

Bakalárska práca



**České
vysoké
učení technické
v Praze**

F3

**Fakulta elektrotechnická
Katedra řídicí techniky**

System riadenia letu malého vírniku

Dominik Beňo

Vedúci: Doc. Ing. Martin Hromčík, Ph.D.

Odbor: Kybernetika a robotika

Študijný odbor: Kybernetika a robotika

Máj 2022

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Beňo** Jméno: **Dominik** Osobní číslo: **492051**
Fakulta/ústav: **Fakulta elektrotechnická**
Zadávající katedra/ústav: **Katedra řídicí techniky**
Studijní program: **Kybernetika a robotika**

II. ÚDAJE K BAKALÁŘSKÉ PRÁCI

Název bakalářské práce:

Systém řízení letu malého vírníku

Název bakalářské práce anglicky:

Flight control system for a small autogyro

Pokyny pro vypracování:

Bakalářská práce pojednává o systému řízení malého autonomního vírníku. Cílem projektu je analýza stávajících řídicích smyček a návrh vybraných nových funkcionalit, pomocí simulačních modelů, komerčních řídicích modulů pro malá UAV a drony, a s využitím metodiky SIL / HIL (software / hardware in-the-loop). Práce na projektu probíhají ve spolupráci s kolegy ze skupiny doc. Kováře z FEL ČVUT, kteří malý autonomní vírník vyvíjejí v rámci projektů pro meteorologické účely.

Pokyny pro vypracování:

- 1) Seznamte se s mechanikou letu vírníku. Proveďte srovnání s dynamikou letadla s pevným křídlem, případně vrtulníkem a dronem. Rozeberte možnost přemapování řídicích vstupů vírníku na vstupní signály běžného letounu.
- 2) Zprovozněte simulační model vírníku ve vhodném vizualizačním prostředí, například v programu FlightGear a zrealizujte propojení simulátoru s programovacím/simulačním prostředím.
- 3) Zprovozněte propojení emulátoru palubní řídicí jednotky vírníku.
- 4) Navrhněte a zrealizujte SIL experimenty zahrnující simulační model vírníku a emulátor palubní řídicí jednotky. Validujte a analyzujte vybrané základní funkcionality.
- 5) Navrhněte systém automatického přistání prostředku, zahrnující například výběr a otestování vhodného senzoru výšky; modelování konečného přiblížení a podrovnání; návrh, implementaci a simulační ověření jednoduchého zákona řízení; případně experimentálního letu k ověření funkčnosti systému nebo vybraných komponent.

Seznam doporučené literatury:

- [1] B. L. Stevens, F. L. Lewis. N. Johnson Aircraft Control and simulation . Third edition, John Wiley & Sons, Inc. 2016
- [2] Roger W. Pratt Johnson Flight Control Systems: Practical Issues in Design and Implementation. Institution of Engineering and Technology, 2000

Jméno a pracoviště vedoucí(ho) bakalářské práce:

doc. Ing. Martin Hromčík, Ph.D., katedra řídicí techniky FEL

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) bakalářské práce:

Datum zadání bakalářské práce: **26.01.2022**

Termín odevzdání bakalářské práce: _____

Platnost zadání bakalářské práce:

do konce letního semestru 2022/2023

doc. Ing. Martin Hromčík, Ph.D.
podpis vedoucí(ho) práce

prof. Ing. Michael Šebek, DrSc.
podpis vedoucí(ho) ústavu/katedry

prof. Mgr. Petr Páta, Ph.D.
podpis děkana(ky)

III. PŘEVZETÍ ZADÁNÍ

Student bere na vědomí, že je povinen vypracovat bakalářskou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací.
Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v bakalářské práci.

Datum převzetí zadání

Podpis studenta

Podakovanie

Chcel by som poďakovať doc. Ing.Martinovi Hromčíkovi, PhD. za jeho odborné vedenie tejto bakalárskej práce a Ing.Jakubovi Kákonovi za konzultácie a technickú podporu počas tvorby tejto práce.

Prehlásenie

Prehlasujem, že som predloženú prácu vypracoval samostatne a že som uviedol všetky použité informačné zdroje v súlade s Metodickým pokynom o dodržovaní etických princípov pri príprave záverečných prací.

V Prahe, 18. mája 2022

.....
Dominik Beňo

Abstrakt

Táto bakalárska práca pojednáva o vírniku a návrhu systému automatického pristávania zvyšujúceho šancu na úspešné pristátie. V rámci práce sú popísané a predstavené simulačné nástroje používané pri vývoji autopilota pre vírnik TF-G2. Pre potreby automatického pristávania sú vybrané a otestované senzory výšky a estimátor výšky. Je predstavená aerodynamika rotoru a vytvorený zjednodušený matematický model pozdĺžnej dynamiky vírniku v priblížení na pristátie. Kvôli pristávaniu je navrhnutý regulátor udržiavajúci vírnik v konštantnom klesaní počas priblíženia, ktorý je otestovaný v simulátore. Zároveň sú navrhnuté tri spôsoby regulácie pristávacieho manévru a otestované v simulátore vírniku TF-G2.

Kľúčové slová: vírnik, estimácia výšky, pozdĺžna dynamika, automatické pristávanie

Vedúci: Doc. Ing. Martin Hromčík,
Ph.D.
Katedra řídicí techniky,
Fakulta elektrotechnická,
České vysoké učení technické v Praze,
Karlovo náměstí 13,
121 35 Praha 2,
Česká republika

Abstract

This bachelor thesis deals with the autogyro and the design of an automatic landing system, increasing the chance of a successful landing. The work describes and introduces simulation tools used to develop autopilot for the TF-G2 autogyro. Altitude sensors are selected and tested for automatic landing with an altitude estimator. The rotor aerodynamics are presented, and a simplified mathematical model of the longitudinal dynamics of the autogyro in the approach to landing is created. A controller is designed to keep the autogyro in a constant descent during the approach, which is tested in the simulator. At the same time, three methods of landing maneuver control are proposed and tested in the TF-G2 autogyro simulator.

Keywords: autogyro, height estimation, longitudinal dynamics, automatic landing

Title translation: Flight control system for a small autogyro

Obsah

1 Úvod	1	6.1.3 Určenie síl a momentov	39
1.1 Ciele práce	1	6.1.4 Nelinearizovaný model	40
2 Porovnanie lietadla, vírniku a helikoptéry	3	6.2 Model regulátoru	41
2.1 Lietadlo	4	6.3 Overenie modelu	42
2.2 Vírnik	5	6.3.1 Overenie rovníc pre rotor . . .	42
2.3 Helikoptéra	6	6.3.2 Overenie modelu vírniku v priblížení	43
3 Vírnik TF-G2	9	7 Implementácia regulátorov	47
3.1 Ovládacie prvky vírniku TF-G2 .	9	7.1 Určenie prenosu	47
3.2 Autopilot PX4	10	7.2 Implementácia priblíženia	47
3.2.1 Flight stack	11	7.3 Implementácia pristávacieho manévru	48
4 Simulačné nástroje	15	7.3.1 Regulácia náklonu vírniku . . .	49
4.1 FlightGear	15	7.3.2 Regulácia vertikálnej rýchlosti s implicitným modelom	50
4.1.1 Komunikácia Matlabu a FlightGear	16	7.3.3 Regulácia vertikálnej rýchlosti s explicitným modelom	51
4.2 SIL Nástroj a QGroundControl .	17	8 Výsledky	53
5 Určenie výšky nad povrchom Zeme	19	9 Záver a práca do budúcnosti	55
5.1 Senzory	19	A Súradnicové systémy	57
5.1.1 Benewake TFmini-S LiDAR .	20	B Literatúra	59
5.1.2 Maxbotix I2CXL-MaxSonar-EZ4	20		
5.2 Porovnanie	20		
5.3 Systém estimácie výšky v PX4 .	21		
5.3.1 Úpravy v systéme estimácie výšky	22		
5.4 Otestovanie a vyhodnotenie systému estimácie výšky	23		
5.4.1 Testovanie senzorov	23		
5.4.2 Testovanie estimácie výšky . .	24		
5.4.3 Vyhodnotenie otestovania senzorov	24		
5.4.4 Vyhodnotenie otestovania estimácie výšky	27		
6 Modelovanie pristávacieho manévru	31		
6.1 Modelovanie vírniku	31		
6.1.1 Určenie kinetickej a potenciálnej energie	32		
6.1.2 Aerodynamika rotoru	33		

Obrázky

1.1	Obrázky vírníkov	2	6.1	Zobrazenie súradnicových systémov	32
2.1	Ukážka veličín v odchýlkovom modeli	3	6.2	Náčrt situácie pristávajúceho vírníku	33
2.2	Zjednodušená ukážka síl pôsobiacich na lietadlo	5	6.3	Ukážka elementu listu rotora z profilu	34
2.3	Ukážka síl roztáčajúcich/brzdiacich síl pôsobiacich na list rotora	5	6.4	Sily pôsobiace na element listu rotora	35
2.4	Porovnanie síl pôsobiacich na vírnik a helikoptéru [1]	7	6.5	Rotor v doprednom lete	37
3.1	Ukážka rozmerov vírníku TF-G2.	9	6.6	Element listu rotora	38
3.2	Graf uORB tém a modulov v PX4 použitý v kapitole 5	12	6.7	Vírnik v doprednom klesavom lete	40
3.3	Implementácia regulátora orientácie v Simulink-u	14	6.8	Odozvy častí regulátora orientácie s vírnikom na skok	42
4.1	Ukážka schémy SIL simulácie autopilotu PX4 s FlightGear pre vírnik TF-G2	18	6.9	Vybrané stavy vírníku počas manévru	44
5.1	Ukážka princípu ToF senzorov [15]	19	6.10	Odozva F_r a M_r	44
5.2	Benewake TFmini-S LiDAR [14]	20	6.11	Vybrané stavy vírníku počas manévru	45
5.3	Maxbotix I2CXL-MaxSonar-EZ4 [16]	20	6.12	Odozva F_r a M_r	45
5.4	Pripevnenie držiaku so senzormi k vírniku	21	7.1	Fitovanie G_z	48
5.5	Zobrazenie vírníku v náklone v jednej osi pri meraní výšky.	23	7.2	Blokový diagram regulácie počas priblíženia	48
5.6	Graf odmeraných hodnôt na plastovej doske	25	7.3	Pristávacie manéver s použitím regulácie náklonu vírníku	50
5.7	Namerané dáta pri rôznych povrchoch	26	7.4	Pristávacie manéver s použitím regulácie vertikálnej rýchlosti s implicitným modelom	51
5.8	Namerané dáta pri rôznych rýchlostiach	26	7.5	Pristávacie manéver s použitím regulácie vertikálnej rýchlosti s explicitným modelom	52
5.9	Porovnanie senzorov pri rôznych povrchoch a rýchlostiach	27			
5.10	Ukážka z logu počas prvého testovacieho vzletu vírníku	28			
5.11	Ukážka z logu pred pádom vírníku	29			
5.12	Ukážka z logu počas testovacieho vzletu vírníku	30			

Tabuľky

5.1 Porovnanie senzorov	21
6.1 Porovnanie vypočítaných hodnôt	42
6.2 Porovnanie mojej simulácie, simulácie a vzorcov podľa [20]	43

Kapitola 1

Úvod

Vývoj vírnikov začal na začiatku 20. storočia španielskym konštruktérom Juanom de la Ciervom, ktorý v roku 1920 zostrojil malý model vírniku, ktorý hodil zo strechy domovu a model hladko pristál. O tri roky neskôr prebehol prvý úspešný let vírniku. Technologický pokrok pri vývoji vírnikov položil základy pre vývoj helikoptéry. V priebehu 20. a 30. rokov 20. storočia vírniky začali byť popularizované ako bezpečné a stroje jednoduché na ovládanie. V priebehu 30. a 40. rokov boli v USA používané aj na transport pošty 1.1a. Vírniky boli počas druhej svetovej vojny aj súčasťou armád, avšak ich použitie bolo prehodnotené kvôli malému spektru úloh, na ktoré mohli byť nasadené. S postupným technickým vývojom sa zdokonaľovala helikoptéra, ktorá kvôli svojim vlastnostiam a letovým schopnostiam začala nahrádzať vírniky [1].

V dnešnej dobe sa vírnik najmä v Európe dostáva pomaly do popredia na športové a rekreačné účely, kde je možné na športových letiskách pozorovať postupný nárast vírnikov, najmä kvôli jeho bezpečnosti a nižšej cene oproti lietadlu 1.1b.

Jednou z hlavných výhod vírniku je jeho lepšia stabilita pri poryvoch vetra. Táto vlastnosť sa hodí pri lete v okolí búrok, kde je výhodnejšie mať vírnik narozdiel od lietadla. Preto sa vírnik TF-G2 používa pri meraniach v okolí búrok pre projekt CRREAT, do ktorého je zapojená aj elektrotechnická fakulta ČVUT, Ústav jadrovej fyziky a Ústav fyziky atmosféry AV ČR. Súčasná verzia vírniku TF-G2 1.1c, pomocou ktorého prebiehajú merania používa voľne dostupný autopilot PX4. Súčasná implementácia vírniku nepodporuje automatické pristávanie, čoho je následkom časté poškodenie podvozku alebo inej časti konštrukcie. To je hlavná motivácia pre automatické pristávanie a túto bakalársku prácu.

1.1 Ciele práce

Cieľom bakalárskej práce je vytvoriť prácu, ktorá položí základy budúcej práce pri vývoji vírniku TF-G2, najmä systému automatického pristávania vírniku. Čitateľ by mal mať po prečítaní základné povedomie o mechanike letu vírniku, súčiastiach PX4, použitých simulačných prostriedkoch a navrhnutých spôsoboch automatického pristávania. Tento cieľ pozostáva z viacerých úloh, ktoré je potrebné splniť.



(a) : Vírnik použitý na prepravu pošty [2]

(b) : Športový vírnik MTOsport 2017 [3]



(c) : Vírnik TF-G2 [4]

Obrázok 1.1: Obrázky vírnikov

■ Vírnik

- Popísať mechaniku letu vírniku
- Popísať použité subsystemy autopilotu PX4, ktoré budú použité pri návrhu automatického pristávania
- Vývoj komunikačného prostriedku medzi simulátorom FlightGear a Simulink-om

■ Estimácia výšky nad povrchom

- Výber senzorov výšky a ich implementácia do systému estimácie výšky do PX4
- Otestovanie senzorov a systému estimácie výšky

■ Model vírniku v priblížení

- Popis aerodynamiky rotoru
- Model vírniku v doprednom klesavom lete
- Overenie modelu

■ Návrh automatického pristátia

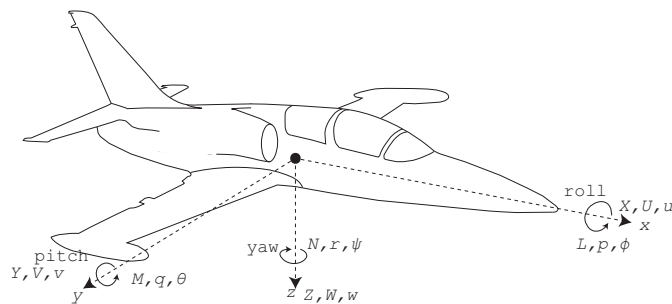
- Návrh regulátoru v priblížení a pristávacieho manévru
- Overenie systému automatikého pristávania v simulátore

Kapitola 2

Porovnanie lietadla, vírniku a helikoptéry

Pre porovnanie dynamiky lietadla a vírniku je potrebné si najprv zaviesť súradnicový systém, označiť a popísať veličiny pohybu. Letová dynamika bude popísaná v takzvanom body súradnicovom systéme vid'. A. Letová dynamika je popísaná v odchýlkovom modeli, kde jednotlivé veličiny ako lineárne rýchlosti, uhlové rýchlosti a poloha lietajúceho stroja (pre jednoduchosť ďalej použijem lietadla) znamenajú odchýlky voči ustálenému stavu, ktorý je reprezentovaný dopredným priamočiarym letom [5].

Veličiny popisujúce stav lietadla sú lineárne rýchlosti reprezentované vektorom $(U V W)^T$, zmeny týchto lineárnych rýchlostí su reprezentované ako $(u v w)^T$. Uhlové rýchlosti $(p q r)^T$ a orientácia je určená pomocou Eulerových uhlov $(\phi \theta \psi)^T$, nazývané po poradí roll, pitch, yaw. Jednotlivé veličiny sú zobrazené na obrázku 2.1. Zároveň na lietadlo pôsobia aerodynamické sily, gravitácia a sily od pohonu lietadla. Po ich rozložení do súradnicového systému a súčte vrámci jednotlivých osí, sú súčty týchto síl označené ako $(X Y Z)^T$. Pre momenty pôsobiace na lietadlo, sa dajú výsledné momenty otáčajúce lietadlo okolo osí súradnicového systému zapísať ako $(L M N)^T$.



Obrázok 2.1: Ukážka veličín v odchýlkovom modeli

Orientácia je popísaná pomocou Eulerových uhlov, ale v matematickom modeli ako uhlové rýchlosti nevystupujú derivácie Eulerových uhlov, ale uhlové rýchlosti $(p q r)$. Preto existuje transformačná matica 2.1 na ich výpočet [5].

Zjednodušene sa lietadlo ovláda pomocou štyroch riadiacich prvkov a to je ťah motoru τ , úroveň vyklopenia výškovky η , vyklopenie smerovky ζ a

vyklopenie krídelok ξ .

$$\begin{pmatrix} p \\ q \\ r \end{pmatrix} = \begin{pmatrix} 1 & 0 & -\sin \theta \\ 0 & \cos \phi & \sin \phi \cos \theta \\ 0 & -\sin \phi & \cos \phi \cos \theta \end{pmatrix} \begin{pmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix} \quad (2.1)$$

Mechanické rovnice sa získajú použitím druhého Newtonovho zákona pre translačný 2.2, ale aj rotačný pohyb 2.3, kde m je hmotnosť a J_x predstavuje moment zotrvačnosti okolo osi x a index i označuje príspevok od i -tej sily(momentu) spôsobeným aerodynamikou, riadiacim prvkom, gravitáciou.

$$m\dot{U} = \sum X_i \quad (2.2)$$

$$J_x \dot{p} = \sum L_i \quad (2.3)$$

Pre získanie stavového modelu je potrebné získať derivácie síl a momentov podľa jednotlivých stavových premenných, kvôli linearizácii. Sily a momenty závisia najmä na tvare, type a umiestnení riadiacich prvkov a vo všeobecnosti sa líšia a pre ich získanie je potrebná podrobnejšia analýza.

2.1 Lietadlo

Lietadlo je lietajúci stroj pozostávajúci z krídel, ktoré využívajú obtekajúci vzduch na generovanie vztlaku, a to mu umožňuje letieť. V tejto sekcii sa budem zaoberať lietadlom s vrtuľovým pohonom. O pohon lietadla sa stará motor, ktorý je často vpredú a má ťažnú vrtuľu. Vrtuľa ťahá lietadlo dopredu a rozdiel tlakov obtekajúceho vzduchu vytvára vztlak.

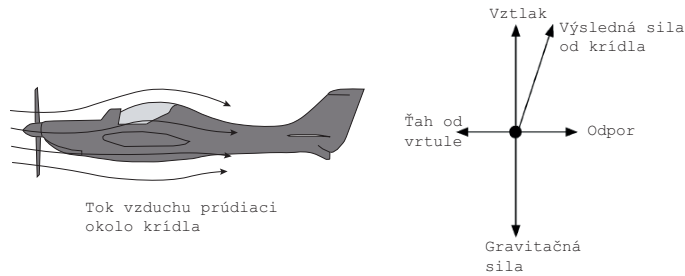
Podľa [5] je možné mechaniku letu lietadla rozdeliť na pozdĺžnu a bočnú. Pre pozdĺžnu sú významné štyri stavy, a to výchylky lineárnej rýchlosti v osiach x , z , uhlová rýchlosť rotácie okolo osi y a náklon okolo osi y ($u w q \theta$). Vstupmi u lietadla vplývajúce na pozdĺžnu mechaniku je ťah motoru a vychýlenie výškovky. Následne je ju možné zapísať v stavovom deskriptorovom tvare, ktorý sa získa dosadením do rovníc 2.2, 2.3 ako

$$\mathbf{M}\dot{\mathbf{x}} = \mathbf{A}'\mathbf{x} + \mathbf{B}'\mathbf{u} \quad (2.4)$$

Inverziou matice \mathbf{M} sa získa stavový model lietadla

$$\begin{pmatrix} \dot{u} \\ \dot{w} \\ \dot{q} \\ \dot{\theta} \end{pmatrix} = \begin{pmatrix} x_u & x_w & x_q & x_\theta \\ z_u & z_w & z_q & z_\theta \\ m_u & m_w & m_q & m_\theta \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} u \\ w \\ q \\ \theta \end{pmatrix} + \begin{pmatrix} x_\eta & x_\tau \\ z_\eta & z_\tau \\ q_\eta & q_\tau \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \eta \\ \tau \end{pmatrix} \quad (2.5)$$

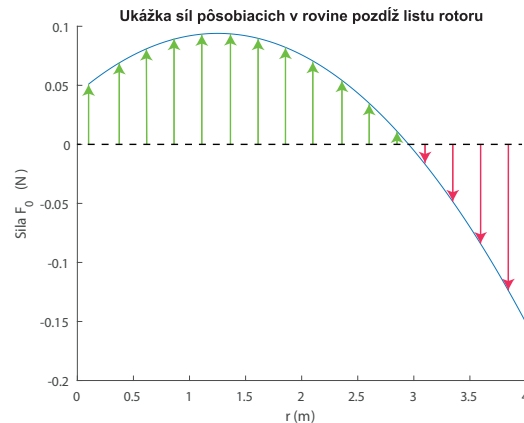
Rovnice popisujúce bočnú mechaniku lietadla sa získajú obdobným spôsobom ($v p \phi \psi$) a vstup ($\zeta \xi$) Zjednodušene u lietadla sa dá povedať, že v doprednom lete motor kompenzuje aerodynamický odpor a poháňa lietadlo. O vztlak sa starajú krídla.



Obrázok 2.2: Zjednodušená ukážka síl pôsobiacich na lietadlo

2.2 Vírník

Vírník je lietajúci stroj, ktorý využíva rotor na získanie potrebného vztlaku na let, pričom rotor na rozdiel od helikoptéry je nepoháňaný a tým pádom sa nestará o pohon. Prúdenie vzduchu pôsobiace na listy rotora spôsobuje moment, ktorý rotor roztáča. Tento jav sa nazýva autorotácia a je možné ho bežne pozorovať v prírode napríklad pri padaní krídlových plodov javoru, tzv. dvojnažiek. Z toho vyplýva, že pri vírníku vždy prúdi vzduch cez rotor nad rovinu otáčania rotora, čo je opak oproti helikoptére [1]. Rotor roztáčajú/-brzdia aerodynamické sily, ktoré sa menia v závislosti na vzdialenosti od osi otáčania rotora. Veľkosť a smer týchto síl taktiež závisí na uhle, pod ktorým vzduch prúdi na list rotora. Podrobnejšie je aerodynamika rotora rozobraná v podkapitole 6.1.2.



Obrázok 2.3: Ukážka síl roztáčajúcich/brzdiacich síl pôsobiacich na list rotora

O pohon vírníku sa podobne ako v prípade lietadla stará samostatný motor s vrtulou. V prípade bežných športových vírníkov ide o vrtulu tlačnú, najmä z dôvodu umiestnenia posádky. Vírník TF-G2 má vrtulu ťažnú. Motor poháňa vírník dopredu a tým pádom neustále prúdi vzduch roztáčajúci rotor.

Riadenie vírníkov sa líši na základe konštrukcie vírníku. Niektoré vírníky

nedokážu nakláňať rotor pozdĺžne a bočne zároveň, namiesto pozdĺžneho náklonu rotora majú výškovku. Vírnik TF-G2, ktorému sa v tejto práci venujem, disponuje nakláňaním rotora v pozdĺžnom a bočnom smere, ktoré nahrádza výškovku a krídelká. Vírnik TF-G2 má ťažnú vrtuľu a smerovku podobne ako lietadlá. Ďalším riadiacim prvkom niektorých vírnikov je kolektív, ktorý môže byť ekvivalentom klapiek v lietadlách. Pomocou kolektívu je možné nastavovať uhol sklonu listov rotora voči rovine rotácie rotora, takzvaný uhol nábehu. Vírniky s týmto ovládacím prvkom su schopné takmer kolmého štartu tzv. jump take-off a skoro kolmého pristátia [1].

Podľa [6] je možné zapísať linearizované rovnice pozdĺžnej mechaniky letu v doprednom priamočiarom lete nasledovne

$$\begin{pmatrix} \dot{u} \\ \dot{w} \\ \dot{q} \\ \dot{\theta} \\ \dot{\Omega} \end{pmatrix} = \begin{pmatrix} x_u & x_w & x_q & x_\theta & x_\Omega \\ z_u & z_w & z_q & z_\theta & z_\Omega \\ m_u & m_w & m_q & m_\theta & m_\Omega \\ 0 & 0 & 1 & 0 & 0 \\ Q_u & Q_w & Q_q & Q_\theta & Q_\Omega \end{pmatrix} \begin{pmatrix} u \\ w \\ q \\ \theta \\ \Omega \end{pmatrix} + \begin{pmatrix} x_\eta \\ z_\eta \\ q_\eta \\ 0 \\ Q_\eta \end{pmatrix} (\eta) \quad (2.6)$$

Porovnaním rovníc 2.5, 2.6 je vidieť, že model lietadla počíta aj s ťahom motoru. Hlavný rozdiel je v stupňoch voľnosti systému. Mechanika letu vírniku má o jeden stupeň voľnosti viac, zodpovedajúci uhlovej rýchlosti otáčania rotora Ω . Preto pri modelovaní vírniku je nutné uvažovať aj rotor a jeho uhlovú rýchlosť, ktorý prispieva k celkovej mechanickej energii systému svojou rotačnou energiou [6].

Medzi vírnikom a lietadlom je viacero rozdielov. Vírnik bežne potrebuje oveľa kratšiu dráhu na vzlet. Kratšia dráha na vzlet je spôsobená najmä tým, že počas štartu predtým než sa vírnik dá do pohybu sa rotor pretočí na výrobcom stanovené otáčky. Potom je potrebná kratšia dráha na získanie dostatočných otáčok rotora, ktoré sú potrebné na generovanie vztlaku. Vírniky potrebujú aj kratšiu pristávaciu dráhu, to je spôsobené tým, že rotor po pristáti sa stále točí a generuje vztlak a odpor, teda pri pristáti rotor funguje ako veľmi efektívna aerodynamická brzda. Výhodou vírniku počas letu oproti lietadlu je väčšia odolnosť proti poryvom vetru. U lietadla závisí vztlak na rýchlosti, ktorou vzduch obteká krídlo. Pri poryve vetru sa rýchlosť môže náhle zmeniť a tým pádom sa zníži vztlak a lietadlo sa môže "prepadnúť". U vírniku vztlak závisí nielen na rýchlosti ktorou obteká vzduch rotorový list, ale aj na otáčkach rotora, čím má pilot väčší reakčný čas zareagovať na poryv.

2.3 Helikoptéra

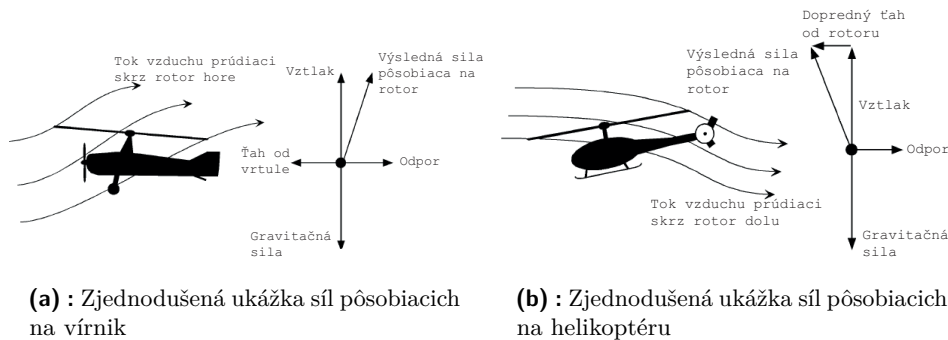
Helikoptéra je lietajúci stroj, ktorý využíva rotor na získanie potrebného vztlaku na let. Rotor helikoptéry je narozdiel od vírniku poháňaný pomocou motoru. Zároveň sa rotor helikoptéry stará aj o pohon v doprednom lete. Avšak v prípade výpadku motoru je možné helikoptéru používať ako vírnik a v autorotácii bezpečne pristáť [7].

Pri helikoptére je potrebné kompenzovať moment, ktorým motor roztáča rotor, aby helikoptéra držala správny smer. To sa dá zabezpečiť viacerými spôsobmi. Najznámejší spôsob akým sa to robí, je umiestniť na koniec helikoptéry ďalší rotor, točiaci sa vo vertikálnej rovine vytvárajúci moment v opačnom smere. Iný spôsob je umiestniť nad seba dva rotory točiace sa oproti sebe tzv. koaxiálne helikoptéry. Tento spôsob sa uplatňuje pri helikoptéroch Kamov Ka-52, ale napríklad aj na helikoptére NASA Ingenuity.

Riadiace prvky helikoptéry pozostávajú z ovládania motoru, ktorý roztáča rotor, nakláňanie roviny otáčania rotora v pozdĺžnom a bočnom smere (cyklika) a prvok umožňujúci zmenu uhlu nábehu listov rotora (kolektív). Pilot má k dispozícii ešte pedále ovládajúce zadný rotor, ktorým je možné meniť smer helikoptéry ψ . Oproti lietadlu pribudol ovládací prvok kolektív, ktorý sa v niektorých vírnikoch nachádza, ale nie všetkých.

Rovnice popisujúce mechaniku letu helikoptéry sa nezvyknú rozdeliť na pozdĺžnu a bočnú časť, ale helikoptéra sa zvyčajne popisuje ako celok [5].

Helikoptéra má viacero výhod narozdiel od vírniku, a to je možnosť kolmého štartu, vzletu, udržania sa na mieste tzv. hover. Helikoptéra sa môže pohybovať dopredu, dozadu, ale aj do strán. Tieto jej letové vlastnosti sú výhodné a v niektorých oblastiach je helikoptéra nenahraditeľná. Nevýhodou helikoptéry voči vírniku, je vyššia náročnosť ju ovládať, väčšia technická zložitosť, komplikovanosť a cena.

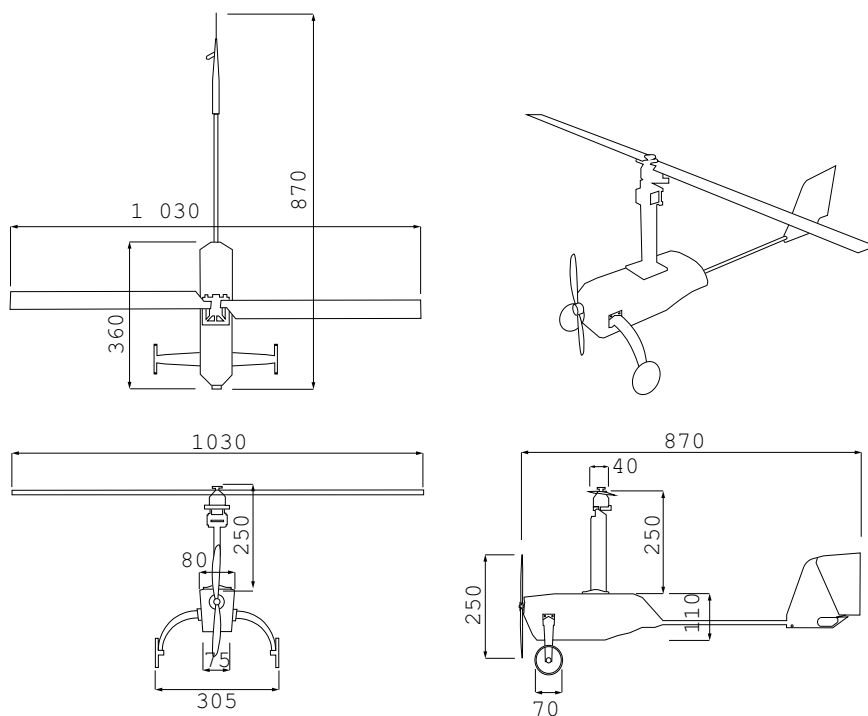


Obrázok 2.4: Porovnanie síl pôsobiacich na vírnik a helikoptéru [1]

Kapitola 3

Vířník TF-G2

Vířník TF-G2 je malý bezpilotný prostriedok spadajúci do kategórie EASA (European Union Aviation Safety Agency) A1-A3, ktorý slúži primárne na meranie pre projekt CRREAS a vývoj a otestovanie 3D tlačných súčiastok pre vířník, senzory a iných komponent. Jeho hlavnou výhodou je ľahká opraviteľnosť dielov pri nehode, keďže väčšina dielov je 3D tlačná a sú dostupné open-source [4]. Hmotnosť vířníku TF-G2 je 1700 g. Rozsah rýchlostí, ktorými sa vířník TF-G2 počas letu pohybuje je v rozmedzí 7 m s^{-1} až 25 m s^{-1} .



Obrázok 3.1: Ukážka rozmerov vířníku TF-G2

3.1 Ovládacie prvky vířníku TF-G2

Vířník TF-G2 má z ovládacích prvkov smerovku, má pozdĺžne a bočné naklápanie rotora a prípuť motoru. Ovládanie smerovky a naklápanie rotora

je ovládané servom. Motor, ktorý roztáča ťažnú vrtuľu je BLDC modelársky motor.

Kvôli namapovaniu vstupov do PX4, ale aj FlightGear sa používajú aj normalizované hodnoty vstupov. Normalizované hodnoty vstupu veličiny x sú označené x_n .

1. Nenormalizované vstupné veličiny:

- Pozdĺžne naklopenie rotoru: $\eta \in \langle 2, 22 \rangle$ ($^\circ$)
- Bočné naklopenie rotoru: $\xi \in \langle -3, 1 \rangle$ ($^\circ$)
- Otáčky motoru: $\tau \in \langle 0, 8830 \rangle$ (RPM)
- Uhol vyklopenia smerovky: $\zeta \in \langle -40, 40 \rangle$ ($^\circ$)

2. Normalizované vstupné veličiny:

- Pozdĺžne naklopenie rotoru: $\eta_n \in \langle -1, 1 \rangle$ (-)
- Bočné naklopenie rotoru: $\xi_n \in \langle -1, 1 \rangle$ (-)
- Otáčky motoru: $\tau_n \in \langle 0, 1 \rangle$ (-)
- Uhol vyklopenia smerovky: $\zeta_n \in \langle -1, 1 \rangle$ (-)

Pre prevod medzi nenormalizovanými a normalizovanými veličinami existujú nasledujúce transformačné vzťahy.

$$\eta = -10\eta_n + 12 \quad (3.1)$$

$$\xi = -2\xi_n - 1 \quad (3.2)$$

$$\tau = 8830\tau_n \quad (3.3)$$

$$\zeta = 40\zeta_n \quad (3.4)$$

3.2 Autopilot PX4

PX4 je profesionálny open-source autopilot pre drony a iné zariadenia, napríklad lietadlá, VTOL, rovere, ponorky, vírniky vyvíjaný celosvetovými vývojármi z akademického, ale aj priemyselného prostredia. Medzi ďalšie výhody okrem veľkej modularity patrí aj veľký výber hardware, ktorý môže byť s PX4 použitý napríklad senzory, ale aj samotné letové kontroléry[8].

Celkovo subsystémy PX4 je možné rozdeliť do dvoch vrstiev a to flight stack a middle ware. Vyššia časť middleware má na starosti všetky všeobecné úlohy, ktoré sú podporované vrámci všetkých platforiem, a to komunikácia s okolím, ovládače senzorov a systém preposielania správ vrámci systému uORB správy. Vrstva flight stack obsahuje navigačné, riadiace algoritmy a ostatné súčasti potrebné pre riadenie letu napríklad regulátory, estimátory polohy, senzory a samotné aktuátory a servá. [8].

■ uORB Správy

uORB je middleware slúžiaci na zasielanie správ v distribuovanom počítaní. Skratka ORB je z Object Request Broker a v PX4 sa používa implementácia uORB. Je to asynchrónne API slúžiace na preposielanie správ vrámci procesov/vláken. Napríklad každý estimátor, regulátor predstavuje samostatnú aplikáciu, ktorá odoberá publikujúce témy [9].

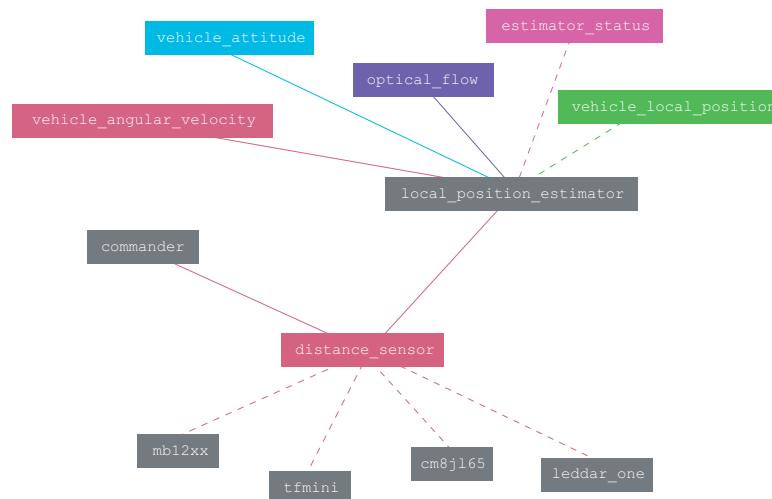
Pomocou uORB môžu jednotlivé subsystémy(moduly) komunikovať a predávať si informácie. Napríklad pre vypočítanie zásahu regulátoru je nutné poznať hodnoty meraných veličín ako je orientácia, rýchlosť a pod, ktoré sa vypočítavajú v inom procese/vlákně ako je regulátor. uORB systém sa skladá z viacerých častí:

- **Správa:** Definuje, aká informácia sa bude posielat a v akom formáte, kde sa špecifikuje aj dátový typ položiek. Táto informácia sa zapisuje do .msg súborov a až následne sa preloží do C/C++ štruktúry. V správach sa posielajú konkrétne informácie, napríklad namerané dáta zo senzorov pre estimátor alebo sa posielajú orientácia prostriedku regulátorom.
- **Téma:** Téma je komunikačný lievik, kde sú správy prijaté a odoslané a pomocou neho sú správy sprostredkované medzi modulmi. Téma zahŕňa jednotlivé správy, ktoré sú publikované/odoberané.
- **Modul:** Modul publikuje/odoberá jednotlivé témy a pracuje s dátami, ktoré sú obsiahnuté v správach jednotlivých tém. Napríklad ako modul je možné si predstaviť senzor vzdialenosti, ktorý publikuje dáta inému modulu, estimátoru. Ten ich pomocou algoritmov spracuje, odošle modulu regulátoru, ktorý pomocou nich vypočíta akčný zásah.
- **Metóda publikovanie/odoberanie:** Každý modul môže publikovať alebo odoberat témy. Publikovanie predstavuje odosielanie dát inému modulu, odoberanie prijímanie dát od iného modulu. Napríklad senzory publikujú namerané dáta a estimátor tému od senzoru odoberá. Po vypočítaní polohy orientácie a iných veličín estimátor tieto dáta publikuje regulátoru, ktorý odoberá danú tému.

Na obrázku 3.2 je časť grafu znázorňujúceho vzťahy medzi jednotlivými modulmi a témami. Moduly sú označené sivou farbou a témy sú označené farebne. Odoberanie je naznačené prerušovanou čiarou a publikovanie plnou. Z grafu je možné pozorovať, že viacero senzorov vzdialeností publikuje správy rovnakej témy `distance_sensor`. Táto téma je odoberaná modulom `local_position_estimator`, ktorý publikuje do iných tém, ale aj do témy `vehicle_local_position`.

■ 3.2.1 Flight stack

Flight stack je rozdelený na viacero častí, pričom niektoré dôležité časti budú rozobraté nižšie.



Obrázok 3.2: Graf uORB tém a modulov v PX4 použitý v kapitole 5

■ Estimátor

Autopilot PX4 používa na estimáciu stavov rozšírený kalmanov filter EKF (Extended Kalman Filter). Estimácia stavov je dôležitá z toho dôvodu, že nie všetky senzory merajú konkrétne veličiny potrebné pre reguláciu, napríklad senzor vzdialenosti nemerá pravú výšku nad zemou, keď je celý prostriedok naklonený. Estimátor dokáže používať viacero senzorov na vypočítanie stavov. Konkrétne v PX4 je EKF algoritmus naimplementovaný v ekf2 module. Celkovo má estimátor 24 stavov[8].

- Uhlová orientácia v kvaterniónoch(4 stavy)
- Rýchlosti v NED systéme(3 stavy)
- Pozícia v NED systéme(3 stavov)
- Offset v IMU a magnetometri(9 stavov)
- Magnetické pole Zeme(3 stavy)
- Rýchlosť vetru v systéme NE(North-East)(2 stavy)

V PX4 môže byť spustených viacero instancií estimátrom EKF. Táto funkcionality je naimplementovaná z dôvodu zvýšenej ochrany pred chybou senzoru, pričom jednotlivé instance používajú rozličné kombinácie senzorov. Nad instanciami EKF je implementovaný EKF selektor, ktorý je schopný identifikovať instanciu EKF s najlepšou konzistenciou dát a používať jej dáta.

EKF môže používať viacero druhov senzorov ako inerciálna meracia jednotka(IMU), magnetometer, barometer, senzor vzdialenosti, GPS, kameru a senzor na meranie rýchlosti prostriedku voči vzduchu(airspeed).

■ Regulátor pozície

Vírník TF-G2 nemá svoj vlastný regulátor pozície. Pre vírník TF-G2 sú použité regulátory pre lietadlá s fixným krídlom, ktorý je naimplementovaný v module `fw_pos_control_11`. Regulácia polohy dostáva informáciu o požadovanej polohe lietadla skrz ovládač alebo systém, ktorý plánuje trajektóriu pohybu. Výstupom regulátoru pozície sú požadované hodnoty pre prípusť motoru τ_n , a požadované hodnoty orientácie prostriedku ϕ_{sp} , θ_{sp} . Pričom x_{sp} označuje požadovanú hodnotu veličiny x . Smer, ktorým smeruje pozdĺžna os prostriedku voči severu (heading) sa nastaví pomocou ϕ_{sp} .

Regulátor pozície sa skladá z dvoch regulátorov. TECS (Total Energy Control System) regulátor reguluje podĺžny smer prostriedku. Jeho vstupom je požadovaná výška h_{sp} a požadovaná airspeed v_{sp} . V rámci regulátoru sú implementované dve slučky, ktoré majú rovnaké vstupy, ale líšia sa vo výstupoch. Prvá z nich funguje na princípe celkovej mechanickej energie prostriedku, pričom sa potenciálna energia dopočíta pomocou h_{sp} a kinetická pomocou v_{sp} . Výstupom je τ_n . Výstupom druhej regulačnej slučky je náklon θ_{sp} a ten sa určí na základe pomeru kinetickej a potenciálnej energie. Celkovo sa dá na systém nahliadnuť štýlom, že vstupom je požadovaná celková mechanická energia systému a akčným prvkom je prípusť motoru, pričom pomer potenciálnej a mechanickej energie sa mení zmenou θ , ale zmena náklonu energiu do systému nepridáva [8].

Druhý regulátor sa nazýva L1 a jeho výstupom je požadovaný náklon ϕ_{sp} , regulátor reguluje smer letu (heading), pričom vstupom je aj mód autopilota, či má udržiavať hladinu, krúžiť, dostať sa nad konkrétne miesto a na základe toho ovláda bočný pohyb prostriedku pomocou ϕ_{sp} [10].

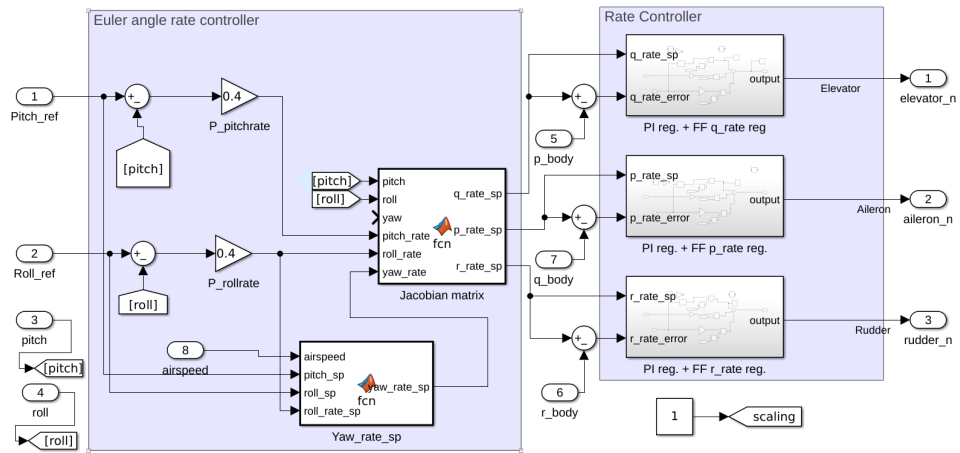
■ Regulátor orientácie

Vírník TF-G2 nemá svoj vlastný regulátor pozície. Využíva regulátor orientácie pre lietadlá s fixným krídlom tzv. FixedWing Attitude Controller, ktorý je implementovaný v module `fw_att_control`. Tento regulátor má na vstupe referenciu náklonov ϕ_{sp} , θ_{sp} , ktorú mu určí nadradená slučka regulátor pozície alebo letový mód autopilota, kde môže ako referencia orientácie byť vstup z diaľkového ovládania. Regulátor orientácie sa skladá z dvoch slučiek. Vonkajšia slučka spracováva referenciu danú ako Eulerove uhly. Regulačnú chybu vynásobí P regulátorom volaním metódy `control_attitude` a následne zmenu Eulerových uhlov pretransformuje do body súradníc pomocou jakobiánu 2.1 zavolaním metódy `control_euler_rate` a tým určí referenciu uhlových rýchlostí vnútornej slučky.

Vnútoraná slučka sa nazýva Rate Controller a reguluje požadované uhlové rýchlosti (p_{sp} q_{sp} r_{sp}), ktoré sú výstupom nadradenej slučky. Výstupom vnútornej slučky sú hodnoty riadiacich signálov η_n , ζ_n , ξ_n , ktoré sa vypočítajú zavolaním metódy `control_bodyrate` [11].

Rozdiel medzi implementáciou tohto regulátoru pre vírník a lietadlo je v tom, že škálovanie na základe airspeed je pre vírník vypnuté. Čo sa týka

implementácie regulátoru pre uhly ϕ , θ ich štruktúra je rovnaká, na základe toho, že je možné pozdĺžnu a bočnú dynamiku nezávisle rozdeliť, ako bolo spravené v kapitole 2. Vnútna slučka regulátoru orientácie je rovnaká pre všetky rýchlosti pqr a líši sa v konštantách.



Obrázok 3.3: Implementácia regulátoru orientácie v Simulink-u

Kapitola 4

Simulačné nástroje

V tejto kapitole rozoberiem použité simulačné nástroje pri tejto bakalárskej práci. Simulácia v tomto prípade zohráva dôležitú úlohu, keďže sa znižuje množstvo testovacích letov, a tým aj riziko poškodenia vírniku. Táto kapitola má zároveň slúžiť ako návod na spoznanie simulačných programov a jednoduchý úvod do práce s nimi.

4.1 FlightGear

FlightGear je letecký simulátor, vytvorený medzinárodnou skupinou dobrovoľníkov a je podporovaný na viacerých platformách (Windows, Linux, Mac atd.) [12]. Jednou z jeho výhod je to, že je to voľne dostupný software pod licenciou GNU. Úpravy simulátoru FlightGear sú umožnené pomocou hierarchického stromu nazvaného "property tree". V ňom sú zoradené vlastnosti simulácie zodpovedajúce jednotlivým kategóriám ako prostredie, počasie, parametre simulácie, poloha a orientácia lietadla, riadiace prvky a pod..., podľa hierarchie. Užívateľ k nim môže pristupovať a meniť ich podľa potreby. Táto vlastnosť je dôkladnejšie popísaná v podkapitole 4.1.1.

FlightGear podporuje viacero modelov letovej dynamiky FDM (Flight dynamics model). Medzi najpoužívanejšie patria JSBSim, YASim, UIUC. Vírnik TF-G2 používa YASim FDM na simuláciu letovej dynamiky vírniku [13]. Jednou z výhod použitia tohto FDM je to, že letová dynamika sa vypočítava na základe geometrie vírniku, t.j. rozmerov a vlastností jednotlivých komponentov, ktoré sú známe, keďže časti vírniku sú vytlačené na 3D tlačiarňi.

Nasledujúcou vlastnosťou FlightGear-u je realistická možnosť simulácie počasia a vetra. To umožňuje testovať autopilota v podmienkach, v ktorých bude vírnik lietať a tak prípadne predísť nehode počas testovacieho letu.

Užívateľ môže FlightGear s prednastavenými nastaveniami v Linuxe spustiť pomocou grafického prostredia alebo terminálu použitím príkazu `fgfs`. Zároveň je možné FlightGear spustiť s určitými parametrami a druhmi lietajúcich strojov. Pre spustenie simulácie vírniku TF-G2 je potrebné mať stiahnutý priečinok so súbormi pre simulátor a umiestniť ho do priečinku `$FG_Root/Aircraft` [12]. Napríklad pre spustenie simulátoru pre vírnik TF-G2 bez vetra sa použije príkaz `fgfs --aircraft=TF-G2 --wind=0@0`. Tieto voliteľné parametre môže užívateľ nájsť po zadaní `fgfs --help --verbose`.

4.1.1 Komunikácia Matlabu a FlightGear

Pre prácu a zoznámenie sa so systémom bol vyvinutý jednoduchý systém na komunikáciu Matlabu/Simulinku a leteckého simulátora FlightGear. Dôvod na implementáciu tohoto nástroja je ten, že nedisponujem FDM, ten je implementovaný vo FlightGear. Lenže vo FlightGear sa ťažko implementujú a realizujú konkrétne riadiace signály na vstupy vírniku (napr. jednotkový skok, impulzy, sínusovka a pod.). Taktiež výhodou Matlabu a Simulinku je to, že je široko dostupné, obsahuje veľa knižníc a je jednoduché na použitie. V Simulinku sa dajú implementovať riadiace slučky a jednoducho nastavovať parametre regulátoru. Výhodou FlightGear-u je to, že obsahuje zložitý model vírniku, počasie, ktoré by sa v Matlabe/Simulinku implementovali zložito a grafiku. Zároveň je možné FlightGear použiť ako vizualizačné prostredie, teda FDM lietadla je v Matlabe/Simulinku a FlightGear pomocou komunikácie zobrazuje lietadlo a jeho polohu.

FlightGear podporuje viacero možností komunikácie napr. sériový port, TCP, UDP protokol. Je potrebné ešte nakonfigurovať komunikačný protokol tzv. "generic protocol"[12]. Pre prípad vírniku TF-G2 je nakonfigurovaný protokol v súbore `my_udp.xml`. Vstup a výstup si definuje užívateľ, môže to byť multiplexovaný/demultiplexovaný signál v Simulink-u. Konkrétne dáta, ktoré sú prenášané pomocou UDP paketov sú definované v komunikačnom protokole. V ňom si užívateľ zadá konkrétny prvok, jeho typ a to, či ide o vstup alebo výstup viď.[12].

Daná "adresa" záznamu (node) je zo stromu vlastností (property tree). Tento strom je možné otvoriť v okne simulátora v hornej lište pod záložkou "Debug -> Browse internal properties". Tam je možné nájsť tzv. stavové premenné [12]. Daný typ a aktuálnu hodnotu vlastnosti je možné vidieť v danom okne.

Pre spustenie komunikujúceho FlightGear je potrebné ho spustiť pomocou príkazu nižšie[12].

```
fgfs --generic=socket,<in/out>,<freq>,<adress>,<port>,<serial/
udp/tcp>,<protocol_filename>
```

Kde:

- `in/out`=definujem či dané nastavenie je pre vstup alebo výstup
- `freq`=frekvencia[Hz], ktorou sa posielajú dáta
- `adress`=IP adresa príjemcu/odosielateľa, zvyčajne `localhost`
- `serial/udp/tcp`=spôsob komunikácie
- `protocol_filename`=názov XML súboru, ktorý definuje komunikáciu

V prípade pre komunikáciu Matlabu a FlightGear v prípade použitia pre vírnik TF-G2 sa FlightGear volá nasledujúcim príkazom.

```
fgfs --aircraft=TF-G2 --generic=socket,out,100,localhost,5501,
udp,my_udp --generic=socket,in,100,localhost,5502,udp,
my_udp
```

Prepojenie simulátoru FlightGear a Simulinku je realizované pomocou subsystému v Simulink-u. Subsystém má port pre vstup a port pre výstup a zabezpečuje komunikáciu s FlightGear. Simulinkový subsystém obsahujúci nastavenú komunikáciu s FlightGear pre vírnik TF-G2 je v súbore `autogyro_model.slx`. V tomto prípade je komunikácia zabezpečená pomocou UDP paketov. Subsystém obsahuje signál pre vstup a výstup. Do/Z nich si užívateľ multiplexuje/demultiplexuje signál, podľa poradia definovaného v komunikačnom protokole a priradí signály. Momentálne subsystém obsahuje funkcionality zrušenia pozastavenia simulácie vo FlightGear, akonáhle začne simulácia v Simulinku. Keďže ovládacie prvky vírniku vo FlightGear sú normalizované, tak subsystém obsahuje aj funkcie na transformáciu ovládacích prvkov z normalizovaných hodnôt do absolútnych hodnôt riadiacich signálov vid. kapitola 3. V nasledujúcich subsystémoch subsystému `autogyro_model.slx` je riešený typ signálov a komunikácia pomocou UDP protokolu.

Takýto spôsob komunikácie je vhodný pre viacero lietadiel, helikoptér a pod. nie iba pre vírnik TF-G2 v leteckom simulátore FlightGear. Umožňuje implementovať jednoduché regulačné slučky s využitím výhod Simulink-u a pokročilého FDM, ktoré je obsiahnuté vo FlightGear. Na komunikáciu s inými lietadlami je potrebné pozmeniť komunikačný protokol, tak aby obsahoval "properties" daného lietadla. Taktiež je potrebné pozmeniť subsystém v Simulinku, aby odpovedal danému počtu vstupujúcich/vystupujúcich signálov, ich typu a pod. Princíp zostáva rovnaký a to je identifikovanie "properties", ktoré sú dôležité, úprava komunikačného protokolu a úprava subsystému v Simulink-u. Takýto subsystém sa dá použiť na logovanie dát, nejakej vybranej časti letu, z FlightGear a ich jednoduchší prenos do prostredia Matlab/Simulink. V prostredí Matlab/Simulink už užívateľ môže použiť široké spektrum nástrojov na identifikáciu, spracovanie signálu, vykresľovanie grafov a podobne.

Po naimplementovaní tohto prostriedku som zistil, že existujú podobné systémy, ktorých úloha je rovnaká a fungujú na podobnom princípe ako mnou naimplementovaný komunikačný prostriedok. Naďalej v práci budem používať mnou implementovaný komunikačný prostriedok.

4.2 SIL Nástroj a QGroundControl

SIL je skratka pre simulovanie softvéru v slučke (Software in the Loop). Ide o proces, v ktorom sa program, regulačné slučky, logika odsimuluje v prostredí simulátoru ešte predtým než sa nahraje na kontrolér. Účelom takejto simulácie je otestovanie správnosti kódu, a tým úspešné odhalenie chýb, ktoré by mohli mať za následok nehodu. Ekvivalentom k SIL je HIL (Hardware in the Loop), kde sa nesimuluje iba program, ale zároveň hardvér, na ktorom kód počas letu pobeží.

V tejto bakalárskej práci je jedným z cieľov vývoj funkcionalít, ktoré budú neskôr implementované do autopilota PX4, ktorý bol popísaný v podkapitole 3.2. SIL je zároveň podporovaná autopilotom PX4. Pre SIL simuláciu autopilota PX4 je podporovaných viacero simulátorov, medzi ktoré patrí

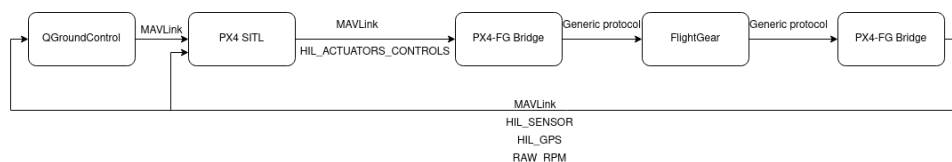
aj FlightGear[8]. Komunikáciu simulátoru FlightGear a SIL simulácie PX4 zabezpečuje PX4-FlightGear Bridge.

Na simuláciu je vhodné mať nainštalovanú aplikáciu QGroundControl, ktorá má vytvorené grafické prostredie pre autopilota PX4, pomocou ktorého môže užívateľ ovládať lietadlo, nastavovať letové režimy autopilota, zobrazit si aktuálnu polohu, rýchlosť, výšku a iné parametre letu. Zároveň je v nej možné plánovať trajektóriu letu, pozrieť si stav lietadla a podobne [8].

To čo sa simuluje pre PX4 a vírnik TF-G2 pozostáva z bežiacich troch programov, ktoré medzi sebou komunikujú na transportnej vrstve pomocou TCP/UDP protokolov. Protokol použitý na vyššej vrstve je MAVLink prípadne generic protokol popísaný v podpodkapitole 4.1.1.

Na obrázku 4.1 je možné vidieť diagram zapojenia PX4 SIL simulácie, FlightGear, QGroundControl a funkciu PX4-Flightgear Bridge. Zároveň je na obrázku znázornené aký protokol sprostredkuje komunikáciu medzi jednotlivými súčasťami simulácie. Pre MAVLink sú vypísané použité druhy správ. V rámci SIL simulácie sa používajú správy pre HIL simuláciu. Špecifikácia obrázku pre vírnik TF-G2 spočíva jedine v type použitých správ protokolu MAVLink. Pre vírnik TF-G2 sú použité nasledujúce MAVLink správy:

- HIL_ACTUATOR_CONTROLS=vestupy na riadiace prvky vírniku
- HIL_SENSOR=dáta zo senzorov
- HIL_GPS=GPS dáta
- RAW_RPM=dodatočne implementovaná MAVLink správa pre vírnik TF-G2 obsahujúca otáčky rotoru



Obrázok 4.1: Ukážka schémy SIL simulácie autopilota PX4 s FlightGear pre vírnik TF-G2

Kapitola 5

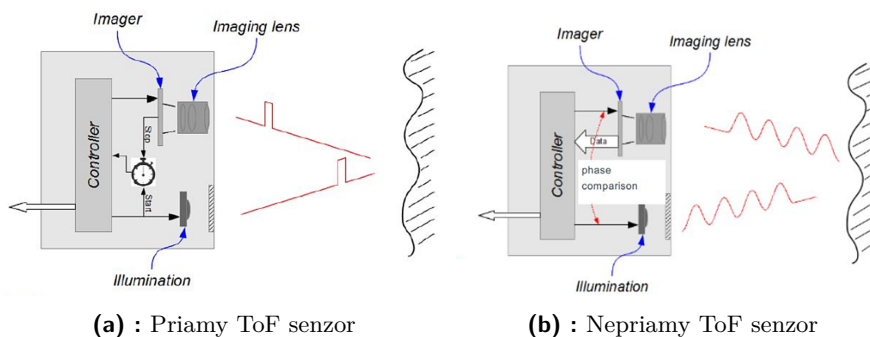
Určenie výšky nad povrchom Zeme

5.1 Senzory

Pre úspešné pristátie je potrebné správne odmerať výšku nad povrchom Zeme z dôvodu zahájenia pristávacieho manévru vírniku, zníženia rýchlosti a bezpečného dosadnutia. Keďže vírnik lieta nad rôznymi povrchmi je dôležité správne merať výšku nad všetkými povrchmi, na ktorých môže vírnik pristáť správne, jedná sa o polia, lúky, asfalt, nespevnené cesty, štrk, piesok. Vyberalo sa spomedzi viacerých senzorov. Najdôležitejším parametrom pre výber bola existencia ovládaču pre PX4 následne maximálna možná odmeraná vzdialenosť a frekvencia dostupnosti nových hodnôt.

Vybrané senzory fungujú na princípe ToF (Time of flight), merajú čas za ktorý sa vyslaný signál vráti do prijímaču. Senzory fungujúce na princípe ToF sa delia do dvoch kategórii: priame a nepriame ToF senzory. V prípade priamych ToF senzorov sa posielajú krátky pulz a následne sa meria čas, za ktorý pulz príde. Nepriame ToF senzory vysielajú modulovaný spojité signály. Prijímač meria fázu prijatého signálu a na základe nej dopočíta čas letu. Zo známeho času letu a na základe známej rýchlosti šírenia signálu prostredím, vieme jednoducho dopočítať vzdialenosť. Rovnica 5.1 vyjadruje vzdialenosť pre nepriamy ToF senzor, kde c je rýchlosť šírenia signálu v prostredí, $\Delta\phi$ značí fázový posun a f je frekvencia modulovaného signálu [14].

$$d = c \frac{1}{4\pi f} \Delta\phi \quad (5.1)$$



Obrázok 5.1: Ukážka princípu ToF senzorov [15]

■ 5.1.1 Benewake TFmini-S LiDAR

Tento senzor je zástupcom nepriamych ToF senzorov. Senzor vysiela modulovaný laserový signál z infračerveného spektra. Senzor už má implementovaný ovládač pre PX4. S PX4 komunikuje pomocou sériového rozhrania UART. Jeho výhodou je maximálny dosah až 12 m a vysoká opakovateľnosť meraní 100 Hz. Slepá zóna senzoru je do 10 cm. Nevýhodou je, že tento senzor má problém s povrchmi, ktoré majú nízku odrazivosť a tým pádom sa môže znížiť jeho maximálny dosah [14].



Obrázok 5.2: Benewake TFmini-S LiDAR [14]

■ 5.1.2 Maxbotix I2CXL-MaxSonar-EZ4

Senzor vysiela vysokofrekvenčný zvuk o frekvencii 42 kHz. Komunikuje s PX4 pomocou I2C rozhrania a má implementovaný ovládač pre PX4. Senzor funguje v rozsahu 0 – 765 cm. Pri dobrých podmienkach je maximálna možná frekvencia odčítania dát 40 Hz. Ako jeho nevýhodu uvažujem celkom nízku frekvenciu meraní, ktorá je nízka najmä pri veľkých vzdialenostiach. Senzor disponuje automatickou kalibráciou a odstránením šumu v reálnom čase každé meranie. Senzor je vhodný na detekciu ľudí, pohybu, pristávanie lietajúcich objektov, autonómnou navigáciu a na varovanie pred kolíziou [16]. V práci budem pre tento senzor používať aj skrátené označenie MB1242, predstavujúce typ senzoru.



Obrázok 5.3: Maxbotix I2CXL-MaxSonar-EZ4 [16]

■ 5.2 Porovnanie

V tabuľke 5.1 sa nachádza zhrnutie a porovnanie vlastností senzorov.

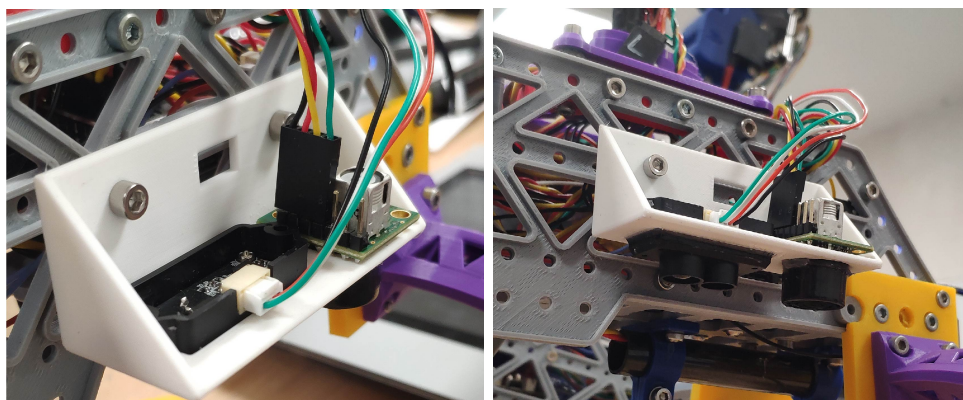
Senzor	TFmini-S	MB1242
Dosah (m)	0.1-12	0-7.65
Opakovanie merania (Hz)	100	40
Komunikácia	UART/I ² C	I ² C
Operačná teplota (°C)	0-60	0-65
Napájacie napätie (V)	5	3-5,5
Hmotnosť (g)	5	6
Spotreba (mW)	700	22
Cena (€)	39,2	34,5

Tabuľka 5.1: Porovnanie senzorov

Podľa užívateľských príručiek majú oba senzory v prostredí, kde sa horšie odráža laser maximálny dosah 7,5 m. Oba senzory majú podobnú hmotnosť a sú malé, takže nie je problém ich namontovať na vírnik TF-G2. Výhodou senzoru na laserovom princípe je tá, že meria častejšie dáta a má väčší dosah. Výhodou ultrazvukového senzoru by mala byť menšia chybovosť spôsobená podložkou.

5.3 Systém estimácie výšky v PX4

Na integráciu senzorov do autopilota PX4 je potrebné mať ovládače pre oba senzory v priečinku `PX4Firmware`, skompilovať autopilota a nahráť do riadiacej jednotky. V aplikácii `QGroundControl` sa povolajú oba senzory a nastaví sa názov konektoru, do ktorého je senzor zapojený. Pre senzory som navrhol držiak, do ktorého sa pripevnia senzory a následne sa držiak z boku pripevní pomocou skrutiek k vírniku.



Obrázok 5.4: Pripevnenie držiaku so senzormi k vírniku

PX4 umožňuje pridať senzor výšky tzv. rangefinder do rozšíreného Kalmanovho filtru EKF (Extended Kalman Filter) na estimáciu výšky nad terénom zapnutím parametru `EKF2_RNG_AID`. V tomto prípade EKF kombinuje dáta z viacerých senzorov GPS, IMU (inerciálna meracia jednotka), barometer, senzor

výšky a vypočíta odhad výšky. Primárny senzor, z ktorého sa počíta výška, sa nastavuje pomocou `EKF2_HGT_MODE` parametru, na výber je spomedzi GPS, barometru, senzoru vzdialenosti alebo kamery[8].

Následne sa nastavujú parametre kvôli polohe senzoru výšky voči ťažisku lietadla, pre transformáciu hodnoty zo senzoru, do súradníc vírniku. Vzdialenosť odmeraná senzorom výšky sa nachádza v uORB správach témy `distance_sensor`. Estimovaná výška nad povrchom zeme z EKF sa nachádza v uORB správach `vehicle_local_position` pod položkou `dist_bottom` [8].

Ďalšou možnosťou ako implementovať senzor do EKF je nastavenie parametru `EKF2_GND_MAX_HGT`. Nastavenie toho parametru nastaví výšku, pod ktorou sa má zapnúť estimácia výšky nad povrchom pomocou EKF použitím senzoru vzdialenosti. Táto funkcia je implementovaná najmä kvôli kompenzácii efektu, spôsobeného tlakom vzduchu pri zemi, ktorý dokáže výrazne ovplyvniť senzor tlaku a tým zároveň aj meranie výšky pri pristávaní[8].

5.3.1 Úpravy v systéme estimácie výšky

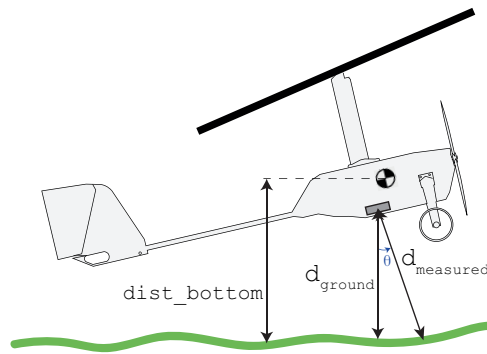
Aktuálne PX4 používa na estimáciu výšky ako zdroj dát jeden senzor vzdialenosti. Senzor je vybraný na základe najnižšieho čísla instance senzoru[17]. V mojom prípade to je ultrazvukový senzor s kratším dosahom. Z toho dôvodu som upravil a do kódu pridal prepínanie senzorov výšky adaptované z [18]. Primárne sa bude používať senzor výšky, ktorý má väčší rozsah, v prípade chyby merania sa senzor prepne na ďalší senzor, ktorý meria validnú hodnotu, indikovanú hodnotou `signal_quality` v správach uORB témy `distance_sensor`. O validite merania rozhoduje ovládač senzorov. Do PX4 boli pridané aj logovacie signály, ktoré by mali pomôcť zhodnotiť funkciu estimátoru.

Do správy `vehicle_local_position` som dodatočne implementoval položku `dist_bottom_raw`. Do tejto položky sa bude v priebehu letu zaznamenávať pre násobení nameraná hodnota zo senzorov d_{measured} , kvôli kompenzácii náklonov vírniku.

$$d_{\text{ground}} = k d_{\text{measured}} \quad (5.2)$$

Koeficient k z rovnice 5.2, súvisí s kompenzáciou náklonov vírniku ak je rotácia popísaná v Eulerových uhloch. Podľa obrázku 5.5 je vidieť, že vzťah pre kolmú vzdialenosť je možné triviálne pomocou Pytagorovej vety odvodiť a nachádza sa v rovnici 5.3. Ekvivalentne sa na problém dá pozrieť použitím rotačnej matice zo zafixovaného lokálneho súradnicového systému do body súradnicového systému pevne spojeného s konštrukciou vírniku. Položka v poslednom riadku a poslednom stĺpci tejto matice, je hľadaný koeficient k . Je potrebné dodať, že pri výpočte v PX4 sa nepoužívajú priamo hodnoty uhlov, ale estimovaná rotačná matica, ktorá sa prepočítava vrámci behu rozšíreného Kalmanovho filtru [17].

$$d_{\text{ground}} = \cos(\phi) \cos(\theta) d_{\text{measured}} \quad (5.3)$$



Obrázok 5.5: Zobrazenie vírniku v náklone v jednej osi pri meraní výšky.

Následne som pridal do logu informáciu o aktuálnom použitom senzore merania vzdialenosti `dist_bottom_sensor_instance` v podobe čísla instance senzoru. Zároveň som pridal aj správu o tom, či je povolený range aid `aid_suitable`. Oba informácie sú implementované v rozšírení uORB témy `vehicle_local_position`.

5.4 Otestovanie a vyhodnotenie systému estimácie výšky

V tejto podkapitole popíšem postup akým otestujem oba senzory, ich spojenie s EKF a zhodnotím dosiahnuté výsledky.

5.4.1 Testovanie senzorov

Otestovanie senzorov prebehne vo viacerých častiach, pričom ma bude zaujímať nalogovaná hodnota `current_distance`.

1. Sensory pevne umiestnim, s tým že budú merať horizontálnu vzdialenosť v rozsahu 0.3 – 12 m od tienidla, ktoré predstavuje plastová doska o ploche zhruba 2 m². S plastovou doskou budem hýbať pozdĺž natiahnutého pásma, pričom ako referenčnú vzdialenosť budem považovať hodnotu odčítanú z pásma.
2. Sensory otestujem na 3 rôznych povrchoch, a to tráva, betón a asfalt. Porovnam dáta zo senzorov v rôznych výškach nad Zemou za účelom otestovania, ako daný senzor meria na danom povrchu, meranie prebiehalo v rozsahu výšok 0.3 – 3.9 m v prípade trávy a asfaltu. Na betóne boli senzory otestované v rozsahu 0.3 – 7.8 m. Z dôvodu krátkeho rebríku, pomocou ktorého prebehlo meranie, je v prípade trávy a asfaltu malá maximálna referenčná vzdialenosť. Ako referenčná výška bola hodnota odčítaná z pásma.
3. Sensory otestujem aj v rôznych rýchlostiach a na dvoch povrchoch, a to asfalte a tráve v zafixovanej výške. V prípade asfaltu otestujem senzory v

rýchlostiach 20 km h^{-1} , 35 km h^{-1} , 60 km h^{-1} , 75 km h^{-1} . Na trávnom povrchu budú senzory otestované v rýchlostiach 20 km h^{-1} , 35 km h^{-1} .

5.4.2 Testovanie estimácie výšky

Po otestovaní senzorov prebehne testovanie estimátoru, počas letu. Na to je potrebné nastaviť niekoľko parametrov. Pred odletom bude potrebné odmerať a nastaviť parametre `EKF2_RNG_POS_X`, `EKF2_RNG_POS_Y`, `EKF2_RNG_POS_Z` a `EKF2_RNG_PITCH`, ktoré definujú polohu senzoru voči polohe ťažiska. Taktiež je potrebné nastaviť parametre pre fúziu senzorov `EKF2_RNG_A_HMAX=10`, kde uvažujem maximálnu povolenú hodnotu, teda 10 m. Nastaví sa parameter `EKF2_RNG_A_VMAX`. Avšak povolené hodnoty tohto parametru sú $0 - 2 \text{ m s}^{-1}$. Keďže vírnik nelietá takou malou rýchlosťou nastaví sa na testovací let hodnota rýchlosti vylyvajúca z porovnania senzorov počas jazdy autom. Predpokladám, že parameter slúži pre estimáciu hlavne na drony, aby sa zamedzilo šumu zo senzorov, keď sa rýchlo mení povrch. Nastaví sa parameter `EKF2_RNG_AID=1`, kde bude primárny senzor nastavený barometer, ktorému zodpovedá `EKF2_HGT_MODE=0`.

1. Vírnik bude letieť v konštantnej výške a krúžiť v režime hold. Účelom bude otestovať kompenzáciu náklonov implementovaným estimátorom.
2. Vírnik preletí ovály. Pri rovnej časti sa bude meniť výška $3 - 10 \text{ m}$. Zaujímať ma bude aj kompenzácia náklonov pri prechode z rovnej časti do oblúku a v rôznych výškach.
3. Odletí sa viacero priblížení na pristátie. Ich účelom bude otestovať estimátor pri prechode do výšok, v ktorých sú schopné merať senzory.

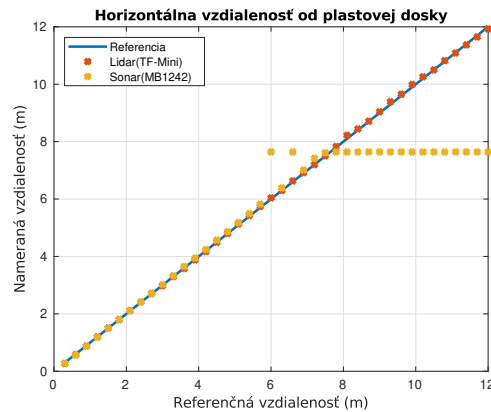
5.4.3 Vyhodnotenie otestovania senzorov

Vyhodnotenie nameraných dát.

1. Po odmeraní na plastovej doske som dostal závislosť zobrazenú na obrázku 5.6. Z grafu pozorujem, saturáciu ultrazvukového senzoru na hodnote $7,65 \text{ m}$. Čo sa týka ultrazvukového senzoru, vrámci jeho rozsahu na nameraných dátach pozorujem dve hrubé chyby, ktoré sú spôsobené tým, že senzor dosku nezachytil a dostal sa do saturácie. Chyba sa postupne so zväčšujúcou vzdialenosťou mierne zväčšuje, čo zodpovedá javu popísanému v dokumentácii, že malé objekty sa zdajú byť senzoru ďalej, ako reálne sú [16]. Priemerná absolútna hodnota chýb meraní ultrazvukovým senzorom vrámci jeho rozsahu je $6,3 \text{ cm}$. Maximálna absolútna chyba senzoru je 22 cm pri referenčnej vzdialenosti $7,2 \text{ m}$.

Pri lidare je priemerná absolútna chyba merania $2,8 \text{ cm}$. Maximálna absolútna chyba merania je 12 cm , a to pre vzdialenosť $8,1 \text{ m}$. Podľa užívateľského manuálu pre senzor, vrámci rozsahu $0,1 - 6 \text{ m}$ je maximálna absolútna chyba senzoru 6 cm [14]. To sa meraním potvrdilo. Pre

vzdialenosti 6 – 12 m je chyba 1% z merania. Pri každom meraní z daného rozsahu je meranie pod chybou deklarovanou výrobcom okrem merania vo vzdialenosti 8,1 m, ktoré bolo podľa dát nepresne odmerané. Nepresnosť mohla byť spôsobená vetrom, ktorý hýbal s doskou počas merania.



Obrázok 5.6: Graf odmeraných hodnôt na plastovej doske

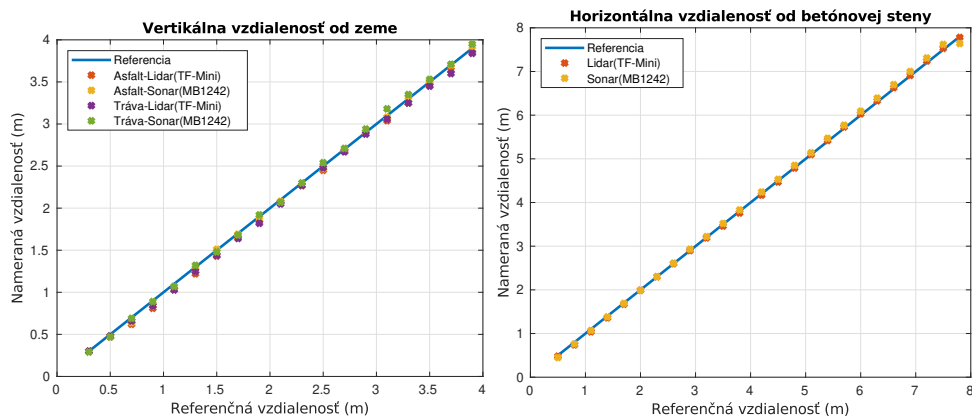
2. Namerané dáta som vložil do obrázkov 5.7a, 5.7b. Tentokrát boli senzory otestované v menšom rozsahu výšok, kvôli výškovej limitácii rebríku. Toto meranie neslúžilo na určenie presnosti merania, ale na prípadné zistenie rozdielov medzi senzormi, spôsobenými povrchom. Z pozorovaných priebehov nevidím rozdiel v meraniach medzi trávou a asfaltom.

V prípade ultrazvukového senzoru som po odčítaní nameraných hodnôt pre asfalt od hodnôt pre trávku získal priemernú chybu 2,7 cm. Tento rozdiel nepovažujem za dostatočne veľký a môže byť spôsobený pohybmi ruky, pásma, náklonom senzorov. Rozdiel hodnôt nezávisí na zväčšujúcej/zmenšujúcej sa výške merania.

Pri meraní výšky lidarom nepozorujem z nameraných dát rozdiel medzi meraním výšky nad trávou a asfaltom v malej výške. Po odčítaní nameraných hodnôt pre asfalt od hodnôt pre trávku, som získal priemerný rozdiel 0,1 cm. Podobne ako v prípade sonaru konštatujem, že v malej výške nie je rozdiel medzi meraním nad trávou a asfaltom. Rozdiel hodnôt nezávisí na zväčšujúcej/zmenšujúcej sa výške merania.

V prípade betónu som meral horizontálnu vzdialenosť od múru, nie vertikálnu, a pri iných referenčných vzdialenostiach, preto nie je možné porovnať namerané hodnoty s trávou a asfaltom. Pre lidar som vypočítal priemernú absolútnu chybu od referencie 5,4 cm a pre sonar 2,5 cm, pričom tieto hodnoty sú blízke tým z prvého bodu porovnania.

3. Dáta boli odmerané z auta, pričom držiak senzorov bol vo vzdialenosti zhruba 10 cm od karosérie auta. Senzory boli zafixované výške 89 cm a priebeh sa logoval. Rýchlosť auta počas merania, bola udržiavaná pomocou tempomatu.



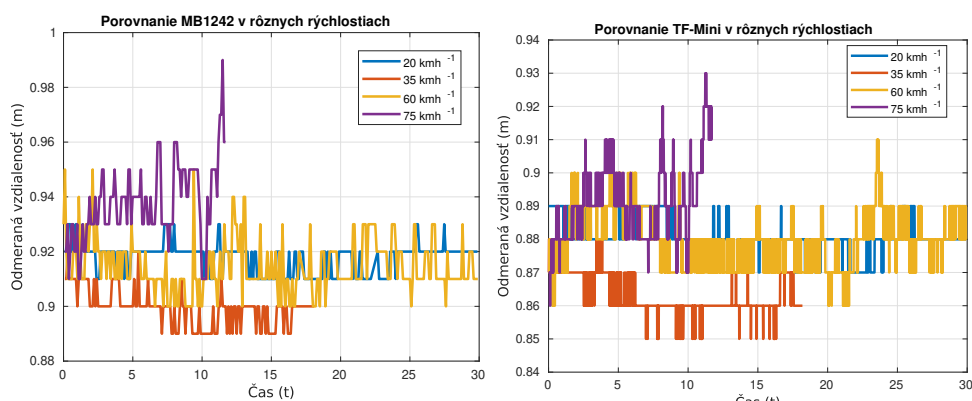
(a) : Graf odmeraných hodnôt výšky nad trávou a asfaltom

(b) : Graf odmeraných hodnôt od betónovej steny

Obrázok 5.7: Namerané dáta pri rôznych povrchoch

V prípade sonarových a lidarových meraní na asfalte sa rozptyl nameraných hodnôt v závislosti na zvyšujúcej sa rýchlosti zväčšoval 5.8, pričom zmena hodnôt nie je výrazná a prijateľná. Výrazný rozdiel v hodnote pri rýchlosti 75 km h^{-1} je spôsobený pravdepodobne kmitaním držiacu. Z dôvodu bezpečnosti a ochrany senzorov pred spadnutím a poškodením, nie sú časy meraní pri rôznych rýchlostiach rovnaké.

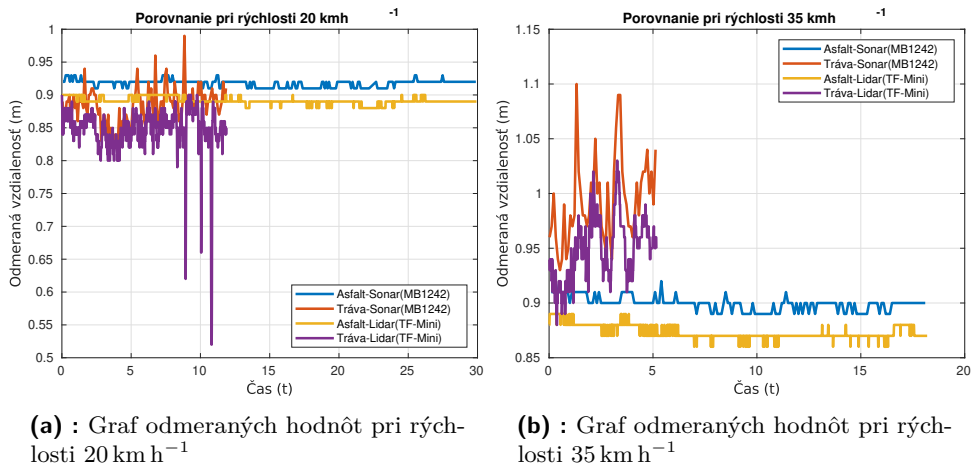
V prípade meraní na tráve pozorujem podobné priebehy sigálov, pričom sonar má väčší rozptyl meraní. Avšak lidar trikrát zmeral výšku, ktorá sa líši o viac ako 10 cm, pravdepodobne kvôli vlastnosti testovaného povrchu. Pri zvýšení rýchlosti o 15 km h^{-1} sa zväčšil rozptyl meraní podobne ako na asfalte. Porovnanie senzorov na tráve a asfalte sa nachádza na obrázku 5.9



(a) : Graf odmeraných hodnôt senzoru MB1242

(b) : Graf odmeraných hodnôt senzoru TF-Mini

Obrázok 5.8: Namerané dáta pri rôznych rýchlostiach



Obrázok 5.9: Porovnanie senzorov pri rôznych povrchoch a rýchlostiach

5.4.4 Vyhodnotenie otestovania estimácie výšky

Otestovanie systému estimácie výšky prebehlo 9.5.2022. Pred testovacím letom sa nastavil parameter `EKF2_RNG_A_VMAX` na 30 m s^{-1} napriek obmedzeniu, na základe toho, že pri meraní senzorov z auta nebol spozorovaný vplyv rýchlosti na meranie senzorov. Nanešťastie sa nepodarilo spraviť ani jeden z požadovaných manévrov kvôli pádu vírniku. Z logu je zachytený iba štart a časť letu pred pádom. Z letových logov som spozoroval, že senzor MB1242 nemeral správne. Ako vidno na obrázku 5.10 senzor bol po väčšinu letu saturovaný, z dôvodu veľkej výšky letu. Avšak v oblasti malej výšky nad povrchom, detekovanej senzorom TF-Mini mal aj senzor MB1242 merať. Zároveň počas letu senzor zapisoval do logu údaje o nesprávnom odmeraní dát.

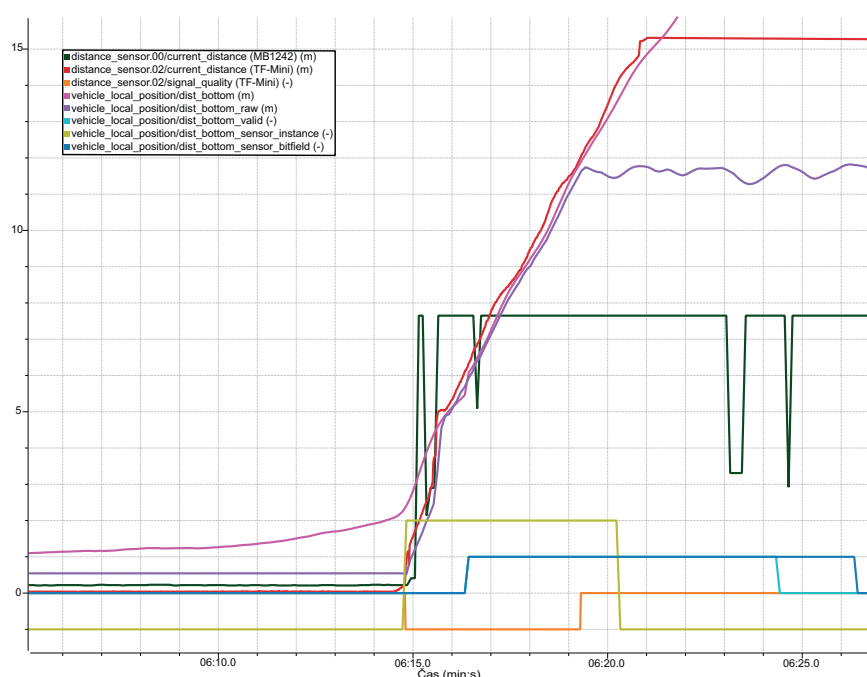
Nesprávne meranie senzoru MB1242 môže byť spôsobené umiestnením senzoru na vírniku. Pri montáži a testovaní senzorov som spozoroval rovnaký priebeh ako na obrázku 5.11, kde TF-Mini meralo správne a MB1242 šumel a bol často saturovaný. Tento jav bol spôsobený umiestnením senzorov, keďže po premiestnení senzorov nižšie meralo správne aj MB1242, senzoru mohol prekážať podvozok alebo baterka. Pri lete mohlo dôjsť k oscilácii podvozkovej nohy a tým pádom bol rušený ultrazvukový senzor. Pred nasledujúcim testovacím letom sa senzory umiestnia nižšie na baterku.

Okrem problémov so senzorom MB1242 fungoval systém merania výšky správne. Štart letu prebiehal z platformy na streche automobilu. Počas rozbehu auta je možné pozorovať na základe `dist_bottom_sensor_bitfield`, že bol ako senzor výšky pre fúziu vybratý barometer, ktorému zodpovedá 0. To je vysvetlené tým, že fúzia sa zapne až potom keď platí `in_air == True`. Počas štartu je možné pozorovať ako so zvyšujúcou sa rýchlosťou auta stúpa hodnota `dist_bottom`, pretože barometer zaznamenal nižší tlak.

Po zapnutí fúzie v čase 6:06,4 pozorujem, že hodnoty `dist_bottom` a `dist_bottom_raw` sú menšie než odmeraná výška senzorom TF-Mini, keďže

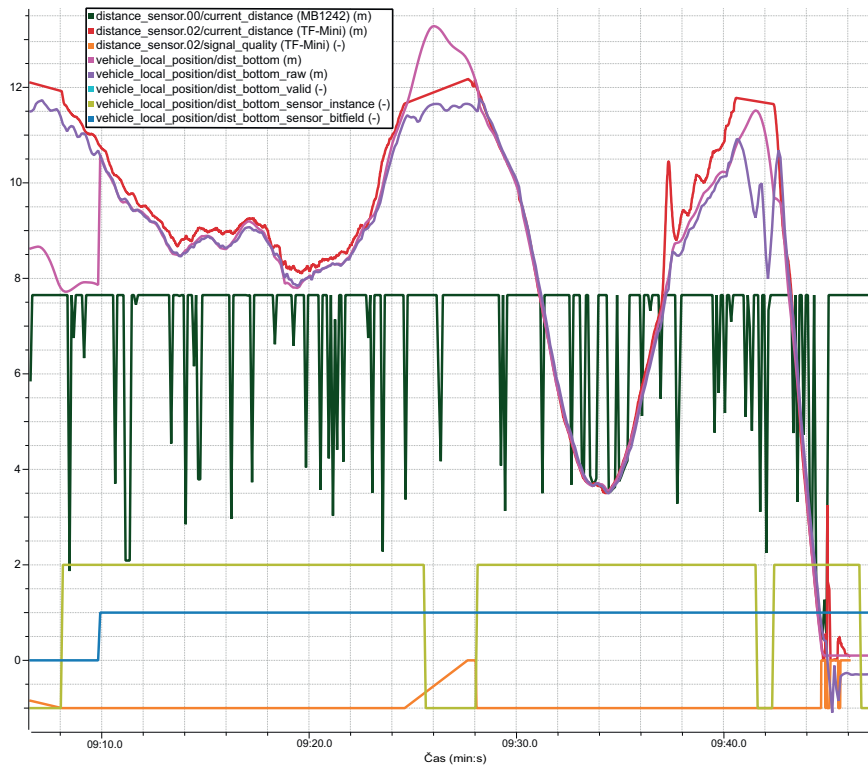
vírník v stúpaní bol naklonený. Selektor senzorov počas celého vzletu poskytoval hodnotu `dist_bottom_sensor_instance = 2`, čo zodpovedá lidarom TF-Mini, keďže selektor primárne vyberá dáta zo senzoru s najväčším rozsahom, poskytujúcim validné dáta. Z logu pozorujem, že pri vzlete v prvom pokuse informácia o validite výšky z estimátora `dist_bottom_valid` je výška validná aj potom, čo `signal_quality` je nulové. Zároveň informácia o používanom senzore na meranie výšky sa nezmení okamžite so signálom o kvalite dát zo senzoru. Pri indikácii použitého aktuálneho senzoru vzdialenosti je daný jav pozorovaný taktiež.

Hodnota `dist_bottom_raw` zodpovedá hodnote výšky zo senzorov, ktorá je po dosiahnutí maximálnej vzdialenosti pre senzor saturovaná medzami z ovládaču senzoru a prenásobená kompenzačným členom z rotačnej matice.



Obrázok 5.10: Ukážka z logu počas prvého testovacieho vzletu vírniku

Pri prvom lete bol pokus o nízky prelet počas ktorého sa vírník dostal do vývrtky a spadol. Časť toho letu na obrázku 5.11 sa dá použiť na zhodnotenie systému estimácie výšky. Z tejto časti letu vidno, že MB1242 nefungoval správne. Zároveň v čase 9:08 vidno, ako sa z barometru prepne senzor použitý na fúziu na senzor vzdialenosti prechod `dist_bottom_sensor_instance` z -1 (žiaden senzor vzdialenosti) na 2 (TF-Mini). Vidno aj prechod `dist_bottom` na hodnoty `dist_bottom_raw`, ktoré kompenzujú hodnotu zo senzoru celý čas. Po prekročení výšky, kedy je nameraná hodnota validná si EKF sám estimoval vzdialenosť od zeme 9:26 až 9:28, pričom podľa EKF sa použil senzor vzdialenosti, avšak senzor vzdialenosti použitý nebol. Okolo času zhruba 9:38 si EKF poradil so zakmitaním `dist_bottom` avšak hodnoty z 9:42 nemajú praktický zmysel, keďže vtedy bol už vírník vo vývrтке.

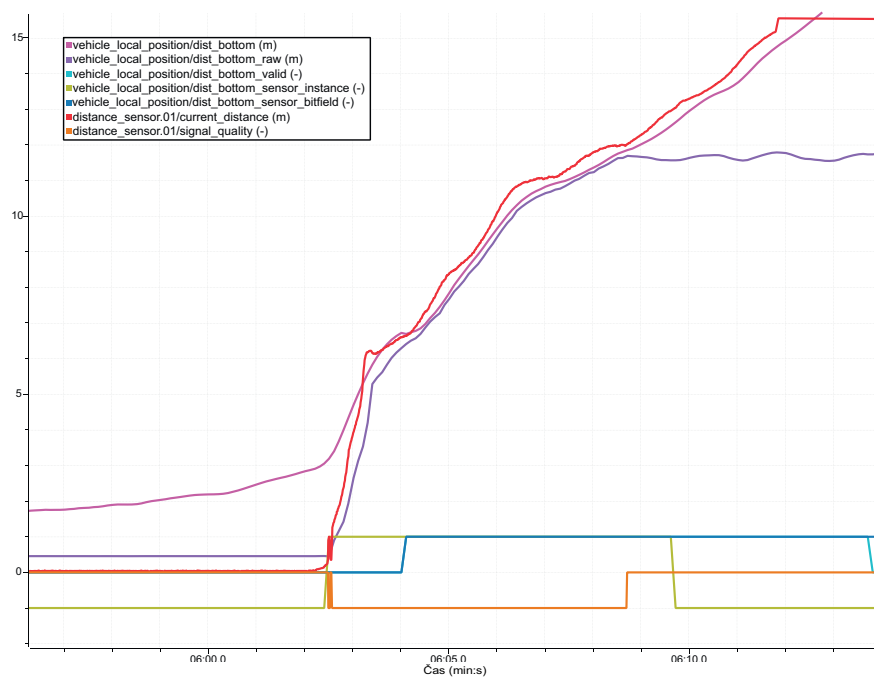


Obrázok 5.11: Ukážka z logu pred pádom vírniku

Po páde vírniku sa vymenili poškodené diely a vyskúšal sa druhý štart za účelom otestovania senzorov a estimátoru. Kvôli chybovým hláškam senzoru MB1242 a obáv zo zahľtenia zápisu na SDkartu chybovými hláškami (v prechádzajúcom lete bol počas letu problém s logovaním) sa odpojil MB1242 a letelo sa iba s pripojeným senzorom TF-Mini. Pravdepodobne kvôli uvoľneným/prasknutým dielom z predchádzajúceho pádu bol problém vírnik ovládať a vírnik opätovne spadol, takže sa nepodarilo odletieť manévrou na overenie estimácie výšky. Z dostupných dát mám k dispozícii na overenie estimátoru iba štart 5.12 ktorý je podobný tomu vyššie opísanému.

Z nalogovaných dát usudzujem, že nie je konzistentná v ovládači TF-Mini vzhľadom na to, že senzor posiela dáta o vzdialenosti väčšie než je 12 m aj keď by nemal. Zároveň, by ovládače senzorov mali poskytovať lepšie údaje o kvalite dát. Na základe logov a doteraz nameraných údajov hodnota `signal_quality` rovná 0 znamená, že dáta nie sú validné a hodnota -1 znamená, že senzor nevie určiť informáciu o validite dát. Z nameraných údajov z predchádzajúcej časti venujúcej sa senzorom som si všimol, že keď senzor má kvalitu dát -1 tak meria správne. Napriek tomu, že EKF poskytuje údaje o validite odhadu výšky `dist_bottom_valid` nie je správne sa spoliehať iba na dáta `dist_bottom`.

Pri pristávaní vírniku navrhujem pri práci s výškovými dátami pracovať s viacerými údajmi, ktoré môžu poskytovať informáciu o validite estimovanej výšky. Pri pristávaní navrhujem kontrolovať `dist_bottom_valid`, `dist_bottom_sensor_instance` sa nerovná -1 a `dist_bottom_sensor_bitfield`



Obrázok 5.12: Ukážka z logu počas testovacieho vzletu vírníku

sa rovná 1. Táto kombinácia by mala zaistiť, že je validná vzdialenosť od zeme, meria sa so sensorom vzdialenosti, ktorý meria vo svojom rozsahu a že na fúziu dát sa používa sensor vzdialenosti.

Kapitola 6

Modelovanie pristávacieho manévru

V tejto kapitole sa zameriam na matematické modelovanie pristávacieho manévru, ktorý bude pozostávať z priblíženia a samotného pristávacieho manévru. Počas priblíženia a pristávacieho manévru bude namodelovaný systém predstavovať vírnik spoločne s už naimplementovaným regulátorom polohy, viď. podkapitola 3.2.1. Teda vstup systému s autopilotom bude požadovaný náklon vírniku.

6.1 Modelovanie vírniku

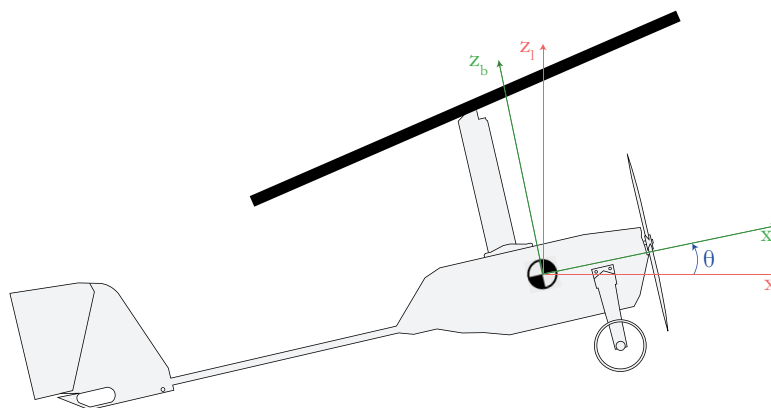
Účelom modelovania vírniku bude získať diferenciálne rovnice, popisujúce dynamiku systému. Na modelovanie použijem Euler Lagrange-ov prístup so zovšeobecnenými silami a momentmi za účelom získania pohybových rovníc 6.1, kde $\mathcal{L}(\mathbf{q})$, je Lagrangeova funkcia, definovaná ako $\mathcal{L}(\mathbf{q}) = \mathcal{T}(\mathbf{q}) - \mathcal{V}(\mathbf{q})$. $\mathcal{T}(\mathbf{q})$ je kinetická energia a $\mathcal{V}(\mathbf{q})$ je potenciálna energia. \mathbf{Q} je zovšeobecný vektor síl a momentov a \mathbf{q} je vektor nezávislých súradníc v tomto prípade podľa obrázku 6.2.

$$\frac{d}{dt} \frac{\partial \mathcal{L}}{\partial \dot{\mathbf{q}}} - \frac{\partial \mathcal{L}}{\partial \mathbf{q}} = \mathbf{Q} \quad (6.1)$$

$$\mathbf{q} = \begin{pmatrix} x \\ z \\ \theta \\ \psi \end{pmatrix} \quad (6.2)$$

Pri modelovaní vírniku zavediem 2 súradnicové systémy. Predpokladám, že vírnik sa pohybuje v rovine. Teda, čo sa translačného pohybu týka má dva stupne voľnosti. V rotačnom pohybe má vírnik iba 1 stupeň voľnosti. Týmto predpokladom sa mi modelovanie zjednoduší a budem pracovať s menším počtom nezávislých súradníc. Ďalší stupeň voľnosti popisujúci vírnik, je uhol natočenia rotoru. Jeden súradnicový systém označím ako lokálny. V tomto súradnicovom systéme budem premenné označovať pomocou písmena l . Druhý súradnicový systém označím pomocou písmena b . Tento súradnicový systém je pevne spojený s telom vírniku. Oba súradnicové systémy sú naznačené na obrázku 6.1. Vstupom systému bude riadiaci signál η_n pre servo ovládajúce pozdĺžny náklon rotoru. Uhol náklonu rotoru spôsobený servom označím ako

ϵ_s . Pri modelovaní zanedbávam bočný vietor a predpokladám vietor fúkajúci v smere, resp. protismere osi x_l .



Obrázok 6.1: Zobrazenie súradicových systémov

Modelovanie serva

Servo nakláňajúce rotor popíšem ako systém, ktorého vstupom je riadiaci signál pozdĺžneho náklonu rotoru η_m a jeho výstup je uhol pozdĺžneho náklonu rotoru, spôsobený servom ϵ_s .

Samotné modelárske servo je namodelované ako systém prvého rádu, pričom tento systém je popísaný diferenciálnou rovnicou [19]

$$0,04 \dot{x} + x = \eta_m \quad (6.3)$$

Riadiaci signál η_m pre servo v PX4, ale aj v simulátore Flightgear nadobúda hodnoty v intervale $< -1, 1 >$. Ako bolo spomenuté v 3.1

existuje lineárna transformácia medzi výstupom zo serva x a uhlom pozdĺžneho náklonu rotoru ϵ_s .

Kaskádovým zapojením modelu serva a lineárnej transformácie dostanem model serva, ktorý môžem substituovať v rovniach obsahujúcich ϵ_s .

6.1.1 Určenie kinetickej a potenciálnej energie

Podľa obrázku 6.2 vidím, že vírnik koná behom letu translačný a rotačný pohyb. Zároveň rotor vírniku rotuje a prispieva tým k celkovej kinetickej energii vírniku. Preto sa kinetická energia vírniku bude skladať z rotačnej a translačnej časti.

Translačnú kinetickú $\mathcal{T}_{\text{trans}}$ energiu vyjadrím nasledujúcim vzorcom.

$$\mathcal{T}_{\text{trans}} = \frac{1}{2} m (\dot{x}^2 + \dot{z}^2) \quad (6.4)$$

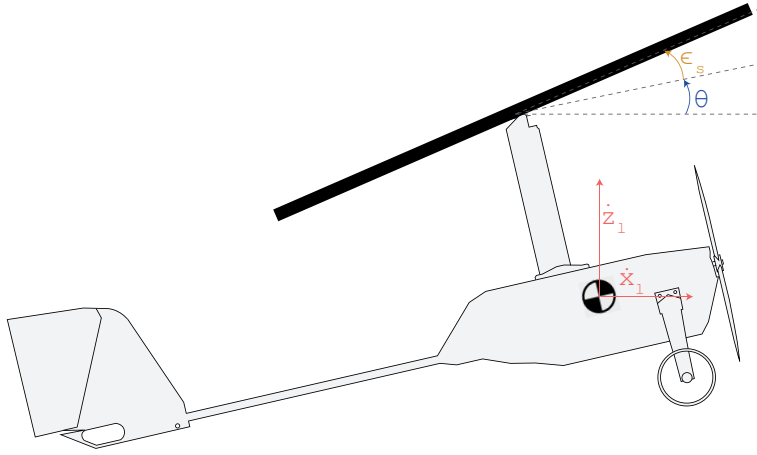
Pre rotačnú energiu vírniku \mathcal{T}_{rot} platí vzťah 6.5, kde J_{vehicle} je moment zotrvačnosti vírniku voči osi kolmej na x_l, z_l .

$$\mathcal{T}_{\text{rot}} = \frac{1}{2} J_{\text{vehicle}} \dot{\theta}^2 + \mathcal{T}_{\text{rotrotor}} \quad (6.5)$$

Kinetická energia rotora $\mathcal{T}_{\text{rotrotor}}$, pozostáva z viacerých rotačných energií a je vyjadrená vzťahom 6.6.

$$\mathcal{T}_{\text{rotrotor}} = \frac{1}{2} J_{r\psi} \dot{\psi}^2 + \frac{1}{2} J_{r\text{tilt}} \dot{\epsilon}_s^2 + \frac{1}{2} (J_{r\text{tilt}} + m_r d_r^2) \dot{\theta}^2 \quad (6.6)$$

V rovnici 6.6 zodpovedá prvý člen pravej strany rotačnej energii dvoch rotujúcich listov rotora s úhlovou rýchlosťou $\dot{\psi}$. Druhý člen vyjadruje rotačnú energiu rotora, pričom rotácia je spôsobená pohybmi serva nakláňajúceho rotor. Pri výpočte momentu zotrvačnosti $J_{r\text{tilt}}$ uvažujem rotujúci disk o priemere $2L$, kde L značí dĺžku listu rotora. Tretí člen na pravej strane je vyjadrením rotačnej kinetickej energie rotora, pričom rotácia je spôsobená náklonom celého vírniku, a teda zmenou uhlu θ . Pri tejto úvahe predpokladám, že tyč spájajúca rotor a telo vírniku je tuhé teleso. Vzdialenosť d_r určuje vzdialenosť stredu rotora od ťažiska vírniku a m_r je hmotnosť rotora.



Obrázok 6.2: Náčrt situácie pristávajúceho vírniku

Potenciálnu energiu vírniku $\mathcal{V}(\mathbf{q})$ získam ako

$$\mathcal{V} = m g z_l \quad (6.7)$$

kde m je hmotnosť vírniku a g gravitačné zrýchlenie.

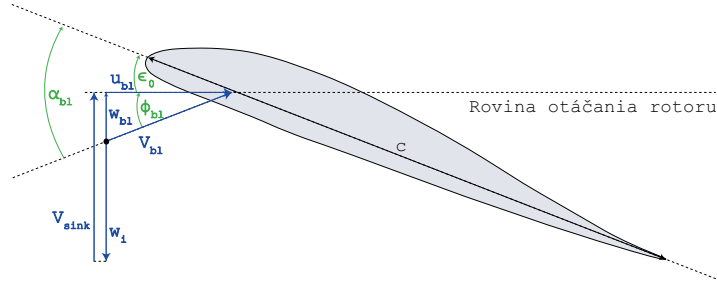
6.1.2 Aerodynamika rotora

Najprv popíšem samotnú lopatku rotora a uhol, ktorý zvierajú prúdnicu vzduchu a os lopatky. Podľa obrázku 6.3 vidno, že uhol prúdiaceho vzduchu, takzvaný uhol nábehu α_{bl} závisí na viacerých veličinách. Uhol ϵ_{bl} je uhol, ktorý zvierajú tetiva lopatky rotora s pozdĺžnou osou vírniku x_b . ϕ_{bl} je uhol zvierajúci vektor prúdiaceho vzduchu s rovinou rotácie rotora. Tento uhol sa vypočíta pomocou nasledujúceho vzťahu.

$$\phi_{bl} = \arctan \frac{w_{bl}}{u_{bl}} \quad (6.8)$$

Pri veľkých rýchlostiach rotoru je možné vo vzťahu 6.8 vziať do úvahy $u_{bl} \approx V_{bl}$. Preto pre malý uhol ϕ_{bl} platí nasledujúci vzťah.

$$\phi_{bl} = \arcsin \frac{w_{bl}}{V_{bl}} \approx \frac{w_{bl}}{V_{bl}} \quad (6.9)$$



Obrázok 6.3: Ukážka elementu listu rotoru z profilu

Rýchlosť w_{bl} je rýchlosť pretekajúceho vzduchu cez rovinu rotácie rotoru. Je vyjadrená pomocou jednoduchého vzťahu 6.11. V prípade vertikálneho klesavého letu je $V_{sink} = -\dot{z}$ rýchlosť klesania vírniku. Rýchlosť w_i je indukovaná rýchlosť rotoru. Všeobecne je indukovanú rýchlosť rotoru náročné určiť, vzhľadom na to, že nepoznáme presné parametre rotoru a závisí na veľkosti sily rotoru[20].

Na určenie zovšeobecnených síl a momentov pôsobiacich na vírnik je potrebné pochopiť aerodynamiku vírniku a sily, ktoré vznikajú pri rotácii rotoru. Použil som metódu BET(Blade element theory) [7]. Lopatky rotoru touto metódou budem analyzovať v stave vertikálneho klesania. Rovnice získané pre rotor vo vertikálnom klesavom lete použijem, pre vírnik letiaci v klesavom doprednom lete, v priblížení.

■ Vertikálne klesanie

V prvom prípade konštatujem, že na danom elemente lopatky rotoru je rýchlosť u_{bl} konštantná, nemení sa s časom na uhle natočenia rotoru ψ a vypočítam ju podľa vzorca 6.10.

$$u_{bl} = \dot{\psi} r_{bl} \quad (6.10)$$

$$w_{bl} = V_{sink} - w_i \quad (6.11)$$

Podľa [7] boli dopočítané aerodynamické sily vztlaku dL a odporu dD na element lopatky rotoru s plochou $dS_{bl} = c dy$ podľa rovníc 6.12 a 6.13. Tieto dve sily pôsobia v aerodynamickom systéme. Odpor dD pôsobí v smere vektora rýchlosti prúdiaceho vzduchu V_{bl} a vztlak dL je kolmý na odpor [5].

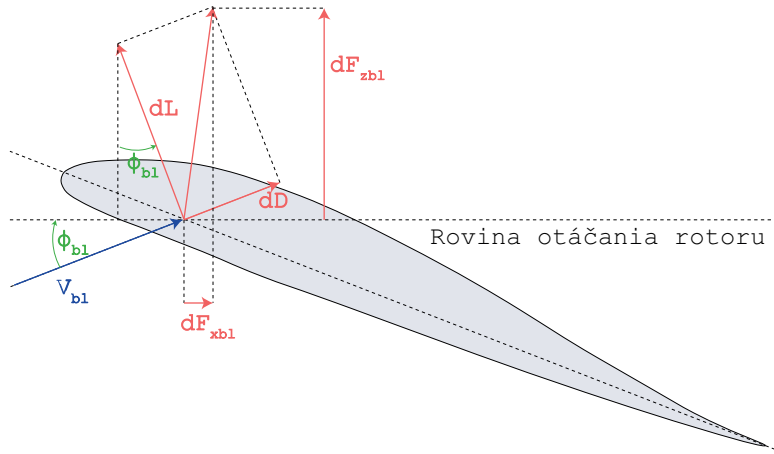
$$dL = \frac{1}{2} \rho V_{bl}^2 C_{L_{bl}} c dy \quad (6.12)$$

$$dD = \frac{1}{2} \rho V_{bl}^2 C_{D_{bl}} c dy \quad (6.13)$$

kde c je dĺžka tetivy listu rotora, ρ je hustota vzduchu, $C_{L_{bl}}$ je koeficient vzlaku a $C_{D_{bl}}$ je koeficient aerodynamického odporu. Po transformácii oboch elementov síl vid. obrázok 6.4 som dostal nasledujúce vzťahy.

$$dF_{x_{bl}} = dL \sin \phi_{bl} - dD \cos(\phi_{bl}) \quad (6.14)$$

$$dF_{z_{bl}} = dL \cos \phi_{bl} + dD \sin(\phi_{bl}) \quad (6.15)$$



Obrázok 6.4: Sily pôsobiace na element listu rotoru

Pre analytické vyriešenie rovníc 6.14,6.15 je potrebné spraviť aproximácie a zjednodušenia podľa [7].

- Uhol ϕ_{bl} je považovaný za malý v oblasti, ďalej od stredu otáčania rotora. Teda $\cos \phi_{bl} \approx 1$ a $\sin \phi_{bl} \approx \phi_{bl}$. Za predpokladu, že v oblasti blízko pri osi rotácie rotora (uhol ϕ_{bl} je veľký) sú dopredné sily dF_x zanedbateľne malé, kvôli malej rýchlosti V_{bl} . Tým pádom sa mi nepresnosť v silách $dF_{x_{bl}}$ a $dF_{z_{bl}}$ spôsobená aproximáciou členov $\cos \phi_{bl}$ a $\sin \phi_{bl}$ neprejaví.
- Rýchlosť V_{bl} , kvôli malému uhlu ϕ_{bl} vid. obrázok 6.3 je aproximovaná $V_{bl} \approx u_{bl}$.

Po aproximácii som dostal nasledujúce rovnice pre rotor v prvom vyššie spomenutom stave, kde sily od oboch lopatiek rotora sú osovo symetrické podľa osi rotácie rotora.

$$dF_r = N_b (dL + dD \phi_{bl}) \quad (6.16)$$

$$dM_r = N_b dF_{x_{bl}} y = N_b (dL \phi_{bl} - dD) y \quad (6.17)$$

Rovnica 6.17 zodpovedá momentu sily, ktorým je rotor roztáčaný a rovnica 6.16 vyjadruje silu od rotora, ktorá pôsobí v kolmom smere na os otáčania rotora.

Pred integrovaním rovníc 6.16, 6.17 je potrebné ešte určiť hodnotu koeficientov $C_{L_{bl}}$ a $C_{D_{bl}}$, ktoré sú závislé na uhle nábehu α_{bl} . Vzťah popisujúci dané závislosti je nelineárny a značne závisí od profilu listu a meraní v aerodynamickom tuneli a bol zlinearizovaný podľa [20].

$$\alpha_{bl} = \phi_{bl} + \epsilon_0 \quad (6.18)$$

$$C_{L_{bl}} \approx C_{L_{bl0}} + C_{L_{bl\alpha}} \alpha_{bl} \quad (6.19)$$

Pre výpočet $C_{D_{bl}}$ platí podľa [20] nasledujúci vzťah

$$C_{D_{bl}} \approx \frac{C_{L_{bl}}}{n_{Gl}} \quad (6.20)$$

v ktorom n_{Gl} predstavuje kĺzavosť lopatky.

Po dosadení do rovníc 6.16, 6.17, ich úprave a následnej integrácii pozdĺž listu rotora som dostal rovnice pre silu od rotora F_r a moment roztáčajúci rotor M_r v závislosti na uhlovej rýchlosti rotácie rotora $\dot{\psi}$ a rýchlosti vzduchu kolmo prúdiaceho na rotor V_{sink} .

$$F_r(V_{\text{sink}}, \dot{\psi}) = c_1(w_i - V_{\text{sink}})^2 + c_2\dot{\psi}(w_i - V_{\text{sink}}) + c_3\dot{\psi}^2 \quad (6.21)$$

$$M_r(V_{\text{sink}}, \dot{\psi}) = c_4(w_i - V_{\text{sink}})^2 + c_5\dot{\psi}^2 + c_6\dot{\psi}(w_i - V_{\text{sink}}) \quad (6.22)$$

Pričom platia vzťahy pre výpočet koeficientov 6.23 - 6.28

$$c_1 = \frac{N_b c \rho (R - r_0) C_{L_{bl\alpha}}}{2 n_{Gl}} \quad (6.23)$$

$$c_2 = -\frac{N_b c \rho (R^2 - r_0^2) (C_{L_{bl0}} + C_{L_{bl\alpha}} \epsilon_0 + C_{L_{bl\alpha}} n_{Gl})}{4 n_{Gl}} \quad (6.24)$$

$$c_3 = \frac{N_b c \rho (R^3 - r_0^3) (C_{L_{bl0}} + C_{L_{bl\alpha}} \epsilon_0)}{6} \quad (6.25)$$

$$c_4 = \frac{C_{L_{bl\alpha}} N_b c \rho (R^2 - r_0^2)}{4} \quad (6.26)$$

$$c_5 = -\frac{N_b c \rho (R^4 - r_0^4) (C_{L_{bl0}} + C_{L_{bl\alpha}} \epsilon_0)}{8 n_{Gl}} \quad (6.27)$$

$$c_6 = -\frac{N_b c \rho (R^3 - r_0^3) (C_{L_{bl0}} n_{Gl} - C_{L_{bl\alpha}} + C_{L_{bl\alpha}} \epsilon_0 n_{Gl})}{6 n_{Gl}} \quad (6.28)$$

■ Indukovaná rýchlosť

Všeobecne je náročné správne určiť veľkosť indukovanej rýchlosti rotoru, vzhľadom na to, že nepoznáme presné parametre rotoru a závisí na veľkosti sily rotoru 6.21 a sila rotoru závisí na indukovanej rýchlosti w_i [19].

Závislosť indukovanej rýchlosti w_i na sile od rotoru F_r je popísaná v rovnici 6.29 [20].

$$w_i = K_{w_i} \sqrt{\frac{F_r}{\rho \pi R^2}} \quad (6.29)$$

Problém v tomto prípade je, že pre modelovanie vírniku v priblížení potrebujem diferenciálnu rovnicu na výpočet w_i , ale zároveň na w_i je závislá sila rotoru F_r . Preto na výpočet rýchlosti w_i použijem približný linearizovaný vzťah.

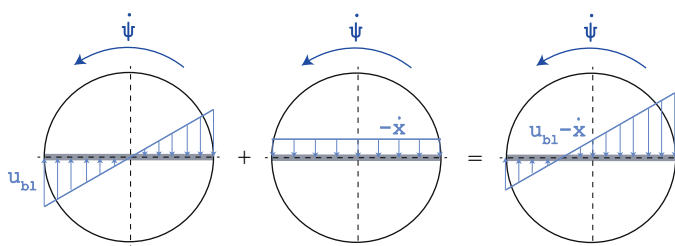
Podľa [20] a simulácie rotoru so závažím vo vertikálnom klesavom lete 6.3.1 je známe to, že pri danom scenári je indukovaná rýchlosť $w_i = 6,5 \text{ m s}^{-1}$. Po ustálení prechodových dejov som získal hodnoty $V_{\text{sink}} = 7,8 \text{ m s}^{-1}$ a $\dot{\psi} = 33,6 \text{ rad s}^{-1}$. Dosadením 6.21 do rovnice 6.29 je indukovaná rýchlosť funkciou uhlovej rýchlosti otáčania sa rotoru $\dot{\psi}$ a rýchlosti vzduchu V_{sink} . Rovnicu získanú dosadením aproximujem Taylorovým polynómom druhého rádu pre funkciu dvoch premenných. Po zderivovaní a zjednodušení je indukovaná rýchlosť vyjadrená ako

$$w_i \approx k_0 V_{\text{sink}}^2 + k_1 \dot{\psi}^2 + k_2 V_{\text{sink}} \dot{\psi} + k_3 V_{\text{sink}} + k_4 \dot{\psi} + k_5 \quad (6.30)$$

Analyticky vyjadrené koeficienty sa nachádzajú v priložených Matlabovských súboroch, kvôli ich zložitosti.

■ Dopredné klesanie

Pri doprednom lete s rýchlosťou \dot{x}_l nie sú oba listy rotoru rovnomerné zatažené. Nerovnomerné zataženie je spôsobené tým, že sa k lineárne rastúcej rýchlosti u_{bl} pozdĺž listu rotoru pričíta rovnaká zložka na oboch listoch spôsobená dopredným letom. To má za následok nerovnomerné rozloženie aerodynamických síl a následný vznik momentu ktorý nakláňa vírnik do strany 6.5. Tento problém sa čiastočne rieši naklápaním listov rotoru o uhol β takzvaný flapping, ktorý závisí na veľkosti radiálnej rýchlosti a uhle natočenia rotoru ψ [7].



Obrázok 6.5: Rotor v doprednom lete

Jav zobrazený na obrázku 6.5, je oscilácia výslednej sily rotoru F_r , ktorá závisí na uhle natočenia rotoru. Na uhle natočenia závisí aj sila, ktorá brzdí rotor v doprednom pohybe [20].

V prípade dopredného letu vírniku pre účel priblíženia sú zanedbané vyššie spomenuté vplyvy, ktoré by spôsobili skomplikovanie výpočtu[21]. Napríklad zavedenie nelinearit v podobe $\sin \psi$, $\cos \psi$ a linearizácia týchto funkcií pre

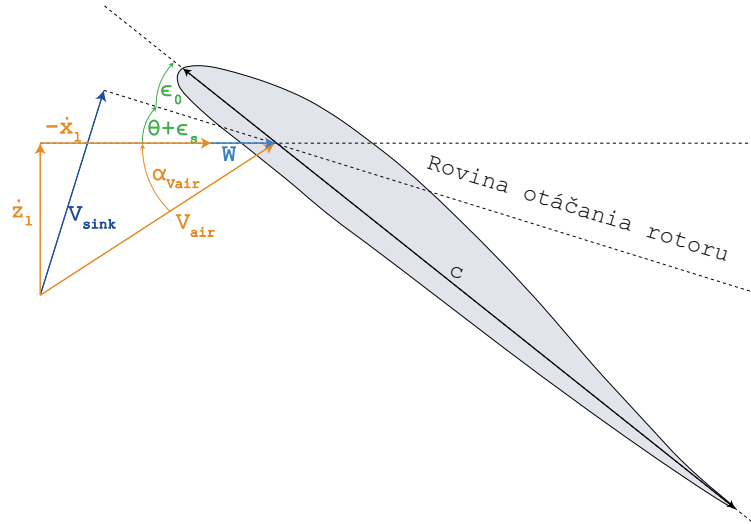
reguláciu nepripadá do úvahy, keďže na to, aby vírnik letel musí sa rotor otáčať. Analytické riešenie rovníc by bolo tiež náročnejšie.

V tomto prípade uvažujem, že uhol nábehu α_{bl} závisí aj na uhle náklonu celého vírniku θ a náklonu rotoru spôsobeného nakláňajúcim servom ϵ_s . Pre zjednodušenie uvažujem, že rotor klesá podobne ako v prípade 6.1.2. Pričom rýchlosti V_{sink} bude zodpovedať pretransformovaný vektor rýchlosti vírniku s pripočítaným vetrom W , ktorý je ako poruchový signál a fúka výlučne v osi x_l , pričom jeho kladný smer som zaviedol v proti smere osi x_l .

Uhol, ktorý zvierá vektor rýchlosti s rovinou otáčania rotoru označím ako uhol nábehu α . Uhol $\alpha_{V_{\text{air}}}$ je uhol, ktorý zvierá vektor prúdiaceho vzduchu V_{air} s osou x_l . Celá situácia je nakreslená na obrázku 6.6

$$\alpha = \theta + \epsilon_s + \alpha_{V_{\text{air}}} \quad (6.31)$$

$$\alpha_{V_{\text{air}}} = \arctan \frac{\dot{z}}{-\dot{x} + W} \quad (6.32)$$



Obrázok 6.6: Element listu rotoru

Vektor rýchlosti vzduchu V_{air} vypočítam použitím Pytagorovej vety

$$V_{\text{air}} = \sqrt{\dot{z}^2 + (-\dot{x} + W)^2} \quad (6.33)$$

Projekciou V_{air} na os kolmú k rovine otáčania rotoru získam rýchlosť V_{sink} , tangenciálnu zložku rýchlosti V_{air} , paralelnú s rovinou otáčania rotoru, zanedbám [20].

$$V_{\text{sink}} = V_{\text{air}} \sin \alpha \quad (6.34)$$

Uhol ϕ_{bl} sa vypočíta podľa rovnice 6.8 a uhol α_{bl} podľa rovnice 6.18. Rýchlosť w_{bl} sa dopočíta použitím 6.11. Pre vírnik v klesavom a doprednom lete použijem rovnice 6.14, 6.15, popisujúce aerodynamické sily pôsobiace na element listu rotoru. [20].

Po dosadení do rovníc 6.16, 6.17, ich úprave a následnej integrácii pozdĺž listu rotora som dostal rovnice pre silu od rotora F_r a moment pôsobiaci na rotor M_r v závislosti na uhlovej rýchlosti rotácie rotora $\dot{\psi}$ a vertikálnej rýchlosti klesania vírniku \dot{z}_l , doprednej rýchlosti vírniku \dot{x}_l , uhle náklonu vírniku θ , uhle náklonu spôsobeným servom ϵ_s a rýchlosti vetru W .

6.1.3 Určenie síl a momentov

Na vírnik pôsobí viacero síl. Pri pristávacom manévri a priblížení bude vypnutý motor vírniku. Na základe toho sa zjednoduší matematický model vírniku, pretože neuvažujem ťahovú silu od motoru a moment, ktorý pôsobí.

Na vírnik pôsobí sila od rotoru F_r , ktorú som dopočítal v prechádzajúcej časti. Ďalšou pôsobiacou silou je odporová sila vzduchu F_{air} [20].

Na vírnik pôsobí aj gravitačná sila F_g , ktorú dostanem zderivovaním Lagrangeovej funkcie \mathcal{L} .

Silu rotoru si rozložím do lokálneho súradnicového systému. Keďže rovina rotácie rotora zvierá s osou x_l uhol $\theta + \epsilon_s$ tak vodorovná zložka sily F_r je vyjadrená ako

$$F_{r_{x_1}} = F_r \sin(\theta + \epsilon_s) \quad (6.35)$$

Vertikálnu zložku F_r vypočítam podľa

$$F_{r_{z_1}} = F_r \cos(\theta + \epsilon_s) \quad (6.36)$$

Odporovú silu od vzduchu obtekajúceho vírnik vyjadrím použitím Newtonovho vzorca [20]

$$F_{\text{air}} = \frac{1}{2} C_v \rho S_v V_{\text{air}}^2 \quad (6.37)$$

kde C_v je koeficient odporu vzduchu a S_v predstavuje plochu priečneho prierezu vírniku zo smeru, z ktorého prúdi vzduch. V prípade klesajúceho vírniku sa táto plocha bude meniť s uhlom θ no jej zmeny nebudú veľké, vzhľadom na to, že vírnik počas priblíženia nebude výrazne meniť orientáciu.

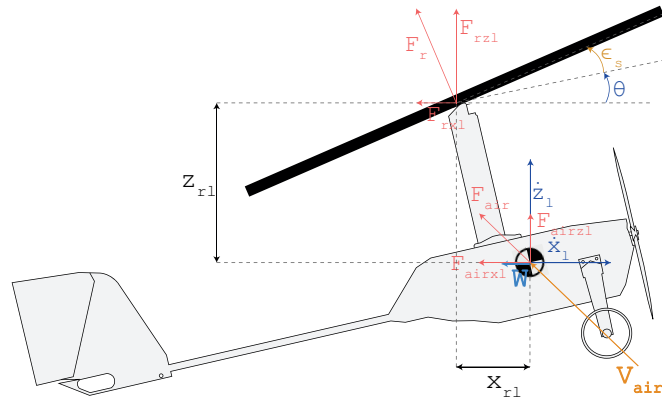
Silu F_{air} rozložím do lokálneho súradnicového systému, kde jej zložky dopočítam podobne ako v prípade sily F_r .

$$F_{\text{air}_{x_1}} = F_{\text{air}} \cos(\alpha_{V_{\text{air}}}) \quad (6.38)$$

$$F_{\text{air}_{z_1}} = F_{\text{air}} \sin(\alpha_{V_{\text{air}}}) \quad (6.39)$$

Na určenie momentov síl pôsobiacich na vírnik počas priblíženia je potrebné vyjadriť si vzdialenosti osi otáčania rotora a ťažiska vírniku v lokálnom súradnicovom systéme x_{r_1} , z_{r_1} pre jednoduchšie použitie. Z konštrukcie vírniku je známa táto vzdialenosť d_r a takisto aj vzdialenosti x_{r_b} , z_{r_b} , ktoré sú v súradnicovom systéme pevne spojenom s vírnikom konštanty, za predpokladu, že konštrukcia vírniku je tuhé teleso. Uhol, ktorý zvierá spojnice osi otáčania rotora s ťažiskom vírniku a pozdĺžnou osou vírniku označím γ . Následne je možné určiť vzdialenosti x_{r_1} , z_{r_1} pomocou nasledujúcich rovníc

$$x_{r_1} = d_r \cos(\gamma - \theta) \quad (6.40)$$



Obrázok 6.7: Vírník v doprednom klesavom lete

$$z_{r1} = d_r \sin(\gamma - \theta) \quad (6.41)$$

Na vírník pôsobia aj momenty, ktoré otáčajú vírník, následkom čoho sa mení uhol θ . Ako zjednodušenie predpokladám, že vírník sa otáča okolo svojho ťažiska. Výsledný moment M_v pôsobiaci na vírník vypočítam ako

$$M_v = -F_{r_{x1}} x_{r1} + F_{r_{z1}} z_{r1} \quad (6.42)$$

kde veličiny x_{r1} , z_{r1} predstavujú vzdialenosť rotoru od ťažiska vírníku v lokálnom súradnicovom systéme.

6.1.4 Nelinearizovaný model

Po zderivovaní Lagrangeovej funkcie a dosadení síl a momentov z predchádzajúcej sekcie na pravú stranu rovnice 6.1 som dostal nasledujúce diferenciálne rovnice popisujúce vírník v doprednom klesavom lete.

$$\frac{d}{dt} \frac{\delta \mathcal{L}}{\delta \dot{x}} = -F_{r_{x1}} - F_{air_{x1}} \quad (6.43)$$

$$\frac{d}{dt} \frac{\delta \mathcal{L}}{\delta \dot{z}} = F_{r_{z1}} + F_{air_{z1}} \quad (6.44)$$

$$\frac{d}{dt} \frac{\delta \mathcal{L}}{\delta \dot{\theta}} = -F_{r_{z1}} x_{r1} + F_{r_{x1}} z_{r1} \quad (6.45)$$

$$\frac{d}{dt} \frac{\delta \mathcal{L}}{\delta \dot{\psi}} = M_r \quad (6.46)$$

Vírník popíšem ôsmimi stavmi, ktoré sú reprezentované ako stavový vektor \mathbf{x} . Derivácie jednotlivých stavov som získal použitím rovníc 6.43 - 6.46.

$$\mathbf{x} = \left(x \quad \dot{x} \quad z \quad \dot{z} \quad \theta \quad \dot{\theta} \quad \psi \quad \dot{\psi} \right)^T \quad (6.47)$$

Pre derivácie stavov platia nasledujúce diferenciálne rovnice

$$\dot{x}_1 = x_2 \quad (6.48)$$

$$\dot{x}_2 = \frac{F_{rx_1} - F_{air_{x_1}}}{m} \quad (6.49)$$

$$\dot{x}_3 = x_4 \quad (6.50)$$

$$\dot{x}_4 = \frac{F_{rz_1} + F_{air_{z_1}}}{m} - g \quad (6.51)$$

$$\dot{x}_5 = x_6 \quad (6.52)$$

$$\dot{x}_6 = \frac{F_{rz_1} x_{r_1} + F_{rx_1} z_{r_1}}{m d_r^2 + J_{r_{tilt}} + J_{vehicle}} \quad (6.53)$$

$$\dot{x}_7 = x_8 \quad (6.54)$$

$$\dot{x}_8 = \frac{M_r}{J_r} \quad (6.55)$$

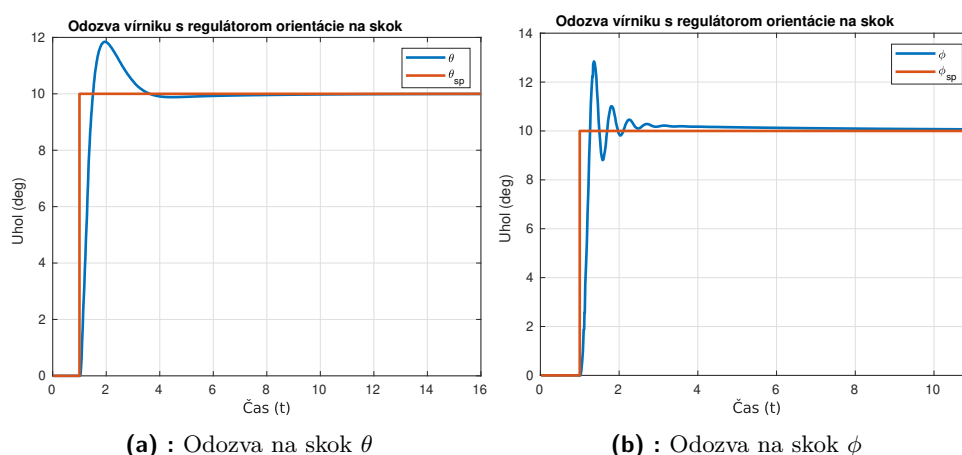
Model vírniku je implementovaný v prostredí Simulink za použitia S-funkcie. Jej výhoda spočíva v tom, že zmeny modelu sú jednoduchšie implementovateľné vzhľadom na to, že väčšina vzťahov je symbolicky dopyčovaná použitím MATLAB-u, model závisí na viacerých silách, ktoré sú v S-funkcii analyticky vyjadrené a model pôsobí kompaktnejšie a prehľadnejšie [19]. Servo som naimplementoval pomocou bloku prenosovej funkcie a zaradil ho pred model vírniku na vstup ϵ_s .

6.2 Model regulátoru

Podľa podkapitoly 3.2.1 a [11] som pre potreby simulácie pristávacieho manévru implementoval regulátor orientácie v Simulink-u vid. obrázok 3.3. Regulátor orientácie sa nachádza v súbore `fw_att_controller.slx` naimplementovaný ako subsystém. Konkrétne v implementácii tohoto regulátoru pre vírniky je vypnuté škálovanie podľa airspeed. Nad regulátorom orientácie som naimplementoval vyššiu slučku, ktorá nastavuje regulátoru pozície hodnoty ϕ_{sp} z dôvodu udržania vírniku v priamočiarom lete. Táto regulačná slučka je realizovaná pomocou PI regulátoru s konštantami $k_p = 3$, $k_i = 0.5$.

Model tohoto regulátoru slúži na vývoj nadradených slučiek v podobe regulátoru na priblíženie a pristávací manéver, ktoré budú dávať referenciu θ_{sp} . Pre jednoduchosť obsahuje subsystém regulátoru vstup na prípusť motoru, kvôli tomu, že vrámci subsystému regulátoru je integrovaný aj subsystém na komunikáciu Simulink-u a FlightGear.

Pre overenie regulátorov som nechal odsimulovať vírnik spoločne s regulátorom polohy 6.8. Odozvy na skok referencie prebehli pri počiatočných podmienkach $\theta_0 = 0^\circ$, $\phi_0 = 0^\circ$, $V_{air} = 21.5 \text{ m s}^{-1}$.



Obrázok 6.8: Odozvy častí regulátoru orientácie s vírnikom na skok

6.3 Overenie modelu

6.3.1 Overenie rovníc pre rotor

V časti venujúcej sa rotoru vo vertikálnom klesavom lete 6.1.2 som dostal rovnice pre silu rotoru F_r 6.21 a moment roztáčajúci rotor M_r 6.22. Po dosadení hodnôt koeficientov pre referenčný rotor som dopočítal nasledujúce hodnoty F_r , M_r pri klesaní s rýchlosťou $\dot{z} = -7,7 \text{ m s}^{-1}$ a uhlovou rýchlosťou rotoru $\dot{\psi} = 32,5 \text{ rad s}^{-1}$ a porovnal ich s hodnotami sily a momentu pre referenčný rotor [20].

	F_r (N)	M_r (Nm)
Rovnice 6.21, 6.22	3319	0,07
Literatúra [20]	3804	≈ 0

Tabuľka 6.1: Porovnanie vypočítaných hodnôt

V porovnaní pozorujem rozdiely v sile rotoru F_r . Tento rozdiel je spôsobený tým, že autor nezískal silu F_r integrovaním cez list rotoru, ale rozdelením rotoru na 10 rovnako dlhých častí a sčítaním 10 aerodynamických síl získaných jednotlivými elementmi a následným prenasobením získanej sily od listu rotoru počtom listov rotoru [20].

Autor takisto porovnal klesavú rýchlosť V_{sink} a uhlovú rýchlosť otáčania rotoru v klesavom lete v závislosti na závaží o hmotnosti m zaveseného pod rotor. Keďže v literatúre nie je určený presný matematický model simulácie, navrhol som matematický model zodpovedajúci danej problematike. Systém je popísaný diferenciálnymi rovnicami 6.56, 6.57. Systém som simuloval v programe MATLAB a ako použitý numerický solver som zvolil *ode45* a pre porovnanie som vzal hodnoty v ustálenom stave po prechodovom deji.

$$\ddot{z} = \frac{F_r(\dot{z}, \dot{\psi})}{m} - g \quad (6.56)$$

$$\ddot{\psi} = \frac{M_r(\dot{z}, \dot{\psi})}{J_r} \quad (6.57)$$

V [20] sú hodnoty vertikálnej rýchlosti a uhlovej rýchlosti otáčania rotoru vypočítané pomocou v literatúre odvodených analytických vzorcov a simulácie, ktorá počíta aj s vplyvom Machovho čísla.

m (kg)	V_{sink} (m s ⁻¹)			$\dot{\psi}$ (rad s ⁻¹)		
	Vzorce	Sim. [20]	Sim. 6.56 6.57	Vzorce	Sim. [20]	Sim. 6.56, 6.57
392	7,7	7,7	7,8	32,5	32,5	33,6
784	10,9	11,1	11,0	46,1	44,3	47,5
1176	13,3	14,1	13,5	56,4	52,1	58,2

Tabuľka 6.2: Porovnanie mojej simulácie, simulácie a vzorcov podľa [20]

Z tabuľky 6.2 pozorujem mierny rozdiel v hodnotách, ktorý je najmä spôsobený tým, že môj model nepozná presnú závislosť pre indukovanú rýchlosť w_i , ale len linearizovanú 6.30. No napriek tomu považujem môj model za dostatočne presný.

6.3.2 Overenie modelu vírniku v priblížení

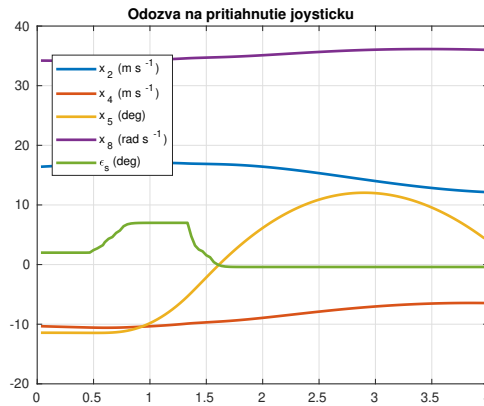
Vzhľadom na to, že nepoznám presné parametre vírniku TF-G2, najmä tie týkajúce sa rotoru, nemôžem simulovať systém a porovnať ho s odozvou reálneho vírniku TF-G2 respektíve modelom implementovaným v simulátore FlightGear [13]. Zároveň vírnik, s ktorého rotorom som pracoval v predchádzajúcej časti ako s referenčným rotorom sa odlišuje konštrukčne od vírniku TF-G2, najmä v rozmeroch a hmotnosti, keďže ide o vírnik, ktorý ovláda pilot. Preto nemám ako reálne porovnať odozvy získané mojím matematickým modelom. Overenie môjho matematického modelu, bude spočívať v overení toho, čo sa deje s vírnikom pri priblížení a zhodnotení toho, čo sa deje so stavmi, silou F_r , momentom M_r v reakcii na riadiaci signál z joysticku.

Vírnik ovládam pomocou joysticku a zobrazujem jeho stav v Simulinku a vizualizujem ho pomocou FlightGear-u. Kvôli citlivosti riadenia z joysticku som nastavil menší rozsah uhlu ϵ_s , a to v rozmedzí $\langle -3^\circ, 7^\circ \rangle$, aby bol ľahko ovládateľný. Referenčný vírnik [20] má rozsah $\langle -5^\circ, 15^\circ \rangle$. Simulinkovský model, obsahujúci matematický model vírniku so vstupmi/výstupmi a vizualizáciou sa nachádza v súbore `autogyro_approach_model.slx`.

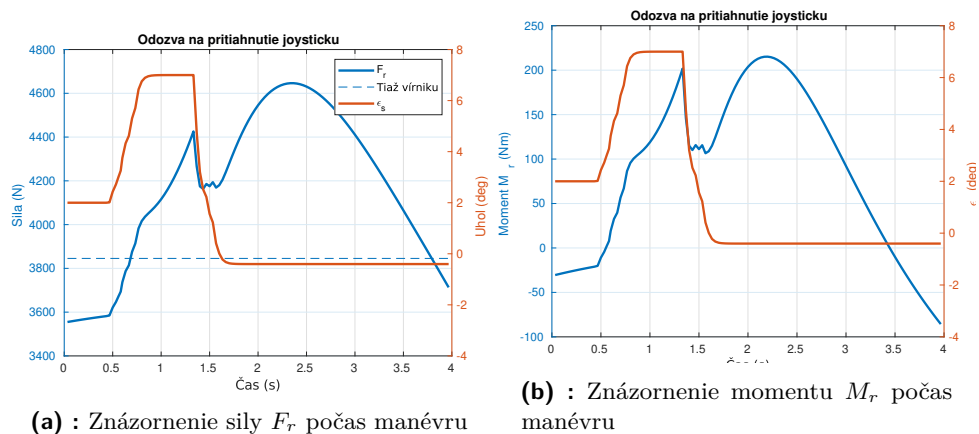
Najprv je potrebné určiť si stav, v ktorom chcem spustiť simuláciu. Konkrétne mi ide o priblíženie vírniku, teda dopredný klesavý let. Matematický model je navrhnutý tak, aby vírnik klesal, teda stav x_4 je záporný a dopredná rýchlosť, stav x_4 kladný, uhlová rýchlosť otáčania rotoru, stav x_8 kladný. Zaujímá ma celková vzdialenosť, ktorú vírnik preletí počas priblíženia a pristávacieho manévru, teda požadujem $x_0 = 0$ m. Vírnik nechám letieť s doprednou rýchlosťou $x_2 = 12$ m s⁻¹. Ako počiatočnú výšku pre manéver zvolím $x_3 = 500$ m. V priblížení chcem, aby vírnik mal klesavú rýchlosťou $x_4 = -5$ m s⁻¹. Nechcem aby vírnik menil svoj náklon, teda $x_6 = 0$ rad s⁻¹.

Na otočenom uhle rotoru nezáleží takže ho inicializujem na $x_7 = 0$ rad. Hodnotu náklonu vírniku x_5 a otáčky rotoru x_8 dopočítam z rovníc 6.49, 6.51, pričom ľavú stranu položíam rovnú nule, s cieľom nemať žiadne zrýchlenie v osi x_l a z_l . Po dopočítaní sústavy rovníc som dostal hodnoty $x_5 = -1,43^\circ$ a $x_8 = 42,56 \text{ rad s}^{-1}$.

Odozva vírniku na riadiací signál pritiažnutia joysticku, zväčšenie ϵ_s , je taká, že sa zvýšia otáčky rotoru a nakláňa sa vírnik v kladnom smere uhlu θ . Podobný priebeh sa deje aj s vírnikom v simulátore Flightgear. Zväčší sa sila F_r a dôjde k zmenšeniu vertikálnej a horizontálnej rýchlosti vírniku. Vírnik sa správa tak ako som očakával. To podporujú priebehy na obrázkoch 6.9, 6.10. Zaujímavý priebeh sa nachádza v sile F_r , momente M_r a stave x_5 . Na nich pozorujem, že x_5 sa rýchlo zmení s priebehom vstupu a následne sila a moment sa zväčšuje, pretože rastie uhol náklonu θ a teraz najmä od tohto stavu sa odvíjajú zvyšné. Fragment letu je krátky, aby zvyšné zásahy joystickom nepôsobili rušivo, pretože po krátkej dobe bolo nutné vírnik korigovať.



Obrázok 6.9: Vybrané stavy vírniku počas manévru



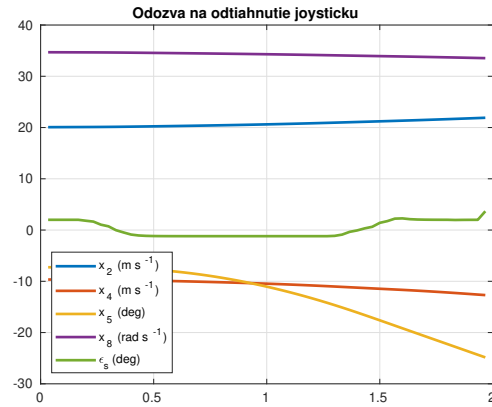
(a) : Znázornenie sily F_r počas manévru

(b) : Znázornenie momentu M_r počas manévru

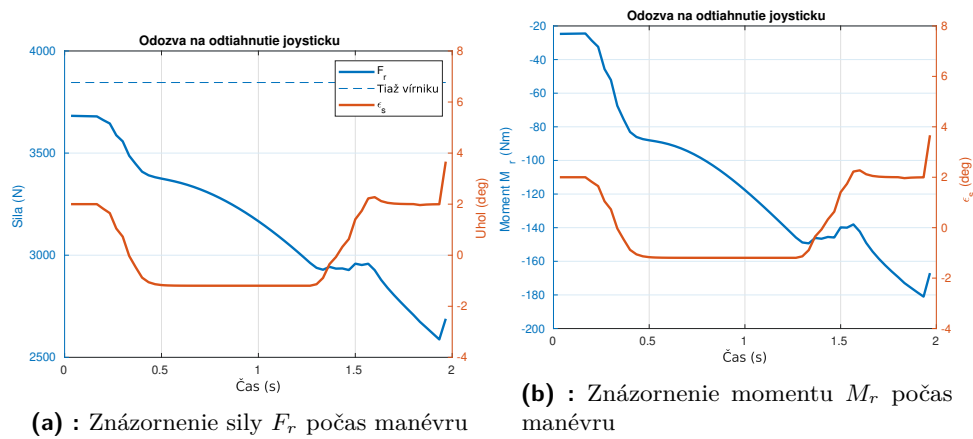
Obrázok 6.10: Odozva F_r a M_r

Pri odtiahnutí páky joysticku klesajú otáčky rotoru, to je tým, že sa zmenší uhol α v rovnici 6.34. Vírnik začne rýchlejšie klesať a zväčší svoju doprednú

rýchlosť. Čo sa týka zmeny uhlu θ tá je pomalšia ako v prípade pritiažnutia joysticku. To potvrdzujú priebehy na obrázkoch 6.9, 6.10. Zväčšovanie vektora rýchlosti vírniku zodpovedá menšej sile F_r ako v prípade pritiažnutia páky joysticku. Fragment letu je krátky, pretože hneď bolo potrebné spraviť zásah, inak by sa simulácia dostala do podmienok, na ktoré nie je navrhnutý model.



Obrázok 6.11: Vybrané stavy vírniku počas manévru



(a) : Znázornenie sily F_r počas manévru

(b) : Znázornenie momentu M_r počas manévru

Obrázok 6.12: Odozva F_r a M_r

Kapitola 7

Implementácia regulátorov

V tejto kapitole navrhнем regulačné slučky, za účelom automatického pristávania. Systém, ktorý chcem regulovať je vírnik TF-G2 s regulátorom orientácie. Podľa 6, bude vstupom systému θ_{sp} , keďže sa zameriavam na pozdĺžny pohyb a smer vírnik bude držať prednastavený smer letu a nebude ho počas pristávania meniť. Napriek tomu, že bol nainplementovaný model regulátoru orientácie nie sú známe parametre rotoru vírniku TF-G2, a preto nie je zatiaľ možné použiť model vírniku z predchádzajúcej kapitoly, pokiaľ nebudú určené neznáme parametre vírniku TF-G2. Preto v tejto kapitole navrhнем regulátory postupmi, pomocou ktorých sa budú implementovať regulátory na automatické pristávanie vírniku TF-G2. Na otestovanie regulačných slučiek som použil model regulátoru orientácie zo sekcie 6.2.

7.1 Určenie prenosu

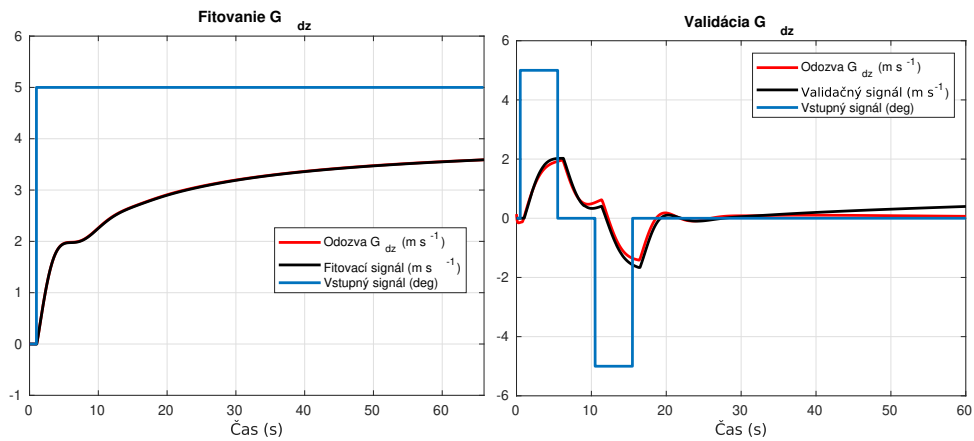
Keďže nie je známy stavový model vírniku TF-G2, tak si potrebný prenos dôležitý na návrh regulátoru identifikujem pomocou System Identification Toolbox.

Prenos $G_z(s)$ predstavuje prenos z θ_{sp} na vertikálnu rýchlosť vírniku. Vírnik s regulátorom orientácie som fitoval na systém s piatimi pólmi a troma nulami. Počet pólov vyplýva z toho, že lietadlo má v pozdĺžnej dynamike štyri póly [5]. Použitý regulátor orientácie je implementovaný PID regulátorom 6.2, ktorý má jeden pól. V uzavretej slučke má systém päť pólov. Počet núl opovedá tomu, aby sedel fit a validácia prenosu. Tento prenos som fitoval v pracovnom bode $\theta_0 = -30^\circ$, $V_{air} = 21,5 \text{ m s}^{-1}$, pričom rýchlosť klesania je $\dot{z}_0 = -11,7 \text{ m s}^{-1}$. Grafy z fitovania a validácie G_z sú na obrázku 7.1.

$$G_z(s) = \frac{-22,7s^3 + 671,6s^2 + 1070s + 261,3}{s^5 + 33,65s^4 + 638,8s^3 + 2329s^2 + 2920s + 461,3} \quad (7.1)$$

7.2 Implementácia priblíženia

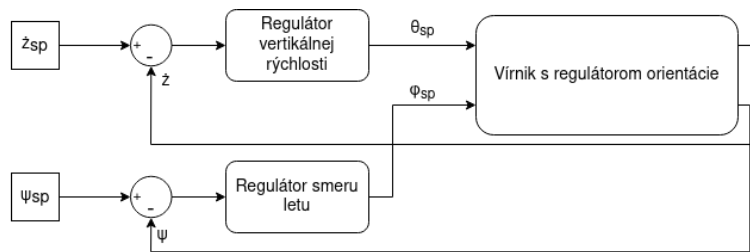
Priblíženie implementujem pomocou PI regulátoru, ktorý reguluje rýchlosť klesania. Tento regulátor je v nadradenej slučke regulátoru orientácie a bude mu nastavovať požadovaný náklon vírniku θ_{sp} . Počas priblíženia bude vírnik



(a) : Vstup a výstup na fitovanie prenosu (b) : Vstup a výstup na validáciu prenosu

Obrázok 7.1: Fitovanie G_{dz}

klesať prednastavenou konštantnou rýchlosťou $\dot{z}_l = 10 \text{ m s}^{-1}$. V priblížení vírnik bude vírnik zároveň udržiavať prednastavený smer letu 6.2. Priblíženie sa skončí začatím pristávacieho manévru v prednastavenej výške nad povrchom zeme h_L .



Obrázok 7.2: Blokový diagram regulácie počas priblíženia

7.3 Implementácia pristávacieho manévru

Pristávací manéver som implementoval pomocou troch rôznych postupov, pričom ten vhodný sa vyberie až na základe testovacieho letu. Na pristávací manéver sa kladú viaceré požiadavky vyplývajúce z limitácie konštrukcie vírniku, ale aj aktuálne dostupných technických možností.

- Limit maximálnej doprednej rýchlosti pri dosadnutí vyplývajúci z aktuálnej konštrukcie vírniku. Dopredná rýchlosť je veľmi kritická, pretože podvozok musí zvládať veľký odpor od povrchu a nerovnosti terénu, preto sa pri návrhu regulátorov budem snažiť o jej minimalizáciu aj za cenu zvýšenia rýchlosti klesania.
- Limit maximálnej rýchlosti klesania vyplýva z konštrukcie podvozku. Počas písania bakalárskej práce boli k vírniku pridané dve podvozkové

nohy, kvôli rozloženiu záťaže, avšak v bode dosadnutia sa snažím o vertikálnu rýchlosť v intervale $\dot{z}_l \in \langle 0, -2 \rangle \text{ m s}^{-1}$.

- Výška naklesaná počas manévru sa musí nachádzať v rozsahu senzorov vzdialenosti. Preto sa kladie podmienka na rýchly manéver, keďže minimálna rýchlosť klesania vírniku je 5 m s^{-1} , čo je pri aktuálne použitých senzoroch čas pre manéver $\approx 2 \text{ s}$.
- Snaha sa čo najmenej spoliehať na senzor vzdialenosti, ktorý môže šumieť a vnášať do spätnej väzby nepresnosti.
- Vo výške 30 cm nad povrchom nastavím hodnotu náklonu rotoru $\eta_n = 0$. Je to z toho dôvodu, že sa často pri pristátí ohne tyč, na ktorej je umiestnený rotor a tyč, na ktorej je pripevnená smerovka. To ohnutie je do takej miery, že rotor rozseká smerovku. Preto je snaha v určitej výške nakloniť rotor dopredu a zabrániť poškodeniu smerovky. Zároveň zo skúsenosti pilotov vírniku je potrebné rotor po pristátí nakloniť dopredu, pretože vírnik kvôli roztočenému rotoru začne cúvať.

Počas pristávacieho manévru vypínam reguláciu smeru letu, ale bude sa udržiavať hodnota ϕ_{sp} z konca priblíženia, to zabráni rozkmitaniu systému počas pritiahnutia rotoru a spomalení vírniku a nie výraznú zmenu smeru letu.

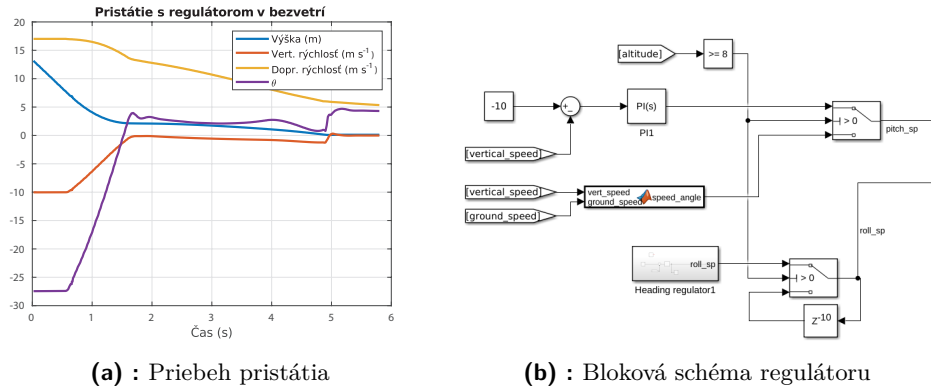
7.3.1 Regulácia náklonu vírniku

Pristávací manéver pozostáva z regulácie uhla náklonu vírniku θ . Počas pristávacieho manévru používam priamo regulátor orientácie vírniku. Hodnoty θ_{sp} sú rovné uhlu, ktorý zvierá vektor rýchlosti $-V_{\text{air}}$ s vektorom rýchlosti \dot{x}_l . Výšku, v ktorej sa začne samotný pristávací manéver som pre model vírniku v simulátore FlightGear určil ako $h_L = 8 \text{ m}$.

Výhodou tohoto spôsobu pristávania je, že sa vírnik príliš nenakloní počas pristávania, keďže uhol nie je veľký a znižuje sa zároveň so znižujúcou sa rýchlosťou klesania. Výhodou nie veľkého náklonu je to, že vírnik rýchlo nezíska klesavú rýchlosť. Výška h_L je nastavená tak, aby vírnik chvíľu letel v doprednom lete nad zemou. Počas tohoto letu síce vírnik získa malú klesavú rýchlosť (mierne sa nakloní) avšak počas dopredného letu v naklonení stratí značnú časť doprednej rýchlosti, ktorá je kritická pri pristátí a je snaha ju minimalizovať. Zároveň získaná klesavá rýchlosť, ktorá sa nachádza v rozmedzí $1 - 2 \text{ m s}^{-1}$ je pre podvozok prijateľná. Pri dosadnutí v nulovom protivetri je vertikálna rýchlosť vírniku $\dot{z}_l = -1,2 \text{ m s}^{-1}$ a dopredná rýchlosť $\dot{x}_l = 6 \text{ m s}^{-1}$. Pri dosadnutí s protivetrom 5 m s^{-1} je vertikálna rýchlosť vírniku $\dot{z}_l = -1,5 \text{ m s}^{-1}$ a dopredná rýchlosť $\dot{x}_l = 0,5 \text{ m s}^{-1}$.

Nasledujúcou výhodou tohto systému pristávania je, že sa v spätnej väzbe nespolieham na senzor výšky, ktorý môže do spätnej väzby zaniest šum a tým výrazne ovplyvniť pristávací manéver. Pristáva sa za predpokladu, že pristávací plocha neobsahuje diery a iné nerovnosti. Na senzore vzdialenosti závisí iba časť, kedy ma vírnik začať pristávací manéver.

Zo simulátoru mám vyskúšané, že vírnik týmto spôsobom dokáže pristáť aj v protivetri do 5 m s^{-1} , pričom protivietor pomáha zmenšiť doprednú rýchlosť na pre podvozok prijateľné hodnoty. Pri vyšších hodnotách rýchlosti vetru by bolo vhodné implementovať kompenzáciu doprednej rýchlosti motorom aby vírnik nezačal cúvať a vyhol by sa nestabilite.



Obrázok 7.3: Pristávací manéver s použitím regulácie náklonu vírniku

7.3.2 Regulácia vertikálnej rýchlosti s implicitným modelom

Pristávací manéver pozostáva z regulácie vertikálnej rýchlosti, pomocou toho istého regulátora ako pre prípad priblíženia. Požadované hodnoty vertikálnej rýchlosti vyplývajú z požadovanej trajektórie výšky 7.2, kde τ_h je časová konštanta deju [22].

$$h(t) = h_L e^{-\frac{t}{\tau_h}} \quad (7.2)$$

Pre zapojenie na obrázku 7.4b platí pre regulačnú odchýlku vertikálnej rýchlosti vzťah 7.3 a za predpokladu, že v otvorenej slučke je veľké zosílenie $e(t) \rightarrow 0$ je rovnica 7.2 riešením rovnice 7.3.

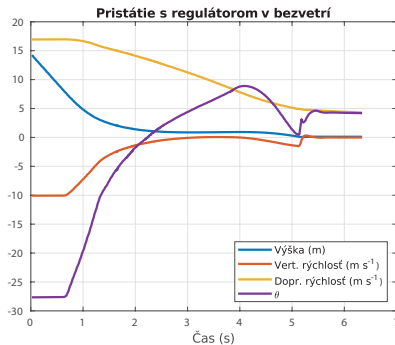
$$e(t) = -\frac{h}{\tau_h} - \dot{h} \quad (7.3)$$

Pre začiatok pristávacieho manévru platí $\dot{h}(0) = -10 \text{ m s}^{-1}$, $e(0) = 0$ a požadujem $h(0) = h_L = 5 \text{ m}$. Dosadením do rovnice 7.3 som vypočítal časovú konštantu $\tau_h = 0.2 \text{ s}$.

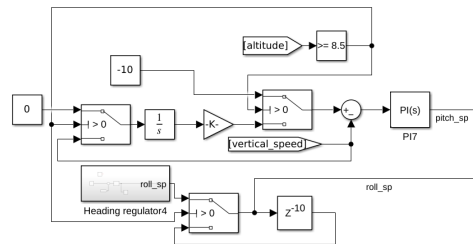
Kvôli tomu, že regulátor nesleduje dokonale referenciu som zväčšil výšku, v ktorej začína manéver na $h_L = 8,5 \text{ m}$. Počas pristávania sledujem podobný jav ako v prvom prípade, kde vírnik prestane klesať v určitej výške nad povrchom a v ďalšom lete zase naberá klesavú rýchlosť, počas toho ako stráca doprednú rýchlosť. Narozdiel od prvého regulátora sa vírnik nakloní výraznejšie a tým pádom viac brzdí, pričom získa väčšiu klesavú rýchlosť. Jej hodnota v momente dosadnutia je $\dot{z}_l = -1,5 \text{ m s}^{-1}$ a dopredná rýchlosť $\dot{x}_l = 5 \text{ m s}^{-1}$. Pri dosadnutí s protivetrom 5 m s^{-1} je vertikálna rýchlosť vírniku $\dot{z}_l = -1,4 \text{ m s}^{-1}$ a dopredná rýchlosť $\dot{x}_l = 0.1 \text{ m s}^{-1}$.

Táto implementácia využíva senzor výšky počas celého pristávania, čo môže mať nevýhodu v tom, keď senzor výšky nespoľahlivo odmeria výšku. Tento problém je možné vyriešiť integráciou vertikálnej rýchlosti a podobným zapojením do spätnej väzby ako v zapojení so senzorom výšky. Toto riešenie v simulátore dosahuje rovnaké výsledky ako v zapojení s výškou, kvôli tomu, že meranie výšky v simulátore je dokonalé a nesimuluje senzor.

Po odsimulovaní v scenári s protivetrom je možné pristávať do rýchlosti protivetru 5 m s^{-1} , inak vírnik začne cúvať.



(a) : Pribeh pristátia



(b) : Bloková schéma regulátoru

Obrázok 7.4: Pristávací manéver s použitím regulácie vertikálnej rýchlosti s implicitným modelom

7.3.3 Regulácia vertikálnej rýchlosti s explicitným modelom

Pristávací manéver používa regulátor vertikálnej rýchlosti pre priblíženie. Požadované hodnoty vertikálnej rýchlosti splňujú rovnicu 7.4. Čas pristávacieho manévru sa určí zintegrováním konštanty 1 a tým sa dopočítajú hodnoty požadovaných hodnôt vertikálnej rýchlosti 7.5.

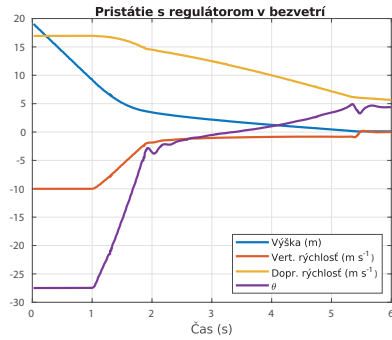
$$\dot{z} = -10 e^{-\frac{t}{0.4}} \quad (7.4)$$

Zintegrováním rovnice 7.4 podľa času v intervale $t \in \langle 0, \infty \rangle$ dostanem počiatočnú výšku, ktorá je potrebná na pristávací manéver, za podmienky dokonalého sledovania referencie. Hodnota výšky získaná integráciou sú 4 m. Kvôli nedokonalému sledovaniu referencie som zvolil počiatočnú výšku $h_L = 9,3 \text{ m}$.

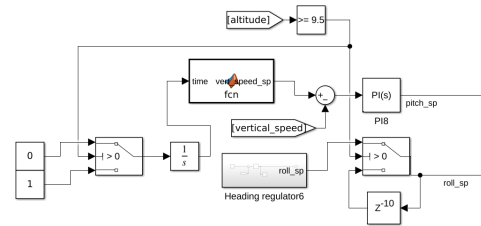
Vírnik sa počas pristávania nespráva ako v predchádzajúcich dvoch prípadoch, že sa mu začne zvyšovať rýchlosť klesania. Vírnik postupne znižuje klesavú rýchlosť a následne dosadne v porovnaní s predchádzajúcimi spôsobmi celkom malým uhlom náklonu θ . V čase dosadnutia mal vírnik vertikálnu rýchlosť $\dot{z}_l = -0,8 \text{ m s}^{-1}$ a doprednú rýchlosť $\dot{x}_l = 6,2 \text{ m s}^{-1}$. Pri dosadnutí s protivetrom 5 m s^{-1} je vertikálna rýchlosť vírniku $\dot{z}_l = -0,8 \text{ m s}^{-1}$ a dopredná rýchlosť $\dot{x}_l = 1,6 \text{ m s}^{-1}$.

Výhodou tohoto riešenia je, že sa nespolieham na senzor výšky v spätnej väzbe, ale iba počas inicializácie pristávajúceho manévru. Zároveň je celý

manéver plynulejší a menej náročný na rýchle pohyby roztočeného rotoru. Zo simulátoru je možné pristávať v protivetre do rýchlosti 5 m s^{-1} , pri vyšších rýchlostiach je potrebné kompenzovať odporovú silu vetra motorom, aby vírnik nezačal cúvať.



(a) : Pribeh pristátia



(b) : Bloková schéma regulátoru

Obrázok 7.5: Pristávací manéver s použitím regulácie vertikálnej rýchlosti s explicitným modelom

Kapitola 8

Výsledky

Väčšina cieľov stanovených v podkapitole 1.1 bola splnená. Ich zoznam je uvedený nižšie.

- Vírník
 - Mechanika letu vírniku bola popísana vrámci spôsobu akým sa roztáča rotor 6.1.2 a vírník som porovnal s helikoptérou a lietadlom 2. Vyvinul, otestoval a použil som komunikačný prostriedok spájajúci FlightGear a Simulink v podkapitole 4.1.1.
- Estimácia výšky nad povrchom
 - Vybral som senzory vzdialenosti 5.1. Podarilo sa mi spojiť senzory s PX4 a implementoval som selektor, ktorý vyberá vhodný senzor vzdialenosti na určenie výšky 5.3. Otestoval som senzory vzdialenosti 5.4.3. Nepodarilo sa mi otestovať systém estimácie výšky v PX4, tak ako som plánoval. Cieľ sa nepodarilo splniť z dôvodu havárie vírniku TF-G2 a už sa nestihol ďalší testovací let do odovzdania bakalárskej práce.
- Model vírniku v priblížení
 - Vysvetlil som princípy aerodynamiky rotoru v klesavom vertikálnom lete a popísal vzniknuté aerodynamické sily analytickými rovnicami 6.1.2. Naimplementoval som matematický model pozdĺžnej mechaniky letu vírniku v klesavom doprednom lete pomocou S-Funkcie 6.1.4. Podarilo sa mi overiť matematický model rotoru vo vertikálnom klesavom lete 6.3.1. Nepodarilo sa mi overiť model vírniku TF-G2 v doprednom klesavom lete s reálnymi letovými dátami, pretože nie sú známe parametre vírniku TF-G2 a nepodarilo sa mi nájsť literatúru, ktorá obsahuje model vírniku s takými parametrami aké má môj model, tak som overil model iba kvalitatívne 6.3.2.
- Návrh automatického pristávania
 - Implementoval som PI regulátor pomocou, ktorého je vírník TF-G2 v priblížení 7.2. Podarilo sa mi implementovať tri spôsoby automatického pristávania a overiť ich funkčnosť v simulátore FlightGear 7.3.

Kapitola 9

Záver a práca dobudúcna

Táto bakalárska práca je súhrnom viacerých častí, ktoré uvádzajú čitateľa do problematiky návrhu a vývoja súčastí malého bezpilotného vírniku TF-G2, vyvinutého pre vedecké experimenty projektu CRREAT. Popísal som mechaniku letu vírniku, vyvinul nástroj na prepojenie simulátoru FlightGear a Simulink-u, ktorý nie je použiteľný len pre vírnik TF-G2. Vytvoril som matematický model pozdĺžnej mechaniky letu vírniku a navrhol algoritmy pre automatické pristávanie.

Počas práce na tejto bakalárskej práci vzniklo viacero myšlienok a nápadov, ktoré môžu stavať na tejto bakalárskej práci. Preto bola táto bakalárska práca písaná tak aby niekto, kto bude na danom projekte chcieť pracovať bol oboznámený s problematikou vírniku, použitých simulačných prostriedkov a autopilotom PX4, keďže mne osobne zabralo veľa času zorientovať sa v širokej problematike predstavenej v tejto práci.

Ako projekt, ktorý by mohol byť pokračovaním bakalárskej práce je možné navrhnúť toolchain, ktorý by umožňoval identifikovať neznáme parametre vírniku, na základe letového logu. Táto práca by umožnila overiť implementovaný matematický model pozdĺžnej mechaniky letu vírniku. Ponúka sa implementovať spomenuté regulátory pre automatické pristávanie do autopilotu PX4 a vyskúšať ich. Na základe matematického modelu by bolo možné navrhnúť postup ako ladiť jednotlivé regulátory a tým zmenšiť počet testovacích letov vírniku a zefektívniť celý proces. Ako ďalší projekt sa ponúka overiť estimátor výšky a upraviť selektor senzorov vzdialenosti tak, aby pokrýval širšie spektrum chýb merania oboch senzorov vzdialenosti.

Mojim cieľom po odovzdaní tejto bakalárskej práce je pokračovať a zrealizovať automatické pristátie, ktorému dala táto bakalárska práca základ. Automatické pristávanie má potenciál výrazne znížiť nehodovosť pri pristávaní vírniku TF-G2 spôsobenú ľudským faktorom, čo by malo za následok zredukovanie počtu opráv a kratšie prestávky medzi jednotlivými letmi vírniku.

Dodatok A

Súradnicové systémy

Určenie súradnicových systémov a transformácii medzi nimi je dôležité pre získanie rovníc popisujúcich systém v požadovanom súradnicovom systéme.

■ NED Systém

NED(North East Down) je pravotočivý ortogonálny súradnicový systém. Tento súradnicový systém ma svoju osu x smerujúcu na sever. Osa y tohto súradnicového systému smeruje na východ. Osa z mieri kolmo k zemi. Počiatok tohto súradnicového systému sa nachádza v ťažisku lietadla [23]. Veličiny vyjadrené v tomto súradnicovom systéme označím spodným indexom NED.

■ Earth systém

Tento súradnicový systém je skoro rovnaký s NED súradnicovým systémom. Rozdiel medzi nimi, je v tom, že osa x mieri v smere letu [5]. Veličiny vyjadrené v tomto súradnicovom systéme označím spodným indexom E.

■ Body systém

Je to pravotočivý ortogonálny súradnicový systém, pevne spojený s telom prostriedku. Pohyb je vyjadrený v zmenách veličín vzťahnutých k osiam. Jeho počiatok je zvyčajne umiestnený v ťažisku. Osa x smeruje v smere letu a osa z mieri kolmo dolu [5]. Veličiny vyjadrené v tomto súradnicovom systéme označím spodným indexom b.

Dodatok B

Literatúra

- [1] J. Leishman, “Development of the autogiro: A technical perspective,” *Journal of Aircraft - J AIRCRAFT*, vol. 41, pp. 765–781, 07 2004.
- [2] P. S. Archives, “Kellett Autogiro flying mail off the roof of the Philadelphia Post Office.” Dostupné z: <https://explorepahistory.com/displayimage.php?imgId=1-2-1371>. Citované (18.05.2022).
- [3] AutoGyro, “Mtosport 2017.” Dostupné z: <https://www.auto-gyro.com/en/Gyroplane/AutoGyro-Models/MT0sport-2017/>. Citované (17.05.2022).
- [4] ThunderFly, “FlightGear TF-G2.” Dostupné z: <https://github.com/ThunderFly-aerospace/TF-G2>, 2022. Citované (18.05.2022).
- [5] R. W. Pratt, *Flight Control Systems: Practical issues in design and implementation*. Stevenage: IET, 2000.
- [6] D. Thomson and S. Houston, “Advances in understanding autogyro flight dynamics,” in *64th Helicopter Society Annual Forum*, (Montreal, Canada), p. 13, 01 2008.
- [7] J. G. Leishman, *Principles of Helicopter aerodynamics*. Cambridge aerospace series, Cambridge: Cambridge University Press, 2000.
- [8] PX4 Autopilot, “PX4 User Guide.” Dostupné z: <https://docs.px4.io/v1.12/en/>, 2021.
- [9] L. Meier, D. Honneger, and M. Pollefeys, “A node-based multithreaded open source robotics framework for deeply embedded platforms,” in *ICRA*, 2015. Citované (18.05.2022).
- [10] PX4 Autopilot, “FixedWingPositionControl.cpp.” Dostupné z: https://github.com/ThunderFly-aerospace/PX4Firmware/blob/tf/master/src/modules/fw_pos_control_11/FixedwingPositionControl.cpp, 2022. Citované (18.05.2022).
- [11] PX4 Autopilot, “FixedWingAttitudeControl.cpp.” Dostupné z: https://github.com/ThunderFly-aerospace/PX4Firmware/tree/tf/master/src/modules/fw_att_control, 2022. Citované (18.05.2022).

- [12] “FlightGear wiki.” Dostupné z: <https://wiki.flightgear.org/>, 2021. Citované (18.05.2022).
- [13] Garry R. Neely, “Guide to YASim.” Dostupné z: <http://www.buckarooshangar.com/flightgear/yasimtut.html>, 2013. Citované (18.05.2022).
- [14] Benewake (Beijing) Co. Ltd., “Product Manual of TFmini-S.” Dostupné z: https://cdn.sparkfun.com/assets/8/a/f/a/c/16977-TFMini-S_-_Micro_LiDAR_Module-Product_Manual.pdf. Citované (18.05.2022).
- [15] Terabee, “TIME-OF-FLIGHT PRINCIPLE: INDIRECT VS. DIRECT TOF SENSORS.” Dostupné z: <https://www.terabee.com/time-of-flight-principle/>, 2022. Citované (18.05.2022).
- [16] MaxBotix Inc., “MB1242 datasheet.” Dostupné z: https://www.maxbotix.com/documents/I2CXL-MaxSonar-EZ_Datasheet.pdf, 2021. Citované (18.05.2022).
- [17] PX4 Autopilot, “EKF Source code.” Dostupné z: <https://github.com/PX4/PX4-Autopilot/tree/master/src/modules/ekf2>, 2022. Citované (18.05.2022).
- [18] Daniel Agar, “Pull Request: ekf2: update distance_sensor selection to constantly monitor all instances.” Dostupné z: <https://github.com/ecmnet/Firmware/commit/92041f29b1f05dd80f89c7844c3937b56d93ceff>, 2021. Citované (18.05.2022).
- [19] J. F. Torgerson, “Simulation and control design of a gliding autogyro for precision airdrop,” Master’s thesis, Massachusetts Institute of Technology. Dept. of Aeronautics and Astronautics., Cambridge, MA, 2005.
- [20] H. Duda and J. Seewald, *Flugphysik der Tragschrauber: verstehen und berechnen*. Berlin: Springer Vieweg, 2016.
- [21] J. A. Bautista-Medina, R. Lozano, and A. Osorio-Cordero, “Modeling and control of a single rotor composed of two fixed wing airplanes,” *Drones*, vol. 5, no. 3, 2021.
- [22] R. Lungu, M. Lungu, and L. T. Grigorie, “Automatic control of aircraft in longitudinal plane during landing,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 49, no. 2, pp. 1338–1350, 2013.
- [23] B. L. Stevens, F. L. Lewis, and E. N. Johnson, *Aircraft Control and Simulation: Dynamics, Controls Design, and Autonomous Systems*. Hoboken: John Wiley & Sons, Incorporated, 2015.