu. U jednokrokových se vychází pouze z aktuálního stavu, kdežto vícekrokové využívají historii stavů [18].

Nejjednodušší numerickou metodou pro řešení diferenciálních rovnic, je jednokroková, jednoduchá Eulerova metoda. Jedná se však o metodu, která se v praxi moc často nevyužívá. Vzorec Eulerovy metody vypadá následovně [18]:

$$y(t+h) = y(t) + hf(t, y(t)) \quad (2.29)$$

Mnohem přesnějších výsledků při řešení diferenciálních rovnic dosahují metody Runge-Kutta. Metody provádějí výpočty koeficientů, díky kterým je dosaženo při stejném kroku výrazně přesnějších výsledků než při využití Eulerovy metody. Počet koeficientů, které metoda počítá, určuje řád metody. V praxi se nejčastěji používá metoda Runge-Kutta čtvrtého řádu [18]:

$$\begin{aligned} k_1 &= hf(t, y(t)) \\ k_2 &= hf\left(t + \frac{h}{2}, y(t) + \frac{k_1}{2}\right) \\ k_3 &= hf\left(t + \frac{h}{2}, y(t) + \frac{k_2}{2}\right) \\ k_4 &= hf(t+h, y(t) + k_3) \\ y(t+h) &= y(t) + \frac{k_1}{6} + \frac{k_2}{3} + \frac{k_3}{3} + \frac{k_4}{6} \end{aligned} \quad (2.30)$$

2.6 Řídicí systém letounu

Ovládací systém letounu slouží k ovládní letounu během letu a skládá se z ovládacích ploch, ovládacích prvků v kokpitu, kterými se, mimo jiné, ovládají i ovládací plochy a také z propojovací kabeláže [8].

Základními ovládacími prvky v kokpitu jsou:

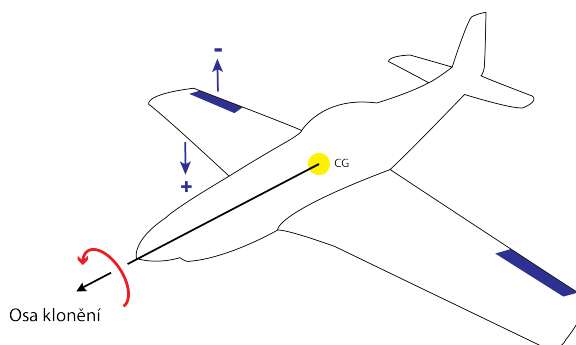
- řídicí páka,
- nožní řízení,
- ovládní vztlakových klapek,
- páka ovládní motoru.

2.6.1 Řídicí páka

Základním ovládacím prvkem v kokpitu je řídicí páka, pomocí které má pilot možnost ovládat výškové kormidlo a křídélka. Páka je obvykle ovládána jednou rukou pohybem do stran a dopředu či dozadu [8].

Křídélka

Běžné letadlo má dvě křídélka, které jsou umístěné na konci každého z křídel. Jedná se o malé plochy, pomocí kterých je možné měnit vztlak na jednotlivých křídlech. Díky tomu je pak ovlivněn úhel klonění letounu [8].



Obrázek 2.9: Křídélka

Křídélka v kokpitu ovládá pilot pomocí řídicí páky pohybem doprava či doleva. Pokud tedy pilot pohne řídicí pákou například doprava, pravé křídélko se pootočí směrem nahoru, kdežto levé naopak dolů. Proudící vzduch pak začne působit na obě křídélka. Vychýlením řídicí páky doprava má za následek snížení vztlaku na pravém křídle. Na levém křídle se vztlak naopak zvýší. Tím vznikne kladný moment a letadlo se začne točit kolem osy X_B v kladném směru. Pokud je pákou pohnuto doleva, nastane stejný efekt v opačném směru. Letoun se tak bude kolem stejné osy točit záporným směrem [8].

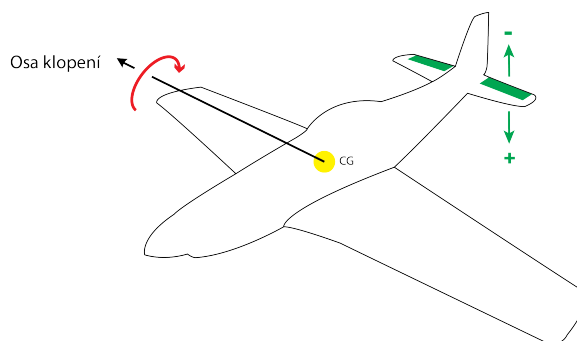
Na obrázku 2.9 je možné vidět, kde se křídélka na letounu nacházejí a jakým způsobem jsou vychýleny. Pokud je tedy pravé křídélko vychýleno do záporné pozice, vzniká kladný moment a letoun se tedy otáčí okolo osy klonění v kladném směru.

Výškové kormidlo

Výškové kormidlo se ovládá, stejně jako křídélka, pomocí řídicí páky, avšak pohybem dopředu či dozadu. Jedná se o dvě horizontálně orientované plochy na konci ocasní plochy letounu, díky kterým může pilot měnit úhel klopení [8].

Aby bylo z ustáleného horizontálního letu dosaženo například kladného nabývání úhlu, musí pilot přitáhnout páku k sobě, čímž pootočí výškové kormidlo směrem nahoru do záporné pozice. Proudící vzduch začne působit na výškové kormidlo, díky čemuž je následně tlačeno ocas letounu dolů. Letadlo se tak začne točit kolem osy Y_B kladným směrem. V takovém případě letoun začne letět při vyšším úhlu náběhu. Vztlaková síla spolu se silou odporovou tak začne stoupat. Opačným pohybem pákou pilot naopak dosáhne záporného klopení letounu [8].

Na obrázku 2.9 je znázorněna pozice výškového kormidla a také, jakým způsobem jsou dvě plochy vychýleny.



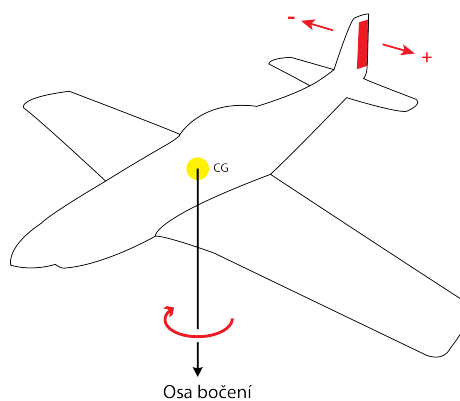
Obrázek 2.10: Výškové kormidlo

2.6.2 Nožní řízení

Dalším základním ovládacím prvkem v kokpitu je nožní řízení skládající se ze dvou pedálů. Každý pilot má k dispozici dva, levý a pravý. Pomocí těchto pedálů je ovládáno směrové kormidlo letadla. U většiny letounů slouží pedály také k řízení kol a jejich brzdění [8].

Směrové kormidlo

Směrové kormidlo je vertikálně orientovaná plocha, která, stejně jako výškové kormidlo, je umístěna v koncové části letounu. Oproti křídélkům a výškovému kormidlu se směrové kormidlo ovládá dvěma pedály. Pomocí směrového kormidla má pilot možnost měnit úhel bočení [8].



Obrázek 2.11: Směrové kormidlo

Při sešlápnutí pravého pedálu, se výškové kormidlo pootočí směrem doprava do záporné pozice. Proudící vzduch začne působit na plochu kormidla. Vznikne tím kladný moment a ocas letadla je tak tlačěn doleva. Přední část se naopak začne točit doprava, díky čemuž se letadlo pootočí kolem osy Z_B kladným směrem. Při opačném postupu, tedy sešlápnutí levého pedálu, se začne letoun točit kolem osy Z_B záporným směrem [8].

Obrázek 2.11 opět zobrazuje pozici plochy spolu se směrem kladného a záporného vychýlení.

2.6.3 Ovládání vztlakových klapek

Dalším prvkem, pomocí kterého je možné ovlivnit pohyb letadla, jsou vztlakové klapky. Nacházejí se na zadních stranách obou křídel. Pomocí klapek je možné měnit zakřivení profilu křídla, díky čemuž vzroste vztlak. Díky zvýšenému vztlaku je tak snížena hranice pádové rychlosti. Vytažením klapek se navíc zvýší i odpor, což může být užitečné při přípravě na přistání, kdy chce pilot snížit rychlost letounu. Typickými situacemi, kdy se klapky využívají, jsou tedy vzlet a přistání letounu [8].

V případě, že se pilot chystá na přistání, vytáhne vztlakové klapky, díky kterým vzroste odpor a vztlak. Zvýšený odpor začne letadlo postupně zpomalovat. Větší vztlak má poté za následek redukování pádové rychlosti. Díky tomu je možné klesat k zemi při nižších rychlostech a přistát tak i na menší vzdálenosti [8].

2.6.4 Páka ovládání motoru

Nedílnou součástí kokpitu letounu je páka pro ovládání výkonu motoru. Každý motor má v kokpitu svou vlastní páku. Pokud je tedy letoun poháněn například čtyřmi motory, pilot má v kokpitu k dispozici čtyři páky pro ovládání motoru. Pomocí této páky je ovládán ventil regulující přísun směsi vzduchu a paliva do motoru. Pokud chce pilot přidat výkon na motoru, pohne pákou směrem dopředu a ventil se otevře. Díky tomu je dodáno do motoru větší množství směsi. Opačným směrem, tedy přisunutím páky k sobě, pilot ventil zavře [8].

Kapitola 3

Principy vizualizace palubních letových dat

Jednou z nedílnou součástí letectví jsou vizualizační prvky či systémy leteckých dat. Bez těchto dat si dnešní let letounem prakticky neumíme představit. Od počátku 20. století, kdy bylo uskutečněny první řízené lety, se za více než 100 let vizualizace dat výrazně změnila. Z původně jednoúčelových přístrojů se časem stávaly multifunkční, které umožňovaly zobrazit několik informací najednou. Letectví také zasáhl rychlý rozvoj digitálních technologií, které dopomohly nejen k pohodlnějšímu čtení leteckých dat, ale i k snazší navigaci.

Historický vývoj vizualizace leteckých dat celkově dělíme do tří období [6]:

- Mechanická éra.
- Elektro-mechanické éra.
- Elektro-optická éra.

3.1 Mechanická éra

Na počátku 19. století, kdy bylo letectví ještě v počátcích, přicházejí na scénu bratři Wrightové se svým strojem, aby v roce 1903 ukázali světu, že je možné do vzduchu vznést stroj lehčí než vzduch. S tímto počínem se zároveň začínají psát dějiny vývoje vizualizačních leteckých systémů, kdy začíná tzv. mechanická éra [6].

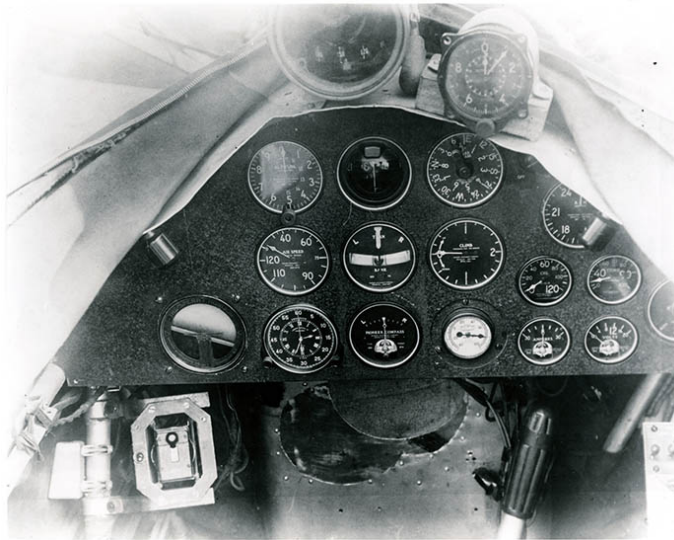
Již první letoun, se kterým bratři Wrightové vzlétli, měl několik přístrojů pro zobrazení leteckých dat. Stroj, symbolicky pojmenován Flyer I., nesl měřič vzdušné rychlosti a ukazatel směru větru, tzv. anemometr, přístroj pro měření otáček motoru a stopky, díky kterým pilot mohl měřit dobu, po kterou se udržel ve vzduchu [6].

Bratři Wrightové se tak stali průkopníky vizualizačních systémů, které se však po dobu jednoho desetiletí výrazně neměnily. Samotné přístroje navíc ani zdaleka nevykazovaly sto procentní spolehlivost a pilot se tak při letu stále musel spoléhat na své smysly [6].

K výraznému pokroku v oblasti letectví pomohla až 1. světová válka, kdy byl kladen větší důraz na spolehlivost a bezpečnost letecké techniky. Díky potřebám armád, které chtěly v novém bojovém prostoru předčit druhou stranu, došlo k výrazným inovacím jak v oblasti výkonu či designu letadel, tak i v oblasti leteckých přístrojů. Velký rozmach letectví se po válce také projevil v civilní sféře. Důležitým milníkem v oblasti vizualizace dat byl let přes Atlantický oceán, který uskutečnil v roce 1927 Charles Lindbergh s letounem Spirit

of St. Louis. Tento letoun měl na své přístrojové desce celkem 11 přístrojů, kterými byly např. rychloměr, výškoměr, zatačkoměr či indikátory tlaku paliva a oleje [6].

Další výraznou událostí, která posunula svět letectví dopředu, byl let poručíka Jamese Doolittla, který o dva roky později po Lindberghově letu, uskutečnil let "naslepo", kdy k letu využíval pouze leteckých přístrojů. Klíčem k úspěchu byla spolupráce s Paulem Kollsmanem, vynálezcem citlivého výškoměru, a Elmerem Sperrym, průkopníkem gyroskopických systémů. Díky této spolupráci, byl Doolittlův letoun Consolidated NY-2 osazen mimo jiné umělým horizontem či Kollsmanovým výškoměrem. Palubní deska letounu je na obrázku 3.1 [6].



Obrázek 3.1: Přístrojová deska Consolidated NY-2 [21]

3.2 Elektro-mechanická éra

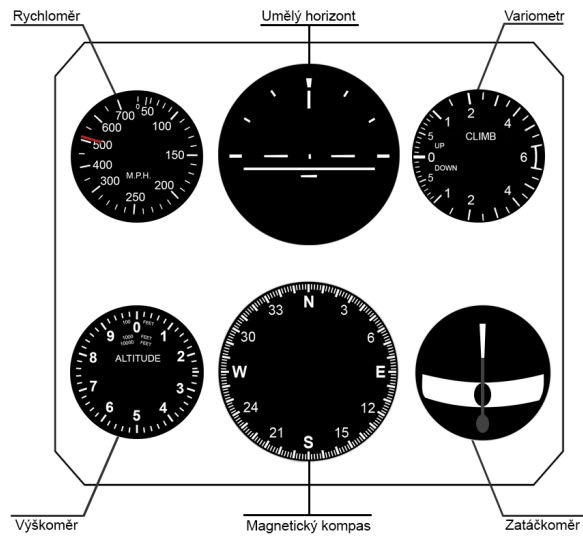
V období 30. let 20. století se na letadlové přístrojové desky dostávají první přístroje založené na elektro-mechanickém principu. Vůbec prvním letounem, který měl všechny přístroje elektrizované, pak byl Lockheed Electra. Narůstal i samotný počet přístrojů, přičemž svou cestu do kokpitu si našly také pokročilé navigační nástroje. Pilot měl tak k dispozici mnoho důležitých informací [6].

Větší počet přístrojů si však vyžadoval větší pozornost. To ovšem způsobovalo problémy, které se projevily hlavně v období 2. světové války, kdy bylo potřeba rychle zjišťovat letové informace. Z důvodu špatně navržené přístrojové desky však bylo čtení leteckých dat z důležitých přístrojů častokrát velmi neintuitivní a mohlo i pilota stát život. Z tohoto důvodu došlo k standardizaci rozložení leteckých přístrojů. Vznikl tak standart známý jako "základní šestice", ze kterého se později vyvinul model "uspořádání do T" [6].

3.2.1 Základní šestice

Základní šestice je standard vytvořen v roce 1938 britskou RAF, který byl ihned aplikován do několika britských letadel, mezi kterými byly stíhací letoun Supermarine Spitfire či čtyřmotorový bombardér Avro Lancaster. Model zahrnoval šest nejdůležitějších přístrojů pro

pilota. Jsou jimi výškoměr, rychloměr, zatáčkoměr, umělý horizont, variometr a magnetický kompas [6]. Schéma základní šestice je vyobrazeno na obrázku 3.2.



Obrázek 3.2: Základní šestice



Obrázek 3.3: Uspořádání do T

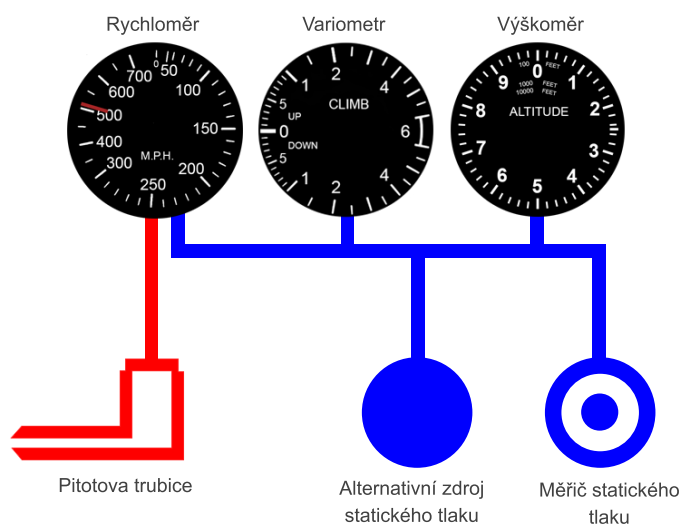
3.2.2 Uspořádání do T

Koncem 50. let 20. století vznikl v USA ze základní šestice nový standard pro rozložení přístrojů známý jako uspořádání do T. Toto uspořádání bylo produktem projektu Integrated Instrument System (IIS), kdy požadavkem bylo umožnění rychlejšího čtení leteckých dat z přístrojů. Výsledkem bylo zobrazování leteckých dat (úhel náběhu, rychlost, vertikální rychlost, výška a Machovo číslo) na vertikálních páskách, které nahradily klasické kulaté budíky. Vznikly také nové přístroje kombinující několik přístrojů ze základní šestice dohromady.

Jsou jimi umělý horizont (ADI) a indikátor horizontální situace (HSI). Model je možné vidět na obrázku 3.3, kde rozložení přístrojů připomíná písmeno T, odtud jeho název [6].

3.2.3 Přístroje vzdušných veličin

Důležitou součástí palubní desky tvoří přístroje vzdušných veličin. Ty využívají dynamický a statický tlak. K měření důležitých dat slouží soustava skládající se s Pitotovy trubice, měřiče statického tlaku a samotných přístrojů [11]. Soustavu je možné vidět na obrázku 3.4.



Obrázek 3.4: Pitot-statická soustava

Pitotova trubice je většinou umístěna pod křídlem, kde do malého otvoru proudí vzduch, který je následně převeden na tlak. Většina Pitotových trubic mají v sobě zabudované vyhřívání, které slouží jako prevence proti zamrznutí. Pokud by trubice zamrzla, mělo by to za následek zobrazování špatných výsledků na přístrojích [11].

Vstup do měřiče statického vzduchu je na letadle naopak umístěn tak, aby výsledek nebyl ovlivněn proudem vzduchu nabíhajícího proti letadlu. Některá letadla mají k dispozici i alternativní zdroje statického vzduchu [11].

Přístroje, které využívají tohoto systému jsou:

- *Rychloměr* - měří rozdíl mezi oběma tlaky, který označujeme jako dynamický. Pokud se zvedne rychlost proudění vzduchu přes Pitotovu trubici, stoupne i dynamický tlak, který pootočí ručičku rychloměru.
- *Výškoměr* - využívá pouze statického tlaku, který funguje jako barometr. Hodnoty zobrazené na přístroji indikují nadmořskou výšku, nikoliv vzdálenost mezi letadlem a zemí. Přístroj je možné také kalibrovat v závislosti na okolních podmínkách.
- *Variometr* - stejně jako u výškoměru, vstupem variometru je pouze statický tlak. Jakmile začne letadlo stoupat, tlak začne klesat. Přístroj poté patřičným způsobem pootočí ručičkou a informuje tak pilota o rychlosti stoupaní [11].

3.2.4 Umělý horizont (ADI)

Umělý horizont ADI vznikl na základě tří přístrojů základní šestky, kde jejich funkce byly zkombinovány do jednoho přístroje. ADI se skládá z umělého horizontu, zatáčkoměru a letového direktoru. Tento přístroj ukazuje pilotovi informace spojené s polohou letadla vůči zemi [6].



Obrázek 3.5: Umělý horizont ADI [5]

Přístroj, který je možno vidět na obrázku 3.5, se skládá ze čtyř částí:

- *Umělý horizont* - nachází se uprostřed přístroje a jedná se o sféru rozdělenou do dvou částí. Horní část je nabarvená modrou barvou a představuje oblohu. Spodní, černá část značí zemi. Napříč sférou prochází stupnice informující o úhlu stoupání, resp. klonění. Úhel klonění letadla, které symbolizuje žlutá značka uprostřed, je znázorněn na horní straně rámečku.
- *Letový direktor* - skládá se ze dvou, na sebe kolmých, linek. Letový direktor je využíván ve spojení s autopilotem. Pokud je autopilot vypnut, pilot používá přístroj k jeho orientaci vůči danému směru. Jakmile je však autopilot zapnut, kurz je nastaven směrem, kam ukazuje letový direktor.
- *Zatáčkoměr* - připojen na spodní straně přístroje.
- *Indikátor poruch* - upozorňují na nesrovnalosti mezi dvěma přístroji. Pokud dojde k takové situaci, je vyžadován zásah posádky [6].

3.2.5 Indikátor horizontální situace (HSI)

HSI, stejně jako ADI, kombinuje funkce tří přístrojů do jednoho. Jsou jimi gyroskopický a magnetický kompas, radiokompas a dálkoměr. Díky HSI vidí snadněji pilot pozici letadla vůči vybranému kurzu a pomáhá tak v jeho orientaci [10].

Přístroj zobrazený na obrázku 3.6 ukazuje hned několik informací:

- *Směr letu* - indikován pomocí magnetického kompasu, který se otáčí vůči značce letadla připevněné uprostřed přístroje.
- *Vzdálenost* - letounu od cíle, která je zobrazena v levém horním rohu.

- *Požadovaný směr* - který se je zobrazen pomocí oranžové šipky na ciferníku kompasu. Tento směr si pilot nastavuje sám pro své potřeby.
- *Kurz k cíli a odchylku letadla vůči němu* - kde kurz je nastaven buďto ručně nebo stanicí vybavenou např. přistávacím systémem ILS. Zobrazován je půlenou šipkou na kompasu spolu s číselnou hodnotou ve stupních v pravém horním rohu. O odchylce letadla vůči nastavenému kurzu informuje bílý proužek uprostřed, kdy pokud letí letadlo správným směrem, tvoří s rozpůlenou šipkou jeden celek [10].



Obrázek 3.6: Indikátor horizontální situace [10]

3.3 Elektro-optická éra

Další etapou ve vývoji vizualizace leteckých dat je elektro-optická éra, kdy se v 70. letech 20. století dostávali do popředí digitální systémy, které zasáhli také do letectví. Díky nástupu nových technologií se výrazným způsobem změnilo principy zobrazování letových dat, kdy jejich čtení bylo pro piloty pohodlnější a nevyžadovalo tolik úsilí. S rozmachem digitalizace se zlepšily také rádiové a navigační systémy [6].

Původně se digitální technologie objevila pouze v armádě, kdy klasické budíky byly nahrazovány displeji typu CRT, které dokázaly zobrazovat většinu důležitých dat. Tyto displeje byly pouze monochromatické v odstínu zelené, avšak do letecké dopravy, která byla rovněž na vzestupu, se nakonec dostaly multifunkční barevné displeje [6].

Výzkum zabývající se využitím těchto technologií proběhl v USA během 70. let 20. století jako součást programu pro vývoj nadzvukové dopravy. Projekt byl však zrušen a přesunut pod křídla NASA, kde výsledku projektu posloužily k vývoji letounu Boeing 737 jako součást programu Terminal Configured Vehicle (TCV). Prvními letouny, které využívaly pokročilých letových displejů, byly Boeing 757 a 767 [6].

3.3.1 Primární letový displej

Primární letový displej je v kokpitu pilotův nejdůležitější zdroj informací. Displej, který kombinuje všechny elektro-mechanické přístroje T-rozložení dohromady, zobrazuje data na principu CRT nebo LCD technologie [20].

Rozložení prvků se u různých výrobců může lišit, avšak většina z nich uprostřed obrazovky zobrazuje funkce přístroje ADI, tedy umělý horizont v kombinaci se zatačkoměrem a letovým direktorem. Rychlost letounu je nejčastěji umístěna na levém okraji obrazovky. Vpravo je potom zobrazen výška a vertikální rychlost. Kompas je obvykle na spodní straně displeje [20]. Příklad primárního letového displeje je možné vidět na obrázku 3.7.



Obrázek 3.7: Primární letový displej [29]

3.3.2 Průhledový displej

Průhledový displej je dnes v letectví široce používané zařízení, které zobrazuje důležité letové data přímo v zorném poli pilota. Díky tomu je čtení informací mnohem pohodlnější a rychlejší. Data, které jsou promítána na malém průhledném displeji, se zdají být daleko před letounem, díky čemuž oči pilota nemusí neustále zaostřovat na menší vzdálenosti [3].

Počátky vývoje průhledových displejů sahají do 40. let 20. století, kdy byly do pilotních kabin montovány holografické zaměřovače, kdy samotný zaměřovač se na displeji pohyboval s ohledem na rychlost letounu a jeho zatačení. Díky tomu bylo pro pilota snadnější zasáhnout nepřítele v manévrovacím boji. Kvality těchto zaměřovačů si oblíbili také piloti



Obrázek 3.8: Průhledový displej [3]

operující v nočních misích, kdy se jim na displeji před nimi zobrazovaly data s navigačních systémů [13].

Výrazným milníkem v oblasti průhledových displejů dosáhlo Královské námořnictvo Velké Británie, které v roce 1958 představilo námořní bitevník Blackburn Buccaneer. Ten byl stavěn na nízké lety ve velké rychlosti, během kterých měl odhazovat bomby na cíl. Pro piloty však bylo velmi náročné sledovat letecké přístroje a bombardovací zaměřovač zároveň, Námořnictvo proto přišlo s kombinovaným zaměřovačem pro primární výzbroj a odhazování bomb s ohledem na rychlost a výšku letounu. V této době byl také poprvé zaveden termín "Head-Up display", tedy "průhledový displej". [14].

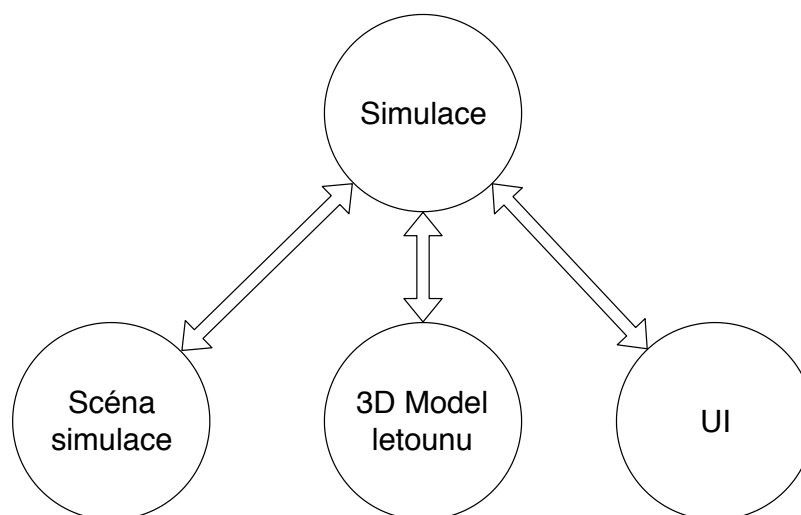
Průhledové displeje byly dlouho dobu doménou pouze armády, nicméně v 70. letech 20. století si našly cestu i do civilního letectví.

Dnes průhledový displeje poskytují výrazně více informací, než tomu bylo v první polovině 20. století. Opět kombinují funkce několika klasických přístrojů do jednoho a jsou tak vlastně druhým primárním displejem [3]. Příklad průhledového displeje je možné vidět na obrázku 3.8.

Kapitola 4

Návrh 3D kokpitu a simulačního jádra

Architektura navrženého simulátoru se skládá z několika entit, kdy centrem celého simulátoru je simulační jádro. Toto jádro vytváří dynamický model letounu, díky kterému je možné simulovat dynamiku letu. Simulátor se dále skládá ze simulační scény, ve které celá simulace probíhá, uživatelského rozhraní a 3D modelu letounu včetně kokpitu. Schéma leteckého simulátoru je zobrazeno na obrázku 4.1.



Obrázek 4.1: Schéma simulátoru

Tato kapitola popisuje návrh simulačního jádra provádějící samotnou simulaci dynamiky letu, tvorbu 3D modelu letounu a kokpitu a návrh letového modelu letounu.

4.1 Technologie použité pro realizaci návrhu

V této sekci jsou popsány technologie a programy, které byly využity pro realizaci návrhů.

4.1.1 3DS Max

3DS Max je program určený pro tvorbu 3D grafiky, vizualizací a animací. Je běžně používán v postprodukci při tvorbě reklam či filmů. Program se často využívá při tvorbě herní grafiky [2].

Při renderování snímků v 3DS Maxu je možné využít několika technologií. Mezi ně patří například radiozita a global illumination. Vestavěn je i skriptovací jazyk MaxScript. Další funkce lze přidávat pomocí API rozhraní a na něm postavených zásuvných modulů [2].

Pomocí vestavěného modulu Character Studio, je možné snadno vytvářet pohyby postav a tvorů, využívající inverzní kinematiky. Vestavěný modul Reactor je nástroj pro fyzikální reakce těles, který simuluje reakce pevných a měkkých těles, textilií, gravitační síly, atd [2].

Program byl využit při modelování 3D modelu letounu a kokpitu. Pomocí 3DS Maxu byly také aplikovány textury na jednotlivé objekty.

4.1.2 Unity 3D

Unity 3D je multiplatformní herní jádro podporující vývoj her pro PC, konzole, mobilní zařízení a web [24].

Poskytuje možnosti vývoje pro 2D i 3D hry libovolného žánru a zaměření. Kromě grafického prostředí pro tvorbu, podporuje také tvorbu skriptů především v jazyce C# a UnityScript, který má podobnou syntaxi jako Javascript. Vývojářům je k dispozici jak bezplatná, tak i placená verze programu [24].

Unity bylo využito jako vývojové prostředí pro tvorbu samotného simulátoru, které poskytuje jednoduchou tvorbu a správu scény. Veškerá logika chování byla naprogramována v jazyce C#, kdy k tomuto jazyku je v Unity k dispozici sada knihoven přizpůsobené právě pro tvorbu v tomto jádru. Dalším důvodem, proč bylo Unity při implementaci využito, je podpora virtuální reality.

4.1.3 Virtuální realita a HTC Vive

Virtuální realita je technologie, která simuluje reálné prostředí a umožňuje tak uživateli s tímto prostředím interagovat. Díky tomu je tak vytvořena iluze, kdy samotný uživatel nabývá dojmu, že je opravdu součástí simulovaného prostředí [28].

Technologie poskytuje širokou škálu využití, mezi které patří například herní průmysl, který tímto poskytuje hráčům zcela nový zážitek. Technologie je také využívána při simulování situací, které jsou v běžném životě nebezpečné, drahé či nepraktické. Mezi tyto situace je možné řadit například trénink pilotů či doktorů [28].

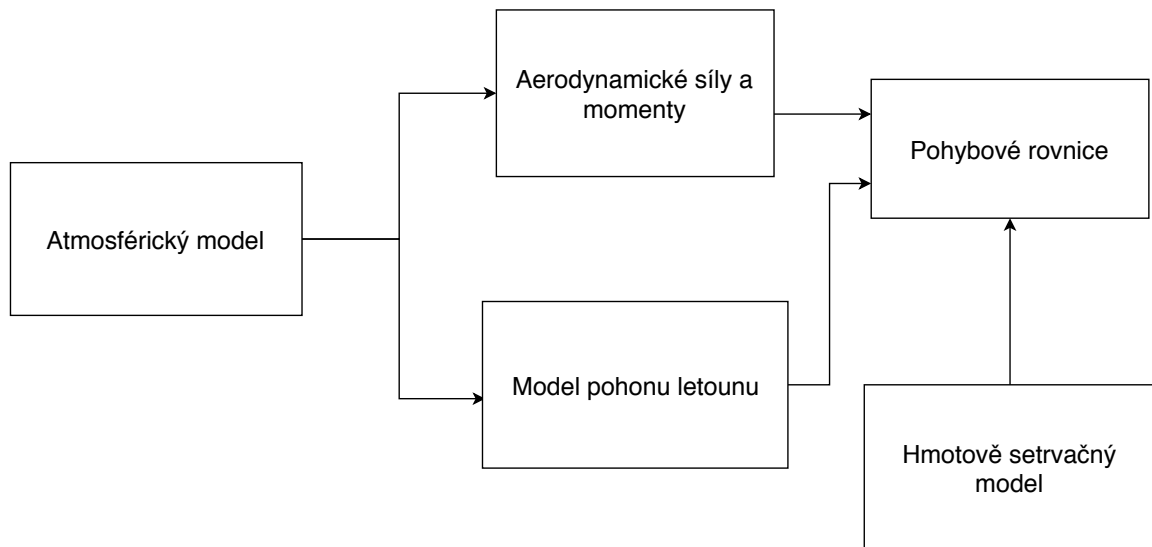
K vytvoření virtuální reality se běžně používají speciální brýle, která obsahují dva malé displeje. Díky nim je možné vytvořit 3D scénu. V brýlích jsou také zabudovány gyroskopy a pohybové senzory, které snímají pohyb hlavy. Prostorový pohyb hlavy je detekován pomocí dvou externích pohybových senzorů [28].

HTC Vive je soustava poskytující virtuální realitu. Skládá se ze speciálních brýlí, dvou vnějších pohybových senzorů a dvou ovládačů, díky kterým je možné interagovat s virtuálním prostředím. HTC Vive umožňuje uživateli procházet prostorem o velikosti až 5x5

metrů. Při průchodu tímto prostorem, je uživatel díky propracovaným systémům informován o překážce nacházející se před ním. Díky tomu je zabráněno možným kolizím [12].

4.2 Simulační jádro

Navržené simulační jádro je rozděleno do několika částí, kdy schéma celého jádra je možné vidět na obrázku 4.2. Struktura jádra je navržena na základě pohybových rovnic a jejich závislostí tak, aby jednotlivé závislosti tvořily logicky oddělené celky, které dohromady utváří jednotné simulační jádro simulující dynamiku letounu.



Obrázek 4.2: Schéma simulačního jádra

4.2.1 Pohybové rovnice

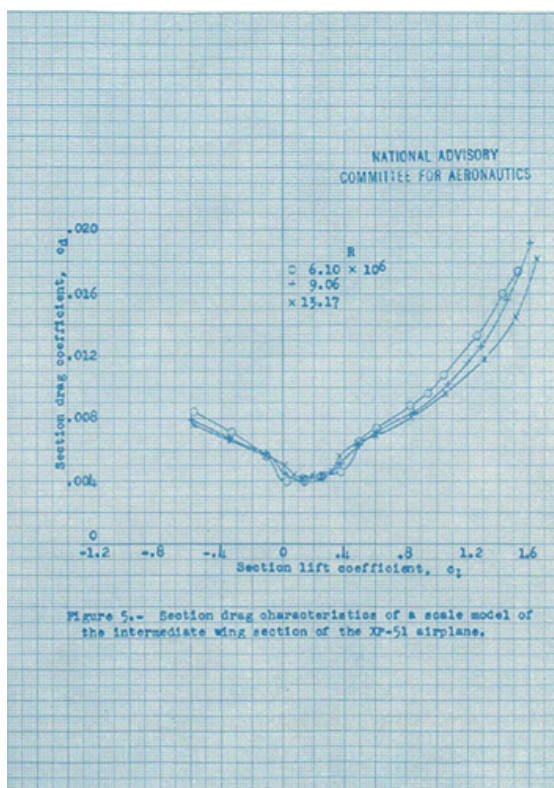
Prvním a velmi důležitým modulem jádra, je modul řešící soustavu dvanácti pohybových rovnic popsanych v kapitole 2. Pomocí těchto rovnic jsou popsány pozice, rychlost, poloha a úhlová rychlost letounu. Díky těmto veličinám je možné v průběhu simulace určit pozici a orientaci letounu v simulované scéně.

4.2.2 Reprezentace aerodynamických sil a momentů

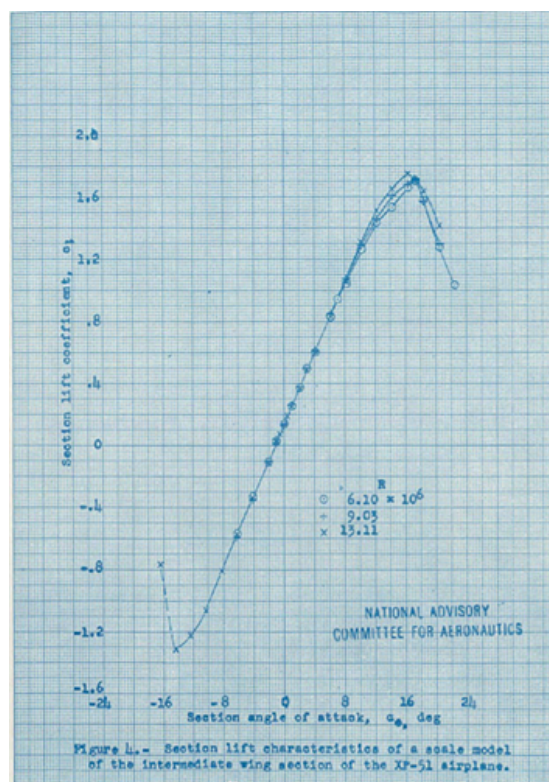
Principy výpočtu aerodynamických sil a momentů jsou rovněž popsány v kapitole 2. Jejich velikost závisí na dynamickém tlaku, geometrických vlastnostech daného letounu a bezrozměrných aerodynamických koeficientech. Tyto koeficienty jsou velmi důležitou složkou, protože nám definují reálné vlastnosti letounu, na který jsou aplikované pohybové rovnice. Bez těchto aerodynamických koeficientů bychom nedocílili věrohodného chování konkrétního stroje, jelikož nám definují stabilitu a ovládací charakteristiky daného letounu.

Každý z aerodynamických koeficientů lze rozdělit na dílčí koeficienty vztahující se k jednotlivým faktorům ovlivňující chování letounu. Mezi tyto faktory patří například rychlost letu, pozice ovládacích prvků nebo úhly náběhu a vybočení [23]. Stejný přístup byl zvolen i při návrhu tohoto simulátoru, kdy jsou nejdříve vypočítány dílčí koeficienty, které po následném sečtení vytvoří celkový aerodynamický koeficient pro danou sílu či moment.

Příklady aerodynamických koeficientů a jejich závislostí je možné vidět na obrázcích 4.3 a 4.4. Koeficienty vyjadřují aerodynamické vlastnosti křídla letounu XP-51, tedy prototypu P-51 Mustang.



Obrázek 4.3: Koeficient odporu křídla letounu XP-51 v závislosti na koeficientu vztlaku [1].



Obrázek 4.4: Koeficient vztlaku křídla letounu XP-51 v závislosti na úhlu náběhu [1].

4.2.3 Atmosférický model

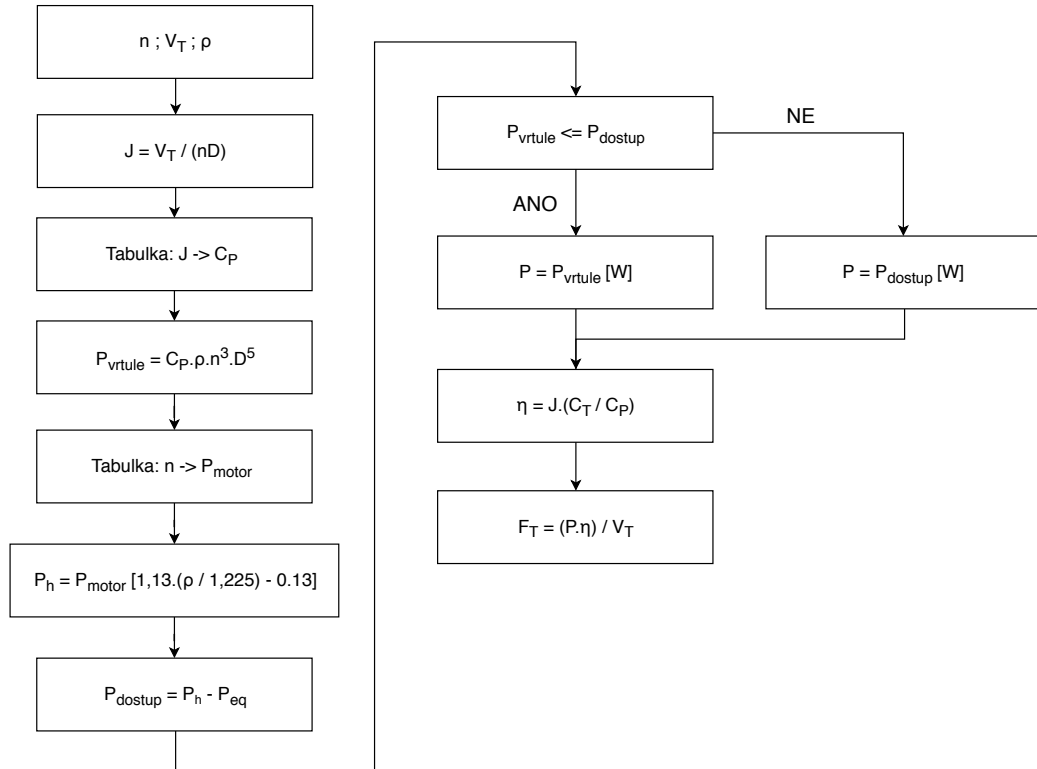
Další důležitou částí jádra je modul pro simulaci atmosféry Země. Při výpočtech se využívá principů a definic modelu standardní atmosféry ISA. Atmosférický model počítá mnoho nezbytných veličin potřebných pro výpočet, mimo jiné i aerodynamických sil a momentů. Mezi tyto veličiny řadíme například dynamický tlak, rychlost zvuku nebo Machovo či Reynoldsovo číslo.

4.2.4 Model pohonu letounu

Posledním, rovněž velmi podstatným modulem letového systému, je modul simulující pohon letounu. Výstupem modulu je velikost tahové síly pohánějící letoun vpřed. V letectví se používá několik typů motorů. Tento letecký simulátor se však soustředí pouze na simulaci vrtulového letounu poháněného pístovým motorem. Modul je proto navržen pro simulaci pouze tohoto typu pohonu.

Jelikož bylo potřeba vytvořit fungující model v relativně krátké době, model motoru byl lehce zjednodušen. Model tak neobsahuje palivový či chladicí systém. Byly však zachovány principy pro výpočet potřebného výkonu. Použitý model byl inspirován z knihy od F. L.

Lewis a B. L. Stevensena [23]. Výraznou roli hrají opět bezrozměrné koeficienty k danému modelu motoru a letounu. Schéma modelu je znázorněno na obrázku 4.5.



Obrázek 4.5: Model pohonu letounu

Vstup tvoří otáčky vrtule n , pravá vzdušná rychlost V_T a hustota vzduchu ρ . Nejdříve je vypočten postupný poměr J , který je získán následovně [23]:

$$J = \frac{V_T}{nD} \quad (4.1)$$

kde D je průměr vrtule. Na základě tohoto poměru je z tabulky výkonových koeficientů získán příslušný koeficient C_P , který je dosazen do rovnice pro výpočet výkonu spotřebovaného vrtulí [23]:

$$P_{vrtule} = C_P \rho n^3 D^5 \quad (4.2)$$

Další potřebnou veličinou je aktuální výkon motoru P_{motor} . Ten je pomocí interpolace získán z tabulky na základě aktuálních otáček motoru. Závislost výkonu na otáčkách je možné získat z grafů v technických dokumentech k danému letounu [31]. Výkon motoru se však s nadmořskou výškou mění a je tak nutné jej přepočítat [23]:

$$P_h = P_{motor} \left[1.13 \left(\frac{\rho}{1.225} \right) - 0.13 \right] \quad (4.3)$$

Nyní je potřeba určit, kolik výkonu má motor pro vrtuli k dispozici. To se zjistí následně [23]:

$$P_{dostup} = P_h - P_{vyb} \quad (4.4)$$

kde P_{vyb} je výkon potřebný pro ostatní zařízení v letounu. Pro zjednodušení není tato spotřeba uvažována a je tedy rovna nule, tudíž vše, co motor vyprodukuje, může poskytnout vrtuli. V dalším kroku se musí rozhodnout, zda je motor schopný poskytnout výkon, který vrtule vyžaduje. Pokud tedy platí $P_{vrtule} \leq P_{motor}$, celkový výkon P je roven P_{vrtule} a vrtuli je tak vyhověno dostatečným výkonem. V opačném případě vrtule obdrží omezený výkon, kdy platí že $P = P_{motor}$ [23].

Nakonec je vypočítán celkový tah produkovaný motorem, resp. vrtulí, kdy formule pro výpočet síly vypadá následovně [23]:

$$F_T = \frac{P\eta}{V_T} \quad (4.5)$$

kde η představuje účinnost vrtule, která se vypočítá následovně [23]:

$$\eta = J \frac{C_T}{C_P} \quad (4.6)$$

C_P je výkonový koeficient využitý v rovnici 4.2 a C_T je koeficient tahový, který se získá opět z datové tabulky na základě postupného poměru.

4.2.5 Iterace výpočtu

Veškeré výpočty probíhají v iteracích během celé simulace. Mezi jednotlivými iteracemi jsou pevně dané intervaly. Je důležité vhodně zvolit velikost intervalu Δt mezi jednotlivými iteracemi. Obecně platí, že čím menší interval zvolíme, tím přesnější hodnoty popisující stav letounu dostaneme. Je však nutné brát ohled na omezený výpočetní zdroj. Musí se tak najít ideální interval, kdy bude dostačovat výpočetní zdroj a zároveň výsledky pohybových rovnic budou dostatečně přesné [32].

Pro řešení pohybových rovnic byla zvolena jednoduchá Eulerova metoda s intervalem 0.003 s. Metoda je sice nejjednodušší ze všech numerických metod, při takhle malém intervalu však poskytuje dostatečnou přesnost výpočtu. Takto malý interval bylo možné zvolit hlavně díky tomu, že během simulace nedochází k velkému množství výpočtů. Pokud by tomu bylo jinak, musela by být využita jedna ze složitějších numerických metod pro řešení diferenciálních rovnic.

4.3 3D model letounu a kokpitu

V leteckých simulátorech jde primárně o simulaci reálného chování letounu v prostoru, případně jiných strojů schopné vznést se do vzduchu a udržet se v něm. Nedílnou součástí je však i vizualizace celé simulace. Je proto nezbytné vytvořit, mimo jiné, i 3D modely konkrétních strojů a jejich kokpitů, které jsou následně připojeny k simulátoru.

Model letounu, vytvořen pro tento letecký simulátor, byl inspirován známým stíhacím letounem z období 2. světové války, North American P-51D Mustang. Výběr byl učiněn ze dvou důvodů. Jednak z důvodu mé oblíbenosti ke strojům z daného období, a jednak z důvodu toho, že ke zvolenému modelu je na internetu k dispozici mnoho technických dokumentací obsahující potřebná data k vytvoření příslušného 3D a letového modelu.

4.3.1 P-51D Mustang



Obrázek 4.6: P-51D Mustang [9]

P-51D Mustang byl jednomístný americký stíhací letoun z období 2. světové války vyráběný firmou North American Aviation. Jednalo se o záchytný stíhač dlouhého doletu (až 3700 km), který od roku 1944 sloužil jak v Evropě, kde většinou doprovázel spojenecké bombardéry při náletech na Německo, tak i nad Tichým oceánem proti Japonsku. Mustang byl u pilotů velmi oblíbený a platil za jednu z nejrychlejších vrtulových stíhaček v období války [25].

Poháněn byl motorem Packard V-1650-7 Merlin o maximálním bojovém výkonu 1720 hp (1282.6 kW). Jedná se licencovanou verzi legendárního motoru Rolce-Royce Merlin nasazených do britských stíhaček Spitfire. Díky tomuto motoru byl Mustang schopný dosáhnout maximální rychlosti až 705 km/h a to ve výšce 7620 metrů [25].

Výzbroj byla tvořena 6 kulometry Browning M2 ráže 12.7 mm, kdy v každém křídle byly po 3 kulometech. Mohl také nést maximálně 908 kg pum a 6 raket HVAR kalibru 127 mm [25].

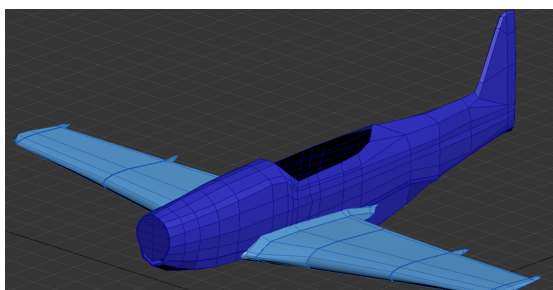
4.3.2 Tvorba 3D modelu

3D model Mustangu byl vytvořen ve studentské verzi 3DS Maxu na základě technického nákresu letounu (diazotypie, blueprint), kdy při modelování byly využity bokorys, půdorys a přední pohled letounu. Pro doladění modelu byly využity fotografie reálného stroje. Jednotlivé ovládací prvky letounu a vrtule jsou modelovány jako samostatné objekty, na které se následně aplikují vizuální transformace reagující na vstup z kláves.

Jelikož pro modelování kokpitu nebyl nalezen rozumný technický nákres, při vytváření modelu kokpitu posloužily fotografie zachycující kokpit z různých úhlů pohledu. Velmi přínosným byl manuál k P-51D ze hry DCS [7], pomocí kterého bylo také možné věrohodně rekonstruovat ukazatele na přístrojové desce. Výsledný kokpit je však jistým způsobem zjednodušen. Nebyly namodelovány některé přístroje nacházející se v kokpitu, jelikož nejsou v simulátoru využívány. Hlavní rozložení ovládacích prvků a přístrojové desky však zůstalo zachováno. Oproti vnějšímu modelu má kokpit výrazně větší počet samostatných objektů. Jsou jimi ovládací prvky spojené s ovládáním letounu, a ručičky jednotlivých budíků ukazující letecké data.

Modelování samotného modelu probíhalo v několika etapách:

- Prvním etapou byla příprava bluperintu. Ten se musel rozdělit, například přímo v 3DS Maxu, na 3 klíčové části kterými jsou půdorys, bokorys a pohled z přední části. Tyto části se následně aplikovaly na připravené plochy jako textura. Plochy se vůči sobě nacházely v kolmé pozici podél jednotlivých os.
- Jakmile byla předloha letounu připravena, přišlo na řadu vytváření low-poly modelu, tedy hrubého náznaku tvaru letounu. Jedná se o postup, kdy je vybráno vhodné primitivum (základní těleso), ze kterého se následně tvaruje hrubý náznak modelu. Trup a křídla byly modelovány samostatně.
- Další etapou bylo zpřesňování geometrie pomocí přidávání hran do modelu. Díky tomu byl model složen z více polygonů a jeho tvar byl tak přesnější. Na obrázku 4.7 je možné vidět model trupu a křídel již se zpřesněnou geometrií.

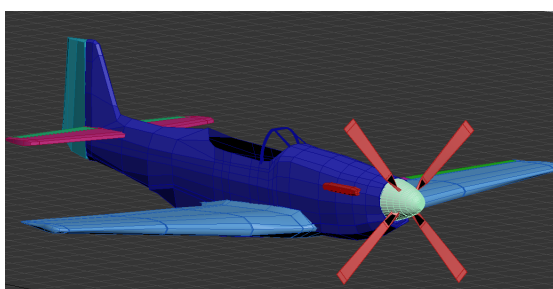


Obrázek 4.7: Křídla a trup se zpřesněnou geometrií

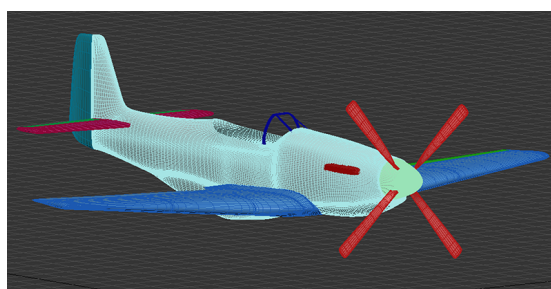


Obrázek 4.8: Model podvozkového kola před aplikováním Turbosmooth

- Po dostatečném zpřesnění geometrie trupu a křídel byly na místech ovládacích ploch letounu odstraněny části modelu a na místo nich vytvořeny modely jednotlivých ploch. Dále byly namodelovány vrtule a podvozek. Model podvozku je zobrazen na obrázku 4.8.



Obrázek 4.9: Model před aplikováním Turbosmooth

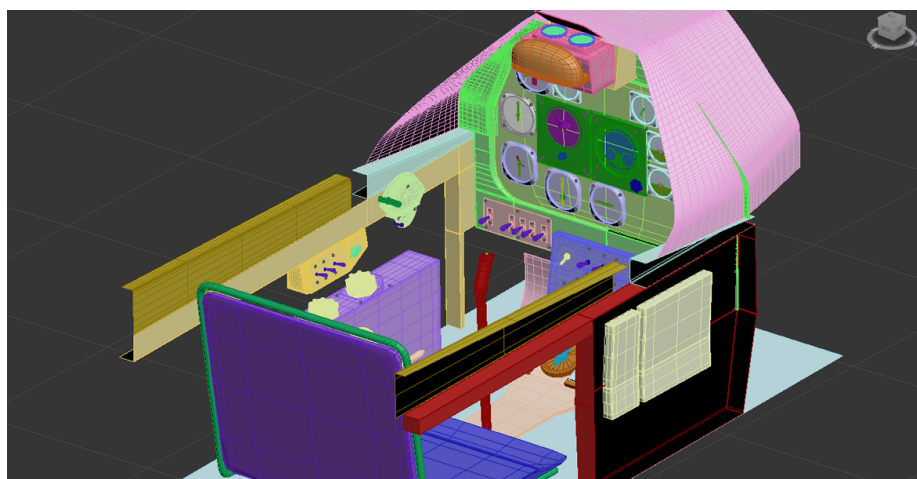


Obrázek 4.10: Model po aplikování Turbosmooth

- V další etapě modelování se na jednotlivé objekty aplikoval modifikátor Turbosmooth. Na obrázcích 4.9 a 4.10 je vidět rozdíl mezi modelem, na který ještě Turbosmooth použit nebyl, a modelem, na který už ano. Jedná se o modifikátor, který objektům výrazně přidá počet polygonů a zjemní jejich tvary. Po aplikování Turbosmooth některé

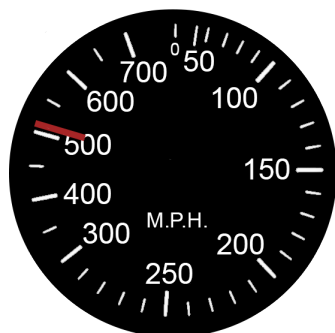
objekty však nedosahovaly požadovaných tvarů. Chyběly například ostré hrany v některých částech trupu a křídel. Bylo tak nutné pomocí různých technik zpřesňování geometrie model upravit.

- Jakmile byla geometrie vyladěná a model dosahoval požadovaného vzhledu, přišla na řadu tvorba kokpitu. V tomto případě nebyl k dispozici žádný blueprint a veškeré modelování probíhalo na základě několika fotografií a intuice. Postup byl prakticky stejný jako u modelování vnějšího modelu. Nejdříve se vytvořily low-poly modely objektů, ze kterých se skládá celý kokpit. Poté se geometrie objektů zpřesnila, aby na ně mohl následně být aplikován modifikátor TurboSmooth. Výsledný model kokpitu je zobrazen na obrázku 4.11.



Obrázek 4.11: Konečný model kokpitu

- Po dokončení kokpitu, bylo nutné provést spojení kokpitu s modelem letounu, jelikož modelování probíhalo odděleně.
- Závěrečnou etapou bylo aplikování textur pomocí UVW mapování. Jedná se o techniku, kdy je vytvořena mapa povrchu objektu, která je "rozstříhána" na několik částí.



Obrázek 4.12: Textura rychloměru



Obrázek 4.13: Přístrojová deska s otexturovanými přístroji

Na základě mapy je pak možné v grafickém programu vytvořit požadovanou texturu objektu. Jedná se o jednoduchý a přehledný způsob mapování textur. Příklad textury

jednoho z přístrojů je možné vidět na obrázku 4.12. Textura je shodná s reálnou podobou ukazatele rychlosti v P-51D. Na obrázku 4.13 je poté zobrazena přístrojová deska se všemi otexturovanými přístroji.

4.4 Letový model

Samotné 3D modely jsou jen vizuální realizací, které samy o sobě nezajistí chování strojů odpovídající realitě. Je tak nutné vytvořit i příslušný letový model obsahující mnoho vlastností vztahující se k danému letounu, které jsou nezbytné k věrohodné simulaci. Těmito vlastnostmi jsou např. geometrické vlastnosti letounu, aerodynamické koeficienty či vlastnosti spojené s pohonem letounu. Většinou se jedná o jeden či více souborů, která tato data obsahuje. Obsah a typ souborů se může lišit v závislosti na realizaci jednotlivých simulátorů.

Tento letecký simulátor je postaven na simulaci jednoho konkrétního letounu. Základní technická data, získaná z technických manuálů a publikací [17][25][30], jsou tak již pevně nastavena do příslušných proměnných. Tabulky s koeficienty spojené s aerodynamikou a pohonem letounu je však jednodušší vytvářet dynamicky načtením dat z příslušného souboru při spuštění simulátoru.

Pro tento účel byl vytvořen soubor typu XML, který obsahuje dva typy tabulek, ve kterých jsou dané koeficienty uloženy. Typy tabulek se liší v tom, zda je konkrétní koeficient funkcí jedné či dvou proměnných.

Jak bylo zmíněno v předchozí sekci, výpočet aerodynamických sil a momentů v navrženém simulátoru probíhá na principu počítání dílčích koeficientů. Výpočet celkových koeficientů, v aerodynamickém souřadném systému, vypadá následovně [23]:

Silové koeficienty

$$\begin{aligned} C_D &= C_{D_\alpha} + C_{D_{Mach}} + C_{D_\beta} + C_{D_{Flaps}} \delta_F + C_{D_{Gear}} + C_{D_\eta} \eta + C_{D_\xi} \\ C_Y &= C_{Y_\beta} \\ C_L &= C_{L_\alpha} + C_{L_{Flaps}} \delta_F + C_{L_\eta} \eta \end{aligned} \quad (4.7)$$

Momentové koeficienty

$$\begin{aligned} C_l &= C_{l_\beta} + C_{l_{p^*}} p^* + C_{l_{r^*}} r^* + C_{l_\xi} + C_{l_\alpha} + C_{l_\zeta} \zeta \\ C_m &= C_{m_\alpha} + C_{m_{Mach}} + C_{m_{Flaps}} \delta_F + C_{m_{Gear}} + C_{m_\eta} + C_{m_{q^*}} q^* \\ C_n &= C_{n_\beta} + C_{n_{r^*}} r^* + C_{n_\zeta} \zeta \end{aligned} \quad (4.8)$$

kde p^* , q^* a r^* jsou bezrozměrné úhlové rychlosti, díky kterým zůstane výsledný koeficient bezrozměrný [23]:

$$\begin{aligned} p^* &= p \frac{b}{2V_T} \\ q^* &= q \frac{\bar{c}}{2V_T} \\ r^* &= r \frac{b}{2V_T} \end{aligned} \quad (4.9)$$

Hodnoty jednotlivých aerodynamických koeficientů byly vypočteny pomocí DATCOMu [19], který je schopen na základě geometrických vlastností letounu vypočítat jednotlivé dílčí koeficienty.

Kapitola 5

Implementace

V této kapitole bude popsána implementace všech součástí leteckého simulátoru. Nejdříve je přiblížen návrh a následná implementace uživatelského rozhraní. Následuje sekce zabývající se tvorbou scény, ve které simulace probíhá. Dále je popsána implementace letounu v Unity. Ta zahrnuje kolizi letounu s terénem, logiku kamer, transformaci řídicích prvků a ukazatelů letových dat, a také ovládání motoru. Následuje popis realizace simulačního jádra simulátoru. Poslední část kapitoly je věnována implementaci virtuální reality do prostředí leteckého simulátoru.

5.1 Uživatelské rozhraní

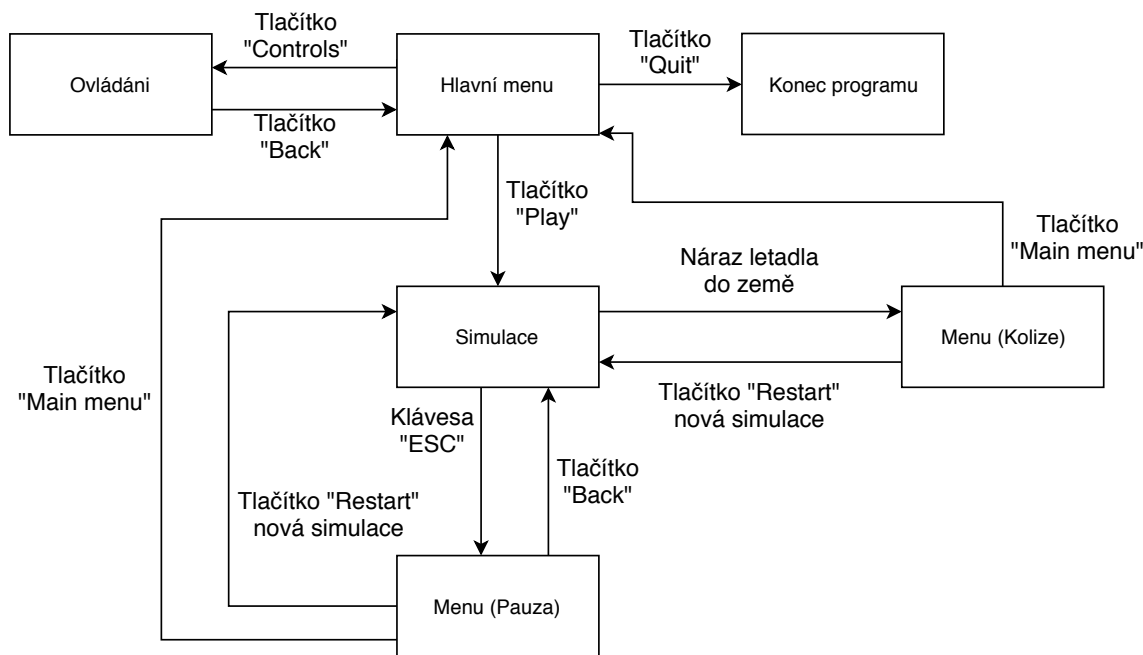
Uživatelské rozhraní představuje prostředníka mezi uživatelem a programem, pomocí kterého tyto dvě entity spolu komunikují. Je stěžejní navrhnout a následně implementovat uživatelské rozhraní tak, aby působilo na uživatele jednoduše a intuitivně. Uživatel musí vědět, co každý prvek rozhraní znamená a také jak celý program ovládat.

5.1.1 Ovládání aplikace

Program je rozdělen do dvou hlavních scén, kdy tou první je scéna *Menu*, ve které je implementována hlavní nabídka. Ve druhé scéně *Game* probíhá samotná simulace. Architekturu programu je také možné popsat pomocí stavového automatu, který je znázorněn na obrázku 5.1. V průběhu programu tak uživatel přechází mezi jednotlivými stavy, které jsou vyvolány určitými akcemi. Jsou jimi buď stisk příslušného tlačítka, nebo náraz letounem do země.

Při spuštění programu je načtena scéna *Menu*. Program se tak nachází ve výchozím stavu *Hlavní menu*. Načtena je tak hlavní nabídka, která obsahuje tři tlačítka:

- Tlačítko "Play"- Tímto tlačítkem je spuštěna samotná simulace letu. Program načte scénu *Game*, přičemž přejde do stavu *Simulace*.
- Tlačítko "Quit"- Uživatel stisknutím tohoto tlačítka ukončí program, který přejde do stavu *Konec programu*.
- Tlačítko "Controls"- Po stisknutí přejde program do stavu *Ovládání*. Zde si uživatel může zobrazit buďto kompletní seznam kláves, pomocí kterých je možno simulaci ovládat, nebo způsob ovládání pomocí HTC Vive ovládačů. Do hlavní nabídky se uživatel vrátí opět pomocí tlačítka "Back".



Obrázek 5.1: Stavový automat programu

Po spuštění simulace, je možné ji přerušit několika způsoby. Prvním z nich je případ, kdy uživatel stiskne klávesu "ESC". Běh simulace se pozastaví a program se dostane do stavu *Menu (Pauza)*, kdy je zobrazena uživateli nabídka se třemi tlačítky. Pomocí tlačítka "Back" může uživatel pokračovat v simulaci. V případě, že je stisknuto tlačítko "Restart", dojde k opětovnému načtení scény *Game*. V obou případech se program vrátí do stavu *Simulace*. Stisknutím tlačítka "Main menu" uživatel zruší celou simulaci a je přesměrován do hlavní nabídky. Program načte scénu *Menu* a přesune se do stavu *Main menu*.

Druhým způsobem, jakým je možno přerušit simulaci, je narazit letounem do země a vyvolat tak kolizi. V takovém případě je uživateli zobrazena nabídka, která jej informuje o havárii. K dispozici má pak dvě možnosti. Buďto restartovat simulaci pomocí tlačítka "Restart", nebo se vrátit do hlavní nabídky stisknutím tlačítka "Main menu".

Obsluhu tlačítek většinou provádějí vestavěné skripty, které jsou v základu aplikované na objektech typu *Button*. Pomocí grafického rozhraní v Unity se skriptu předá objekt a událost, která má být nad objektem provedena při stisknutí daného tlačítka. Jedná se většinou o situace, kdy je potřeba pouze skrýt nějaký objekt ve prospěch jiného. Pokud je však nutno složitějších událostí, jako třeba změna barvy tlačítka, či načtení jiné scény, jsou využity vlastní skripty.

Jedním z nich je skript *MainMenu.cs*, který při stisknutí na příslušné tlačítko buďto načte novou scénu, nebo vypne program. Dále je to skript *PMenu.cs* obsluhující přerušovací nabídku.

5.1.2 Ovládání letounu

Mnoho leteckých simulátorů podporuje ovládání pomocí analogových ovládacích prvků, kterými jsou logickou volbou pro uživatele požadující reálný zážitek ze simulace. Analogové ovládací prvky však nejsou součástí klasické výbavy počítačových sestav. K ovládání základních ovládacích prvků se tak nabízí klávesnice.

Navržené rozložení kláves pro ovládání většiny ovládacích prvků je soustředěné kolem kláves W,A,S a D. Toto rozložení je vhodné pro ovládání levou rukou. Zvolené rozložení je inspirováno počítačovými hrami, kde je potřeba intuitivního rozložení kláves z důvodu rychlých reakcí. Ne jinak je tomu i u leteckých simulátorů. Většina kláves určená k ovládání letounu je tak v dobrém dosahu všech prstů levé ruky. Díky tomu uživatel nemusí nahmatávat klávesy po celé klávesnici. Výjimku jen představuje ovládání otáček a nastavení pamatováčku na kompasu. Otáčky se ovládají šipkami a pamatováček kompasu se nastavuje pomocí kláves P a L. Zde je vhodné klávesy ovládat pravou rukou. Je využita i myš, díky které je možné se během simulace rozhlížet.

Ovládání simulátoru pomocí klávesnice je implementováno následovně:

- W, S - Ovládání výškovky (S - nahoru, W - dolů),
- A, D - Ovládání křidélek (A - levá nahoru, D - pravá nahoru),
- Q, E - Ovládání směrovky (Q - doleva, E - doprava),
- R, T - Ovládání klapek (R - vysunout, T - zasunout),
- G - Zasunutí/Vysunutí podvozku,
- ↑, ↓ - Ovládání otáček motoru (↑ - zvýšit otáčky, ↓ - snížit otáčky),
- P, L - Ovládání pamatováčku (P - rotace doprava, L - rotace doleva),
- V - Změna pohledu (vypnuté VR),
- C - Rozhlížení.

Jelikož simulátor podporuje i virtuální realitu, bylo navrženo ovládání také na dva ovládače HTC Vive. Implementace virtuální reality spolu s ovládáním je popsáno v jedné z dalších sekcí této kapitoly.

5.1.3 Zobrazení simulačních dat

Důležitou součástí uživatelského rozhraní simulátoru je zobrazení simulačních dat uživateli. Mezi tato simulační data, v případě leteckého simulátoru, patří např. rychlost, otáčky nebo výška letounu. Jelikož simulátor poskytuje dva způsoby pohledu uživatele, kokpitu a pozice za letounem, je vhodné navrhnout a vytvořit dva způsoby zobrazení simulačních dat.

Způsob, jakým jsou zobrazena simulační data při pohledu z kokpitu, je zřejmý. Veškerá data jsou zobrazována pomocí reálných 3D ukazatelů na přístrojové desce. Ukazatele však nezobrazují všechny simulační data. Například přístroj pro zobrazování pozice vztlakových klapek či podvozku na přístrojové desce neexistuje. To lze odvodit pouze z pozice příslušných pák ovládající daný prvek. Proto je nutné aplikovat transformace i na tyto 3D modely ovládacích prvků.

Při pohledu z pozice za letounem však uživatel nemá k dispozici žádné 3D modely přístrojů. Virtuální realizace jednotlivých ukazatelů někde na obrazovce by byla velmi nepraktická. Simulační data je tak nutné zobrazit textově. Počet zobrazených dat je na rozdíl od kokpitu zredukován na nejdůležitější veličiny potřebné pro udržení letounu ve vzduchu. Těmito veličinami jsou rychlost, výška a počet otáček motoru. Tyto veličiny jsou ještě doplněny o pozici vztlakových klapek a podvozku. Celkové rozložení zobrazení simulačních dat je zobrazeno na obrázku 5.2.



Obrázek 5.2: Rozložení textového výstup simulačních dat na obrazovce

5.2 Scéna simulace

Důležitým prvkem simulátoru je scéna, ve které je celá simulace uskutečněna. Při absenci jakéhokoliv herního světa, by se stroj pohyboval pouze v prázdném prostoru bez možnosti jakékoliv orientace, čímž by simulace ztrácela význam. Navržená scéna navíc plní i roli estetickou, kdy uživatel má dojem, že se pohybuje v reálném prostředí.

Při vytváření krajiny je nutné myslet na její rozměry, které by měly být dostatečně velké. V leteckých simulátorech jde primárně o simulaci letu, kdy se letoun pohybuje prostorem v relativně velkých rychlostech. Pokud by scéna měla rozměry pouze 2x2 kilometry, uživatel by se velice brzo dostal na její konec a musel by tak změnit směr letu. Je proto vhodné vytvořit svět ve větším měřítku.

Na úkor rozměru můžou být naopak sníženy detaily prostředí, které pomáhají scéně k realističtější podobě. Pokud se počítá s tím, že po většinu času simulace uživatel stráví pohybem ve vzduchu, do prostředí nemusí být zasazeny objekty např. trávy či velký počet budov. Stejná myšlenka platí i pro textury aplikované na terén či objekty. Úroveň detailů je určena celkovými požadavky na výsledný produkt.

Tato kapitola nejdříve popisuje princip, jakým je scéna v Unity reprezentovaná a následně postup, jakým byla celková scéna simulace vytvořena.

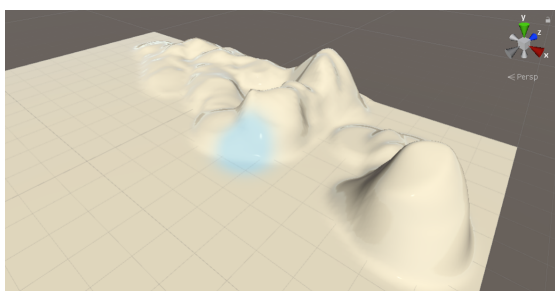
5.2.1 Základní scéna v Unity

Základní scéna v Unity se skládá z kamery a světelného zdroje. Do této scény je následně možné vkládat, kromě nových kamer a světelných zdrojů, také samotné objekty scény. Scéna jako taková se jeví jako jeden globální souřadný systém. V tomto systému se nacházejí všechny objekty scény, které jsou ve scéně uloženy hierarchicky. Pokud je aplikovaná transformace na nadřazený uzel, jsou globálně ovlivněny všechny podřízené uzly. Základní transformace, které je možné aplikovat na jednotlivé uzly, jsou posun, rotace a změna velikosti.

V Unity je možné vytvářet několik scén, mezi kterými je možné v průběhu aplikace přecházet. To je možné využít například při vytváření hlavní nabídky, kdy při spuštění aplikace je načtena scéna s hlavní nabídkou a při provedení příslušné akce, se načte hlavní scéna se hrou.

5.2.2 Tvorba celkové scény simulace

Tvorba celkové scény simulace byla rozdělena do několika etap. V první etapě byl vytvořen objekt *Terrain*, který v Unity slouží jako reprezentace terénu. Rozloha terénu byla nastavena na 10x10 km. Nově vytvořený terén je však v základu jen rovná plocha, kterou tak bylo nutné vyprofilovat do podoby připomínající reálnější krajinu. Za účelem profilování byly využity nástroje, které poskytuje samotný objekt *Terrain*. Při samotném procesu profilování, využívající princip výškových map, je možné vidět výsledek ihned v náhledu scény (obr. 5.3), díky čemuž je celkový proces urychlen. Výsledný terén (obr. 5.4) má hornatý profil, kdy některé hory dosahují výšky až 2000 m. Hory, tvořící přirozené překážky pro letoun, jsou umístěny převážně po okraji mapy. Tvoří tak přírodní horizont krajiny.

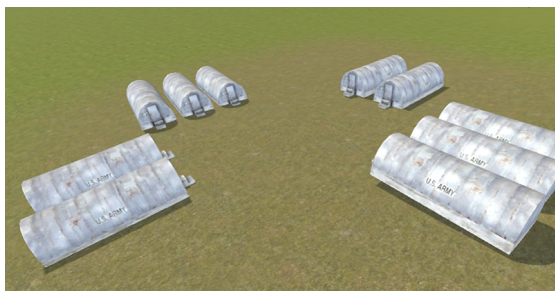


Obrázek 5.3: Profilování terénu

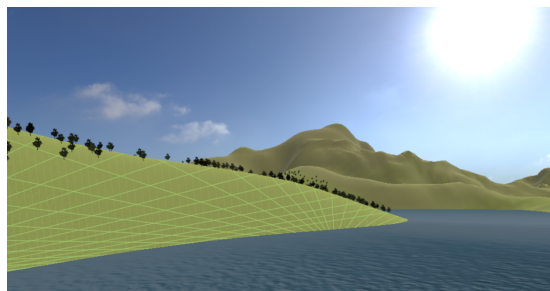


Obrázek 5.4: Výsledná podoba terénu

V druhé etapě byly na výsledný terén aplikovány textury. Stejně jako profilování terénu proběhlo texturování přímo v editoru Unity pomocí nástroje pro aplikování textur, který rovněž nabízí objekt *Terrain*. Na celý terén byla nejdříve aplikovaná jedna primární textura trávy. Na základní texturu se postupně vrstvy další, které byly nanášeny pomocí štětce určený pro aplikaci textur. Díky této technice byly vytvořeny cesty a oblasti značící změnu výškového profilu krajiny.



Obrázek 5.5: Vojenská základna ve scéně



Obrázek 5.6: Pohled na výslednou krajinu

Další etapa zahrnovala vkládání detailů v podobě budov, stromů a vodních ploch. Pomocí stromů, jako součást vývojářského balíčku podporující vývoj krajiny v Unity¹, byly vytvořeny oblasti lesů. Tvorbu lesů urychlil nástroj umožňující hromadné umístování stromů na terén. Nástroj umožňuje nastavit hustotu, s jakou se budou stromy na terén vkládat a také velikost barevných rozdílů jednotlivých stromů. Hustota stromů, tvořící výsledné lesní plochy, byla zvolena tak, aby nebylo vkládáno příliš velké množství stromů, které by výrazně ovlivňovalo výpočetní výkon. Scéna sice neobsahuje žádné městské zástavby, byly do

¹World Builder - <https://assetstore.unity.com/packages/tools/terrain/world-builder-11333>

ní však zasazeny objekty vojenských bunkrů. Bunkry byly umístěny vedle sebe do skupin připomínající vojenské základny (obr. 5.5). Modely bunkrů byly rovněž staženy z obchodu Unity². Vodní plochy jsou ve scéně reprezentovány jako plochy, na které jsou jen aplikovány textury imitující vodu.

Jakmile byla vytvořena krajina, následovalo vytvoření reálné oblohy, která je realizována jako skybox. Skybox je obal okolo celé scény, na kterém je textura prostředí nacházející se za horizontem vytvořené krajiny. Z obchodu Unity byl stažen balíček poskytující reálné textury denní a noční krajiny³. Jedná se o fotografie, kdy spodní část fotografií tvoří povrch země. Spodní část textury však byla zakryta pomocí efektu mlhy, kterou je možné v nastavení textury zapnout a nastavit její hustotu. Díky efektu mlhy je tak vidět pouze obloha. Pohled na výslednou krajinu je možné vidět na obrázku 5.6.

5.3 Implementace letounu v Unity

3D model letounu, vytvořený v 3DS Maxu, byl importován do Unity a následně umístěn do scény. Jelikož model nebyl v 3DS Maxu zcela otexturován, bylo vytvořeno několik materiálů v Unity, které byly následně aplikovány na neotexturované části letounu. Výslednou podobu letounu a kokpitu je možné vidět na obrázcích 5.7 a 5.8.



Obrázek 5.7: Konečný model letounu v Unity



Obrázek 5.8: Konečný model kokpitu letounu v Unity

Nad vloženým modelem pak bylo nutné vytvořit hlavní uzel nacházející se v hierarchii scény nejvýše. Nad tímto nadřazeným uzlem jsou poté během simulace prováděny transformace. Jsou jimi změna pozice a rotace (orientace) letounu. Transformace provádí skript Simulator.cs, který je aplikován na nově vzniklý uzel.

Na hlavní uzel jsou krom skriptu Simulator.cs, který je popsán v další sekci, aplikovány ještě následující skripty:

- Controls.cs - Aplikuje transformace na ovládací plochy letounu a jejich příslušné ovládací prvky v kokpitu. Skript také provádí transformaci nad podvozkem letounu. Veškeré transformace jsou prováděny na základě vstupu z klávesnice.
- InstrumentPanel.cs - Aplikuje transformace na ručičky ukazatelů na přístrojové desce na základě letových dat vypočítaných simulačním jádrem.
- PropellerRotation.cs - Aplikuje transformaci na vrtuli v podobě rotace.
- Collision.cs - Skript detekuje kolizi letounu s terénem.

²Army Bunker - <https://assetstore.unity.com/packages/3d/props/army-bunker-2093>

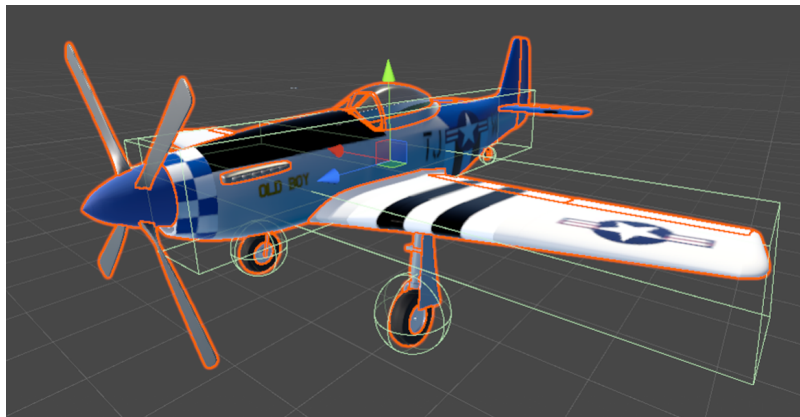
³Ultra Skybox - <https://assetstore.unity.com/packages/vfx/shaders/ultra-skybox-fog-115414>

- CamControll.cs - Skript pro ovládání kamer.

5.3.1 Kolize letounu s terénem

Aby bylo možné detekovat kolizi letounu se zemí a zabránit tak jeho průletu skrz terén, byly na hlavní uzel také aplikovány komponenty *Rigidbody*, *Collider* a skript *Collision.cs*.

V Unity, je díky komponentě typu *Rigidbody*, pohyb objektů řízen fyzikálním jádrem Unity. V tomto případě však bylo této komponentě, aplikované na hlavní uzel, zrušena gravitace a anulovány veškeré veličiny, jelikož byla aplikována jen proto, aby bylo možno detekovat kolizi s terénem.



Obrázek 5.9: Jednoduchý kolizní model vytvořený z několika colliderů

Samotný model letounu importovaný z 3DS Maxu však kolizi nedetekuje. Bylo proto potřeba vytvořit jeho kolizní model složený z několika jednoduchých komponent typu *Collider*. Jednoduchý kolizní model je zobrazený na obrázku 5.9. Pokud dojde ke kontaktu jakékoli části kolizního modelu s terénem, je detekovaná kolize, která se následně příslušným způsobem obsluží.

Zda došlo ke kolizi, kontroluje během simulace neustále skript *Collision.cs*, který jakmile je kolize detekována, pozastaví běh simulace a zobrazí nabídku spolu s informací, že došlo k havárii.

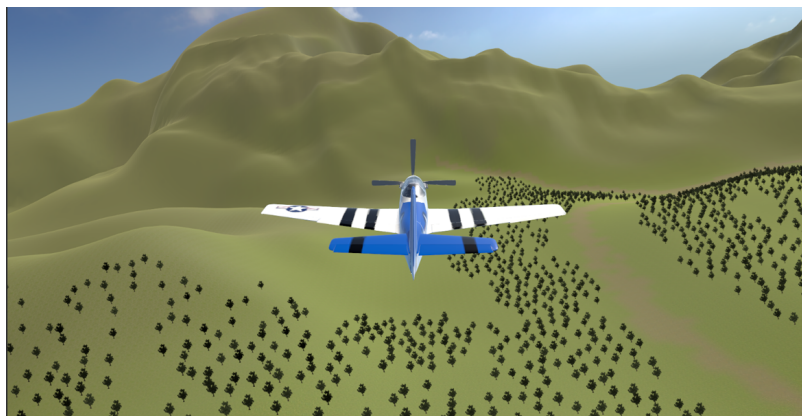
5.3.2 Kamery

Aby byly kamery během simulace vždy v relativní pozici vůči letounu, byly v hierarchii scény umístěny pod hlavní uzel letounu. Na tento uzel byl pak aplikován skript *CamControll.cs* uskutečňující rotaci a změnu kamer na základě vstupu z klávesnice a myši.

Při spuštění simulace nejdříve skript zkontroluje, zda je zapojeno zařízení virtuální reality. Pokud ano, je aktivován uzel obsahující kameru podporující virtuální realitu a dva ovládače. Ostatní dvě kamery jsou deaktivovány. Během simulace proto nelze přepínat mezi pohledy. Uživatel má k dispozici pouze pohled z kokpitu. Je rovněž deaktivovaný textový výpis simulačních dat, jelikož v kokpitu jsou k dispozici ukazatele zobrazující jednotlivá data.

V případě, kdy však zařízení virtuální reality zapojeno není, jsou využívány zbývající dvě kamery. Mezi kamerami lze v průběhu simulace přepínat pomocí klávesy V. Při spuštění simulace je v základu aktivovaná kamera umístěna za letounem. Její pozice vůči letadlu je

zobrazena na obrázku 5.10. Spolu s ní je aktivován také textový výpis simulačních dat. V případě že uživatel stiskem klávesy přepne pohled do kokpitu, je textový výpis deaktivován.

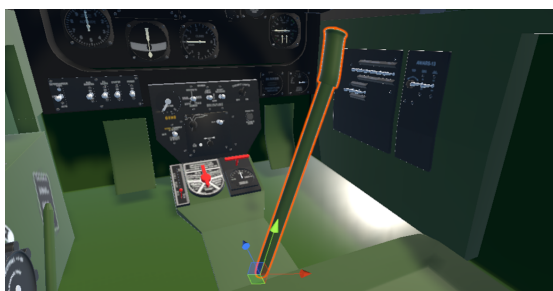


Obrázek 5.10: Pozice vnější kamery vůči letounu

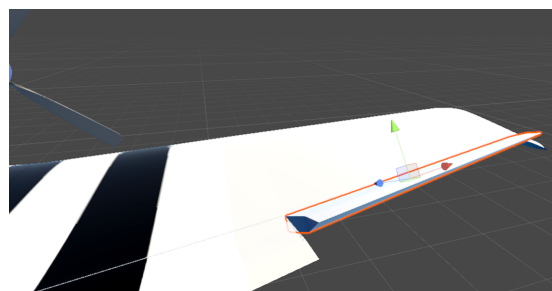
V průběhu letu může uživatel rotovat dvěma klasickými kamerami, a díky tomu se tak rozhlížet. V případě vnější kamery, je střed rotace nastaven v centru letounu. Při rozhlížení je díky tomu kamera rotována kolem celého letounu. Střed rotace vnitřní kamery je pak shodný s pozicí kamery samotné. Při rozhlížení je tak imitována rotace hlavy pilota. Rotaci kamer uživatel vyvolá stiskem klávesy C. Jakmile je klávesa stisknuta, dochází k načítání vstupu z myši v podobě změny pozice podél vertikální a horizontální osy. Na základě získané hodnoty násobenou rychlostí, s jakou je kamera rotována, je prováděná samotná rotace kamery. Jakmile je klávesa uvolněna, kamera se vrátí do původní pozice.

5.3.3 Transformace ovládacích prvků

Transformace ovládacích prvků letounu, kde jejich modely jsou v hierarchii scény umístěny pod uzly, které byly vytvořeny nad každým z ovládacích prvků, obsluhuje skript Controls.cs. Samotný skript si mimo objektů, na které jsou transformace aplikovány, uchovává také jejich maximální hodnotu rotace ve stupních, dále krok rotace, rovněž ve stupních, a také aktuální pozici v normalizovaném tvaru, tedy hodnotu v intervalu od -1 do 1. Tato normalizovaná hodnota poté složí k výpočtu aerodynamických koeficientů.



Obrázek 5.11: Vizualizace posunu řídicí páky



Obrázek 5.12: Vizualizace rotace pravého křídélka

V průběhu simulace skript kontroluje, zda byla stisknutá klávesa ovládající jeden z množiny ovládacích prvků. Pokud ano, dojde ke kontrole, zda již daný ovládací prvek nedosáhl

maximálního stupně rotace. V případě že ne, je danému objektu nastaven příslušný úhel rotace a také se aktualizuje hodnota aktuální pozice prvku v normalizovaném tvaru. Příklad vizualizace rotace některých z ovládacích prvků je možné vidět na obrázcích 5.11 a 5.12. Pokud stiskne uživatel klávesu pro ovládání klapky či podvozku, mimo rotace je ještě provedena aktualizace textového výstupu informující o jejich pozici. V případě podvozku se jen vypíše, zda je podvozek otevřen či zavřen. U klapky to je informace o úhlu vyklopení.

Všechny rotace jsou uskutečněny relativně vůči nadřazenému uzlu, kterým je samotný letounu, a nikoliv vůči středu globálního souřadného systému.

5.3.4 Transformace ukazatelů leteckých dat

Logika pohybu ručiček v závislosti na naměřených veličinách, je implementována ve skriptu InstrumentPanel.cs. Transformace jsou, stejně jako u ovládacích prvků, aplikovány na nadřazené uzly jednotlivých ručiček. Veškeré rotace jsou prováděny kolem osy Z. Ve skriptu je k většině ukazatelům vytvořeno několik konstant a proměnných uchovávajících následující hodnoty:

- Maximální/Minimální hodnota veličiny - Konstanta uchovává maximální hodnotu, kterou je možné na ukazateli příslušné veličiny zobrazit. V případě, že minimální hodnota na přístroji není nula, je vytvořena také proměnná uchováující minimální hodnotu.
- Aktuální naměřená hodnota - Do proměnné je ukládána aktuální naměřená hodnota dané veličiny
- Maximální/Minimální úhel rotace - Konstanty určují hranice rotace ručičky ve stupních, tedy maximální a minimální úhel vůči výchozí pozici.
- Cílový úhel rotace - Hodnota určuje, jaký úhel rotace má být nastaven ručičce ukazatele vůči její základní pozici.



Obrázek 5.13: Kompas s pamatováčkem

Výpočet cílového úhlu rotace probíhá během každého snímku. Hodnota cílového úhlu je vypočtena pomocí interpolace, kdy vstupem interpolační funkce je aktuální naměřená

hodnota, minimální/maximální hodnota na ukazateli a minimální/maximální úhel rotace ručičky. Díky interpolaci je vyřešen problém vizualizace, kdy stupnice ukazatele je rozdělena do několika intervalů, ve kterých se ručička otáčí s jinou rychlostí. Na vstup interpolační funkce se tak vždy nastaví na základě aktuální naměřené hodnoty příslušné minimální/maximální hodnoty veličin a stupňů. Příkladem přístroje s rozdělenou stupnicí je rychloměr, jehož stupnici, rozdělenou do čtyř částí je možné vidět na obrázku 4.12.

Před výpočtem cílových úhlů rotace některých ručiček, je ještě potřeba provést konverzi z metrické soustavy SI do soustavy imperiálních jednotek. Tento případ se opět týká i rychloměru, kdy výpočet veličiny v simulačním jádru probíhá v metrech za sekundu, zatímco přístroj měří rychlost v mílích za hodinu.

Při transformaci některých ukazatelů však interpolace využita není. Jsou to kompas, umělý horizont, indikátor vybočení a hodiny. V případě hodin, je rotace určena na základě aktuálního času. U ostatních ukazatelů je rotace ukazatele provedena na základě aktuální hodnoty úhlu bočení nebo klonění letounu.

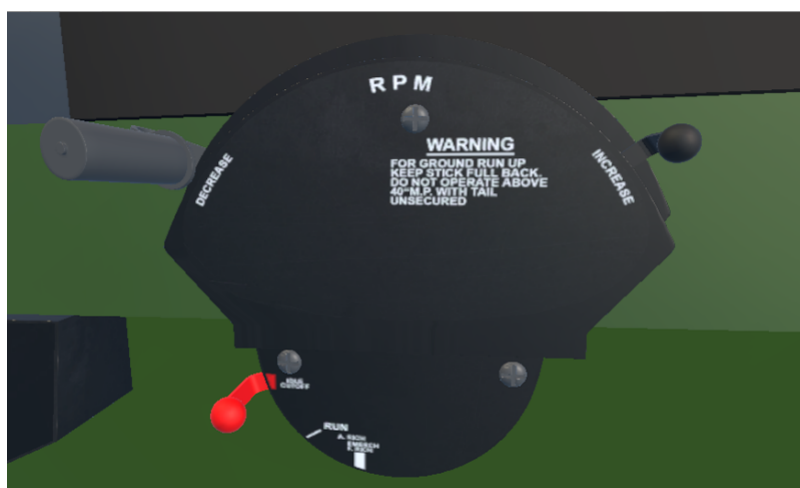
Na přístrojové desce letounu se nachází kompas, který má krom ukazatele skutečného směru letu také tzv. pamatováček. Jedná se o ručičku, pomocí které si pilot nastaví směr, jakým chce letět. Pamatováček je možné si nastavit i v tomto simulátoru, kdy uživatel pomocí kláves P a L provádí rotaci doprava či doleva.

Na obrázku 5.13 je vidět, že zatímco letoun letí na sever, uživatel si nastavil pamatováček na severo-východ.

5.3.5 Ovládání otáček vrtule

Posledním ze skriptů, aplikovaný na hlavní uzel, je PropellerRotations.cs. Jedná se o skript, který v průběhu simulace čeká na vstup uživatele. V případě, že uživatel stiskne příslušnou klávesu, je aktualizovaná hodnota uchováající počet otáček za minutu (RPM) na motoru. Hodnotu poté využívá řada tříd pro výpočet potřebných veličin. Otáčky je však nutno kontrolovat a udržovat v určitém intervalu. Ten je stanoven od 1800 do 3000 RPM [27]. Počáteční hodnota otáček je nastavena na 2000 RPM.

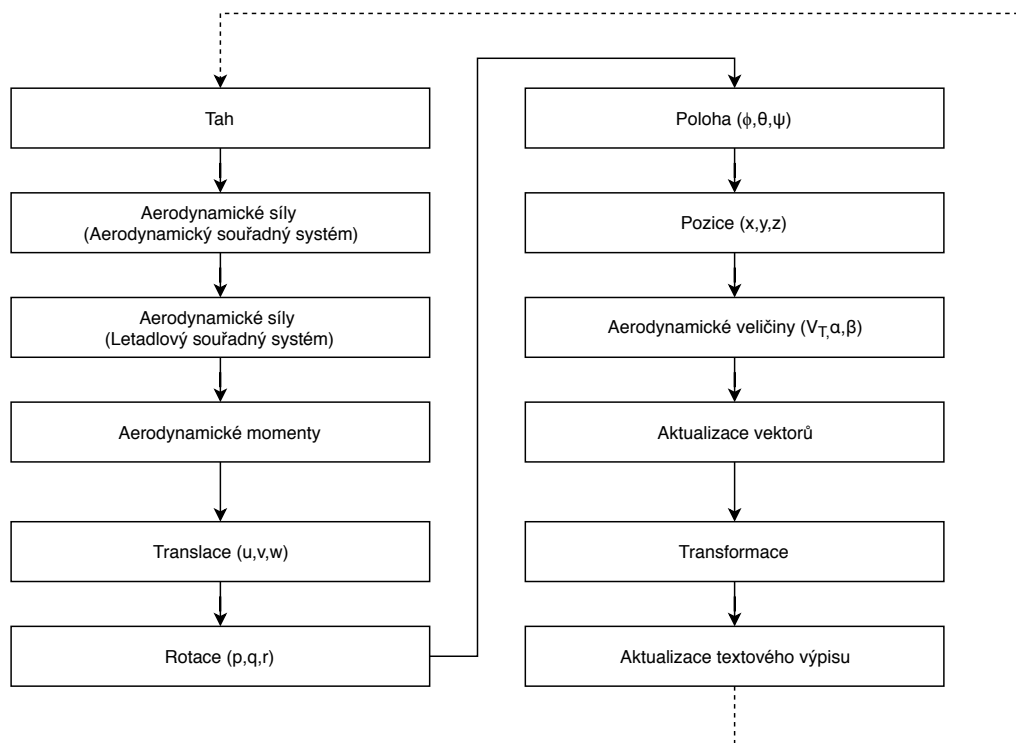
Skript dále také řídí rotaci plynové páky (obr. 5.14) uvnitř kokpitu. Úhel rotace páky je opět počítán pomocí interpolace.



Obrázek 5.14: Plynová páka

5.4 Simulační jádro

Centrum simulačního jádra, a tedy i celého simulátoru, tvoří třída *Simulator* ve skriptu *Simulator.cs*. Třída implementující hlavní iteraci výpočtu, znázorněná na obrázku 5.15, je odvozená od základní třídy *MonoBehavior*, od které je každý skript v Unity odvozen.



Obrázek 5.15: Iterace výpočtu

Před spuštěním samotné simulace nejdříve metoda *Start()* provede inicializaci simulace. Ta, v případě že je třída odvozená právě od základní třídy *MonoBehavior*, je vyvolána před prvním snímkem simulace. Jako první jsou vytvořeny nové instance potřebných tříd. Významy jednotlivých tříd bude popsán dále. Následně se nastaví proměnné uchováující aktuální hodnoty aerodynamických sil a momentů na výchozí hodnoty. Následně jsou vytvořeny vektory, které během simulace uchovávají jednotlivé složky rychlosti letounu, úhlové rychlosti, pozice a polohy letounu. Hodnoty, kterými jsou jednotlivé složky vektorů inicializovány, jsou zároveň počátečními podmínkami pro řešení pohybových rovnic.

První z tříd, která je v metodě *Start()* inicializována, je třída *Tables*. Při vytvoření její nové instance jsou inicializovány tabulky složené ze slovníků a listů. Tabulky slouží k uchování aerodynamických a výkonových koeficientů. Následně je vyvolána metoda, ve které se načte XML soubor s tabulkami koeficientů. Soubor je rozparsován a získaná data jsou uložena do příslušných tabulek, ke kterým se během celé simulace přistupuje. Jednotlivé koeficienty jsou z tabulek získávány pomocí lineární či bilineární interpolace, kdy typ interpolace závisí na druhu tabulky, tedy zda je koeficient funkcí jedné či dvou proměnných.

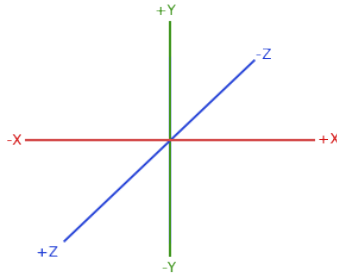
Po dokončení inicializace pomocí metody *Start()*, je během celé simulace v pevně daných intervalech vyvolávána metoda *FixedUpdate()*, ve které probíhá celá iterace výpočtu.

V první části iterace je vypočtena tahová síla. Ta je výstupem metody *Thrust()*. Tah však závisí na výkonu produkovaného motorem letounu. Výkon počítá metoda *Power()*. Obě

metody jsou implementované ve třídě *PropEngine* modelující pohon letounu. Při výpočtu tahu je využívána třída *Atmosphere* poskytující hodnotu hustoty vzduchu a *Propeller*, která mimo to, že animuje rotaci vrtule, poskytuje aktuální počet otáček vrtule. Celý princip výpočtu tahu je shodný s postupem popsáným v kapitole 4.

V další části iterace proběhne výpočet aerodynamických sil a momentů, kdy každá síla a moment jsou reprezentovány svou vlastní třídou. Výsledné síly a momenty jsou výsledkem součtu dílčích koeficientů. Stejně jako při výpočtu tahové síly, tak i zde je využívána třída *Atmosphere* k výpočtu veličin nutných nejen k získání konkrétních koeficientů z tabulek, ale také k výpočtu celkové síly či momentu. Jelikož jsou pohybové rovnice odvozeny v letadlovém souřadném systému, a výpočet jednotlivých sil probíhá v aerodynamickém souřadném systému, je nutno výsledné síly a momenty přetransformovat do letadlového systému.

V okamžiku, kdy jsou vypočteny aerodynamické síly a momenty, jsou k dispozici všechny potřebné veličiny s aktuálními hodnotami, které figurují ve výpočtu pohybových rovnic. Jednotlivé pohybové rovnice jsou implementovány jako samostatné metody ve třídě *Simulator*. Výsledky těchto metod jsou násobeny intervalem mezi jednotlivými snímky, kdy velikost intervalu je rovný 0.003 sekundy. K součinu je následně přičten aktuální stav počítané složky z konkrétního vektoru. Výsledek je následně dočasně uložen do určené proměnné.



Obrázek 5.16: Souřadný systém v Unity [15]

Jakmile jsou přepočteny nové stavy složek všech čtyř vektorů, určí se na základě složek u, v, w vektoru \vec{V} nová pravá vzdušná rychlost a úhly náběhu a skluzu. Následuje aktualizace hodnot vektorů.

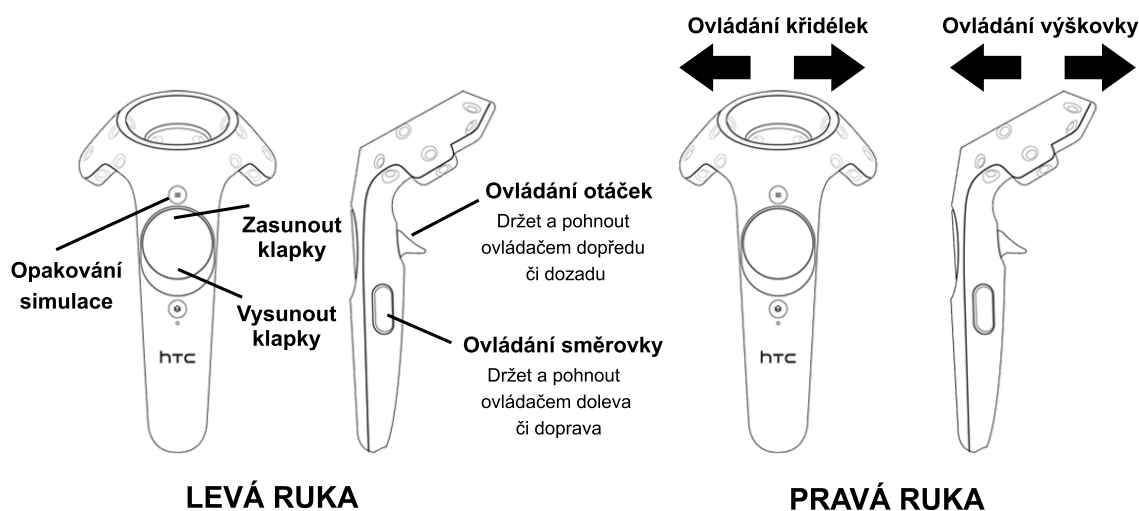
Závěrečnou fází iterace představuje aktualizace textového výstupu simulačních dat a také transformace 3D modelu letounu ve scéně na základě nově vypočtených stavů. V Unity je však orientace os souřadného systému jiná než ta v letadlovém souřadném systému. Souřadný systém v Unity je možné vidět na obrázku 5.16. Platí, že osa X (Unity) se shoduje s osou Y (BFF), osa Y (Unity) s osou -Z (BFF) a osa Z (Unity) s osou X (BFF). Je tak nutné provést přiřazení hodnot pozičnímu vektoru, definující pozici letounu v souřadném systému Unity, ve správném pořadí. Rotace letounu ve scéně je realizována pomocí Eulerových úhlů, kdy hodnoty vypočtených úhlů, je rovněž potřeba přiřadit jednotlivým složkám vektoru rotace ve správném pořadí. A to tak, že úhlu Φ (Unity) je přiřazen úhel $-\Theta$ (BFF), úhlu Θ (Unity) se přiřadí úhel Ψ (BFF) a konečně úhlu Ψ (Unity) je přiřazen úhel $-\Phi$ (BFF). Provedením transformací letounu je ukončen jeden cyklus simulační smyčky.

5.5 Podpora virtuální reality

Simulátor poskytuje podporu virtuální reality od HTC Vive. Implementace podpory byla umožněna díky modulu SteamVR⁴, který byl stažen z obchodu Unity. Je nutné mít také nainstalovanou službu Steam a aplikaci SteamVR. Balíček SteamVR poskytuje přednastavený objekt obsahující kameru a dva ovládače. Tento objekt byl umístěn do scény pod hlavní uzel letounu.

Co se týče kamery, bylo nutné pouze nastavit její optimální pozici v kokpitu. Veškerá logika pohybu kamery v závislosti na pohybu hlavy s brýlemi byla již implementována modulem SteamVR. V případě ovládačů muselo být ovládání nejdříve navrženo, které se následně implementovalo.

5.5.1 Návrh ovládání



Obrázek 5.17: Navržené ovládání pomocí HTC Vive

Ovládání je navrženo tak, aby imitovalo reálné řízení letounu, tedy v pravé ruce je držena řídicí páka, zatímco zbytek ovládacích prvků je ovládáno pomocí levé ruky.

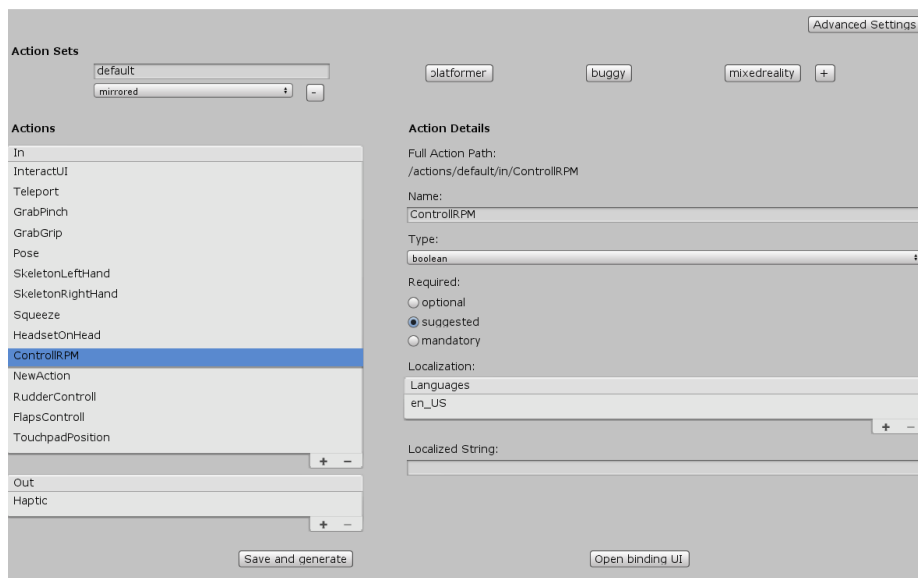
Levou rukou tedy lze ovládat počet otáček, kdy uživatel musí nejdříve stisknout "spoušť", aby bylo zahájení snímání polohy ovládače. Poté pohybem ovládače dopředu či dozadu uživatel ovládá otáčky. Pomocí levého ovládače je také možné vysunout či zasunout klapky a restartovat simulaci. Jelikož nebyly k dispozici žádné nožní ovládače, je ovládání nožního řízení realizováno pomocí levé ruky, kdy je nejdříve nutno stisknout a držet příslušné tlačítko. Poté lze pohybem ovládače doleva či doprava otáčet směrovým kormidlem.

V pravé ruce je ovládání zřejmé, a sice pohybem ovládače dopředu či dozadu lze ovládat výškové kormidlo, zatímco pohybem ovládače doprava či doleva se ovládají křidélka. Navržené ovládání pomocí ovládačů od HTC Vive je zobrazeno na obrázku 5.17.

⁴SteamVR plugin - <https://assetstore.unity.com/packages/tools/integration/steamvr-plugin-32647>

5.5.2 Implementace ovládání

Navržené ovládání bylo následně nutno implementovat. Modul SteamVR, poskytující podporu virtuální reality, využívá svého systému pro správu uživatelského vstupu, kdy se vytvoří vlastní schéma akcí, kterým se nastaví daný typ vstupu. Rozhraní pomocí kterého jsou tyto akce vytvářeny je možné vidět na obrázku 5.18. Lze vidět, že například v případě ovládání otáček, kdy je nutno nejdříve stisknout konkrétní tlačítko, se vytvořila akce *ControlLRPM* typu boolean, kdy se pouze zjišťuje, zda bylo tlačítko stisknuto či nikoliv. Akce byla vytvořena v seznamu *default*. Lze si však vytvořit i vlastní seznam akcí.



Obrázek 5.18: Grafické rozhraní na vytváření akcí pro tlačítka VR ovládačů

Jakmile byly vytvořeny všechny potřebné akce, byl vygenerován JSON soubor uchovávající tyto akce a data s nimi spojené. Vytvořené akce se následně pomocí uživatelského rozhraní SteamVR přiřadily ke konkrétním tlačítkům ovládače.

Následně se ve skriptu *Controls.cs*, obsluhující ovládání letounu, vytvořily objekty konkrétních typů. Typ objektu se shodoval s typem akce. Z těchto objektů jsou poté získávány vstupní hodnoty. V případě tlačítek to je pouze stav, zda bylo či nebylo stisknuto. Pokud ano, je provedena odpovídající událost. V případě výškového kormidla a křidélek jsou po celou dobu z příslušných objektů získávány souřadnice, díky kterým je možné zjistit, zda došlo ke změně v pozici ovládače podél os X a Z. Na základě těchto změn jsou odpovídajícím způsobem transformovány ovládací prvky.

Kapitola 6

Zhodnocení dosažených výsledků a možná vylepšení

Dosažené výsledky byly zhodnoceny na základě testování, které bylo rozděleno do dvou částí. V první části probíhalo testování praktické, díky kterému byly postupně odstraňovány hlavní implementační chyby. Druhá část se pak skládala z uživatelského testování několika respondentů, kdy na základě jejich hodnocení bylo možné zhodnotit výsledný simulátor a případně určit i další směr vývoje.

6.1 Praktické testování

Praktické testování probíhalo v průběhu celého vývoje simulátoru, kdy cílem tohoto testování bylo odhalit chyby v jeho implementaci.

Testování zahrnovalo kontrolu správného chování letounu při vychýlení řídicích ploch. Bylo kontrolováno, zda při vychýlení konkrétní řídicí plochy do určité pozice, vznikne správně orientovaný moment. Pokud ano, došlo ke správné rotaci letounu kolem dané osy.

Dalším, velmi důležitým předmětem testování, byla kontrola správného fungování ukazatelů. Tedy zda ručičky rotují ve správném směru a ukazují hodnoty takové, jaké byly simulátorem vypočítány. S tím souvisela i kontrola správné konverze jednotek u některých přístrojů.

6.2 Uživatelské testování

Uživatelské testování mělo za cíl odhalit klady a zápory výsledného simulátoru a případně zjistit možný směr budoucího vývoje.

Testováno bylo devět respondentů, mezi kterými byli jak zkušení uživatelé leteckých simulátorů, tak i lidé, kteří s leteckými simulátory mají zkušenost nulovou. Nejprve bylo respondentům představeno ovládání a poté jim bylo sděleno, na co se při testování zaměřit. Následně se přešlo k samotnému testování, které probíhalo ve dvou fázích. V první fázi byli vyzváni, aby simulátor otestovali bez VR technologie, kdy simulátor ovládali pomocí klávesnice a myši. Druhá fáze pak zahrnovalo využití VR brýlí a ovládačů pro ovládání letounu. Během testování probíhala volná konverzace ohledně jejich pocitů z ovládání a celkového zpracování. Na konci testování byl testerům předložen dotazník, který vyplnili.

Dotazník byl zaměřen na testování tří vlastností simulátoru. Jsou jimi grafické zpracování, funkčnost a zapojení virtuální reality. Ke každé z vlastností bylo v dotazníku vytvořeno

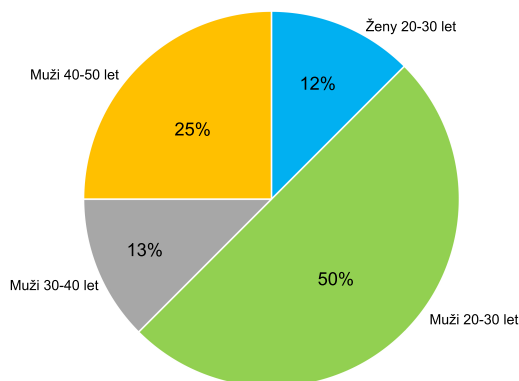
několik otázek. U většiny otázek bylo možné vybrat jednu z pěti odpovědí. Každé z odpovědí příslušela známka na stupnici od 1 do 5, kdy nejlepší známkou byla 1.

Před samotným testováním byly stanoveny tři hypotézy, kdy uživatelské testování mělo tyto hypotézy potvrdit či vyvrátit:

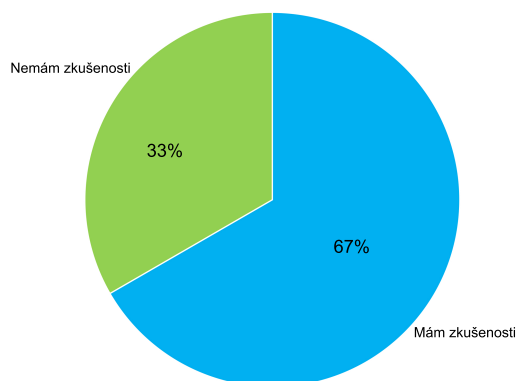
- Hypotéza č. 1 - Grafické zpracování simulátoru je adekvátní a vhodné k využití.
- Hypotéza č. 2 - Funkční prvky simulátoru vytváří očekávanou odezvu a letový model má adekvátní reakce.
- Hypotéza č. 3 - S využitím virtuální reality se orientace v kokpitu a preciznost pilotáže letounu zlepšila.

6.3 Vyhodnocení

Jak již bylo zmíněno, uživatelského testování se zúčastnilo devět osob. Věkové a genderové zastoupení je znázorněno grafem na obrázku 6.1. Graf na obrázku 6.2 poté znázorňuje procentuální rozdělení lidí se zkušenostmi ze simulátoru.



Obrázek 6.1: Genderové a věkové rozložení testovací skupiny



Obrázek 6.2: Zkušenosti se simulátory v testovací skupině

Grafické zpracování simulátoru

- Celkové grafické zpracování - 1.75
- Úroveň grafického zpracování kokpitu - 1.75
- Grafické zpracování prostředí simulace - 3.0
- Pozice a orientace kamer ve scéně - 1.75
- Grafické zpracování uživatelského rozhraní - 1.875

Celkově je tedy, podle testujících, simulátor po grafické stránce zpracován více než kvalitně. Testující nejvíce zaujalo zpracování modelu letounu a hlavně detaily kokpitu. Pozice kamer všem respondentům vyhovovaly. Nejnižší známku u grafického zpracování pak dostalo zpracování prostředí simulace, ve kterém se simulace odehrává. S tímto hodnocením lze souhlasit, jelikož prostředí je relativně jednoduché bez výraznějších detailů. Pro účely tohoto simulátoru však dostačují. Hypotéza č. 1 tedy byla potvrzena.

Funkcionalita simulátoru

- Odezva ovládání simulátoru pomocí klávesnice - 1.875
- Adekvátní chování letového modelu - 1.85
- Čitelnost letových dat na palubních ukazatelích - 2.125

U funkcionality simulátoru nejvíce záleželo na hodnocení letového modelu. Ten nakonec dostal celkovou známku 1.85. Podle respondentů je tedy chování letounu více než kvalitní. Celkově se tak letoun v průběhu simulace chová podle očekávání, bez náznaků výrazných anomálií. Co se týče ostatních vlastností, čitelnost letových dat z palubních ukazatelů na přístrojové desce byla pro uživatele relativně snadná. Ovladatelnost letounu pomocí klávesnice rovněž nepůsobila testerům výraznější problémy. Hypotéza č. 2 tak byla potvrzena.

Virtuální realita

- Orientace uživatele v prostředí bez VR brýlí - 2.625
- Orientace uživatele v prostředí s VR brýlemi - 1.375
- Navržené ovládání pomocí VR ovladačů - 1.5
- Subjektivní dojem ze simulátoru s využitím VR - 1.625
- Čitelnost letových dat na ukazatelích pomocí VR - 3.0

Podle výsledků je patrné, že s využitím VR technologie se zvýšila orientace v kokpitu. Testerům se dále výrazně více zamlouvalo řízení pomocí VR ovladačů. To jim, na rozdíl od klávesnice, připadalo lehčí a intuitivnější. Co se však zhoršilo, byla čitelnost dat z ukazatelů letových veličin. Důvodem však byla kvalita zobrazovačů VR brýlí od HTC Vive. Při jejich využívání byly přístroje rozmazané a nešly z větší vzdálenosti dobře přečíst. Celkově však VR technologie zvýšila kvalitu simulace a hypotéza č. 3 byla potvrzena.

6.4 Možnosti budoucího vývoje

Výsledný simulátor dosahuje požadovaných výsledků, kdy při simulaci dynamiky letounu nebyly nalezeny chyby, které by výrazně vybočovaly od reálného chování. Grafická stránka rovněž sklídila úspěch. Nicméně najdou se možnosti budoucího vývoje spojené hlavně s estetickými vlastnostmi simulátoru. Je možné zmínit například zvukové efekty či grafické zpracování scény simulace. Mezi grafickými vylepšeními scény můžeme zmínit implementaci reálných mraků do scény, vytvoření větší množství a variability vegetace či vložení městské zástavby včetně letiště. Dalším možným vylepšením je přidání reakcí se zemí, díky kterým je možné s letounem přistát a pohybovat se po zemi.

Kapitola 7

Závěr

Cílem této práce bylo vytvořit jednoduchý letecký simulátor s podporou virtuální reality, který by implementoval vlastní simulaci dynamiky letounu, a to bez použití jakéhokoliv, již vytvořeného, fyzikálního jádra.

Aby bylo dosaženo požadovaného výsledku, muselo být navrženo a implementováno jádro simulace, které by autenticky simulovalo dynamiku letounu. K tomu bylo nutné nastudovat, jakým způsobem je popsán pohyb letounu v prostoru a na čem tento pohyb závisí. Dále se seznámit s principy, jakým je běžný letoun ovládán. V neposlední řadě bylo nezbytné nastudovat problematiku modelování pohonu letounu.

Další část vývoje simulátoru tvořila tvorba scény simulace. Tato tvorba zahrnovala profilování terénu, aplikování textur na terén, vkládání objektů do scény a imitace reálné oblohy.

Podstatnou část práce pak představovala rekonstrukce vybraného letounu, na kterém je simulace provedena. Bylo tak nezbytné vytvořit odpovídající 3D model letounu a kokpitu, kdy byl kladen důraz zejména na vytvoření reálných ukazatelů letových veličin. Důležitá část tvůrčího procesu byla realizace co možná nejpřesnějšího letového modelu.

Neméně důležitou částí vývoje simulátoru, bylo navržení a vytvoření přívětivého uživatelského prostředí. Posledním krokem ve vývoji pak představovala implementace virtuální reality, zahrnující návrh a realizaci ovládání letounu pomocí dvou ovládacích zařízení od HTC Vive.

Výsledný simulátor byl nakonec podroben uživatelskému testování, které prokázalo, že se jedná o simulátor s kvalitním grafickým zpracováním a letovým modelem blížícím se reálnému chování letounu. Díky virtuální realitě poté byla hodnota simulátoru zvýšena o větší realističnost.

Literatura

- [1] ABBOT, F. T.; UNDERWOOD, W. J.: *Wind Tunnel Investigation of Profile Drag and Lift of an Intermediate Wing Section of the XP-51 Airplane with Beveled Trailing-Edge and Contour Ailerons*. 1943, [Online; navštíveno 8.2.2019].
URL https://crgis.ndc.nasa.gov/crgis/images/9/94/Wind-Tunnel_Investigation_of_Profile_Drag_and_Lift..._Abbot%2C_Underwood%2C_1943.pdf
- [2] Autodesk: *3DS Max*. 1996, [Online; navštíveno 28.4.2019].
URL <https://www.autodesk.com/products/3ds-max/features>
- [3] Boeing.com: *Inovative 787 Flight Deck Design for Efficiency, Comfort, and Commonality*. [Online; navštíveno 27.1.2019].
URL https://www.boeing.com/commercial/aeromagazine/articles/2012_q1/3/
- [4] CAI, G.; CHEN, B.M.; LEE, T.H.: *Unmanned Rotorcraft Systems*. Springer, 2011, ISBN 978-0-85729-634-4.
- [5] Concorde Collector: *Cockpit Instruments from the Concorde Development Fleet*. 2009, [Online; navštíveno 27.1.2019].
URL <http://developmentfleetinstruments-en463.blogspot.com>
- [6] COOK, M. V.: *Flight Dynamics Principles, Second Edition*. Elsevier, 2007, ISBN 978-0-7506-9627-6.
- [7] Digital Combat Simulator: *DCS P-51 Mustang - Flight Manual*. 2012, [Online; navštíveno 10.2.2019].
URL <http://cdn.akamai.steamstatic.com/steam/apps/223770/manuals/DCS%20P-51D%20Flight%20Manual%20EN.pdf>
- [8] Federal Aviation Administration, Aviation Supplies & Academics : *Pilot's Handbook of Aeronautical Knowledge*. Aviation Supplies & Academics Inc, 2016, ISBN 9781619544734.
- [9] Flyingheritage.org: *North American P-51D Mustang*. [Online; navštíveno 28.4.2019].
URL <https://flyingheritage.org/Explore/The-Collection/United-States/North-American-P-51D-Mustang.aspx>
- [10] Hoggit DCS World Wiki: *Basic Airmanship*. 2016, [Online; navštíveno 27.1.2019].
URL https://wiki.hoggitworld.com/view/Basic_Airmanship
- [11] HOUSTON, S.: *The Pitot-Static System Powers Aircraft Instruments*. 2019, [Online; navštíveno 27.1.2019].

- URL <https://www.thebalancecareers.com/aircraft-systems-pitot-static-system-282605>
- [12] HTC, VALVE C.: *Vive Virtual Reality System*. 2016, [Online; navštíveno 30.4.2019].
URL <https://www.vive.com/us/product/vive-virtual-reality-system/>
- [13] HURST, R.: *RAF Fixed and Free-mounted Reflector Gunsights*. 2002, [Online; navštíveno 27.1.2019].
URL <https://forum.axishistory.com/viewtopic.php?p=160621&sid=51c60b7d43f1923c2bb9a94c598a6ff1>
- [14] KIM, J. W.: *Rupture of the Virtual*. Macalester College, 2016, ISBN 978-0-692-75828-1.
- [15] Learnopengl.com: *Right handed coordinate system*. [Online; navštíveno 28.4.2019].
URL https://learnopengl.com/img/getting-started/coordinate_systems_right_handed.png
- [16] MULDER, J. A.; VAN STAVEREN, W. H. J.; VAN DER VAART, J. C.: *Flight Dynamics (lecture notes)*. 2000, Technical report, Delft University of Technology.
- [17] NASA: *P-51 "Mustang" Archives Collection*.
URL https://crgis.ndc.nasa.gov/historic/P-51_
- [18] PERINGER, P.: *Modelování a simulace - Studijní opora*. 2012.
- [19] Public Domain Aeronautical Software: *Digital Datcom*. 2017, [Online; navštíveno 26.4.2019].
URL <http://www.pdas.com/datcom.html>
- [20] Skybrary.aero: *Primary Flight Display (PFD)*. 2017, [Online; navštíveno 27.1.2019].
URL [https://www.skybrary.aero/index.php/Primary_Flight_Display_\(PFD\)](https://www.skybrary.aero/index.php/Primary_Flight_Display_(PFD))
- [21] Smithsonian National Air and Space Museum: *Consolidated NY-2 (Husky), Doolittle Blind Flight Aircraft, Cockpit*. [Online; navštíveno 27.1.2019].
URL <https://airandspace.si.edu/collection-objects/consolidated-ny-2-husky-doolittle-blind-flight-aircraft-cockpit-photograph>
- [22] SONNEVELDT, L.: *Nonlinear F-16 Model Description*. 2006. Delft University of Technology, Faculty of Aerospace Engineering.
- [23] STEVENS, B. L.; LEWIS, F. L.: *Aircraft Control and Simulation, 2nd Edition*. Wiley, 2003, ISBN 0-471-37145-9.
- [24] Technologies, Unity: *Unity Editor*. 2005, [Online; navštíveno 28.4.2019].
URL <https://unity3d.com/>
- [25] The Aviation History Online Museum: *P-51D Mustang*. [Online; navštíveno 10.2.2019].
URL <http://www.aviation-history.com/north-american/p51.html>
- [26] TUPULARKA, E.G.: Flight dynamics - II, Stability and control, Chapter 7, Lecture 23 [přednáška]. In: *Nptel.ac.in*. [Online; navštíveno 20.1.2018].
URL https://nptel.ac.in/courses/101106043/downloads/033_Chapter%207_L23.pdf

- [27] unix_nerd: *Forum thread: P-51D propeller technical data request*. 2007, [Online; navštíveno 11.2.2019].
URL <https://ww2aircraft.net/forum/threads/p-51d-propeller-technical-data-request.24166/>
- [28] Virtual Reality Society: *What is Virtual Reality?* 2017, [Online; navštíveno 30.4.2019].
URL <https://www.vrs.org.uk/virtual-reality/what-is-virtual-reality.html>
- [29] Wikipedia.org: *Primary Flight Display*. 2018, [Online; navštíveno 27.1.2019].
URL https://en.wikipedia.org/wiki/Primary_flight_display
- [30] WWII Aircraft Performance: *P-51 Mustang Performance*. [Online; navštíveno 10.2.2019].
URL <http://www.wwiiaircraftperformance.org/mustang/mustangtest.html>
- [31] WWII Aircraft Performance: *Pursuit Single Engine P-51B-1-NA, AAF No. 43-12093*. [Online; navštíveno 10.2.2019].
URL <http://www.wwiiaircraftperformance.org/mustang/V-1650-3-low-blower-curve.jpg>
- [32] ZYSKOWSKI, M. K.: Modeling and Simulation Technologies Conference and Exhibit. In *Aircraft Simulation Techniques Used In Low-Cost, Commercial Software*, Austin, Texas, 2003.

Příloha A

Doprovodný dotazník

Doprovodný dotazník

Před Vámi leží dotazník k Bakalářské práci zabývající se leteckou simulací. Cílem práce bylo navrhnout a vytvořit letecký simulátor, který by podporoval virtuální realitu a zvýšil by tak autenticitu celé simulace. Tento dotazník slouží k získání zpětné vazby na základě které, jsou odhaleny kladné a záporné vlastnosti simulátoru.

O Vás:

1. **Váš věk:** 20-30 let 30-40 let 40-50 let více
2. **Vaše pohlaví:** Muž Žena
3. **Máte zkušenosti s nějakým leteckým simulátorem?:** Ano Ne
4. **Vyzkoušel/a jste někdy virtuální realitu v leteckém simulátoru?:** Ano Ne

Grafické zpracování:

5. Celkové grafické zpracování simulátoru:

- Skvělé Kvalitní Dobré Dostatečné Špatné

6. Úroveň grafického zpracování kokpitu:

- Skvělá Kvalitní Dobrá Dostatečná Špatná

7. Grafické zpracování prostředí simulace:

- Skvělé Kvalitní Dobré Dostatečné Špatné

8. Pozice a orientace kamer ve scéně:

- Skvělé Kvalitní Dobré Dostatečné Špatné

9. Grafické zpracování uživatelského rozhraní:

- Skvělé Kvalitní Dobré Dostatečné Špatné

Funkcionalita simulátoru:

10. Odezva ovládání simulátoru pomocí klávesnice:

- Skvělé Kvalitní Dobré Dostatečné Špatné

11. Chování letového modelu:

- Reálné Kvalitní Dobré Dostatečné Špatné

12. Čitelnost letových dat na palubních ukazatelích:

- Skvělá Kvalitní Dobrá Dostatečná Špatná

Virtuální realita:

13. Orientace v prostředí bez VR brýlí:

Skvělá Kvalitní Dobrá Dostatečná Špatná

14. Orientace v prostředí pomocí VR brýlí:

Skvělá Kvalitní Dobrá Dostatečná Špatná

15. Je ovládání simulátoru pomocí VR technologie snadnější?:

Ano Ne

16. Navržené ovládání pomocí VR ovládačů:

Skvělý Kvalitní Dobrý Dostatečný Špatný

17. Subjektivní dojem ze simulátoru s využitím VR technologie:

Skvělý Kvalitní Dobrý Dostatečný Špatný

18. Čitelnost letových dat na ukazatelích s využitím VR technologie:

Skvělá Kvalitní Dobrá Dostatečná Špatná

19. Co Vás nejvíce zaujalo:

20. Co se Vám nelíbilo:

21. Co byste vylepšili či přidali:
