

Západočeská univerzita v Plzni
Fakulta aplikovaných věd
Katedra kybernetiky

DIPLOMOVÁ PRÁCE

Návrh řízení kvadrotorové helikoptéry

Plzeň, 2015

Bc. Zdeněk Bouček

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 15. května 2015

.....
vlastnoruční podpis

Abstrakt

Tato diplomová práce se zabývá návrhem řízení letu kvadrotorové helikoptéry. Nejprve jsou popsány základní charakteristiky chování kvadrotorové helikoptéry. Následně je uveden matematický model, který je poté rozšířen o další aerodynamické jevy. Dále je popsán unscentovaný Kalmanův filtr, který je připraven pro odhadování orientace a polohy helikoptéry na základě modelu, generovaného řízení a měření palubních senzorů. Poté je navrženo řízení otáček rotorů, regulátor orientace a regulátor pro řízení polohy kvadrotorové helikoptéry. Nakonec je regulátor otestován v simulaci.

Klíčová slova

kvadrotorová helikoptéra, matematické modelování, unscentovaný Kalmanův filtr, PID regulátor, LQ regulátor

Abstract

This diploma thesis is focused on design of flight control of quadrotor UAV. At first quadrotor UAV's basic characteristics of behavior are described. Further the mathematical model is presented and extended with other aerodynamic effects. After that the unscented Kalman filter is described and prepared for the estimation of orientation and position of UAV through the mathematical model, generated control and measurement from onboard sensors. In the next part rotor speed control, orientation control and position controller of quadrotor UAV are designed. Finally the flight control is tested in simulation.

Keywords

quadrotor UAV, mathematical modeling, unscented Kalman filter, PID controller, LQ controller

Obsah

1	Úvod	1
2	Návrh modelu kvadrotorové helikoptéry	2
2.1	Princip pohybu kvadrotorové helikoptéry a popis prostoru	2
2.2	Základní matematický popis chování kvadrotorové helikoptéry	5
2.3	Síly působící na tělo kvadrotorové helikoptéry	6
2.4	Momenty působící na tělo kvadrotorové helikoptéry	7
2.5	Kompletní matematický popis chování kvadrotorové helikoptéry	8
3	Návrh algoritmu pro odhad orientace a polohy kvadrotorové helikoptéry	10
3.1	Unscentovaný Kalmanův filtr	10
3.1.1	Unscentovaná transformace	10
3.1.2	Popis unscentovaného Kalmanova filtru	12
3.2	Diskrétní nelineární model kvadrotorové helikoptéry	15
3.2.1	Diskretizace nelineárního modelu	16
3.2.2	Výpočet diskrétního nelineárního modelu kvadrotorové helikoptéry .	16
3.3	Senzory	19
3.3.1	Inerciální měřicí jednotka	19
3.3.2	Měření pro unscentovaný Kalmanův filtr	22
3.3.3	Úprava stavu kvadrotorové helikoptéry pro predikci měření	22
4	Návrh řídicího algoritmu kvadrotorové helikoptéry pro sledování zadané trajektorie	23
4.1	Regulátor otáček rotorů	23
4.2	Plánování otáček rotorů	26
4.3	Regulátor orientace kvadrotorové helikoptéry	27
4.4	Regulátor polohy kvadrotorové helikoptéry	30
5	Simulace řízené kvadrotorové helikoptéry	36
5.1	Parametry modelu kvadrotorové helikoptéry	36
5.2	Parametry regulátorů a unscentovaného Kalmanova filtru	37
5.3	Výsledky simulace	39
5.3.1	Stabilizace kvadrotorové helikoptéry a let do určeného bodu	39
5.3.2	Let kvadrotorové helikoptéry dle generované trajektorie	42
5.3.3	Odhad stavu kvadrotorové helikoptéry při stabilizaci a letu do určeného bodu	42

6	Závěr	51
A	Implementace funkcí v Matlabu	54
A.1	Unscenovaný Kalmanův filtr	54
A.2	Diskrétní model kvadrotorové helikoptéry	56
A.2.1	Diskretizace spojitého modelu	56
A.3	Diskrétní model kvadrotorové helikoptéry	60

Seznam symbolů

symbol	název	jednotka
acc_i	měření zrychlení z akcelerometru v ose i'	$\text{m}\cdot\text{s}^{-1}$
α	parametr pro výpočet sigma bodů	-
β	parametr pro výpočet sigma bodů	-
D	koeficient odporu prostředí	-
D_v	odpor prostředí	N
$e^{\Omega \times}$	rotační matice s $\mathbf{SO}(3)$	-
F	síla působící na helikoptéru v její soustavě	N
g	tíhové zrychlení	$\text{m}\cdot\text{s}^{-2}$
G_z	gravitační síla	N
gps_i	měření polohy z modulu GPS na ose i	m
$gyro_i$	měření úhlové rychlosti z gyroskopu kolem osy i'	$\text{rad}\cdot\text{s}^{-1}$
h	vzdálenost osy rotoru od těžiště helikoptéry na ose z'	m
I	matice momentu setrvačnosti helikoptéry	$\text{kg}\cdot\text{m}^2$
I_M	koeficient setrvačnosti rotorů	-
K	parametr proporční složky regulátoru otáček motoru	-
κ	parametr pro výpočet sigma bodů	-
k_β	koeficient tuhosti rotoru	-
k_f	koeficient třepotání listu	-
k_M	koeficient momentu rotorů	-
k_T	koeficient tahu rotorů	-
l	délka ramene helikoptéry	m
m	celková hmotnost helikoptéry	kg
M	moment síly působící na helikoptéru v její soustavě	$\text{N}\cdot\text{m}$
$M_{f\phi}, M_{f\theta}$	momenty síly vytvářené vlivem třepotání listů rotoru	$\text{N}\cdot\text{m}$
M_ϕ, M_θ, M_ψ	momenty síly vytvářené translačním pohybem rotorů	$\text{N}\cdot\text{m}$
mag_i	měření magnetického pole z magnetometru v ose i'	μT
N	parametr filtrace derivační složky regulátoru otáček motoru	-
ω_i	rychlost otáčení i -tého rotoru	$\text{rad}\cdot\text{s}^{-1}$
Ω	orientace kvadrotorové helikoptéry v její soustavě	rad
Ω_\times	rotační matice v $\mathfrak{so}(3)$	rad
Ω^*	požadovaná orientace kvadrotorové helikoptéry v její soustavě	rad
ϕ, θ, ψ	Eulerovy úhly kvadrotorové helikoptéry v její soustavě	rad
ϕ^*, θ^*, ψ^*	požadované úhly kvadrotorové helikoptéry v její soustavě	rad

symbol	název	jednotka
$P_{k k}$	filtrační odhad kovarianční matice	
$P_{k k-1}$	predikční odhad kovarianční matice	
PWM_i	pulzně šířkový modulovaný signál pro motor i	%
r	poloha helikoptéry v inerciální soustavě	m
r'	poloha helikoptéry v soustavě helikoptéry	m
T	tah rotorů	N
T_d	derivační časová konstanta regulátoru otáček motoru	-
T_i	integrační časová konstanta regulátoru otáček motoru	-
$T_{s_{acci}}$	perioda měření akcelerometru číslo i	s
T_{s_d}	perioda diskrétního nelineárního modelu	s
$T_{s_{gps}}$	perioda měření modulu GPS	s
$T_{s_{gyro}}$	perioda měření gyroskopu	s
T_{s_m}	perioda regulátoru otáček motoru	s
$T_{s_{mag}}$	perioda měření magnetometru	s
T_{s_o}	perioda pro regulátory orientace	s
$T_{s_{x'}}$	perioda pro regulátory polohy na souřadnicích x'	s
$T_{s_{y'}}$	perioda pro regulátory polohy na souřadnicích y'	s
T_{s_z}	perioda regulátoru tahu	s
T_t	časová konstanta mechanismu pro zabránění unášení integrační složky regulátoru otáček motoru	-
x, y, z	složky polohy v inerciální soustavě	m
x', y', z'	složky polohy v soustavě helikoptéry	m
$\hat{x}_{k k}$	filtrační odhad střední hodnoty	
$\hat{x}_{k k-1}$	predikční odhad střední hodnoty	

Seznam obrázků

2.1	Umístění kvadrotorové helikoptéry v prostoru	3
2.2	Znázornění poměru tahu rotorů pro naklánění helikoptéry. Zdroj: [2, str. 6]	3
2.3	Znázornění poměru tahu rotorů pro sklánění helikoptéry. Zdroj: [2, str. 6]	4
2.4	Znázornění poměru tahu rotorů pro otáčení helikoptéry. Zdroj: [2, str. 7]	4
2.5	Vliv zdánlivého větru na pohyb kvadrotorové helikoptéry	7
2.6	Vliv třepotání listů rotorů na kvadrotorovou helikoptéru (zelený list se v prostoru pohybuje relativně rychleji a červený se pohybuje pomaleji) . .	8
2.7	Schéma modelu kvadrotorové helikoptéry (blok <i>Matrix Multiply</i> provádí maticové násobení mezi prvním a druhým vstupem)	9
3.1	(a) transformace sigma bodů skrz nelineární funkci a vypočítané normální rozložení; (b) linearizace nelineární funkce pomocí Taylorova rozvoje. Zdroj: [14]	11
3.2	Ukázka unscentované transformace střední hodnoty a kovariance, zleva doprava: skutečná stř. hodnota a kovariance, linearizovaná (používáno v rozšířeném Kalmanově filtru), unscentovaná transformace. Zdroj: [4]	13
3.3	Porovnání odhadu střední hodnoty a kovariance pomocí rozšířeného Kalmanova filtru (EKF) a unscentovaného Kalmanova filtru (UKF). Zdroj: [14]	14
4.1	Regulační schéma kvadrotorové helikoptéry	23
4.2	Schéma regulačního smyčky s regulovaným systémem S a paralelním PID regulátorem (w - požadovaná hodnota, e - regulační odchylka, u - řízení, y - výstup)	24
4.3	Schéma regulačního smyčky s regulovaným systémem S a PID regulátorem (w - požadovaná hodnota, e - regulační odchylka, u - řízení, \tilde{u} - omezené řízení, y - výstup)	25
4.4	Schéma zapojení regulátoru i -tého motoru do řídicího systému	26
4.5	Schéma regulátoru orientace	30
4.6	Schéma LQ regulátorů orientace doplněných o kompenzaci třepotání listů rotorů (4.6a - regulace po ϕ , 4.6b - regulace po θ a 4.6c - regulace po ψ) . .	31
4.7	Schéma regulátoru polohy	33
4.8	Schéma transformace z inerciálního systému do systému helikoptéry pro regulátor polohy a plánování orientace a tahu (pozn.: blok <i>Matrix Multiply</i> provádí maticové násobení mezi prvním a druhým vstupem)	33
4.9	Schéma regulátoru polohy v souřadné soustavě helikoptéry	33
4.10	Schéma plánování orientace a tahu kvadrotorové helikoptéry	34
4.11	Vliv tahu T při natočení ϕ na ose y'	34

4.12	Vliv tahu T při natočení θ na ose x'	35
5.1	Otáčky rotorů při letu do požadovaného bodu	40
5.2	Poloha kvadrotorové helikoptéry a požadovaná poloha při letu do cílového bodu	41
5.3	Orientace kvadrotorové helikoptéry při letu do požadovaného bodu	41
5.4	Trajektorie letu kvadrotorové helikoptéry a cílový bod letu	42
5.5	Otáčky rotorů při letu po trajektorii	43
5.6	Poloha kvadrotorové helikoptéry a požadovaná poloha při letu po trajektorii	43
5.7	Orientace kvadrotorové helikoptéry při letu po trajektorii	44
5.8	Trajektorie letu kvadrotorové helikoptéry a požadovaná letová trajektorie .	44
5.9	Poloha kvadrotorové helikoptéry a její odhad při letu do cílového bodu . .	45
5.10	Rychlost kvadrotorové helikoptéry a její odhad při letu do cílového bodu .	45
5.11	Orientace kvadrotorové helikoptéry a její odhad při letu do cílového bodu .	46
5.12	Úhlová rychlost kvadrotorové helikoptéry a její odhad při letu do cílového bodu	46
5.13	Trajektorie letu kvadrotorové helikoptéry a její odhad při letu do cílového bodu	47
5.14	Poloha kvadrotorové helikoptéry a její odhad při letu po trajektorii	48
5.15	Rychlost kvadrotorové helikoptéry a její odhad při letu po trajektorii . . .	48
5.16	Orientace kvadrotorové helikoptéry a její odhad při letu po trajektorii . . .	49
5.17	Úhlová rychlost kvadrotorové helikoptéry a její odhad při letu po trajektorii	49
5.18	Trajektorie letu kvadrotorové helikoptéry a její odhad při letu po trajektorii	50

Seznam tabulek

5.1	Parametry modelu kvadrotorové helikoptéry pro simulaci	37
5.2	Parametry regulátoru otáček rotorů	37
5.3	Parametry pro výpočet LQ regulátorů orientace	38
5.4	Parametry LQ regulátorů pro regulaci orientace	38
5.5	Parametry pro výpočet LQ regulátorů polohy	39
5.6	Parametry LQ regulátorů pro regulaci polohy	39
5.7	Parametry unscentovaného Kalmanova filtru	40

1 Úvod

Cílem této práce je popsat typ UAV¹, který se nazývá kvadrotorová helikoptéra. Kvadrotorová helikoptéra je druh vrtulového letounu se čtyřmi rotory.

Multirotory² obecně jsou dnes velmi oblíbené zejména kvůli široko oblasti využití. Disponují také velkou variabilitou, lze je osadit nejrůznějšími zařízeními, ať už kamerou pro levné natáčení leteckých záběrů, speciálními senzory např. pro sběr dat v blízkosti aktivních vulkánů nebo termokamery pro policejní pátrání. Helikoptéry mají také jednoduchou konstrukci. Pohyb helikoptéry je většinou řízen pouze změnou otáček rotorů. Helikoptéry se vyrábějí a prodávají v různých velikostech a v různých cenových relacích, takže si je mohou dovolit i amatéři.

V této práci bude provedeno seznámení se způsobem tvorby letového modelu kvadrotorové helikoptéry, návrhu řízení letu a zpracování dat z palubních sensorů.

Ve druhé kapitole bude uveden způsob, jakým se kvadrotorová helikoptéra pohybuje, bude vytvořen matematický popis pohybu kvadrotorové helikoptéry v šesti stupních volnosti, který bude nakonec rozšířen o nějaké další aerodynamické vlivy na chování helikoptéry.

Ve třetí kapitole bude popsán matematický model, který může být použit k odhadování chování helikoptéry. Dále bude popsán unscatovaný Kalmanův filtr, který bude sloužit k odhadu stavu kvadrotorové helikoptéry. Nakonec bude uvedeno, jakým způsobem připravit pro filtr měření z palubních sensorů helikoptéry.

V další kapitole bude uvedeno, z jakých částí by se měl skládat řídicí systém kvadrotorové helikoptéry. Dále bude vysvětleno, jak navrhnout regulátor rotorů s některými rozšířeními. V poslední části kapitoly bude popsán návrh regulátoru orientace, tedy natáčení helikoptéry, a také návrh regulátoru polohy (řízení letu v prostoru).

Pátá kapitola bude obsahovat ověření funkce navržených regulátorů a filtru.

¹unmanned aerial vehicle - bezpilotní letoun

²vrtulník s větším počtem rotorů

2 Návrh modelu kvadrotorové helikoptéry

V této části bude nejprve popsán princip pohybu kvadrotorové helikoptéry, v závislosti na otáčkách rotorů. Dále bude popsán třídimenzionální kartézský souřadnicový systém, ve kterém se helikoptéra pohybuje. Poté bude popsáno chování kvadrotorové helikoptéry formou matematického modelu na základě Newton-Eulerových rovnic.

Matematický model bude vytvořen za účelem simulací a návrhu regulátoru polohy a návrhu algoritmu pro odhad orientace a polohy helikoptéry. Nakonec bude model doplněn za účelem zpřesnění popisu chování o jevy, které ovlivňují chování helikoptéry při translačním letu.

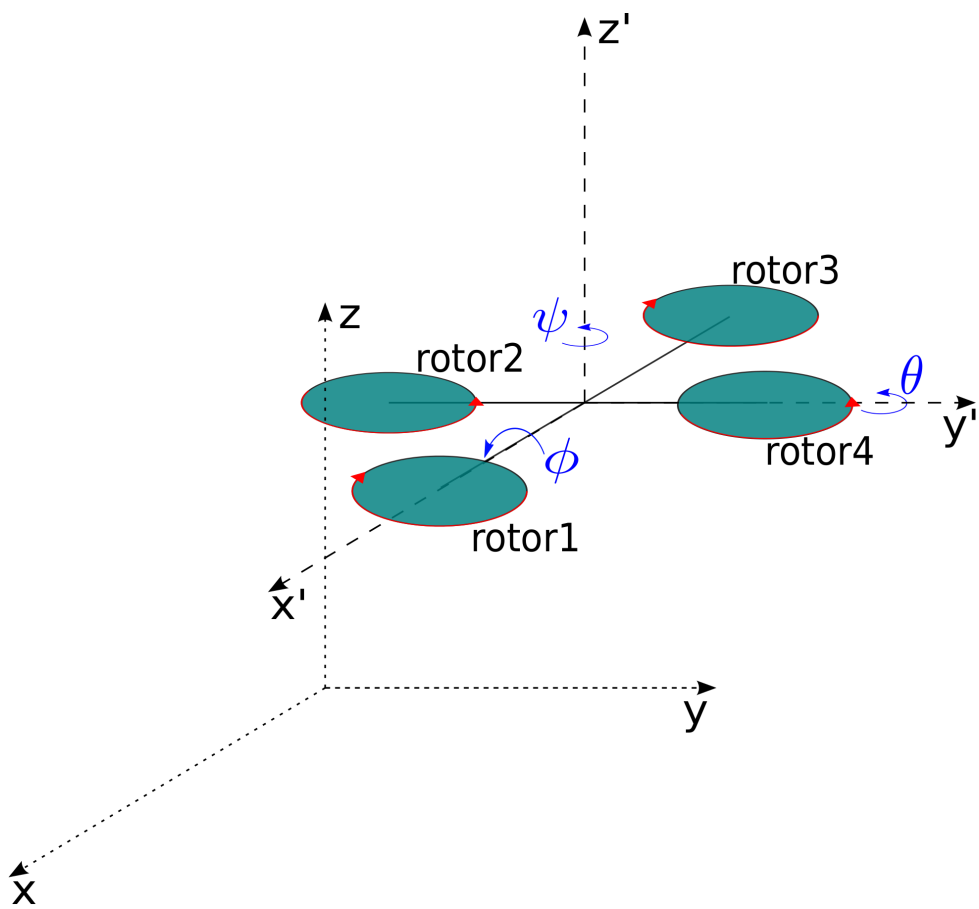
2.1 Princip pohybu kvadrotorové helikoptéry a popis prostoru

Prostor, ve kterém se kvadrotorová helikoptéra pohybuje, lze popsat v inerciálním souřadném systému (slouží k určení polohy helikoptéry) a v souřadném systému kvadrotorové helikoptéry, který slouží zejména k popisu působení sil a momentů na letoun. Poloha helikoptéry v souřadném systému je značena jako $r = [x, y, z]^T$ a polohu v souřadném systému helikoptéry budeme značit $r' = [x', y', z']^T$ (slouží například k popisu polohy rotoru vzhledem k helikoptéře).

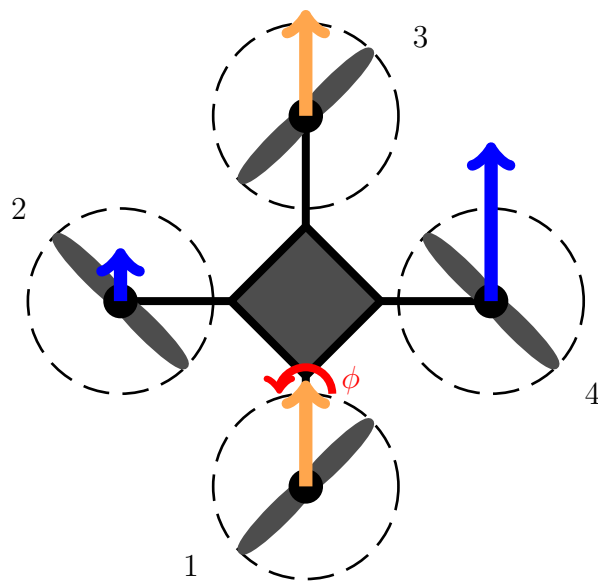
Orientace helikoptéry je popsána v souřadném systému helikoptéry pomocí Eulerových úhlů $\Omega = [\phi, \theta, \psi]^T$. Oba systémy jsou zakresleny na obrázku 2.1, kde jsou popsány jednotlivé rotory, směr jejich otáčení (červeně) a rotace helikoptéry.

Kvadrotorová helikoptéra se pohybuje v prostoru pomocí rotorů, které produkují v závislosti na otáčkách tah, což je síla, která působí u každého rotoru ve směru z' .

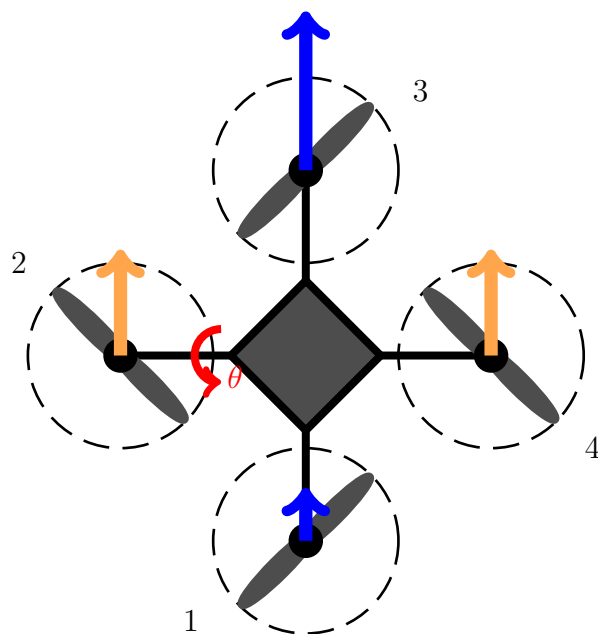
V závislosti na poměru tahu mohou rotory vytvářet momenty, které způsobují změnu orientace helikoptéry. Změna orientace v závislosti na poměru otáček rotorů je popsána na obrázcích 2.2 - změna náklonu, 2.3 - změna sklonu a 2.4 - změna otočení helikoptéry. Tah jednotlivých rotorů je na obrázcích znázorněn délkou šipky, která vede ze středu rotoru a směr rotace je znázorněn červenou šipkou.



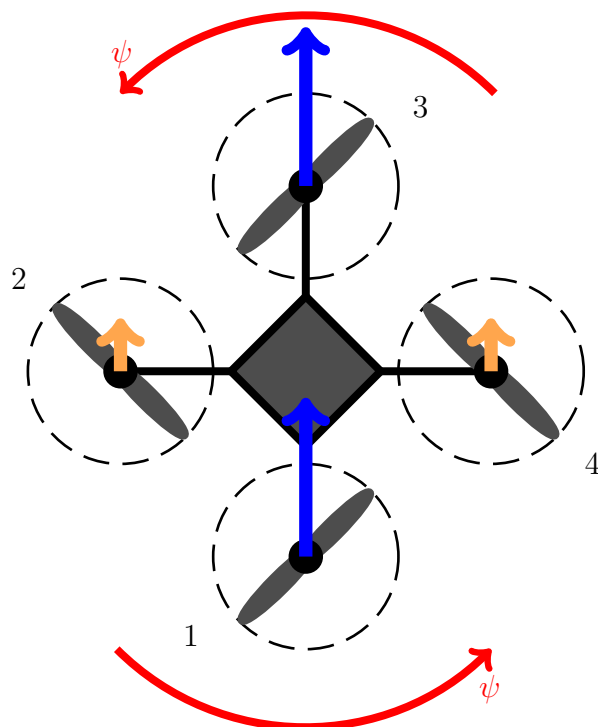
Obrázek 2.1: Umístění kvadrotorové helikoptéry v prostoru



Obrázek 2.2: Znázornění poměru tahu rotorů pro naklání helikoptéry. Zdroj: [2, str. 6]



Obrázek 2.3: Znázornění poměru tahu rotorů pro sklánění helikoptéry. Zdroj: [2, str. 6]



Obrázek 2.4: Znázornění poměru tahu rotorů pro otáčení helikoptéry. Zdroj: [2, str. 7]

2.2 Základní matematický popis chování kvadrotorové helikoptéry

Základní matematický popis vychází z Newtonovy (2.1) a Eulerovy (2.2) rovnice, kde m je hmotnost helikoptéry, \ddot{r} je zrychlení helikoptéry vůči inerciální soustavě, G_z (2.3) je gravitační síla a R (2.5) je rotační maticí pro přepočítání síly F , která působí na tělo helikoptéry, ze soustavy helikoptéry do inerciální soustavy. I je matice momentu setrvačnosti helikoptéry (prvky matice jsou jednotlivé momenty setrvačnosti), $\ddot{\Omega}$ je úhlové zrychlení kvadrotorové helikoptéry v soustavě helikoptéry, $\dot{\Omega}$ je úhlová rychlost helikoptéry v soustavě helikoptéry a M je moment, který působí na tělo helikoptéry.

$$m \cdot \ddot{r} = -G_z + R \cdot F \quad (2.1)$$

$$I \cdot \ddot{\Omega} = \dot{\Omega} \times I \cdot \dot{\Omega} + M \quad (2.2)$$

$$G_z = m \cdot [0 \ 0 \ g]^T \quad (2.3)$$

$$I = \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{yx} & I_{yy} & I_{yz} \\ I_{zx} & I_{zy} & I_{zz} \end{bmatrix} \quad (2.4)$$

Rotační matici popíšeme využitím $\mathbf{SO}(3)$ (viz [1, kap. 9]) pomocí exponenciálních map a Rodriguesova vzorce. Předvedeme tak problému s případnou blokáží Kardanova závěsu (také známý jako *gimbal lock*) (viz [5]) a následně ztrátě stupně volnosti. Výsledný vztah pro rotační matici je popsán v (2.5) a (2.6), kde $I_{3,3}$ je jednotková matice o rozměrech 3×3 . Inverzní rotační matice je rovna transponované, jelikož rotační matice je ortogonální.

$$R^T \cdot R = I_{3,3}$$

$$R^T = R^{-1}$$

$$\Omega_{\times} = \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix}_{\times} = \begin{bmatrix} 0 & -\psi & \theta \\ \psi & 0 & -\phi \\ -\theta & \phi & 0 \end{bmatrix}$$

$$\Phi = \sqrt{\phi^2 + \theta^2 + \psi^2}$$

$\Phi \neq 0$:

$$R = e^{\Omega} = e^{\Omega_{\times}} = I_{3,3} + \frac{\sin \Phi}{\Phi} \Omega_{\times} + \frac{1 - \cos \Phi}{\Phi^2} \Omega_{\times}^2 \quad (2.5)$$

$\Phi = 0$:

$$R = e^{\Omega} = e^{\Omega_{\times}} = I_{3,3} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2.6)$$

Základní model v rozepsané formě s dosazením je popsán vztahy (2.7), (2.8) a (2.9).

$$\Phi = 0 : \quad \begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = - \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \frac{F}{m} \quad (2.7)$$

$$\Phi \neq 0 : \quad \begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = - \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} + \left(\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} + \frac{\sin \Phi}{\Phi} \Omega_{\times} + \frac{1 - \cos \Phi}{\Phi^2} \Omega_{\times}^2 \right) \cdot \frac{F}{m} \quad (2.8)$$

$$\begin{bmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} = \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{yx} & I_{yy} & I_{yz} \\ I_{zx} & I_{zy} & I_{zz} \end{bmatrix}^{-1} \cdot \left(\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \times \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{yx} & I_{yy} & I_{yz} \\ I_{zx} & I_{zy} & I_{zz} \end{bmatrix} \cdot \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} + M \right) \quad (2.9)$$

Většina jevů, které zásadně ovlivňují chování helikoptéry, je způsobena otáčením rotorů helikoptéry (viz [2, kap. 3]). Jevy můžeme rozdělit na ty, které lze popsat jako momenty síly a na ty, které lze popsat jako síly.

2.3 Síly působící na tělo kvadrotorové helikoptéry

Síly F lze rozdělit do dvou částí. První část je celkový tah rotorů T a druhá odpor prostředí [8]. Obě tyto síly budou popsány v souřadnicích helikoptéry a musí být proto v (2.1) přenásobeny rotační maticí $e^{\Omega_{\times}}$.

Celkový tah je součtem tahů, které produkují jednotlivé motory. Stejně jako jednotlivé tahy působí ve směru osy z' a můžeme jej popsat vztahem (2.10).

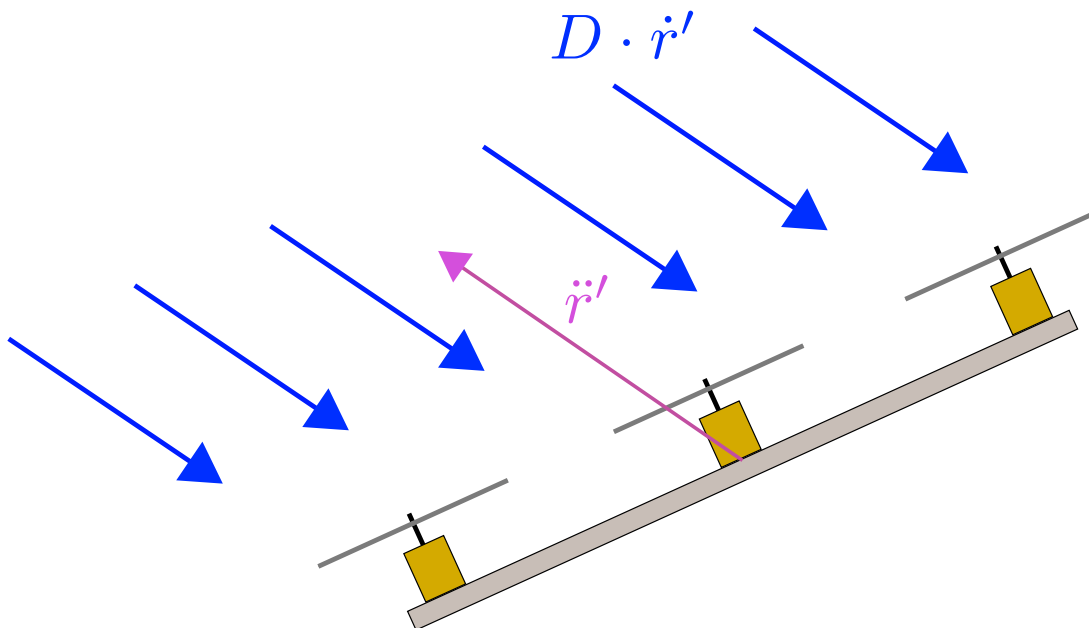
$$T = \begin{bmatrix} 0 & 0 & k_T \cdot \sum_{i=0}^N \omega_i^2 \end{bmatrix}^T, \quad N = 4 \quad (2.10)$$

Odpor prostředí D_v působí proti pohybu helikoptéry formou tzv. zdánlivého větru (viz obrázek 2.5). V našem případě je prostředím helikoptéry vzduch. Budeme předpokládat, že vzduch má stálou teplotu i tlak a nedochází k jeho proudění (větru). Při takovém zjednodušení je odpor, který vzduch působí proti pohybu helikoptéry, závislý pouze na rychlosti letu helikoptéry v souřadném systému helikoptéry a lze popsat vztahem (2.11), kde D je koeficient odporu prostředí.

$$D_v = D \cdot \dot{r}^2 = \begin{bmatrix} D_x \cdot \dot{x}'^2 \\ D_y \cdot \dot{y}'^2 \\ D_z \cdot \dot{z}'^2 \end{bmatrix} \quad (2.11)$$

Výsledné působící síly F lze popsat v souřadném systému helikoptéry vztahem (2.12).

$$F = T - D_v = \begin{bmatrix} 0 \\ 0 \\ k_T \cdot \sum_{i=0}^4 \omega_i^2 \end{bmatrix} - \begin{bmatrix} D_x \cdot \dot{x}'^2 \\ D_y \cdot \dot{y}'^2 \\ D_z \cdot \dot{z}'^2 \end{bmatrix} \quad (2.12)$$



Obrázek 2.5: Vliv zdánlivého větru na pohyb kvadrotorové helikoptéry

2.4 Momenty působící na tělo kvadrotorové helikoptéry

Momenty lze rozdělit na momenty, které jsou způsobeny otáčením rotorů, a momenty, které jsou způsobeny translačním letem helikoptéry. Otáčení rotoru vytváří moment M_ψ , který může způsobit otáčení helikoptéry kolem osy z . Velikost momentu M_ψ (2.15) závisí na poměru a změnách otáček rotorů. Další část, (2.13) a (2.14), závisí na tom, jak se rozprostírá tah mezi rotory (poměrem tahu mezi rotory vznikají momenty M_ϕ a M_θ). k_M je koeficient momentu rotoru a I_M je koeficient momentu setrvačnosti rotoru.

$$M_\phi = l \cdot k_T (-\omega_2^2 + \omega_4^2) \quad (2.13)$$

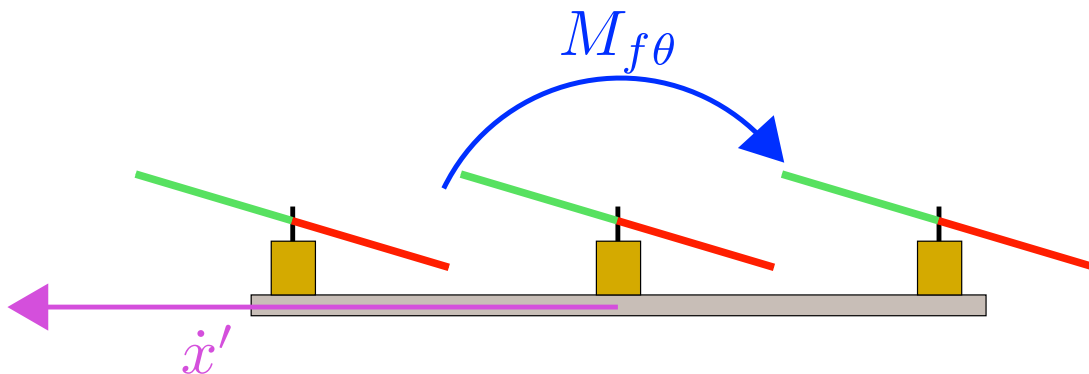
$$M_\theta = l \cdot k_T (-\omega_1^2 + \omega_3^2) \quad (2.14)$$

$$M_\psi = \sum_{i=1}^N (-1)^{i+1} \cdot (k_M \cdot \omega_i^2 + I_M \cdot \dot{\omega}_i), \quad N = 4 \quad (2.15)$$

Moment, který vzniká v důsledku translačního letu, je označován jako „třepotání listů rotoru“ [8] (také „blade flapping“) a projevuje se obecně u vrtulníků (viz obrázek 2.6). Je způsoben tím, že list rotoru, který se otáčí ve směru pohybu vrtulníku má vyšší rychlost než list který se zrovna otáčí opačným směrem. Důsledkem vyšší rychlosti generuje list vyšší tah a dochází k ohýbání listu směrem nahoru. „Pomalejší“ list je ohýbán dolů. To znamená, že každý list během jedné otáčky rotoru kmitne nahoru a dolů.

Jev má u kvadrotorové helikoptéry stabilizující účinek, jelikož způsobuje naklápění celé helikoptéry, která následkem naklonění působí tahem proti pohybu letu. Tento jev je důležitý i při konstrukci rotoru, protože kdyby nebyly listy rotoru dostatečně pružné, tak by se zmíněné mechanické kmitání mohlo přímo přenášet na motory a tak je poškodit.

Třepotání listů rotoru lze popsat jako (2.16) a (2.17), kde k_β je koeficient tuhosti rotoru, T je tah rotorů, h je vertikální vzdálenost rotoru od těžiště helikoptéry a $a_{1s,x}$ a



Obrázek 2.6: Vliv třepotání listů rotorů na kvadrotorovou helikoptéru (zelený list se v prostoru pohybuje relativně rychleji a červený se pohybuje pomaleji)

$a_{1s,y}$ jsou úhly vychýlení rotorů ve směru x a y . Úhly $a_{1s,x}$ a $a_{1s,y}$ lze vypočítat pomocí vztahu $a_{1s,x} = k_f \dot{x}'$ a $a_{1s,y} = k_f \dot{y}'$. Dosazením a aproximací $\sin a_{1s,x} \approx a_{1s,x}$ získáme (2.18) a (2.19), kde k_f je koeficient třepotání listu a \dot{x}' a \dot{y}' jsou translační rychlosti kvadrotorové helikoptéry v jejím souřadném systému.

$$M_{f\theta} = -4(k_\beta \cdot a_{1s,x} + T \cdot h \cdot \sin a_{1s,x}) \quad (2.16)$$

$$M_{f\phi} = -4(k_\beta \cdot a_{1s,y} + T \cdot h \cdot \sin a_{1s,y}) \quad (2.17)$$

$$M_{f\theta} = -4(k_\beta + T \cdot h) \cdot k_f \cdot \dot{x}' \quad (2.18)$$

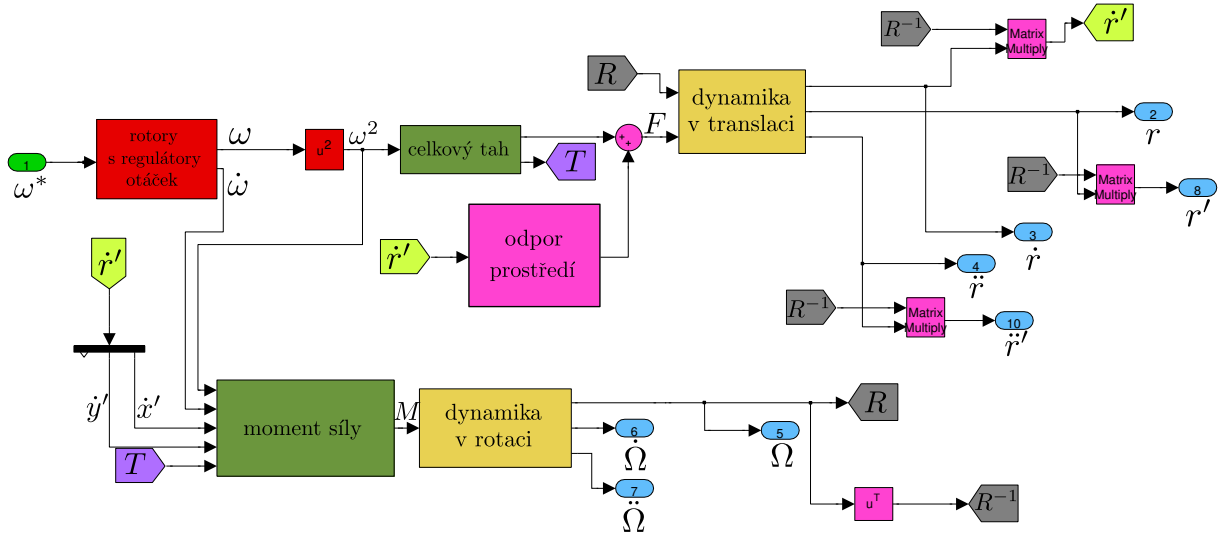
$$M_{f\phi} = -4(k_\beta + T \cdot h) \cdot k_f \cdot \dot{y}' \quad (2.19)$$

Výsledné působící momenty M lze popsat v souřadném systému helikoptéry vztahem (2.20).

$$M = \begin{bmatrix} M_\phi + M_{f\phi} \\ M_\theta + M_{f\theta} \\ M_\psi \end{bmatrix} = \begin{bmatrix} l \cdot k_T \cdot (-\omega_2^2 + \omega_4^2) - 4 \cdot (k_\beta + T \cdot h) \cdot k_f \cdot \dot{y}' \\ l \cdot k_T \cdot (-\omega_1^2 + \omega_3^2) - 4 \cdot (k_\beta + T \cdot h) \cdot k_f \cdot \dot{x}' \\ \sum_{i=1}^4 (-1)^{i+1} \cdot (k_M \cdot \omega_i^2 + I_M \cdot \dot{\omega}_i) \end{bmatrix} \quad (2.20)$$

2.5 Kompletní matematický popis chování kvadrotorové helikoptéry

Na základě modelu z části 2.2 a popisu jevů z části 4.2 a 4.2 nyní sestavíme kompletní model (schéma modelu na obrázku 2.7). Model získáme dosazením (2.12) do (2.8), (2.7) a (2.20) do (2.9). Výsledné rovnice, které tvoří matematický model kvadrotorové helikoptéry, jsou zapsány v (2.21), (2.22) a (2.23).



Obrázek 2.7: Schéma modelu kvadrotorové helikoptéry (blok *Matrix Multiply* provádí maticové násobení mezi prvním a druhým vstupem)

$$\Phi = 0 :$$

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = - \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} + \frac{1}{m} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} -D_x \cdot \dot{x}'^2 \\ -D_y \cdot \dot{y}'^2 \\ k_T \cdot (\sum_{i=0}^4 \omega_i^2) - D_z \cdot \dot{z}'^2 \end{bmatrix} \quad (2.21)$$

$$\Phi \neq 0 :$$

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = - \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} + \frac{1}{m} \cdot \left(\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} + \frac{\sin \Phi}{\Phi} \Omega_{\times} + \frac{1 - \cos \Phi}{\Phi^2} \Omega_{\times}^2 \right) \cdot \begin{bmatrix} -D_x \cdot \dot{x}'^2 \\ -D_y \cdot \dot{y}'^2 \\ k_T \cdot (\sum_{i=0}^4 \omega_i^2) - D_z \cdot \dot{z}'^2 \end{bmatrix} \quad (2.22)$$

$$\begin{bmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} = \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{yx} & I_{yy} & I_{yz} \\ I_{zx} & I_{zy} & I_{zz} \end{bmatrix}^{-1} \cdot \left(\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \times \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{yx} & I_{yy} & I_{yz} \\ I_{zx} & I_{zy} & I_{zz} \end{bmatrix} \cdot \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} + \begin{bmatrix} l \cdot k_T \cdot (-\omega_2^2 + \omega_4^2) - 4 \cdot (k_{\beta} + T \cdot h) \cdot k_f \cdot \dot{y}' \\ l \cdot k_T \cdot (-\omega_1^2 + \omega_3^2) + -4 \cdot (k_{\beta} + T \cdot h) \cdot k_f \cdot \dot{x}' \\ \sum_{i=1}^4 (-1)^{i+1} \cdot (k_M \cdot \omega_i^2 + I_M \cdot \dot{\omega}_i) \end{bmatrix} \right) \quad (2.23)$$

3 Návrh algoritmu pro odhad orientace a polohy kvadrotorové helikoptéry

Cílem této kapitoly je navrhnout algoritmus, který bude schopen dle měření odhadnout stav kvadrotorové helikoptéry. Stav kvadrotorové helikoptéry se skládá z polohy, rychlosti, orientace (popis pomocí Eulerových úhlů) a úhlové rychlosti. Pro odhad stavu helikoptéry bude využit unscentovaný Kalmanův filtr (dále UKF). UKF bude k odhadu stavu využívat nelineární diskretní matematický model kvadrotorové helikoptéry, měření z palubních senzorů helikoptéry a řízení, které generuje regulátor polohy v podobě požadovaných otáček rotorů.

3.1 Unscentovaný Kalmanův filtr

V této části bude popsána uncentovaná transformace, která je použita v unscentovaném Kalmanově filtru. Poté bude popsán problém odhadu orientace a polohy kvadrotorové helikoptéry a bude doplněn algoritmus UKF.

3.1.1 Unscentovaná transformace

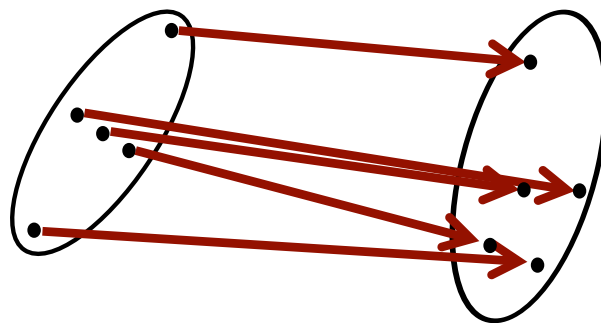
Unscentovaná transformace [6, 15] pracuje na principu volby několika bodů, které jsou následně distribuovány skrze nelineární funkci. Z distribuovaných vážených bodů je určeno normální rozdělení na základě střední hodnoty a kovariance bodů. Porovnání unscentované transformace s linearizací pomocí Taylorova rozvoje (používá se v rozšířeném Kalmanově filtru) je znázorněno na obrázku 3.1.

Volené body se nazývají „sigma body“ (také sigma points). Sigma body jsou $X^{[i]}$, kde i je číslo příslušného sigma bodu. Ke každému sigmabodu je přiřazena také příslušná váha $w^{[i]}$ taková, aby platilo 3.1, (3.2) a (3.3), kde \hat{x} je střední hodnota a P_x je kovarianční matice.

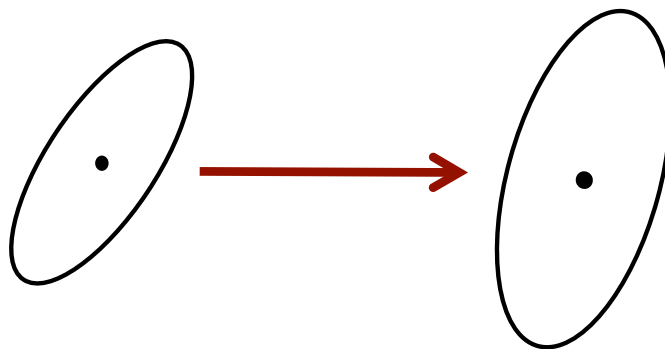
$$\sum_i w^{[i]} = 1 \quad (3.1)$$

$$\hat{x} = \sum_i w^{[i]} X^{[i]} \quad (3.2)$$

$$P_x = \sum_i w^{[i]} (X^{[i]} - \hat{x})(X^{[i]} - \hat{x})^T \quad (3.3)$$



(a)



(b)

Obrázek 3.1: (a) transformace sigma bodů skrz nelineární funkci a vypočítané normální rozložení; (b) linearizace nelineární funkce pomocí Taylorova rozvoje. Zdroj: [14]

Sigma body jsou voleny dle (3.4), (3.5) a (3.6), kde n je dimenze pravděpodobnostního rozdělení, λ je volitelný parametr a $(\cdot)_i$ je operátor, který z matice vybírá i -tý sloupec. V (3.5) a (3.6) vidíme, že je třeba počítat odmocninu z matice. Odmocninu lze řešit dvěma způsoby, první je pomocí diagonalizace, druhý je pomocí Choleského dekompozice. V našem případě použijeme Choleského dekompozici. Výpočet je numericky stabilní, je obsažen v prostředí Matlab¹ a snadno jej lze implementovat i v jiném programovacím jazyku².

$$X^{[0]} = \hat{x} \quad (3.4)$$

$$X^{[i]} = \hat{x} + \left(\sqrt{(n + \lambda) P_x} \right)_i, \quad i = 1, \dots, n \quad (3.5)$$

$$X^{[i]} = \hat{x} - \left(\sqrt{(n + \lambda) P_x} \right)_{i-n}, \quad i = n + 1, \dots, 2n \quad (3.6)$$

Výpočet váhy $w_m^{[i]}$ pro střední hodnotu a váhy $w_c^{[i]}$ pro kovarianci je popsán v (3.7). Vidíme, že výpočet váhy pro jednotlivé sigma body je závislý na dimenzi pravděpodobnostního rozdělení n a parametrech α , β a λ .

$$\begin{aligned} w_m^{[0]} &= \frac{\lambda}{n + \lambda} \\ w_c^{[0]} &= w_m^{[0]} + (1 - \alpha^2 + \beta) \\ w_m^{[i]} = w_c^{[i]} &= \frac{1}{2(n + \lambda)}, \quad i = 1, \dots, 2n \end{aligned} \quad (3.7)$$

V (3.8) jsou vztahy pro volbu parametrů. Parametry κ a α ovlivňují, jak daleko se budou nacházet sigma body od střední hodnoty \hat{x} . Parametr β je $\beta = 2$ pro normální pravděpodobnostní rozdělení.

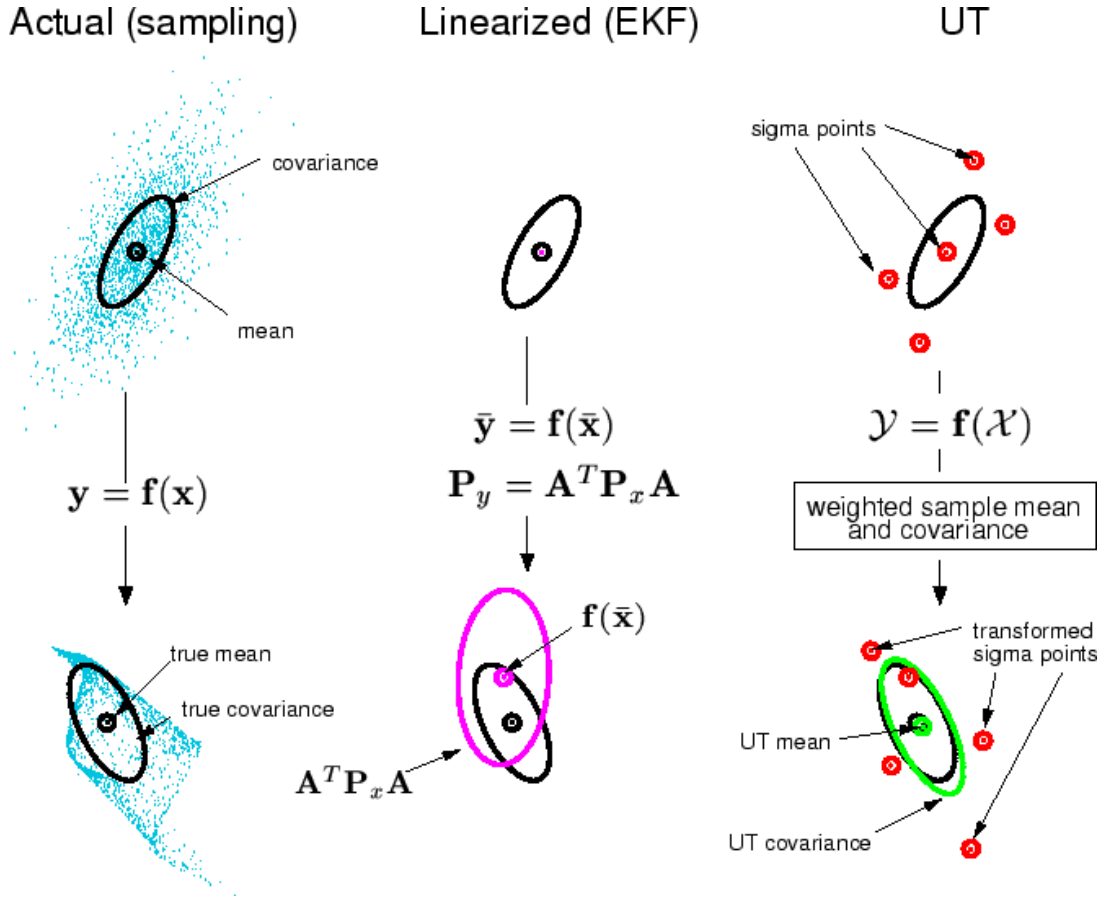
$$\begin{aligned} \kappa &\geq 0 \\ \alpha &\in (0, 1) \\ \lambda &= \alpha^2 (n + \kappa) - n \\ \beta &= 2 \end{aligned} \quad (3.8)$$

3.1.2 Popis unscentovaného Kalmanova filtru

K problému odhadu orientace a polohy kvadrotorové helikoptéry přistoupíme jako k odhadu stavu nelineárního stochastického systému, který můžeme popsat jako (3.9) a (3.10), kde k je časový okamžik, x_k je nepozorovatelný stav, z_k je pozorovatelný (měřitelný) signál, který nám dává informaci o systému a u_k je známý vstup, kterým je systém buzen. w_k je vnitřní šum procesu, který popíšeme jako $w_k \sim \mathcal{N}(0, Q_k)$ (normální rozdělení pravděpodobnosti s nulovou střední hodnotou a kovariancí Q_k). v_k je šum měření, který popíšeme jako $v_k \sim \mathcal{N}(0, R_k)$.

¹Choleského dekompozice je v Matlabu zastoupena funkcí `chol(A)`, kde A je odmocňovaná matice

²viz několik implementací na http://rosettacode.org/wiki/Cholesky_decomposition



Obrázek 3.2: Ukázka unscentované transformace střední hodnoty a kovariance, zleva doprava: skutečná stř. hodnota a kovariance, linearizovaná (používáno v rozšířeném Kalmanově filtru), unscentovaná transformace. Zdroj: [4]

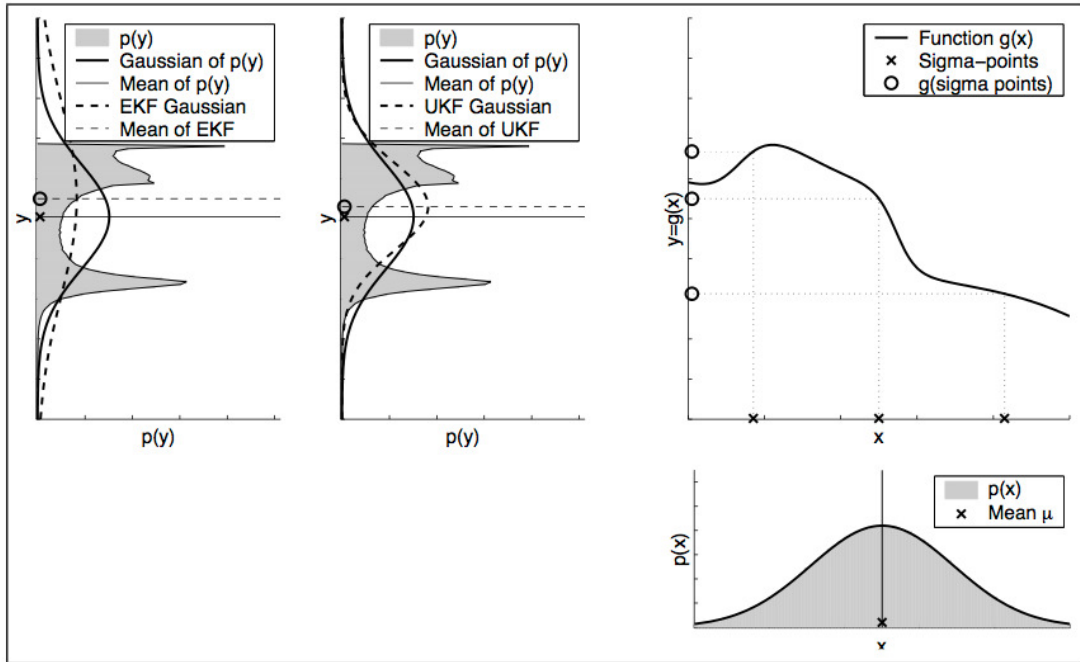
$$x_{k+1} = g_k(u_k, x_k) + w_k, \quad k = 0, 1, 2, \dots \quad (3.9)$$

$$z_k = h_k(x_k) + v_k, \quad k = 0, 1, 2, \dots \quad (3.10)$$

Pro odhad stavu systému použijeme estimační algoritmus. Tento algoritmus vychází ze standardního Kalmanova filtru. Kalmanův filtr provádí rekurzivní výpočet filtrovaných a predikovaných středních hodnot a kovariančních matic. Vzhledem k tomu, že algoritmus bude provádět odhad stavu nelineárního systému, zvolíme unscentovaný Kalmanův filtr [6, 15, 14, 3]. UKF využívá k distribuci střední hodnoty a kovariance unscentovanou transformaci 3.1.1. Porovnání UKF s rozšířeným Kalmanovým filtrem, jež se také používá k estimaci stavu nelineárních systémů, se nacházejí na obrázcích 3.2 a 3.2.

Nyní budou popsány jednotlivé kroky pro výpočet odhadu pomocí UKF. Pro start samotného algoritmu je třeba použít apriorní hustotu pravděpodobnosti, která je popsána pomocí střední hodnoty a kovariance v (3.11), kde \bar{x}_0 je apriorní střední hodnota a P_0 je apriorní kovariance.

$$k = 0 : \quad \hat{x}_{0|-1} = E[x_0] = \bar{x}_0, \quad P_{0|-1} = \text{cov}[x_0] = P_0 \quad (3.11)$$



Obrázek 3.3: Porovnání odhadu střední hodnoty a kovariance pomocí rozšířeného Kalmanova filtru (EKF) a unscentovaného Kalmanova filtru (UKF). Zdroj: [14]

Nyní můžeme vypočítat filtrační odhad na základě měření z_k . Výsledný odhad střední hodnoty označíme $\hat{x}_{k|k}$ a odhad kovariance $P_{k|k}$. Postup výpočtu je popsán v (3.12), kde $X_{k|k-1}$ jsou sigma body, které jsou vypočítány na základě apriorního odhadu (případně v dalším kroku na základě predikce), $Z_{k|k-1}$ jsou sigma body které byly distribuované skrz nelineární funkci $h(\cdot)$. \hat{z} je predikované měření a S_k je kovariance inovace³. R_k je předpokládaná kovariance gaussovského šumu s nulovou střední hodnotou, který se projevuje nepřesností měření (nepřesností funkce $h(\cdot)$). $P_{k|k-1}^{x,y}$ je vzájemná kovarianční matice a K_k je Kalmanův zisk. Po výpočtu filtračního odhadu můžeme provést predikci.

³chyba predikce měření

Filtrace:

$$\begin{aligned}
X_{k|k-1} &= \left(\hat{x}_{k|k-1} \quad \hat{x}_{k|k-1} + \gamma \sqrt{P_{k|k-1}} \quad \hat{x}_{k|k-1} - \gamma \sqrt{P_{k|k-1}} \right) \\
Z_{k|k-1} &= h \left(X_{k|k-1} \right) \\
\hat{z}_{k|k-1} &= \sum_{i=0}^{2n} w_m^{[i]} Z_{k|k-1}^{[i]} \\
S_k &= \sum_{i=0}^{2n} w_c^{[i]} \left(Z_{k|k-1}^{[i]} - \hat{z}_{k|k-1} \right) \left(Z_{k|k-1}^{[i]} - \hat{z}_{k|k-1} \right)^T + R_k \\
P_{k|k-1}^{x,z} &= \sum_{i=0}^{2n} w_c^{[i]} \left(\bar{X}_{k|k-1}^{[i]} - \hat{x}_{k|k-1} \right) \left(Z_{k|k-1}^{[i]} - \hat{z}_{k|k-1} \right)^T \\
K_k &= P_{k|k-1}^{x,z} S_k^{-1} \\
\hat{x}_{k|k} &= \hat{x}_{k|k-1} + K_k \left(z_k - \hat{z}_{k|k-1} \right) \\
P_{k|k} &= P_k - K_k S_k K_k^T
\end{aligned} \tag{3.12}$$

Predikovanou střední hodnotu označíme $\hat{x}_{k|k-1}$ a kovarianci $\bar{P}_{k|k-1}$. Výpočet je popsán v (3.13), kde γ je parametr pro výpočet sigma bodů (z části 3.1.1 $\gamma = \sqrt{n + \lambda}$, $X_{k-1|k-1}$ jsou vypočítané sigma body, $X_{k|k-1}^*$ jsou sigma body distribuované skrze nelineární funkci $g(\cdot, \cdot)$ a u_k je vstup nelineárního systému v kroku k . Po provedení výpočtu predikce získáme odhad střední hodnoty a kovariance, navýšíme krok algoritmu $k = k + 1$ a budeme pokračovat výpočtem filtračního odhadu.

Predikce:

$$\begin{aligned}
X_{k-1|k-1} &= \left(\hat{x}_{k-1|k-1} \quad \hat{x}_{k-1|k-1} + \gamma \sqrt{P_{k-1|k-1}} \quad \hat{x}_{k-1|k-1} - \gamma \sqrt{P_{k-1|k-1}} \right) \\
X_{k|k-1}^* &= g(u_k, X_{k|k-1}) \\
\hat{x}_{k|k-1} &= \sum_{i=0}^{2n} w_m^{[i]} X_{k|k-1}^{*[i]} \\
P_{k|k-1} &= \sum_{i=0}^{2n} w_c^{[i]} \left(X_{k|k-1}^{*[i]} - \hat{x}_{k|k-1} \right) \left(X_{k|k-1}^{*[i]} - \hat{x}_{k|k-1} \right)^T + Q_k
\end{aligned} \tag{3.13}$$

Implementace algoritmu UKF v *Matlabu* je popsána v A.1.

3.2 Diskrétní nelineární model kvadrotorové helikoptéry

Model kvadrotorové helikoptéry, který bude diskrétní v čase, bude vytvořen na základě nelineárního modelu z části 2. Diskretizaci provedeme pomocí Taylorovy metody [9]. Taylorova metoda pracuje s tvarovačem 0. řádu.

3.2.1 Diskretizace nelineárního modelu

Diskretizace je provedena za předpokladu, že na systém působí vstup, který je po částech konstantní (vždy se změní v diskrétním čase. Než bude uveden postup diskretizace, je třeba definovat obecný model. Spojitý model bude popsán ve formě stavového popisu (3.14), kde $x(t)$ je stav systému v čase t , $u(t)$ je vstup systému v čase t , který je do dalšího časového okamžiku konstantní, $f(\cdot)$ je nelineární funkce, která popisuje vnitřní dynamiku systému a $g(\cdot)$ je nelineární funkce, která popisuje, jak působí vstup systému $u(t)$ na změnu stavu $x(t)$.

$$\frac{dx(t)}{dt} = f(x(t)) + g(x(t))u(t) \quad (3.14)$$

Diskretizace systému je popsána pomocí vztahu (3.15), kde $x(k)$ je stav v časovém okamžiku $t = t_k = kT_{sd}$, T_{sd} je vzorkovací perioda modelu a $A^{[j]}$ je rekurzivně popsáno pomocí (3.16) a (3.17), kde $\frac{\partial \cdot}{\partial x}$ je parciální derivace podle stavu x .

$$x(k+1) = x(k) + \sum_{j=1}^{\infty} \frac{T_{sd}^j}{j!} \left. \frac{d^j x}{dt^j} \right|_{t_k} = x(k) + \sum_{j=1}^{\infty} A^{[j]}(x(k), u(k)) \frac{T_{sd}^j}{j!} \quad (3.15)$$

$$A^{[j]}(x, u) = f(x) + ug(x) \quad (3.16)$$

$$A^{[j+1]}(x, u) = \frac{\partial A^{[j]}(x, u)}{\partial x} (f(x) + ug(x)) \quad (3.17)$$

Finální vztah pro diskretizaci je popsán v (3.18) pro maximální stupeň rozvoje Taylorovy řady N_d .

$$x(k+1) = x(k) + \sum_{j=1}^{N_d} A^{[j]}(x(k), u(k)) \frac{T_{sd}^j}{j!} \quad (3.18)$$

3.2.2 Výpočet diskrétního nelineárního modelu kvadrotorové helikoptéry

Zde vytvořený model bude využit pro predikci chování kvadrotorové helikoptéry v UKF. Nejprve je třeba upravit model z části 2 do stavového popisu. Pro zjednodušení modelu zanedbáme v modelu odpor prostředí a třepotání listů rotorů (pro zrychlení výpočtu nového stavu). Vektor stavu kvadrotorové helikoptéry a řízení jsou popsány v (3.19) a spojitý stavový popis je popsán v (3.20).

$$\begin{aligned}
\begin{bmatrix} x_1(t) \\ x_3(t) \\ x_5(t) \end{bmatrix} &= r(t), \quad \begin{bmatrix} x_2(t) \\ x_4(t) \\ x_6(t) \end{bmatrix} = \begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_3(t) \\ \dot{x}_5(t) \end{bmatrix}, \quad \begin{bmatrix} x_7(t) \\ x_9(t) \\ x_{11}(t) \end{bmatrix} = \Omega(t), \quad \begin{bmatrix} x_8(t) \\ x_{10}(t) \\ x_{12}(t) \end{bmatrix} = \begin{bmatrix} \dot{x}_7(t) \\ \dot{x}_9(t) \\ \dot{x}_{11}(t) \end{bmatrix}, \\
\begin{bmatrix} u_1(t) \\ u_2(t) \\ u_3(t) \end{bmatrix} &= e^{[x_7, x_9, x_{11}]^T} \cdot \begin{bmatrix} 0 \\ 0 \\ T(t) \end{bmatrix}, \quad \begin{bmatrix} u_4 \\ u_5 \\ u_6 \end{bmatrix} = \begin{bmatrix} M_\phi(t) \\ M_\theta(t) \\ M_\psi(t) \end{bmatrix}
\end{aligned} \tag{3.19}$$

$$\dot{x}(t) = \begin{bmatrix} x_2(t) \\ 0 \\ x_4(t) \\ 0 \\ x_6(t) \\ -g \\ x_8(t) \\ x_{10}(t) \cdot x_{12}(t) \cdot \frac{I_y - I_z}{I_x} \\ x_{10}(t) \\ x_8(t) \cdot x_{12}(t) \cdot \frac{I_x - I_z}{I_y} \\ x_{12}(t) \\ x_8(t) \cdot x_{10}(t) \cdot \frac{I_x - I_y}{I_z} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{u_1(t)}{m} \\ 0 \\ \frac{u_2(t)}{m} \\ 0 \\ \frac{u_3(t)}{m} \\ 0 \\ \frac{u_4(t)}{I_x} \\ 0 \\ \frac{u_5(t)}{I_y} \\ 0 \\ \frac{u_6(t)}{I_z} \end{bmatrix} \tag{3.20}$$

Nyní můžeme provést diskretizaci dle (3.18). Maximální stupeň rozvoje Taylorovy řady zvolíme $N_d = 2$. Získaný stavový popis je zapsán v (3.21), kde T_{s_d} je perioda diskrétního modelu kvadrotorové helikoptéry.

$$x_{k+1} = \begin{bmatrix}
x_{1,k} + x_{2,k}T_{sd} + \frac{u_{1,k}T_{sd}^2}{2m} \\
x_{2,k} + \frac{u_{1,k}T_{sd}}{m} \\
x_{3,k} + x_{4,k}T_{sd} + \frac{u_{2,k}T_{sd}^2}{2m} \\
x_{4,k} + \frac{u_{2,k}T_{sd}}{m} \\
x_{5,k} + x_{5,k}T_{sd} + \frac{u_{3,k}T_{sd}^2}{2m} \\
x_{6,k} + \frac{u_{3,k}T_{sd}}{m} \\
x_{7,k} + T_{sd}x_{8,k} + \frac{T_{sd}^2}{2I_x} (u_{4,k} + x_{10,k}x_{12,k} (I_y - I_z)) \\
x_{8,k} + \frac{T_{sd}^2 (I_y - I_z)}{2I_x} \left[\frac{x_{12,k}}{I_y} (u_{5,k} + x_{8,k}x_{12,k} (I_x - I_z)) \dots \right. \\
\left. \dots + \frac{x_{10,k}}{I_z} (u_{6,k} + x_{8,k}x_{10,k} (I_x - I_y)) \right] \dots \\
\dots + \frac{T_{sd}}{I_x} (u_{4,k} + x_{10,k}x_{12,k} (I_y - I_z)) \\
x_{9,k} + T_{sd}x_{10,k} + \frac{T_{sd}^2}{2I_y} (u_{5,k} + x_{8,k}x_{12,k} (I_x - I_z)) \\
x_{10,k} + \frac{T_{sd}^2 (I_x - I_z)}{2I_y} \left[\frac{x_{12,k}}{I_x} (u_{4,k} + x_{10,k}x_{12,k} (I_y - I_z)) \dots \right. \\
\left. \dots + \frac{x_{8,k}}{I_z} (u_{6,k} + x_{8,k}x_{10,k} (I_x - I_y)) \right] \dots \\
\dots + \frac{T_{sd}}{I_y} (u_{5,k} + x_{8,k}x_{12,k} (I_x - I_z)) \\
x_{11,k} + T_{sd}x_{12,k} + \frac{T_{sd}^2}{2I_z} (u_{6,k} + x_{8,k}x_{10,k} (I_x - I_y)) \\
x_{12,k} + \frac{T_{sd}^2 (I_x - I_y)}{2I_z} \left[\frac{x_{10,k}}{I_x} (u_{4,k} + x_{10,k}x_{12,k} (I_y - I_z)) \dots \right. \\
\left. \dots + \frac{x_{8,k}}{I_y} (u_{5,k} + x_{8,k}x_{12,k} (I_x - I_z)) \right] \dots \\
\dots + \frac{T_{sd}}{I_z} (u_{6,k} + x_{8,k}x_{10,k} (I_x - I_y))
\end{bmatrix} \quad (3.21)$$

Jelikož estimace stavu bude prováděna na základě vstupu ve formě požadovaných otáček rotorů, je třeba doplnit potřebné přepočty (3.22). Tah a momenty jsou popsány pro časový okamžik k . V popisu momentu síly $M_{\psi,k}$ je obsažena i setrvačnost rotoru, pro jejíž výpočet je potřeba znát i otáčky rotorů z předchozího kroku. Z tohoto důvodu je třeba funkci $g(u_k, X_k)$ rozšířit na $g(u_k, u_{k-1}, X_k)$. Implementace diskrétního modelu jako sady funkcí v *Matlabu* je popsána v A.2.

$$T_k = k_T \cdot \sum_{i=1}^4 \omega_{i,k}^2 \quad (3.22)$$

$$M_{\phi,k} = l \cdot k_T \cdot (-\omega_{2,k}^2 + \omega_{4,k}^2) \quad (3.23)$$

$$M_{\theta,k} = l \cdot k_T \cdot (-\omega_{1,k}^2 + \omega_{3,k}^2) \quad (3.24)$$

$$M_{\psi,k} = \sum_{i=1}^4 (-1)^{i+1} \left[k_M \cdot \omega_{i,k}^2 + \frac{I_m}{T_{sd}} (\omega_{i,k} - \omega_{i,k-1}) \right] \quad (3.25)$$

3.3 Senzory

K odhadu stavu pomocí Kalmanova filtru je třeba mít měření, které nám dává nějakou informaci o reálném stavu objektu, jehož stav určíme. V oblasti multirotorových helikoptér je informace získávána z měření inerciální měřící jednotky⁴[12]. V této části bude popsáno jakým způsobem měření z IMU připravit pro UKF a také bude popsána funkce $h(\cdot)$, která je použita v UKF a slouží k predikci měření.

3.3.1 Inerciální měřící jednotka

Inerciální měřící jednotka se skládá v základu z tříosého gyroskopu a tříosého akcelerometru. Gyroskop je zařízení, které je schopné měřit úhlovou rychlost objektu, na kterém je umístěno. Oproti tomu akcelerometr měří zrychlení objektu. V měření akcelerometru je obsaženo i tzv. statické zrychlení, které je způsobeno gravitační silou Země (při klidovém stavu je v měření ve třech osách rozloženo gravitační zrychlení). Statické zrychlení je potřeba pro měření zrychlení objektu buď odfiltrovat a nebo na základě znalosti velikosti gravitačního zrychlení a orientace objektu vůči gravitačnímu poli odečíst.

Na základě měření z akcelerometru a gyroskopu lze provádět stabilizační řízení kvadrotorové helikoptéry, pro řízení polohy je třeba ale tyto dva senzory doplnit ještě o modul GPS⁵ a magnetometr, s jehož pomocí je možné při využití informace o natočení kolem osy x' a y' zjistit natočení helikoptéry oproti magnetickému pólu a používat jej jako digitální kompas. Pro zjednodušení budeme předpokládat, že senzory jsou zkalibrovány pro použití na kvadrotorové helikoptěře.

Měření polohy, rychlosti

Nejprve popíšeme měření polohy a rychlosti kvadrotorové helikoptéry. V (3.26) je popsáno, jakým způsobem vykompenzovat statické zesílení, transformace měření ze souřadného systému helikoptéry do inerciálním souřadném systému a následné převedení zrychlení z g do $m \cdot s^{-2}$. acc_{raw_i} je tedy měřené zrychlení v g na jednotlivých osách, které obsahuje i statické zrychlení, argument rotační matice je složen z předchozího odhadu eulerových úhlů a g je gravitační zrychlení. Zpracované měření akcelerometru je označeno acc .

⁴v angličtině označována jako IMU - inertial measurement unit

⁵Global Positioning System - globální polohovací systém

$$acc = \begin{bmatrix} acc_x \\ acc_y \\ acc_z \end{bmatrix} = g \cdot \left(e^{[x_{7,k} \quad x_{9,k} \quad x_{11,k}]_x} \cdot \begin{bmatrix} acc_{raw_x} \\ acc_{raw_y} \\ acc_{raw_z} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \right) \quad (3.26)$$

Dále si uvedeme, jakým způsobem bude upraveno měření polohy a rychlosti na základě známého zrychlení z akcelerometru a předchozího odhadu stavu. V (3.27) je popsáno nové měření rychlosti v kroku k , které je vypočítáno pomocí předchozího odhadu rychlosti a měření zrychlení. V (3.28) je nové měření polohy v kroku k , které vychází z předchozího odhadu polohy a z měření rychlosti. Ve vztazích se objevuje perioda T_{sd} , což je perioda diskrétního nelineárního modelu z 3.2 a také perioda spouštění uncentrovaného Kalmanova filtru.

$$\begin{aligned} \dot{x}_{acc,k} &= x_{2,k} + T_{sd} \cdot acc_x \\ \dot{y}_{acc,k} &= x_{4,k} + T_{sd} \cdot acc_y \\ \dot{z}_{acc,k} &= x_{6,k} + T_{sd} \cdot acc_z \end{aligned} \quad (3.27)$$

$$\begin{aligned} x_{acc,k} &= x_{1,k} + T_{sd} \cdot \dot{x}_{acc,k} \\ y_{acc,k} &= x_{3,k} + T_{sd} \cdot \dot{y}_{acc,k} \\ z_{acc,k} &= x_{5,k} + T_{sd} \cdot \dot{z}_{acc,k} \end{aligned} \quad (3.28)$$

Nyní bude popsáno měření polohy a rychlosti v kroku k na základě měření z modulu GPS. Měření polohy je popsáno v (3.29), kde gps_i je měření polohy v dané ose, a měření rychlosti je popsáno v (3.30), kde je opět využito znalosti předchozího odhadu polohy.

$$\begin{aligned} x_{gps,k} &= gps_x \\ y_{gps,k} &= gps_y \\ z_{gps,k} &= gps_z \end{aligned} \quad (3.29)$$

$$\begin{aligned} \dot{x}_{gps,k} &= \frac{1}{T_{sd}} (gps_x - x_{1,k}) \\ \dot{y}_{gps,k} &= \frac{1}{T_{sd}} (gps_y - x_{3,k}) \\ \dot{z}_{gps,k} &= \frac{1}{T_{sd}} (gps_z - x_{5,k}) \end{aligned} \quad (3.30)$$

Měření úhlové rychlosti, eulerových úhlů

V této části bude popsáno měření úhlové rychlosti a eulerových úhlů. Měření bude prováděno pomocí magnetometru a gyroskopu. Nejprve popíšeme zpracování měření z gyroskopu. Měření eulerových úhlů (3.31) v kroku k je prováděno na základě předchozího

odhadu úhlů a gyroskopem měřené úhlové rychlosti. Měření úhlové rychlosti helikoptéry je popsáno v (3.32).

$$\begin{aligned}
\phi_{gyro,k} &= x_{7,k} + T_{sd} \cdot gyro_{\phi} \\
\theta_{gyro,k} &= x_{9,k} + T_{sd} \cdot gyro_{\theta} \\
\psi_{gyro,k} &= x_{11,k} + T_{sd} \cdot gyro_{\psi}
\end{aligned} \tag{3.31}$$

$$\begin{aligned}
\dot{\phi}_{gyro,k} &= gyro_{\phi} \\
\dot{\theta}_{gyro,k} &= gyro_{\theta} \\
\dot{\psi}_{gyro,k} &= gyro_{\psi}
\end{aligned} \tag{3.32}$$

V (3.33) je popsáno měření magnetického pole tříosím magnetometrem (aktuální hodnoty měření jsou mag_i) a výpočet úhlu ψ (rotace helikoptéry okolo osy z'). K výpočtu jsou využity také předchozí odhady úhlů ϕ a θ , tedy $x_{7,k}$ a $x_{9,k}$. Měření $\psi_{mag,k}$ je vypočítáno pomocí funkce arkus tangens (v prostředí *Matlab* a ve většině programovacích jazyků je funkce implementována jako $atan2(\cdot, \cdot)$).

$$\begin{aligned}
x_h &= mag_x \cdot \cos x_{7,k} + mag_y \cdot \sin x_{9,k} \cdot \sin x_{7,k} + mag_z \cdot \sin x_{9,k} \cos x_{7,k} \\
y_h &= mag_x \cdot \cos x_{7,k} + mag_z \sin x_{7,k} \\
\psi_{mag,k} &= \arctan(-y_h, x_h)
\end{aligned} \tag{3.33}$$

Na základě měření mag. pole magnetometrem a předchozího odhadu eulerových úhlů je vypočítána i úhlová rychlost $\dot{\psi}_{mag,k}$ v kroku k . $x_{12,k}$ je upraven pomocí funkce $limit(\cdot)$. Tato funkce je použita, jelikož výpočet $\psi_{mag,k}$ je omezen oborem hodnot funkce arkus tangens, který je roven $H_{\arctan} \in \langle -\pi, \pi \rangle$. Pseudokód, který popisuje funkci se nachází v (3.35)⁶, kde $mod(\cdot, 2\pi)$ vrací zbytek po dělení číslem 2π .

$$\begin{aligned}
x_h &= mag_x \cdot \cos x_{7,k} + mag_y \cdot \sin x_{9,k} \cdot \sin x_{7,k} + mag_z \cdot \sin x_{9,k} \cos x_{7,k} \\
y_h &= mag_x \cdot \cos x_{7,k} + mag_z \sin x_{7,k} \\
\dot{\psi}_{mag,k} &= \frac{1}{T_{sd}} (\arctan(-y_h, x_h) - limit(x_{12,k}))
\end{aligned} \tag{3.34}$$

$$\begin{aligned}
c &= mod(x_{11,k|k-1}, 2\pi) \\
\text{if } c &> \pi \\
c &= c - 2\pi \\
\text{Return } c
\end{aligned} \tag{3.35}$$

⁶v prostředí *Matlab* lze použít funkci $wrapToPi(\cdot)$

3.3.2 Měření pro unscentovaný Kalmanův filtr

Vektor měření v kroku k , který je využit v UKF k filtračnímu odhadu a je značen z_k , bude složen z měření, která byla popsána v části 3.3.1. Vektor z_k je popsán v (3.37).

$$z_k = \begin{bmatrix} x_{gps,k}, y_{gps,k}, z_{gps,k}, x_{acc,k}, y_{acc,k}, z_{acc,k}, \dot{x}_{acc,k}, \dot{y}_{acc,k}, \dot{z}_{acc,k}, \\ \dot{x}_{gps,k}, \dot{y}_{gps,k}, \dot{z}_{gps,k}, \phi_{gyro,k}, \theta_{gyro,k}, \psi_{gyro,k}, \\ \dot{\phi}_{gyro,k}, \dot{\theta}_{gyro,k}, \dot{\psi}_{gyro,k}, \psi_{mag,k}, \psi_{mag,k} \end{bmatrix}^T \quad (3.36)$$

3.3.3 Úprava stavu kvadrotorové helikoptéry pro predikci měření

V této části bude popsáno, jakým způsobem transformovat sigma body pro predikci měření. Tato transformace je v algoritmu UKF zastoupena v (3.12) funkcí $h(\cdot)$. Funkce má za úkol vypočítat na základě odhadovaného stavu odpovídající měření. Výstup funkce $h(\cdot)$ označíme jako $z_{k|k-1}$ a vstup funkce jako $x_{k|k-1}$. Poslední dva prvky $z_{k|k-1}$ musíme opět upravit pomocí funkce $limit(\cdot)$, aby byl prvek omezen stejným oborem hodnot jako v (3.33) a (3.34).

$$z_{k|k-1} = \begin{bmatrix} x_{1,k|k-1}, x_{3,k|k-1}, x_{5,k|k-1}, x_{2,k|k-1}, x_{4,k|k-1}, x_{6,k|k-1}, x_{1,k|k-1}, x_{3,k|k-1}, x_{5,k|k-1}, \\ x_{2,k|k-1}, x_{4,k|k-1}, x_{6,k|k-1}, x_{7,k|k-1}, x_{9,k|k-1}, x_{11,k|k-1}, \\ x_{8,k|k-1}, x_{10,k|k-1}, x_{12,k|k-1}, limit(x_{11,k|k-1}), limit(x_{12,k|k-1}) \end{bmatrix}^T \quad (3.37)$$

4 Návrh řídicího algoritmu kvadrotorové helikoptéry pro sledování zadané trajektorie

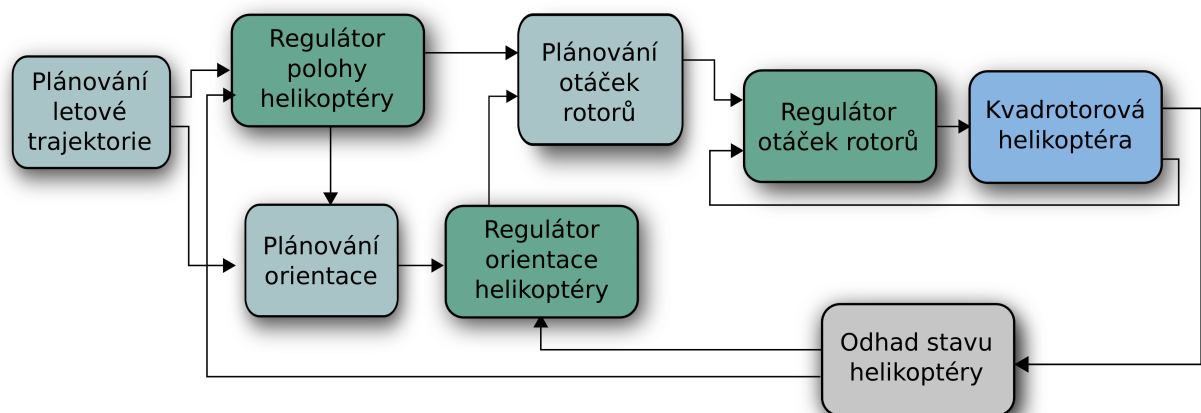
V této části bude navržen regulátor, který bude mít za úkol generovat takové řízení, jehož výsledkem bude plně autonomní let kvadrotorové helikoptéry podle zadané trasy. Regulátor lze rozdělit do několika úrovní, které bude možné zpracovat odděleně.

Tyto části jsou regulátor otáček motorů, orientace helikoptéry, polohy helikoptéry a algoritmus generující letovou trajektorii dle zadaných bodů (viz obrázek 4.1). Dekompozice regulátoru na samostatné bloky je výhodná zejména z hlediska jednoduchého návrhu, testování, ale i možnosti mít pro různé regulační smyčky různé periody pro generování řízení.

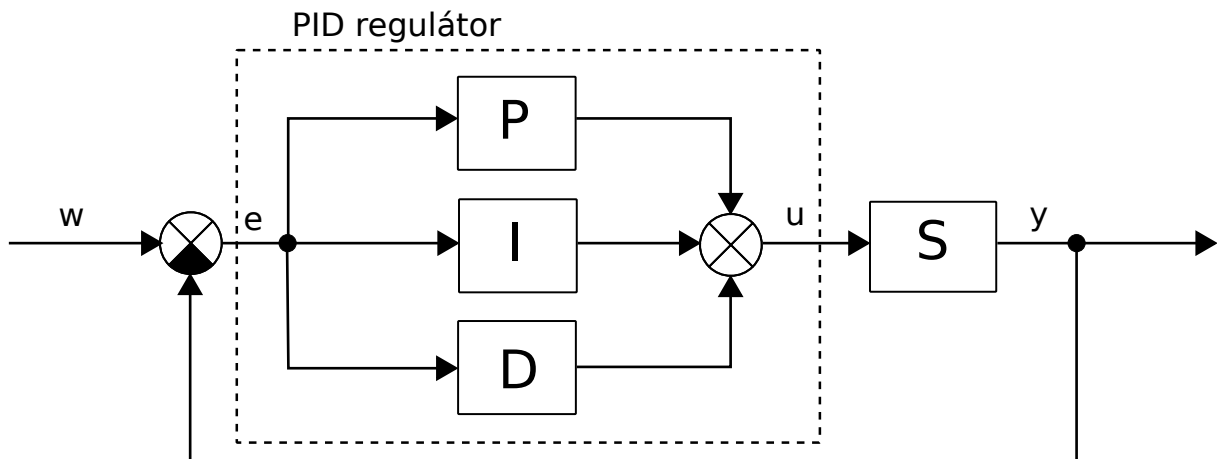
4.1 Regulátor otáček rotorů

Regulátor otáček rotoru je součástí nejnižší regulační smyčky a bude navržen pro soustavu BLDC motoru¹ zatíženého na hřídeli rotorem. Měřený výstup systému, otáčky i -tého motoru, je v jednotkách RPM ($\text{ot} \cdot \text{min}^{-1}$). Budeme předpokládat, že je helikoptéra osazena jedním typem motoru, jedním typem rotoru (s rozdílem, že jeden pár je konstruován

¹*brushless direct current motor* - bezkartáčový stejnosměrný motor



Obrázek 4.1: Regulační schéma kvadrotorové helikoptéry



Obrázek 4.2: Schéma regulačního smyčky s regulovaným systémem S a paralelním PID regulátorem (w - požadovaná hodnota, e - regulační odchylka, u - řízení, y - výstup)

pro otáčení v opačném směru) a jednotlivé kusy se od sebe neliší. Na základě tohoto předpokladu nebude nutné navrhovat regulátor pro každý motor zvlášť.

PID regulátor

Pro regulaci motorů zvolíme paralelní (neinteraktivní) PID regulátor [13]. Nejprve si uvedeme schéma 4.2 a vztah (4.1) pro ideální regulátor, kde K je zesílení regulátoru, T_i je integrační časová konstanta a T_d je derivační časová konstanta. Regulátor motorů bude mírně odlišný.

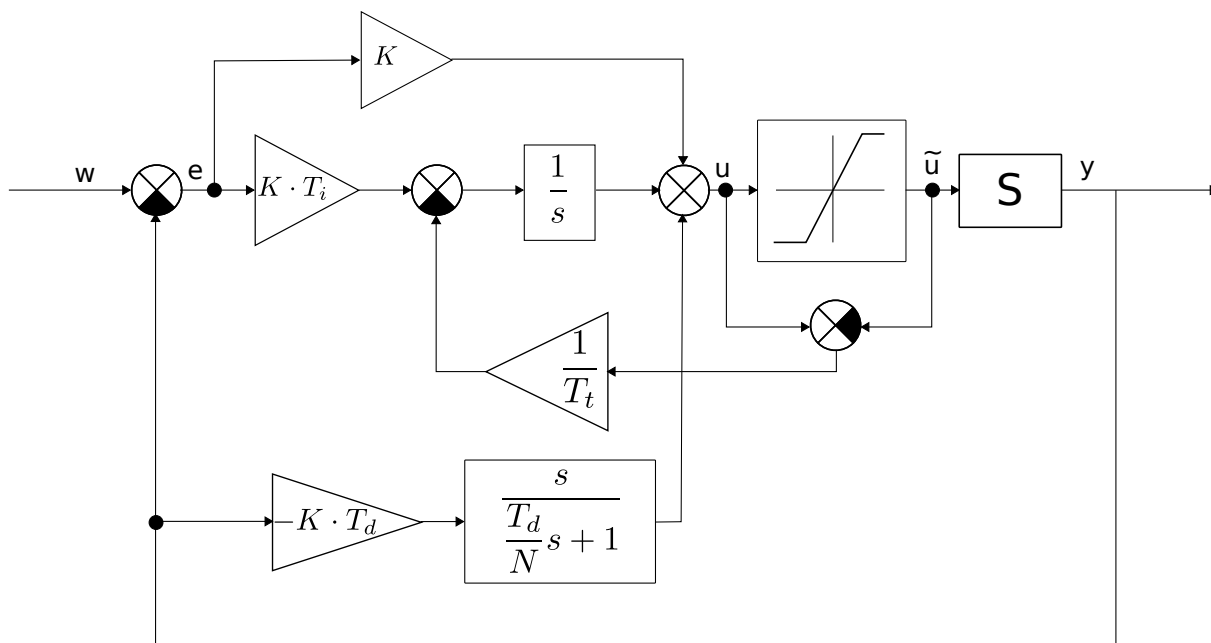
$$\begin{aligned} e(t) &= w(t) - y(t) \\ u(t) &= K \cdot \left[e(t) + \frac{1}{T_i} \int_0^t e(t) dt + T_d \frac{de(t)}{dt} \right] \end{aligned} \quad (4.1)$$

Vzhledem k tomu, že jsou otáčky motoru omezeny shora (maximální otáčky, kterých může zatížený motor dosáhnout) i zdola (pokud by klesl počet otáček pod určitou mez, mohl by se motor zastavit), bude výhodné zabránit unášení integrační složky (tzv. „anti-windup“ PID). Mechanismus, který toto umožňuje, funguje na principu porovnávání manipulované veličiny u (výstup regulátoru) s omezenou manipulovanou veličinou \tilde{u} (po oříznutí výstupu regulátoru) a následném odečtení váženého rozdílu od vstupu integrační složky.

Další změna vůči klasickému PID regulátoru bude ta, že derivační složka nebude mít na vstupu přímo regulační odchylku e , ale měřenou procesní hodnotu y (v našem případě měřené otáčky motoru). Tak zamezíme velké derivaci při prudké změně požadované hodnoty w (požadované otáčky motoru). Ze stejného důvodu bude derivační složka také filtrována.

Sestavení PID regulátoru pro regulaci otáček motoru

Výsledné vztahy pro spojitý regulátor jsou uvedeny v (4.2) a jeho schéma je znázorněno na obrázku 4.3. Ve vztazích (4.2) je uveden Laplaceův obraz [10, kap. 2] popisu v čase.



Obrázek 4.3: Schéma regulačního smyčky s regulovaným systémem S a PID regulátorem (w - požadovaná hodnota, e - regulační odchylka, u - řízení, \tilde{u} - omezené řízení, y - výstup)

Symbolem s je značen argument Laplaceovy transformace. Rovnice jsou rozděleny na čtyři části. $U_P(s)$ popisuje výpočet proporcionální složky regulátoru, $U_I(s)$ výpočet integrální složky a $U_D(s)$ derivační složky. Poslední vztah slouží k výpočtu výstupu regulátoru. Parametr N ovlivňuje míru filtrace derivační složky a parametr T_t je časová konstanta, která ovlivňuje rychlost zabránění unášení integrační složky.

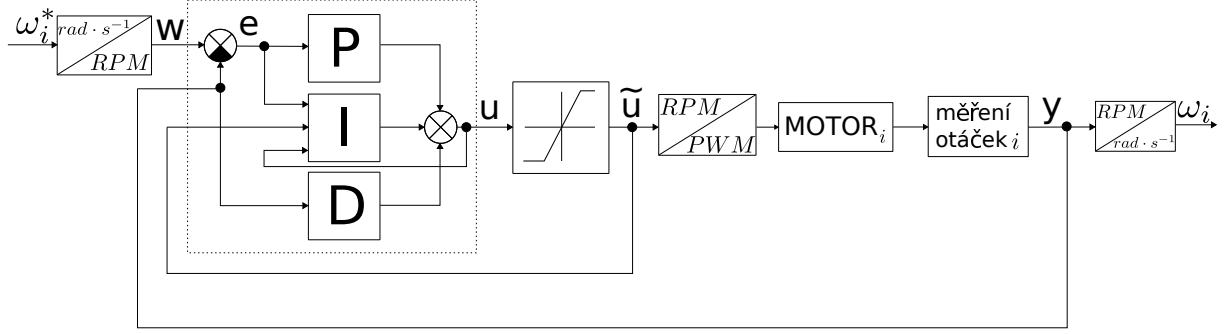
$$\begin{aligned}
 U_P(s) &= K \cdot E(s) \\
 U_I(s) &= \left(K \cdot T_i - \frac{U(s) - \tilde{U}(s)}{T_t} \right) \cdot \frac{1}{s} \\
 U_D(s) &= -\frac{K \cdot T_d \cdot s}{\frac{T_d}{N} \cdot s + 1} Y(s) \\
 U(s) &= U_P(s) + U_I(s) + U_D(s)
 \end{aligned} \tag{4.2}$$

Pro použití v palubním počítači helikoptéry převedeme regulátor na regulátor diskretní v čase. Diskretizace vztahů (4.2) pro derivační člen a mechanismus proti unášení integrační složky bude provedena jednodušší obdélníkovou aproximací

$$s = \frac{1}{T_{sm}} \cdot (1 - z^{-1}),$$

kde z je argument Z-transformace a T_{sm} je zvolená vzorkovací frekvence, a integrační člen zdiskretizujeme lichoběžníkovou (Tustinovou) aproximací

$$s = \frac{2}{T_{sm}} \cdot \frac{1 - z^{-1}}{1 + z^{-1}}.$$



Obrázek 4.4: Schéma zapojení regulátoru i -tého motoru do řídicího systému

Výsledný tvar diskrétního PID regulátoru s filtrovanou derivační složkou a zabraňující uvažování integrační složky je popsán pomocí diferenčních rovnic (4.3).

$$\begin{aligned}
 u_P(t) &= K \cdot e(t) \\
 u_I(t) &= u_I(t-1) + \frac{K \cdot T_i \cdot T_{sm}}{2} [e(t) + e(t-1)] + \frac{T_{sm}}{T_t} \cdot [\tilde{u}(t-1) - u(t-1)] \\
 u_D(t) &= \frac{T_d}{T_d + N \cdot T_{sm}} \cdot [u_D(t-1) + K \cdot N \cdot (y(t-1) - y(t))] \\
 u(t) &= u_P(t) + u_I(t) + u_D(t)
 \end{aligned} \tag{4.3}$$

Zapojení regulátoru otáček motoru do regulační smyčky

Nyní specifikujeme zapojení regulátoru otáček motoru v řídicím systému kvadrotorové helikoptéry (viz obrázek 4.4). Nejprve jsou přivedeny požadované otáčky i -tého motoru ω_i^* , dále je ω_i^* převedeno z $\text{rad}\cdot\text{s}^{-1}$ na RPM ($\text{ot}\cdot\text{min}^{-1}$) a je přivedeno do regulátoru jako požadovaná hodnota w ($w = \frac{\omega_i^*}{2\pi} \cdot 60$). Dále je na vstup regulátoru přivedeno měření otáček y a omezené řízení \tilde{u} . Dle vztahů (4.3) je vygenerováno řízení u , které je omezeno maximem a minimem možných otáček na \tilde{u} . \tilde{u} je převedeno pomocí převodní funkce na signál PWM² pro motor i . Převodní funkci je třeba získat na základě závislosti PWM/RPM.

Dále jsou měřeny otáčky motoru, které jsou jako výstupní proměnná y odesílány na vstup regulátoru a také převáděny z RPM na $\text{rad}\cdot\text{s}^{-1}$ pro vyšší vrstvu řídicího systému.

4.2 Plánování otáček rotorů

V této části bude popsán vztah, který bude sloužit k přepočtu z požadovaného tahu a momentů na požadované otáčky. Přes tento vztah bude možné působit regulátory polohy a orientace, které budou popsány v dalších částech, na regulátory, které regulují otáčky motoru.

Síly a momenty síly, které působí na helikoptéru, jsme popsali v části a . Pro přepočet budou použity momenty M_ϕ , M_θ a M_ψ a síla T , tedy celkový tah. Ostatní síly a momenty nejsou přímo závislé na otáčkách rotorů a mohou být kompenzovány např. v regulátoru

²Pulse Width Modulation - pulzně šířková modulace

polohy a orientace. Pro získání přepočtu bude také ze vztahu pro M_ψ (2.15) vypuštěna složka $I_M \cdot \dot{\omega}_i$. Nyní můžeme uveďte znovu vztahy pro výpočet tahu a momentů na základě otáček (4.4).

$$\begin{aligned} T &= k_T (\omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2) \\ M_\phi &= l \cdot k_T (-\omega_2^2 + \omega_4^2) \\ M_\theta &= l \cdot k_T (-\omega_1^2 + \omega_3^2) \\ M_\psi &= k_M (\omega_1^2 + \omega_2^2 + \omega_3^2 - \omega_4^2) \end{aligned} \quad (4.4)$$

Z (4.4) vidíme, že lze vztahy přepsat do formy matice (4.5).

$$\begin{bmatrix} T \\ M_\phi \\ M_\theta \\ M_\psi \end{bmatrix} = \begin{bmatrix} k_T & k_T & k_T & k_T \\ 0 & -l \cdot k_T & 0 & l \cdot k_T \\ -l \cdot k_T & 0 & l \cdot k_T & 0 \\ k_M & k_M & k_M & k_M \end{bmatrix} \cdot \begin{bmatrix} \omega_1^2 \\ \omega_2^2 \\ \omega_3^2 \\ \omega_4^2 \end{bmatrix} \quad (4.5)$$

Výsledný vztah (4.6) pro výpočet otáček na základě tahu a momentů získáme inverzí vztahu (4.5).

$$\begin{bmatrix} \omega_1^2 \\ \omega_2^2 \\ \omega_3^2 \\ \omega_4^2 \end{bmatrix} = \begin{bmatrix} \frac{1}{4 \cdot k_T} & 0 & -\frac{1}{2 \cdot l \cdot k_T} & \frac{1}{4 \cdot k_M} \\ \frac{1}{4 \cdot k_T} & \frac{1}{2 \cdot l \cdot k_T} & 0 & -\frac{1}{4 \cdot k_M} \\ \frac{1}{4 \cdot k_T} & 0 & \frac{1}{2 \cdot l \cdot k_T} & \frac{1}{4 \cdot k_M} \\ \frac{1}{4 \cdot k_T} & \frac{1}{2 \cdot l \cdot k_T} & 0 & -\frac{1}{4 \cdot k_M} \end{bmatrix} \cdot \begin{bmatrix} T \\ M_\phi \\ M_\theta \\ M_\psi \end{bmatrix} \quad (4.6)$$

Dále je potřeba poznamenat, že otáčky rotorů, které získáme výpočtem pomocí (4.6) jsou umocněné na druhou, takže pro výpočet požadovaných otáček je třeba výsledek odmocnit s tím, že pokud je hodnota ω_i^2 záporná, tak ji odmocnit jako kladnou, ale poté ji doplnit o záporné znaménko. Pokud by tato podmínka nebyla dodržena, tak by nebylo dosaženo požadovaného tahu a momentů.

4.3 Regulátor orientace kvadrotorové helikoptéry

V této části bude navrženo řízení orientace kvadrotorové helikoptéry. Výsledkem bude regulátor, jež dokáže regulovat pohyb helikoptéry kolem Eulerových úhlů, které jsou označeny na obrázku 2.1. Základem řízení budou diskretní LQ regulátory³ (viz [7, kap. 2]).

Diskretní LQ regulátor je optimální stavový regulátor a pro jeho návrh je potřeba získat diskretní lineární stavový popis. Před návrhem regulátoru nejprve provedeme linearizaci rovnice (2.9) kolem pracovního bodu $\hat{\Omega}_r$.

Linearizaci získáme rovnicí (4.7), kde $\Delta\ddot{\Omega}$ je úhlové zrychlení, $\Delta\dot{\Omega}$ úhlová rychlost, ΔM moment, který působí na tělo helikoptéry, $\hat{\Omega}_r$ je již zmíněný pracovní pod, kolem kterého provádíme linearizaci a $I_{3,1}$ je vektor $[1 \ 1 \ 1]^T$.

³Linear-quadratic regulator - lineární kvadraticky optimální regulátor

$$\Delta\ddot{\Omega} = I^{-1} \left(I_{3,1} \times I \cdot \dot{\Omega}_r + \dot{\Omega}_r \times I_{3,1} \right) \cdot \Delta\dot{\Omega} + I^{-1} \cdot \Delta M \quad (4.7)$$

Linearizovaný model můžeme dále zjednodušit tak, že v rovnici použijeme diagonální matici momentů setrvačnosti I . Pokud matice I není diagonální, předpokládáme, že vzhledem k charakteristickému symetrickému tvaru kvadrotorové helikoptéry jsou prvky matice I na diagonále dominantní. Můžeme tedy okolní prvky vypustit, případně matici vynásobit poměrem normy diagonální matice a matice původní. Výsledkem tohoto zjednodušení je soustava rovnic (4.8), (4.9) a (4.10).

$$\Delta\ddot{\phi} = -\frac{(\psi_r + \theta_r) \cdot (I_y - I_z)}{I_x} \cdot \Delta\dot{\phi} + \frac{1}{I_x} \cdot (\Delta M_\phi \cdot \Delta M_{f\phi}) \quad (4.8)$$

$$\Delta\ddot{\theta} = \frac{(\phi_r + \psi_r) \cdot (I_x - I_z)}{I_y} \cdot \Delta\dot{\theta} + \frac{1}{I_y} \cdot (\Delta M_\theta \cdot \Delta M_{f\theta}) \quad (4.9)$$

$$\Delta\ddot{\psi} = -\frac{(\phi_r + \theta_r) \cdot (I_x - I_y)}{I_z} \cdot \Delta\dot{\psi} + \frac{1}{I_z} \cdot \Delta M_\psi \quad (4.10)$$

Z výše uvedených rovnic lze vypožorovat, že rovnice nejsou na sobě závislé a rotační pohyb kolem každého Eulerova úhlu lze řídit zvlášť. Za účelem návrhu tří nezávislých regulátorů je třeba z linearizovaných rovnic (4.8), (4.9) a (4.10) vytvořit tři nezávislé stavové popisy 4.11, 4.12 a 4.13.

$$\begin{aligned} \phi : \\ x_1 = \Delta\phi; x_2 = \Delta\dot{\phi} = \dot{x}_1; u_\phi = \Delta M_\phi + \Delta M_{f\phi} \\ \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{(\psi_r + \theta_r) \cdot (I_y - I_z)}{I_x} & 0 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{I_x} \end{bmatrix} \cdot u_\phi \end{aligned} \quad (4.11)$$

$$\begin{aligned} \theta : \\ x_1 = \Delta\theta; x_2 = \Delta\dot{\theta} = \dot{x}_1; u_\theta = \Delta M_\theta + \Delta M_{f\theta} \\ \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ \frac{(\phi_r + \psi_r) \cdot (I_x - I_z)}{I_y} & 0 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{I_y} \end{bmatrix} \cdot u_\theta \end{aligned} \quad (4.12)$$

$$\begin{aligned} \psi : \\ x_1 = \Delta\psi; x_2 = \Delta\dot{\psi} = \dot{x}_1; u_\psi = \Delta M_\psi \\ \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{(\phi_r + \theta_r) \cdot (I_x - I_y)}{I_z} & 0 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{I_z} \end{bmatrix} \cdot u_\psi \end{aligned} \quad (4.13)$$

Následně je možné provést diskretizaci stavových popisů 4.11, 4.12 a 4.13. Diskretizaci provedeme pomocí aproximace (4.14). Výsledkem je diferenční rovnice (4.15), kde T_{so} je perioda modelu.

$$\frac{dx(t)}{dt} \approx \frac{x(t) - x(t-1)}{T_{so}} \quad (4.14)$$

$$x(t) = (I_{2,2} - A \cdot T_{s_o})^{-1} \cdot x(t-1) + (I_{2,2} - A \cdot T_{s_o})^{-1} \cdot B \cdot T_{s_o} \cdot u(t) \quad (4.15)$$

Diskrétní lineární model získáme dosazením matic z (4.8), (4.9) a (4.10) do (4.15). Finální diskrétní stavový popis pro rotační pohyb kolem Eulerova úhlu ϕ je popsán v 4.16, pro θ v 4.17 a pro ψ v 4.18.

$$\begin{aligned} \phi : \\ \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} &= \frac{\begin{bmatrix} I_x & I_x \cdot T_{s_o} \\ (\psi_r + \theta_r) \cdot (I_y - I_z) \cdot T_{s_o} & I_x \end{bmatrix}}{I_x + T_{s_o}^2 (\psi_r + \theta_r) \cdot (I_y - I_z)} \cdot \begin{bmatrix} x_1(t-1) \\ x_2(t-1) \end{bmatrix} \\ &+ \frac{\begin{bmatrix} T_{s_o}^2 \\ T_{s_o} \end{bmatrix}}{I_x + T_{s_o}^2 (\psi_r + \theta_r) \cdot (I_y - I_z)} \cdot u_\phi(t) \end{aligned} \quad (4.16)$$

$$\begin{aligned} \theta : \\ \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} &= \frac{\begin{bmatrix} I_y & I_y \cdot T_{s_o} \\ (\phi_r + \psi_r) \cdot (I_x - I_z) \cdot T_{s_o} & I_y \end{bmatrix}}{I_y + T_{s_o}^2 (\phi_r + \psi_r) \cdot (I_x - I_z)} \cdot \begin{bmatrix} x_1(t-1) \\ x_2(t-1) \end{bmatrix} \\ &+ \frac{\begin{bmatrix} T_{s_o}^2 \\ T_{s_o} \end{bmatrix}}{I_y + T_{s_o}^2 (\phi_r + \psi_r) \cdot (I_x - I_z)} \cdot u_\theta(t) \end{aligned} \quad (4.17)$$

$$\begin{aligned} \psi : \\ \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} &= \frac{\begin{bmatrix} I_z & I_z \cdot T_{s_o} \\ (\phi_r + \theta_r) \cdot (I_x - I_y) \cdot T_{s_o} & I_z \end{bmatrix}}{I_z + T_{s_o}^2 (\phi_r + \theta_r) \cdot (I_x - I_y)} \cdot \begin{bmatrix} x_1(t-1) \\ x_2(t-1) \end{bmatrix} \\ &+ \frac{\begin{bmatrix} T_{s_o}^2 \\ T_{s_o} \end{bmatrix}}{I_z + T_{s_o}^2 (\phi_r + \theta_r) \cdot (I_x - I_y)} \cdot u_\psi(t) \end{aligned} \quad (4.18)$$

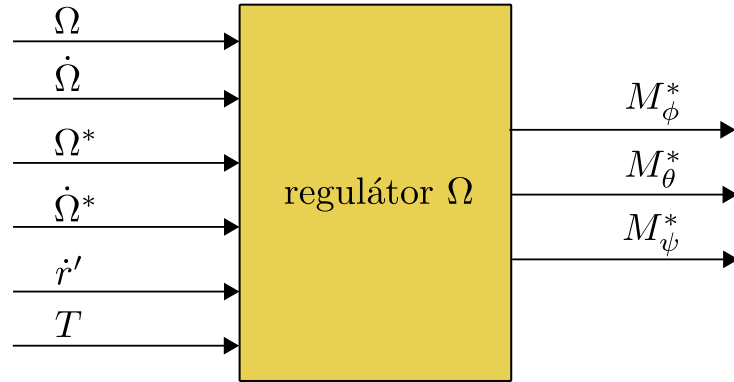
Nyní je připraveno vše potřebné pro návrh diskrétního LQ regulátoru orientace s nekonečným časovým horizontem. LQ regulátor bude navržen na základě minimalizace kritéria (4.19), kde J je hodnota kritéria, $x(t)$ je stav řízeného systému v čase t , $u(t)$ je řízení v čase t a Q_i a R_i jsou parametry regulátoru i , které ovlivňují rychlost a energii řízení.

$$J = \sum_{t=1}^{\infty} (x(t)^T Q_i x(t) + u(t)^T R_i u(t)) \quad (4.19)$$

Výsledný regulátor je ve tvaru (4.20), kde F je parametr regulátoru. Parametr F vypočítáme pomocí (4.21), kde P lze získat jako řešení diskrétní algebraické Riccatiovy rovnice (4.22).

$$u = -Fx(t) \quad (4.20)$$

$$F = (R + B^T P B)^{-1} B^T P A \quad (4.21)$$



Obrázek 4.5: Schéma regulátoru orientace

$$P = Q + A^T \left(P - PB (R + B^T PB)^{-1} B^T P \right) A \quad (4.22)$$

Nyní je potřeba doplnit, že LQ regulátor bude použit s úpravou, která dovoluje regulaci řízeného systému do požadovaného stavu $x^*(t)$. Na vstupu regulátoru nebude tedy stav $x(t)$, jak je popsáno v rovnici (4.20), ale regulační odchylka $e_x(t)$, kde $e_x(t) = x(t) - x^*(t)$. Řešení Riccatiový rovnice a výsledný regulátor bude vypočítán v kapitole ... Je třeba uvést, že výstup regulátoru orientace nepočítá s třepotáním listů rotorů, proto je

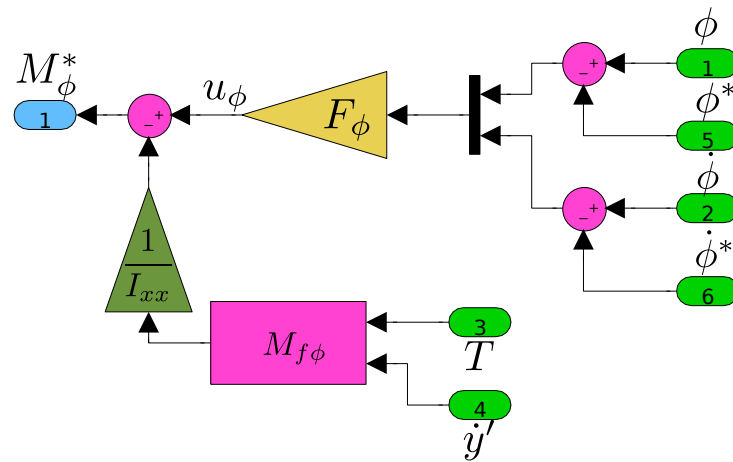
třeba ve výstupu regulátoru tento jev kompenzovat $u_\Omega = \begin{bmatrix} M_\phi^* \\ M_\theta^* \\ M_\psi^* \end{bmatrix} = \begin{bmatrix} u_\phi - \frac{M_{f\phi}}{I_{xx}} \\ u_\theta - \frac{M_{f\theta}}{I_{yy}} \\ u_\psi \end{bmatrix}$. Schéma

regulátoru orientace je vyobrazeno na obrázku 4.5 a schémata jednotlivých regulátorů se pak nacházejí na obrázku 4.6.

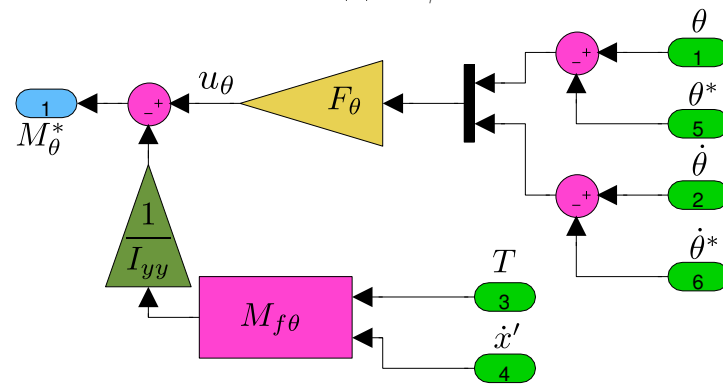
4.4 Regulátor polohy kvadrotorové helikoptéry

V této části bude navržen regulátor pohybu kvadrotorové helikoptéry v inerciálním souřadném systému. Jádrem regulátoru budou opět tři diskrétní LQ regulátory. Systém nebude linearizovat kolem pracovního bodu, ale pomocí vstupu systému.

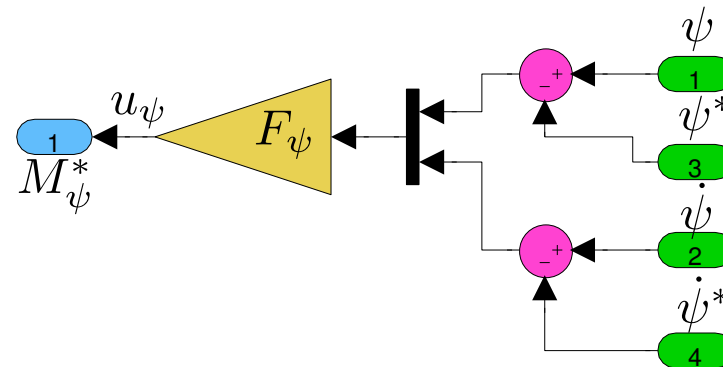
Řízení bude odvozeno z rovnice (2.1), kde je nelinearita systému způsobena rotační maticí e^Ω . Problém nelinearity budeme řešit tak, že regulační odchylku přenásobíme inverzní rotační maticí (4.23). Tímto krokem transformujeme regulační odchylku z inerciálního souřadného systému do souřadného systému helikoptéry.



(a) LQ $_{\phi}$



(b) LQ $_{\theta}$



(c) LQ $_{\psi}$

Obrázek 4.6: Schéma LQ regulátorů orientace doplněných o kompenzaci třepotání listů rotorů (4.6a - regulace po ϕ , 4.6b - regulace po θ a 4.6c - regulace po ψ)

$$\begin{aligned}
e_r &= \begin{bmatrix} x \\ y \\ z \end{bmatrix} - \begin{bmatrix} x^* \\ y^* \\ z^* \end{bmatrix} \\
e_{\dot{r}} &= \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} - \begin{bmatrix} \dot{x}^* \\ \dot{y}^* \\ \dot{z}^* \end{bmatrix} \\
e_{r'} &= (e^{\Omega_x})^{-1} \cdot e_r \\
e_{\dot{r}'} &= (e^{\Omega_x})^{-1} \cdot e_{\dot{r}}
\end{aligned} \tag{4.23}$$

Zjednodušené lineární systémy, pro které budou navrženy LQ regulátory, jsou popsány v (4.24), (4.25) a (4.26). Vstup systému obsahuje pouze potřebnou složku (znázorněno operátorem $[]_i$, kde i je požadovaná složka.

$$\begin{aligned}
x_1 = x'; \quad x_2 = \dot{x}_1 = \dot{x}'; \quad u_{x'} &= \left[(e^{\Omega_x})^{-1} \cdot -\frac{G_z}{m} + \frac{1}{m} \cdot F \right]_{x'} \\
\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} &= \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \cdot u_{x'}
\end{aligned} \tag{4.24}$$

$$\begin{aligned}
x_1 = y'; \quad x_2 = \dot{x}_1 = \dot{y}'; \quad u_{y'} &= \left[(e^{\Omega_x})^{-1} \cdot -\frac{G_z}{m} + \frac{1}{m} \cdot F \right]_{y'} \\
\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} &= \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \cdot u_{y'}
\end{aligned} \tag{4.25}$$

$$\begin{aligned}
x_1 = z'; \quad x_2 = \dot{x}_1 = \dot{z}'; \quad u_{z'} &= \left[(e^{\Omega_x})^{-1} \cdot -\frac{G_z}{m} + \frac{1}{m} \cdot F \right]_{z'} \\
\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} &= \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \cdot u_{z'}
\end{aligned} \tag{4.26}$$

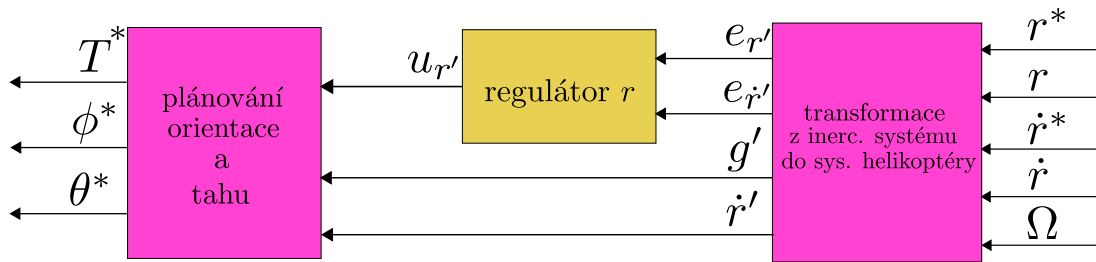
Diskretizací systémů dle vztahu (4.15) získáme popis systémů ve tvaru (4.27), (4.28) a (4.29).

$$\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} = \begin{bmatrix} 1 & T_{s_{x'}} \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_1(t-1) \\ x_2(t-1) \end{bmatrix} + \begin{bmatrix} T_{s_{x'}}^2 \\ T_{s_{x'}} \end{bmatrix} \cdot u_{x'}(t) \tag{4.27}$$

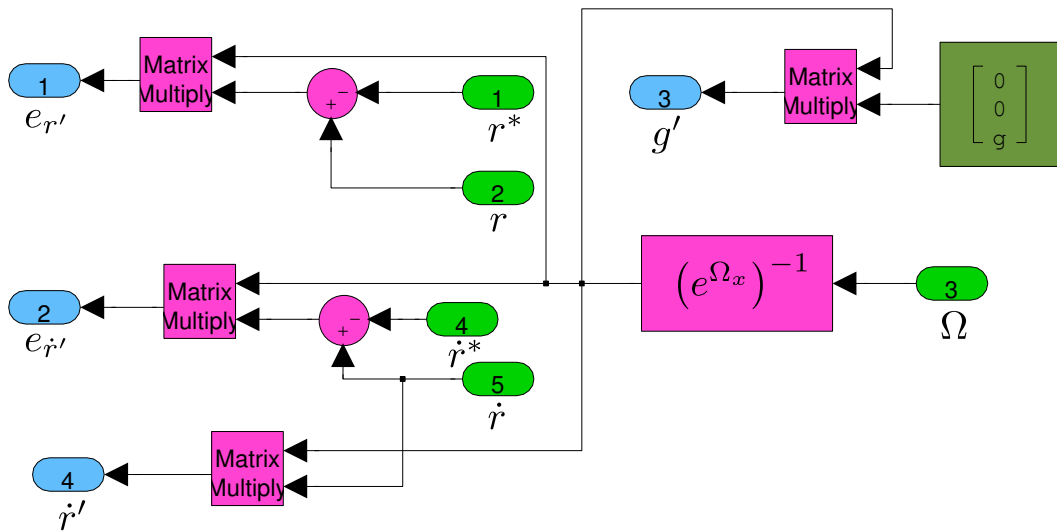
$$\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} = \begin{bmatrix} 1 & T_{s_{y'}} \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_1(t-1) \\ x_2(t-1) \end{bmatrix} + \begin{bmatrix} T_{s_{y'}}^2 \\ T_{s_{y'}} \end{bmatrix} \cdot u_{y'}(t) \tag{4.28}$$

$$\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} = \begin{bmatrix} 1 & T_{s_z} \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_1(t-1) \\ x_2(t-1) \end{bmatrix} + \begin{bmatrix} T_{s_z}^2 \\ T_{s_z} \end{bmatrix} \cdot u_{z'}(t) \tag{4.29}$$

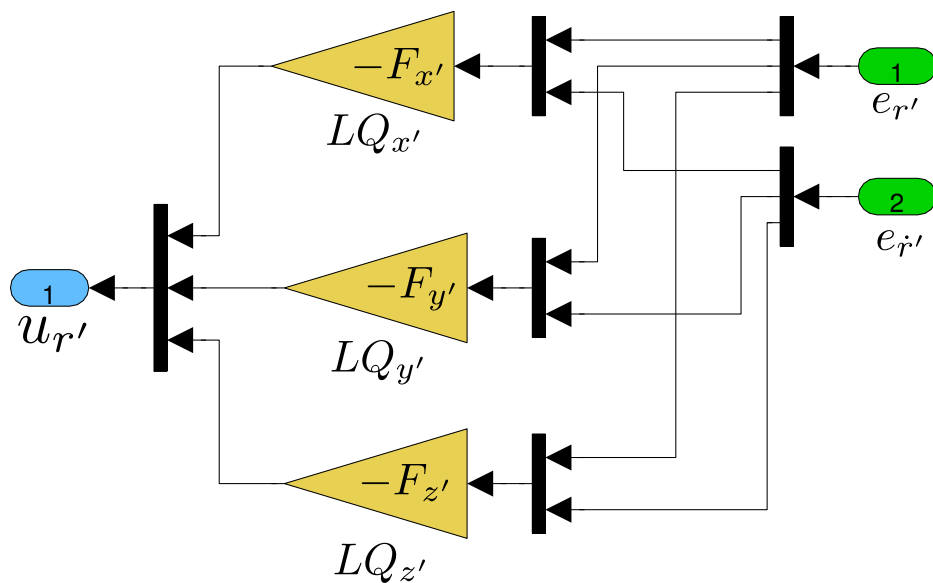
Regulátor bude v části 5.2 navržen opět podle (4.19), (4.20), (4.21) a (4.22) z části 5.1. Je třeba uvést, že musíme na výstupu regulátorů kompenzovat gravitační zrychlení a



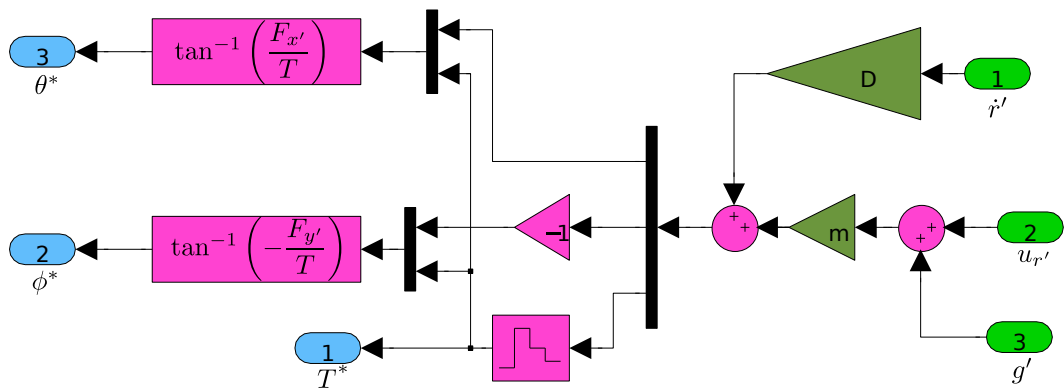
Obrázek 4.7: Schéma regulátoru polohy



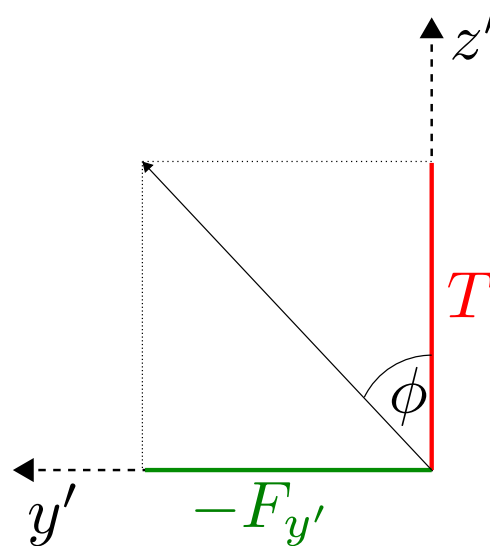
Obrázek 4.8: Schéma transformace z inerciálního systému do systému helikoptéry pro regulátor polohy a plánování orientace a tahu (pozn.: blok *Matrix Multiply* provádí maticové násobení mezi prvním a druhým vstupem)



Obrázek 4.9: Schéma regulátoru polohy v souřadné soustavě helikoptéry



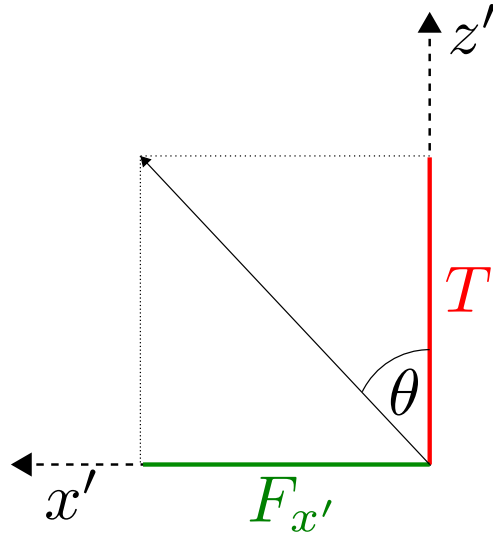
Obrázek 4.10: Schéma plánování orientace a tahu kvadrotorové helikoptéry



Obrázek 4.11: Vliv tahu T při natočení ϕ na ose y'

odpor vzduchu. Tím získáme požadovaný celkový tah rotorů T^* . Poloha na x' a y' se bude regulovat pomocí požadovaných úhlů ϕ^* a θ^* . Na obrázku 4.11 a 4.12 je znázorněno, jak se tah T z osy z' projeví na ose x' a y' pomocí natočení helikoptéry o ϕ a θ . Výsledný přepočet výstupu z LQ regulátorů lze přepočítat pomocí vztahu (4.30).

Schéma regulátoru polohy, který generuje T^* , ϕ^* a θ^* , je zakresleno na obrázku 4.7. Schéma části regulátoru, kde je prováděna transformace potřebných složek z inerciálního souřadného systému do souřadného systému helikoptéry, je znázorněno na obrázku 4.8. Schéma s LQ regulátory polohy v souřadném systému helikoptéry je znázorněno na obrázku 4.9 a schéma části regulátoru, kde je prováděna kompenzace gravitačního zrychlení, kompenzace odporu vzduchu, výpočet požadovaných úhlů ϕ^* , θ^* a výpočet požadovaného tahu T^* se nachází na obrázku 4.10.



Obrázek 4.12: Vliv tahu T při natočení θ na ose x'

$$\begin{aligned}
 g' &= (e^\Omega)^{-1} \cdot [0 \ 0 \ g]^T \\
 [F_{x'} \ F_{y'} \ T^*]^T &= D \cdot \dot{r}' + m \cdot (g' + u_{r'}) \\
 u_{T^*, \phi^*, \theta^*} &= \begin{bmatrix} T^* \\ \tan^{-1} \left(-\frac{F_{y'}}{T^*} \right) \\ \tan^{-1} \left(\frac{F_{x'}}{T^*} \right) \end{bmatrix}
 \end{aligned} \tag{4.30}$$

5 Simulace řízené kvadrotorové helikoptéry

V této části bude ověřena funkčnost regulátorů, které byly navrženy v části 4 na modelu, který byl popsán v části 2. Nejprve budou uvedeny všechny parametry modelu a regulátorů, které jsou potřeba pro běh simulace a následně bude provedeno několik simulací, které otestují chování regulovaného modelu v různých situacích. K provedení simulací bude využito prostředí *Matlab* a *Simulink*.

5.1 Parametry modelu kvadrotorové helikoptéry

Parametry modelu kvadrotorové helikoptéry jsou uvedeny v tabulce 5.1. Parametry odpovídají větší kvadrotorové helikoptéry s hmotností 4 kg a délkou ramene 0,4 m. Značení odpovídá značení z předchozích kapitol, RPM_{max} jsou maximální možné otáčky každého rotoru a RPM_{min} jsou nejnižším možné otáčky.

Matice setrvačnosti, která bude použita v modelu, je popsána v (5.1). Byla vypočítána v programu *Solidworks* na základě zjednodušeného návrhu těla helikoptéry.

$$I_{\text{model}} = \begin{bmatrix} 0.383722 & -0.180333 & 0.013896 \\ -0.180333 & 0.500561 & -0.011772 \\ 0.013896 & -0.011772 & 0.873406 \end{bmatrix} \quad (5.1)$$

Zjednodušená diagonální matice setrvačnosti, která byla využita v diskretním nelineárním modelu z části a při výpočtu diskretního modelu pro návrh regulátoru stabilizace v části , je popsána v (5.2). Byla vytvořena na základě matice z programu *Solidworks* pro počátek souřadnic v těžišti následnou diagonalizací¹.

$$I_{\text{diskretizace}} = \begin{bmatrix} 0.2448 & 0 & 0 \\ 0 & 0.2639 & 0 \\ 0 & 0 & 0.4997 \end{bmatrix} \quad (5.2)$$

Pro simulaci dynamiky rotorů byl vytvořen spojitý přenos (5.3) na základě přechodové charakteristiky motoru AXI 4120/20 s pravotočivou vrtulí 15x8 od firmy Fiala propellers pomocí Strejcovy metody [11]. Vstupem přenosu jsou požadované otáčky ω_i^* v *RPM* a výstupem jsou otáčky rotoru ω_i v *RPM* (i označuje číslo rotoru, jelikož pro simulaci každého rotoru byl zvolen stejný model).

¹diagonalizace byla provedena vynulováním prvků mimo diagonálu a následným přenásobením poměru eukleidovských norem původní matice a diagonální matice

parametr	hodnota
g	9.81
k_T	$3.8502 \cdot 10^{-4}$
k_M	$1.45 \cdot 10^{-5}$
k_f	0.01
k_β	0.06
m	4
l	0.4
h	0.03
I_m	$3.357 \cdot 10^{-12}$
D_x	0,00628875
D_y	0,00628875
D_z	0,05031
RPM_{max}	3000
RPM_{min}	500

Tabulka 5.1: Parametry modelu kvadrotorové helikoptéry pro simulaci

parametr	hodnota
T_{sm}	0.005
T_d	0.0226
T_i	52.7664
T_t	1
N	10
K	2.1930

Tabulka 5.2: Parametry regulátoru otáček rotorů

$$F_s(p) = \frac{184611.0787}{p^3 + 171.4p^2 + 9795p + 1.866 \cdot 10^5} \quad (5.3)$$

5.2 Parametry regulátorů a unscentovaného Kalmanova filtru

Parametry PID regulátoru otáček rotoru (viz tabulka 5.2) byly navrženy pomocí nástroje PID Control Laboratory² na základě simulačního modelu rotoru (5.3). Ladění regulátoru bylo provedeno s ohledem na rychlé odregulování, na energii vynaloženou regulátorem a s ohledem na tvar citlivostní funkce regulační smyčky.

Po nastavení regulátoru rotorů byl proveden návrh řídicího algoritmu orientace z části 5.1. Parametry diskretizovaných stavových popisů, na jejichž základě byly vypočítány parametry LQ regulátorů, se nacházejí v tabulce 5.3. Parametry byly upravovány s cílem bezproblémové vzájemné interakce regulátorů (pokud by měl jeden LQ regulátor příliš velký akční zásah, nebylo by možné odregulovat ostatní zásahy kvůli omezenému rozsahu

²viz <http://www.pidlab.com/cs/>

parametr	hodnota
T_{s_o}	0.01
$\dot{\phi}_r$	$\frac{\pi}{4}$
$\dot{\theta}_r$	$\frac{\pi}{4}$
$\dot{\psi}_r$	$\frac{\pi}{8}$
Q_ϕ	$\begin{bmatrix} 1.6211 & 0 \\ 0 & 0.7205 \end{bmatrix}$
Q_θ	$\begin{bmatrix} 1.6211 & 0 \\ 0 & 0.7205 \end{bmatrix}$
Q_ψ	$\begin{bmatrix} 3.6476 & 0 \\ 0 & 4.5032 \end{bmatrix}$
R_ϕ	0.04
R_θ	0.04
R_ψ	0.111

Tabulka 5.3: Parametry pro výpočet LQ regulátorů orientace

parametr	hodnota
F_ϕ	$\begin{bmatrix} 6.6360 & 3.1035 \end{bmatrix}$
F_θ	$\begin{bmatrix} 6.0095 & 3.0921 \end{bmatrix}$
F_ψ	$\begin{bmatrix} 1.9371 & 1.5858 \end{bmatrix}$

Tabulka 5.4: Parametry LQ regulátorů pro regulaci orientace

otáček rotorů). Také bylo při návrhu cílem rychle regulovat systém za účelem sledování řízení z vyšší vrstvy (regulátory polohy generující ϕ^* a θ^*). Parametry samotných regulátorů orientace se nacházejí v tabulce 5.4.

Po nastavení regulátoru orientace byl proveden návrh regulátoru polohy z části 4.4. Parametry stavových popisů, na jejichž základě byli vypočítány parametry LQ regulátorů, se nacházejí v tabulce 5.5. Parametry byly voleny s cílem řízení letu helikoptéry po generované trajektorii. Parametry samotných regulátorů polohy se nacházejí v tabulce 5.6.

Parametry uncentovaného Kalmanova filtru (viz tabulka 5.7) byly nastaveny pro odhad polohy, rychlosti, orientace a úhlového zrychlení kvadrotorové helikoptéry. Pro měření zrychlení byly použity dva akcelerometry (pro zpřesnění odhadu). $T_{s_{gps}}$ je perioda měření modulu GPS, $T_{s_{acc1}}$ a $T_{s_{acc2}}$ jsou periody měření prvního a druhého akcelerometru, $T_{s_{gyro}}$ je perioda měření gyroskopu, $T_{s_{mag}}$ je perioda měření magnetometru, T_{sd} je perioda, ve které byl spouštěn algoritmus UKF, a perioda diskretního nelineárního modelu, R_i jsou předpokládané variance šumu senzoru i .

Maticе Q a R byly zvoleny jako diagonální matice. Maticе Q má na diagonále prvky $[0.1, 0.01, 0.1, 0.01, 0.1, 0.01, 0.01, 0.001, 0.01, 0.001, 0.01, 0.001]$ a matice R má na diagonále prvky $[R_{gps}, R_{gps}, R_{gps_z}, R_{gps}, R_{gps}, R_{gps_z}, R_{acc1}, R_{acc1}, R_{acc1}, R_{acc1}, R_{acc1}, R_{acc1}, R_{acc2}, R_{acc2}, R_{acc2}, R_{acc2}, R_{acc2}, R_{acc2}, R_{gyro}, R_{gyro}, R_{gyro}, R_{gyro}, R_{gyro}, R_{gyro}, R_{mag}, R_{mag}]$, kde se nacházejí variance chyby měření v pořadí poloha z gps, poloha z prvního akcelerometru, rychlost z prvního akcelerometru, poloha z druhého akcelerometru, rychlost z druhého akcelerometru, úhel z gyroskopu, úhlová rychlost z gyroskopu, úhel ψ z magnetometru a

parametr	hodnota
$T_{s_{x',y'}}$	0.05
T_{s_z}	0.01
Q_x	$\begin{bmatrix} 1.5625 & 0 \\ 0 & 1 \end{bmatrix}$
Q_y	$\begin{bmatrix} 1.5625 & 0 \\ 0 & 1 \end{bmatrix}$
Q_z	$\begin{bmatrix} 1 & 0 \\ 0 & 0.0044 \end{bmatrix}$
R_x	1
R_y	1
R_z	0.0044

Tabulka 5.5: Parametry pro výpočet LQ regulátorů polohy

parametr	hodnota
F_x	$\begin{bmatrix} 2.7921 & 2.8154 \end{bmatrix}$
F_y	$\begin{bmatrix} 2.7921 & 2.8154 \end{bmatrix}$
F_z	$\begin{bmatrix} 14.5882 & 5.5618 \end{bmatrix}$

Tabulka 5.6: Parametry LQ regulátorů pro regulaci polohy

úhlová rychlost $\dot{\psi}$ z magnetometru.

5.3 Výsledky simulace

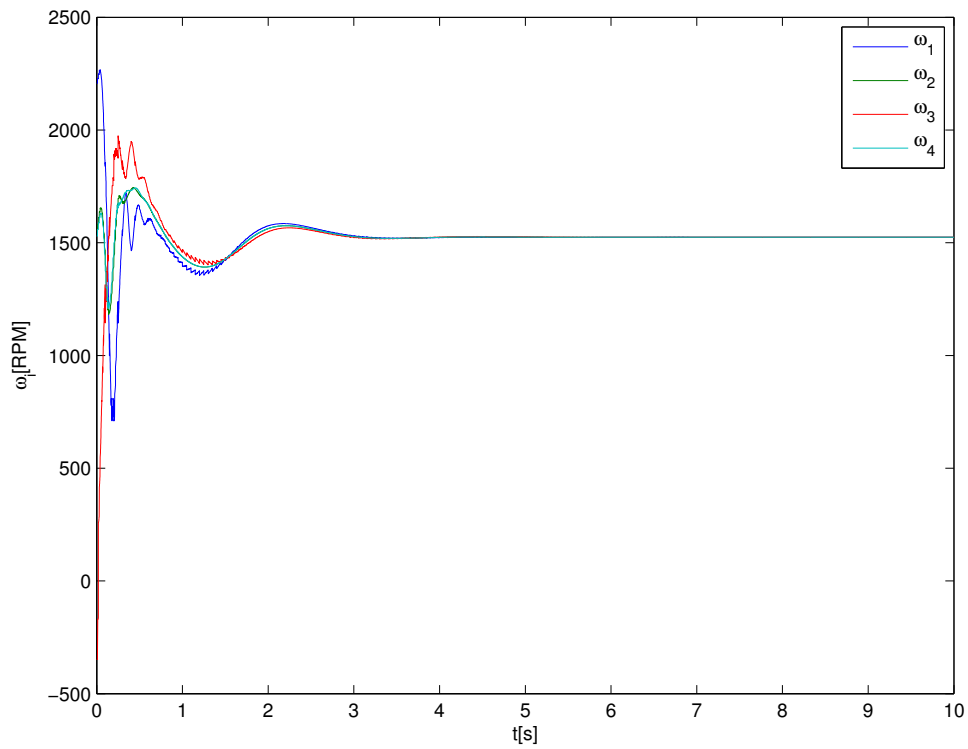
Pro ověření funkce regulátorů byly provedeny dva druhy simulací. První simulace měla za cíl stabilizaci helikoptéry z počátečních podmínek a let do určeného bodu. Ve druhé simulaci byl prováděn let po zadané trajektorii. Obě simulace byly provedeny v prostředí *Simulink* při relativní toleranci $1 \cdot 10^{-3}$ s proměnným krokem pomocí řešitele (solveru) „ode45 (Dormand_Prince)“. Na počátku všech simulací mají rotory helikoptéry nulové otáčky. Na datech z obou simulacích, které byly následně zaneseny šumem, je proveden odhad stavu helikoptéry pomocí UKF.

5.3.1 Stabilizace kvadrotorové helikoptéry a let do určeného bodu

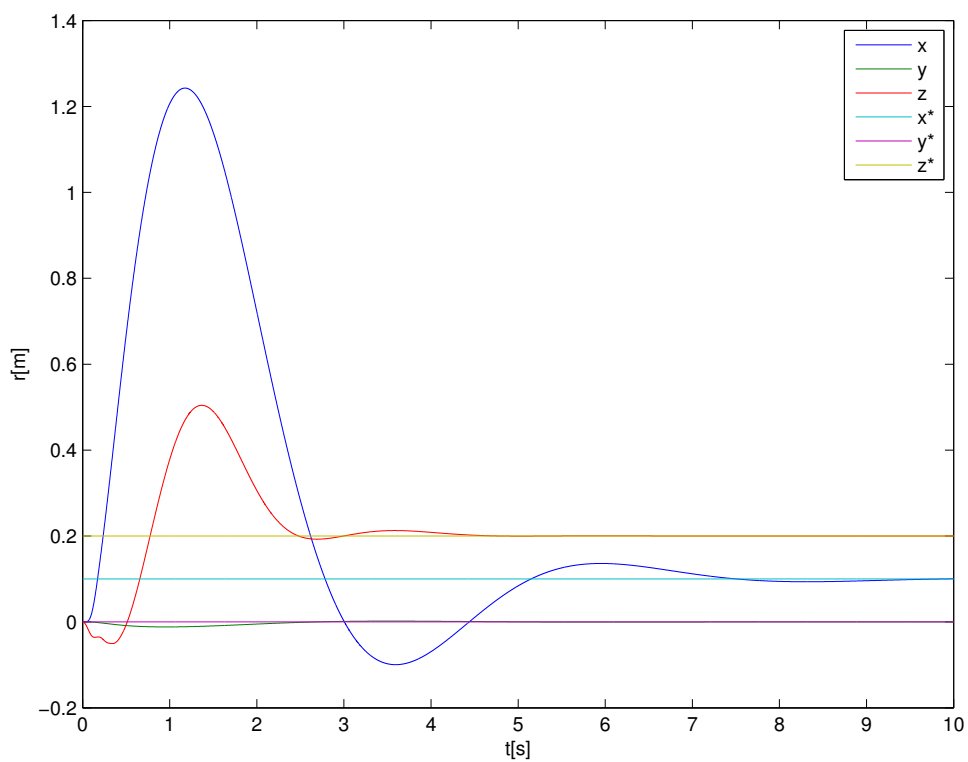
Stabilizace kvadrotorové helikoptéry a let do určeného bodu byl proveden za počátečních podmínek $r_0 = [0, 0, 0]^T$, $\dot{r}_0 = [0, 0, 0]^T$, $\Omega_0 = [0.01, \frac{pi}{4}, 0]^T$ a $\dot{\Omega}_0 = [0, 0, 0]^T$. Cílový bod byl nastaven jako $r^* = [0.1, 0, 0.2]^T$ a požadované natočení helikoptéry bylo $\psi^* = 0$. Výsledky simulace jsou znázorněny na obrázcích 5.1, 5.2, 5.3 a 5.4. Na obrázcích vidíme, že došlo k překmitu na poloze x z důvodu počátečního náklonu $\theta_0 = \frac{\pi}{4}$. Další překmit je pozorovatelný na poloze z z důvodu nabrání požadované výšky. I přes počáteční náklon doletěla helikoptéra v simulaci do požadovaného bodu.

parametr	hodnota
α	0.2
β	2
κ	3
T_{sgps}	0.01
T_{sacc1}	0.001
T_{sacc2}	0.001
T_{sgyro}	0.001
T_{smag}	0.01
T_{sd}	0.001
R_{gps}	9
R_{gps_z}	25
R_{acc1}	$4.872 \cdot 10^{-4}$
R_{acc2}	$1.5398 \cdot 10^{-7}$
R_{gyro}	$1.444 \cdot 10^{-5}$
R_{mag}	0.01

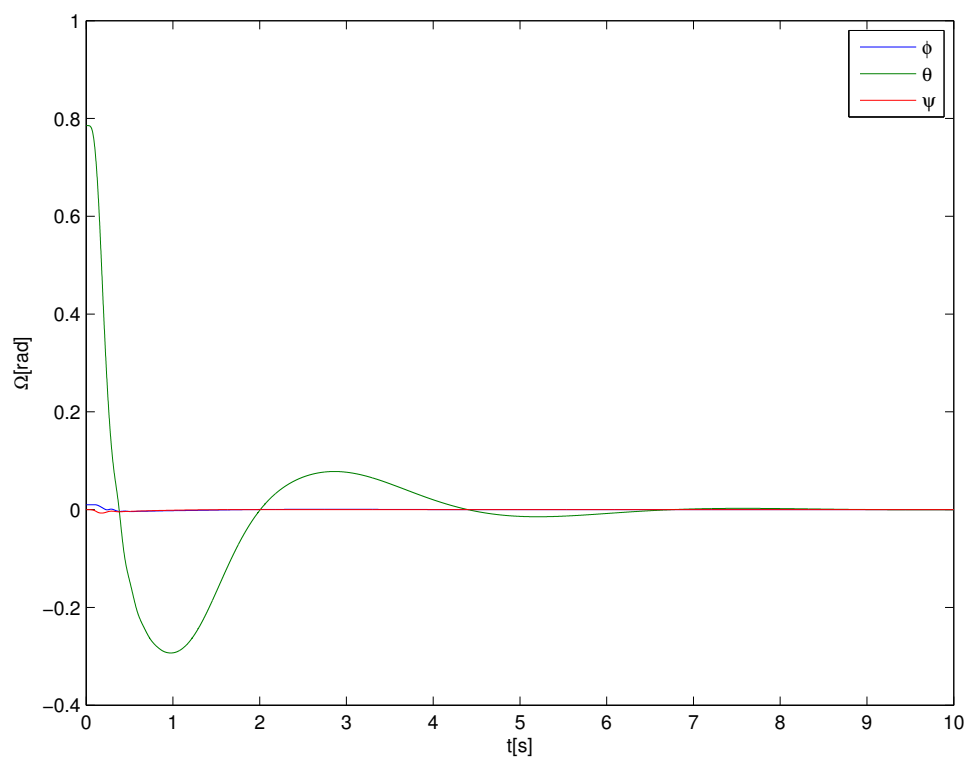
Tabulka 5.7: Parametry unscentovaného Kalmanova filtru



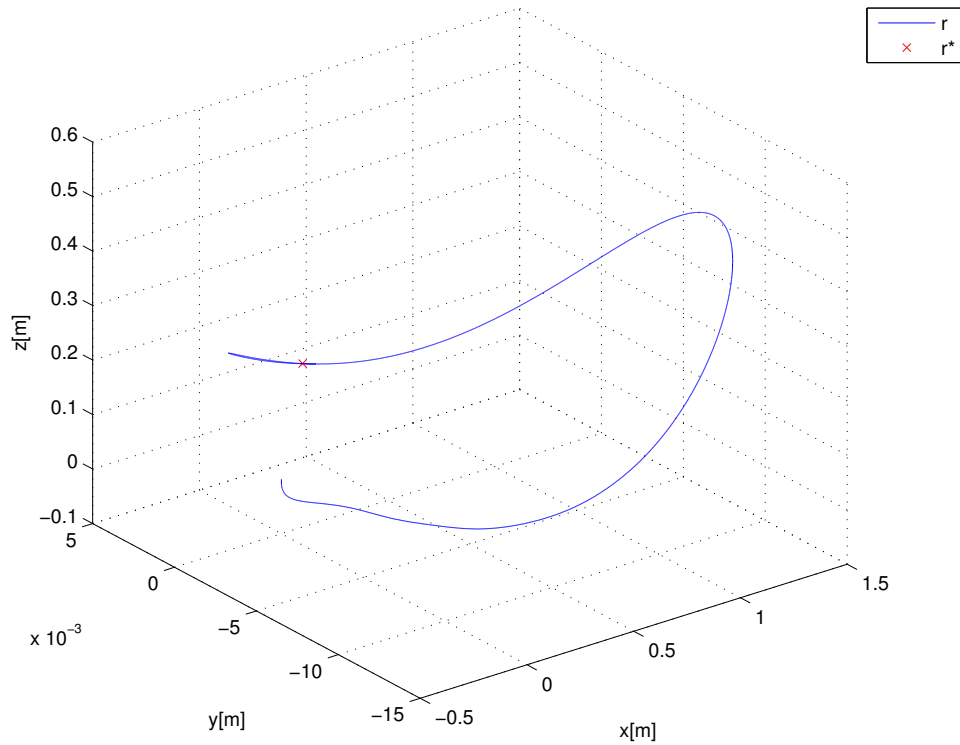
Obrázek 5.1: Otáčky rotorů při letu do požadovaného bodu



Obrázek 5.2: Poloha kvadrotorové helikoptéry a požadovaná poloha při letu do cílového bodu



Obrázek 5.3: Orientace kvadrotorové helikoptéry při letu do požadovaného bodu



Obrázek 5.4: Trajektorie letu kvadrotorové helikoptéry a cílový bod letu

5.3.2 Let kvadrotorové helikoptéry dle generované trajektorie

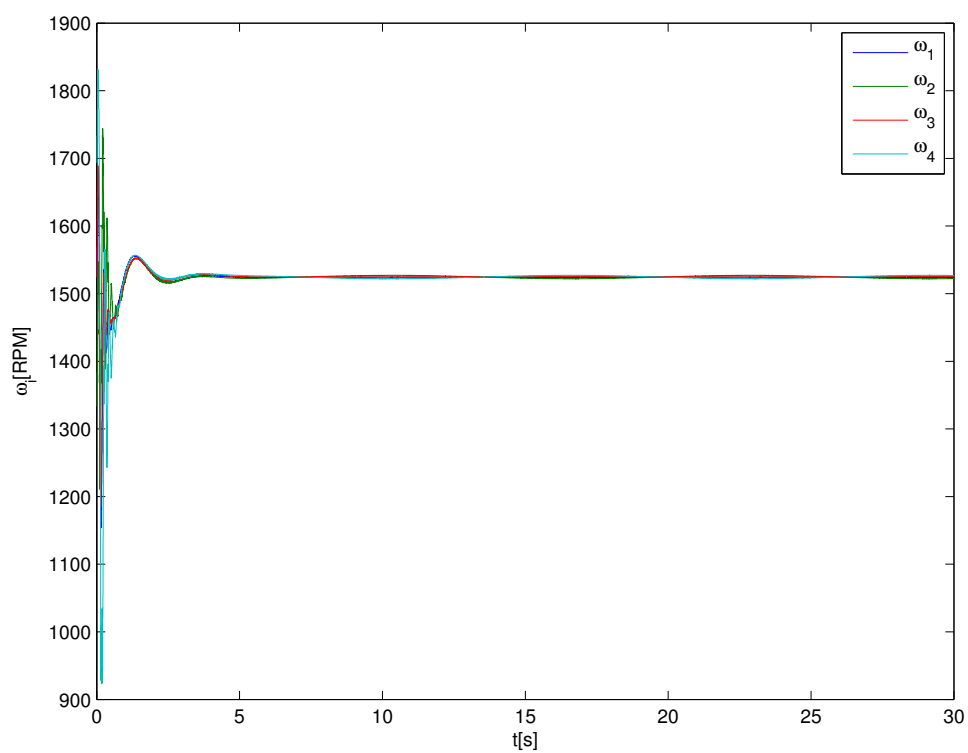
Let kvadrotorové helikoptéry byl generován pro požadovanou trajektorii (5.4) s otočení helikoptéry (5.5). Počáteční podmínky helikoptéry byly nulové. Výsledky simulace jsou na obrázcích 5.5, 5.6, 5.7 a 5.8. Vidíme, že helikoptéra i přes vzdálenost počáteční polohy od požadované trajektorie po čase tuto trajektorii sleduje.

$$r^* = \begin{bmatrix} \sin(0.5 \cdot t) \\ \sin\left(0.5 \cdot t - \frac{\pi}{2}\right) \\ 0.1 \cdot t \end{bmatrix} \quad (5.4)$$

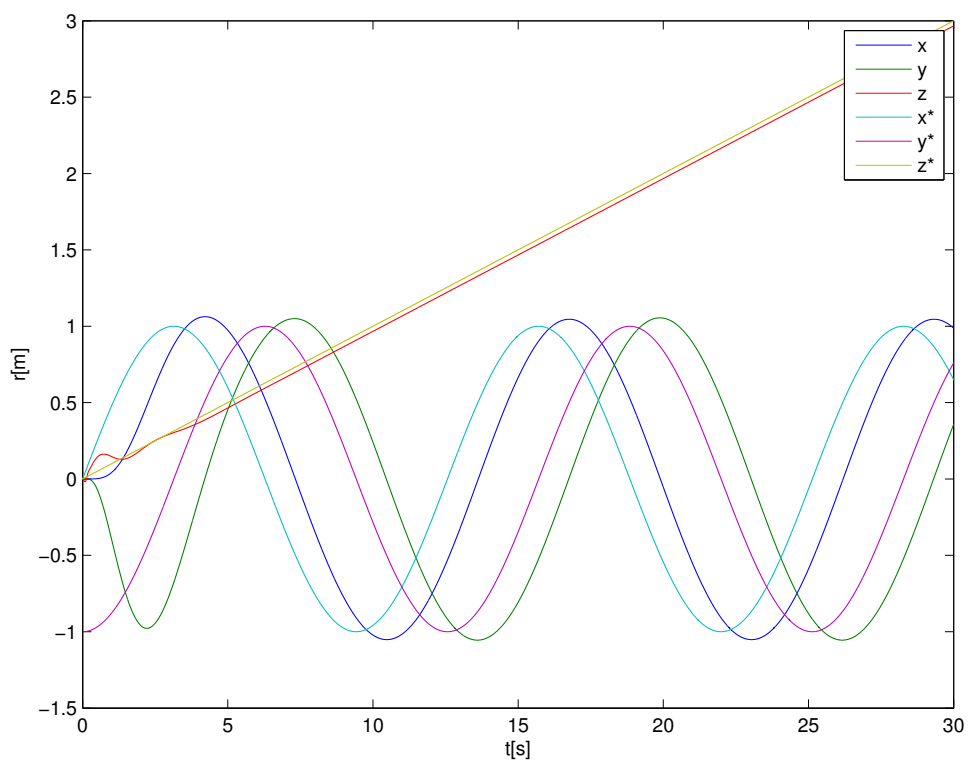
$$\psi^* = 0 \quad (5.5)$$

5.3.3 Odhad stavu kvadrotorové helikoptéry při stabilizaci a letu do určeného bodu

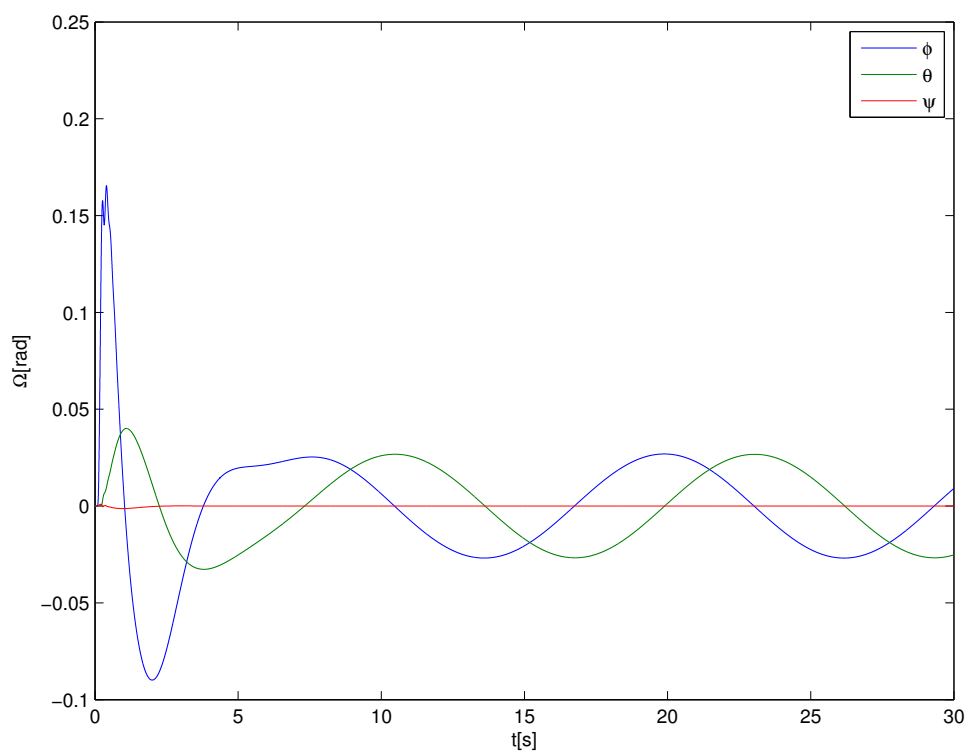
Odhad stavu kvadrotorové helikoptéry byl proveden na základě zaznamenaných dat ze simulace z části 5.3.1. \bar{x}_0 bylo zvoleno na základě počátečních podmínek simulace a matice P_0 byla zvolena jako diagonální matice s prvky $[1, 1, 1, 1, 1, 1, 0.1, 0.1, 0.1, 0.01, 0.01, 0.01]$ na diagonále. Data byla zašuměna bílým šumem s nulovou střední hodnotou a variancí, která odpovídá prvkům matice R z části 5.2 pro dané senzory. Výsledné odhady a porovnání s nezašuměnými daty je na obrázcích 5.9, 5.10, 5.11, 5.12 a 5.13. Z obrázků lze vypořizovat, že odhady pomocí UKF se blíží nezašuměným hodnotám.



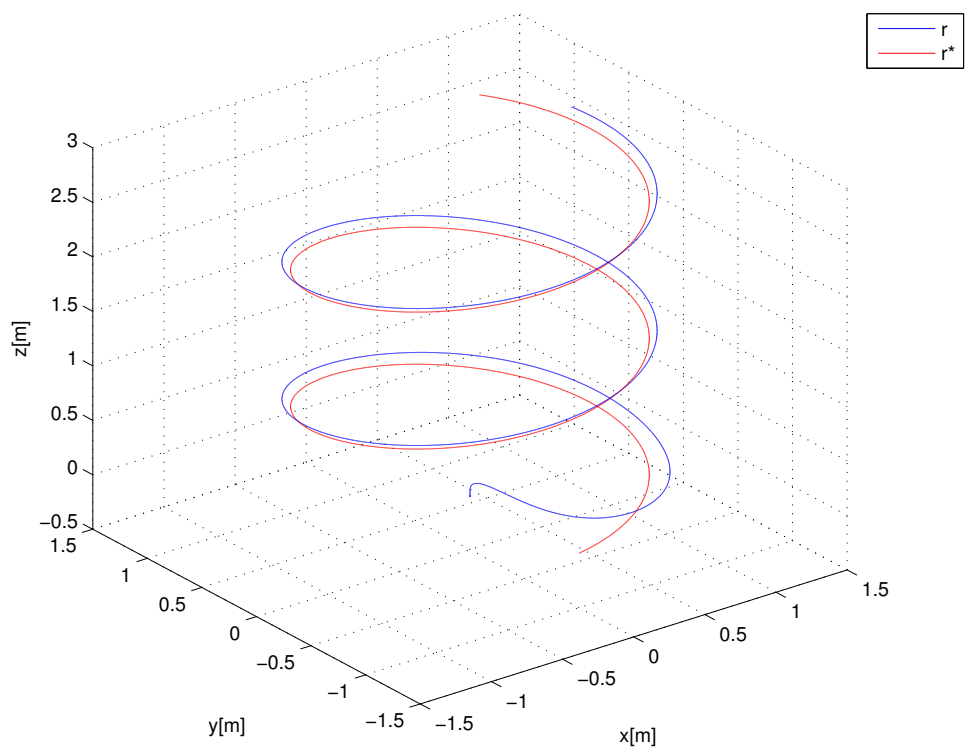
Obrázek 5.5: Otáčky rotorů při letu po trajektorii



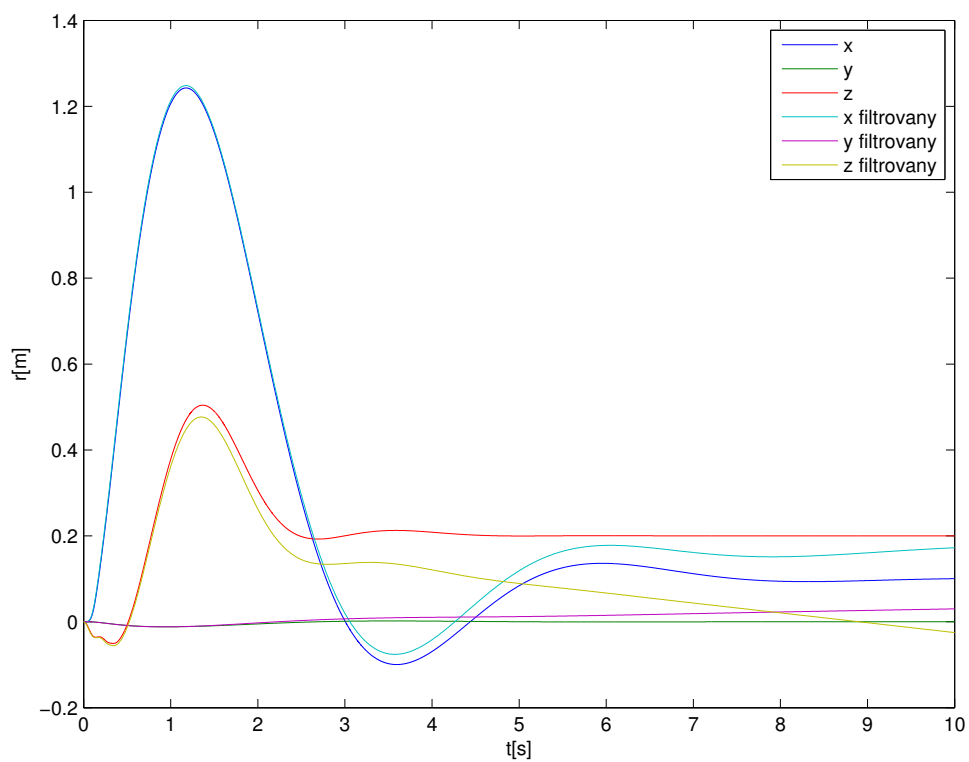
Obrázek 5.6: Poloha kvadrotorové helikoptéry a požadovaná poloha při letu po trajektorii



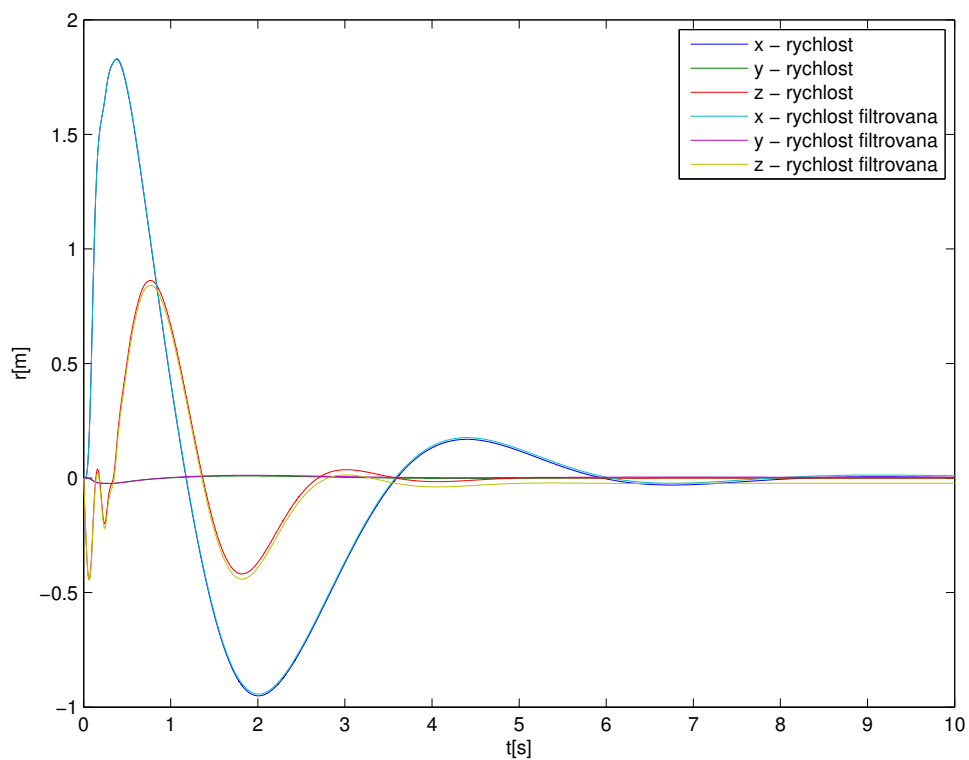
Obrázek 5.7: Orientace kvadrotorové helikoptéry při letu po trajektorii



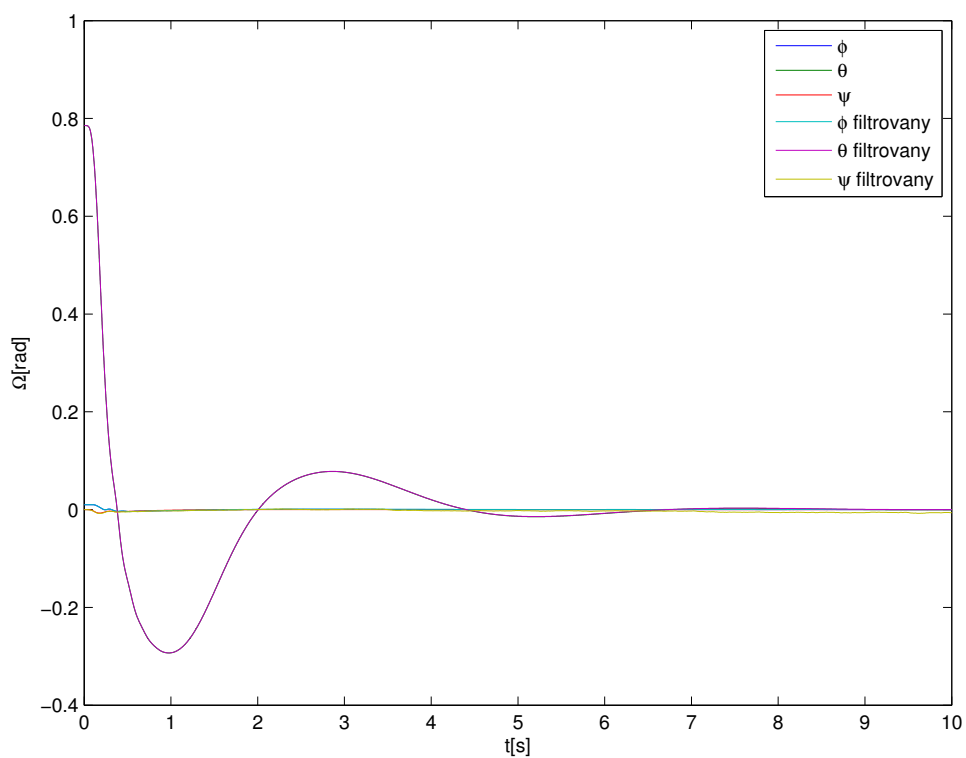
Obrázek 5.8: Trajektorie letu kvadrotorové helikoptéry a požadovaná letová trajektorie



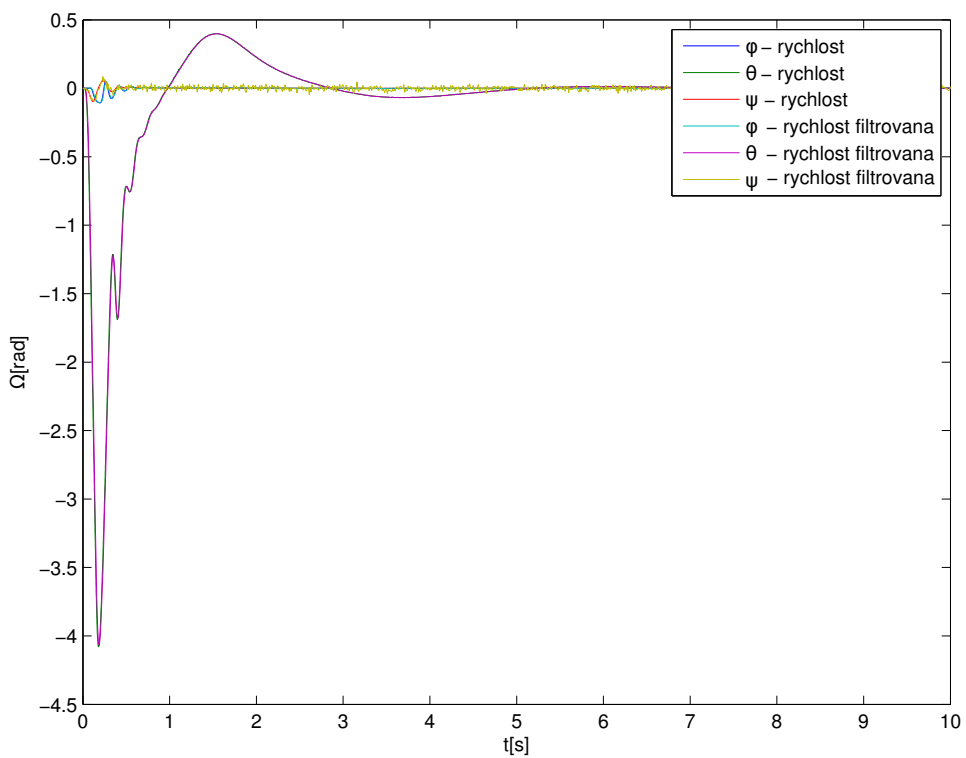
Obrázek 5.9: Poloha kvadrotorové helikoptéry a její odhad při letu do cílového bodu



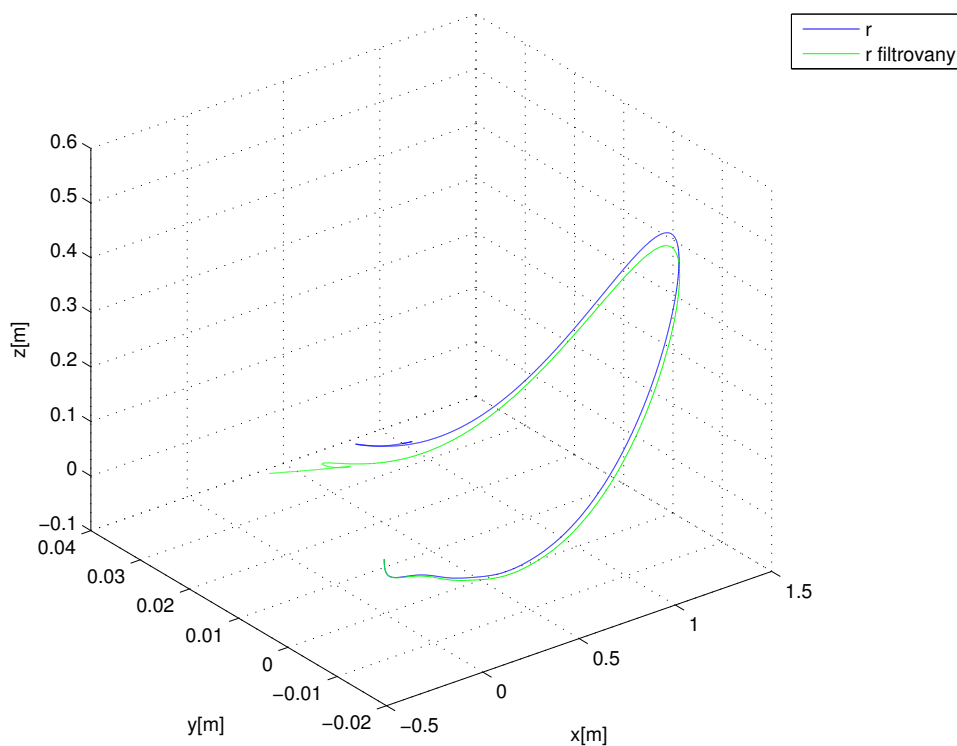
Obrázek 5.10: Rychlost kvadrotorové helikoptéry a její odhad při letu do cílového bodu



Obrázek 5.11: Orientace kvadrotorové helikoptéry a její odhad při letu do cílového bodu



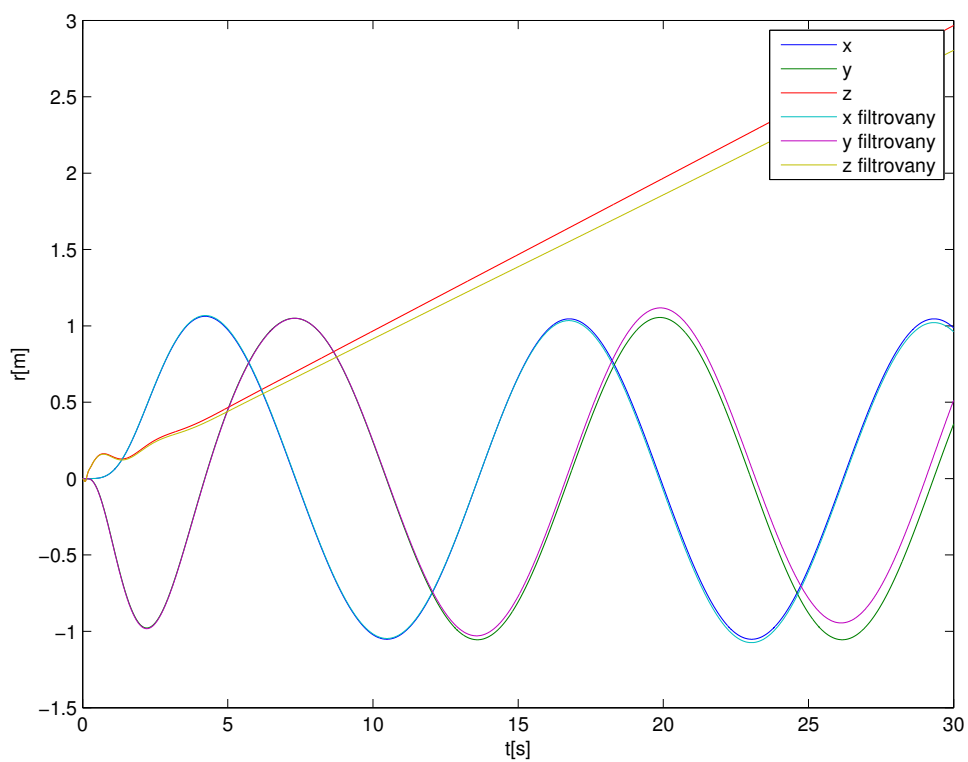
Obrázek 5.12: Úhlová rychlost kvadrotorové helikoptéry a její odhad při letu do cílového bodu



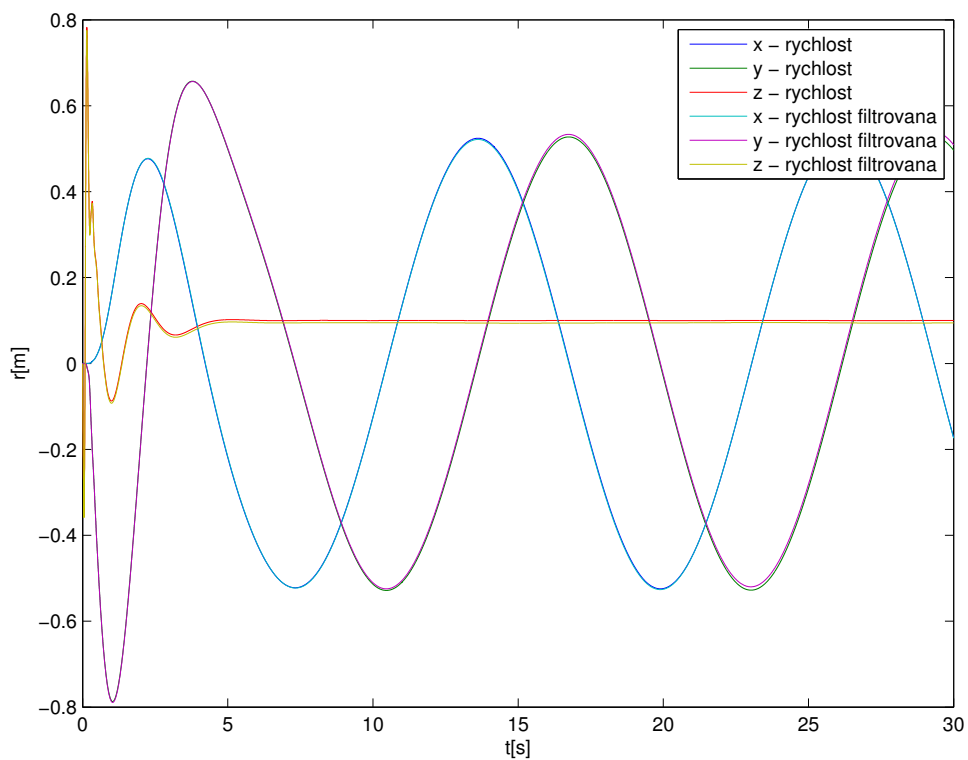
Obrázek 5.13: Trajektorie letu kvadrotorové helikoptéry a její odhad při letu do cílového bodu

Odhad stavu kvadrotorové helikoptéry při letu po generované trajektorii

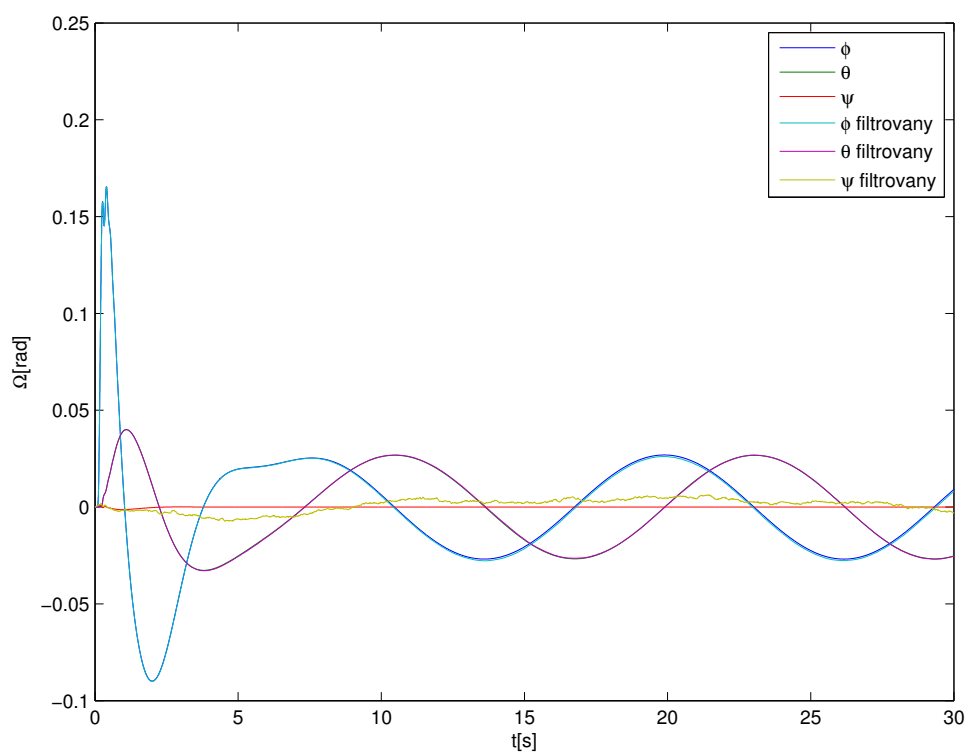
Pro odhad stavu na datech ze simulovaného letu po požadované trajektorii bylo zvoleno opět \bar{x}_0 na základě počátečních podmínek simulace z části 5.3.2. Matice P_0 byla použita stejná jako v 5.3.3. Data, na nichž byl prováděn odhad byla opět zašuměna jako v části 5.3.2. Výsledný odhad a nezašuměná data jsou na obrázcích 5.14, 5.15, 5.16, 5.17 a 5.18.



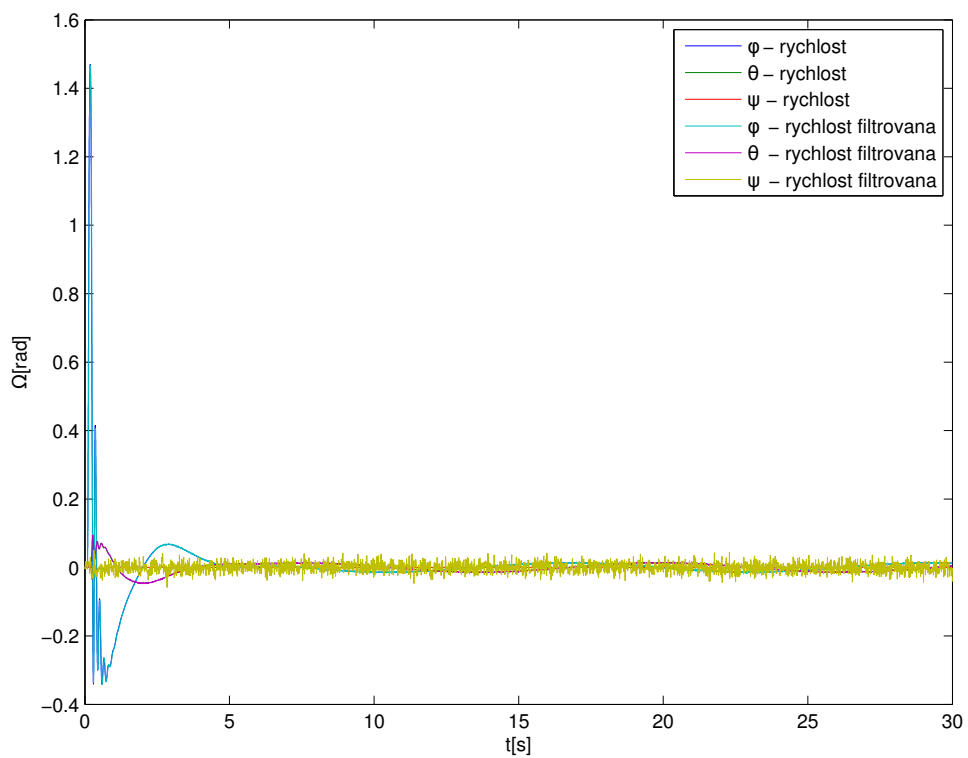
Obrázek 5.14: Poloha kvadrotorové helikoptéry a její odhad při letu po trajektorii



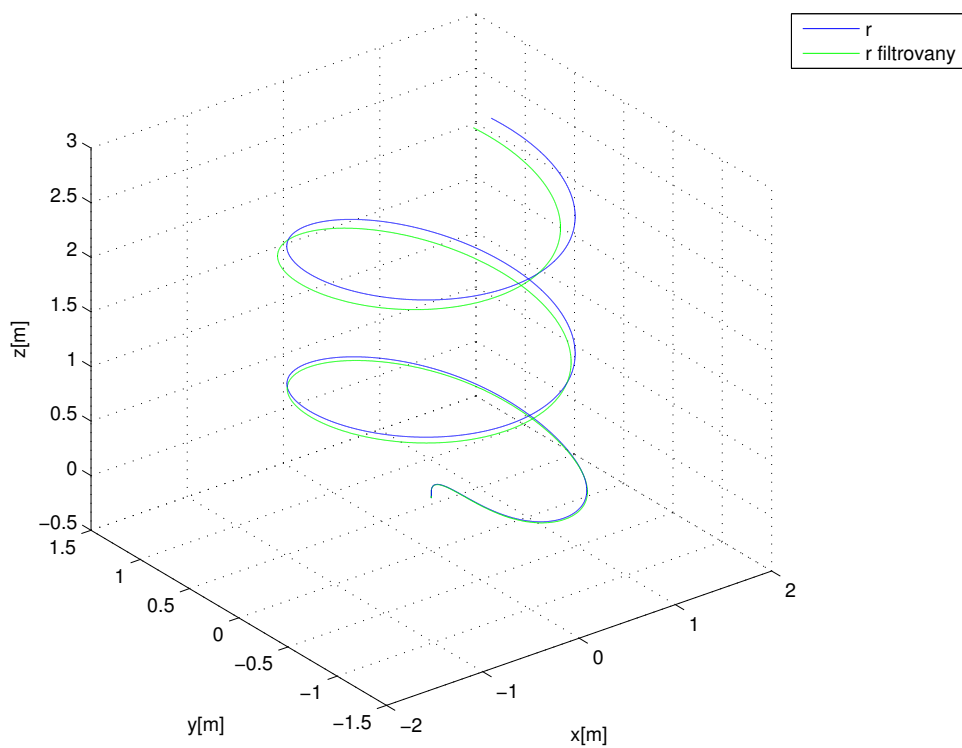
Obrázek 5.15: Rychlost kvadrotorové helikoptéry a její odhad při letu po trajektorii



Obrázek 5.16: Orientace kvadrotorové helikoptéry a její odhad při letu po trajektorii



Obrázek 5.17: Úhlová rychlost kvadrotorové helikoptéry a její odhad při letu po trajektorii



Obrázek 5.18: Trajektorie letu kvadrotorové helikoptéry a její odhad při letu po trajektorii

6 Závěr

V této práci byl popsán druh multirotoru, který se nazývá kvadrotorová helikoptéra. Bylo popsáno chování kvadrotorové helikoptéry a následně byl navržen matematický model, který byl rozšířen o popis některých aerodynamických jevů. Dále byl popsán algoritmus, který slouží k odhadu stavu kvadrotorové helikoptéry. Algoritmus byl doplněn diskretním nelineárním modelem kvadrotorové helikoptéry. Poté byl navržen regulátor rotoru a řídicí algoritmy orientace a polohy kvadrotorové helikoptéry.

Matematický model kvadrotorové helikoptéry, který popisuje let kvadrotorové helikoptéry v závislosti na otáčkách rotorů, byl vytvořen za účelem použití při simulacích, k návrhu algoritmů řídicích let helikoptéry a za účelem návrhu algoritmu pro odhad stavu helikoptéry. Model byl doplněn o jednoduchý popis odporu prostředí a popis vlivu třepotání listů rotorů na let kvadrotorové helikoptéry. Tento model by mohl být dále rozšířen o další prvky (například o vliv tlaku vzduchu na tah rotorů, či zvýšení tahu rotorů při letu nízko nad zemí).

Algoritmus pro odhad stavu kvadrotorové helikoptéry byl otestován v simulaci na zkušebních výstupech vytvořeného modelu a byla ověřena funkčnost při odhadu stavu helikoptéry na základě těchto dat. Algoritmus by mohl být doplněn o podrobnější diskretní model, nebo upraven pro zpracování měření z dalších typů senzorů.

Návrh řídicího algoritmu kvadrotorové helikoptéry byl zaměřen na sledování zadané trajektorie. Návrh byl rozdělen do několika samostatných celků. Nejprve byl navržen diskretní PID regulátor pro řízení otáček rotorů, který byl rozšířen o filtraci derivační složky a mechanismus proti unášení integrační složky. Následně byl popsán výpočet požadovaných otáček rotorů na základě požadovaného tahu a požadovaných momentů síly. Nakonec byl řídicí systém doplněn o řízení orientace a polohy helikoptéry, které bylo navrženo na základě nelineárních vztahů z matematického modelu kvadrotorové helikoptéry. Jádrem řízení orientace a polohy bylo několik LQ regulátorů. Spojením těchto algoritmů vznikl algoritmus pro řízení letu kvadrotorové helikoptéry do zadaného bodu, či podle generované trajektorie. Výsledný řídicí algoritmus byl otestován při simulacích v prostředí *Simulink* a splnil požadavky, které na něj byly kladeny.

Zdrojové soubory vytvořeného dokumentu, část použité literatury a soubory, které byly vytvořeny v prostředí *Matlab* a *Simulink* jsou přiloženy na CD.

Postupy které jsou popsány v této práci, poslouží zejména při aplikaci na reálné kvadrotorové helikoptěře. Také je možné využít algoritmus řízení letu k tvorbě složitějšího algoritmu, který by pomocí rozpoznávání obrazu detekoval objekty, které se nacházejí na požadované letové trajektorii a prováděl úpravu požadované trajektorie za účelem obletu objektu. Dále je možné doplnit řídicí algoritmus o algoritmus, který by prováděl mapování nějaké oblasti.

Literatura

- [1] BLANCO, José-Louis. *A tutorial on SE(3) transformation parameterizations and on-manifold optimization*. Malaga: University of Malaga, 2010, 57 s. Dostupné z: http://mapir.isa.uma.es/~jlblanco/papers/jlblanco2010geometry3D_techrep.pdf
- [2] BOUČEK, Zdeněk. *Návrh simulačního modelu a řízení kvadrotorové helikoptéry*. Plzeň, 2013. Bakalářská práce (Bc.). Západočeská univerzita v Plzni, Fakulta aplikovaných věd. Vedoucí práce Miroslav Flídr.
- [3] DUNÍK, Jindřich, Miroslav ŠIMANDL a Ondřej STRAKA. 2012. *Unscented Kalman Filter: Aspects and Adaptive Setting of Scaling Parameter*. In: IEEE Transactions on Automatic Control [online]. roč. 57, s. 2411-2416 [cit. 2015-05-13]. DOI: 10.1109/TAC.2012.2188424. ISSN 0018-9286. Dostupné z: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6155073>
- [4] Example of the UT for mean and covariance propagation. a) actual, b) first-order linearization (EKF), c) UT, [obrázek] In: *The Unscented Kalman Filter* [online]. Center for Spoken Language Understanding, 2000 [vid. 2015-05-05]. Dostupné z: <http://www.cslu.ogi.edu/nsl/ukf/node6.html>
- [5] JONES, Eric M. a Paul FJELD. *Gimbal Angles, Gimbal Lock, and a Fourth Gimbal for Christmas*. NASA [online]. 2000 [cit. 2015-02-15]. Dostupné z: <http://www.hq.nasa.gov/alsj/gimbals.html>
- [6] JULIER, S.J. a J.K. UHLMANN. 2004. *Unscented Filtering and Nonlinear Estimation*. In: Proceedings of the IEEE [online]. s. 401-422 [cit. 2015-05-02]. DOI: 10.1109/JPROC.2003.823141. ISSN 0018-9219. Dostupné z: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1271397>
- [7] HAVLENA, Vladimír a Jan ŠTECHA. *Moderní teorie řízení*. Praha: Ediční středisko ČVUT, 1999. ISBN 80-01-02095-9. Dostupné z: https://moodle.dce.fel.cvut.cz/pluginfile.php/6198/mod_page/content/20/Moderni_teorie_rizeni.pdf
- [8] HUANG, Haomiao, G.M. HOFFMANN, S.L. WASLANDER a C.J. TOMLIN. *Aerodynamics and control of autonomous quadrotor helicopters in aggressive maneuvering*. In: 2009 IEEE International Conference on Robotics and Automation [online]. IEEE, 2009, s. 3277-3282 [cit. 2015-02-15]. ISBN 978-1-4244-2788-8. DOI: 10.1109/ROBOT.2009.5152561. Dostupné z: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5152561>

- [9] KAZANTZIS, N., K.T. CHONG, J.H. PARK a A.G. PARLOS. 2003. Control-relevant discretization of nonlinear systems with time-delay using Taylor-Lie series. *Proceedings of the 2003 American Control Conference, 2003* [online]. IEEE, (1): 149-154 [cit. 2014-12-17]. DOI: 10.1109/ACC.2003.1238929. ISBN 0-7803-7896-2. ISSN 0743-1619. Dostupné z: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1238929>
- [10] MELICHAR, Jiří. *Lineární systémy 1: Učební text*. 2011. Dostupné z: http://www.kky.zcu.cz/uploads/courses/l1/LS1_Ucebni_texty_2011.pdf
- [11] Metody identifikace systémů z přechodových charakteristik. *Katedra automatizační techniky a řízení* [online]. [cit. 2015-02-26]. Dostupné z: http://www.352.vsb.cz/uc_texty/Identifikace/str/metody.htm#ma3
- [12] Quadcopter Control-Function Layers. 2013. *HefnyCopter Quadcopter Firmware* [online]. [cit. 2015-05-03]. Dostupné z: http://hefnycopter.net/index.php?option=com_content&view=article&id=22:quadcopter-control-function-layers&catid=9&Itemid=103
- [13] SCHLEGEL, Miloš. *Průmyslové PID regulátory: tutorial*. In: [online]. REX Controls s.r.o., 2003 [cit. 2015-02-20]. Dostupné z: http://matlab.fei.tuke.sk/raui/doc/PIDTutor_CZ.pdf
- [14] STACHNISS, Cyrill. 2013. *Robot Mapping: Unscented Kalman Filter* [online]. [cit. 2015-04-28]. Dostupné také z: <http://ais.informatik.uni-freiburg.de/teaching/ws12/mapping/pdf/slam05-ukf.pdf>
- [15] STACHNISS, Cyrill. *SLAM Course - 06 - Unscented Kalman Filter (2013/14; Cyrill Stachniss)*. In: *Youtube*[online]. 11. 11. 2013 [vid. 2015-02-18] Dostupné z: <https://www.youtube.com/watch?v=DwDzmweTKsQ&index=1>

A Implementace funkcí v Matlabu

V této části budou uvedeny některé funkce, pomocí nichž byl prováděn odhad stavu kvadrotorové helikoptéry.

A.1 Unscentovaný Kalmanův filtr

```
1 function [ x, P ] = ukf( x, u, uold, P, z, Q, R, alpha, beta,
    kappa )
2 % Unscented Kalman Filter for estimation of quadrotor helicopter
    's state
3 %
4 % based on UKF by Yi Cao (copyright enclosed in licence.txt)
5 %
6 % input:
7 %     x - last known/estimated state
8 %     u - known input
9 %     uold - previous known input
10 %     P - last known/estimated covariance matrix
11 %     z - vector of measurement
12 %     Q - covariance matrix of discrete model
13 %     R - covariance matrix of measurement
14 %     alpha - parameter for distribution of sigma-points
15 %     beta - parameter for distribution of sigma-points
16 %     kappa - parameter for distribution of sigma-points
17 %
18 % output:
19 %     x - new estimated state
20 %     P - new covariance matrix of state estimation
21 %
22
23     L = length(x);           % length of state vector
24     m = length(z);         % number of measurements
25     lambda = alpha^2*(L+kappa)-L; % calc lambda (how far
        sigma points will be from mean)
26     c = L+lambda;
27     Wm = [ lambda/c 0.5/c+zeros(1,2*L) ]; % weights of mean
28     Wc = Wm;               % weights of covariance
```

```

29     Wc(1) = Wc(1) + (1 - alpha^2 + beta);           % first weight
        Wc differs from Wm
30     c = sqrt(c);
31     X = sigmas(x,P,c); % calc sigma points around x
32     [x1,X1,P1,X2] = utf(X, u, uold, Wm, Wc, L, Q); %
        unscented transform of process
33     [z1,Z1,P2,Z2] = uth(X1, Wm, Wc, m, R); % unscented
        transform of measurement
34     P12 = X2*diag(Wc)*Z2';
35     K = P12*inv(P2); % calc Kalman gain
36     x = x1+K*(z-z1); % calc new estimation of state vector
37     P = P1 - K*P12'; % calc new covariance matrix of state
38 end
39
40 function [ y, Y, P, Y1 ] = utf( X, u, uold, Wm, Wc, n, R )
41 % Unscented Transformation of process
42
43     L = size(X,2); % number of sigma points
44     y = zeros(n,1);
45     Y = zeros(n,L);
46     for k = 1:L
47         Y(:,k) = quadmodel(X(:,k), u, uold); %
            transformed sampling points
48         y = y + Wm(k)*Y(:,k); % transformed mean
49     end
50     Y1 = Y - y(:,ones(1,L)); % transformed deviation
51     P = Y1*diag(Wc)*Y1' + R; % transformed covariance
52 end
53
54 function [ y, Y, P, Y1 ] = uth( X, Wm, Wc, n, R )
55 % Unscented Transformation of measurement
56
57     L = size(X,2); % number of sigma points
58     y = zeros(n,1);
59     Y = zeros(n,L);
60     for k = 1:L
61         Y(:,k) = hmeas(X(:,k)); % transformed sampling
            points
62         y = y + Wm(k)*Y(:,k); % transformed mean
63     end
64     Y1 = Y - y(:,ones(1,L)); % transformed
        deviation
65     P = Y1*diag(Wc)*Y1' + R; % transformed
        covariance
66 end
67

```

```

68 function X = sigmas ( x, P, c )
69 % calculates sigma points around reference point x
70 % input:
71 %     x - state vector
72 %     P - covariance matrix
73 %     c - parameter of unscented transformation
74 % output:
75 %     X - sigma points
76 %
77
78     A = c*chol(P)';
79     Y = x(:, ones(1, length(x)));
80     X = [x Y+A Y-A];           % sigma points
81 end

```

A.2 Diskrétní model kvadrotorové helikoptéry

A.2.1 Diskretizace spojitého modelu

```

1 % Derivation of mathematical model of the quadrotor helicopter
  by Taylor method
2 %     author: Zdenek Boucek
3
4 %% this script do the discretization of nonlinear model of
  quadrotor helicopter
5 %
6 % model is derived in two parts , first part is orientation part
  in xklori
7 % and second is position part in xklpos
8
9 clc;
10 clear all;
11
12 %% parameters of discretization
13 syms T real; % period of mathematical model
14 syms x1 x2 x3 x4 x5 x6 x7 x8 x9 x10 x11 x12 g m Ix Iy Iz u1 u2
  u3 u4 R real; % model variables
15 params = [g m Ix Iy Iz]; % parameters of model [gravitational
  acc , mass ,
16
  % moment of inertia in x', y', and
  z']
17 xk=[x1 x2 x3 x4 x5 x6]; % state vector [x x' y y' z z'] in
  position part
18
  % or [phi phi' theta theta' psi
  psi'] in
19 % orientation part

```

```

20 u = [u1 u2 u3 u4]; % input of model [thrust, moment on phi,
    moment on theta, moment on psi],
21 % in xk1pos used as [thrust in x', thrust
    in y' and thrust in z']
22 l = 2; % degree of Taylor approximation
23
24
25 % derivation of orientation model of quadrotor helicopter
26 xk1ori = quadrotor_orientation_theory(xk(7:12)', u', params, T, l);
27
28 % derivation of position model of quadrotor helicopter without
    input
29 xk1pos = quadrotor_position_theory(xk(1:6)', u', params, T, l);

```

```

1 function [ xk1 ] = quadrotor_position_theory( xk, u, params, T,
    l )
2 %% QUADROTOR_POSITION_THEORY discretize position part of
    quadrotor discrete model
3 % input:
4 %   xk - state variables
5 %   u - system input variables
6 %   params:
7 %       [g, m, Ix, Iy, Iz] -> [gravitational acc, mass,
8 %                               moment of inertia in x', y', and
    z']
9 %   T - period of discret model
10 %   l - maximum degree of Taylor expansion series for
    discretization
11 % output:
12 %   xk1 - discretized model of position part of quadrotor
13
14 A = [xk(2);
15      0;
16      xk(4);
17      0;
18      xk(6);
19      -params(1)] + [0;
20      u(1);
21      0;
22      u(2);
23      0;
24      u(3)]/params(2);
25 fg = A;
26 xk1 = zeros(6,1);
27
28 for i = 1:l
29     xk1 = xk1 + A*T^i/factorial(i);
30     if i == l
31         break;
32     else
33         A = jacobian(A,xk)*fg;
34     end
35 end
36
37 xk1 = xk1 + xk;
38
39 end

```

```

1 function [ xk1 ] = quadrotor_orientation_theory( xk, u, params,
    T, l )
2 %% QUADROTOR_ORIENTATION_THEORY discretize orientation part of
    quadrotor discrete model
3 % input:
4 %   xk - state variables
5 %   u - system input variables
6 %   params:
7 %       [g, m, Ix, Iy, Iz] -> [gravitational acc, mass,
8 %                               moment of inertia in x', y', and
9 %                               z']
9 %   T - period of discret model
10 %   l - maximum degree of Taylor expansion series for
    discretization
11 % output:
12 %   xk1 - discretized model of orientation part of quadrotor
13
14 A = [xk(2);
15      xk(4)*xk(6)*(params(4)-params(5))/params(3);
16      xk(4);
17      xk(2)*xk(6)*(params(3)-params(5))/params(4);
18      xk(6);
19      xk(2)*xk(4)*(params(3)-params(4))/params(5)] + [0;
20      u(2)/params(3);
21      0;
22      u(3)/params(4);
23      0;
24      u(4)/params(5)];
25 fg = A;
26 xk1 = zeros(6,1);
27
28 for i = 1:l
29     xk1 = xk1 + A*T^i/factorial(i);
30     if i == l
31         break;
32     else
33         A = jacobian(A,xk)*fg;
34     end
35 end
36
37 xk1 = xk1 + xk;
38
39 end

```

A.3 Diskrétní model kvadrotorové helikoptéry

```
1 function [ xnew ] = quadmodel( x, w, wold)
2 %QUADMODEL simulate behavior of quadrotor helicopter in discrete
   time
3 % returns new state x (position and orientation of quadrotor),
4 % which is calculated with use of last known state x (in input)
5 % w (current rotor speed vector) and wold (previous rotor speed
   vector)
6
7 %% parameters of quadrotor helicopter
8 kT = 2.98e-6; % coefficient of the rotor thrust [-]
9 kM = 1.14e-7; % coefficient of the rotor torque [-]
10 l = 0.225; % length of the arm [m]
11 Im = 3.357e-5; % coefficient of the rotor inertia [-]
12 g = 9.81; % gravitational acceleration [m.s^2]
13 m = 0.468; % mass of helicopter [kg]
14 Ix = 4.856e-3; % moment of inertia of x' axis [kg.m^2]
15 Iy = 4.856e-3; % moment of inertia of y' axis [kg.m^2]
16 Iz = 8.801e-3; % moment of inertia of z' axis [kg.m^2]
17 T = 0.01; % period discrete model [s]
18
19 u = quadinput( w, wold, [kT, kM, l, Im] ); % calculate
   thrust and torque
20 [ xnew ] = discretemodel( x, u, [g, m, Ix, Iy, Iz, T] ); %
   calculate new state of helicopter
21
22 end
```

```

1 function [ u ] = quadinput( w, w_old, params )
2 %QUADINPUT returns generated torque and thrust of quadrotor
   helicopter based on angular speed of rotors
3 % input:
4 %     w ~ vector of rotor speed
5 %     w_old ~ previous vector of rotor speed
6 %     params ~ parameters of quadrotor helicopter
7 % output:
8 %     u(1) ~ total thrust generated by rotors of helicopter
9 %     u(2) ~ torque around x' axis
10 %     u(3) ~ torque around y' axis
11 %     u(4) ~ torque around z' axis
12 %
13
14 %% parameters of quadrotor helicopter
15 T = 0.01;
16 kT = params(1); % coefficient of the rotor thrust
17 kM = params(2); % coefficient of the rotor moment
18 l = params(3); % length of the arm
19 if length(1) < 0
20     error('discretemodel: Wrong length of helicopter arm !!.
           ');
21 end
22 Im = params(4); % coefficient of the rotor inertia
23
24 u = zeros(1,4);
25 u(1) = kT*sum(w.^2); % T - rotor thrust
26 u(2) = l*kT*(-w(2)^2 + w(4)^2); % Mroll - torque around
   x' axis
27 u(3) = l*kT*(-w(1)^2 + w(3)^2); % Mpitch - torque around
   y' axis
28 u(4) = 0;
29 for i = 1:length(w)
30     u(4) = u(4) + (((-1)^(i+1))*(kM*w(i)^2 + Im*(w(i) - w_old(i)
           )/T)); %Myaw - torque around z' axis
31 end
32
33 end

```



```

1 function [ x1 ] = discretemodel( x, u, params)
2 %DISCRETEMODEL simulates motion dynamics of quadrotor helicopter
3 % Simulation depends on input u where:
4 %     u(1) ~ total thrust generated by rotors of helicopter
5 %     u(2) ~ torque around x' axis
6 %     u(3) ~ torque around y' axis
7 %     u(4) ~ torque around z' axis
8 % last known state x where:
9 %     position:
10 %         x(1) ~ position of helicopter in x
11 %         x(2) ~ speed helicopter in x
12 %         x(3) ~ position helicopter in y
13 %         x(4) ~ speed helicopter in y
14 %         x(5) ~ position helicopter in z
15 %         x(6) ~ speed helicopter in z
16 %     orientation:
17 %         x(7) ~ angle phi
18 %         x(8) ~ speed of roll
19 %         x(9) ~ angle theta
20 %         x(10) ~ speed of pitch
21 %         x(11) ~ angle psi
22 %         x(12) ~ speed of yaw
23 % Output of simulation is new state , which is returned in vector
    x1
24 %
25
26 if length(x) < 12
27     error('discretemodel: Cannot execute! Short vector of
            state x. ');
28 end
29 if length(u) < 4
30     error('discretemodel: Cannot execute! Short input vector
            u. ');
31 end
32 if length(params) < 6
33     error('discretemodel: Cannot execute! Short vector of
            parameters params. ');
34 end
35
36 %% parameters of quadrotor helicopter
37 g = params(1); % gravitational acceleration [m.s^-2]
38 m = params(2); % mass of helicopter [kg]
39 if m <= 0
40     error('discretemodel: Wrong mass of helicopter!');
41 end
42 Ix = params(3); % moment of inertia of x' axis [kg.m^2]

```

```

43 Iy = params(4); % moment of inertia of y' axis [kg.m^2]
44 Iz = params(5); % moment of inertia of z' axis [kg.m^2]
45 T = params(6); % period of discrete model [s]
46 x1 = zeros(1, length(x));
47 %% compute new state
48 if norm([x(7),x(9),x(11)]) <= 1e-6
49     % if euler angles are near [0,0,0]^T
50     % orientation part
51     x1(7) = x(7) + T*x(8) + (T^2*(u(2)/Ix + (x(10)*x(12)*(Iy
52         - Iz))/Ix))/2;
53     x1(8) = x(8) + (T^2*((x(12)*(u(3)/Iy + (x(8)*x(12)*(Ix -
54         Iz))/Iy)*(Iy - Iz))/Ix + (x(10)*(u(4)/Iz + (x(8)*x
55         (10)*(Ix - Iy))/Iz)*(Iy - Iz))/Ix))/2 + T*(u(2)/Ix +
56         (x(10)*x(12)*(Iy - Iz))/Ix);
57     x1(9) = x(9) + T*x(10) + (T^2*(u(3)/Iy + (x(8)*x(12)*(Ix
58         - Iz))/Iy))/2;
59     x1(10) = x(10) + (T^2*((x(12)*(u(2)/Ix + (x(10)*x(12)*(
60         Iy - Iz))/Ix)*(Ix - Iz))/Iy + (x(8)*(u(4)/Iz + (x(8)*
61         x(10)*(Ix - Iy))/Iz)*(Ix - Iz))/Iy))/2 + T*(u(3)/Iy +
62         (x(8)*x(12)*(Ix - Iz))/Iy);
63     x1(11) = x(11) + T*x(12) + (T^2*(u(4)/Iz + (x(8)*x(10)*(
64         Ix - Iy))/Iz))/2;
65     x1(12) = x(12) + (T^2*((x(10)*(u(2)/Ix + (x(10)*x(12)*(
66         Iy - Iz))/Ix)*(Ix - Iy))/Iz + (x(8)*(u(3)/Iy + (x(8)*
67         x(12)*(Ix - Iz))/Iy)*(Ix - Iy))/Iz))/2 + T*(u(4)/Iz +
68         (x(8)*x(10)*(Ix - Iy))/Iz);
69     % position part
70     x1(1) = x(1) + x(2)*T;
71     x1(2) = x(2);
72     x1(3) = x(3) + x(4)*T;
73     x1(4) = x(4);
74     x1(5) = x(5) + T*x(6) - (T^2*(g - u(1)/m))/2;
75     x1(6) = x(6) - T*(g - u(1)/m);
76 else
77     % orientation part
78     x1(7) = x(7) + T*x(8) + (T^2*(u(2)/Ix + (x(10)*x(12)*(Iy
79         - Iz))/Ix))/2;
80     x1(8) = x(8) + (T^2*((x(12)*(u(3)/Iy + (x(8)*x(12)*(Ix -
81         Iz))/Iy)*(Iy - Iz))/Ix + (x(10)*(u(4)/Iz + (x(8)*x
82         (10)*(Ix - Iy))/Iz)*(Iy - Iz))/Ix))/2 + T*(u(2)/Ix +
83         (x(10)*x(12)*(Iy - Iz))/Ix);
84     x1(9) = x(9) + T*x(10) + (T^2*(u(3)/Iy + (x(8)*x(12)*(Ix
85         - Iz))/Iy))/2;
86     x1(10) = x(10) + (T^2*((x(12)*(u(2)/Ix + (x(10)*x(12)*(
87         Iy - Iz))/Ix)*(Ix - Iz))/Iy + (x(8)*(u(4)/Iz + (x(8)*
88         x(10)*(Ix - Iy))/Iz)*(Ix - Iz))/Iy))/2 + T*(u(3)/Iy +

```

```

70         (x(8)*x(12)*(Ix - Iz))/Iy);
x1(11) = x(11) + T*x(12) + (T^2*(u(4)/Iz + (x(8)*x(10)*
71         Ix - Iy))/Iz))/2;
x1(12) = x(12) + (T^2*((x(10)*(u(2)/Ix + (x(10)*x(12)*
72         Iy - Iz))/Ix)*(Ix - Iy))/Iz + (x(8)*(u(3)/Iy + (x(8)*
x(12)*(Ix - Iz))/Iy)*(Ix - Iy))/Iz))/2 + T*(u(4)/Iz +
73         (x(8)*x(10)*(Ix - Iy))/Iz);
% position part
R = rotation([x(7),x(9),x(11)]); % computer
74         rotation matrix from new angles [phi, theta, psi]
F = R*[0;0;u(1)]; % rotate thrust with rotation
75         matrix
x1(1) = x(1) + x(2)*T + (F(1)*T^2)/(2*m);
76         x1(2) = x(2) + (T*F(1))/m;
77         x1(3) = x(3) + x(4)*T + (F(2)*T^2)/(2*m);
78         x1(4) = x(4) + (T*F(2))/m;
79         x1(5) = x(5) + T*x(6) - (T^2*(g - F(3)/m))/2;
80         x1(6) = x(6) - T*(g - F(3)/m);
81 end
82
83 end

```