

ZÁPADOČESKÁ UNIVERZITA V PLZNI
FAKULTA APLIKOVANÝCH VĚD
KATEDRA KYBERNETIKY

DIPLOMOVÁ PRÁCE

PLZEŇ 2018

Bc. TOMÁŠ MYSLIVEC

ZADÁNÍ DIPLOMOVÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Bc. Tomáš MYSLIVEC**
Osobní číslo: **A16N0145P**
Studijní program: **N3918 Aplikované vědy a informatika**
Studijní obor: **Kybernetika a řídicí technika**
Název tématu: **Omni-směrové bezpilotní letadlo**
Zadávací katedra: **Katedra kybernetiky**

Z á s a d y p r o v y p r a c o v á n í :

1. Seznamte se se současnými konstrukcemi UAVs (Unmanned Aerial Vehicles), s jejich modely a příslušnými algoritmy řízení pohybu.
2. Zvolte jednu konkrétní variantu omni-směrového UAV a pro ni vypracujte podrobný matematický model.
3. Pro tuto variantu navrhnete vhodnou strategii řízení.
4. Implementujte model UAV spolu s jeho řídicím systémem v programovém prostředí Matlab/Simulink. Funkčnost demonstřujte pomocí virtuálního modelu.
5. Navrhnete konstrukční provedení reálného UAV a implementujte příslušný řídicí systém v programovém prostředí REX.


Rozsah grafických prací: dle potřeby
Rozsah kvalifikační práce: 40-50 stránek A4
Forma zpracování diplomové práce: tištěná
Seznam odborné literatury:

BRESCIANINI, Dario a Raffaello D'ANDREA. Design, modeling and control of an omni-directional aerial vehicle. In: 2016 IEEE International Conference on Robotics and Automation (ICRA) [online]. IEEE, 2016, s. 3261-3266. DOI: 10.1109/ICRA.2016.7487497. ISBN 978-1-4673-8026-3.
LUUKKONEN, Teppo. Modelling and control of quadcopter. Independent research project in applied mathematics, Espoo, 2011.
RONGIER, P., E. LAVAREC a F. PIERROT. Kinematic and Dynamic Modeling and Control of a 3-Rotor Aircraft. In: Proceedings of the 2005 IEEE International Conference on Robotics and Automation [online]. IEEE, 2005, s. 2606-2611. DOI: 10.1109/ROBOT.2005.1570506. ISBN 0-7803-8914-X.


Vedoucí diplomové práce: Prof. Ing. Miloš Schlegel, CSc.
Katedra kybernetiky

Datum zadání diplomové práce: 2. října 2017

Termín odevzdání diplomové práce: 18. května 2018


Doc. Dr. Ing. Vlasta Radová
děkanka




Prof. Ing. Josef Psutka, CSc.
vedoucí katedry

V Plzni dne 2. října 2017

Prohlášení

Předkládám tímto k posouzení a obhajobě diplomovou práci zpracovanou na závěr studia na Fakultě aplikovaných věd Západočeské univerzity v Plzni.

Prohlašuji, že jsem diplomovou práci vypracoval samostatně a výhradně s použitím odborné literatury a pramenů, jejichž úplný seznam je její součástí.

V Plzni dne

.....

vlastnoruční podpis

Poděkování

Tímto bych rád v první řadě poděkoval panu prof. Ing. Miloši Schlegelovi, CSc., vedoucímu této diplomové práce za odborné vedení, cenné rady a připomínky, poskytnuté materiály a ochotu při zpracovávání této diplomové práce.

Dále bych chtěl poděkovat panu Ing. Martinu Švejdovi, Ph.D. za rady a připomínky při realizaci matematického modelu.

Abstrakt

Tato diplomová práce se zabývá odvozením matematického modelu a řídicího algoritmu pro všesměrové bezpilotní letadlo. Současně probíhá i odvození jednoduššího modelu kvadrokoptéry, kde lze lépe testovat funkčnost odvozených pohybových rovnic. Nejdříve je model odvozen pomocí Eulerových úhlů. Ty jsou ovšem příčinou singulárních poloh, proto je proveden převod do kvaternionů. Stabilizační řídicí algoritmus regulace rychlosti je navržen pomocí stavové zpětné vazby metodou LQR, která je formulována tak, aby uzavřený systém zachoval symetrii otevřeného systému. Dále je navržena obdobným způsobem polohová regulace. Pro reálný případ neúplné znalosti stavu je též navržen stabilizační regulátor.

Klíčová slova: omnikoptéra, kvadrokoptéra, matematický model, stavová zpětná vazba, LQ regulace, symetrické systémy, stabilizace

Abstract

This master's thesis studies the problem of a mathematical model and a control algorithm for omni directional unmanned aerial vehicle. Equations for a quadcopter is also derived at the same time. The model for the quadcopter is very useful for testing correctness of equations of motion. For the first time Euler's angles are used, but there is a problem with singularities. So quaternions are the better option. The stabilizing control algorithm for speed regulation is created by a state feedback where the LQR is used, which maintains the symmetry of the open-loop system in the closed-loop control system. There is also a position control which is created by modifying the state feedback. The stabilizing control algorithm is also designed for an incomplete system state.

Keywords: omnicopter, quadcopter, mathematical model, state feedback, LQR, symmetric systems, stabilization

Obsah

Úvod	1
1 UAV - rozdělení, konstrukce a řídicí systémy	3
1.1 Unmanned Aerial Vehicles	3
1.1.1 Aplikace UAV	4
1.2 Konstrukce UAV	6
1.2.1 Kvadrokoptéra - statické rotory	6
1.2.2 Kvadrokoptéra - dynamické rotory	7
1.2.3 Kvadrokoptéra - speciální případ	8
1.2.4 Hexakoptéra	8
1.2.5 Omnikoptéra	9
1.3 Použitý řídicí systém	9
2 Matematický model	12
2.1 Pravoúhlá soustava souřadnic	12
2.2 Pevná a pohyblivá soustava souřadnic	12
2.3 Matice rotace R	13
2.4 Newton-Eulerova metoda	17
2.4.1 Derivace matice rotace R	18
2.4.2 Vztah mezi úhly a úhlovou rychlostí - matice W	19
2.4.3 Síla působící na těleso - vyjádření v pohyblivé soustavě	20
2.4.4 Moment hybnosti tělesa - vyjádření v pohyblivé soustavě	21
2.5 Rotory	23
2.6 UAV - působící síla a moment	25
2.6.1 Kvadrokoptéra	26
2.6.2 Omnikoptéra	27
2.7 Určení matice setrvačnosti - J_1	29
2.8 Nelineární model	31
2.8.1 Rovnovážný stav	32
2.9 Kvaterniony	35
2.9.1 Základní vztahy	36
2.9.2 Součin dvou kvaternionů	37
2.9.3 Rotace vektoru pomocí kvaternionu	39
2.9.4 Derivace kvaternionu	40
2.9.5 Úhlová rychlost	40
2.9.6 Druhá derivace kvaternionu	41
2.9.7 Vztah mezi Eulerovými úhly a kvaterniony	41
2.9.8 Nelineární model vyjádřený v kvaternionech	42

3	Návrh řídicího systému	44
3.1	Linearizace	44
3.1.1	Kvadroptéra	45
3.1.2	Omnikoptéra	48
3.2	Jordanova normální forma	52
3.2.1	Homogenní koule v prostoru bez gravitace	53
3.2.2	UAV	54
3.3	Stavová zpětná vazba	55
3.3.1	LQR	55
3.3.2	Symetrické LQR	56
3.4	Kvadroptéra - stabilizace	58
3.5	Omnikoptéra - stabilizace	66
3.6	Kvadroptéra - polohová regulace	76
3.7	Omnikoptéra - polohová regulace	79
3.8	Kvadroptéra - omezený stav, gamepad	81
3.9	Omnikoptéra - omezený stav, gamepad	85
4	3D virtuální model	89
4.1	Kvadroptéra - 3D model	89
4.2	Omnikoptéra - 3D model	90
4.3	VR Sink blok	90
4.4	Simulink - zapojení	91
4.5	Simulink - gamepad	92
4.6	Simulink - 3D animace	93
5	Konstrukce a příprava pro reálnou aplikaci ŘS	94
5.1	Konstrukce UAV	94
5.2	Přípravné zapojení	96
6	Závěr	98
	Reference	100

Seznam obrázků

1	Rotace okolo jednotlivých os - roll, pitch, yaw	12
2	<i>Aerial Target</i> - dobová fotografie - [1]	3
3	<i>DJI Mavic Air</i> - [2]	5
4	<i>MQ-1B Predator</i> - [3]	6
5	Kvadroptéra se statickými rotory	7
6	Kvadroptéra s otočnými rotory - [4]	7
7	Kvadroptéra - <i>Purdue Robotics</i> - [5]	8
8	Hexakoptéra - [6]	8
9	Omnikoptéra - [13]	9
10	Základní schéma řídicího systému UAV	10
11	Pravoúhlá soustava souřadnic	12
12	Vztah mezi soustavami	13
13	Rotace okolo jednotlivých os	15
14	Rotace - úhel α	15
15	Rotace - úhel β	15
16	Rotace - úhel γ	15
17	Otáčení rotorů - kvadroptéra	25
18	Rotor v prostoru	26
19	Rozmístění rotorů	26
20	Omnikoptéra - směry rotorů	27
21	Omnikoptéra - směry rotorů	27
22	Omnikoptéra - směr rotorů v ose \vec{z}_1	28
23	Experiment	29
24	Simulační ověření rovnovážného stavu - <i>Matlab/Simulink</i>	33
25	Simulační ověření rovnovážného stavu - <i>Matlab/Simulink</i>	34
26	Kardanův závěs	35
27	Kardanův závěs - ztráta stupně volnosti	35
28	Kvaternion - rotace	37
29	Řízení - z počáteční podmínky $x_{eq(1.polovina)} = [-2 \ 0 \ -1 \ 0 \ 0 \ 0]$	60
30	Pozice - z počáteční podmínky $x_{eq(1.polovina)} = [-2 \ 0 \ -1 \ 0 \ 0 \ 0]$	61
31	Úhly - z počáteční podmínky $x_{eq(1.polovina)} = [-2 \ 0 \ -1 \ 0 \ 0 \ 0]$	61
32	Řízení - z počáteční podmínky $x_{eq(1.polovina)} = [0 \ -2 \ -1 \ 0 \ 0 \ 0]$	62
33	Pozice - z počáteční podmínky $x_{eq(1.polovina)} = [0 \ -2 \ -1 \ 0 \ 0 \ 0]$	62
34	Úhly - z počáteční podmínky $x_{eq(1.polovina)} = [0 \ -2 \ -1 \ 0 \ 0 \ 0]$	63
35	Řízení - z počáteční podmínky $x_{eq(1.polovina)} = [-2 \ -1 \ -1 \ \pi \ 0 \ 0]$	64
36	Pozice - z počáteční podmínky $x_{eq(1.polovina)} = [-2 \ -1 \ -1 \ \pi \ 0 \ 0]$	64
37	Úhly - z počáteční podmínky $x_{eq(1.polovina)} = [-2 \ -1 \ -1 \ \pi \ 0 \ 0]$	65

38	Pozice - z počáteční podmínky $x_{eq(1.polovina)} = [0\ 0\ -1.5\ 0\ 0\ 0]$ pro \hat{F} . . .	70
39	Úhly - z počáteční podmínky $x_{eq(1.polovina)} = [0\ 0\ -1.5\ 0\ 0\ 0]$ pro \hat{F} . . .	71
40	Pozice - z počáteční podmínky $x_{eq(1.polovina)} = [0\ 0\ -1.5\ 0\ 0\ 0]$ pro F . . .	71
41	Úhly - z počáteční podmínky $x_{eq(1.polovina)} = [0\ 0\ -1.5\ 0\ 0\ 0]$ pro F . . .	72
42	Pozice - z počáteční podmínky $x_{eq(1.polovina)} = [-2\ 0\ -1\ 0\ 0\ 0]$	73
43	Úhly - z počáteční podmínky $x_{eq(1.polovina)} = [-2\ 0\ -1\ 0\ 0\ 0]$	73
44	Pozice - z počáteční podmínky $x_{eq(1.polovina)} = [2\ 0\ -1\ 0\ 0\ 0]$	74
45	Úhly - z počáteční podmínky $x_{eq(1.polovina)} = [2\ 0\ -1\ 0\ 0\ 0]$	74
46	Pozice - z počáteční podmínky $x_{eq(1.polovina)} = [-2\ -1\ -1\ \pi\ 0\ 0]$	75
47	Úhly - z počáteční podmínky $x_{eq(1.polovina)} = [-2\ -1\ -1\ \pi\ 0\ 0]$	75
48	Řízení - $x_{ref(1.polovina)} = [0.1\ 0.5\ -1\ 0\ 0\ 0.5]$	77
49	Pozice - $x_{ref(1.polovina)} = [0.1\ 0.5\ -1\ 0\ 0\ 0.5]$	78
50	Úhly - $x_{ref(1.polovina)} = [0.1\ 0.5\ -1\ 0\ 0\ 0.5]$	78
51	Pozice - $x_{ref(1.polovina)} = [-0.1\ 0.2\ -0.8\ 0.1\ 0.2\ 0.4]$	80
52	Úhly - $x_{ref(1.polovina)} = [-0.1\ 0.2\ -0.8\ 0.1\ 0.2\ 0.4]$	80
53	Pozice - let ovládaný gamepadem	83
54	Úhly - let ovládaný gamepadem	83
55	Působení skokových poruch do řídicího signálu - let ovládaný gamepadem . . .	84
56	x/y - let ovládaný gamepadem	84
57	Pozice - let ovládaný gamepadem	87
58	Úhly - let ovládaný gamepadem	87
59	x/y - let ovládaný gamepadem	88
60	Virtuální model - kvadrokoptéra - <i>Blender</i>	89
61	Virtuální model - omnioptéra - <i>Blender</i>	90
62	<i>Simulink VR Sink</i> - zapojení - kvadrokoptéra	91
63	<i>Simulink VR Sink</i> - zapojení - omnioptéra	92
64	XBOX 360 gamepad - [driver for macOS]	92
65	<i>VR Sink</i> - vizualizace	93
66	<i>D800-X8</i> [31]	94
67	<i>D800-X8</i> - specifikace [31]	95
68	<i>D800-X8</i> - ramena s motory [31]	95
69	<i>REX</i> - základní schéma	97

Seznam značení a zkratek

symbol	název
A	matice dynamiky lineárního systému
A_p	rozšířená matice dynamiky lineárního systému
A_r	plocha opsaná vrtulí rotoru
B	matice řízení lineárního systému
B_i	i -tý sloupec matice řízení B
B_p	rozšířená matice řízení lineárního systému
b	koeficient tření
C	výstupní matice lineárního systému
$C(\vec{\varepsilon})$	pomocná matice při výpočtu součinu kvaternionů
D	vzdálenost mezi závěsy dvou kyvadel
$\vec{\varepsilon}$	část kvaternionu s vektorem, okolo kterého je provedena rotace
\vec{F}	vektor síly
\vec{F}_g	gravitační síla
\vec{F}_r	výslednice sil generovaná rotory
g	tíhové zrychlení
h	délka závěsu kyvadla
I	vstupní proud
I_0	proud bez zátěže
$I_{i \times i}$	jednotková matice příslušných rozměrů
J	matice setrvačnosti
J_c	kriteriální funkce LQR
J_F	Jordanova forma
K_T	konstanta tahu rotoru
K_t	proporcionální konstanta točivého momentu (1)
K_τ	proporcionální konstanta točivého momentu (2)
$K_{\tau\tau}$	konstanta točivého momentu rotoru
K_v	zpětné elektromotorické napětí
L	Lagrangián
l	délka ramene kvadroptéry
m	hmotnost UAV
$\vec{\eta}$	vektor rotace okolo os
θ	úhel natočení objektu zavěšeného na provazů
P	matice pozic rotorů vyjádřena v pohyblivé soustavě pevně spojené s tělesem
P_m	výkon motoru rotoru
P_h	výkon potřebný ke vzletu
$Q(\vec{q})$	matice ze součinu dvou kvaternionů
Q_L	zobecněná síla
\vec{q}	kvaternion

symbol	název
\mathbf{R}	matice rotace
$\mathbf{R}_{1 \rightarrow 0}$	matice rotace pro převod z pohyblivé do inerciální soustavy
$\mathbf{R}_{0 \rightarrow 1}$	matice rotace pro převod z inerciální do pohyblivé soustavy
\mathbf{R}_q	matice rotace (kvaterniony)
$\mathbf{R}_q 1 \rightarrow 0$	matice rotace (kvaterniony) pro převod z pohyblivé do inerciální soustavy
$\mathbf{R}_q 0 \rightarrow 1$	matice rotace (kvaterniony) pro převod z inerciální soustavy do pohyblivé
R_m	odpor motoru
$\vec{\rho}$	vektor spojující těžiště tělesa s jednotlivými body, kde působí momenty
ρ_w	odpor vzduchu
T	tah rotoru
T_L	kinetická energie tělesa
$\vec{\tau}$	celkový moment působící na těleso
τ_m	moment generovaný rotorem
\vec{u}	vektor řízení
V	napětí na motoru
V_L	potencionální energie tělesa
v_h	indukovaná rychlost motoru
\vec{v}	vektor rychlosti
\mathbf{W}	transformační matice mezi úhlovou rychlostí v pohyblivé soustavě a derivací úhlů v inerciální soustavě
$\mathbf{W}_{1 \rightarrow 0}$	transformační matice mezi úhlovou rychlostí v pohyblivé soustavě a derivací úhlů v inerciální soustavě
$\mathbf{W}_{0 \rightarrow 1}$	transformační matice mezi derivací úhlů v inerciální soustavě a úhlovou rychlostí v pohyblivé soustavě
$W(\vec{q})$	transformační matice mezi úhlovou rychlostí v pohyblivé soustavě a derivací kvaternionů v inerciální soustavě
$\vec{\omega}$	úhlová rychlost
ω_m	úhlová rychlost motoru
\vec{x}	stav systému
χ	matice natočení jednotlivých rotorů vyjádřena v pohyblivé soustavě souřadnic
\vec{y}	výstupní vektor systému
ϑ	úhel otočení u kvaternionu
$\vec{\xi}$	vektor polohy

Pro rozlišení zda-li se příslušný vektor nebo matice uvažuje v inerciální nebo pohyblivé soustavě je zavedeno následující značení, při kterém bude pro ukázkou využito vektoru $\vec{\xi}$:

- $\vec{\xi}_0$ je reprezentace vektoru $\vec{\xi}$ v inerciální soustavě
- $\vec{\xi}_1$ je vztažen k pohyblivé soustavě pevně spojené s tělesem

Vektory, které jsou uvažovány v jednotlivých soustavách, jsou následující

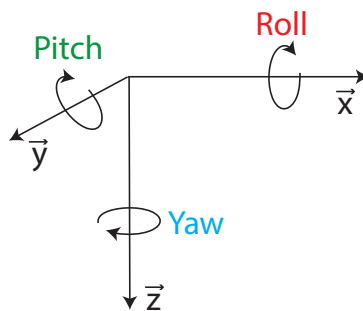
$$\vec{\xi} = \begin{bmatrix} x \\ y \\ z \end{bmatrix}, \quad \vec{v} = \begin{bmatrix} u \\ v \\ w \end{bmatrix}, \quad \vec{\eta} = \begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix}, \quad \vec{\omega} = \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}, \quad \vec{F} = \begin{bmatrix} f_x \\ f_y \\ f_z \end{bmatrix}, \quad \vec{\tau} = \begin{bmatrix} \tau_x \\ \tau_y \\ \tau_z \end{bmatrix},$$

dále jsou v jednotlivých soustavách vyjadřovány i některé jejich derivace. V případě uvažování několika různých vektorů v sumě je pak zavedeno:

- $\sum_{i=1}^N \vec{\xi}_{0i}$ je reprezentace vektorů $\vec{\xi}_i$ s indexy i v inerciální soustavě
- $\sum_{i=1}^N \vec{\xi}_{1i}$ je reprezentace vektorů $\vec{\xi}_i$ s indexy i v pohyblivé soustavě pevně spojené s tělesem

V některých úlohách (hlavně v letectví) se využívá značení

$$\vec{\eta} = \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} = \begin{bmatrix} \text{"roll"} \\ \text{"pitch"} \\ \text{"yaw"} \end{bmatrix}.$$



Obrázek 1: Rotace okolo jednotlivých os - roll, pitch, yaw

zkratka	význam
CD	Compact Disc
LQR	Linear-Quadratic Regulator
RGB	Red Green Blue
UAV	Unmanned Aerial Vehicle
UAVs	Unmanned Aerial Vehicles
USA	United States of America
USB	Universal Serial Bus
VR	Virtual Reality
VRML	Virtual Reality Modeling Language

Úvod

Cílem této práce je navržení matematického modelu a řídicího algoritmu omnisměrového bezpilotního letadla. Omnisměrové bezpilotní letadlo je inspirováno konstrukčním provedením klasických dronů, proto lze použít výraz „omnikoptéra“. Omnikoptéra je tedy druh vrtulového letounu, který v tomto případě disponuje osmi rotory. Rotory nejsou umístěny pouze jedním směrem, aby bylo docíleno možného pohybu ve všech osách. V tomto ohledu se liší od klasických kvadrokoptér, které mají všechny rotory umístěny pouze jedním směrem a nemají možnost přímého pohybu do všech stran.

Drony jsou v dnešní době velmi oblíbené zejména díky možnostem jejich využití. Na takovýto dron se dá umístit nepřeborné množství různých zařízení, díky kterým jsou schopny vykonávat nejrůznější úkoly. V drtivé většině případů jsou zde zastoupeny klasické kvadrokoptéry s občasnými modifikacemi v umístění a počtu rotorů.

Problémy návrhu vhodného řízení pro drony se zabývá celá řada článků. Za zmínku stojí například článek *Design, Modeling and Control of an Octocopter* [18], který se zabývá sice návrhem oktokopty, ale jako jeden z mála má i podstatnou část věnovanou odvození matematického modelu. Dále je také důležitý článek *Design, Modeling and Control of an Omni-Directional Aerial Vehicle* [13], který je sice podstatně stručnější, ale přichází s myšlenkou všesměrového dronu, která je v této diplomové práci využita. Ovšem ani v jedné práci není více rozebrán problém singularit při využití Eulerových úhlů a hlavně návrh za pomoci kvaternionů, který tento problém vyřeší. Z toho důvodu je řada zdrojů diplomové práce věnována čistě problematice kvaternionů. Mnoho publikací se věnuje zejména stavovým zpětným vazbám nebo využívá PD regulátory pro stabilizaci nebo polohové řízení. V této práci je využito návrhu symetrického stavového regulátoru. Touto problematikou se zabývá práce *Symetrické systémy* [30].

Jelikož pro důkladné porozumění této problematice je nutné začít s co možná nejjednoduššími modely, na které lze následně navázat, je znatelná část věnována i návrhu klasické kvadrokoptéry. Matematický model se oproti složitější omnikoptéře zase tolik neliší, je zde pouze větší množství rotorů a jiné výslednice celkových sil a momentů. V práci se tedy prolínají dva návrhy UAVs, kde je hlavním úkolem lépe porozumět problematice dronů a uvědomit si výhody právě omnikoptéry, která nabízí možnost mnohem preciznějších aplikací.

Řídicí systém je také navrhnout pro obě dvě varianty. V první fázi je nutné UAV stabilizovat, kde je pro jednoduchost předpokládáno, že je znám celý stavový vektor. Pro stabilizaci je využito symetrické LQR, které díky svým podmínkám v návrhu respektuje symetrii systému a nabízí tedy mnohem efektivnější řídicí systém. Dále je přistoupeno k polohové regulaci, ve které je využit již vypočtený stabilizační řídicí systém. Stavová zpětná vazba je rozšířena o integrátory, které v případě říditelnosti systému zaručují nulovou odchylku v požadovaných stavech. Problém nastává v reálných aplikacích, jelikož není možné znát přímo celý stav. Z tohoto důvodu je navržen stabilizační algoritmus, který nepředpokládá znalost celého stavu, a následně doplněn o jednoduché řízení, díky kterému je možné s dronem létat. Všechny tyto regulace jsou navrženy

jak pro kvadrokoptéru tak i pro omnikoptéru.

Poslední část se věnuje reálné konstrukci dronu, kde se využívá modulární stavebnicová konstrukce pro stavbu dronů. Tato konstrukce bude v budoucnu použita pro reálnou aplikaci zde navržených řídicích systémů. Z tohoto důvodu je připraveno zapojení také pro řídicí systém REX, kterým bude celý dron řízen.

Důležitou součástí práce je i virtuální 3D model, na kterém je možné v reálném čase zkoumat navržené řídicí algoritmy a simulovat řízení dronu z pohledu pilota. Tato 3D simulace je zakomponována do *Simulinku*, přičemž 3D modely dronů byly vytvořeny v programu Blender a následně exportovány do vhodného formátu. Pro snadnější ovládání je k *Simulinku* zapojen pomocí speciálního bloku gamepad, díky kterému se dá model ovládat v reálném čase.

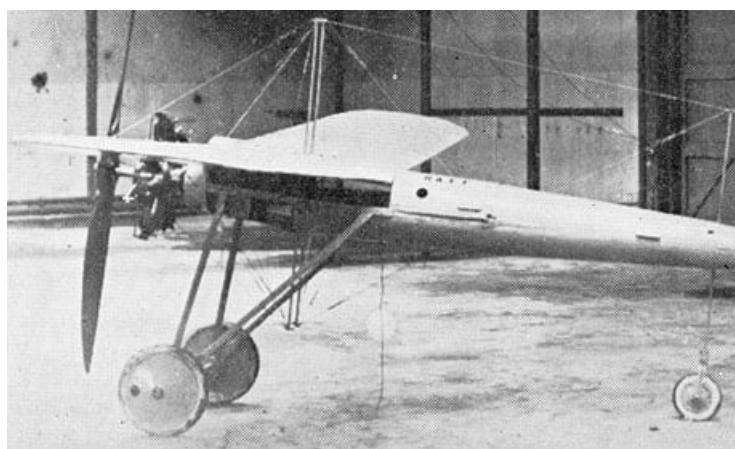
1 UAV - rozdělení, konstrukce a řídicí systémy

Tato část je věnována základním pojmům z oblasti bezpilotních letadel. Také je zaměřena pozornost na jejich konstrukci a použití. Nedílnou součástí je také základní rozdělení a prozkoumání možností řízení.

1.1 Unmanned Aerial Vehicles

Unmanned Aerial Vehicle (zkráceně UAV) je anglický název pro bezpilotní letouny nebo též známějšího ekvivalentu - drony. Jedná se v podstatě o jakékoliv letadlo, které neobsahuje posádku a řízení se provádí pomocí dálkového řízení, nebo je v UAV implementován složitější autonomní řídicí systém. Díky tomuto systému může být následně UAV využito na nejrůznější aplikace, při kterých není nutné zásah lidského operátora. Drony využívají aerodynamické síly ke vzletu, které mohou být generovány různými rotory nebo tryskami.

Pod názvem UAV (či dron) si lze tedy představit celou škálu různých letadýlek až po větší armádní letouny. Rotory, trysky či jakékoliv jiné generátory sil, které jsou schopny UAV vznést do vzduchu, mohou být také různým způsobem umístěny. Za první bezpilotní letoun lze považovat vynález profesora Archibalda Montgomeryho Lowa, který v roce 1916 sestrojil letoun pod názvem „*Aerial Target*“ (Vzdušný cíl).



Obrázek 2: *Aerial Target* - dobová fotografie - [1]

Další vývoj bezpilotních letadel následoval hlavně v USA a to díky probíhající první světové válce. Mezi další známé patřil i například „*Hewitt-Sperry Automatic Airplane*“. Bepilotní letadla prodělaly určitý vývoj a nyní převládá snaha upustit od klasických tvarů konstrukcí letadel a zajistit stabilnější a lépe ovladatelné konstrukce. Proto je tato práce zaměřena především na UAV, které mají rotory umístěny ve vertikální poloze. Takováto konstrukce zajišťuje mnohem přesnější sledování trajektorie a především možnost levitace, která je u některých aplikací kritická.

1.1.1 Aplikace UAV

Nasazování bezpilotních letadel se v dnešní době těší stále větší oblibě. I pokud se jedná o civilní aplikace, kde jsou drony využívány především na pořizování snímků a videí nebo jako modely na létání. Jsou to především malé rozměry a vysoká schopnost manévrovat, díky čemuž jsou UAV schopné se dostat do těžko přístupných míst.

Na základní rozdělení sektorů, kde se drony využívají, se dá dívat různými pohledy. Zde budou uvedeny pouze ty základní. Jak již bylo zmíněno, v dnešní době si drony našly oblibu především v civilních aplikacích.

- **Civilní aplikace**

Zde nacházejí drony uplatnění především ve filmové a fotografické tvorbě a také v modelářství. Díky kompaktním rozměrům jste schopni drona přenášet i v malém batohu, ze kterého ho lze během pár minut připravit k létání a pořizování snímků. Je kladen především důraz na pořizovací náklady a proto zde najdeme především drony s kratšími dolety, které jsou dány hlavně nízkou kapacitou baterií. Také množství použitých senzorů zde pokulhává. Jedná se především o akcelerometry nebo gyroskopy a v lepších případech i o GPS navigace, které se používají hlavně pro automatické návraty a umožňují lepší stabilizaci na základě informace o poloze. Samozřejmě v případě GPS navigace nelze mluvit o přesnosti v cm nebo dokonce v mm.

Samozřejmě zde nelze mluvit o autonomních dronech. Tyto drony jsou především konstruovány pro dálkové ovládání, i když některé modely disponují i několika zajímavými funkcemi. Jedná se například o automatické sledování osob, kdy dron díky kameře rozpoznává pohyb člověka a toho pak sleduje. Toto využívají například lyžaři nebo snowboardisté, kteří díky této funkci mohou například analyzovat styl a nedostatky sjezdu. Nebo také mají předdefinovanou trajektorii letu, během které samy například obkrouží budovu a pořídí zajímavé snímky.

Jedna z nejnámějších firem je *DJI*, která se specializuje na civilní drony a nabízí různou variaci modelů dle funkcí a použití.



Obrázek 3: *DJI Mavic Air* - [2]

- **Průmyslové aplikace**

Obliba dronů roste i v průmyslových aplikacích. Od charakteru úkonů, které mají být vykonávány, se i odvíjí typ UAV. Z větší části se využívají kvalitnější modely z civilních aplikací. Aplikace v průmyslu nabízí nepřehledné množství nejrůznějších úkonů. Lze s nimi monitorovat nedostupná místa, měřit s nimi koncentrace plynů, pořizovat různé kartografické údaje, měřit intenzitu signálů před stavbou přijímací antény, přepravovat různě velké náklady a tak dále.

Zde je již velký prostor i pro autonomní aplikace. Ty mohou spočívat v různých již zmiňovaných úkonech, které ovšem dron provádí zcela samostatně, a následně jen odešle zprávu o výsledku. Nabízí se zde například kontroly kritických a těžko dostupných míst v průmyslových halách, kde by drony byly schopny neustále monitorovat a měřit nejrůznější místa a součástky. Za zmínku určitě stojí i autonomní drony, které jsou schopny stavět menší mosty a stavby, nebo umísťovat nejrůznější zařízení do pro člověka nedostupných míst.

Samozřejmě většina velice zajímavých aplikací je stále ve vývoji, ale jak současný trend napovídá, tak v budoucnu budou mít UAV nepřehledné množství využití.

- **Vojenské aplikace**

Asi nejzajímavější aplikace vzhledem k náročnosti řídicích systémů nabízí právě vojenské využití. V tomto případě se využívají spíše klasické konstrukce trupu letadel a to především kvůli rychlosti, která je u vojenských aplikacích velice důležitá.

Nejčastěji se vojenské drony využívají především k průzkumným misím, kde je důležitá nenápadnost a rychlost. Ovšem čím dál tím více je snaha o umísťování nejrůznějších zbraní na drony, kde takovýto dron, který je vybaven kamerou a puškou, je schopen zaměřit konkrétní hledanou osobu a eliminovat ji, nebo i malé samohybné miny, kdy se na malého drona umístí nálož a ta je ve správný čas odpálena.

V případě vojenských aplikací platí ovšem přísné utajení, takže tuto oblast není možné úplně celou zmapovat. Mezi nejznámější vojenské UAV pak například patří *MQ-1B Predator*, který patří *U.S. Air Force*.



Obrázek 4: *MQ-1B Predator* - [3]

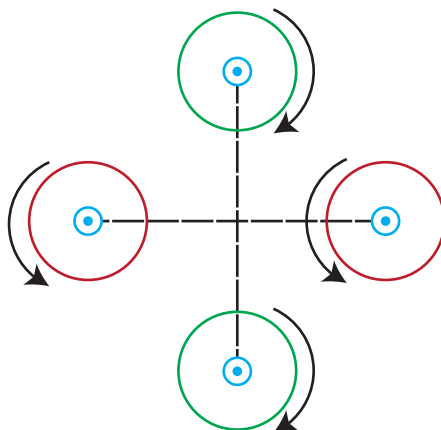
V tomto případě se nejedná o žádný malý dron, jeho rozpětí křídel je 14,8 m a je schopen unést až 500 kg zátěže. Jeho maximální rychlost je 218 km/h. Nese dvojici protiletadlových a dvojici protitankových laserově naváděných raket. Další vybavení je samozřejmě možné dle charakteru mise dodat.

1.2 Konstrukce UAV

Zde bude věnována pozornost UAV s klasickými rotory, které jsou ve vertikální poloze. Jelikož je drtivá většina dronů velice podobných, budou zde popsány pouze některé běžné nebo zajímavější konstrukce.

1.2.1 Kvadroptéra - statické rotory

Jeden z nejzákladnějších konstrukcí dronů, kde se celé tělo skládá ze 4 rotorů, které jsou pevně umístěny většinou na vrcholech čtverce, je klasická kvadroptéra. Všechny rotory působí silově směrem vzhůru. Je zde ovšem nutné eliminovat působící točivý moment, které rotory generují. Proto se vždy dvojice rotorů otáčí jedním směrem a druhá dvojice opět opačným směrem.

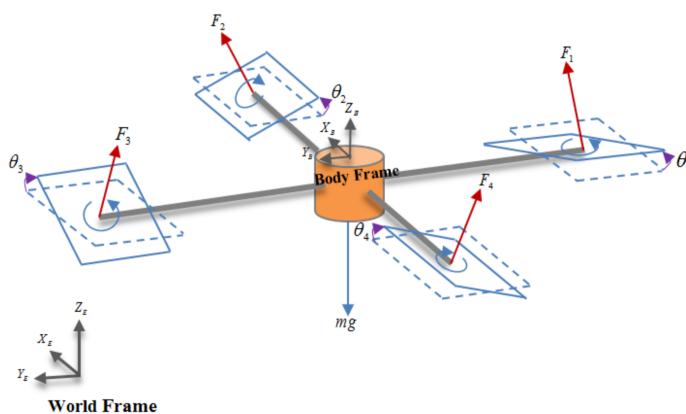


Obrázek 5: Kvadroptéra se statickými rotory

Tento způsob zajišťuje dobrou možnost levitace a pohybu ve vertikální ose. Avšak pokud by se měl dron začít pohybovat horizontálně, nastává problém. Pro horizontální pohyb je nutné dron natočit. Není s ním tedy možné přesně sledovat požadovaný úhel při nulové rychlosti nebo požadovanou trajektorii bez natočení, což ale pro většinu aplikací není překážkou.

1.2.2 Kvadroptéra - dynamické rotory

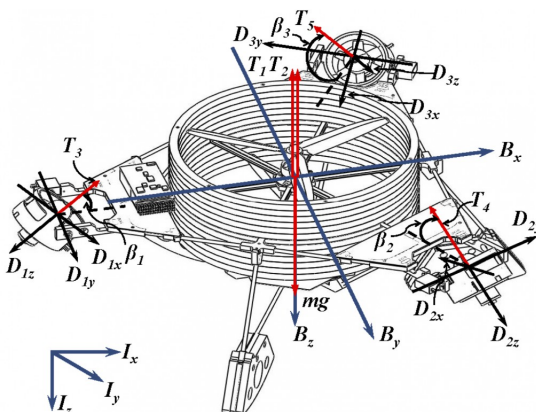
Tento způsob se od předešlého liší pouze tím, že každý rotor je umístěn na kloubu, který je schopen daný rotor natáčet potřebným směrem. Tato konstrukce výrazně vylepší manévrovací schopnosti ovšem za cenu mnohem složitějšího řídicího algoritmu, kde je mimo úhlové rychlosti nutné řídit i natočení rotorů.



Obrázek 6: Kvadroptéra s otočnými rotory - [4]

1.2.3 Kvadroptéra - speciální případ

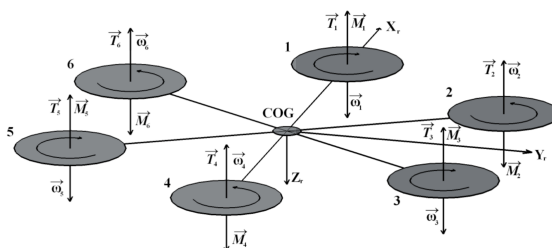
Při návrhu konstrukce dronu se není nutné omezovat na klasické čtvercové uspořádání rotorů, ale lze je v podstatě libovolně umístit. Důležité je však brát v úvahu vhodnost takovéto konfigurace. Například velice zajímavou konstrukcí se zabývá *Multi-Scale Robotics and Automation Lab - Purdue Robotics*, kde navrhli UAV, které má jeden velký statický rotor a další tři menší, kterými je již možné hýbat.



Obrázek 7: Kvadroptéra - Purdue Robotics - [5]

1.2.4 Hexakoptéra

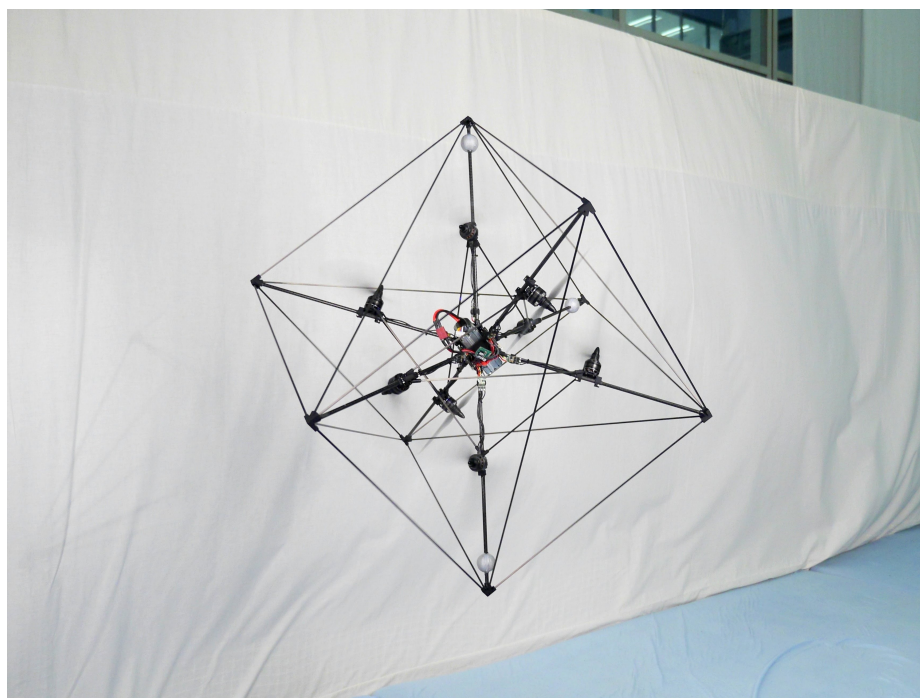
Hexakoptéra je jen modifikace klasické kvadroptéry, kde jsou přidány další 2 rotory. Podobně je tomu i u jiných modifikacích. Lze přidat libovolný počet rotorů, ovšem je vhodné přidávat rotory po dvojicích, aby byla lépe vyřešena eliminace točivého momentu. Přidáním dalších aktuátorů lze dosáhnout lepších vlastností. Nevýhodou je, že dochází k nejednoznačnosti při výpočtu řízení a pak je nutné vybírat z nekonečné množiny řízení to optimální. K tomu dochází, když má systém větší počet aktuátorů než je stupňů volnosti.



Obrázek 8: Hexakoptéra - [6]

1.2.5 Omnikoptéra

Jak již název napovídá, jedná se o všesměrové UAV, kde bude možný pohyb všemi směry i natáčení v libovolné ose. Pro konfiguraci rotorů je tedy důležité, aby bylo následně možné se s dronem pohybovat bez omezení. Těchto konfigurací existuje celá řada, ovšem zde bude pozornost věnována článku [13], který právě jedno takové všesměrové UAV popisuje. Jedná se zde o krychli, ve které je na diagonálách umístěno celkem 8 rotorů, které jsou natočeny různými směry. Vždy platí, že dvojice, která je umístěna na stejné diagonále, má stejný směr působení síly.

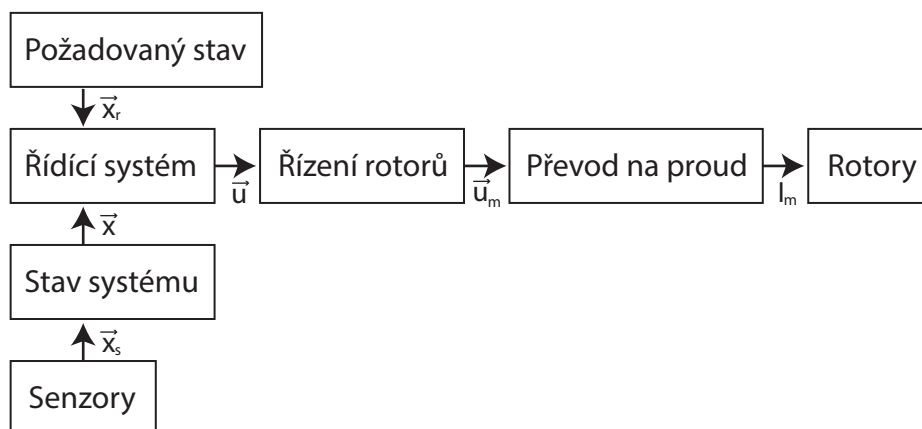


Obrázek 9: Omnikoptéra - [13]

Tato konfigurace umožňuje libovolný pohyb všemi směry i natáčení v libovolném úhlu, což může být v některých aplikacích naprosto zásadní. Podrobnějšímu popisu této omnikoptéry bude věnována samostatná kapitola i s matematickým modelem.

1.3 Použitý řídicí systém

Každý řídicí systém je jedinečný, proto je zcela nemožné popsat univerzální řídicí systém, který by správně fungoval u všech UAVs. Řídicí systémy se liší už jen tím, co se od daného UAV očekává. V této části bude proto popsán nejčastěji používaný koncept řídicího schématu, který má drtivá většina UAVs stejná.



Obrázek 10: Základní schéma řídicího systému UAV

Předchozí obrázek zjednodušeně popisuje klasický řídicí systém běžného UAV. Toto schéma se skládá v podstatě ze 2 částí. V první části se nějakým způsobem získá informace o UAV - stav systému. Ve druhé části pak probíhá samotné řízení. Značení je pak následující:

- \vec{x} je vektor stavu systému, kde jsou údaje o poloze, natočení a jejich derivacích
- \vec{x}_r udává požadovaný stav, který má být dosažen řídicím systémem
- \vec{x}_s sdružuje všechny údaje ze senzorů, které jsou následně použity k určení současného stavu systému
- \vec{u} je řízení, které je výstupem řídicího systému a je využito jako referenční signál pro řídicí systém rotorů, kde je zadávána požadovaná úhlová rychlost
- \vec{u}_m udává výstup řídicího systému rotorů, který je výsledkem na požadovanou úhlovou rychlost jednotlivých rotorů
- I_m je proud, díky kterému se rotory točí požadovanou úhlovou rychlostí

Největším problémem je samozřejmě stav systému. Ten není přesně znám, jelikož senzory nemohou přesně měřit polohu a rychlost translace, proto se stav UAV vždy odhaduje. K odhadu se využívá nejrůznějších metod. Ve většině případů se používají různé modifikace metody nejmenších čtverců nebo Kalmanova filtru. V případě reálných aplikací je tato část naprosto zásadní, jelikož bez správné informace o stavu systému není možné tento systém správně řídit.

Řídicí systém je pak složen z dalších několika částí. V první je vypočten řídicí signál, který většinou představuje požadovanou hodnotu úhlových rychlostí jednotlivých rotorů. Další částí jsou pak řídicí systémy jednotlivých rotorů, které zpracují požadavky na úhlové rychlosti a na základě toho zásobují jednotlivé rotory příslušným řídicím proudem. Opět záleží na aplikaci UAV, většinou se ale využívají stavové zpětné regulátory pro stabilizaci a různé modifikace PID regulátorů pro pohyb v prostoru. U levnějších variant se pro stabilizaci využívají informace

pouze z translačního zrychlení a z úhlových rychlostí. Díky tomu je UAV s jistou přesností schopno setrvat v ustáleném stavu, nebo se velice nepatrně natáčet či pohybovat. Samozřejmě dražší modely již disponují například GPS nebo mnohem přesnějšími senzory polohy a tak jsou schopny mnohem přesnější práce a především samostatného pohybu a plnění úkolů.

2 Matematický model

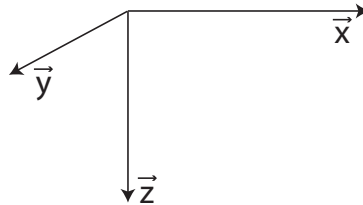
Zde bude pozornost zaměřena na matematický model UAV. Nejdříve jsou odvozeny všechny potřebné vztahy, které jsou kritické pro správné sestavení matematického popisu systému. Celkem jsou vytvořeny dva modely. Jeden model je klasická kvadrokoptéra s pevně umístěnými rotory a jako druhý je model omnikoptéry, která je navržena dle článku [13].

K odvození je využita Euler-Newtonova metoda, která využívá základní Newtonovy a Eulerovy vztahy pro působení sil a momentů na tělesa. Pro zjednodušení jsou uvažovány dvě soustavy souřadnic - inerciální a pohyblivá soustava pevně spojená s tělesem. Pro převod je využívána tzv. matice rotace. Odvození je nejdříve provedeno s pomocí Eulerových úhlů, které jsou pro představu přijatelnější, ovšem s jejich použitím mohou nastávat problémy, proto je celý model převeden do tzv. kvaternionů.

Tato kapitola také obsahuje jednotlivá odvození pro působení sil a momentů, které generují rotory. Dále je uveden postup pro experimentální určení momentu setrvačnosti tělesa.

2.1 Pravoúhlá soustava souřadnic

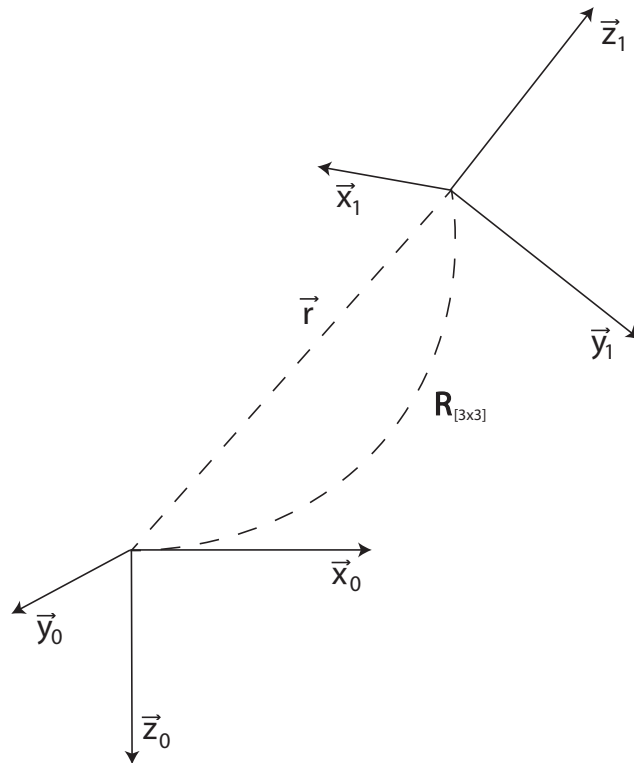
Pro tuto práci byla zvolena pravoúhlá soustava souřadnic, tedy soustava souřadnic ve tvaru $\vec{x}\vec{y}\vec{z}$. Jelikož se tato práce zabývá omni-směrovým bezpilotním letadlem, je celá soustava souřadnic otočena směrem dolů. Takto situovaná soustava souřadnic se využívá především v letectví, kde má své opodstatnění.



Obrázek 11: Pravoúhlá soustava souřadnic

2.2 Pevná a pohyblivá soustava souřadnic

V mnoha případech je vhodné použít pevnou (inerciální) a pohyblivou soustavu souřadnic, která se pohybuje vůči inerciální soustavě. Převod mezi soustavami je pak dán tzv. maticí rotace. Tento způsob mnohé fyzikální vztahy zjednoduší. Pokud se síly, které skrze rotory působí, uvažují v pohyblivé soustavě pevně spojené s tělesem, pak tyto síly působí neustále konstantním směrem.



Obrázek 12: Vztah mezi soustavami

Soustava souřadnic $\vec{x}_0\vec{y}_0\vec{z}_0$ se pomocí vektoru \vec{r} přesunula do nové pozice a pomocí matice rotace \mathbf{R} změnila svou orientaci. Tímto vznikla nová soustava souřadnic $\vec{x}_1\vec{y}_1\vec{z}_1$ (Obrázek 12). Je vhodné poznamenat, že v této práci je uvažováno $\vec{r} = 0$, tedy obě dvě soustavy souřadnic sdílejí stejný počátek.

2.3 Matice rotace R

Matice rotace popisuje rotaci soustavy souřadnic do nové polohy určené nově vzniklým souřadným systémem. Libovolný vektor \vec{i} lze vyjádřit v obou soustavách, tedy

$$\vec{i} = a_0\vec{x}_0 + b_0\vec{y}_0 + c_0\vec{z}_0 = a_1\vec{x}_1 + b_1\vec{y}_1 + c_1\vec{z}_1, \quad (2.1)$$

kde prvky s dolním indexem 0 reprezentují původní soustavu souřadnic $\vec{x}_0\vec{y}_0\vec{z}_0$ a prvky s dolním indexem 1 reprezentují jinou soustavu souřadnic $\vec{x}_1\vec{y}_1\vec{z}_1$.

Úpravou rovnice (2.1), kde jsou jednotlivé vektory postupně vynásobeny, je získán následující tvar

$$a_0(\vec{x}_0, \vec{x}_0) + b_0(\vec{y}_0, \vec{x}_0) + c_0(\vec{z}_0, \vec{x}_0) = a_1(\vec{x}_1, \vec{x}_0) + b_1(\vec{y}_1, \vec{x}_0) + c_1(\vec{z}_1, \vec{x}_0), \quad (2.2)$$

$$a_0(\vec{x}_0, \vec{y}_0) + b_0(\vec{y}_0, \vec{y}_0) + c_0(\vec{z}_0, \vec{y}_0) = a_1(\vec{x}_1, \vec{y}_0) + b_1(\vec{y}_1, \vec{y}_0) + c_1(\vec{z}_1, \vec{y}_0), \quad (2.3)$$

$$a_0(\vec{x}_0, \vec{z}_0) + b_0(\vec{y}_0, \vec{z}_0) + c_0(\vec{z}_0, \vec{z}_0) = a_1(\vec{x}_1, \vec{z}_0) + b_1(\vec{y}_1, \vec{z}_0) + c_1(\vec{z}_1, \vec{z}_0). \quad (2.4)$$

Rovnice (2.2), (2.3) a (2.4) lze vyjádřit v maticovém tvaru, tedy

$$\underbrace{\begin{bmatrix} a_0 \\ b_0 \\ c_0 \end{bmatrix}}_{\vec{i}_0} = \underbrace{\begin{bmatrix} (\vec{x}_1, \vec{x}_0) & (\vec{y}_1, \vec{x}_0) & (\vec{z}_1, \vec{x}_0) \\ (\vec{x}_1, \vec{y}_0) & (\vec{y}_1, \vec{y}_0) & (\vec{z}_1, \vec{y}_0) \\ (\vec{x}_1, \vec{z}_0) & (\vec{y}_1, \vec{z}_0) & (\vec{z}_1, \vec{z}_0) \end{bmatrix}}_{\mathbf{R}} \underbrace{\begin{bmatrix} a_1 \\ b_1 \\ c_1 \end{bmatrix}}_{\vec{i}_1}, \quad (2.5)$$

kde \vec{i}_0 je vektor \vec{i} vyjádřený v soustavě souřadnic $\vec{x}_0\vec{y}_0\vec{z}_0$, \vec{i}_1 je vektor \vec{i} vyjádřený v soustavě souřadnic $\vec{x}_1\vec{y}_1\vec{z}_1$, \mathbf{R} je daná matice rotace a a_n, b_n, c_n jsou příslušné složky vektoru \vec{i}_n , kde n je celé číslo, které jednoznačně identifikuje danou soustavu souřadnic. Lze si všimnout, že matice \mathbf{R} je ortogonální a tedy

$$\mathbf{R}^{-1} = \mathbf{R}^T. \quad (2.6)$$

Eulerova věta

Jakákoliv změna orientace tuhého tělesa může být dosažena rotací tělesa kolem jediné osy procházející libovolně zvoleným referenčním bodem.

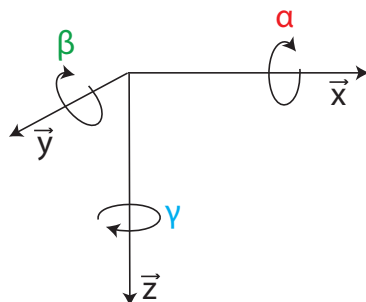
Nyní již bude odvozen celkový tvar matice rotace \mathbf{R} . Nejdříve budou rozepsány jednotlivé rotace okolo osy \vec{x} , \vec{y} a \vec{z} , tedy

$$R_1(\alpha) = \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{bmatrix}}_{\text{rotace okolo osy } \vec{x}}, \quad (2.7)$$

$$R_2(\beta) = \underbrace{\begin{bmatrix} \cos(\beta) & 0 & \sin(\beta) \\ 0 & 1 & 0 \\ -\sin(\beta) & 0 & \cos(\beta) \end{bmatrix}}_{\text{rotace okolo osy } \vec{y}}, \quad (2.8)$$

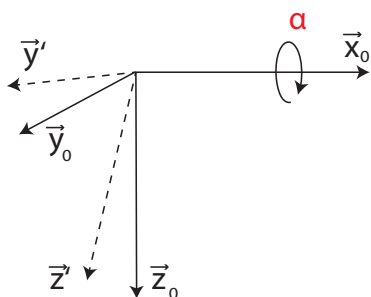
$$R_3(\gamma) = \underbrace{\begin{bmatrix} \cos(\gamma) & -\sin(\gamma) & 0 \\ \sin(\gamma) & \cos(\gamma) & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{\text{rotace okolo osy } \vec{z}}, \quad (2.9)$$

kde α, β a γ jsou příslušné úhly rotací okolo jednotlivých os (obrázek 13).

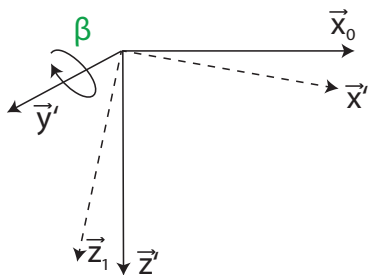


Obrázek 13: Rotace okolo jednotlivých os

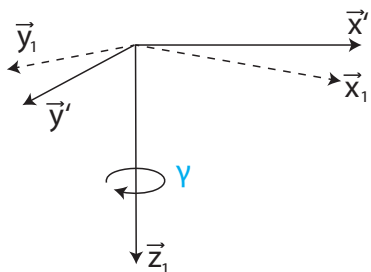
Nyní bude provedena postupná rotace okolo všech tří os. Pro tento případ bude rotováno z inerciální soustavy souřadnic $\vec{x}_0\vec{y}_0\vec{z}_0$ do soustavy pevně spojené s tělesem $\vec{x}_1\vec{y}_1\vec{z}_1$.

Obrázek 14: Rotace - úhel α

Nejdříve bude provedena rotace o úhel α . Při této rotaci dochází ke změně soustavy souřadnic z $\vec{x}_0\vec{y}_0\vec{z}_0$ na $\vec{x}_0\vec{y}'\vec{z}'$. Pro soustavy, které se nacházejí mezi inerciální a pohyblivou, bude zavedeno značení \vec{x} , \vec{y} nebo \vec{z} .

Obrázek 15: Rotace - úhel β

Dále bude provedena rotace o úhel β . Při této transformaci dochází ke změně soustavy souřadnic z $\vec{x}_0\vec{y}'\vec{z}'$ na $\vec{x}\vec{y}'\vec{z}_1$.

Obrázek 16: Rotace - úhel γ

A poslední transformací je rotace o úhel γ . Po rotaci je již transformace kompletní a dochází ke změně soustavy souřadnic z $\vec{x}\vec{y}'\vec{z}_1$ na pohyblivou soustavu souřadnic pevně spojenou s tělesem $\vec{x}_1\vec{y}_1\vec{z}_1$.

Tedy pro libovolný vektor \vec{i} platí

$$\vec{i}_{\vec{x}_1\vec{y}_1\vec{z}_1} = \vec{i}_1 = R_3(\gamma) \vec{i}_{\vec{x}\vec{y}\vec{z}_1}, \quad (2.10)$$

$$\vec{i}_{\vec{x}\vec{y}\vec{z}_1} = R_2(\beta) \vec{i}_{\vec{x}_0\vec{y}\vec{z}}, \quad (2.11)$$

$$\vec{i}_{\vec{x}_0\vec{y}\vec{z}} = R_1(\alpha) \vec{i}_{\vec{x}_0\vec{y}_0\vec{z}_0} = R_1(\alpha) \vec{i}_0, \quad (2.12)$$

což lze psát ve tvaru

$$\vec{i}_1 = \underbrace{R_3(\gamma)R_2(\beta)R_1(\alpha)}_{\mathbf{R}_{0 \rightarrow 1}} \vec{i}_0. \quad (2.13)$$

Výsledná matice rotace je pak získána pomocí rovnic (2.7), (2.8) a (2.9), tedy pro transformaci z inerciální soustavy $\vec{x}_0\vec{y}_0\vec{z}_0$ do pohyblivé soustavy pevně spojené s tělesem $\vec{x}_1\vec{y}_1\vec{z}_1$ je využito matice rotace

$$\mathbf{R}_{0 \rightarrow 1} = R_3(\gamma)R_2(\beta)R_1(\alpha),$$

roznásobeno

$$\mathbf{R}_{0 \rightarrow 1} = \begin{bmatrix} c_\beta c_\gamma & -c_\alpha s_\gamma + s_\alpha s_\beta c_\gamma & s_\alpha s_\gamma + c_\alpha s_\beta c_\gamma \\ c_\beta s_\gamma & c_\alpha c_\gamma + s_\alpha s_\beta s_\gamma & -s_\alpha c_\gamma + c_\alpha s_\beta s_\gamma \\ -s_\beta & s_\alpha c_\beta & c_\alpha c_\beta \end{bmatrix}, \quad (2.14)$$

kde $s_i = \sin(i)$ a $c_i = \cos(i)$.

Naopak pro převod z pohyblivé soustavy pevně spojené s tělesem do inerciální soustavy je vztah dán

$$\begin{aligned} \mathbf{R}_{1 \rightarrow 0} &= \mathbf{R}_{0 \rightarrow 1}^{-1} = \mathbf{R}_{0 \rightarrow 1}^T \\ &= (R_3(\gamma)R_2(\beta)R_1(\alpha))^{-1} \\ &= R_1^T(\alpha)R_2^T(\beta)R_3^T(\gamma), \end{aligned}$$

roznásobeno

$$\mathbf{R}_{1 \rightarrow 0} = \begin{bmatrix} c_\beta c_\gamma & c_\beta s_\gamma & -s_\beta \\ -c_\alpha s_\gamma + s_\alpha s_\beta c_\gamma & c_\alpha c_\gamma + s_\alpha s_\beta s_\gamma & s_\alpha c_\beta \\ s_\alpha s_\gamma + c_\alpha s_\beta c_\gamma & -s_\alpha c_\gamma + c_\alpha s_\beta s_\gamma & c_\alpha c_\beta \end{bmatrix}. \quad (2.15)$$

Pro pozdější výpočty je zavedeno následující značení:

$$\mathbf{R} = \mathbf{R}_{1 \rightarrow 0}, \quad (2.16)$$

$$\mathbf{R}^T = \mathbf{R}_{0 \rightarrow 1}. \quad (2.17)$$

2.4 Newton-Eulerova metoda

Bude využito následujících principů:

1. **Eulerův 1. pohybový zákon** vychází z 2. Newtonova zákona a popisuje změnu hybnosti tělesa, která je rovna celkové síle \vec{F} působící na těleso, tedy

$$\frac{d(m\vec{v})}{dt} = \vec{F}, \quad (2.18)$$

kde m je hmotnost tělesa, \vec{v} je vektor rychlosti tělesa a \vec{F} je celková síla působící na těleso.

2. **Eulerův 2. pohybový zákon** je také analogií k 2. Newtonovu zákonu a zabývá se časovou změnou momentu hybnosti, která je rovna celkovému momentu působící na těleso, tedy

$$\frac{d(J\vec{\omega})}{dt} = \vec{\tau}, \quad (2.19)$$

kde J je moment setrvačnosti tělesa (matice setrvačnosti), $\vec{\omega}$ je úhlová rychlost tělesa a $\vec{\tau}$ je celkový moment působící na těleso.

3. **Newtonův 3. zákon** se také nazývá Zákon akce a reakce. Pokud těleso A působí silou \vec{F}_A a momentem $\vec{\tau}_A$ na těleso B , tak současně těleso B působí na těleso A silou

$$\vec{F}_B = -\vec{F}_A \quad (2.20)$$

a momentem

$$\vec{\tau}_B = -\vec{\tau}_A. \quad (2.21)$$

4. **Princip superpozice** Newton definoval jako doplněk svých 3 zákonů, proto bývá někdy označován jako 4. Newtonův zákon. Pokud bude na hmotný bod působit například N sil, tak výslednice všech sil bude vypočtena jako

$$\vec{F} = \vec{F}_1 + \vec{F}_2 + \vec{F}_3 + \dots + \vec{F}_N = \sum_{i=1}^N \vec{F}_i. \quad (2.22)$$

Celkovou sílu působící na těžiště tělesa lze díky principu superpozice (2.22) napsat jako

$$\vec{F} \triangleq \sum_{i=1}^N \vec{F}_i = \sum_{i=1}^N m_i \dot{\vec{v}}_i. \quad (2.23)$$

Celkový moment působící na těžiště tělesa lze tedy také vyjádřit jako

$$\vec{\tau} \triangleq \sum_{i=1}^N \vec{\tau}_i = \sum_{i=1}^N \vec{\rho}_i \times \vec{F}_i = \frac{d}{dt} \left[\sum_{i=1}^N (\vec{\rho}_i \times m_i \dot{\vec{\rho}}_i) \right] = \frac{d}{dt} \vec{L}, \quad (2.24)$$

kde \vec{L} je celkový moment hybnosti tělesa, $\vec{\rho}_i$ je vektor spojující těžiště tělesa s jednotlivými body, kde působí příslušné momenty.

Poznámka:

Newtonovy rovnice jsou platné pouze pro inerciální soustavu souřadnic. Ovšem je možné je vyjádřit pomocí matice rotace v jakékoliv jiné soustavě souřadnic.

Hlavní problém rovnice (2.19) je časová variace matice momentu setrvačnosti J , pro který platí

$$J_0 = \mathbf{R}J_1\mathbf{R}^T, \quad (2.25)$$

kde \mathbf{R} je příslušná matice rotace (2.16), J_0 je moment setrvačnosti vyjádřený v inerciální soustavě souřadnic a J_1 je moment setrvačnosti v soustavě souřadnic spojené s tělesem (pohyblivá soustava).

2.4.1 Derivace matice rotace \mathbf{R}

Jelikož není matice rotace \mathbf{R} v čase invariantní, je nutné odvodit vztah pro její derivaci v čase.

Pro každý vektor vyjádřený v inerciální soustavě platí:

$$\vec{\xi}_0 = \mathbf{R}\vec{\xi}_1. \quad (2.26)$$

Derivace libovolného vektoru vyjádřeného v inerciální soustavě je dána vztahem

$$\dot{\vec{\xi}}_0 = \dot{\mathbf{R}}\vec{\xi}_1 + \mathbf{R}\dot{\vec{\xi}}_1, \quad (2.27)$$

jelikož je $\vec{\xi}_1$ časově invariantní, lze tedy psát

$$\dot{\vec{\xi}}_0 = \dot{\mathbf{R}}\vec{\xi}_1 = \dot{\mathbf{R}}\mathbf{R}^T\vec{\xi}_0. \quad (2.28)$$

Obecně platí, že

$$\dot{\vec{\xi}}_0 = \vec{\omega}_0 \times \vec{\xi}_0. \quad (2.29)$$

Vektorový součin lze též vyjádřit maticově ve tvaru

$$\vec{a} \times \vec{b} = \underbrace{\begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix}}_{\hat{a}} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}. \quad (2.30)$$

S využitím vztahu (2.30) lze vztah (2.29) upravit jako

$$\dot{\vec{\xi}}_0 = \hat{\omega}_0\vec{\xi}_0, \quad (2.31)$$

kde

$$\hat{\omega}_0 = \begin{bmatrix} 0 & -\omega_{z_0} & \omega_{y_0} \\ \omega_{z_0} & 0 & -\omega_{x_0} \\ -\omega_{y_0} & \omega_{x_0} & 0 \end{bmatrix}. \quad (2.32)$$

S využitím vztahů (2.28) a (2.31) lze získat již finální tvar pro změnu matice rotace (2.16) v závislosti na čase:

$$\begin{aligned}\widehat{\omega}_0 \vec{\xi}_0 &= \dot{\mathbf{R}} \mathbf{R}^T \vec{\xi}_0, \\ \widehat{\omega}_0 &= \dot{\mathbf{R}} \mathbf{R}^T, \\ \widehat{\omega}_0 \mathbf{R} &= \dot{\mathbf{R}}, \\ \dot{\mathbf{R}} &= \vec{\omega}_0 \times \mathbf{R} = \widehat{\omega}_0 \mathbf{R}.\end{aligned}\tag{2.33}$$

Tento vztah lze stejnou úpravou převést i na vztah pro $\dot{\mathbf{R}}^T$. Tedy je získán vztah

$$\begin{aligned}\widehat{\omega}_1 \vec{\xi}_1 &= \dot{\mathbf{R}}^T \mathbf{R} \vec{\xi}_1, \\ \widehat{\omega}_1 &= \dot{\mathbf{R}}^T \mathbf{R}, \\ \widehat{\omega}_1 \mathbf{R}^T &= \dot{\mathbf{R}}^T, \\ \dot{\mathbf{R}}^T &= \vec{\omega}_1 \times \mathbf{R}^T = \widehat{\omega}_1 \mathbf{R}^T.\end{aligned}\tag{2.34}$$

2.4.2 Vztah mezi úhly a úhlovou rychlostí - matice W

Vztahy pro převod mezi jednotlivými soustavami jsou již definovány maticí rotace \mathbf{R} (2.16). Pozornost bude dále zaměřena na vztah mezi úhlovou rychlostí pohyblivé soustavy a derivací úhlů vyjádřených v inerciální soustavě. Pro tento vztah se využívá podobná matice jako v případě převodu vektorů mezi soustavami.

Jelikož jsou uvažovány dvě soustavy souřadnic, bude odvozen vztah pro převod z pohyblivé do inerciální soustavy. Lze provést i převod mezi úhlovou rychlostí v inerciální soustavě souřadnic a derivací úhlů též v inerciální soustavě souřadnic, ovšem v tomto případě bude výhodnější provést zároveň i převod mezi soustavami. Obecně neplatí rovnost mezi derivací úhlů, která je uvažována v inerciální soustavě souřadnic, a úhlovou rychlostí, která je naopak uvažována v pohyblivé soustavě souřadnic pevně spojené s tělesem. Z tohoto důvodu je nutné definovat transformační matice mezi těmito veličinami.

Převod mezi inerciální soustavou a pohyblivou soustavou souřadnic je realizován pomocí matice rotace (2.14), která je získána z celkem tří rotací - $R_3(\gamma)R_2(\beta)R_1(\alpha)$. Pro tento výpočet bude využit vztah (2.34), ze kterého vyplývá, že

$$\widehat{\omega}_1 = \dot{\mathbf{R}}^T \mathbf{R}.\tag{2.35}$$

S využitím vztahu (2.32) vztaženým pro pohyblivou soustavu souřadnic pevně spojenou s tělesem lze psát

$$\widehat{\omega}_1 = \begin{bmatrix} 0 & -\omega_{z_1} & \omega_{y_1} \\ \omega_{z_1} & 0 & -\omega_{x_1} \\ -\omega_{y_1} & \omega_{x_1} & 0 \end{bmatrix} = \dot{\mathbf{R}}^T \mathbf{R}.\tag{2.36}$$

Nyní stačí vyjádřit jednotlivé úhlové rychlosti, ty jsou dány

$$\omega_{x_1} = \cos(\beta)\cos(\gamma)\dot{\alpha} - \sin(\gamma)\dot{\beta}, \quad (2.37)$$

$$\omega_{y_1} = \cos(\gamma)\dot{\beta} + \cos(\beta)\sin(\gamma)\dot{\alpha}, \quad (2.38)$$

$$\omega_{z_1} = \dot{\gamma} - \sin(\beta)\dot{\alpha}, \quad (2.39)$$

maticově zapsáno

$$\vec{\omega}_1 = \begin{bmatrix} \cos(\beta)\cos(\gamma) & -\sin(\gamma) & 0 \\ \cos(\beta)\sin(\gamma) & \cos(\gamma) & 0 \\ -\sin(\beta) & 0 & 1 \end{bmatrix} \underbrace{\begin{bmatrix} \dot{\alpha} \\ \dot{\beta} \\ \dot{\gamma} \end{bmatrix}}_{\dot{\vec{\eta}}_0}. \quad (2.40)$$

Posledním krokem je vyjádření derivace úhlů v inerciální soustavě souřadnic, tedy

$$\dot{\vec{\eta}}_0 = \underbrace{\begin{bmatrix} \frac{\cos(\gamma)}{\cos(\beta)} & \frac{\sin(\gamma)}{\cos(\beta)} & 0 \\ -\sin(\gamma) & \cos(\gamma) & 0 \\ \frac{\sin(\beta)\cos(\gamma)}{\cos(\beta)} & \frac{\sin(\beta)\sin(\gamma)}{\cos(\beta)} & 1 \end{bmatrix}}_{\mathbf{W}} \vec{\omega}_1, \quad (2.41)$$

kde matice \mathbf{W} jasně definuje vztah mezi úhlovou rychlostí vyjádřenou v pohyblivé soustavě souřadnic pevně spojené s tělesem a derivacemi úhlů, které jsou vztaženy pro inerciální soustavu souřadnic.

2.4.3 Síla působící na těleso - vyjádření v pohyblivé soustavě

Síla působící na těleso vyjádřená v inerciální soustavě je dána vztahem (2.23), tedy

$$\vec{F}_0 = m\dot{\vec{v}}_0. \quad (2.42)$$

Jednoduchou úpravou lze získat vztah

$$\dot{\vec{v}}_0 = \frac{1}{m}\vec{F}_0. \quad (2.43)$$

Dále je vztah (2.26) využit na vyjádření rychlosti v pohyblivé soustavě

$$\vec{v}_1 = \mathbf{R}^T \vec{v}_0. \quad (2.44)$$

Pokud je vztah (2.44) zderivován, lze získat

$$\begin{aligned} \dot{\vec{v}}_1 &= \dot{\mathbf{R}}^T \vec{v}_0 + \mathbf{R}^T \dot{\vec{v}}_0, \\ &= \vec{\omega}_1 \times \mathbf{R}^T \vec{v}_0 + \mathbf{R}^T \frac{1}{m} \vec{F}_0, \\ &= \vec{\omega}_1 \times \vec{v}_1 + \frac{1}{m} \vec{F}_1. \end{aligned}$$

Výsledný vztah pro sílu působící na těleso, který je vyjádřen v pohyblivé soustavě pevně spojené se zemí, je dán

$$\vec{F}_1 = m\dot{\vec{v}}_1 - m(\vec{\omega}_1 \times \vec{v}_1). \quad (2.45)$$

2.4.4 Moment hybnosti tělesa - vyjádření v pohyblivé soustavě

Moment hybnosti tělesa vyjádřený v inerciální soustavě, který působí vůči těžišti tělesa, lze psát jako

$$\begin{aligned} \vec{L}_{0i} &= \vec{\rho}_{0i} \times m_i \dot{\vec{\rho}}_{0i}, \\ &= \vec{\rho}_{0i} \times m_i(\vec{\omega}_0 \times \vec{\rho}_{0i}), \end{aligned}$$

s využitím matice rotace (2.14)

$$\begin{aligned} \vec{L}_{0i} &= \mathbf{R}\vec{\rho}_{1i} \times m_i(\mathbf{R}\vec{\omega}_0 \times \mathbf{R}\vec{\rho}_{1i}), \\ &= \mathbf{R}\vec{\rho}_{1i} \times m_i\mathbf{R}(\vec{\omega}_0 \times \vec{\rho}_{1i}), \\ &= \mathbf{R}[\vec{\rho}_{1i} \times m_i(\vec{\omega}_0 \times \vec{\rho}_{1i})], \end{aligned}$$

jelikož je vektorový součin antikomutativní, tedy

$$\vec{a} \times \vec{b} = -\vec{b} \times \vec{a}, \quad (2.46)$$

s využitím vztahů (2.29) a (2.46) lze získat vztah

$$\begin{aligned} \vec{L}_{0i} &= \mathbf{R}[-m_i(\vec{\rho}_{1i} \times (\vec{\rho}_{1i} \times \vec{\omega}_1))], \\ &= \mathbf{R}[-m_i\widehat{\rho}_{1i}(\vec{\rho}_{1i} \times \vec{\omega}_1)], \\ &= \mathbf{R}[-m_i\widehat{\rho}_{1i}\widehat{\rho}_{1i}\vec{\omega}_1], \\ &= \mathbf{R}[\underbrace{-m_i\widehat{\rho}_{1i}\widehat{\rho}_{1i}}_{J_{1i}}]\mathbf{R}^T\vec{\omega}_0. \end{aligned}$$

Moment hybnosti je tedy definován jako

$$\vec{L}_{0i} \triangleq \mathbf{R}J_{1i}\vec{\omega}_1, \quad (2.47)$$

kde

$$J_{1i} = \sum_{i=1}^N m_i \widehat{\rho}_{1i} \widehat{\rho}_{1i} = \sum_{i=1}^N m_i \begin{bmatrix} y_i^2 + z_i^2 & -x_i y_i & -x_i z_i \\ -x_i y_i & x_i^2 + z_i^2 & -y_i z_i \\ -x_i z_i & -y_i z_i & x_i^2 + y_i^2 \end{bmatrix}. \quad (2.48)$$

Celkový moment hybnosti je tedy suma všech momentů hybnosti působících na těžiště tělesa vyjádřený v inerciální soustavě:

$$\vec{L}_0 = \sum_{i=1}^N \vec{L}_{0i} = \sum_{i=1}^N \vec{\rho}_{0i} \times m_i \dot{\vec{\rho}}_{0i} = \sum_{i=1}^N \mathbf{R}J_{1i}\vec{\omega}_1 \triangleq \mathbf{R}J_1\vec{\omega}_1. \quad (2.49)$$

Tedy rovnice (2.24) vyjádřená v inerciální soustavě má následující tvar

$$\vec{\tau}_0 \triangleq \frac{d}{dt}(\mathbf{R}J_1\vec{\omega}_1) = \dot{\mathbf{R}}J_1\vec{\omega}_1 + \mathbf{R}J_1\dot{\vec{\omega}}_1 = \widehat{\omega}_0\mathbf{R}J_1\vec{\omega}_1 + \mathbf{R}J_1\dot{\vec{\omega}}_1. \quad (2.50)$$

S využitím matice rotace (2.15) lze vztah (2.24) převést do pohyblivé soustavy pevně spojené s tělesem:

$$\begin{aligned} \vec{\tau}_1 &= \mathbf{R}^T\vec{\tau}_0 = \mathbf{R}^T\widehat{\omega}_0\mathbf{R}J_1\vec{\omega}_1 + J_1\dot{\vec{\omega}}_1, \\ &= \widehat{(\mathbf{R}^T\omega_0)}J_1\vec{\omega}_1 + J_1\dot{\vec{\omega}}_1, \\ &= \widehat{\omega}_1J_1\vec{\omega}_1 + J_1\dot{\vec{\omega}}_1, \\ &= \vec{\omega}_1 \times (J_1\vec{\omega}_1) + J_1\dot{\vec{\omega}}_1. \end{aligned}$$

Výsledný vztah vyjádřený v pohyblivé soustavě pevně spojené s tělesem má tedy podobu

$$\vec{\tau}_1 = \vec{\omega}_1 \times (J_1\vec{\omega}_1) + J_1\dot{\vec{\omega}}_1. \quad (2.51)$$

V předešlém odvození byla využita identita

$$\mathbf{R}\widehat{\mathbf{a}}\mathbf{R}^T = \widehat{(\mathbf{R}\vec{a})}, \quad (2.52)$$

kterou lze odvodit následujícím způsobem:

$$\begin{aligned} \mathbf{R}\widehat{\mathbf{a}}\mathbf{R}^T\vec{b} &= \mathbf{R}(\vec{a} \times \mathbf{R}^T\vec{b}), \\ \mathbf{R}\widehat{\mathbf{a}}\mathbf{R}^T\vec{b} &= (\mathbf{R}\vec{a}) \times (\mathbf{R}\mathbf{R}^T\vec{b}), \\ \mathbf{R}\widehat{\mathbf{a}}\mathbf{R}^T\vec{b} &= (\mathbf{R}\vec{a}) \times \vec{b}, \\ \mathbf{R}\widehat{\mathbf{a}}\mathbf{R}^T\vec{b} &= \widehat{(\mathbf{R}\vec{a})}\vec{b}, \\ \mathbf{R}\widehat{\mathbf{a}}\mathbf{R}^T &= \widehat{(\mathbf{R}\vec{a})}. \end{aligned}$$

2.5 Rotory

Zde bude pozornost přesunuta na odvození rovnic pro rotory UAV. Pro jednoduchost se předpokládá, že všechny rotory, které UAV obsahuje, jsou identické. Výsledný tah bude tedy jen vektorový součet všech sil jednotlivých rotorů. Následující vztahy jsou převzaty z článku [15].

Nejdříve je nutné určit točivý moment:

$$\tau_m = K_t(I - I_0), \quad (2.53)$$

kde τ_m je točivý moment motoru, K_t je proporcionalní konstanta točivého momentu, I vstupní proud, I_0 je proud bez zátěže,

$$V = IR_m + K_v\omega_m, \quad (2.54)$$

kde V je napětí na motoru, R_m je odpor motoru, K_v je zpětné elektromotorické napětí a ω_m je úhlová rychlost motoru.

$$P_m = IV = \frac{(\tau_m + K_t I_0)(K_t I_0 R_m + \tau_m R_m + K_t K_v \omega_m)}{K_t^2}, \quad (2.55)$$

kde P_m je výkon motoru.

Předpokládá se, že $R_m \approx I_0 \approx 0$, tedy

$$P_m = \frac{K_v \tau \omega_m}{K_t}. \quad (2.56)$$

Výkon potřebný ke vzletu

$$P_h = T v_h, \quad (2.57)$$

kde T je tah rotoru a lze ho vyjádřit jako

$$T = 2\rho_w A_r \nu_h^2, \quad (2.58)$$

kde ρ_w je odpor vzduchu, A_r je plocha, kterou opisuje vrtule rotoru, a ν_h je indukovaná rychlost motoru.

Točivý moment lze napsat i ve tvaru

$$\tau_m = K_\tau T, \quad (2.59)$$

kde K_τ je jiná proporcionalní konstanta točivého momentu. Rovnici (2.55) lze díky předpokladu zjednodušit na

$$P_m = \frac{K_v \tau_m \omega_m}{K_t} = \frac{K_v K_\tau T \omega_m}{K_t}. \quad (2.60)$$

Výkon, který je potřeba ke vzletnutí je definován vztahem

$$P_h = \frac{T^{3/2}}{\sqrt{2\rho_w A_r}}. \quad (2.61)$$

Pokud jsou tedy dány rovnice (2.60) do rovnosti s (2.61) je získán vztah

$$\frac{K_v K_\tau T \omega_m}{K_t} = \frac{T^{3/2}}{\sqrt{2\rho_w A_r}}. \quad (2.62)$$

Tedy celkový tah rotoru lze vypočítat jako

$$T = \left(\underbrace{\left(\frac{K_v K_\tau \sqrt{2\rho_w A_r}}{K_t} \right)}_{K_T^{1/2}} \omega_m \right)^2, \quad (2.63)$$

výsledný a zjednodušený tvar, který bude v tomto případě použit, má tvar

$$T = K_T \omega_m^2. \quad (2.64)$$

S využitím vztahu (2.64) a (2.59) lze točivý moment rotoru psát jako

$$\tau_m = \underbrace{K_\tau K_T}_{K_{\tau\tau}} \omega_m^2. \quad (2.65)$$

Konstanty K_T a $K_{\tau\tau}$ se určují simulačně.

2.6 UAV - působící síla a moment

Pokud bude uvažován pouze zjednodušený model, kde je uvažována soustava složená z hmotných bodů, která má definovanou matici setrvačnosti, lze pak celý problém převést pouze na působení tahů a momentů jednotlivých rotorů v různých směrech a orientacích. Následující rovnice byly získány z článku [13] a [18].

Celková síla, která je generována všemi rotory, je definována vztahem

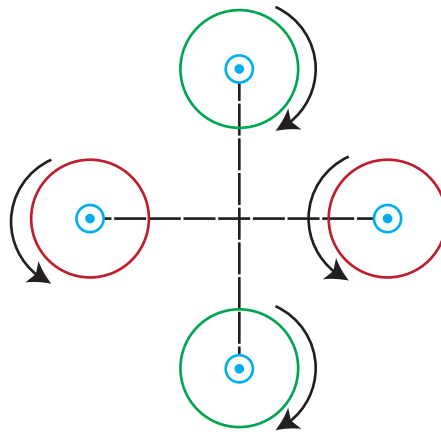
$$\vec{F}_r = \sum_{i=1}^N T_i \vec{\chi}_i, \quad (2.66)$$

kde \vec{F}_r je síla generovaná všemi rotory, T_i je tah i -tého rotoru - vztah (2.64) a $\vec{\chi}_i$ je vektor orientace i -tého rotoru vyjádřený v pohyblivé soustavě pevně spojené s tělesem.

Podobně je tomu i u celkového momentu generovaného rotory, tedy

$$\vec{\tau}_r = \sum_{i=1}^N \vec{P}_i \times \vec{\chi}_i T_i + \sum_{i=1}^N (-1)^{i+1} \vec{\chi}_i \tau_{m,i}, \quad (2.67)$$

kde τ_r je celkový moment rotorů, \vec{P}_i je vektor pozice rotoru vyjádřený v pohyblivé soustavě pevně spojené s tělesem, $\tau_{m,i}$ je moment i -tého rotoru - vztah (2.65) a člen $(-1)^{i+1}$ reprezentuje skutečnost, že všechny rotory se neotáčejí stejným směrem, ale vždy dvojice rotorů se otáčí proti sobě, viz obrázek 17.



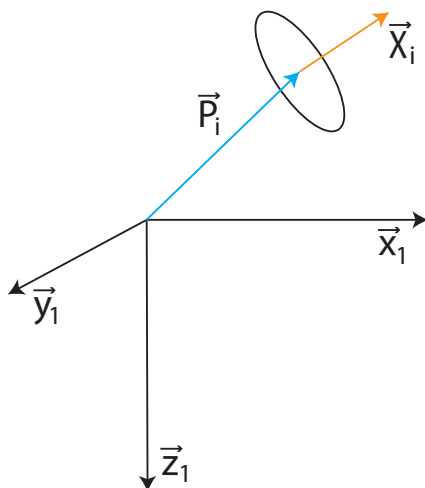
Obrázek 17: Otáčení rotorů - kvadroptéra

Obrázek (17) popisuje, jak je vždy nutné uspořádat jednotlivé rotory a zvolit směr jejich otáčení. Díky tomuto uspořádání se kompenzuje rotace okolo osy \vec{z} . Pokud by se tedy všechny rotory otáčely stejným směrem, tak by UAV neustále rotovalo, jelikož by výsledný moment otáčení směřoval neustále v opačném směru otáčení vrtulí.

Pro pozdější použití je definováno následující značení matic:

$$\chi = (\vec{\chi}_1, \vec{\chi}_2, \dots, \vec{\chi}_N), \quad P = (\vec{P}_1, \vec{P}_2, \dots, \vec{P}_N), \quad (2.68)$$

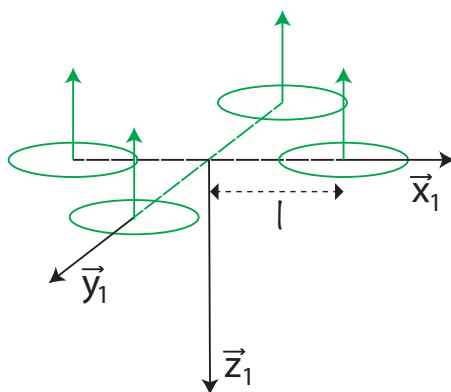
kde matice χ se skládá z vektorů orientace rotoru, matice P se skládá z vektorů pozice rotorů a N je celkový počet rotorů na UAV. Všechny tyto vektory jsou vyjádřeny v pohyblivé soustavě pevně spojené s tělesem.



Obrázek 18: Rotor v prostoru

2.6.1 Kvadroptéra

V této variantě je uvažováno, že rotory jsou pevně spojeny s konstrukcí UAV a tedy směřují jedním směrem. Jednotlivé rotory jsou dále umístěny symetricky na jedné ploše a jsou od středu ve vzdálenosti l .



Obrázek 19: Rozmístění rotorů

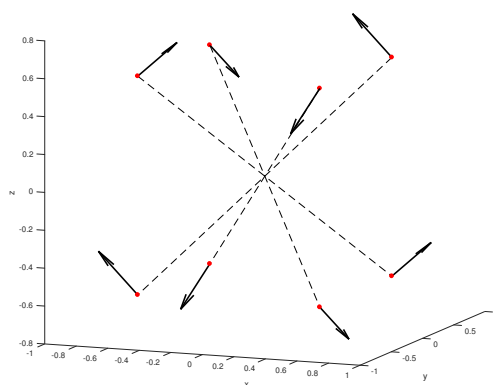
Jelikož všechny rotory míří stejným směrem (Obrázek 19), je nutné využít vztahu (2.67), kde se vždy 2 rotory budou otáčet opačným směrem, než zbylá dvojice rotorů. Jednotlivé matice

pro polohu (P) a pro orientaci (χ) jsou tedy

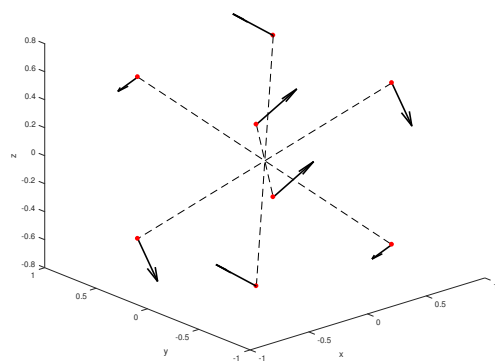
$$P = l \begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & -1 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad \chi = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & -1 & -1 & -1 \end{bmatrix}.$$

2.6.2 Omnikoptéra

Klasická kvadrokoptéra je omezena díky stejnému natočení rotorů, což znamená, že není schopna například přímočarého pohybu po ose \vec{x} bez natočení celého dronu. Následující konfigurace na rozdíl od kvadrokoptéry nabízí mnohem více možností pohybu.



Obrázek 20: Omnikoptéra - směry rotorů



Obrázek 21: Omnikoptéra - směry rotorů

Jednotlivé matice pro polohu (P) a pro orientaci (χ) v tomto případě jsou

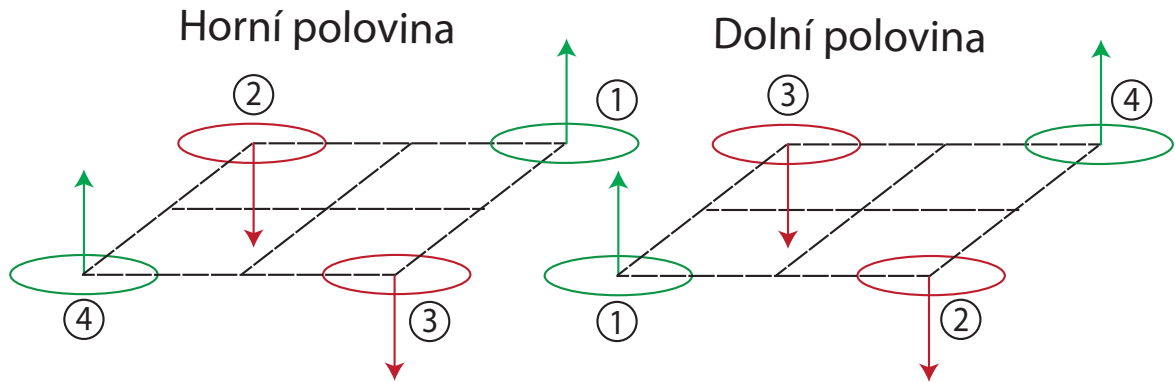
$$P = l \frac{1}{\sqrt{3}} \begin{bmatrix} 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ -1 & -1 & -1 & -1 & 1 & 1 & 1 & 1 \end{bmatrix}, \chi = \begin{bmatrix} -a & b & -b & a & a & -b & b & -a \\ b & a & -a & -b & -b & -a & a & b \\ -c & c & c & -c & -c & c & c & -c \end{bmatrix},$$

$$\text{kde } a = \frac{1}{2} + \frac{1}{\sqrt{12}}, \quad b = \frac{1}{2} - \frac{1}{\sqrt{12}}, \quad c = \frac{1}{\sqrt{3}}.$$

Nejdříve bude provedena jednoduchá analýza natočení rotorů. Rotory musí být vhodně umístěny, aby nedocházelo k žádné samovolné rotaci okolo žádné osy. Nyní bude pozornost věnována natočení rotorů χ , tedy

$$\chi = \begin{bmatrix} -a & b & -b & a & a & -b & b & -a \\ b & a & -a & -b & -b & -a & a & b \\ \underbrace{-c}_{\textcircled{1}} & \underbrace{c}_{\textcircled{2}} & \underbrace{c}_{\textcircled{3}} & \underbrace{-c}_{\textcircled{4}} & \underbrace{-c}_{\textcircled{4}} & \underbrace{c}_{\textcircled{3}} & \underbrace{c}_{\textcircled{2}} & \underbrace{-c}_{\textcircled{1}} \end{bmatrix}, \quad (2.69)$$

vždy dvojice rotorů je tedy natočena stejným směrem. Lze si i všimnout, že polovina rotorů je natočena do horní poloviny a druhá naopak otočena do dolní poloviny.



Obrázek 22: Omnikoptéra - směr rotorů v ose \vec{z}_1

Podobně je tomu tak i v jiných osách, z čehož všeho vyplývá, že takto zvolené umístění a naklonění rotorů vhodně eliminuje nežádoucí samovolné rotace ve všech osách. Vztah (2.67) lze tedy využít ve tvaru

$$\vec{\tau}_r = \sum_{i=1}^N \vec{P}_i \times \vec{\chi}_i T_i + \sum_{i=1}^N \vec{\chi}_i \tau_{m,i}, \quad (2.70)$$

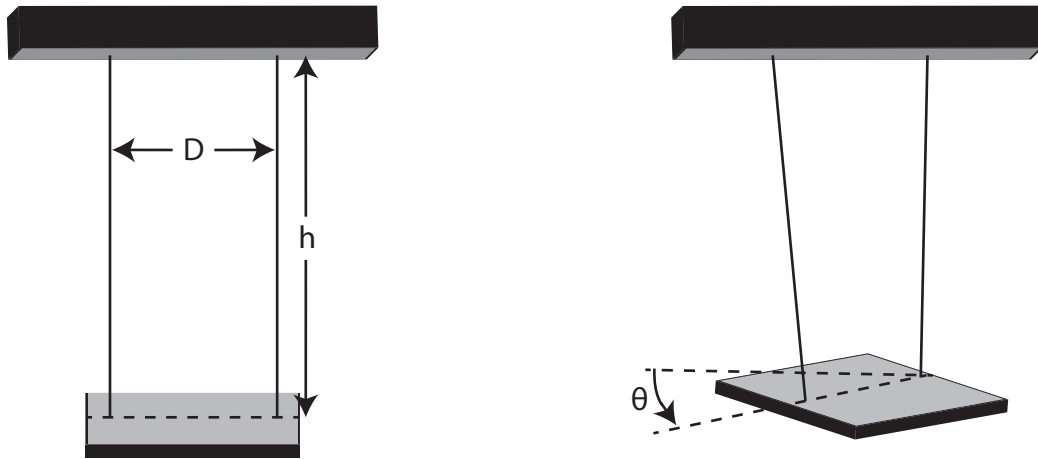
kde již není nutné využívat členu pro změnu směru rotace.

2.7 Určení matice setrvačnosti - J_1

Matice setrvačnosti je určena v pohyblivé soustavě souřadnic, jelikož je v této soustavě konstantní a není nutné brát v úvahu natočení UAV, což by bylo nutné v inerciální soustavě souřadnic. Protože jsou oba dva zkoumané druhy UAV symetrické, lze matici setrvačnosti J_1 zjednodušit do tvaru

$$J_1 = \begin{bmatrix} J_{1\ xx} & 0 & 0 \\ 0 & J_{1\ yy} & 0 \\ 0 & 0 & J_{1\ zz} \end{bmatrix}. \quad (2.71)$$

Výpočet matice setrvačnosti bude proveden dle článků [18] a [20]. Zde je výpočet zaměřen na experiment, kdy se zkoumané těleso zavěsí. Je nutné, aby byl závěs proveden symetricky vzhledem k ose, ve které je moment setrvačnosti zkoumán. Tento závěs je znázorněn na následujícím obrázku - 23.



Obrázek 23: Experiment

Těleso je tedy zavěšeno na dvou tenkých lankách délky h , které jsou od sebe ve vzdálenosti D , následně je těleso otočeno o úhel θ . Následující postup je ilustrován pro osu rotace \vec{z}_1 .

V tomto případě je pro výpočet využita Euler-Lagrangeova metoda. Lagrangián je definován jako

$$L = T_L - V_L, \quad (2.72)$$

kde T_L je kinetická energie tělesa a V_L je potenciální energie tělesa. Lagrangeova rovnice je pak ve tvaru

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\theta}} \right) - \frac{\partial L}{\partial \theta} = Q_L, \quad (2.73)$$

kde Q_L představuje zobecněnou sílu. Nyní bude definována kinetická a potenciální energie,

$$T_L = \frac{1}{2} J_{zz} \dot{\theta}^2 + \frac{1}{2} m \dot{z}^2, \quad (2.74)$$

$$V = mgz, \quad (2.75)$$

kde J_{zz} je hledaný moment setrvačnosti, m je hmotnost tělesa, g je gravitační konstanta a z je definováno dle [20] jako

$$z = h \left(1 - \sqrt{1 - \frac{1}{2} \left(\frac{D}{h} \right)^2 (1 - \cos(\theta))} \right). \quad (2.76)$$

Pokud je vztah (2.76) zderivován, je získán poslední potřebný člen, tedy

$$\dot{z} = \frac{dz}{dt} = \frac{\frac{1}{L} \left(\frac{D}{2} \right)^2 \sin(\theta)}{\sqrt{1 - \frac{1}{2} \left(\frac{D}{h} \right)^2 (1 - \cos(\theta))}} \dot{\theta}. \quad (2.77)$$

Pro zjednodušení je zvoleno $h \approx 2D$ a je předpokládáno, že výchylky $\theta \leq \frac{\pi}{8}$. Z čehož vyplývá, že $\dot{z}^2 \approx \dot{\theta}^2 \cdot 10^{-3}$. Při využití všech předpokladů a vztahů (2.72) - (2.77) lze získat vztah

$$J_{zz}\ddot{\theta} + \frac{mg \left(\frac{D}{2\sqrt{h}} \right)^2 \sin(\theta)}{\sqrt{1 - \frac{1}{2} \left(\frac{D}{h} \right)^2 (1 - \cos(\theta))}} = 0. \quad (2.78)$$

Za předpokladu malé výchylky úhlu θ lze $\cos(\theta) \approx 1$ a $\sin(\theta) \approx \theta$, vztah (2.78) je zjednodušen, tedy

$$\ddot{\theta} + \frac{mgD^2}{4hJ_{zz}}\theta = 0, \quad (2.79)$$

který má řešení ve tvaru

$$\theta = c_1 \cos \left(\sqrt{\frac{mgD^2}{4hJ_{zz}}} t \right) + c_2 \sin \left(\sqrt{\frac{mgD^2}{4hJ_{zz}}} t \right), \quad (2.80)$$

kde c_1 a c_2 jsou libovolné konstanty. Pokud $\dot{\theta}(0) = 0$, je konstanta $c_2 = 0$. Za využití tohoto předpokladu a periody T lze vyjádřit setrvačnosti jako

$$J_{zz} = \frac{mg(hT)^2}{16\pi^2 h}. \quad (2.81)$$

Stejný experiment lze provést i v osách \vec{x}_1 a \vec{y}_1 a tím získat celkovou matici setrvačnosti zkoumaného UAV.

2.8 Nelineární model

Nelineární model je sestaven pomocí předem odvozených pohybových rovnic. Konkrétně to jsou rovnice (2.45), (2.51), (2.66) a (2.67). Pro zjednodušení výpočtů je model uvažován v pohyblivé soustavě souřadnic pevně spojené s tělesem. Finální vztahy jsou pak jen pomocí matice rotace \mathbf{R} (2.16) a pomocí matice \mathbf{W} (2.41) převedeny do inerciální soustavy. Tedy rovnice, která popisuje působení síly na UAV, je

$$\vec{F}_1 = \vec{F}_r + \underbrace{mg\mathbf{R}^T}_{\vec{F}_g} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \quad (2.82)$$

kde nově přidaný člen \vec{F}_g reprezentuje gravitační sílu, která je vyjádřena v pohyblivé soustavě souřadnic. Síla \vec{F}_r je generována rotory dle vztahu (2.66). Rovnice popisující moment, který působí na pohyblivé těleso, je ve tvaru

$$\vec{\tau}_1 = \vec{\tau}_r. \quad (2.83)$$

Moment $\vec{\tau}_r$ je generovaný rotory dronu ze vztahu (2.67). Tyto rovnice je nutné vyjádřit ve tvaru, který popisuje zrychlení translačního a rotačního pohybu, tedy

$$\dot{\vec{v}}_1 = \vec{\omega}_1 \times \vec{v}_1 + \frac{1}{m} \vec{F}_r + g\mathbf{R}^T \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \quad (2.84)$$

$$\dot{\vec{\omega}}_1 = \frac{1}{J_1} (\vec{\tau}_r - (\vec{\omega}_1 \times J_1 \vec{\omega}_1)). \quad (2.85)$$

Jako vstupní signál je určena úhlová rychlost jednotlivých rotorů

$$u_i = \omega_{mi}^2, \quad (2.86)$$

kde index i jednoznačně reprezentuje daný rotor z celkového počtu N . Jako poslední krok je převedení rovnic (2.82) a (2.83) do inerciální soustavy souřadnic, kde

$$\dot{\vec{v}}_0 = \mathbf{R} \dot{\vec{v}}_1. \quad (2.87)$$

Pro převod do úhlové rychlosti, která je vyjádřena v inerciální soustavě souřadnic je využit vztah (2.41), tedy

$$\dot{\vec{\eta}}_0 = \mathbf{W} \dot{\vec{\omega}}_1,$$

který je ještě nutné zderivovat, tím je získán finální vztah

$$\ddot{\vec{\eta}}_0 = \dot{\mathbf{W}} \dot{\vec{\omega}}_1 + \mathbf{W} \ddot{\vec{\omega}}_1. \quad (2.88)$$

V reálném prostředí ovšem působí v rovnicích (2.82) a (2.83) více dalších rušivých sil a momentů, čímž lze dosáhnout přesnější podobnosti s realitou. Jedná se například o rychlost větru, odpor vzduchu, vibrace na vrtulích, tření v rotorech a jiné. V první řadě je však důležité vytvořit co možná nejjednodušší model, na kterém lze získat jisté zkušenosti v oblasti chování UAVs.

2.8.1 Rovnovážný stav

Výpočet rovnovážného stavu je důležitý zejména při linearizaci systému, které bude věnována následující kapitola. Pro jednoduchost je rovnovážný stav zvolen jako

$$\vec{\xi}_{eq} = \begin{bmatrix} x_{eq} \\ y_{eq} \\ z_{eq} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -1 \end{bmatrix}, \quad \vec{v}_{eq} = \begin{bmatrix} u_{eq} \\ v_{eq} \\ w_{eq} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \quad \vec{\eta}_{eq} = \begin{bmatrix} \alpha_{eq} \\ \beta_{eq} \\ \gamma_{eq} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \quad \dot{\vec{\eta}}_{eq} = \begin{bmatrix} \dot{\alpha}_{eq} \\ \dot{\beta}_{eq} \\ \dot{\gamma}_{eq} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \quad (2.89)$$

tedy všechny veličiny jsou nulové kromě polohy v ose \vec{z}_0 , kde je UAV umístěno 1m nad zemí. Všechny tyto veličiny jsou vztaženy k inerciální soustavě souřadnic.

Kvadroptéra

V tomto případě se výpočet velice zjednoduší. Jelikož se musí každá z dvojic rotorů otáčet jiným směrem, aby nedocházelo k otáčení celé kvadroptéry podle ose \vec{z}_1 , je řízení všech vrtulí zvoleno jednotně, tedy

$$u_1 = u_{eq}, \quad u_2 = u_{eq}, \quad u_3 = u_{eq}, \quad u_4 = u_{eq}. \quad (2.90)$$

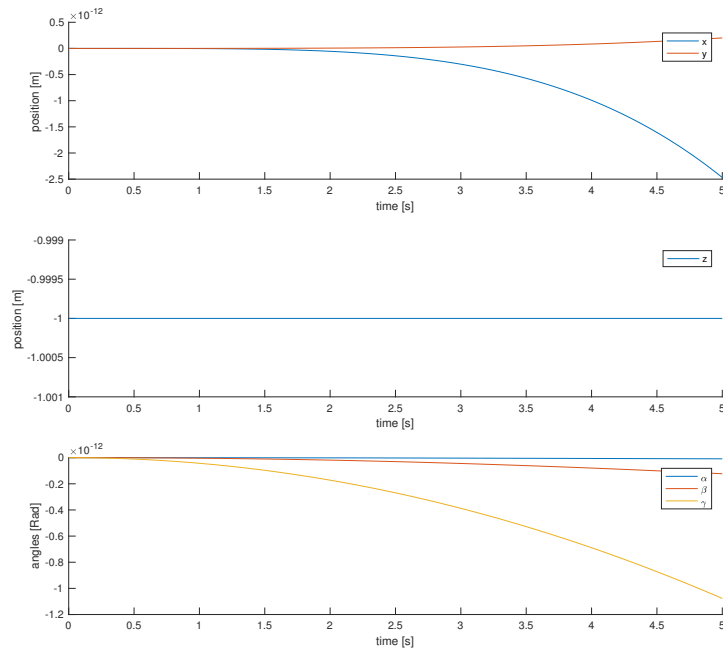
Směr otáčení jednotlivých rotorů není dán řízením, ale je zakomponován již při návrhu konstrukce kvadroptéry. Vzhledem k tomu, že takovéto nastavení zabrání v rovnovážném stavu působení točivého momentu podél osy \vec{z}_1 , bude dále využit pouze vztah na výpočet síly generované rotory. S využitím vztahu (2.64), (2.66) a (2.82) lze celkovou sílu generovanou v rovnovážném stavu psát jako

$$\vec{F}_{eq} = \begin{bmatrix} 0 \\ 0 \\ -4K_T u_{eq} + mg \end{bmatrix}, \quad (2.91)$$

řízení, které udrží UAV ve zvoleném rovnovážném stavu (2.89) je pak ve tvaru

$$u_{eq} = \frac{1}{4} \frac{mg}{K_T}. \quad (2.92)$$

Vhodné je tento výsledek ověřit i simulačně, to je provedeno v programovém prostředí *Matlab/Simulink*. Do nelineárních rovnic je jako vstup využit právě vypočtený vztah (2.92).

Obrázek 24: Simulační ověření rovnovážného stavu - *Matlab/Simulink*

Ze simulace - obrázek 24 lze vidět, že UAV setrvává v rovnovážné poloze. Mírné výchyly jsou dány numerickými chybami, které vznikají při výpočtech v programu *Matlab*. Jedná se zejména o numerické vyjádření řízení u_{eq} .

Omnikoptéra

I v tomto případě jsou rotory rozmístěny a hlavně natočeny symetricky - viz (2.69). Zde se přímo nabízí varianta, že by čtveřice rotorů měla řízení u_{eq} a zbylá čtveřice by naopak měla řízení $-u_{eq}$. Bohužel tato úvaha by velice dobře fungovala pouze bez přítomné gravitace. Tato možnost sice eliminuje všechny momenty i působení síly v osách \vec{x}_1 a \vec{y}_1 , ale působící gravitační síla stáhne UAV směrem k zemi v ose \vec{z}_0 . Je tedy nutné řešit soustavu rovnic

$$\begin{bmatrix} \vec{F}_{eqX} \\ \vec{F}_{eqY} \\ \vec{F}_{eqZ} \\ \vec{\tau}_{eqX} \\ \vec{\tau}_{eqY} \\ \vec{\tau}_{eqZ} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad (2.93)$$

kde vztahy pro sílu a moment, které generují rotory, jsou získány ze vztahů (2.64), (2.65), (2.66)

a (2.82). Ze vztahu (2.93) je pouze nutné vyjádřit řízení u_i pro jednotlivé rotory:

$$u_{1eq} = u_{4eq}, \quad (2.94)$$

$$u_{2eq} = \frac{1}{8} \frac{-gmK_{\tau\tau} \sqrt{3} - 2gmlK_T \sqrt{3} + 8K_T^2 l u_{4eq}}{K_T^2 l}, \quad (2.95)$$

$$u_{3eq} = \frac{1}{8} \frac{-gmK_{\tau\tau} \sqrt{3} - 2gmlK_T \sqrt{3} + 8K_T^2 l u_{4eq}}{K_T^2 l}, \quad (2.96)$$

$$u_{4eq} = u_{4eq}, \quad (2.97)$$

$$u_{5eq} = -\frac{1}{24} \frac{\sqrt{3} (3gmK_{\tau\tau} - 6gmlK_T - 8K_T^2 \sqrt{3} l u_{7eq})}{K_T^2 l} \quad (2.98)$$

$$u_{6eq} = u_{7eq}, \quad (2.99)$$

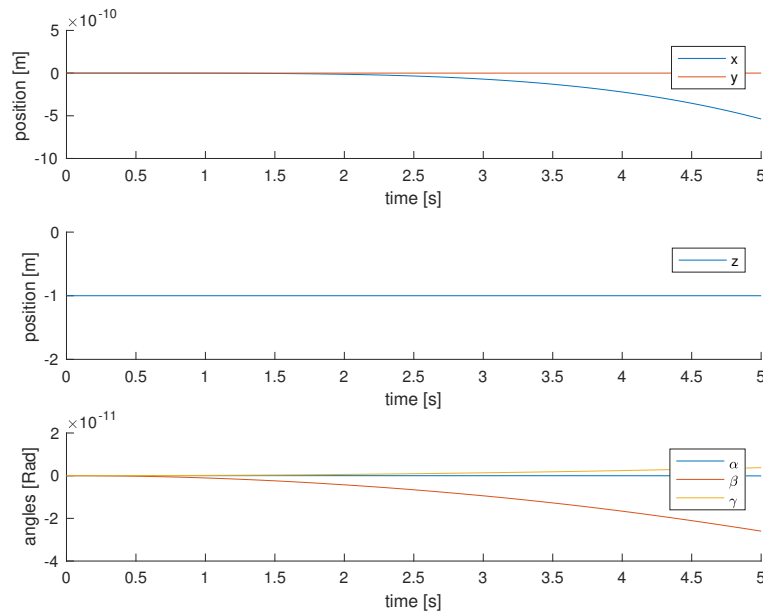
$$u_{7eq} = u_{7eq}, \quad (2.100)$$

$$u_{8eq} = -\frac{1}{24} \frac{\sqrt{3} (3gmK_{\tau\tau} - 6gmlK_T - 8K_T^2 \sqrt{3} l u_{7eq})}{K_T^2 l}. \quad (2.101)$$

U vztahů (2.94) a (2.101) si lze všimnout, že 2 řízení - u_{4eq} a u_{7eq} jsou volitelná, u_{2eq} a u_{3eq} jsou shodné stejně jako dvojice u_{5eq} a u_{8eq} . Pro zachování symetrie je vhodné zvolit řízení u_{4eq} a u_{7eq} stejné, tedy je zvoleno jako

$$u_{4eq} = u_{7eq} = 10. \quad (2.102)$$

I v tomto případě budou výsledky ověřeny i simulačně. Nelineární rovnice byly opět nasimulovány v programovém prostředí *Matlab/Simulink*.

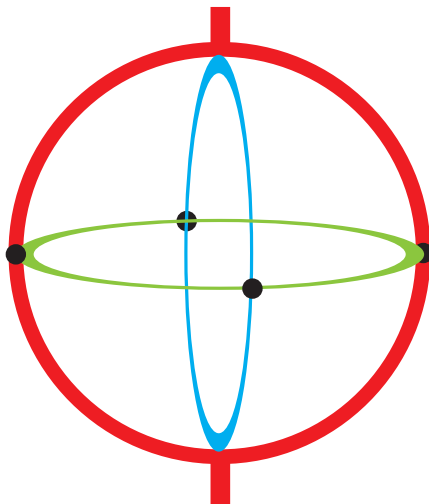


Obrázek 25: Simulační ověření rovnovážného stavu - *Matlab/Simulink*

I v tomto případě jsou výsledky velice uspokojivé. UAV zůstává v rovnovážném stavu. Tento rovnovážný stav bude následně využit při linearizaci.

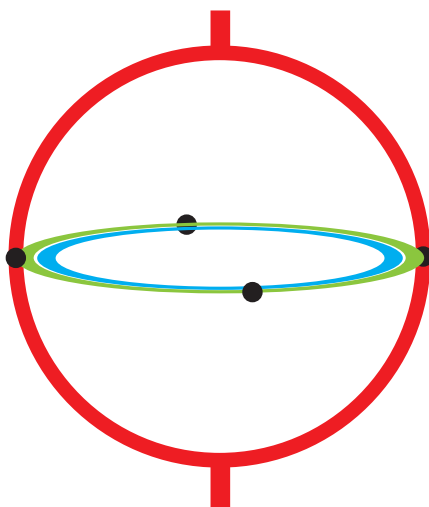
2.9 Kvaterniony

Pokud bude pro simulaci využívána matice rotace \mathbf{R} (2.16) a matice \mathbf{W} (2.41), tedy reprezentace v Eulerovo úhlech, je zde nebezpečí, že nastane jev zvaný "gimbal lock". Tento jev se projevuje ztrátou jednoho stupně volnosti. Ilustrace bude provedena na tzv. Kardanovo závěsu.



Obrázek 26: Kardanův závěs

Pokud bude modrý kruh pootočen o $\pi/2$ rad (rotace dle osy \vec{y}) a červený kruh bude pokračovat v rotaci, změní se i rotace modrého kruhu. Tedy jeden stupeň volnosti bude ztracen.



Obrázek 27: Kardanův závěs - ztráta stupně volnosti

Problémem tohoto chování jsou singulární polohy, které nastávají při reprezentaci s Eulerovými úhly. Tento problém nastává, pokud se $\beta = \pm \frac{\pi}{2}$. Pro tento případ matice rotace (2.15) degeneruje na následující tvar

$$\mathbf{R}_{1 \rightarrow 0} \left(\beta = \frac{\pi}{2} \right) = \begin{bmatrix} 0 & 0 & -1 \\ -c_\alpha s_\gamma + s_\alpha c_\gamma & c_\alpha c_\gamma + s_\alpha s_\gamma & 0 \\ s_\alpha s_\gamma + c_\alpha c_\gamma & -s_\alpha c_\gamma + c_\alpha s_\gamma & 0 \end{bmatrix}, \quad (2.103)$$

$$\mathbf{R}_{1 \rightarrow 0} \left(\beta = -\frac{\pi}{2} \right) = \begin{bmatrix} 0 & 0 & 1 \\ -c_\alpha s_\gamma - s_\alpha c_\gamma & c_\alpha c_\gamma - s_\alpha s_\gamma & 0 \\ s_\alpha s_\gamma - c_\alpha c_\gamma & -s_\alpha c_\gamma - c_\alpha s_\gamma & 0 \end{bmatrix}, \quad (2.104)$$

yní lze určit pouze rozdíl, respektive součet úhlů α a γ , tedy nastává zde nejednoznačnost při řešení zpětné transformace. Aby nedocházelo k těmto jevům, byly zavedeny tzv. kvaterniony.

2.9.1 Základní vztahy

Kvaterniony jsou čtyřrozměrné vektory, které obsahují jednu reálnou a tři imaginární složky. Následující vztahy a odvození jsou získány za pomoci článků [18], [25], [26], [27], [28] a [29]. Imaginární části jsou označeny jako i , j a k a platí pro ně

$$i^2 = j^2 = k^2 = ijk = -1. \quad (2.105)$$

Kvaternion \mathbf{q} je definovaný jako

$$\mathbf{q} = q_0 + q_1 i + q_2 j + q_3 k, \quad (2.106)$$

kde q_0 , q_1 , q_2 a q_3 jsou reálná čísla. Ze vztahu (2.106) lze získat vektorovou reprezentaci kvaternionu:

$$\vec{\mathbf{q}} \equiv [q_0 \ q_1 \ q_2 \ q_3]^T \quad (2.107)$$

Norma kvaternionu je dána následujícím vztahem

$$\|\vec{\mathbf{q}}\| = \sqrt{q_0^2 + q_1^2 + q_2^2 + q_3^2} = 1, \quad (2.108)$$

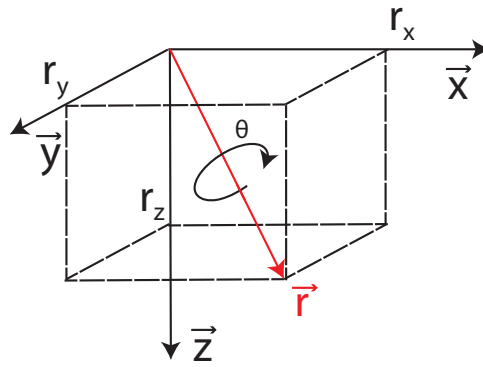
jelikož je norma jednotková, je tento kvaternion nazýván jednotkovým kvaternionem. Tento kvaternion je definován jako

$$\vec{\mathbf{q}} \equiv [\eta \ \vec{\epsilon}]^T, \quad (2.109)$$

kde jednotlivé složky jsou

$$\eta = \cos \frac{\vartheta}{2}, \quad \vec{\epsilon} = \sin \frac{\vartheta}{2} \vec{r} = [\epsilon_x \ \epsilon_y \ \epsilon_z], \quad (2.110)$$

kde ϑ je úhel o který je provedena rotace a \vec{r} je vektor okolo kterého je rotace provedena.



Obrázek 28: Kvaternion - rotace

Pro kvaterniony platí také speciální pravidla pro operace. Inverze kvaternionu je definovaná

$$\vec{q}^{-1} = \frac{\bar{\vec{q}}}{\|\vec{q}\|}, \quad (2.111)$$

kde $\bar{\vec{q}}$ je konjugovaný kvaternion a ten je ve tvaru

$$\bar{\vec{q}} = [q_0 \quad -q_1 \quad -q_2 \quad -q_3]^T. \quad (2.112)$$

2.9.2 Součin dvou kvaternionů

Další potřebnou operací je součin dvou kvaternionů, ten je definován vztahem

$$\vec{q}_1 \otimes \vec{q}_2 = [\eta_1 \quad \vec{\epsilon}_1]^T \otimes [\eta_2 \quad \vec{\epsilon}_2]^T \quad (2.113)$$

$$= [\eta_1 \eta_2 - \vec{\epsilon}_1 \cdot \vec{\epsilon}_2 \quad \eta_1 \vec{\epsilon}_2 + \eta_2 \vec{\epsilon}_1 + \vec{\epsilon}_1 \times \vec{\epsilon}_2]^T, \quad (2.114)$$

kde \cdot je skalární součin a \times je vektorový součin. Další možností je vyjádření součinu pomocí matic, tedy

$$\vec{q}_1 \otimes \vec{q}_2 = Q(\vec{q}_1) \vec{q}_2, \quad (2.115)$$

kde $Q(\vec{q}_1)$ je dáno vztahem

$$Q(\vec{q}_1) = \begin{bmatrix} \eta_1 & -\vec{\epsilon}_1^T \\ \vec{\epsilon}_1 & \eta_1 I_{3 \times 3} + C(\vec{\epsilon}_1) \end{bmatrix} \quad (2.116)$$

kde

$$C(\vec{\epsilon}_1) = \begin{bmatrix} 0 & -\epsilon_{1,3} & \epsilon_{1,2} \\ \epsilon_{1,3} & 0 & -\epsilon_{1,1} \\ -\epsilon_{1,2} & \epsilon_{1,1} & 0 \end{bmatrix}. \quad (2.117)$$

Vztah (2.117) vychází ze vztahu (2.30), tedy je to část z vektorového součinu $\vec{\epsilon}_1 \times \vec{\epsilon}_2$. Vztah (2.115) je tedy ve tvaru

$$\vec{\mathbf{q}}_1 \otimes \vec{\mathbf{q}}_2 = \begin{bmatrix} \eta_1 & -\vec{\epsilon}_1^T \\ \vec{\epsilon}_1 & \eta_1 I_{3 \times 3} + C(\vec{\epsilon}_1) \end{bmatrix} \begin{bmatrix} \eta_2 \\ \vec{\epsilon}_2 \end{bmatrix}, \quad (2.118)$$

$$= \underbrace{\begin{bmatrix} q_{1,0} & -q_{1,1} & -q_{1,2} & -q_{1,3} \\ q_{1,1} & q_{1,0} & -q_{1,3} & q_{1,2} \\ q_{1,2} & q_{1,3} & q_{1,0} & -q_{1,1} \\ q_{1,3} & -q_{1,2} & q_{1,1} & q_{1,0} \end{bmatrix}}_{Q(\vec{\mathbf{q}}_1)} \begin{bmatrix} \eta_2 \\ \epsilon_{2,1} \\ \epsilon_{2,2} \\ \epsilon_{2,3} \end{bmatrix}. \quad (2.119)$$

Násobení kvaternionů ale není komutativní, pro tento případ je vhodné využít konjugovanou matici $\overline{Q}(\vec{\mathbf{q}}_1)$. V součinu kvaternionů platí, že

$$\vec{\mathbf{q}}_1 \otimes \vec{\mathbf{q}}_2 = Q(\vec{\mathbf{q}}_1) \vec{\mathbf{q}}_2, \quad (2.120)$$

pokud by bylo pořadí obráceno, byl by vztah ve tvaru

$$\vec{\mathbf{q}}_2 \otimes \vec{\mathbf{q}}_1 = Q(\vec{\mathbf{q}}_2) \vec{\mathbf{q}}_1, \quad (2.121)$$

což je rovno vztahu

$$\vec{\mathbf{q}}_2 \otimes \vec{\mathbf{q}}_1 = \overline{Q}(\vec{\mathbf{q}}_1) \vec{\mathbf{q}}_2, \quad (2.122)$$

kde konjugovaná matice $\overline{Q}(\vec{\mathbf{q}}_1)$ je dána

$$\overline{Q}(\vec{\mathbf{q}}_1) = \begin{bmatrix} \eta_1 & -\vec{\epsilon}_1^T \\ \vec{\epsilon}_1 & \eta_1 I_{3 \times 3} - C(\vec{\epsilon}_1) \end{bmatrix} = \begin{bmatrix} q_{1,0} & -q_{1,1} & -q_{1,2} & -q_{1,3} \\ q_{1,1} & q_{1,0} & q_{1,3} & -q_{1,2} \\ q_{1,2} & -q_{1,3} & q_{1,0} & q_{1,1} \\ q_{1,3} & q_{1,2} & -q_{1,1} & q_{1,0} \end{bmatrix}. \quad (2.123)$$

Dále platí následující vztahy

$$Q(\overline{\vec{\mathbf{q}}}) = Q(\vec{\mathbf{q}})^T, \quad (2.124)$$

$$\overline{Q}(\overline{\vec{\mathbf{q}}}) = \overline{Q}(\vec{\mathbf{q}})^T. \quad (2.125)$$

2.9.3 Rotace vektoru pomocí kvaternionu

Jednotkový kvaternion je výhodné použít pro převod vektorů mezi soustavami. Vztah pro převod mezi soustavami je definován

$$\begin{bmatrix} 0 \\ \vec{\xi}_0 \end{bmatrix} = \vec{\mathbf{q}} \otimes \begin{bmatrix} 0 \\ \vec{\xi}_1 \end{bmatrix} \otimes \vec{\mathbf{q}}^{-1} \quad (2.126)$$

$$= \vec{\mathbf{q}} \otimes \begin{bmatrix} 0 \\ \vec{\xi}_1 \end{bmatrix} \otimes \bar{\vec{\mathbf{q}}} \quad (2.127)$$

$$= Q(\vec{\mathbf{q}}) \begin{bmatrix} 0 \\ \vec{\xi}_1 \end{bmatrix} \otimes \bar{\vec{\mathbf{q}}} \quad (2.128)$$

$$= Q(\vec{\mathbf{q}}) \bar{Q}(\bar{\vec{\mathbf{q}}}) \begin{bmatrix} 0 \\ \vec{\xi}_1 \end{bmatrix} \quad (2.129)$$

$$= Q(\vec{\mathbf{q}}) \bar{Q}^T(\bar{\vec{\mathbf{q}}}) \begin{bmatrix} 0 \\ \vec{\xi}_1 \end{bmatrix}, \quad (2.130)$$

je vhodné poznamenat, že

$$Q(\vec{\mathbf{q}}) \bar{Q}^T(\bar{\vec{\mathbf{q}}}) = \bar{Q}^T(\bar{\vec{\mathbf{q}}}) Q(\vec{\mathbf{q}}). \quad (2.131)$$

Vztah $Q(\vec{\mathbf{q}}) \bar{Q}^T(\bar{\vec{\mathbf{q}}})$ lze dále ještě upravit, čímž je získán maticový tvar

$$\begin{bmatrix} 0 \\ \vec{\xi}_0 \end{bmatrix} = \begin{bmatrix} 1 & 0^T \\ 0 & \mathbf{R}_{q \ 1 \rightarrow 0} \end{bmatrix} \begin{bmatrix} 0 \\ \vec{\xi}_1 \end{bmatrix}, \quad (2.132)$$

kde $\mathbf{R}_{q \ 1 \rightarrow 0}$ je dáno vztahem

$$\mathbf{R}_{q \ 1 \rightarrow 0} = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1q_2 - q_0q_3) & 2(q_0q_2 + q_1q_3) \\ 2(q_0q_3 + q_1q_2) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2q_3 - q_0q_1) \\ 2(q_1q_3 - q_0q_2) & 2(q_0q_1 + q_2q_3) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix}, \quad (2.133)$$

tedy lze psát, že

$$\vec{\xi}_0 = \mathbf{R}_{q \ 1 \rightarrow 0} \vec{\xi}_1, \quad (2.134)$$

podobně lze udělat i zpětnou transformaci, kdy

$$\vec{\xi}_1 = \mathbf{R}_{q \ 1 \rightarrow 0}^T \vec{\xi}_0. \quad (2.135)$$

Platí zde podobný vztah jako u Eulerových úhlů (2.6), tedy

$$\mathbf{R}_{q \ 1 \rightarrow 0}^T = \mathbf{R}_{q \ 0 \rightarrow 1}, \quad (2.136)$$

bude tedy využito stejné značení, kdy

$$\begin{aligned} \mathbf{R}_q &= \mathbf{R}_{q \ 1 \rightarrow 0}, \\ \mathbf{R}_q^T &= \mathbf{R}_{q \ 0 \rightarrow 1}. \end{aligned}$$

2.9.4 Derivace kvaternionu

Derivace kvaternionu je definována vztahem

$$\dot{\mathbf{q}} = \frac{1}{2} \mathbf{q} \otimes \begin{bmatrix} 0 \\ \vec{\omega}_1 \end{bmatrix} \quad (2.137)$$

$$= \frac{1}{2} Q(\mathbf{q}) \begin{bmatrix} 0 \\ \vec{\omega}_1 \end{bmatrix} \quad (2.138)$$

$$= \frac{1}{2} \begin{bmatrix} q_0 & -q_1 & -q_2 & -q_3 \\ q_1 & q_0 & -q_3 & q_2 \\ q_2 & q_3 & q_0 & -q_1 \\ q_3 & -q_2 & q_1 & q_0 \end{bmatrix} \begin{bmatrix} 0 \\ \vec{\omega}_1 \end{bmatrix}, \quad (2.139)$$

tento vztah lze zjednodušit na tvar, kdy je z matice $Q(\mathbf{q})$ odstraněn sloupec, který by byl nulován díky vektoru $[0 \ \vec{\omega}_1]^T$, tedy

$$\dot{\mathbf{q}} = \frac{1}{2} \begin{bmatrix} \cancel{q_0} & -q_1 & -q_2 & -q_3 \\ \cancel{q_1} & q_0 & -q_3 & q_2 \\ \cancel{q_2} & q_3 & q_0 & -q_1 \\ \cancel{q_3} & -q_2 & q_1 & q_0 \end{bmatrix} \begin{bmatrix} \emptyset \\ \vec{\omega}_1 \end{bmatrix}, \quad (2.140)$$

$$\dot{\mathbf{q}} = \frac{1}{2} W(\mathbf{q}) \vec{\omega}, \quad (2.141)$$

kde $\vec{\omega}_1$ je úhlová rychlost vyjádřená v pohyblivé soustavě a matice $W(\mathbf{q})$ je dána vztahem

$$W(\mathbf{q}) = \begin{bmatrix} -q_1 & -q_2 & -q_3 \\ q_0 & -q_3 & q_2 \\ q_3 & q_0 & -q_1 \\ -q_2 & q_1 & q_0 \end{bmatrix}. \quad (2.142)$$

Kvaternion \mathbf{q} je zde uvažován v inerciální soustavě souřadnic.

2.9.5 Úhlová rychlost

Stejným způsobem jako v předešlé kapitole lze postupovat pro vyjádření úhlové rychlosti. Úhlová rychlost je tedy dána jako

$$\begin{bmatrix} 0 \\ \vec{\omega}_1 \end{bmatrix} = 2 \bar{\mathbf{q}} \otimes \dot{\mathbf{q}} \quad (2.143)$$

$$= 2Q^T(\mathbf{q}) \dot{\mathbf{q}}. \quad (2.144)$$

Zde je opět provedena redukce matice $Q^T(\mathbf{q})$, tedy

$$\begin{bmatrix} \emptyset \\ \vec{\omega}_1 \end{bmatrix} = 2 \begin{bmatrix} \cancel{q_0} & \cancel{q_1} & \cancel{q_2} & \cancel{q_3} \\ -q_1 & q_0 & q_3 & -q_2 \\ -q_2 & -q_3 & q_0 & q_1 \\ -q_3 & q_2 & -q_1 & q_0 \end{bmatrix} \dot{\mathbf{q}} \quad (2.145)$$

a je získán tvar

$$\vec{\omega}_1 = 2W^T(\vec{\mathbf{q}})\dot{\vec{\mathbf{q}}}, \quad (2.146)$$

kde matice $W^T(\vec{\mathbf{q}})$ je dána

$$W^T(\vec{\mathbf{q}}) = \begin{bmatrix} -q_1 & q_0 & q_3 & -q_2 \\ -q_2 & -q_3 & q_0 & q_1 \\ -q_3 & q_2 & -q_1 & q_0 \end{bmatrix}. \quad (2.147)$$

2.9.6 Druhá derivace kvaternionu

Další nutný vztah je druhá derivace kvaternionu $\ddot{\vec{\mathbf{q}}}$, která bude využita pro tvorbu simulačního modelu. Pro odvození bude využit vztah (2.141), tedy

$$\dot{\vec{\mathbf{q}}} = \frac{1}{2}W(\vec{\mathbf{q}})\vec{\omega}_1,$$

kde bude aplikována derivace dle času. Druhou derivaci kvaternionu lze získat, když

$$\ddot{\vec{\mathbf{q}}} = \frac{d\dot{\vec{\mathbf{q}}}}{dt} = \frac{d}{dt} \left(\frac{1}{2}W(\vec{\mathbf{q}})\vec{\omega}_1 \right) \quad (2.148)$$

$$= \frac{1}{2}\dot{W}(\vec{\mathbf{q}})\vec{\omega}_1 + \frac{1}{2}W(\vec{\mathbf{q}})\dot{\vec{\omega}}_1. \quad (2.149)$$

Nyní jsou již skoro všechny potřebné vztahy definovány. Jediné, co zbývá, je definovat převod mezi kvaterniony a Eulerovými úhly.

2.9.7 Vztah mezi Eulerovými úhly a kvaterniony

Jelikož bude celý model uvažován v kvaternionech, bude vhodné pro vizualizaci převádět kvaterniony opět do Eulerových úhlů. Eulerovy úhly jsou pro představu mnohem jednodušší a názornější. Vztah pro převod z Eulerových úhlů do kvaternionů je dán

$$\vec{\mathbf{q}} = \begin{bmatrix} \cos(\gamma/2) \\ 0 \\ 0 \\ \sin(\gamma/2) \end{bmatrix} \otimes \begin{bmatrix} \cos(\beta/2) \\ 0 \\ \sin(\beta/2) \\ 0 \end{bmatrix} \otimes \begin{bmatrix} \cos(\alpha/2) \\ \sin(\alpha/2) \\ 0 \\ 0 \end{bmatrix}, \quad (2.150)$$

$$\vec{\mathbf{q}} = \begin{bmatrix} \cos(\alpha/2) \cos(\beta/2) \cos(\gamma/2) + \sin(\alpha/2) \sin(\beta/2) \sin(\gamma/2) \\ \sin(\alpha/2) \cos(\beta/2) \cos(\gamma/2) - \cos(\alpha/2) \sin(\beta/2) \sin(\gamma/2) \\ \cos(\alpha/2) \sin(\beta/2) \cos(\gamma/2) + \sin(\alpha/2) \cos(\beta/2) \sin(\gamma/2) \\ \cos(\alpha/2) \cos(\beta/2) \sin(\gamma/2) - \sin(\alpha/2) \sin(\beta/2) \cos(\gamma/2) \end{bmatrix}. \quad (2.151)$$

Pro opačný převod lze využít matice rotace $\mathbf{R}_{1 \rightarrow 0}$ (2.15) a $\mathbf{R}_{q \ 1 \rightarrow 0}$ (2.133). Pokud budou porovnány jednotlivé složky matic, tedy

$$\begin{aligned} & \begin{bmatrix} c_\beta c_\gamma & c_\beta s_\gamma & -s_\beta \\ -c_\alpha s_\gamma + s_\alpha s_\beta c_\gamma & c_\alpha c_\gamma + s_\alpha s_\beta s_\gamma & s_\alpha c_\beta \\ s_\alpha s_\gamma + c_\alpha s_\beta c_\gamma & -s_\alpha c_\gamma + c_\alpha s_\beta s_\gamma & c_\alpha c_\beta \end{bmatrix} \\ & = \\ & \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1 q_2 - q_0 q_3) & 2(q_0 q_2 + q_1 q_3) \\ 2(q_0 q_3 + q_1 q_2) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2 q_3 - q_0 q_1) \\ 2(q_1 q_3 - q_0 q_2) & 2(q_0 q_1 + q_2 q_3) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix}, \end{aligned}$$

Lze získat následující vztahy

$$\alpha = \operatorname{atan}2(2(q_0 q_1 + q_2 q_3), q_0^2 - q_1^2 - q_2^2 + q_3^2), \quad (2.152)$$

$$\beta = \operatorname{arcsin}(2(q_0 q_2 - q_1 q_3)), \quad (2.153)$$

$$\gamma = \operatorname{atan}2(2(q_0 q_3 + q_1 q_2), q_0^2 + q_1^2 - q_2^2 - q_3^2). \quad (2.154)$$

2.9.8 Nelineární model vyjádřený v kvaternionech

Nelineární model se z větší části nebude od původního modelu, který je vyjádřený v Eulerových úhlech, lišit. Hlavním rozdílem bude rozšíření počtu neznámých o dva. Eulerovy úhly a jejich úhlové rychlosti jsou zde vyjádřeny v kvaternionech, tedy v každém případě je o jeden parametr více. Opět budou vyjádřeny v pohyblivé soustavě, které budou následně převedeny do inerciální soustavy. Rovnice, která popisuje působení síly na pohyblivou soustavu souřadnic je ve tvaru

$$\vec{F}_1 = \vec{F}_r + \underbrace{mg\mathbf{R}_q^T}_{\vec{F}_g} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}. \quad (2.155)$$

Lze si všimnout, že oproti vztahu (2.82) se liší pouze v použití matice rotace, která je zde vyjádřena v kvaternionech. Celkový moment, který působí na UAV je ve stejném tvaru jako (2.83), kde

$$\vec{\tau}_1 = \vec{\tau}_r.$$

Jako v původním případě budou vyjádřeny jednotlivé druhé derivace,

$$\dot{\vec{v}}_1 = \vec{\omega}_1 \times \vec{v}_1 + \frac{1}{m} \vec{F}_r + g\mathbf{R}_q^T \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \quad (2.156)$$

$$\dot{\vec{\omega}}_1 = \frac{1}{J_1} (\vec{\tau}_r - (\vec{\omega}_1 \times J_1 \vec{\omega}_1)). \quad (2.157)$$

Nyní zbývá převést pohybové rovnice do inerciální soustavy souřadnic, tedy

$$\dot{\vec{v}}_0 = \mathbf{R}_q \dot{\vec{v}}_1. \quad (2.158)$$

Pro vyjádření druhé derivace kvaternionu je nutné využít vztah (2.149), díky kterému je získána poslední potřebná rovnice pro nelineární model

$$\ddot{\vec{\mathbf{q}}} = \frac{1}{2}\dot{W}(\vec{\mathbf{q}})\vec{\omega}_1 + \frac{1}{2}W(\vec{\mathbf{q}})\dot{\vec{\omega}}_1. \quad (2.159)$$

Rovnovážný stav

Řídící signál, který je nutný, aby UAV zůstalo v rovnovážném stavu, je beze změny. Jediná nutná modifikace zde spočívá v převedení úhlů a úhlové rychlosti v rovnovážném stavu do reprezentace v kvaternionech. Rovnovážný stav je definován jako

$$\vec{\xi}_{eq} = \begin{bmatrix} x_{eq} \\ y_{eq} \\ z_{eq} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -1 \end{bmatrix}, \quad \vec{v}_{eq} = \begin{bmatrix} u_{eq} \\ v_{eq} \\ w_{eq} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \quad \vec{\mathbf{q}}_{eq} = \begin{bmatrix} q_{0eq} \\ q_{1eq} \\ q_{2eq} \\ q_{3eq} \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad \dot{\vec{\mathbf{q}}}_{eq} = \begin{bmatrix} \dot{q}_{0eq} \\ \dot{q}_{1eq} \\ \dot{q}_{2eq} \\ \dot{q}_{3eq} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}. \quad (2.160)$$

Opět i zde jsou všechny veličiny rovnovážného stavu vztaženy k inerciální soustavě souřadnic.

3 Návrh řídicího systému

V této kapitole bude navržen řídicí systém pro obě uvažované varianty UAV. Nejdříve je provedena linearizace v rovnovážných bodech, které byly vypočteny v minulé kapitole. Pro lineární systém je pak navrženo řízení pomocí stavové zpětné vazby, které je zde využito pro stabilizaci UAV. Stavová zpětná vazba ovšem nutně vyžaduje znalost celého stavu systému.

Pro polohovou regulaci bude stavový popis rozšířen o integrátory, které zabezpečí správnou reakci na referenční signál. Dále se tato kapitola zabývá i případem, kdy není znám celý stavový vektor.

3.1 Linearizace

Lineární model je uvažován ve tvaru

$$\dot{\vec{x}}(t) = A\vec{x}(t) + B\vec{u}(t), \quad (3.1)$$

$$\vec{y}(t) = C\vec{x}(t), \quad (3.2)$$

kde $\vec{x} \in \mathbb{R}^n$ je stavový vektor, $\vec{u} \in \mathbb{R}^m$ je vektor vstupu, $\vec{y} \in \mathbb{R}^p$ je výstupní vektor, $A \in \mathbb{R}^{n \times n}$ je matice dynamiky, $B \in \mathbb{R}^{n \times m}$ je matice řízení a $C \in \mathbb{R}^{p \times n}$ je výstupní matice systému. Parametr n reprezentuje dimenzi stavového vektoru \vec{x} , m je zase dimenze vektoru řízení \vec{u} a p představuje dimenzi výstupního vektoru \vec{y} .

Matice dynamiky A je pak definována jako

$$A = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \frac{\partial f_n}{\partial x_2} & \cdots & \frac{\partial f_n}{\partial x_n} \end{bmatrix}, \quad (3.3)$$

kde f_i jsou příslušné stavové funkce. Podobně lze získat i matici řízení B a matici výstupu C , i v tomto případě se jedná o Jacobiho matice, tedy

$$B = \begin{bmatrix} \frac{\partial f_1}{\partial u_1} & \frac{\partial f_1}{\partial u_2} & \cdots & \frac{\partial f_1}{\partial u_m} \\ \frac{\partial f_2}{\partial u_1} & \frac{\partial f_2}{\partial u_2} & \cdots & \frac{\partial f_2}{\partial u_m} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial u_1} & \frac{\partial f_n}{\partial u_2} & \cdots & \frac{\partial f_n}{\partial u_m} \end{bmatrix}, \quad C = \begin{bmatrix} \frac{\partial h_1}{\partial x_1} & \frac{\partial h_1}{\partial x_2} & \cdots & \frac{\partial h_1}{\partial x_n} \\ \frac{\partial h_2}{\partial x_1} & \frac{\partial h_2}{\partial x_2} & \cdots & \frac{\partial h_2}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial h_n}{\partial x_1} & \frac{\partial h_n}{\partial x_2} & \cdots & \frac{\partial h_n}{\partial x_n} \end{bmatrix}, \quad (3.4)$$

kde h_i je příslušná výstupní funkce systému. Pro oba případy je linearizace provedena v rovnovážném stavu, který byl již předem definován - vztah (2.89), tedy

$$\vec{\xi}_{eq} = \begin{bmatrix} x_{eq} \\ y_{eq} \\ z_{eq} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -1 \end{bmatrix}, \quad \vec{v}_{eq} = \begin{bmatrix} u_{eq} \\ v_{eq} \\ w_{eq} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \quad \vec{\eta}_{eq} = \begin{bmatrix} \alpha_{eq} \\ \beta_{eq} \\ \gamma_{eq} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \quad \dot{\vec{\eta}}_{eq} = \begin{bmatrix} \dot{\alpha}_{eq} \\ \dot{\beta}_{eq} \\ \dot{\gamma}_{eq} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}.$$

Linearizace je provedena na modelu s Eulerovými úhly, které oproti kvaternionům popisují rotaci pouze za pomoci tří parametrů. Kvaterniony při stejném popisu vyžadují čtyři parametry, tedy není to minimální reprezentace a jednotlivé složky jsou mezi sebou již závislé.

Jelikož je kvadroptéra i omnikoptéra linearizována ve stejném rovnovážném bodě, jediný rozdíl bude u řízení pro setrvání v rovnovážném stavu.

Stav \vec{x} je zvolen jako

$$\vec{x} = [x \ y \ z \ \alpha \ \beta \ \gamma \ \dot{x} \ \dot{y} \ \dot{z} \ \dot{\alpha} \ \dot{\beta} \ \dot{\gamma}]^T. \quad (3.5)$$

Jednotlivé stavové funkce jsou

$$f_1 = \dot{x}, \quad (3.6)$$

$$f_2 = \dot{y}, \quad (3.7)$$

$$f_3 = \dot{z}, \quad (3.8)$$

$$f_4 = \dot{\alpha}, \quad (3.9)$$

$$f_5 = \dot{\beta}, \quad (3.10)$$

$$f_6 = \dot{\gamma}, \quad (3.11)$$

$$f_7 = \dot{\nu}_0 \ x, \quad (3.12)$$

$$f_8 = \dot{\nu}_0 \ y, \quad (3.13)$$

$$f_9 = \dot{\nu}_0 \ z, \quad (3.14)$$

$$f_{10} = \ddot{\eta}_0 \ \alpha, \quad (3.15)$$

$$f_{11} = \ddot{\eta}_0 \ \beta, \quad (3.16)$$

$$f_{12} = \ddot{\eta}_0 \ \gamma, \quad (3.17)$$

kde $\dot{\nu}_0$ je získáno ze vztahu (2.87) a $\ddot{\eta}_0$ je ze vztahu (2.88). Jednotlivé vztahy se samozřejmě liší v závislosti na uvažovaném UAV. Jelikož se tato práce zabývá návrhem rovnou dvou UAV, budou pro obě varianty vytvořeny lineární modely v rovnovážném stavu (2.89).

3.1.1 Kvadroptéra

V linearizaci bude využito řízení u_{eq} (2.92), díky které je kvadroptéra schopna setrvat v rovnovážném stavu při působení nulových poruch. Parametr u_{eq} je ve tvaru

$$u_{eq} = \frac{1}{4} \frac{mg}{K_T} = u_{eq1} = u_{eq2} = u_{eq3} = u_{eq4}.$$

Nyní bude přistoupeno k výpočtu matice dynamiky A a matice řízení B , výstupní matice C nyní nebude uvažována. Matice dynamiky A je ve tvaru

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & \frac{K_T(u_{1eq} + u_{2eq} + u_{3eq} + u_{4eq})}{m} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -\frac{K_T(u_{1eq} + u_{2eq} + u_{3eq} + u_{4eq})}{m} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{K_{\tau\tau}(u_{1eq} - u_{2eq} + u_{3eq} - u_{4eq})}{J_{zz}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -\frac{K_{\tau\tau}(u_{1eq} - u_{2eq} + u_{3eq} - u_{4eq})}{J_{zz}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -\frac{lK_T(u_{1eq} - u_{3eq})}{J_{yy}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad (3.18)$$

pokud budou vzaty následující parametry

$$\begin{aligned} J_{xx} &= J_{yy} = 0.1 \text{ kg} \cdot \text{m}^2, \\ J_{zz} &= 0.2 \text{ kg} \cdot \text{m}^2, \\ K_T &= 1, \\ K_{\tau\tau} &= 1, \\ m &= 5 \text{ kg}, \\ l &= 0.1 \text{ m}, \end{aligned}$$

bude matice dynamiky

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 9.81 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -9.81 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}. \quad (3.19)$$

Matice dynamiky A (3.19) má 12 vlastních čísel nulových, z čehož vyplývá, že systém je nestabilní, jelikož se vlastní čísla nenachází v levé otevřené komplexní polorovině. Stejný postup je aplikován i pro matici řízení B , tedy

$$B = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -\frac{K_T}{m} & -\frac{K_T}{m} & -\frac{K_T}{m} & -\frac{K_T}{m} \\ 0 & \frac{lK_T}{J_{xx}} & 0 & -\frac{lK_T}{J_{xx}} \\ \frac{lK_T}{J_{yy}} & 0 & -\frac{lK_T}{J_{yy}} & 0 \\ -\frac{K_{\tau\tau}}{J_{zz}} & \frac{K_{\tau\tau}}{J_{zz}} & -\frac{K_{\tau\tau}}{J_{zz}} & \frac{K_{\tau\tau}}{J_{zz}} \end{bmatrix}, \quad (3.20)$$

kde po dosazení stejných parametrů je získán tvar

$$B = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -0.2 & -0.2 & -0.2 & -0.2 \\ 0 & 1 & 0 & -1 \\ 1 & 0 & -1 & 0 \\ -5 & 5 & -5 & 5 \end{bmatrix}. \quad (3.21)$$

Jelikož je matice dynamiky (3.19) nestabilní, je nutné ji stabilizovat. To je možné pouze tehdy, pokud jsou všechna nestabilní vlastní čísla matice dynamiky A říditelná. Pro splnění říditelnosti musí mít tzv. matice říditelnosti stejný stupeň jako je stavů systému. Matice říditelnosti je definována vztahem

$$M_C = [B \ AB \ \dots \ A^{n-1}B], \quad (3.22)$$

její hodnost, tedy kolik má maximální počet nezávislých řádků/sloupců, je rovna dvanácti, což je i počet stavů systému. Matici dynamiky A (3.19) je tedy možné stabilizovat. Stabilizace bude provedena pomocí stavové zpětné vazby.

3.1.2 Omniskoptéra

U omniskoptéry bude proveden stejný postup. Řízení pro rovnovážný stav je definováno vztahy (2.94) - (2.101). Pro připomenutí je se jedná o

$$\begin{aligned} u_{1eq} &= u_{4eq}, \\ u_{2eq} &= \frac{1}{8} \frac{-gmK_{\tau\tau} \sqrt{3} - 2gmlK_T \sqrt{3} + 8K_T^2 l u_{4eq}}{K_T^2 l}, \\ u_{3eq} &= \frac{1}{8} \frac{-gmK_{\tau\tau} \sqrt{3} - 2gmlK_T \sqrt{3} + 8K_T^2 l u_{4eq}}{K_T^2 l}, \\ u_{4eq} &= u_{4eq}, \\ u_{5eq} &= -\frac{1}{24} \frac{\sqrt{3} (3gmK_{\tau\tau} - 6gmlK_T - 8K_T^2 \sqrt{3} l u_{7eq})}{K_T^2 l}, \\ u_{6eq} &= u_{7eq}, \\ u_{7eq} &= u_{7eq}, \\ u_{8eq} &= -\frac{1}{24} \frac{\sqrt{3} (3gmK_{\tau\tau} - 6gmlK_T - 8K_T^2 \sqrt{3} l u_{7eq})}{K_T^2 l}. \end{aligned}$$

Matice dynamiky nyní bude v obecném tvaru komplikovanější, protože i model omnikoptéry je složitější, tedy

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & a_1 & a_2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -a_1 & 0 & a_3 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & a_4 & a_5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -a_6 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & a_6 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & a_7 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad (3.23)$$

kde jednotlivé složky jsou

$$a_1 = \frac{1}{3} \frac{1}{m} K_T \sqrt{3} (u_{8eq} + u_{4eq} - u_{2eq} - u_{3eq} - u_{7eq} + u_{5eq} + u_{1eq} - u_{6eq}),$$

$$a_2 = -\frac{1}{6} \frac{1}{m} K_T (-3u_{1eq} + u_{1eq} \sqrt{3} - 3u_{2eq} - u_{2eq} \sqrt{3} + 3u_{3eq} + u_{3eq} \sqrt{3} + 3u_{4eq} - u_{4eq} \sqrt{3} + 3u_{5eq} - u_{5eq} \sqrt{3} + 3u_{6eq} + u_{6eq} \sqrt{3} - 3u_{7eq} - u_{7eq} \sqrt{3} - 3u_{8eq} + u_{8eq} \sqrt{3}),$$

$$a_3 = \frac{1}{6} \frac{1}{m} K_T (u_{8eq} \sqrt{3} - u_{4eq} \sqrt{3} + u_{2eq} \sqrt{3} - u_{3eq} \sqrt{3} + u_{7eq} \sqrt{3} - u_{5eq} \sqrt{3} + u_{1eq} \sqrt{3} - u_{6eq} \sqrt{3} + 3u_{1eq} - 3u_{2eq} + 3u_{3eq} - 3u_{4eq} - 3u_{5eq} + 3u_{6eq} - 3u_{7eq} + 3u_{8eq}),$$

$$a_4 = \frac{1}{6} \frac{1}{m} K_T (-3u_{1eq} + u_{1eq} \sqrt{3} - 3u_{2eq} - u_{2eq} \sqrt{3} + 3u_{3eq} + u_{3eq} \sqrt{3} + 3u_{4eq} - u_{4eq} \sqrt{3} + 3u_{5eq} - u_{5eq} \sqrt{3} + 3u_{6eq} + u_{6eq} \sqrt{3} - 3u_{7eq} - u_{7eq} \sqrt{3} - 3u_{8eq} + u_{8eq} \sqrt{3}),$$

$$a_5 = -\frac{1}{6} \frac{1}{m} K_T (u_{8eq} \sqrt{3} - u_{4eq} \sqrt{3} + u_{2eq} \sqrt{3} - u_{3eq} \sqrt{3} + u_{7eq} \sqrt{3} - u_{5eq} \sqrt{3} + u_{1eq} \sqrt{3} - u_{6eq} \sqrt{3} + 3u_{1eq} - 3u_{2eq} + 3u_{3eq} - 3u_{4eq} - 3u_{5eq} + 3u_{6eq} - 3u_{7eq} + 3u_{8eq}),$$

$$a_6 = \frac{1}{3} \frac{1}{J_{zz}} \sqrt{3} (-K_{\tau\tau} u_{1eq} + K_{\tau\tau} u_{2eq} + K_{\tau\tau} u_{3eq} - K_{\tau\tau} u_{4eq} - K_{\tau\tau} u_{5eq} + K_{\tau\tau} u_{6eq} + K_{\tau\tau} u_{7eq} - K_{\tau\tau} u_{8eq} + 2lK_T u_{1eq} - 2lK_T u_{2eq} - 2lK_T u_{3eq} + 2lK_T u_{4eq} - 2lK_T u_{5eq} + 2lK_T u_{6eq} + 2lK_T u_{7eq} - 2lK_T u_{8eq}),$$

$$\begin{aligned}
a_7 = & -\frac{1}{6} \frac{1}{J_{yy}} (6lK_T u_{2eq} - K_{\tau\tau} u_{1eq} \sqrt{3} + 6K_T u_{1eq} + K_{\tau\tau} u_{2eq} \sqrt{3} - 6K_T u_{3eq} - K_{\tau\tau} u_{3eq} \sqrt{3} \\
& - 6lK_T u_{4eq} + K_{\tau\tau} u_{4eq} \sqrt{3} + 6lK_T u_{5eq} + K_{\tau\tau} u_{5eq} \sqrt{3} + 6lK_T u_{6eq} - K_{\tau\tau} u_{6eq} \sqrt{3} \\
& - 6lK_T u_{7eq} + K_{\tau\tau} u_{7eq} \sqrt{3} - 6lK_T u_{8eq} - K_{\tau\tau} u_{8eq} \sqrt{3} + 2l\sqrt{3}K_T u_{1eq} - 2l\sqrt{3}K_T u_{2eq} \\
& + 2l\sqrt{3}K_T u_{3eq} - 2l\sqrt{3}K_T u_{4eq} + 2l\sqrt{3}K_T u_{5eq} - l\sqrt{3}K_T u_{6eq} + 2l\sqrt{3}K_T u_{7eq} \\
& - 2l\sqrt{3}K_T u_{8eq} + 3K_{\tau\tau} u_{1eq} + 3K_{\tau\tau} u_{2eq} - 3K_{\tau\tau} u_{3eq} - 3K_{\tau\tau} u_{4eq} - 3K_{\tau\tau} u_{5eq} - 3K_{\tau\tau} u_{6eq} \\
& + 3K_{\tau\tau} u_{7eq} + 3K_{\tau\tau} u_{8eq}).
\end{aligned}$$

Pokud budou využity tyto parametry

$$\begin{aligned}
J_{xx} = J_{yy} = J_{zz} &= 0.1 \text{ kg} \cdot \text{m}^2, \\
K_T &= 1, \\
K_{\tau\tau} &= 1, \\
m &= 5 \text{ kg}, \\
l &= 0.1 \text{ m},
\end{aligned}$$

bude matice dynamiky A ve tvaru

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 9.81 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -9.81 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}. \quad (3.24)$$

Lze si všimnout zajímavé skutečnosti a to takové, že matice dynamiky otevřeného systému kvadrokoptéry (3.19) a matice dynamiky otevřeného systému omnikoptéry (3.24) jsou shodné. Mají tedy stejná vlastní čísla a v okolí rovnovážného stavu se chovají naprosto totožně. To vyplývá z návrhu matematického modelu. Oba dva modely popisují pohyb tuhého tělesa, kde je řízení kompenzována gravitace. Tedy matice dynamiky (3.24) je také nestabilní a je nutné ji stejným způsobem stabilizovat.

Rozdíl bude ovšem v matici řízení B , jelikož nyní obsahuje UAV 8 akčních prvků. Matice

řízení je ve tvaru

$$B = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -b_1 & -b_2 & b_2 & b_1 & b_1 & b_2 & -b_2 & -b_1 \\ -b_2 & b_1 & -b_1 & b_2 & b_2 & -b_1 & b_1 & -b_2 \\ -b_3 & b_3 & b_3 & -b_3 & -b_3 & b_3 & b_3 & -b_3 \\ \frac{b_4}{J_{xx}} & \frac{b_5}{J_{xx}} & -\frac{b_5}{J_{xx}} & -\frac{b_4}{J_{xx}} & \frac{b_6}{J_{xx}} & \frac{b_7}{J_{xx}} & -\frac{b_7}{J_{xx}} & -\frac{b_6}{J_{xx}} \\ \frac{b_5}{J_{yy}} & -\frac{b_4}{J_{yy}} & \frac{b_4}{J_{yy}} & -\frac{b_5}{J_{yy}} & \frac{b_7}{J_{yy}} & -\frac{b_6}{J_{yy}} & \frac{b_6}{J_{yy}} & -\frac{b_7}{J_{yy}} \\ b_8 & -b_8 & -b_8 & b_8 & -b_9 & b_9 & b_9 & -b_9 \end{bmatrix}, \quad (3.25)$$

kde jednotlivé složky jsou

$$b_1 = \frac{1}{6} \frac{1}{m} K_T (3 + \sqrt{3}),$$

$$b_2 = \frac{1}{6} \frac{1}{m} K_T (-3 + \sqrt{3}),$$

$$b_3 = \frac{1}{3} \frac{1}{m} K_T \sqrt{3},$$

$$b_4 = \frac{1}{6} (-3K_{\tau\tau} + 2l\sqrt{3}K_T - 6lK_T - K_{\tau\tau}\sqrt{3}),$$

$$b_5 = \frac{1}{6} (3K_{\tau\tau} + 2l\sqrt{3}K_T + 6lK_T - K_{\tau\tau}\sqrt{3}),$$

$$b_6 = \frac{1}{6} (3K_{\tau\tau} + 2l\sqrt{3}K_T - 6lK_T + K_{\tau\tau}\sqrt{3}),$$

$$b_7 = \frac{1}{6} (-3K_{\tau\tau} + 2l\sqrt{3}K_T + 6lK_T + K_{\tau\tau}\sqrt{3}),$$

$$b_8 = \frac{1}{3} \frac{1}{J_{zz}} \sqrt{3} (-K_{\tau\tau} + 2lK_T),$$

$$b_9 = \frac{1}{3} \frac{1}{J_{zz}} \sqrt{3} (K_{\tau\tau} + 2lK_T).$$

Po dosazení parametrů je matice (3.25) ve tvaru

$$B = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -0.158 & 0.0424 & -0.0424 & 0.158 & 0.158 & -0.0424 & 0.0424 & -0.158 \\ 0.0424 & 0.158 & -0.158 & -0.0424 & -0.0424 & -0.158 & 0.158 & 0.0424 \\ -0.115 & 0.115 & 0.115 & -0.115 & -0.115 & 0.115 & 0.115 & -0.115 \\ -6.16 & 2.74 & -2.74 & 6.16 & 5.53 & -0.41 & 0.41 & -5.53 \\ 2.87 & 6.43 & -6.43 & -2.87 & -0.42 & -5.78 & 5.78 & 0.42 \\ -2.08 & 2.08 & 2.08 & -2.08 & -3.11 & 3.11 & 3.11 & -3.11 \end{bmatrix}. \quad (3.26)$$

I v tomto případě bude ověřena říditelnost, jelikož ale kvadroptéra je říditelná, lze očekávat, že i omnioptéra bude říditelná, zvláště když má o čtyři akční prvky více. I zde po dosazení do vztahu (3.22) vyjde řád matice říditelnosti 12. Lze tedy matici dynamiky A (3.24) stabilizovat přiřazením stabilních vlastních čísel stavovou zpětnou vazbou.

3.2 Jordanova normální forma

Jordanova normální forma matice A je speciální druh čtvercové matice, která se skládá s Jordanových bloků. Každý blok pak reprezentuje jedno vlastní číslo matice A , tedy

$$J_F = \begin{bmatrix} L_1 & 0 & 0 & \dots & 0 \\ 0 & L_2 & 0 & \dots & 0 \\ 0 & 0 & L_3 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & L_n \end{bmatrix}, \quad (3.27)$$

kde L_i reprezentuje právě zmiňované Jordanovy bloky a n je počet Jordanových bloků. Každý takový Jordanův blok je ve tvaru

$$L_i = \begin{bmatrix} \lambda & 1 & & & \\ & \ddots & \ddots & & \\ & & \ddots & 1 & \\ & & & \ddots & \lambda \end{bmatrix}, \quad (3.28)$$

kde λ jsou vlastní čísla příslušného Jordanovo bloku. Jordanova normální forma se pak vypočte dle vztahu

$$A = T J_F T^{-1}, \quad (3.29)$$

kde A je matice podobná matici J_F a matice T je transformační matice mezi maticemi A a J_F . Z předchozího tedy vyplývá, že všechny vlastní čísla matice A budou na diagonále matice J_F .

3.2.1 Homogenní koule v prostoru bez gravitace

Pro názornost je zde uvedena Jordanova normální forma matice dynamiky systému homogenní koule v prostoru bez gravitace. Na tuto homogenní kouli nepůsobí žádná další síla, tedy ani žádný moment. Rovnice, které popisují působící síly a momenty vyjádřeny v pohyblivé soustavě, jsou

$$\vec{F}_1 = 0, \quad (3.30)$$

$$\vec{\tau}_1 = 0. \quad (3.31)$$

Pohybové rovnice vyjádřeny v pohyblivé soustavě pevně spojené s tělesem jsou tedy ve tvaru

$$\dot{\vec{v}}_1 = \vec{\omega}_1 \times \vec{v}_1, \quad (3.32)$$

$$\dot{\vec{\omega}}_1 = -\frac{1}{J_1}(\vec{\omega}_1 \times J_1 \vec{\omega}_1). \quad (3.33)$$

Tyto rovnice byly získány ze vztahů (2.84) a (2.85), kde jsou doplněny nulové působící síly a momenty. Nyní zbývá jen tyto rovnice převést do inerciální soustavy souřadnic, kdy

$$\dot{\vec{v}}_0 = \mathbf{R} \dot{\vec{v}}_1, \quad (3.34)$$

$$\ddot{\vec{\eta}}_0 = \dot{\mathbf{W}} \vec{\omega}_1 + \mathbf{W} \dot{\vec{\omega}}_1, \quad (3.35)$$

kde matice \mathbf{R} a \mathbf{W} jsou dány vztahy (2.16) a (2.41). Výsledná matice dynamiky A tohoto systému je pak ve tvaru

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}. \quad (3.36)$$

Pokud bude vypočtena Jordanova forma matice dynamiky A (3.36) je získán tvar

$$J_F = \begin{bmatrix} \begin{matrix} 0 & 1 \\ 0 & 0 \end{matrix} & & & & & \\ & \begin{matrix} 0 & 1 \\ 0 & 0 \end{matrix} & & & & \\ & & \begin{matrix} 0 & 1 \\ 0 & 0 \end{matrix} & & & \\ & & & \begin{matrix} 0 & 1 \\ 0 & 0 \end{matrix} & & \\ & & & & \begin{matrix} 0 & 1 \\ 0 & 0 \end{matrix} & \\ & & & & & \begin{matrix} 0 & 1 \\ 0 & 0 \end{matrix} \end{bmatrix}. \quad (3.37)$$

Lze si všimnout, že všechny vlastní čísla jsou nulová a také, že Jordanova forma obsahuje 6 identických Jordanovo bloků. Stejného výsledku je dosaženo jak pro kvadrokoptéru tak i pro omnikoptéru, pokud je řízení $u_i = 0$ a nepůsobí na UAV žádná síla - ani gravitační síla.

3.2.2 UAV

Pokud je UAV umístěno do rovnovážného stavu a řízení odpovídá $u_i = u_{ieq}$, tak pro kvadrokoptéru i omnikoptéru je dosaženo shodného výsledku, tedy Jordanova forma matice dynamiky kvadrokoptéry (3.19) a matice dynamiky omnikoptéry (3.24) je ve shodném tvaru

$$J_F = \begin{bmatrix} \begin{matrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{matrix} & & & \\ & \begin{matrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{matrix} & & \\ & & \begin{matrix} 0 & 1 \\ 0 & 0 \end{matrix} & & \\ & & & \begin{matrix} 0 & 1 \\ 0 & 0 \end{matrix} & \end{bmatrix}. \quad (3.38)$$

Lze si opět všimnout zajímavé skutečnosti. Dva Jordanovy bloky zůstávají nezměněné, nicméně z dalších čtyř se staly dva větší bloky rozměru 4×4 . Toto chování je způsobeno umístěním UAV do rovnovážného stavu a působení externích sil. Pokud je například UAV vychýleno o úhel v ose \vec{x}_0 nebo \vec{y}_0 , gravitační síla již nepůsobí ve směru osy \vec{z}_1 a tím pádem zde již nemáme šest stejných Jordanovo bloků, ale vazby mezi stavy jsou úzce provázány. Jediné dva bloky, které by měly odpovídat otáčení a pohybu ve směru osy \vec{z}_0 , zůstávají nezměněné.

3.3 Stavová zpětná vazba

Stavová zpětná vazba je způsob řízení systému, u kterého je ale nutná znalost celého stavu systému. V některých případech pro stabilizaci systému v uzavřené smyčce postačí znalost jen některých stavů. Stavová zpětná vazba je definována jako

$$\vec{u}(t) = F\vec{x}(t), \quad (3.39)$$

kde $F \in \mathbb{R}^{m \times n}$ je příslušný stavový regulátor. Pro uzavřený systém vychází nový stavový popis ve tvaru

$$\dot{\vec{x}}(t) = \underbrace{(A + BF)}_{\hat{A}} \vec{x}(t). \quad (3.40)$$

Úkolem je pak nalezení zpětnovazební matice F takové, aby matice dynamiky uzavřeného systému \hat{A} byla stabilní.

3.3.1 LQR

Stavová zpětná vazba, tedy nalezení stabilizujícího regulátoru, se dá navrhnout mnoha způsoby. V této práci bude pozornost věnována právě metodě lineárního kvadratického regulátoru. Konkrétně zde bude uvažována varianta s nekonečným horizontem. Hlavním úkolem LQR je minimalizovat kritérium

$$J_c = \int_0^{\infty} (\vec{x}(t)^T Q \vec{x}(t) + \vec{u}(t)^T R \vec{u}(t)) dt, \quad (3.41)$$

kde matice $Q \in \mathbb{R}^{n \times n}$ je pozitivně semidefinitní a musí splňovat ještě podmínku

$$Q = C^T C, \quad (3.42)$$

je-li požadována stabilita uzavřené smyčky, kde (C, D) je stabilizovatelná dvojice. Matice $R \in \mathbb{R}^{m \times m}$ je pozitivně definitní. Přitom matice Q definuje penalizace stavu systému a matice R penalizaci řízení. Výsledné zpětnovazební řízení je pak ve tvaru

$$F = -R^{-1} B^T P, \quad (3.43)$$

kde P je řešením Riccatiovy rovnice, tedy

$$A^T P + P A - P B R^{-1} B^T P + Q = 0. \quad (3.44)$$

Výpočet LQ regulátoru je proveden pomocí příkazu `lqr(...)`, který je implementovaný v *Matlabu*. Hlavní otázkou ale stále zůstává, jak správně zvolit matice Q a R . Nabízí se v podstatě neomezené množství kombinací, ne všechny ale vedou na vhodná řešení.

3.3.2 Symetrické LQR

Obě dvě UAV, kterými se tato práce zabývá, mají svou konstrukci navrženou symetricky. Pokud se například kvadrokoptéra otočí o 90 stupňů, bude stále stejná. Podobně tomu je i u omnikoptéry, která má v sobě také symetrii zakomponovanou. Tu lze ovšem rotovat mnoha jinými způsoby. Proto je velice výhodné využít symetrii i do návrhu řídicího systému. Tímto problémem se zabývá práce [30], zde budou pouze prezentována část výsledků z LQR. Před přistoupením k samotným výsledkům v oblasti symetrie LQR je nutné nejdříve zavést některé pojmy.

Vstupně-výstupní symetrie

Dvojice (A, B) , $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times mq}$, kde matice $B = [B_1, \dots, B_m]$ je složená ze submatic $B_i \in \mathbb{R}^{n \times q}$, $i = 1, \dots, m$, je nazývána vstupně symetrickou, jestliže existuje regulární transformační matice $T \in \mathbb{R}^{n \times n}$ taková, že platí

$$TA = AT, \quad (3.45)$$

$$TB_i = B_{i+1}, \quad i = 1, \dots, m-1, \quad (3.46)$$

$$TB_m = B_1. \quad (3.47)$$

Na základě předchozích podmínek bude definována matice G_{mq} jako

$$G_{mq} = \begin{bmatrix} 0_q & & & I_q \\ I_q & \ddots & & \\ & \ddots & \ddots & \\ & & I_q & 0_q \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & & & 1 \\ 1 & \ddots & & \\ & \ddots & \ddots & \\ & & 1 & 0 \end{bmatrix}}_{G_m} \otimes_k I_q = G_m \otimes_k I_q, \quad (3.48)$$

kde $0_q \in \mathbb{R}^{q \times q}$ je nulová matice, $I_q \in \mathbb{R}^{q \times q}$ je jednotková matice a \otimes_k je tzv. Kroneckerův součin. Matice G_{mq} je permutační maticí a tedy platí $G_{mq}G_{mq}^T = G_{mq}^T G_{mq} = I$. Podmínku (3.46) lze přepsat do tvaru

$$TB = BG_{mq}. \quad (3.49)$$

Systém (A, B, C) , $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times mq}$, $C \in \mathbb{R}^{r \times n}$ kde matice B a C jsou složené po řadě ze submatic $B_i \in \mathbb{R}^{n \times q}$ a $C_i \in \mathbb{R}^{r \times n}$, $i = 1, \dots, m$, $B = [B_1, \dots, B_m]$, $C = [C_1^T, \dots, C_m^T]^T$ je nazván vstupně-výstupně symetrickým, jestliže existuje regulární matice $T \in \mathbb{R}^{n \times n}$ taková, že platí

$$TA = AT,$$

$$TB_i = B_{i+1}, \quad i = 1, \dots, m-1,$$

$$TB_m = B_1,$$

$$C_i T^{-1} = C_{i+1}, \quad i = 1, \dots, m-1, \quad (3.50)$$

$$C_m T^{-1} = C_1. \quad (3.51)$$

Podmínky (3.46) a (3.50) lze přepsat do tvaru

$$\begin{aligned} TB &= BG_{mq}, \\ G_{mr}C &= CT, \end{aligned} \quad (3.52)$$

kde $G_{mq} = G_m \otimes_k I_q$ a $G_{mr} = G_m \otimes_k I_r$.

Vztahy (3.46) a (3.47) platí pouze pokud je vstupní symetrie taková, že transformací matice T jsou postupně vystřídány všechny vstupy systému. Vstupní symetrie ovšem může být různá.

Podmínky pro symetrické LQR

Nechť (A, B) , $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$, $\text{rank } B = m$, je vstupně symetrická dvojice s maticí symetrie $T \in \mathbb{R}^{n \times n}$, $\text{rank } T = n$, a nechť matice $C \in \mathbb{R}^{m \times n}$ doplňuje dvojici (A, B) na vstupně-výstupně symetrický čtvercový systém (C, A, B) , tj. matice C splňuje podmínku

$$C = G^T C T, \quad (3.53)$$

kde

$$G = \begin{bmatrix} 0 & & & 1 \\ 1 & \ddots & & \\ & \ddots & \ddots & \\ & & 1 & 0 \end{bmatrix}. \quad (3.54)$$

Dále nechť dvojice (A, B) je stabilizovatelná, dvojice (C, A) detekovatelná a matice F_s je definována vztahem

$$F_s = R^{-1} B^T P_s, \quad (3.55)$$

kde R je symetrická regulární matice splňující

$$R = G^T R G, \quad (3.56)$$

Q je symetrická pozitivně semidefinitní matice vyhovující podmínce

$$Q = T^T Q T, \quad (3.57)$$

$$Q = C^T C \quad (3.58)$$

a P_s je stabilizující řešení algebraické Riccatiové rovnice

$$A^T P_s + P_s A - P_s B R^{-1} B^T P_s + Q = 0. \quad (3.59)$$

Potom je dvojice $(A - B F_s, B)$ vstupně symetrická s maticí symetrie T .

Simulace - poznámka

Všechny následující simulace budou provedeny na nelineárních modelech (kapitola 2.9.8), kde budou aplikovány lineární regulátory. Čím dále se bude systém nacházet od rovnovážného bodu (2.89), tím lze více očekávat nestabilní chování uzavřené smyčky.

3.4 Kvadrokoptéra - stabilizace

Pro stabilizaci kvadrokoptéry bude využito symetrického lineárního kvadratického regulátoru. Matice symetrie T je pro tento případ zvolena na základě postupného otáčení o 90 stupňů okolo osy \vec{z}_1 . Nalézt matici symetrie lze i matematicky, proces je to ale dlouhý a komplikovaný, tento postup je k nalezení v práci [30]. Matice symetrie T je pro tento případ zvolena ve tvaru

$$T = \begin{bmatrix} 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \end{bmatrix}, \quad (3.60)$$

kde je vycházeno z toho, že pokud bude kvadrokoptéra otočena o 90 stupňů, dojde k posunu

$$\begin{aligned} x &\rightarrow -y, & y &\rightarrow x, & z &\rightarrow z, \\ \alpha &\rightarrow -\beta, & \beta &\rightarrow \alpha, & \gamma &\rightarrow -\gamma, \\ \dot{x} &\rightarrow -\dot{y}, & \dot{y} &\rightarrow \dot{x}, & \dot{z} &\rightarrow \dot{z}, \\ \dot{\alpha} &\rightarrow -\dot{\beta}, & \dot{\beta} &\rightarrow \dot{\alpha}, & \dot{\gamma} &\rightarrow -\dot{\gamma}. \end{aligned}$$

Tato matice splňuje podmínky (3.46) a (3.47) a také platí, že $T^4 = I$. Posledním krokem je matice G , pro ni lze využít vztah (3.48), tedy pro tento systém je G ve tvaru

$$G = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}. \quad (3.61)$$

Matice T a G nyní budou dosazeny do vztahů (3.56) a (3.57). Takto je obdržena soustava lineárních rovnic, po jejímž vyřešení se počet návrhových parametrů zredukuje. Nový tvar matice R bude

$$R_s = \begin{bmatrix} R_1 & r_1 & r_2 & r_1 \\ r_1 & R_1 & r_1 & r_2 \\ r_2 & r_1 & R_1 & r_1 \\ r_1 & r_2 & r_1 & R_1 \end{bmatrix}, \quad (3.62)$$

kde došlo k redukci na 3 návrhové parametry. Matice Q je zredukována na tvar

$$Q_s = \begin{bmatrix} Q_1 & 0 & 0 & q_1 & 0 & 0 & q_4 & 0 & 0 & q_8 & 0 & 0 \\ 0 & Q_1 & 0 & 0 & q_1 & 0 & 0 & q_4 & 0 & 0 & q_8 & 0 \\ 0 & 0 & Q_2 & 0 & 0 & 0 & 0 & 0 & q_5 & 0 & 0 & 0 \\ q_1 & 0 & 0 & Q_3 & 0 & 0 & q_2 & 0 & 0 & q_6 & 0 & 0 \\ 0 & q_1 & 0 & 0 & Q_3 & 0 & 0 & q_2 & 0 & 0 & q_6 & 0 \\ 0 & 0 & 0 & 0 & 0 & Q_4 & 0 & 0 & 0 & 0 & 0 & q_7 \\ q_4 & 0 & 0 & q_2 & 0 & 0 & Q_5 & 0 & 0 & q_3 & 0 & 0 \\ 0 & q_4 & 0 & 0 & q_2 & 0 & 0 & Q_5 & 0 & 0 & q_3 & 0 \\ 0 & 0 & q_5 & 0 & 0 & 0 & 0 & 0 & Q_6 & 0 & 0 & 0 \\ q_8 & 0 & 0 & q_6 & 0 & 0 & q_3 & 0 & 0 & Q_7 & 0 & 0 \\ 0 & q_8 & 0 & 0 & q_6 & 0 & 0 & q_3 & 0 & 0 & Q_7 & 0 \\ 0 & 0 & 0 & 0 & 0 & q_7 & 0 & 0 & 0 & 0 & 0 & Q_8 \end{bmatrix}. \quad (3.63)$$

Lze si tedy všimnout, že v tomto případě se povedla redukce na 16 návrhových parametrů. V ideálním případě by bylo pěkné, kdyby se počet parametrů zredukoval například na jeden. Ale i toto je velké ulehčení. Ve většině případů, pokud nemusí být splněny různé speciální požadavky na regulaci, se matice R a Q volí diagonální.

Redukce záleží na nalezené matici symetrie T a i na matici G , čím větší je v systému symetrie, tím více se počet volných parametrů zredukuje. U kvadrokoptéry by se nejspíše lepší matice symetrie T nalézt nedala, jelikož se postupně všechny vstupy systému vystřídají.

Pro jednoduchost byly matice Q_s a R_s zvoleny jako

$$Q_s = \begin{bmatrix} 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 10 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 10 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad (3.64)$$

$$R_s = I_{4 \times 4}. \quad (3.65)$$

Výsledná matice řízení F je k dispozici na přiloženém CD, vzhledem k její velikosti zde nebude v práci explicitně vypsána. Vlastní čísla, které přiřazuje zvolený stavový regulátor F systému

v uzavřené smyčce, jsou

$$\begin{bmatrix} -3.4895 \\ -1.0960 + 1.4609i \\ -1.0960 - 1.4609i \\ -1.3068 \\ -0.4899 + 0.4i \\ -0.4899 - 0.4i \end{bmatrix} \begin{bmatrix} -3.3432 \\ -1.0873 + 1.4629i \\ -1.0873 - 1.4629i \\ -1.3085 \\ -8.9526 \\ -1.0063 \end{bmatrix}. \quad (3.66)$$

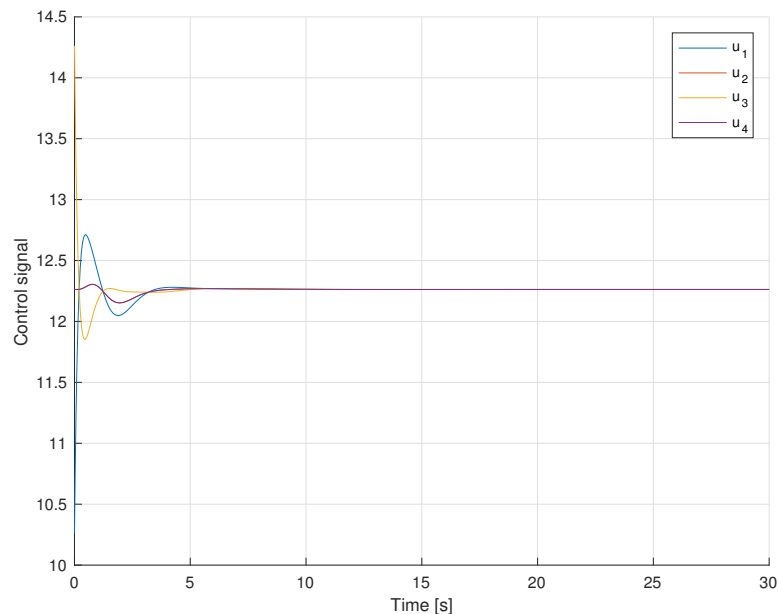
Dvě dvojice sdružených komplexních vlastních čísel budou odpovídat úhlům α a β a jejich derivacím. Výstup těchto stavových proměnných bude mít kmitavý průběh, což je ovšem dáno charakterem systému.

Nyní bude ověřeno, zda je symetrie zachována, pokud bude dron vychýlen z rovnovážného stavu ve směru osy \vec{x}_0 například o -2 m a následně bude to samé provedeno v ose \vec{y}_0 , musí být řízení a stav v prvním a druhém případě svázán vztahem

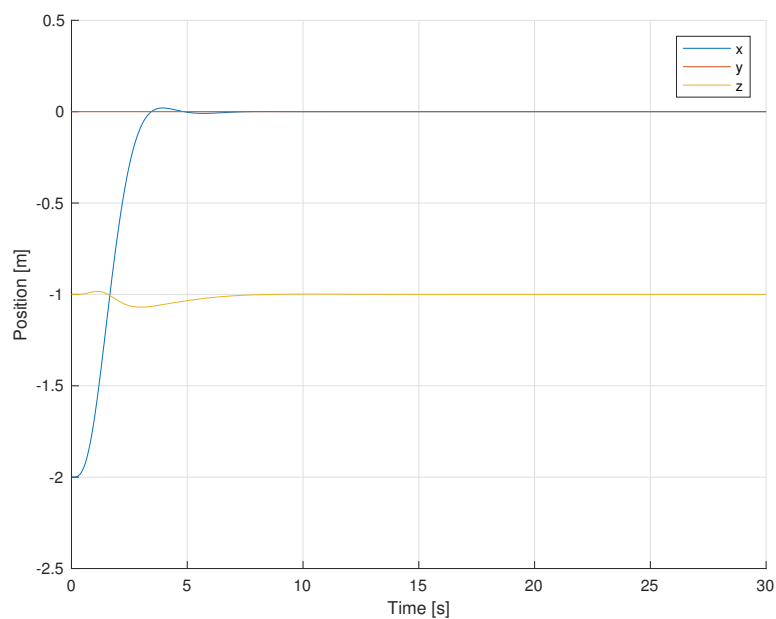
$$\vec{u}_I = G\vec{u}_{II}, \quad (3.67)$$

$$\vec{x}_I = T\vec{x}_{II}, \quad (3.68)$$

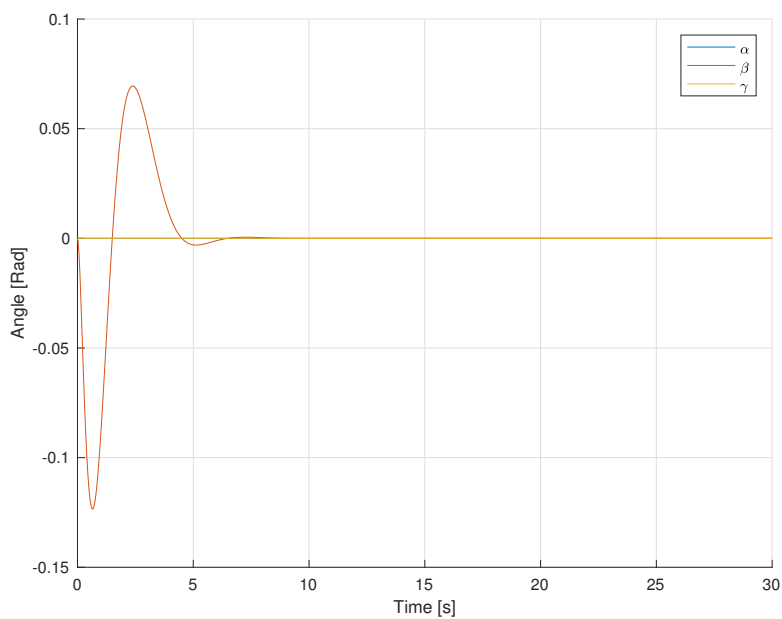
kde řízení \vec{u}_I a stav \vec{x}_I odpovídá vychýlení v ose \vec{x}_0 a řízení \vec{u}_{II} a stav \vec{x}_{II} odpovídá zase vychýlení v ose \vec{y}_0 .



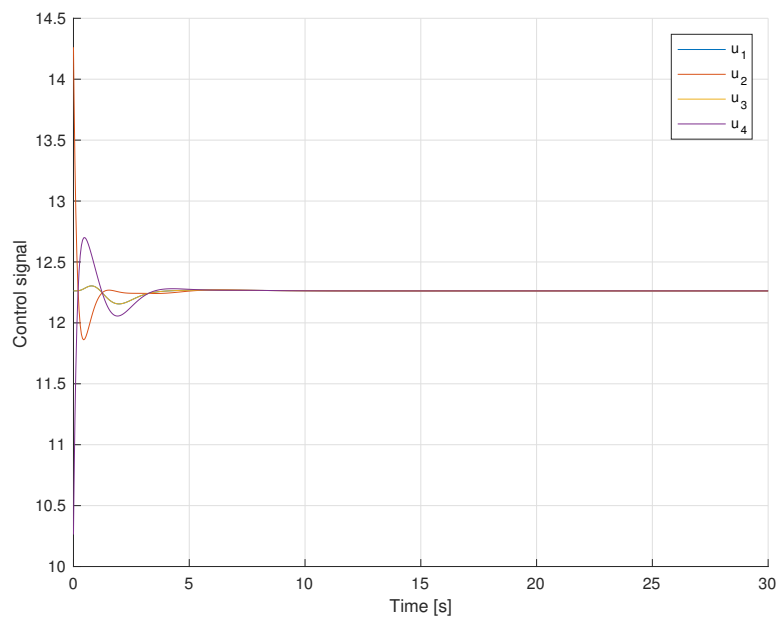
Obrázek 29: Řízení - z počáteční podmínky $x_{eq(1.polovina)} = [-2 \ 0 \ -1 \ 0 \ 0 \ 0]$



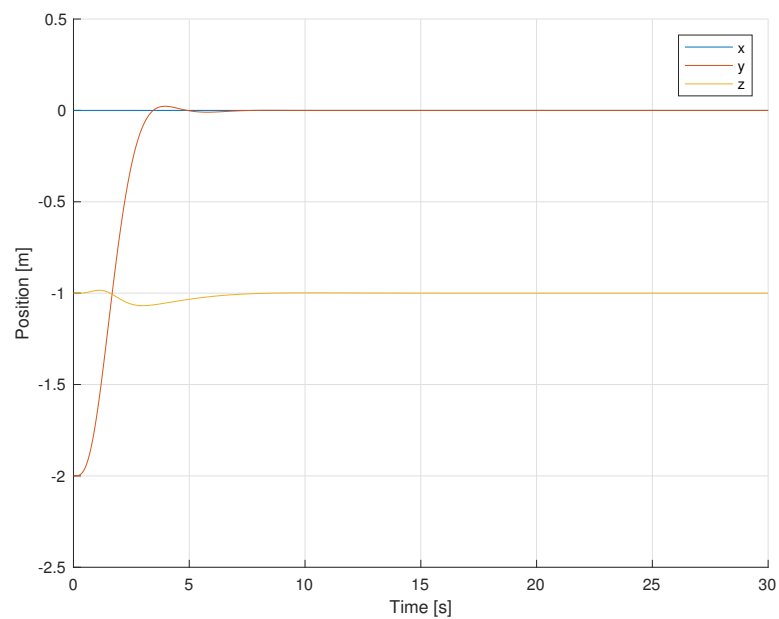
Obrázek 30: Pozice - z počáteční podmínky $x_{eq} (1.polovina) = [-2 \ 0 \ -1 \ 0 \ 0 \ 0]$



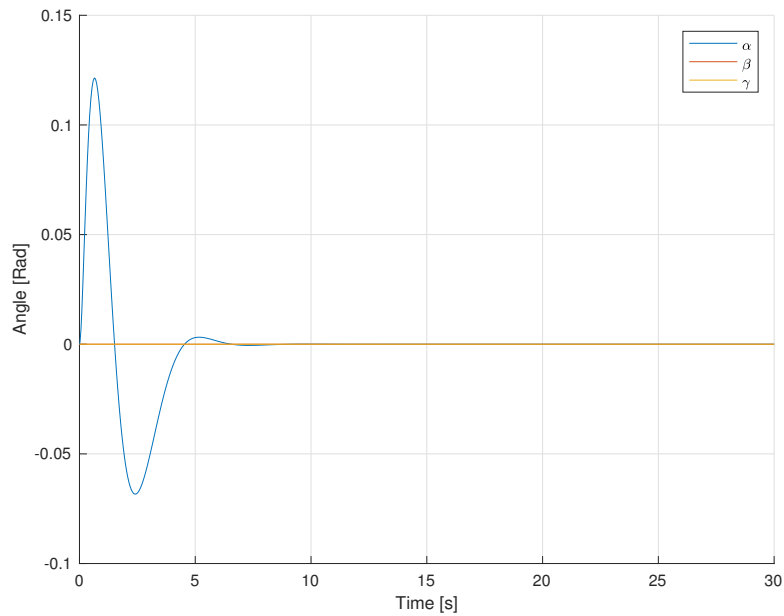
Obrázek 31: Úhly - z počáteční podmínky $x_{eq} (1.polovina) = [-2 \ 0 \ -1 \ 0 \ 0 \ 0]$



Obrázek 32: Řízení - z počáteční podmínky $x_{eq}(1.polovina) = [0 \quad -2 \quad -1 \quad 0 \quad 0 \quad 0]$



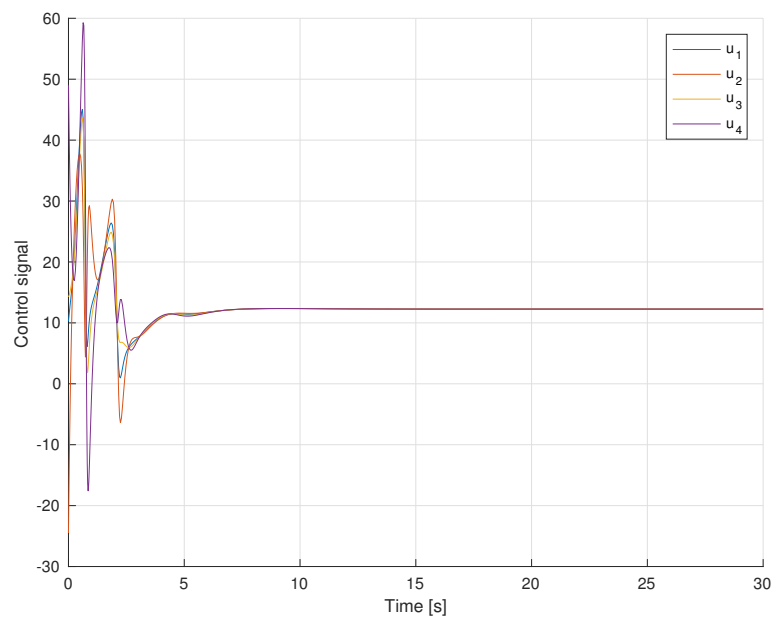
Obrázek 33: Pozice - z počáteční podmínky $x_{eq}(1.polovina) = [0 \quad -2 \quad -1 \quad 0 \quad 0 \quad 0]$



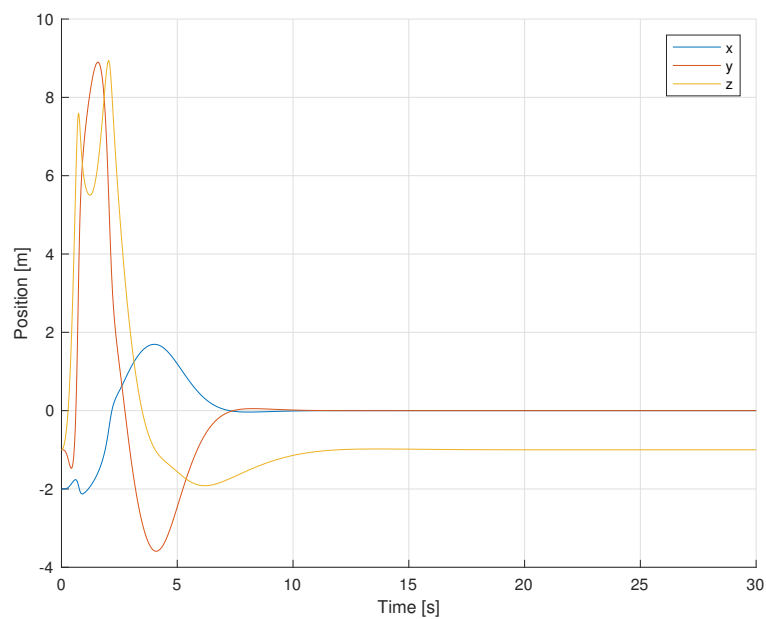
Obrázek 34: Úhly - z počáteční podmínky $x_{eq} (1.polovina) = [0 \quad -2 \quad -1 \quad 0 \quad 0 \quad 0]$

Jak lze vidět z předchozích grafů, symetrie systému je zachována a obě sady grafů splňují vztahy (3.67) a (3.68). Navržený regulátor F tedy zachovává symetrii systému otevřené smyčky i v uzavřené smyčce.

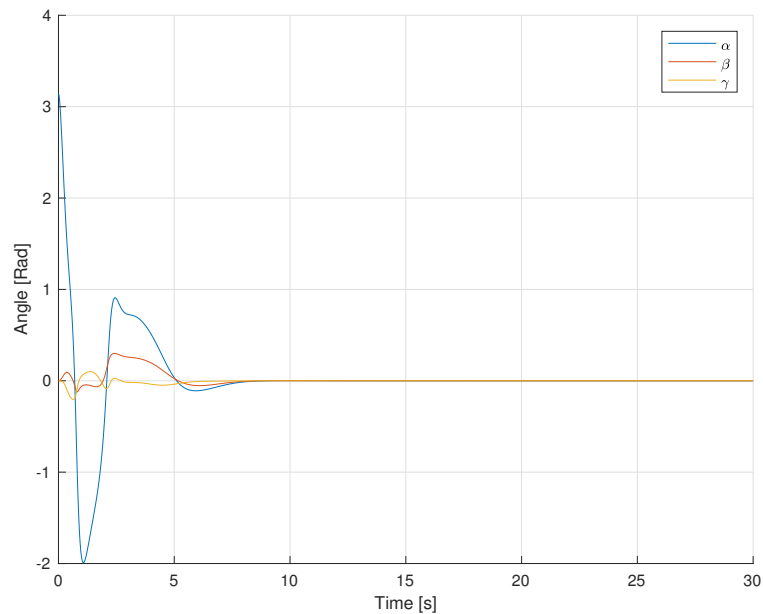
Pro demonstraci funkce stabilizujícího regulátoru je dron otočen směrem dolů, v ose \vec{x}_0 posunut o -2 metry a v ose \vec{y}_0 posunut o -1 m.



Obrázek 35: Řízení - z počáteční podmínky $x_{eq(1.polovina)} = [-2 \quad -1 \quad -1 \quad \pi \quad 0 \quad 0]$



Obrázek 36: Pozice - z počáteční podmínky $x_{eq(1.polovina)} = [-2 \quad -1 \quad -1 \quad \pi \quad 0 \quad 0]$



Obrázek 37: Úhly - z počáteční podmínky $x_{eq} (1.polovina) = [-2 \quad -1 \quad -1 \quad \pi \quad 0 \quad 0]$

Jak lze vidět z grafů, tak regulátor zareagoval správně i v případě otočení drona směrem dolů. Je ale nutné brát v úvahu, že dron musí být v dostatečné výšce, aby mohl být tento obrat proveden. Z grafů je patrné, že pro úspěšné provedení je vhodné mít desetimetrovou rezervu pod dronem. Tato rezerva přímo souvisí s hmotností kvadrokoptéry i s použitým stabilizačním regulátorem. Úhly se dostaly do přijatelných mezí již okolo 6-7 vteřin, to samé platí i pro polohy.

Tento regulátor je ovšem navržen pouze pro rovnovážný stav (2.89). Tedy jeho chování mimo tento stav může samozřejmě vést i k nestabilitě systému, což se ale v daném případě nestalo.

3.5 Omnikoptéra - stabilizace

Nalezení vhodné matice symetrie T bude v tomto případě mnohem náročnější úkol, jelikož omnikoptéra nabízí větší množství kombinací a tím i více možných adeptů na matici symetrie.

Pro tuto práci byla nakonec zvolena matice symetrie ve tvaru

$$T = \begin{bmatrix} -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad (3.69)$$

což odpovídá rotaci okolo osy \vec{z}_1 o 180 stupňů. Matice (3.69) byla vytvořena na základě jednoduchého pootočení celou omnikoptérou, tedy

$$\begin{aligned} x &\rightarrow -x & y &\rightarrow -y & z &\rightarrow z, \\ \alpha &\rightarrow -\alpha & \beta &\rightarrow -\beta & \gamma &\rightarrow \gamma, \\ \dot{x} &\rightarrow -\dot{x} & \dot{y} &\rightarrow -\dot{y} & \dot{z} &\rightarrow \dot{z}, \\ \dot{\alpha} &\rightarrow -\dot{\alpha} & \dot{\beta} &\rightarrow -\dot{\beta} & \dot{\gamma} &\rightarrow \dot{\gamma}. \end{aligned}$$

Matice G bude vytvořena pro případ, kdy

$$TB_1 = B_4, \quad TB_4 = B_1, \quad TB_2 = B_3, \quad TB_3 = B_2, \quad (3.70)$$

$$TB_5 = B_8, \quad TB_8 = B_5, \quad TB_6 = B_7, \quad TB_7 = B_6, \quad (3.71)$$

což lze přepsat do tvaru

$$T [B_1, B_2, B_5, B_6] = [B_4, B_3, B_8, B_7], \quad (3.72)$$

$$T [B_4, B_3, B_8, B_7] = [B_1, B_2, B_5, B_6]. \quad (3.73)$$

V tomto případě lze mluvit o tzv. blokově vstupně symetrickém systému. Pro tento případ je nutné matici G lze zapsat do tvaru

$$G = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}, \quad (3.74)$$

jednotlivé složky této matice odpovídají vztahům (3.70) a (3.71). Matice T a G jsou využity pro vtahy (3.56) a (3.57), kde jsou opět u matic R a Q zredukovány návrhové parametry. Matice penalizující řízení je zredukována na tvar

$$R_s = \begin{bmatrix} R_1 & r_1 & r_6 & r_{10} & r_{15} & r_{17} & r_{20} & r_{22} \\ r_1 & R_2 & r_2 & r_6 & r_{11} & r_{16} & r_{18} & r_{21} \\ r_6 & r_2 & R_2 & r_1 & r_7 & r_{12} & r_{16} & r_{19} \\ r_{10} & r_6 & r_1 & R_3 & r_3 & r_8 & r_{13} & r_{15} \\ r_{15} & r_{11} & r_7 & r_3 & R_3 & r_4 & r_9 & r_{14} \\ r_{17} & r_{16} & r_{12} & r_8 & r_4 & R_4 & r_5 & r_9 \\ r_{20} & r_{18} & r_{16} & r_{13} & r_9 & r_5 & R_4 & r_4 \\ r_{22} & r_{21} & r_{19} & r_{15} & r_{14} & r_9 & r_4 & R_3 \end{bmatrix}, \quad (3.75)$$

v tomto případě se počet návrhových parametrů zredukoval na 25. Matice penalizující jednotlivé stavy je upravena do tvaru

$$Q_s = \begin{bmatrix} Q_1 & q_1 & 0 & q_8 & q_{17} & 0 & q_{22} & q_{28} & 0 & q_{31} & q_{34} & 0 \\ q_1 & Q_2 & 0 & q_5 & q_9 & 0 & q_{20} & q_{23} & 0 & q_{30} & q_{32} & 0 \\ 0 & 0 & Q_3 & 0 & 0 & q_{10} & 0 & 0 & q_{24} & 0 & 0 & q_{33} \\ q_8 & q_5 & 0 & Q_4 & q_2 & 0 & q_{11} & q_{18} & 0 & q_{25} & q_{29} & 0 \\ q_{17} & q_9 & 0 & q_2 & Q_5 & 0 & q_6 & q_{12} & 0 & q_{21} & q_{26} & 0 \\ 0 & 0 & q_{10} & 0 & 0 & Q_6 & 0 & 0 & q_{13} & 0 & 0 & q_{27} \\ q_{22} & q_{20} & 0 & q_{11} & q_6 & 0 & Q_7 & q_3 & 0 & q_{14} & q_{19} & 0 \\ q_{28} & q_{23} & 0 & q_{18} & q_{12} & 0 & q_3 & Q_8 & 0 & q_7 & q_{15} & 0 \\ 0 & 0 & q_{24} & 0 & 0 & q_{13} & 0 & 0 & Q_9 & 0 & 0 & q_{16} \\ q_{31} & q_{30} & 0 & q_{25} & q_{21} & 0 & q_{14} & q_7 & 0 & Q_{10} & q_4 & 0 \\ q_{34} & q_{32} & 0 & q_{29} & q_{26} & 0 & q_{19} & q_{15} & 0 & q_4 & Q_{11} & 0 \\ 0 & 0 & q_{33} & 0 & 0 & q_{27} & 0 & 0 & q_{16} & 0 & 0 & Q_{12} \end{bmatrix}. \quad (3.76)$$

I v tomto případě došlo k významné redukci volných parametrů.

Pokud budou nyní porovnány redukované matice z kvadroptéry - (3.62) a (3.63) s nyní vypočtenými maticemi - (3.75) a (3.76), je vidět, že větší redukce proběhla u kvadroptéry. Tato skutečnost je dána tím, že kvadroptéry symetrie je odvozena od otočení o 90 stupňů okolo osy \vec{z}_1 a omnikoptéra se otáčí okolo stejné osy, ovšem o 180 stupňů.

Matice penalizující řízení (3.75) má nyní rozměr 8×8 , tedy je zde 8 akčních členů. Vzhledem ke skutečnosti, že omnikoptéra má pouze 6 stupňů volnosti, je vhodné vytvořit tzv. virtuální vstupy systému. Pokud by bylo ponecháno 8 vstupů systému, měl by stavový zpětný regulátor nekonečné množství řešení. Z tohoto důvodu je vytvořeno 6 virtuálních vstupů, kde bude již jednoznačné řešení regulátoru F . Jelikož je na volbě návrháře řídicího systému, jak jsou tyto vstupy zvoleny, je vhodné je zvolit tak, aby každý vstup znamenal pohyb jedním směrem, případně otáčení okolo jednoho úhlu. Nové vstupy budou tedy odpovídat

$$\begin{aligned} u_{v\ 1} &= \text{pohyb v ose } \vec{x}_1, \\ u_{v\ 2} &= \text{pohyb v ose } \vec{y}_1, \\ u_{v\ 3} &= \text{pohyb v ose } \vec{z}_1, \\ u_{v\ 4} &= \text{rotace okolo osy } \vec{x}_1, \\ u_{v\ 5} &= \text{rotace okolo osy } \vec{y}_1, \\ u_{v\ 6} &= \text{rotace okolo osy } \vec{z}_1. \end{aligned}$$

Cílem je tedy zavést transformační matici M , která bude schopna z 8 reálných vstupů systému vytvořit 6 virtuálních vstupů. Bude tedy požadováno, aby

$$\vec{u}(t) = M\vec{\hat{u}}(t), \quad (3.77)$$

kde $\vec{\hat{u}}(t) \in \mathbb{R}^6$ je virtuální vstup a $M \in \mathbb{R}^{8 \times 6}$ transformační matice mezi původními a virtuálními vstupy. Matici M lze vypočítat různými způsoby. Zde bylo využito rovnic (2.64), (2.65), (2.66) a (2.82), kde byl pro každý virtuální vstup požadavek, aby bylo zrychlení v daném směru rovno 1 a ostatní byly rovny 0. Tedy každý jeden virtuální vstup odpovídá jednomu sloupci matice M , kde řádky v tomto sloupci jsou dány hodnotami skutečných vstupů $\vec{u}(t)$. Vzhledem ke složitým vztahům zde bude prezentována pouze reálná forma matice M , tedy

$$M = \begin{bmatrix} 10.201 & 7.201 & 10 & 9.2075 & 12.9575 & 10 \\ 8.701 & 8.5 & 7.4019 & 11.0825 & 11.875 & 7.8349 \\ 11.5 & 8.701 & 7.4019 & 8.125 & 11.0825 & 7.8349 \\ 10 & 10 & 10 & 10 & 10 & 10 \\ 9.134 & 7.75 & 10 & 11.0825 & 12.9575 & 10 \\ 6.884 & 8.616 & 11.7321 & 12.9575 & 11.875 & 12.1651 \\ 10 & 10 & 11.7321 & 10 & 11.0825 & 12.1651 \\ 7.75 & 10.866 & 10 & 11.875 & 10 & 10 \end{bmatrix}. \quad (3.78)$$

Obecný tvar matice M je k dispozici na příloženém CD. Vzhledem k této skutečnosti je zavedena nová matice řízení \hat{B} , která je získána jako

$$\hat{B} = BM. \quad (3.79)$$

Pro porovnání, proč je předchozí postup vhodné provést, budou provedeny dvě simulace. V prvním případě není využito virtuálních vstupů a je ponecháno na *Matlabu*, aby vypočetl matici řízení F . V druhém případě bude využito virtuálních vstupů a bude vypočtena jiná matice řízení, tedy \hat{F} , která má za úkol stabilizovat matici dynamiky ze vztahu

$$\dot{\vec{x}}(t) = (A + \hat{B}\hat{F})\vec{x}(t). \quad (3.80)$$

Pro jednoduchost jsou uvažovány všechny matice pro symetrické LQR jako diagonální. Pro různé požadavky se dají samozřejmě tyto matice upravit s dodržáním podmínek symetrie. Jednotlivé matice, které jsou zde využity, jsou ve tvaru

$$Q_s = \hat{Q} = \begin{bmatrix} 10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 30 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 100 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 100 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad (3.81)$$

$$R_s = I_{8 \times 8}, \quad (3.82)$$

$$\hat{R} = I_{6 \times 6}, \quad (3.83)$$

kde \hat{Q} a \hat{R} jsou příslušné matice pro metodu LQR, při které jsou ale využity virtuální vstupy. Výsledné matice regulátorů jsou dostupné na příloženém CD, vzhledem k jejich velikosti nebudou v práci explicitně uváděny. Vlastní čísla, které přiřazuje stavovou zpětnou vazbou regulátor F , jsou ve tvaru

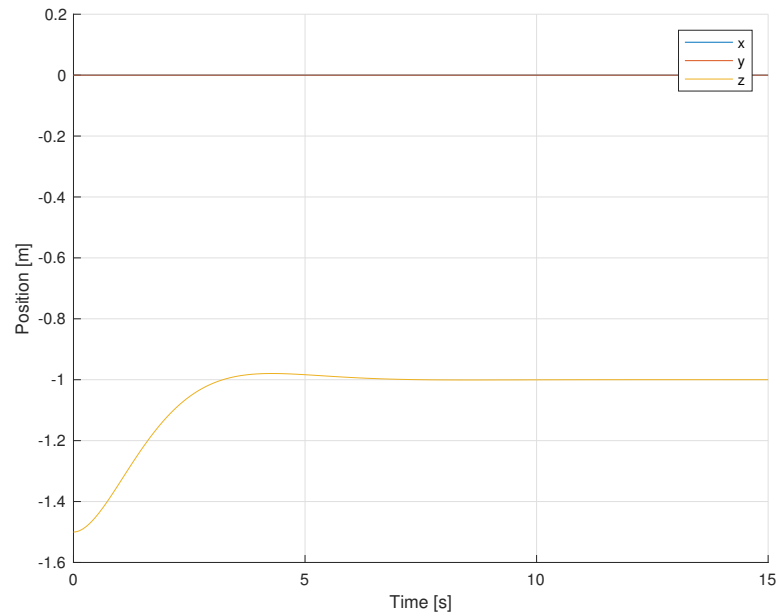
$$\begin{bmatrix} -10.2821 + 4.7727i \\ -10.2821 - 4.7727i \\ -9.9621 + 4.8521i \\ -9.9621 - 4.8521i \\ -7.4365 \\ -1.3345 + 1.1587i \end{bmatrix} \begin{bmatrix} -1.3345 - 1.1587i \\ -1.3347 + 1.1584i \\ -1.3347 - 1.1584i \\ -1.0335 \\ -0.4166 + 0.4110i \\ -0.4166 - 0.4110i \end{bmatrix} \quad (3.84)$$

a v případě využití virtuálních vstupů, tedy regulátoru \hat{F} , jsou pak vlastní čísla přiřazená stavovou zpětnou vazbou dány

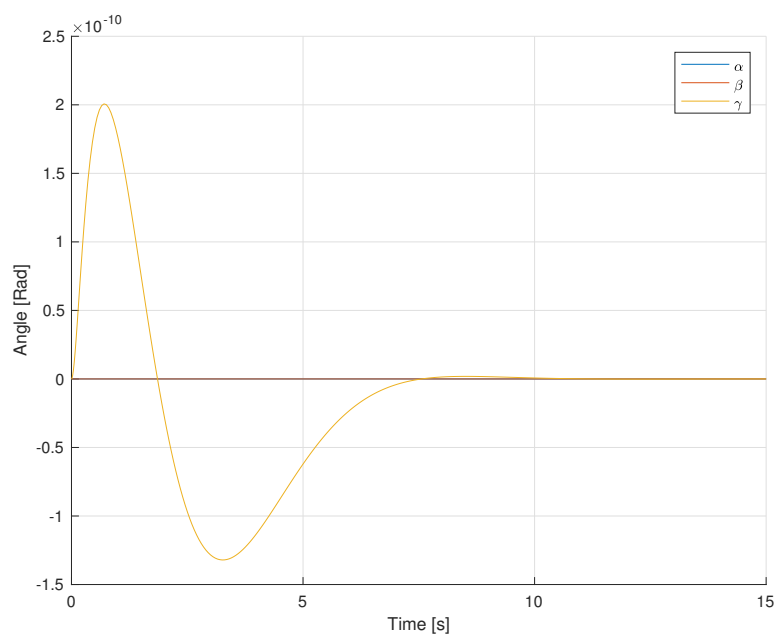
$$\begin{bmatrix} -7.3051 + 4.8741i \\ -7.3051 - 4.8741i \\ -7.0952 + 4.8317i \\ -7.0952 - 4.8317i \\ -4.3859 \\ -1.0271 \end{bmatrix} \begin{bmatrix} -1.3491 + 1.1671i \\ -1.3491 - 1.1671i \\ -1.3492 + 1.1672i \\ -1.3492 - 1.1672i \\ -0.7468 + 0.7333i \\ -0.7468 - 0.7333i \end{bmatrix}. \quad (3.85)$$

Nyní jsou vlastní čísla mnohem méně kmitavá než v případě kvadrokoptéry, ale to je opět dáno charakterem systému. Je to hlavně z toho důvodu, že akční členy jsou umístěny již v různých směrech a dovolují lepší manévrovací schopnosti. Co se týče porovnání vlastních čísel s využitím virtuálních vstupů a využitím klasických vstupů, dá se říci, že díky virtuálním vstupům se při využití stejných návrhových požadavků systém v některých stavech zpomalil.

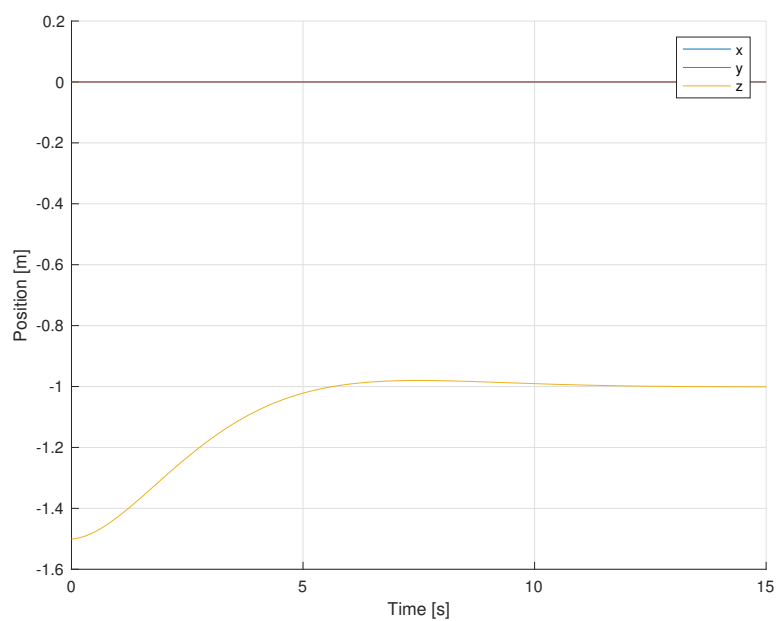
Pro demonstraci funkce vypočtených stavových zpětných regulátorů je i zde provedena simulace.



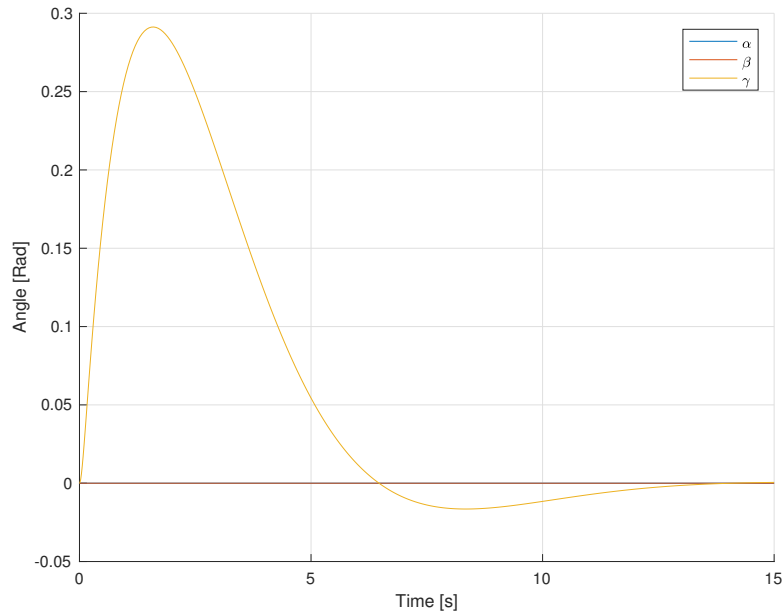
Obrázek 38: Pozice - z počáteční podmínky $x_{eq(1.polovina)} = [0 \ 0 \ -1.5 \ 0 \ 0 \ 0]$ pro \hat{F}



Obrázek 39: Úhly - z počáteční podmínky $x_{eq(1.polovina)} = [0 \ 0 \ -1.5 \ 0 \ 0 \ 0]$ pro \hat{F}



Obrázek 40: Pozice - z počáteční podmínky $x_{eq(1.polovina)} = [0 \ 0 \ -1.5 \ 0 \ 0 \ 0]$ pro F



Obrázek 41: Úhly - z počáteční podmínky $x_{eq(1.polovina)} = [0 \ 0 \ -1.5 \ 0 \ 0 \ 0]$ pro F

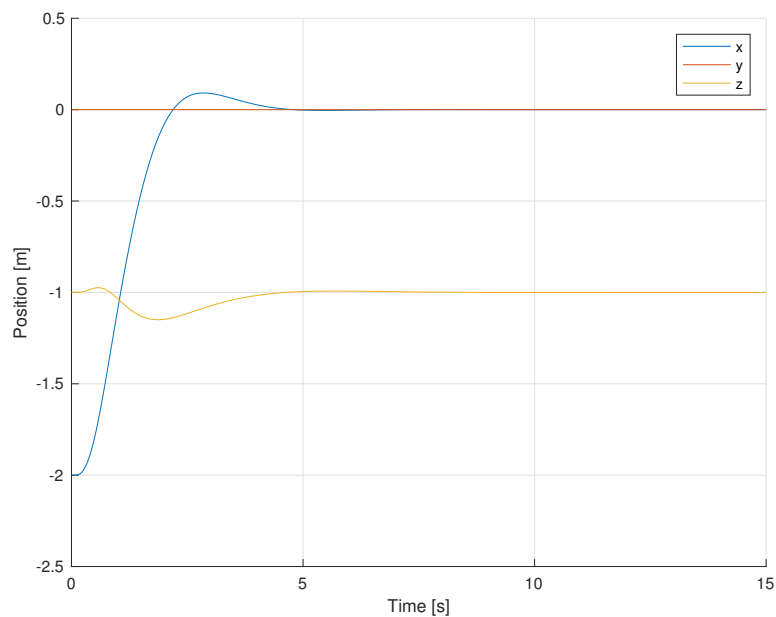
Jak lze vidět, tak při využití virtuálních vstupů je klesání omnikoptéry do rovnovážného stavu přímé a dron se nijak neotáčí. To se nedá říci o původním regulátoru F , který sice systém také stabilizuje, ale zbytečně u toho dochází i ke změně úhlu γ .

Nyní bude ověřeno, zda je symetrie zachována, pokud bude dron vychýlen z rovnovážného stavu ve směru osy \vec{x}_0 například o -2 m a následně bude v té samé ose vychýlen o 2 m, musí být řízení a stav v prvním a druhém případě svázán vztahem

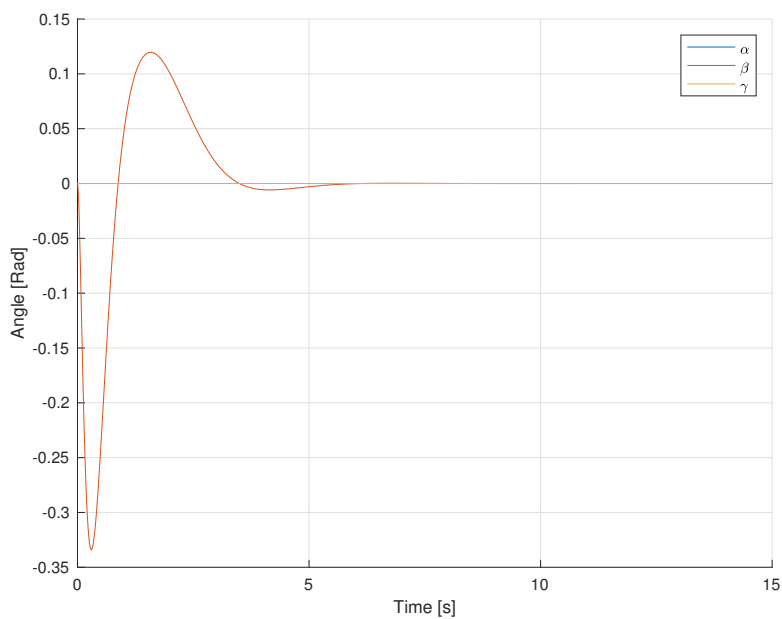
$$\vec{u}_I = G\vec{u}_{II}, \quad (3.86)$$

$$\vec{x}_I = T\vec{x}_{II}, \quad (3.87)$$

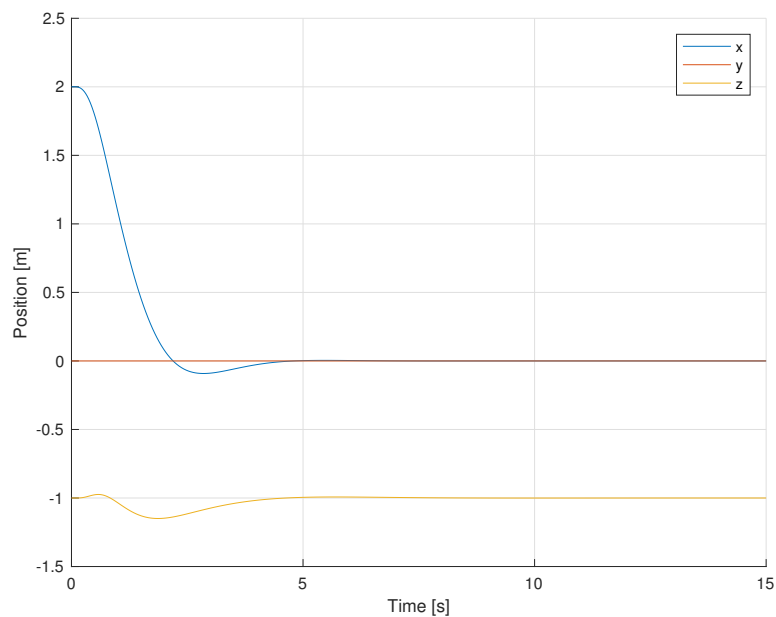
kde řízení \vec{u}_I a stav \vec{x}_I odpovídá vychýlení v ose \vec{x}_0 o -2 m a řízení \vec{u}_{II} a stav \vec{x}_{II} odpovídá zase vychýlení v ose \vec{x}_0 o 2 m. Tyto vztahy jsou platné pro původní regulátor F bez virtuálních vstupů.



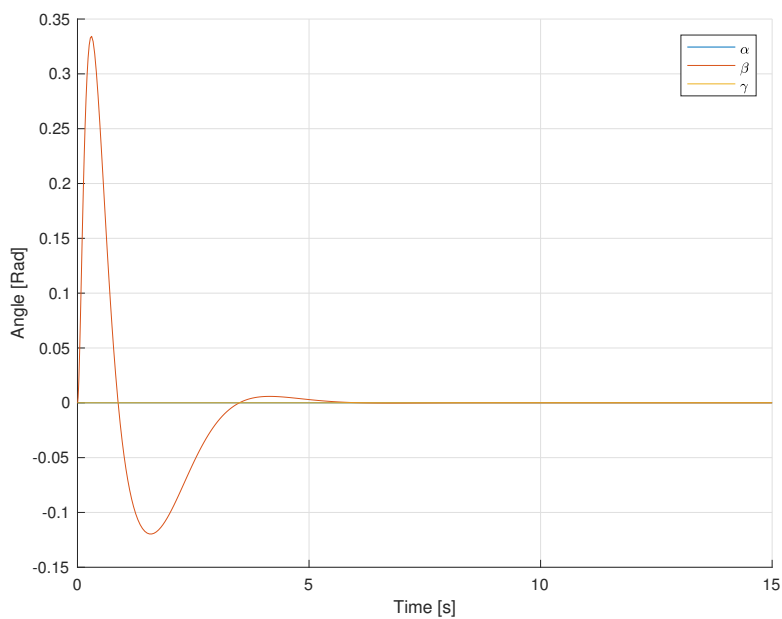
Obrázek 42: Pozice - z počáteční podmínky $x_{eq} (1.polovina) = [-2 \ 0 \ -1 \ 0 \ 0 \ 0]$



Obrázek 43: Úhly - z počáteční podmínky $x_{eq} (1.polovina) = [-2 \ 0 \ -1 \ 0 \ 0 \ 0]$



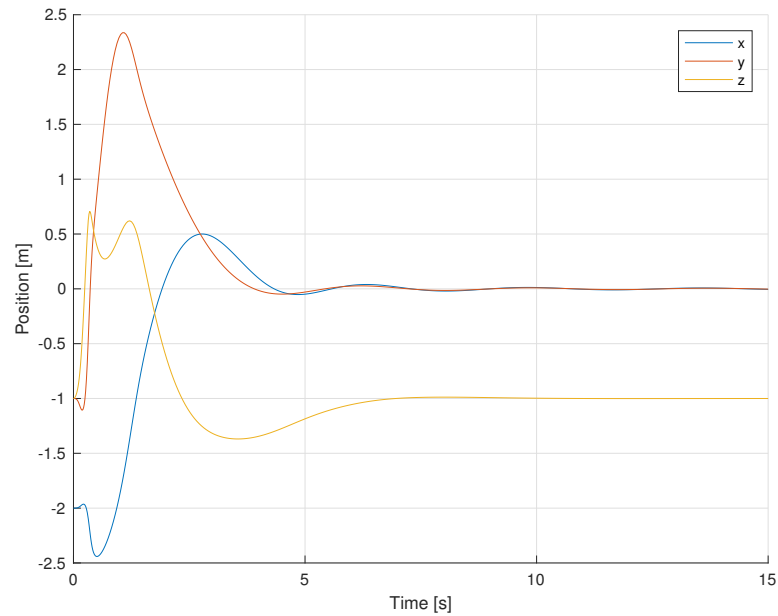
Obrázek 44: Pozice - z počáteční podmínky $x_{eq} (1.polovina) = [2 \ 0 \ -1 \ 0 \ 0 \ 0]$



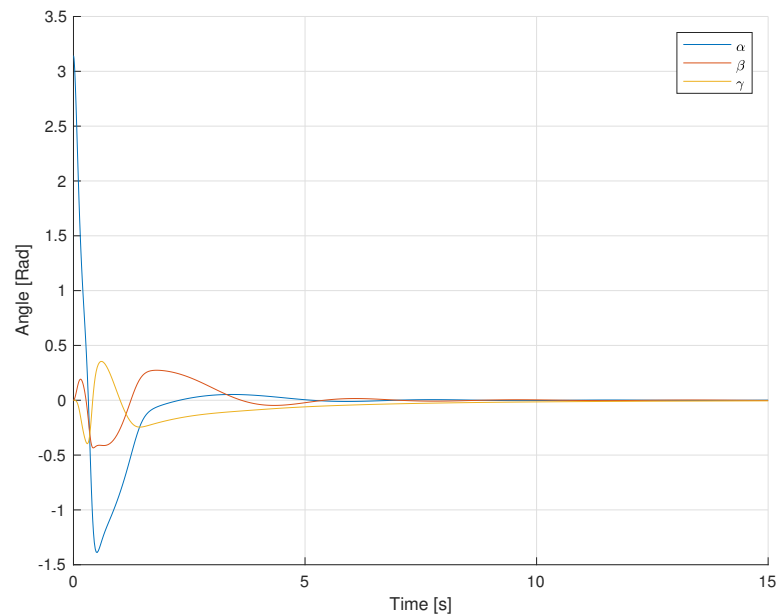
Obrázek 45: Úhly - z počáteční podmínky $x_{eq} (1.polovina) = [2 \ 0 \ -1 \ 0 \ 0 \ 0]$

Jak si lze všimnout, tak díky dodržení symetrie se systém chová také naprosto shodně, pouze reakce v úhlech je zde přímo inverzní, což je ale dáno vztahy (3.86) a (3.87).

Jako v případě kvadrokopty bude regulátor \hat{F} otestován otočením omniskopťery směrem dolů a posunut v osách \vec{x}_0 a \vec{y}_0



Obrázek 46: Pozice - z počáteční podmínky $x_{eq(1.polovina)} = [-2 \quad -1 \quad -1 \quad \pi \quad 0 \quad 0]$



Obrázek 47: Úhly - z počáteční podmínky $x_{eq(1.polovina)} = [-2 \quad -1 \quad -1 \quad \pi \quad 0 \quad 0]$

I v tomto případě regulátor \hat{F} funguje správně, a to i za té skutečnosti, že je navržen pouze

pro rovnovážný bod (2.89). I v tomto případě se projevuje fakt, že omnikoptéra má mnohem lépe uzpůsobené vrtule a tak není propad v případě ose \vec{z}_0 tak markantní. Vzhledem k symetrii systému by v případě otočení o π podle osy α neměl nastat žádný propad a systém by měl nehnutě stát. Ovšem je třeba mít na paměti, že je zde testován stále jen lineární stavový regulátor, který má v sobě stále zahrnut pouze rovnovážný stav, který je z jeho pohledu jiný než v případě otočení.

3.6 Kvadrokoptéra - polohová regulace

Polohová regulace je zde vyřešena rozšířením stavových matic, kde jsou přidány integrátoři pro stavy, které je potřeba regulovat do požadovaných poloh a úhlů. Pro kvadrokoptéru není možné využít polohovou regulaci pro celý stav, jelikož zde není možný přímý pohyb do všech směrů jako v případě omnikoptéry. Pokud je například požadována polohová regulace ve prospěch osy \vec{x}_0 není pak možná regulace úhlu β . Z toho důvodu je regulace omezena na stavy $[x \ y \ z \ \gamma]$. Příslušné upravené stavové matice jsou

$$A_p = \begin{bmatrix} A & 0_{12 \times 4} \\ I_P & 0_{4 \times 4} \end{bmatrix}, \quad (3.88)$$

$$I_P = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad (3.89)$$

$$B_p = \begin{bmatrix} B \\ 0_{4 \times 4} \end{bmatrix}. \quad (3.90)$$

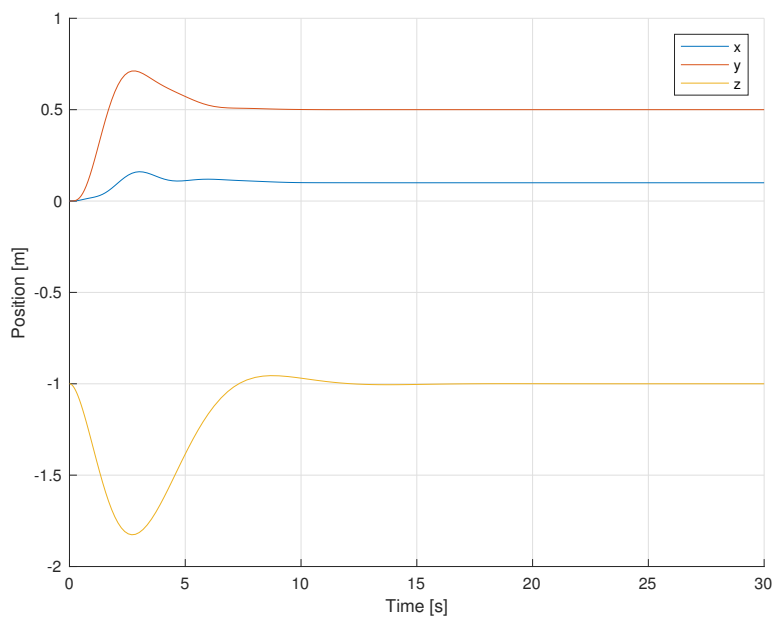
Při simulacích je nutné mít na paměti, že ne všechny požadavky mohou být splněny, jelikož se zde stále jedná o lineární regulátor, který je platný pouze v rovnovážném bodě (2.89) a v jeho velice malém okolí.

Matice Q_p a R_p jsou v tomto případě zvoleny ve tvaru

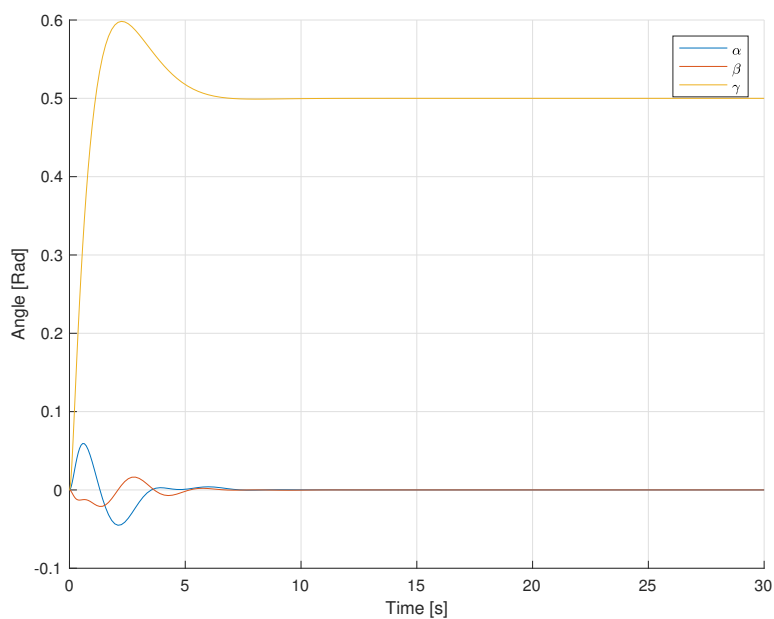
$$Q_p = \begin{bmatrix} Q_s(3.64) & 0 \\ 0 & I_{4 \times 4} \end{bmatrix}, \quad (3.91)$$

$$R_p = R_s(3.65). \quad (3.92)$$

S těmito zvolenými maticemi jsou díky vypočtenému regulátoru F_p přiřazeny systému v uzavřené



Obrázek 49: Pozice - $x_{ref} (1.polovina) = [0.1 \ 0.5 \ -1 \ 0 \ 0 \ 0.5]$



Obrázek 50: Úhly - $x_{ref} (1.polovina) = [0.1 \ 0.5 \ -1 \ 0 \ 0 \ 0.5]$

Jak lze pozorovat, tak v případě polohové regulace je splněn celý zadaný referenční stav. Co se týče omezení, které zde nastávají, tak není problém v případě libovolného požadavku pro polohu, pokud se při pohybu nenatočí kvadroptéra o moc velký úhel ve stavu α a β

(ve většině případů k nestabilitě vede cca. 0.5 Rad). Tedy pokud je zvolena v případě polohy větší vzdálenost, bude vhodné regulátor navrhnout pomalejší, aby se kvadroptéra nenatočila příliš a za delší čas by se postupně přesunula do požadovaného stavu.

3.7 Omnikoptéra - polohová regulace

V případě omnikoptéry je situace mnohem příznivější. Vzhledem k možnému pohybu do všech stran zde není v případě polohové regulace žádné omezení. Matice systému budou pro všechny základní stavy $(x, y, z, \alpha, \beta, \gamma)$ rozšířeny o integrátor. Stavové matice nově vzniklého systému jsou

$$A_p = \begin{bmatrix} A & 0_{12 \times 6} \\ I_P & 0_{6 \times 6} \end{bmatrix}, \quad (3.94)$$

$$I_P = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad (3.95)$$

$$B_p = \begin{bmatrix} B \\ 0_{6 \times 8} \end{bmatrix} M. \quad (3.96)$$

Za povšimnutí stojí fakt, že v případě polohové regulace jsou již využity pouze virtuální vstupy systému, které byly definovány ve stabilizaci.

Matice Q_p a R_p jsou zvoleny v tomto případě ve tvaru

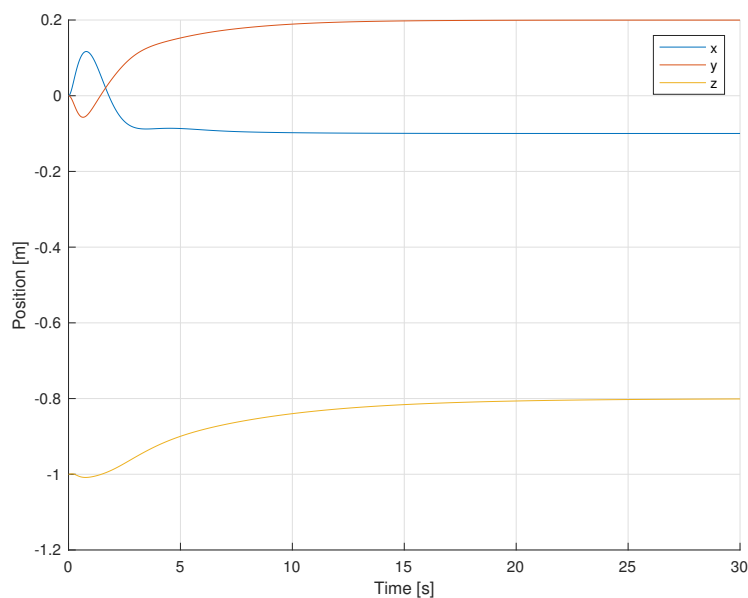
$$Q_p = \begin{bmatrix} Q_s(3.81) & 0 \\ 0 & I_{6 \times 6} \end{bmatrix}, \quad (3.97)$$

$$R_p = \hat{R}(3.83). \quad (3.98)$$

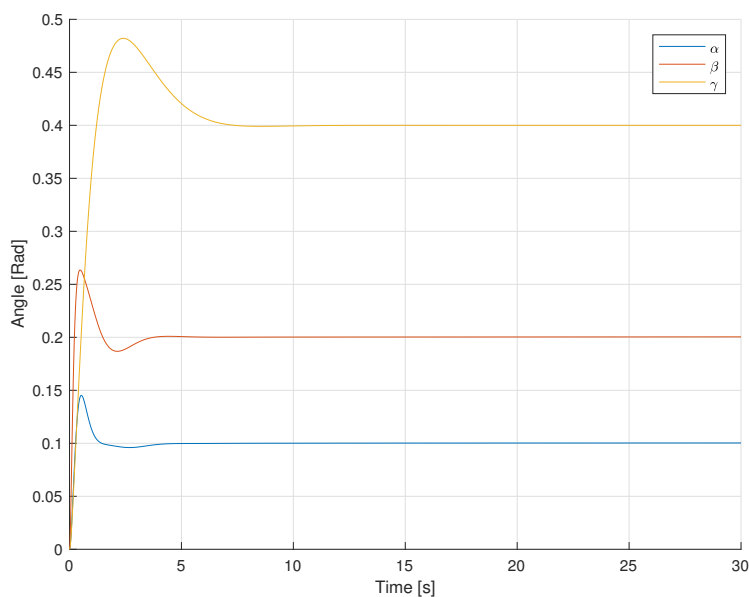
S těmito zvolenými maticemi jsou díky vypočtenému regulátoru F_p přiřazena v uzavřené smyčce tyto vlastní čísla

$$\begin{bmatrix} -7.3048 + 4.8741i \\ -7.3048 - 4.8741i \\ -7.0949 + 4.8318i \\ -7.0949 - 4.8318i \\ -4.3924 \\ -1.3380 + 1.1738i \\ -1.3380 - 1.1738i \\ -1.3380 + 1.1738i \\ -1.3380 - 1.1738i \end{bmatrix} \begin{bmatrix} -0.7412 + 0.7389i \\ -0.7412 - 0.7389i \\ -0.8730 + 0.5133i \\ -0.8730 - 0.5133i \\ -0.3177 \\ -0.3177 \\ -0.1826 \\ -0.02 \\ -0.02 \end{bmatrix}. \quad (3.99)$$

I v případě rozšířeného systému vyšly vlastní čísla systému podobně jako v případě stabilizace. Nyní bude provedeno ověření simulací. V případě polohové regulace je nutné mít na paměti, že vypočtený regulátor bude fungovat pouze v rovnovážném stavu (2.89) a v jeho malém okolí. Z těchto důvodů bude referenční signál volen v blízkosti rovnovážného stavu.



Obrázek 51: Pozice - $x_{ref} (1.polovina) = [-0.1 \ 0.2 \ -0.8 \ 0.1 \ 0.2 \ 0.4]$



Obrázek 52: Úhly - $x_{ref} (1.polovina) = [-0.1 \ 0.2 \ -0.8 \ 0.1 \ 0.2 \ 0.4]$

Požadavky byly ve všech případech splněny. Pokud by bylo požadováno jiné chování jako čas náběhu nebo velikost překmitu, je nutné změnit penalizační matice (3.97) a (3.98). I v tomto případě je rozumné volit požadovaný úhel pro α a β do 0.5 Rad . Což je opět dáno charakteristikou lineárního regulátoru.

3.8 Kvadrokoptéra - omezený stav, gamepad

Vzhledem ke skutečnosti, že v reálném světě nejsme schopni znát celý stav, je vhodné zahrnout do této práce i praktičtější aplikaci. Je vycházeno z předpokladu, že je znám pouze stav

$$\vec{x}_r = [z \ \alpha \ \beta \ \dot{z} \ \dot{\alpha} \ \dot{\beta} \ \dot{\gamma}]^T. \quad (3.100)$$

Za pomoci takto redukováného stavu je řízena celá řada nejrůznějších dronů. Vzhled k využití stavového zpětného regulátoru je nutné vynulovat sloupce regulátoru F , které odpovídají neměřitelným stavům, vhodnou volbou matic Q a R . Pokud by regulátor F obsahoval nenulové prvky příslušné neměřitelným stavům, nebylo by možné ho využít, jelikož by hrozilo nestabilní chování uzavřené smyčky. Je tedy nutné navrhnout regulátor F tak, aby pro stabilizaci nebyla potřeba informace od neměřitelných stavů. V tomto případě ale regulátor F přiřadí jen 7 vlastních čísel v uzavřené smyčce a ostatní vlastní čísla matice dynamiky uzavřeného systému zůstanou nezměněná.

Penalizující matice stavu Q_r a penalizující matice řízení R_r jsou v tomto případě zvoleny ve tvaru

$$Q_r = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 30 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 30 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad (3.101)$$

$$R_r = R_s(3.65), \quad (3.102)$$

Takto zvolené matice Q_r a R_r neberou v úvahu neměřitelné stavy a proto ani žádným způsobem

neovlivní kritériální funkci J_c (3.41) metody LQR. Výsledný stabilizující regulátor je ve tvaru

$$F_r = \begin{bmatrix} 0 & 0 & -1.5811 & 0 & 3.873 & 0 & 0 & 0 & -2.0501 & 0 & 2.3444 & -0.5 \\ 0 & 0 & -1.5811 & 3.873 & 0 & 0 & 0 & 0 & -2.0501 & 2.3934 & 0 & 0.5 \\ 0 & 0 & -1.5811 & 0 & -3.873 & 0 & 0 & 0 & -2.0501 & 0 & -2.3444 & -0.5 \\ 0 & 0 & -1.5811 & -3.873 & 0 & 0 & 0 & 0 & -2.0501 & -2.3934 & 0 & 0.5 \end{bmatrix}. \quad (3.103)$$

Tento stavový zpětný regulátor F_r ovlivňuje pouze omezený stav (3.100) a přiřazuje v uzavřené smyčce tyto vlastní čísla

$$\begin{bmatrix} -1.8174 + 1.6437i \\ -1.8174 - 1.6437i \\ -9.009 \\ -0.82 + 0.7697i \\ -0.82 - 0.7697i \\ -1.7729 + 1.6108i \\ -1.7729 - 1.6108i \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad (3.104)$$

proto vlastní čísla, které odpovídají neměřitelným stavům, zůstávají nulová.

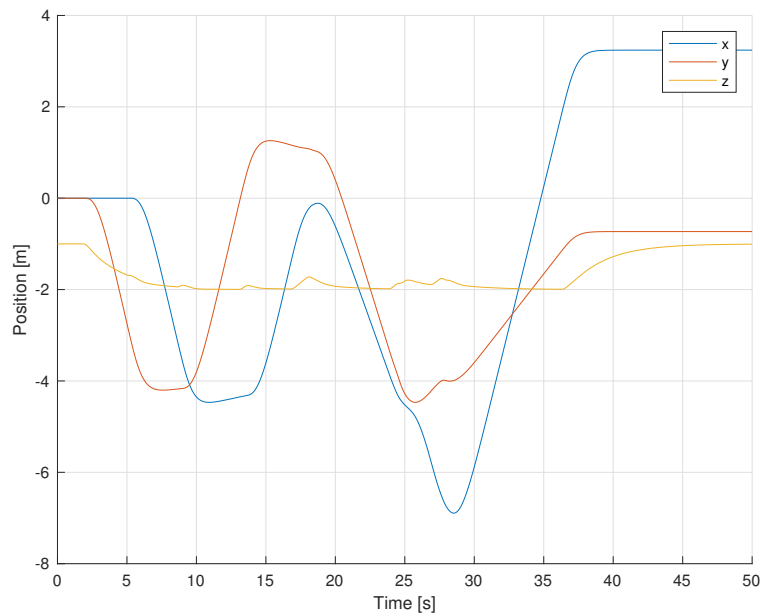
Tento regulátor (3.103) ovšem pouze stabilizuje kvadrokoptéru v daném redukovaném stavu (3.100). Řízení pohybu je zde zvoleno jako působení skokové chyby na řídicí signál. Díky tomu, že vypočtený regulátor (3.103) nijak neřídí stavovou proměnou x , y nebo γ , lze s dronem létat právě jen za pomoci působení skokových chyb, kde ke klasickému řídicímu signálu z regulátoru je připočten další řídicí signál, který se mění na základě toho, jak se má kvadrokoptéra pohybovat ve směru os \vec{x}_1 , \vec{y}_1 nebo se otáčet dle osy \vec{z}_1 . Otáčení dle osy \vec{z}_1 je provedeno porušením rovnováhy rotorů, kde stačí dvojici rotorů zpomalit nebo naopak zrychlit a kvadrokoptéra se začne otáčet příslušným směrem. Podobně je to provedeno pro pohyb v osách \vec{x}_1 a \vec{y}_1 , kde stačí ze dvojice rotorů jeden zrychlit a kvadrokoptéra se začne natáčet a tím i pohybovat příslušným směrem. Pohyb v osách je pak ale možný pouze z pohledu pohyblivé soustavy pevně spojené s tělesem.

Jelikož není uvažována stabilizace pro stavové proměnné x , y a ani jejich derivace, je nutné do simulace zahrnout i sílu, která působí proti pohybu kvadrokoptéry, jinak by se UAV pohybovalo neustále konstantní rychlostí, jelikož v pohybové rovnici (2.155) tato síla není uvažována. Z tohoto důvodu je přidána síla, která působí proti pohybu UAV, ta je pak ve tvaru

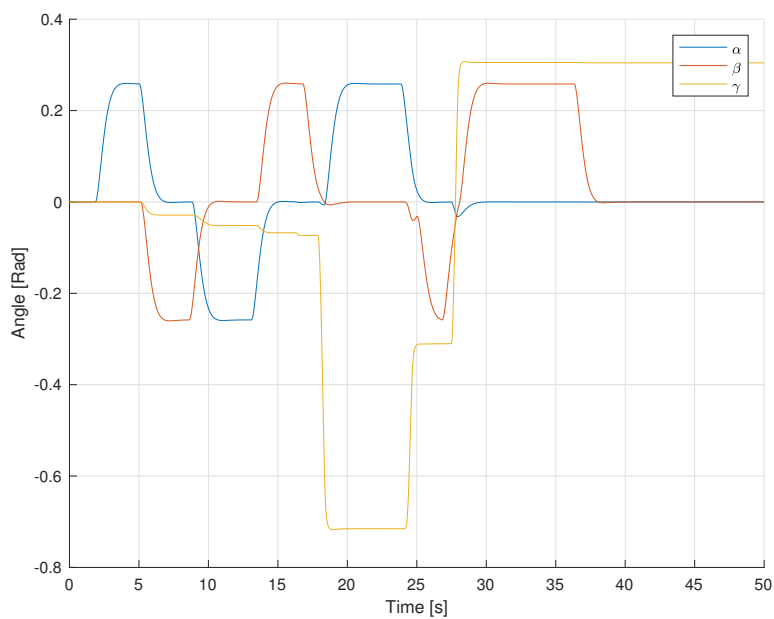
$$F_d = -bv, \quad (3.105)$$

kde b je koeficient tření a v je rychlost. Síla F_d tedy simuluje odpor vzduchu. Pro simulaci je zvoleno $b = 10$.

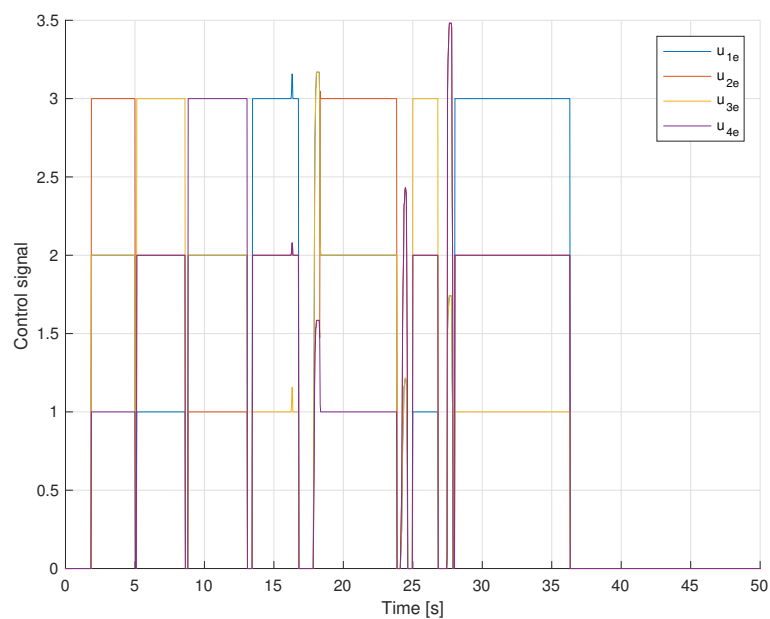
Výše popsané řízení je připojeno na externí ovládací prvek - gamepad. Zapojení je popsáno v kapitole 4.5. Řízení je doplněno o 3D simulaci, která je popsána v kapitole 4.6. Nyní bude přistoupeno k jedné simulaci, kde bude ověřena funkčnost řídicího algoritmu.



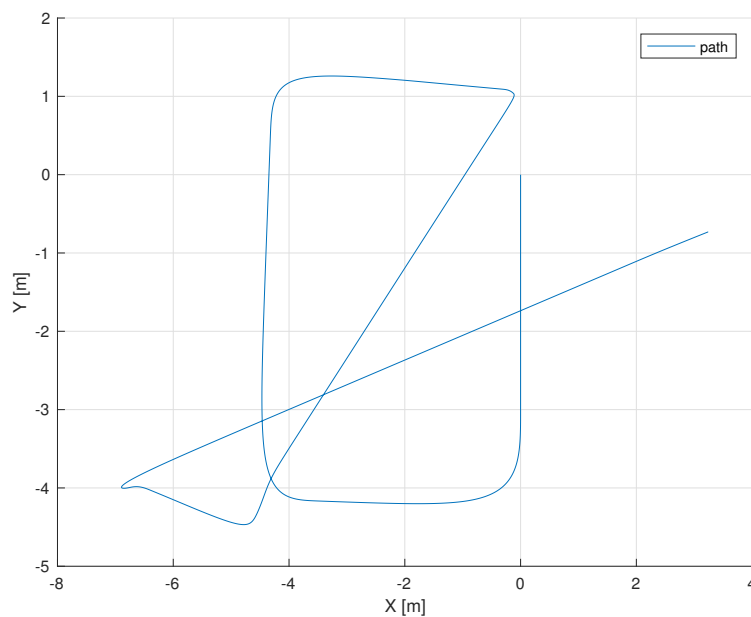
Obrázek 53: Pozice - let ovládaný gamepadem



Obrázek 54: Úhly - let ovládaný gamepadem



Obrázek 55: Působení skokových poruch do řídicího signálu - let ovládaný gamepadem

Obrázek 56: x/y - let ovládaný gamepadem

Celková doba trvání byla 50 vteřin. Jak si lze všimnout z grafů, tak při působení skokových chyb se kvadrokoptéra vždy ve stavových proměnných α a β správně vrátila do počátečního stavu. Jelikož u stavové proměnné z byla dána nižší penalizace, tak návrat do ustáleného stavu

potřeboval více času, na druhou stranu výšková stabilizace není tak kritická na rychlost jako je natočení v α a β .

3.9 Omniskoptéra - omezený stav, gamepad

To, co bylo vytvořeno pro kvadrokoptéru, bude nyní aplikováno i na omniskoptéru. Je znám tedy pouze stav

$$\vec{x}_r = [z \ \alpha \ \beta \ \dot{z} \ \dot{\alpha} \ \dot{\beta} \ \dot{\gamma}]^T. \quad (3.106)$$

Penalizující matice stavu Q_r a penalizující matice řízení R_r jsou v tomto případě zvoleny také ve tvaru

$$Q_r = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 30 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 30 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad (3.107)$$

$$R_r = \hat{R}(3.83). \quad (3.108)$$

Díky této volbě neměřitelné stavy žádným způsobem neovlivňují navržený stabilizující regulátor, ten je ve tvaru

$$F_r = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -3.1623 & 0 & 0 & 0 & 0 & -5.7116 & 0 & 0 & 0 \\ 0 & 0 & 0 & 5.4772 & 0 & 0 & 0 & 0 & 1.5744 & 0 & 0 \\ 0 & 0 & 0 & 0 & 5.4772 & 0 & 0 & 0 & 0 & 1.5534 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad (3.109)$$

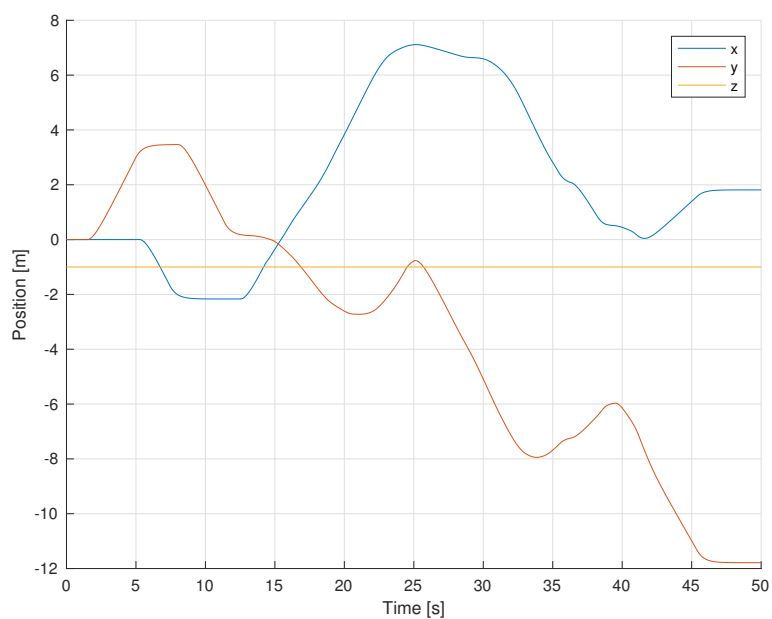
kde přiřazuje matici dynamiky uzavřené smyčky tyto vlastní čísla

$$\begin{bmatrix} -6.0210 + 2.4913i \\ -6.0210 - 2.4913i \\ -5.8312 + 2.5629i \\ -5.8312 - 2.5629i \\ -4.5045 \\ -0.5712 + 0.5534i \\ -0.5712 - 0.5534i \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad (3.110)$$

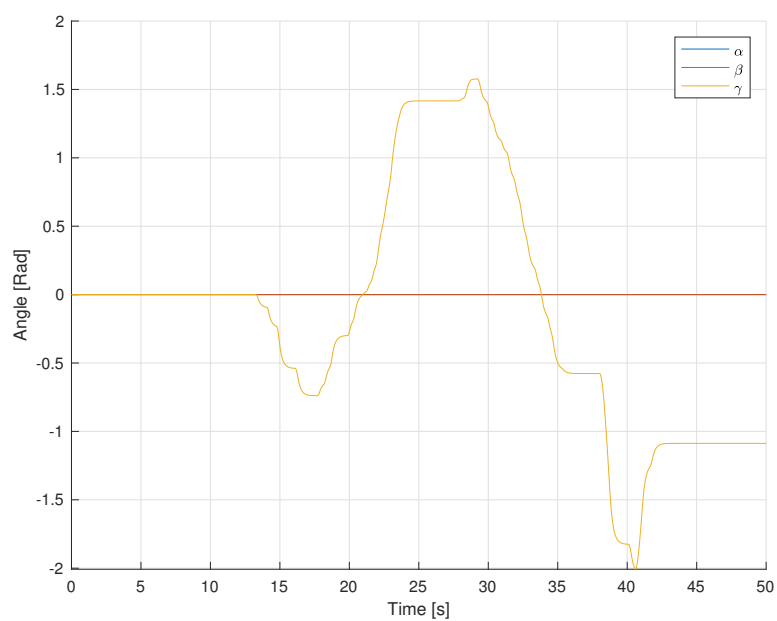
vlastní čísla u neměřitelných stavů nejsou nijak regulátorem ovlivněny a tudíž zůstávají nulová.

Podobně jako u kvadroptéry je řízení pohybu a otáčení vyřešeno pomocí působení skokových poruch na řídicí signál. K tomuto účelu je opět využita matice M (3.78), díky které lze působit chyby pro řídicí signál tak, aby se omnikoptéra pohybovala dle požadavků operátora. Je vhodné poznamenat, že tento způsob řízení je z pohledu pohyblivé soustavy souřadnic pevně spojené s tělesem.

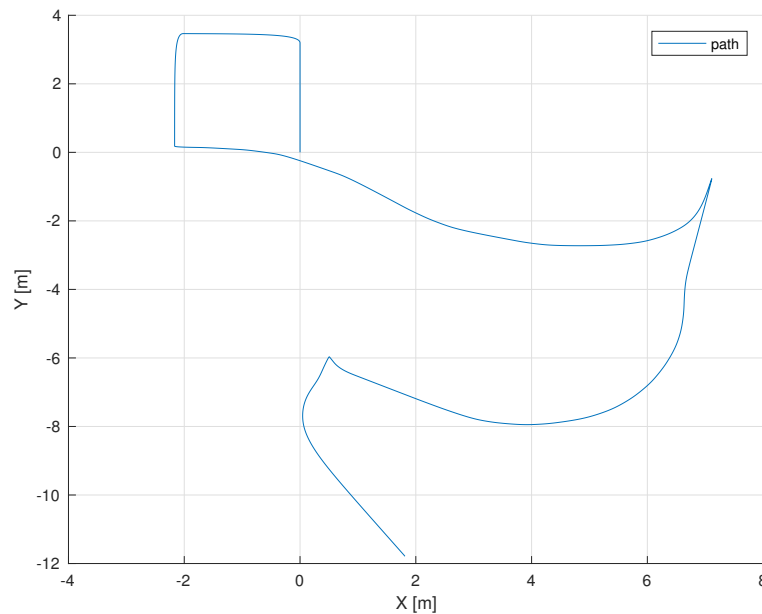
Výše popsané řízení je připojeno na externí ovládací prvek - gamepad. Zapojení je popsáno v kapitole 4.5. Řízení je doplněno o 3D simulaci, která je popsána v kapitole 4.6. I v tomto případě bude provedena krátká simulace, aby byl navržený řídicí algoritmus ověřen. Zde je též využit simulovaný odpor vzduchu - síla (3.105), která postupně tlumí rychlost UAV.



Obrázek 57: Pozice - let ovládaný gamepadem



Obrázek 58: Úhly - let ovládaný gamepadem

Obrázek 59: x/y - let ovládaný gamepadem

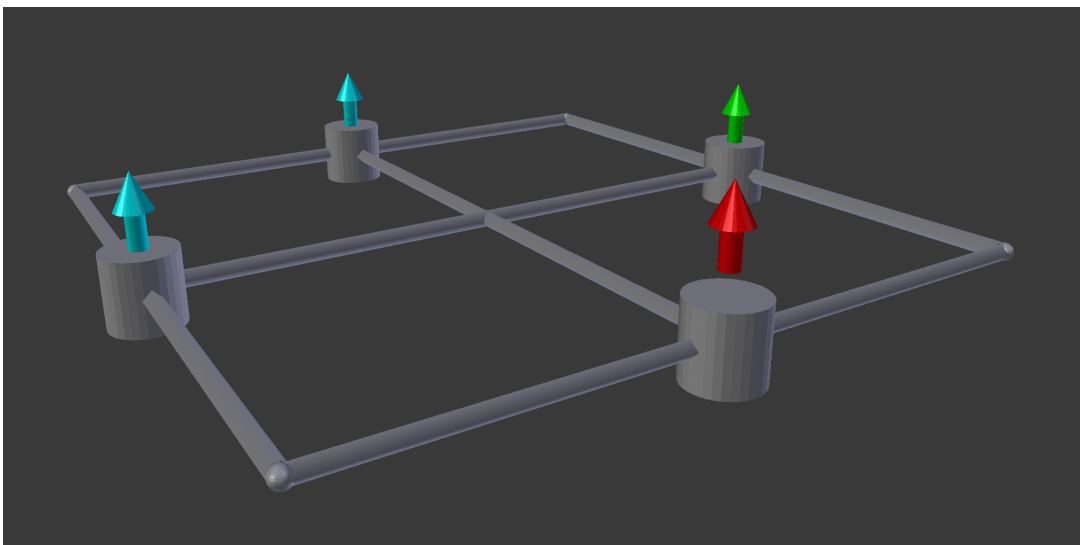
Na rozdíl od kvadrokoptéry si lze všimnout jednoho podstatného rozdílu. Při přímočarém pohybu se žádným způsobem nemění žádný z úhlů. Jediný úhel, který se mění, je úhel γ , ten se mění pouze při požadavku od operátora. I výška omnikoptéry zůstává po celou dobu konstantní. Tedy i u takto jednoduchého řídicího algoritmu lze pozorovat značné rozdíly a výhody oproti klasické kvadrokoptěře.

Ani možnost ovládní výšky v ose \vec{z} není žádným problémem. Ovšem pro snadnější ovládní je pro stavovou proměnou z zavedeno také stabilizující řízení.

4 3D virtuální model

Pro lepší vizualizaci a kontrolu výsledků simulací byl vytvořen virtuální 3D model. Tento model byl vytvořen s ohledem na přenositelnost modelů mezi jednotlivými programy. V práci je primárně využíván *Matlab/Simulink*, který nabízí vlastní 3D vizualizaci, ale je vhodné model udělat pro další použití i pro jiné programy. Proto byl využit program *Blender*, který je bezplatný a nabízí nejrůznější možnosti v oblasti modelování a animace ve 3D. Také je možné model exportovat do nejrůznější škály formátů.

4.1 Kvadrokoptéra - 3D model

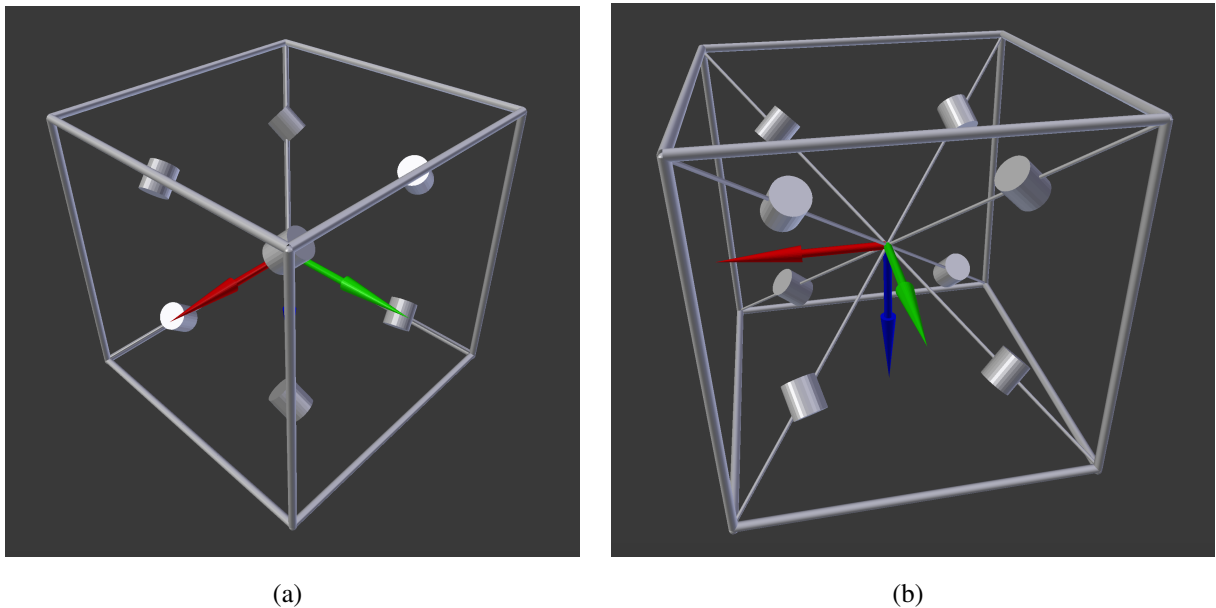


Obrázek 60: Virtuální model - kvadrokoptéra - *Blender*

Model vytvořený v programu *Blender* obsahuje mimo jiné i čtyři šipky, které reprezentují směr a působení síly generované rotory. Červeně a zeleně označená šipka pak určují orientaci dronu. Pozice těchto šipek je následně v simulaci měněna dle velikosti řídicího signálu a i dle otočení pohyblivé soustavy souřadnic.

Model tedy obsahuje 5 samostatných objektů (4 šipky + konstrukce), které je nutné transformovat. Rotace a translace bude u všech probíhat stejným způsobem, ovšem translace ve prospěch osy z_1 u šipek bude ještě přímo závislá na dodávaném řízení, aby bylo možné pozorovat, jak velké rozdíly jsou mezi jednotlivými řízeními. To je vhodné zejména v prvotní fázi návrhu, kdy je nutné ověřit chování odvozeného matematického modelu.

4.2 Omnikoptéra - 3D model



Obrázek 61: Virtuální model - omnikoptéra - *Blender*

U omnikoptéry bylo vzhledem k plynulosti simulace upuštěno od jednotlivých šipek pro každý rotor. Pro lepší orientaci při simulaci se využívá klasických tří šipek, které reprezentují osy $\vec{x}_1, \vec{y}_1, \vec{z}_1$, které jsou v pořadí "Red Green Blue" (RGB).

Zvláště v případě omnikoptéry je 3D model velice užitečný, jelikož díky množství rotorů a jejich různému umístění je vhodné lidskou představivost podpořit 3D simulací.

4.3 VR Sink blok

VR Sink je blok v *Simulinku*, který nabízí možnost jednoduchých 3D animací. Tento blok je schopen pracovat například s formátem *.x3d. Blok má i vlastní zjednodušený 3D editor, který ale spíše slouží na finální úpravy již hotového modelu. Po nahrání modelu do *VR Sink* je nutné zvolit, jaké transformace budou s modelem prováděny. V obou případech (kvadroptéra a omnikoptéra) postačí, když bude zvolena rotace a translace.

Nutné je dát pozor na rozdílnou soustavu souřadnic a rotaci, kterou blok využívá. Klasická pravoúhlá soustava souřadnic a soustava souřadnic bloku *VR Sink* jsou ve vztahu

$$\vec{x} = \vec{x}_{VR}, \quad (4.1)$$

$$\vec{y} = -\vec{z}_{VR}, \quad (4.2)$$

$$\vec{z} = \vec{y}_{VR}. \quad (4.3)$$

Také *VR Sink* nevyužívá rotaci dle Eulerových úhlů ani přímo kvaternionů, ale rotaci popsanou

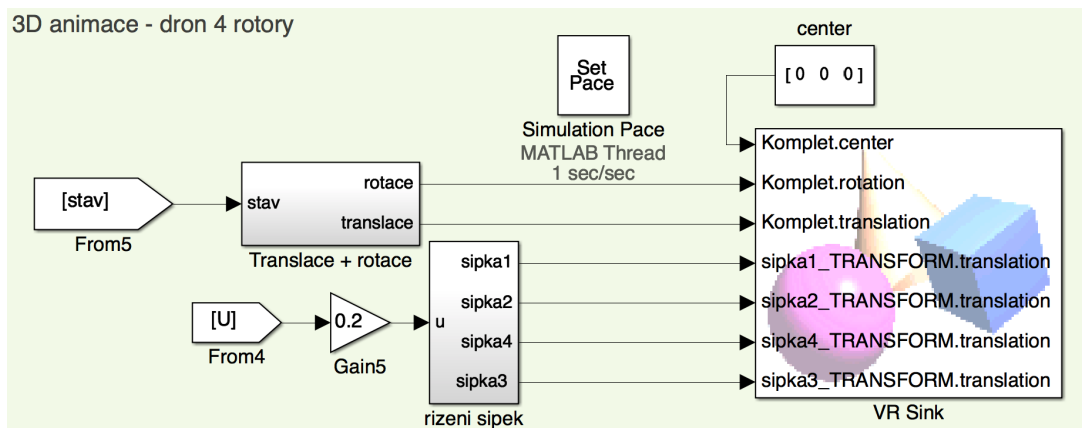
čtyřmi jinými parametry. Jedná se o rotaci definovanou jako

$$VR_{rotace} = \underbrace{[n_x \ n_y \ n_z]}_{\vec{n}}, \quad (4.4)$$

kde složky vektoru \vec{n} reprezentují vektor, dle kterého je rotováno, a parametr θ udává úhel o který je rotováno. Pro tento případ je v *Simulinku* implementovaný blok - *Rotation Matrix to VRML Rotation*, který převádí matici rotace na vztah (4.4).

4.4 Simulink - zapojení

Nyní bude pozornost přesunuta na zapojení bloku *VR Sink*. Blok má nyní nahraný 3D model UAV ve formátu *.x3d a jsou označeny požadované vstupy, tedy požadavky na transformaci drona.

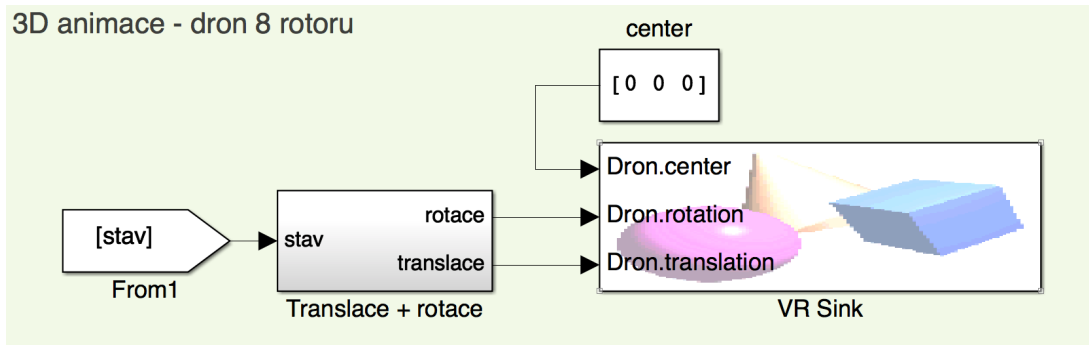


Obrázek 62: *Simulink VR Sink* - zapojení - kvadroptéra

Zapojení kvadroptéry je díky čtyřem šipkám, které ukazují poměr mezi jednotlivým řízením, komplikovanější. Blok *řízení šipek* pak obsahuje polohové údaje o šipkách a dle přicházejícího řízení se mění souřadnice ve směru osy \vec{z}_1 . Stav systému, který přichází do bloku *Translace + rotace*, je patřičně rozdělen a přetransformován pro použití s blokem *VR Sink*. Posledním důležitým blokem je *Simulation Pace*, který simuluje reálný čas a zpomaluje simulaci, díky čemuž je možné měnit referenční požadavky na stav, nebo působit poruchy na systém již během simulace.

Podobné zapojení lze nalézt i na omnioptéře. Zde ale nejsou pro plynulost simulace uvažovány šipky nad rotory. Blok *Simulation Pace* se již nachází ve zbytku simulačního schématu, proto zde na schématu není.

Zapojení pomocí bloku *VR Sink* nabízí vcelku jednoduchou možnost, jak provést základní 3D animace a tím lépe vizuálně ověřit výsledky simulací. Dále je zde možnost pořídít videozáznam animace, nebo uložit momentální snímek, což může pomoci při následné analýze výsledků.



Obrázek 63: Simulink VR Sink - zapojení - omniskopára

4.5 Simulink - gamepad

Pro manuální ovládání v reálném čase je k *Simulinku* připojen gamepad *XBOX 360 Controller*. Zapojení je vcelku jednoduché, jelikož je možné využít již implementovaný blok v *Simulinku* - *Joystick Input*. Ten má celkem dva hlavní výstupy, v prvním jsou obsaženy všechny výstupy z jednotlivých joysticků a ve druhém jsou informace o stisknutí zbylých tlačítek. Komunikace je pak zabezpečena přes USB port. Pokud je k zařízení připojeno více periférií, je možné, že některé budou brány také jako ovládací prvek - joystick, tedy je nutné v bloku zvolit správně ID připojeného ovládacího prvku.



Obrázek 64: XBOX 360 gamepad - [driver for macOS]

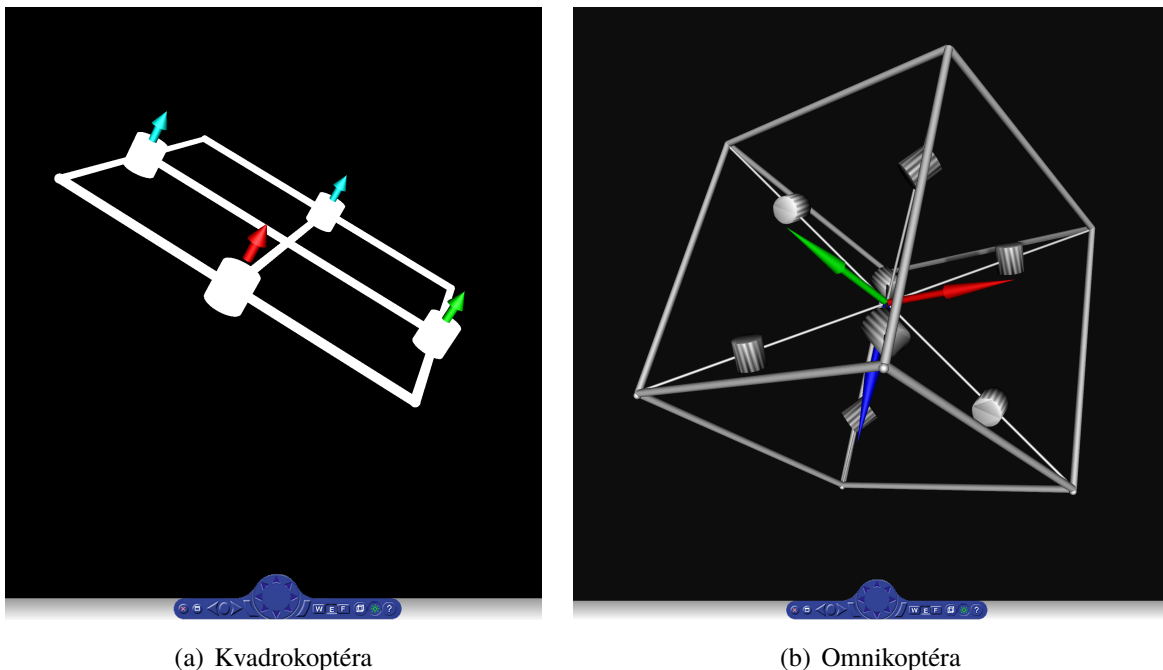
Z gamepadu jsou pak využity pouze tlačítka (LT), (RT), (B), (A), (Y) a (X). Jejich funkce jsou:

- (LT) a (RT) - rotace dle osy \vec{z}_1 ,
- (B) a (X) - pohyb ve směru osy $\vec{x}_1 / -\vec{x}_1$,
- (A) a (Y) - pohyb ve směru osy $\vec{y}_1 / -\vec{y}_1$.

Tuto konfiguraci lze samozřejmě velice jednoduše změnit, ovšem pro tuto diplomovou práci je řízení a simulace omezena na základní požadavky létání, proto není nutné využívat všechny ovládací prvky. Díky gamepadu lze simulovat reálnou odezvu UAV a vyzkoušet různé manévry nebo řídicí systémy, které by v případě reálného testování vedly ke zničení celého drona nebo alespoň řady vrtulí.

4.6 Simulink - 3D animace

Jak již bylo výše zmíněno, celá 3D simulace byla provedena v *Simulinku* díky bloku *VR Sink*. Zde jsou jen pro ilustraci uvedeny oba dva 3D modely UAV v tomto programovém prostředí.



Obrázek 65: *VR Sink* - vizualizace

Blok nabízí i možnost exportu simulace do *.avi. Pro tyto případy lze i přidávat různá pozadí nebo osvětlení pro reálnější imitaci reality. Vzhledem k té skutečnosti, že tento blok není nejlépe optimalizovaný, bylo od těchto věcí upuštěno, aby byl zaručen co možná nejplynulejší běh simulace. Videá jsou součástí práce na přiloženém CD.

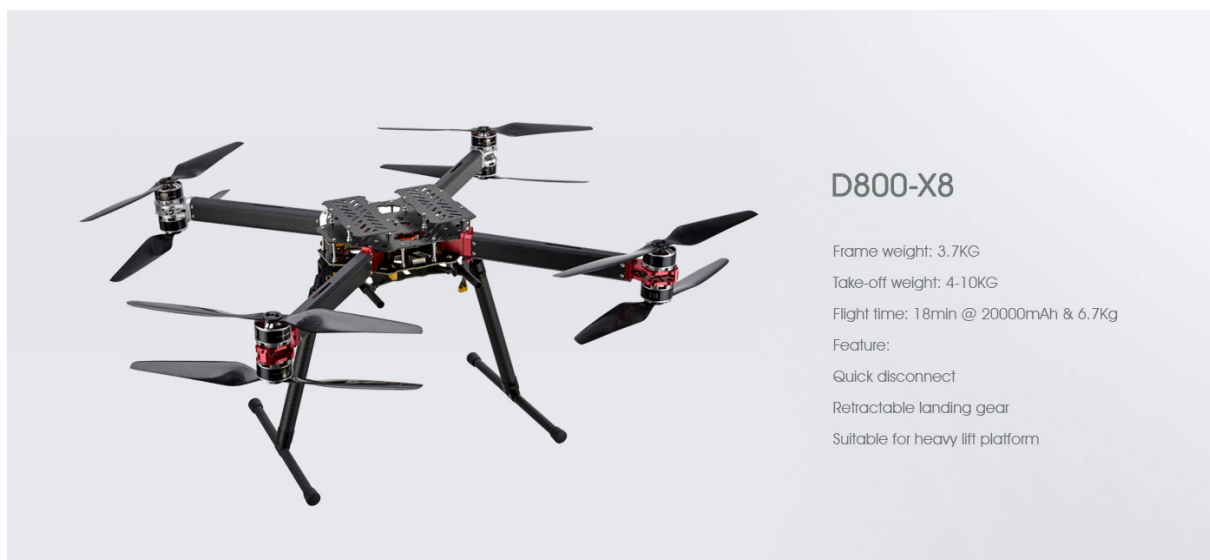
5 Konstrukce a příprava pro reálnou aplikaci ŘS

V této poslední kapitole je pozornost přesunuta na pozdější aplikace řídicích systémů, které byly doposud jen simulačně ověřeny. Je zde představena finální konstrukce, která byla pořízena pro pozdější testovací účely. Vzhledem k tomu, že pro aplikaci řídicího systému bude využit systém *REX*, je zde i prezentován nástin budoucího zapojení.

5.1 Konstrukce UAV

Existuje mnoho způsobů, jak zkonstruovat UAV. Je možnost využít i nějaké běžné stavebnice, nechat si na zakázku vyrobit tělo drona, přistoupit k vlastnímu provedení z dostupných zdrojů nebo si pořídit tzv. modulární stavebnici pro drony. V této práci byla zvolena právě poslední varianta. Tato speciální stavebnice nabízí určitou variabilitu zapojení a zároveň je již osazena rotory a elektronikou, ale zároveň dovoluje připojení libovolné platformy pro její ovládání.

Použitý dron, který je z velké části stavebnicí, nese jméno *D800-X8* a je od firmy *DongYang Smart Technology Co.,Ltd.*. V této sestavě disponuje čtyřmi rameny, každé z nich je osazeno dvěma rotory. Ovšem po dokoupení dalších součástí je jeho tělo možné modifikovat a přidat další ramena s rotory nebo naopak ramena otočit do různých pozic.



Obrázek 66: *D800-X8* [31]

Obrázek 67: *D800-X8* - specifikace [31]Obrázek 68: *D800-X8* - ramena s motory [31]

Typ *D800-X8* je tedy kvadroptéra, ovšem je zde možnost dokoupit ramena a přestavět ji na omnikoptéru, která je v této práci zkoumána. Prvním krokem samozřejmě ale bude zprovoznění klasické kvadroptéry. Není vhodné se ihned přenést na složitější konstrukce, pokud nezvládnete ty základní.

Kvadroptéra typu *D800-X8* váží 3,7 kg bez další zátěže. Zde bude nutno připočítat ještě řídicí elektroniku, která bude na dron umístěna, ale vzhledem k maximální možné váze 10 kg by neměl být problém. Samozřejmě čím bude lehčí, tím lépe se bude ovládat. Maximální možný let také samozřejmě bude záležet na baterii, kterou by ale neměl být problém vyměnit za výkonnější. Co se týče motorů, tak ty jsou typu *BE4114/400KV* a jsou schopny maximálního výkonu 400 W [31] a na dronu se jich nachází celkem osm.

Na tomto setu je velice výhodné to, že pokud by došlo k poškození kterékoliv části, není problém ji v celku rychle vyměnit za novou. Další možností jak snadno zvýšit tah, který generují rotory, je vyměnit dosavadní vrtule za jiné. Na stránkách výrobce jich lze nalézt celou řadu.

5.2 Přípravné zapojení

Jelikož budou navržené řídicí algoritmy reálně testovány na předchozí konstrukci drona v programovém prostředí *REX*, bude zde uvedeno základní zapojení. Jedná se spíše o přípravné schéma, které bude dále upravováno dle potřeby. Pro přehlednost bude vhodné celý řídicí systém rozdělit do několika bloků, přičemž každý bude plnit jinou funkci.

První blok bude věnován zpracování signálů ze senzorů a jejich převodu do stavových veličin. Zde může být v budoucnu umístěn i estimátor, který by na základě dat ze senzorů odhadoval například polohu nebo natočení dronu. V prvotní fázi testování bude tento blok sloužit nejspíše jen pro zpracování dat z akcelerometru nebo gyroskopu, případně nějakého výškového senzoru.

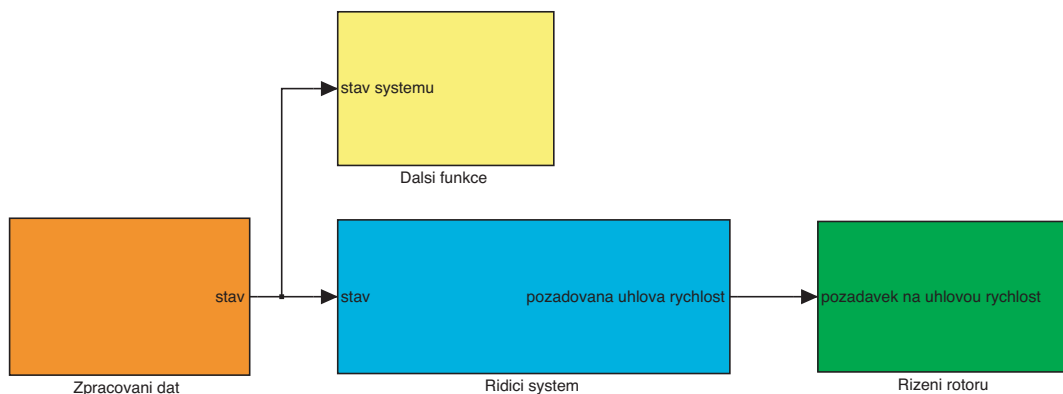
Další blok bude obsahovat hlavní regulátor, tedy v prvotní fázi stavový zpětný regulátor, který byl navržen metodou LQR za pomoci podmínek ze symetrie systémů. V pozdější fázi, kde bude pozornost věnována regulaci podél předem zvolené trajektorie, bude samozřejmě tento blok mnohem komplikovanější. Hlavně v případě omnikoptéry, kde bude vhodné udělat pro každý manévř jeden regulátor, aby se mohl dron otáčet libovolným směrem a bylo maximálně využito jeho možností.

Důležitý blok je i obsluha rotorů. V této práci je uvažováno, že rotory reagují okamžitě a udržují požadovanou úhlovou rychlost, tedy není nutné na ně navrhovat regulátor, který by je udržoval v požadované úhlové rychlosti. Pro reálnou aplikaci bude nutné pro každý rotor navrhnout regulátor. Za předpokladu, že jsou všechny rotory stejné, bude stačit navrhnout jeden regulátor, který bude ale nutné na všech zbylých rotorech otestovat, aby se předešlo případným chybám s řízením. Regulátor bude jako vstup mít požadovanou úhlovou rychlost a jako výstup bude muset mít příslušně velký proud, který bude roztáčet rotor a udržovat konstantní požadovanou úhlovou rychlost. Některé motory tyto regulátory již obsahují, tak jedinou nutností by bylo zahrnout změřenou přechodovou charakteristiku rotorů do návrhu regulátoru.

Další důležitou věcí je komunikace mezi dronem a uživatelem. V první fázi to bude řešeno klasicky přes připojení pomocí *REXu*, kde v počítači budou zadávány nejrůznější požadavky skrze referenční signály. V pozdější fázi bude vhodnější připojit ke dronu dálkový ovladač, díky kterému se bude dát s dronem mnohem lépe létat. Tato konfigurace nejspíše nebude vyřešena celá v *REXu*, ale především přímo v konfiguraci na cílové platformě.

Další dělení již nebude momentálně rozebíráno. Možnosti jsou prakticky neomezené, je možné například zahrnout do volně programovatelného bloku rozpoznávání obličejů nebo objektů pomocí připojené kamery, přidat ke dronu pracovní nástroje a tak dále.

Tedy základní struktura zapojení v *REXu* je hotova. Na následující obrázku je vidět screenshot z programového prostředí *REX*, kde je tato výše popsaná struktura vytvořena. Jedná se čistě o přípravné zapojení, které bude ještě mnohokrát modifikováno dle potřeb na regulaci.



Obrázek 69: REX - základní schéma

Jak již bylo výše zmíněno, jedná se pouze o základní schéma. Konkrétní schéma bude vytvořeno až vzhledem ke konfiguraci a zapojení dronu. Pro stavový zpětný regulátor se pak asi nejvýhodněji nabízí blok *Extended addition and subtraction*. Jeho kombinací lze velice jednoduše vytvořit připojení stavového zpětného regulátoru.

6 Závěr

Hlavní myšlenkou diplomové práce je proniknutí do problematiky dronů, které se v dnešní době těší stále větší oblibě v širokém spektru aplikací. Vedle klasické kvadrokoptéry práce představuje i myšlenku omnikoptéry, která řeší nedostatky v přímočarém pohybu bez natáčení. Všechny simulace a řídicí systémy jsou navrženy pro obě dvě varianty a následně porovnány.

Základem pro úspěšné navržení vhodného řídicího algoritmu je matematický model. Ten byl v této práci vytvořen jak pro kvadrokoptéru tak i pro omnikoptéru. Nejdříve byl model podrobně odvozen pro Eulerovy úhly, kde se pro transformaci mezi jednotlivými soustavami souřadnic využívají tzv. matice rotace. Matice rotace \mathbf{R} (2.16) je ovšem pro $\beta = \pm \frac{\pi}{2}$ singulární, z tohoto důvodu jsou zavedeny kvaterniony. Kvaterniony také nabízejí transformace mezi jednotlivými soustavami souřadnic, ale tentokrát se zde singularity nevyskytují. Ovšem tyto transformace již obsahují 4 prvky a tím pádem jsou i hůře čitelné než Eulerovy úhly. Z tohoto důvodu jsou pro vizualizaci výstupy modelů vždy převedeny do Eulerových úhlů, vzhledem k převodům se mohou v grafech vyskytovat skoky, pokud patřičný úhel překoná hranici $\langle -2\pi, 2\pi \rangle$. Tyto skoky ovšem neznamenají problém, jedná se pouze o vizualizaci. Všechny simulační nelineární modely pracují na rovnicích odvozených s kvaterniony.

Pro řízení je zvolena stavová zpětná vazba, která je realizována metodou LQR doplněnou o podmínky návrhu pro symetrické systémy. Zavedení podmínky symetrie do řídicího systému může velice zjednodušit návrh a vylepšit také vlastnosti regulátoru, kde se zachovává symetrie systému otevřené smyčky v uzavřené smyčce. Většina systémů, které člověk navrhne, mají v sobě v jistém ohledu zakomponovanou určitou symetrii. Při jejím využití se pro případ LQR značně zredukuje počet volných návrhových parametrů. Všechny navržené regulátory v této práci vycházejí z metody LQR. Nejdříve je navržen regulátor pro stabilizaci UAV, který předpokládá znalost celého stavového vektoru. Následně je tento regulátor rozšířen o integrátory, které dovolují přesnou polohovou regulaci. V případě kvadrokoptéry je polohová regulace limitována, což je dáno jejími konstrukčními vlastnostmi. Poslední navržený řídicí systém již předpokládá reálnější znalost stavového vektoru, kdy není možné znát například polohu v ose \vec{x}_0 nebo \vec{y}_0 a jejich rychlosti.

Celá práce je doplněna o 3D vizualizaci, díky které je možné s kvadrokoptérou nebo omnikoptérou létat v simulovaném reálném čase a lépe testovat chování navrženého regulátoru. Pro ještě lepší možnosti ovládání byl k *Simulinku* připojen gamepad, díky kterému lze UAV odvládat.

Poslední část diplomové práce se zaměřuje na popis budoucí testovací konstrukce kvadrokoptéry, která se dá ale díky modulární konstrukci přestavět na omnikoptéru. Tato konstrukce nabízí velkou variabilitu v konečném sestavení a již obsahuje základní elektroniku v základu. Například každé rameno je osazeno 2 rotory, které mají všechny vstupy a výstupy umístěny v konektoru, který je přímo osazen uvnitř nosníku ramene. Také je připraveno základní zapojení pro řídicí systém *REX*, kde probíhá veškeré řízení a obsluha vstupů a výstupů.

Díky základním simulacím, které byly provedeny, lze pozorovat značné výhody v případě nasazení omnikoptéry. Díky vytvoření virtuálních vstupů a aplikaci symetrického řízení lze s omikoptérou létat v libovolném směru při libovolném natočení.

Na tuto práci bude dále navázáno aplikací řídicích algoritmů na platformu *Raspberry Pie*, která bude implementována do konstrukce *D800-X8*. Pro úspěšnou aplikaci řídicích algoritmů je nutné získat všechny potřebné parametry konstrukce zvoleného UAV a jeho rotorů. Dále bude UAV osazeno vhodnými senzory, které by poskytovaly informace pro řídicí algoritmus. Jedná se především o zrychlení v jednotlivých osách, úhlových rychlostech a informaci o výšce. Po úspěšném řízení klasické kvadroptéry lze přistoupit na složitější omnikoptéru. Po aplikaci základních řídicích algoritmů bude nutné již využít t-variantní regulátory, které dokáží dodržovat příslušnou trajektorii a natočení.

Reference

- [1] *The secret history of drones*, The Guardian [online], [cit. 24.3.2018].
Dostupné z:
<https://www.theguardian.com/world/shortcuts/2013/feb/10/secret-history-of-drones-1916>
- [2] *How Do Drones Work And What Is Drone Technology*, DroneZon [online], [cit. 24.3.2018].
Dostupné z:
<https://www.dronezon.com/learn-about-drones-quadcopters/what-is-drone-technology-or-how-does-drone-technology-work/>
- [3] *MQ-1B Predator*, U. S. Air Force [online], [cit. 24.3.2018].
Dostupné z:
<http://www.af.mil/About-Us/Fact-Sheets/Display/Article/104469/mq-1b-predator/>
- [4] NEMATI, A., KUMAR M., *Modeling and Control of a Single Axis Tilting Quadcopter*
In: American Control Conference, 2014.
- [5] *Autonomous Micro Aerial Vehicles*, Multi-Scale Robotics and Automation Lab - Purdue Robotics [online], [cit. 24.3.2018].
Dostupné z: <https://multiscalerobotics.org/research/current/>
- [6] BARÁNEK, Radek, ŠOLC František, *Modelling and Control of a Hexa-copter* Brno University of Technology, 2012.
- [7] HARRISON, H. R., NETTLETON T., *Advanced Engineering Dynamics*, New York: Wiley, 1997.
- [8] SPIEGEL, M. R., PROYKOVA, Y., *Schaum's Outline of Theory and Problems of Theoretical Mechanics: With an Introduction to Lagrange's Equations and Hamiltonian Theory*, New York: McGraw-Hill Book Co, 1980.
- [9] BIEVER, E., *Controlling of an Single Drone, Hovering the Drone During Flight Modes*, Eindhoven: Department of mechanical engineering, Dynamics & control, 2015.
- [10] LEE, Taeyoung, LEOKY, Melvin, MCCLAMROCH, N. Harris, *Geometric Tracking Control of a Quadrotor UAV on SE(3)*. In: 49th IEEE Conference on Decision and Control (CDC), IEEE, 2010, s. 5420-5425.
- [11] HUANG, Qikai (Bruce Wingo). *Mathematical Modeling of Quadcopter Dynamics*. In: Mathematical Sciences Technical Reports (MSTR), 2016, s. 158.
- [12] ÚŘEDNÍČEK, Z., OPLUŠTIL, M., *Equations of Motion and Physical Model of Quadcopter in Plain*. In: CSCC, Santorini, 2014.

-
- [13] BRESCIANINI, Dario, D'ANDREA, Raffaello, *Design, Modeling and Control of an Omni-Directional Aerial Vehicle*. In: IEEE International Conference on Robotics and Automation (ICRA), Stockholm, 2016.
- [14] LUUKKONEN, Teppo, *Modelling and Control of Quadcopter*, Aalto University, 2011.
- [15] SELBY, Will, *Quadrotor System Modeling - Non-linear Equations of Motion* [online], [cit. 23.10.2017].
Dostupné z: <https://www.wilselby.com/research/arducopter/modeling/>
- [16] BOUADI, Hakim, BOUCHOUCHA, M., TADJINE, M., *Sliding Mode Control based on Backstepping Approach for an UAV Type-Quadrotor*. In: International Journal of Applied Mathematics and Computer Sciences, 4., 2007, s. 12-17.
- [17] COOK, M. V., *Flight Dynamics Principles: A Linear Systems Approach to Aircraft Stability and Control*. Burlington, MA: Butterworth-Heinemann/Elsevier, 2007.
- [18] OSCARSON, Oscar, *Design, Modeling and Control of an Octocopter* [online], [cit. 4.11.2017].
Dostupné z: <https://www.diva-portal.org/smash/get/diva2:857660/FULLTEXT01.pdf>
- [19] RONGIER, P., LAVAREC E. a PIERROT F., *Kinematic and Dynamic Modeling and Control of a 3-Rotor Aircraft* [online] In: Proceedings of the 2005 IEEE International Conference on Robotics and Automation . IEEE, 2005, s. 2606-2611.
- [20] JARDIN, Matt R. a MUELLER, Eric R., *Optimized Measurements of UAV Mass Moment of Inertia with a Bifilar Pendulum*. In: AIAA Guidance, Navigation and Control Conference and Exhibit 2007, South Carolina.
- [21] SCHLEGEL, Miloš a KÖNIGSMARKOVÁ, Jana, *Parametric Jordan Form Assignment by State-Derivative Feedback*. In: 2015 International Conference on Process Control (PC), Štrbské Pleso, 2015.
- [22] ROWLAND, Todd a WEISSTEIN, Eric, W., *Jordan Canonical Form*. [online], [cit. 13.3.2018].
Dostupné z: <http://mathworld.wolfram.com/JordanCanonicalForm.html>
- [23] WEISSTEIN, Eric, W., *Jordan Matrix Decomposition*. [online], [cit. 13.3.2018].
Dostupné z: <http://mathworld.wolfram.com/JordanMatrixDecomposition.html>
- [24] LANG, Serge, *Linear Algebra*. Springer-Verlang New York, Inc., 1987.
- [25] DIEBEL, James, *Representing Attitude: Euler Angles, Unit Quaternions and Rotation Vectors*. Stanford University, California, 2006.

-
- [26] *Conversion between quaternions and Euler angles*. IPFS is the Distributed Web [online], [cit. 15.3.2018].
Dostupné z:
https://ipfs.io/ipfs/.../wiki/Conversion_between_quaternions_and_Euler_angles.htm
- [27] ŠVEJDA, Martin, Přednášky k předmětu Úvod do robotiky a mechatroniky (URM), Zápa-
dočeská univerzita v Plzni, 2012.
- [28] SCHWAB, Arend, L., *Quaternions, Finite Rotation and Euler Parameters*. Delft University
of Technology, 2002.
- [29] ADORNO, Bruno, Vilhena, *Robot Kinematic Modeling and Control Based on Dual Qua-
ternion Algebra — Part I: Fundamentals*. Federal University of Minas Gerais, 2017.
- [30] KÖNIGSMARKOVÁ, Jana, *Symetrické systémy*. Práce ke státní doktorské zkoušce, Zápa-
dočeská univerzita v Plzni, 2017.
- [31] *Dong Yang Innovative Technology Limited* [online], [cit. 28.4.2018].
Dostupné z: <http://www.dys.hk>