

Bakalářská práce



České
vysoké
učení technické
v Praze

F3

Fakulta elektrotechnická
Katedra kybernetiky

Odhad a optimalizace spotřeby energie během letu UAV

Miroslav Kovář

Vedoucí: Ing. František Nekovář
Obor: Kybernetika a robotika
Květen 2022

Poděkování

Děkuji v první řadě mému vedoucímu za cenné rady a za zpětnou vazbu během práce na tomto projektu. Dále bych chtěl poděkovat mé rodině a přátelům za podporu po celou dobu studia. V neposlední řadě děkuji ČVUT za možnost získat mnoho nových znalostí a dovedností.

Prohlášení

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

V Praze, 20. května 2022

Abstrakt

Cílem práce je navrhnout způsob minimalizace spotřeby baterie během letu bezpilotní kvadrokoptéry (UAV). Úloha se skládá z několika dílčích částí. V prvé řadě je potřeba spotřebu baterie v závislosti na letové trajektorii výpočtem určit pomocí matematického modelu spotřeby, který může být závislý i na odhadu aerodynamických vlastností UAV. Poté je možné trajektorii podle kritéria spotřeby optimalizovat. Pro potřeby použití optimalizace během letu UAV, tzn. provádět výpočet v reálném čase, je také potřeba najít vhodnou aproximaci odhadu spotřeby během letu i bez použití kompletního dynamického či aerodynamického modelu. Výsledky práce mohou posloužit pro prodloužení letového času UAV používajících MRS systém a pro experimentální odhad spotřeby v závislosti na konstrukci UAV a jeho aerodynamických vlastnostech.

Klíčová slova: spotřeba, odhad, optimalizace, UAV, kvadrokoptéra, Matlab

Vedoucí: Ing. František Nekovář
Multi-robotické systémy - ČVUT FEL,
Karlovo náměstí 13,
Praha 2

Abstract

The aim of this work is to design a way to minimize the battery consumption of unmanned quadcopter (UAV) flight. The task consists of several parts. First of all, it is necessary to determine the battery consumption depending on the flight trajectory by calculation using a mathematical consumption model, which may also depend on the estimation of the aerodynamic properties of the UAV. Then the trajectory can be optimized according to the consumption criteria. For the needs of using in-flight optimization UAV, ie. to perform a real-time calculation, it is also necessary to find a suitable approximation of the in-flight consumption estimation even without the use of a complete dynamic or aerodynamic model. The results of the work can be used to extend the flight time of UAVs using the MRS system and for experimental estimation of consumption depending on the design of the UAV and its aerodynamic properties.

Keywords: consumption, estimation, optimization, UAV, quadcopter, Matlab

Title translation: Estimation and Optimization of Power Consumption during UAV Flight

Obsah

| | |
|---|-----------|
| 1 Úvod | 1 |
| 2 Model spotřeby energie | 5 |
| 2.1 Dynamický model kvadroptéry | 5 |
| 2.2 Elektrický model BLDC motoru . | 8 |
| 2.3 Aproximace spotřeby energie | 9 |
| 3 Odhad spotřeby energie pro danou trajektorii | 11 |
| 4 Optimalizace trajektorie letu UAV | 17 |
| 4.1 Formulace optimalizační úlohy . . | 17 |
| 4.2 Numerická optimalizace v Matlabu | 19 |
| 4.3 Optimalizace v reálném čase . . . | 22 |
| 5 Závěr | 25 |
| Literatura | 27 |
| Zadání práce | 29 |

Obrázky

| | |
|--|----|
| 1.1 Létající doručovací dron <i>Amazon Prime Air</i> [1] | 2 |
| 1.2 Kvadrokoptéra <i>DJI Phantom 2</i> [2] | 3 |
| 2.1 Schéma polohy a orientace kvadrokoptéry vzhledem ke světovému souřadnému systému[3] | 6 |
| 2.2 Schéma pohybu kvadrokoptéry podle rychlosti otáček jednotlivých rotorů[4]. Čím širší je šipka v daném rotoru, tím rychleji se rotor otáčí. | 6 |
| 2.3 Elektrické schéma BLDC motoru kvadrokoptéry[5] | 8 |
| 3.1 Trasa letu kvadrokoptéry | 14 |
| 3.2 Tahy jednotlivých rotorů | 14 |
| 3.3 Okamžitá spotřeba energie jednotlivých rotorů. | 15 |
| 3.4 Kumulativní spotřeba energie jednotlivých rotorů. | 15 |
| 3.5 Celková spotřeba energie během letu | 16 |
| 4.1 Trajektorie vygenerovaná programem <i>FALCON.m</i> | 20 |
| 4.2 Tahy rotorů pro trajektorii vygenerovanou <i>FALCON.m</i> | 20 |

Tabulky

- 3.1 Hodnoty parametrů pro
kvadrokoptéru *DJI Phantom 2* ... 12
- 3.2 Hodnoty parametrů figurujících v
konstantách c_1, \dots, c_9 (2.14-2.22) . 13



Kapitola 1

Úvod

Autonomní létající drony (zkratkou UAV, z anglického *Unmanned Aerial Vehicle*) mají stále více využití a proto roste jejich obliba jak u odborné tak i u laické veřejnosti. Mohou mít připevněnou kameru či termokameru a být tak využity pro monitorování nějaké oblasti, ať už pro vojenské či civilní účely (např. hledání osob v lese, kontrola úrody na poli, inspekce vysokonapěťových stožárů, apod.). Létající drony s připevněnou kamerou najdou uplatnění i v kinematografii, kde se hodí pro natáčení scenerie z ptáčích perspektivy, např. pro dokumentární účely nebo sledování pohybujících se sportovců (běžců či jedoucích cyklistů)[6]. Autonomní létající drony mohou též přenášet např. poštovní balíčky na vzdálenosti několika stovek metrů či jednotek kilometrů, což je vhodné zejména v městském prostředí, kde mají výhodu v rychlosti doručení díky nezávislosti na pozemním provozu. O tento způsob doručování balíčků se pokouší americká doručovatelská společnost Amazon[7]. UAV pronikly i do sportovního prostředí, kde získávají oblibu závody létajících dronů[8].

Létající drony mají ovšem i řadu omezení, z nichž nejzásadnějším je letová výdrž, která se pohybuje v rozmezí 15 až 45 minut. Většina UAV má energii uloženou v lithium-iontových polymerových (LiPo) bateriích. Největšími spotřebiteli energie při letu dronu jsou motory, které roztáčejí vrtule, které se musí neustále točit, aby udržely stroj ve vzduchu. K úspoře energie a následnému zvýšení letové výdrže UAV je mnoho cest. Čím menší je hmotnost letounu, tím méně energie je potřeba vynaložit pro jeho udržení ve vzduchu. Pro snížení hmotnosti dronů se využívají lehké kompozitní materiály, například uhlíková vlákna, z nichž lze vyhotovit rám celého dronu i listy jednotlivých vrtulí. Dále je možné využít baterie s vyšší kapacitou, ovšem takové baterie mívají obvykle i vyšší hmotnost. Další cestou k úspoře energie je použití motorů s lepším



Obrázek 1.1: Létající doručovací dron *Amazon Prime Air*[1]

poměrem výkonu k hmotnosti. Na spotřebě energie se samozřejmě podílí i aerodynamické vlastnosti UAV, lze tedy upravit celkový tvar dronu tak, aby měl menší aerodynamický odpor. Některé studie se zabývaly možností výměny baterie za letu nebo prosté odhození vybité baterie během letu[9], což ovšem není z bezpečnostních důvodů a obav o životní prostředí vhodné pro širší použití v civilních aplikacích. Kromě těchto hardwarových úprav je možné provést i úpravy softwarové, tedy implementovat nějaký algoritmus, který naplánuje trajektorii UAV pro danou úlohu tak, aby byla spotřeba energie na baterii dronu při průletu danou trasou co nejmenší. Tímto způsobem úspory energie se zabývá tato práce.

Hledání trajektorie s minimální spotřebou energie je optimalizační úloha. Optimalizace je jednou z nejčastějších lidských činností, každý z nás hledá např. nejkratší či nejrychlejší cestu z jednoho místa na druhé, nebo způsob jak dosáhnout nějakého cíle s minimálním úsilím. Obecně je optimalizace snahou o nalezení minimální (nebo maximální) hodnoty účelové funkce, která určuje co chceme optimalizovat, za nějakých omezujících podmínek. Minimální (nebo maximální) hodnota účelové funkce za daných omezujících podmínek se nazývá optimum, nebo též optimální hodnota. Typickými příklady optimalizačních úloh jsou např. hledání nejkratší či nejrychlejší cesty autem z jednoho místa na druhé (minimalizace délky trasy nebo doby trvání jízdy), určení ceny produktu (maximalizace zisku) nebo návrh tvaru letadla (minimalizace aerodynamického odporu vzduchu, což je jedna z cest k minimalizaci spotřeby paliva). Pojem optimalizace má svůj význam i v informatice, kde se programátoři snaží optimalizovat svůj kód tak, aby program po spuštění pracoval co nejrychleji, spotřeboval co nejméně paměti nebo využíval co nejméně výpočetního výkonu počítače.



Obrázek 1.2: Kvadrokoptéra *DJI Phantom 2* [2]

V následujících kapitolách této práce je představen obecný matematický model spotřeby energie pro čtyř-rotorový autonomní létající dron (kvadrokoptéru) s ohledem na aerodynamické vlastnosti dronu, vlastnosti motorů a LiPo baterie. Tento model je posléze ilustrován na příkladu komerčně dostupného dronu *DJI Phantom 2*. Dále je představena účelová funkce - spotřeba energie na baterii během letu kvadrokoptéry. Poté je představen program odhadující spotřebu pro předem známou (ne nutně optimální) trajektorii, který je implementován v Matlabu. Dále je formulována optimalizační úloha a popsán způsob hledání optimální trajektorie pomocí nelineárního řešiče (anglicky *solver*) v Matlabu a nakonec program implementovaný v jazyce C pro výpočet optimální trajektorie v reálném čase (tedy použitelný během letu).

Kapitola 2

Model spotřeby energie

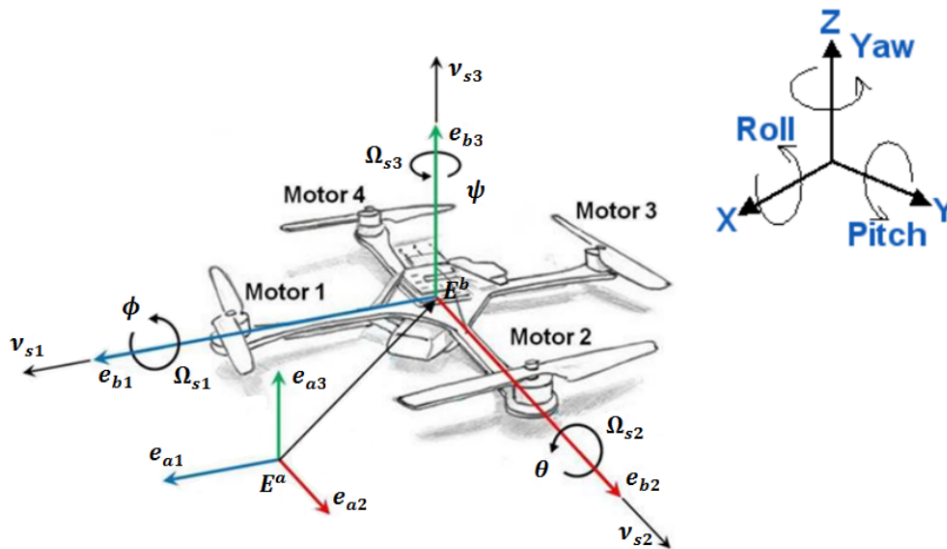
Spotřeba elektrické energie závisí jak na aerodynamických vlastnostech kvadroptéry, tak i na vlastnostech motorů pohánějících vrtule.

2.1 Dynamický model kvadroptéry

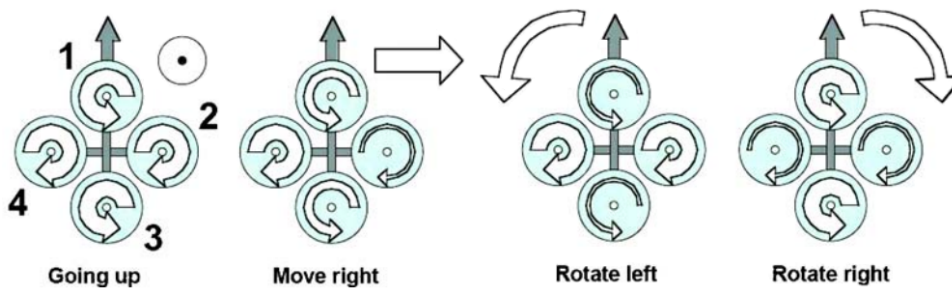
Kvadroptéra má čtyři rotory. Označíme-li jednotlivé rotory po směru hodinových ručiček čísly jedna až čtyři, pak první a třetí rotor se točí proti směru hodinových ručiček (v kladném směru kolem osy z) při pohledu shora a druhý a čtvrtý rotor se točí po směru hodinových ručiček (v záporném směru kolem osy z) při pohledu shora. Každý rotor se točí ve svém směru úhlovou rychlostí $\omega_j \geq 0$, kde $j \in \{1, 2, 3, 4\}$ je index rotoru.

Nechť vektor $q = [x, y, z]^T$ je pozice těžiště dronu vzhledem k (neměnnému) světovému souřadnému systému. Eulerovy úhly kvadroptéry, určující orientaci dronu v prostoru, jsou vyjádřeny jako $\Phi = [\phi, \theta, \psi]^T$, kde ϕ je úhel otočení kolem osy x (anglicky *roll*), θ je úhel otočení kolem osy y (anglicky *pitch*) a ψ je úhel otočení kolem osy z (anglicky *yaw* - ploché otáčky).

Letové manévry provádí kvadroptéra tak, že se každý rotor točí různou rychlostí. Aby kvadroptéra držela na místě (visela ve vzduchu) musí se všechny rotory točit takovou rychlostí ω_g , aby dron překonal tíhové zrychlení g , které jej táhne k zemi. Točí-li se rotory rychlostí nižší než ω_g , pak stroj



Obrázek 2.1: Schéma polohy a orientace kvadroptéry vzhledem ke světovému souřadnému systému[3]



Obrázek 2.2: Schéma pohybu kvadroptéry podle rychlosti otáček jednotlivých rotorů[4]. Čím širší je šipka v daném rotoru, tím rychleji se rotor otáčí.

klesá, točí-li se naopak rychleji, stroj stoupá vzhůru. Pokud se všechny rotory přestanou točit, stroj padá volným pádem k zemi. Za předpokladu slabého nebo žádného bočního větru se stroj pohybuje ve směru osy z (nahoru a dolů) tak, že se všechny rotory otáčejí stejnou rychlostí. Ploché otáčky (doleva nebo doprava kolem osy z bez náklonu kolem os x a y) se provádějí tak, že se vždy dva protilehlé rotory točí rychleji než zbylé dva rotory. Pohyb vpřed, vzad, vlevo nebo vpravo (ve směrech os x a y) se realizuje tak, že se jeden rotor točí rychleji než ostatní (nebo se dva rotory vedle sebe točí rychleji než zbylé dva). Při takových manévrech dochází k bočnímu náklonu dronu (kombinace náklonů kolem os x a y).

Kompletní dynamický model kvadrokoptéry[3, 4, 5] je:

$$\begin{cases} m\ddot{x} = (\sin \phi \sin \psi + \cos \phi \cos \psi \sin \theta)U_1, \\ m\ddot{y} = (\cos \phi \sin \theta \sin \psi - \cos \psi \sin \phi)U_1, \\ m\ddot{z} = (\cos \theta \cos \phi)U_1 - mg, \\ I_x\ddot{\phi} = (I_y - I_z)\dot{\theta}\dot{\psi} + \ell U_2 - J\dot{\theta}\Omega, \\ I_y\ddot{\theta} = (I_z - I_x)\dot{\phi}\dot{\psi} + \ell U_3 + J\dot{\phi}\Omega, \\ I_z\ddot{\psi} = (I_x - I_y)\dot{\phi}\dot{\theta} + U_4, \end{cases} \quad (1)$$

kde m je hmotnost kvadrokoptéry v kilogramech, $g = 9.8066 \text{ m/s}^2$ je gravitační zrychlení, I_x, I_y, I_z jsou setrvačnosti ve směrech x, y, z ; ℓ je vzdálenost rotoru od těžiště kvadrokoptéry, tedy délka ramene dronu, $J = J_m + J_L$ je celkový moment setrvačnosti motoru (součet momentů setrvačnosti motoru a zátěže - listů rotoru), $U_{1..4}$ jsou řídicí vstupy a Ω je rušení[4] pro něž platí:

$$U_1 = \kappa_b(\omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2), \quad (2.1)$$

$$U_2 = \kappa_b(\omega_2^2 - \omega_4^2), \quad (2.2)$$

$$U_3 = \kappa_b(\omega_3^2 - \omega_1^2), \quad (2.3)$$

$$U_4 = \kappa_\tau(\omega_1^2 - \omega_2^2 + \omega_3^2 - \omega_4^2), \quad (2.4)$$

$$\Omega = \omega_1 - \omega_2 + \omega_3 - \omega_4, \quad (2.5)$$

kde κ_b je faktor tahu rotoru a κ_τ je aerodynamický odpor vrtulí. Řídicí vstup U_1 je celkový tah rotorů, vstupy U_2, U_3 a U_4 jsou rotace *roll*, *pitch* a *yaw*.

Moment setrvačnosti listů rotoru J_L je

$$J_L = \frac{1}{4}n_B m_B (r - \epsilon)^2, \quad (2.6)$$

kde n_B je počet listů vrtule, m_B je hmotnost listu vrtule, r je poloměr otáčení vrtule (délka listu rotoru) a ϵ je poloměr osy vrtule (vnitřní poloměr otáčení, tedy poloměr té části rotoru, která je přímo spojena s motorem a která se neopírá o vzduch při letu).

Faktor tahu rotoru κ_b je

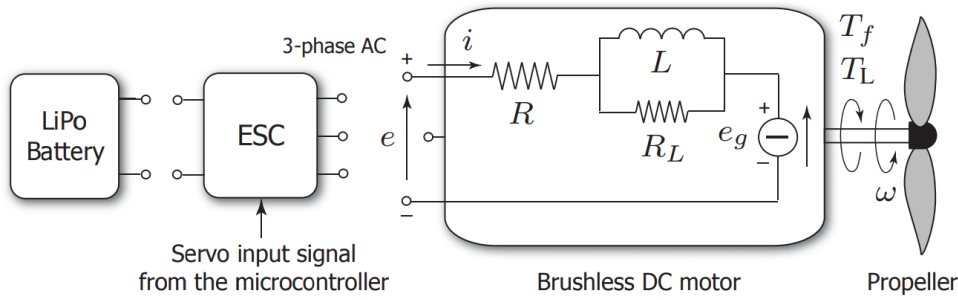
$$\kappa_b = C_T \rho A r^2, \quad (2.7)$$

kde C_T je koeficient tahu vrtule (bezrozměrný), který závisí na aerodynamických vlastnostech vrtule, ρ je hustota vzduchu a $A = \pi r^2$ je plocha, kterou zabírá točící se vrtule.

Aerodynamický odpor vrtulí κ_τ je

$$\kappa_\tau = C_Q \rho A r^3, \quad (2.8)$$

kde $C_Q = C_T \sqrt{C_T/2}$ je součinitel točivého momentu vrtule.



Obrázek 2.3: Elektrické schéma BLDC motoru kvadrokoptéry[5]

2.2 Elektrický model BLDC motoru

Bezkomutátorový stejnosměrný motor (zkratkou BLDC motor, z anglického *brushless direct current motor*) je použit pro pohánění rotorů ve většině kvadrokoptér. Model bateriemi napájeného BLDC motoru bere v úvahu energii rozptýlenou v odporovém a indukčním vinutí a energii potřebnou k překonání vnitřního a zátěžového tření[5].

Okamžitý proud v motoru $i(t)$ je dán vztahem[5]:

$$i(t) = \frac{1}{K_T} \left(T_f + T_L(\omega(t)) + D_f \cdot \omega(t) + (J_m + J_L) \frac{d\omega(t)}{dt} \right), \quad (2.9)$$

kde K_T je konstanta točivého momentu motoru, T_f je třecí moment motoru, $T_L(\omega(t))$ je na úhlové rychlosti závislý třecí moment zatížení způsobený tahem rotoru, D_f je koeficient viskózního tlumení, $\omega(t)$ je úhlová rychlost motoru a J_m a J_L jsou momenty setrvačnosti motoru a zátěže. Velikosti třecího momentu motoru a koeficientu viskózního tlumení jsou zanedbatelné.

Okamžitá napětí na motoru je dáno vztahem[5]:

$$u(t) = R \cdot i(t) + K_E \cdot \omega(t) + L \frac{di(t)}{dt}, \quad (2.10)$$

kde R a L jsou odpor a indukčnost fázového vinutí motoru a K_E je napěťová konstanta motoru. Za ustáleného stavu bude proud v motoru konstantní, derivace proudu bude tudíž nulová a poslední člen této rovnice tedy můžeme zanedbat. Rovnice se tedy zredukuje na tvar

$$u(t) = R \cdot i(t) + u_g(t), \quad (2.11)$$

kde $u_g(t) = K_E \cdot \omega(t)$ působí proti elektromotorické síle motoru (viz obrázek, v němž jsou - oproti českým zvyklostem označovat napětí jako u - napětí značena jako e).

2.3 Aproximace spotřeby energie

V předchozí sekci této kapitoly jsou uvedeny vzorce pro okamžité napětí $u(t)$ a okamžitý proud $i(t)$ na motoru kvadrokoptéry v čase t . Kvadrokoptéra má ovšem motory čtyři a spotřeba energie závisí na každém z nich. Energie je spotřebovávána v každém okamžiku letu, tedy od počátečního času t_0 do konečného času t_f . Celková spotřeba energie E je součtem všech okamžitých spotřeb (tedy integrálem) v každém okamžiku letu na všech motorech zároveň, tedy[5]

$$E = \int_{t_0}^{t_f} \sum_{j=1}^4 u_j(t) i_j(t) dt, \quad (2.12)$$

kde $j \in \{1, 2, 3, 4\}$ je index motoru.

Třecí moment zatížení T_L způsobený tahem rotoru je roven součinu faktoru aerodynamického tahu vrtulek κ_τ s kvadrátem úhlové rychlosti daného motoru, tedy[5]

$$T_L(\omega_j(t)) = \kappa_\tau \omega_j^2(t). \quad (2.13)$$

Po dosazení rovnic pro okamžité napětí (2.11) a okamžitý proud (2.9) s ohledem na vztah pro třecí moment zatížení T_L (2.13) lze rovnici pro celkovou spotřebovanou energii během letu kvadrokoptéry (2.12) přepsat jako[5]

$$E = \int_{t_0}^{t_f} \sum_{j=1}^4 [c_1 + c_2 \omega_j(t) + c_3 \omega_j^2(t) + c_4 \omega_j^3(t) + c_5 \omega_j^4(t) + c_6 \alpha_j(t) + c_7 \alpha_j^2(t) + c_8 \omega_j(t) \alpha_j(t) + c_9 \omega_j^2(t) \alpha_j(t)] dt, \quad (2)$$

kde $\alpha_j(t)$ je úhlové zrychlení motoru j a c_1, c_2, \dots, c_9 jsou konstanty závislé

na parametrech motorů a geometrii vrtulí, jejich hodnoty jsou dány vztahy:

$$c_1 = \frac{RT_f^2}{K_T^2}, \quad (2.14)$$

$$c_2 = \frac{T_f}{K_T} \left(\frac{2RD_f}{K_T} + K_E \right), \quad (2.15)$$

$$c_3 = \frac{D_f}{K_T} \left(\frac{RD_f}{K_T} + K_E \right) + \frac{2RT_f\kappa_\tau}{K_T^2}, \quad (2.16)$$

$$c_4 = \frac{\kappa_\tau}{K_T} \left(\frac{2RD_f}{K_T} + K_E \right), \quad (2.17)$$

$$c_5 = \frac{R\kappa_\tau^2}{K_T^2}, \quad (2.18)$$

$$c_6 = \frac{2RJ T_f}{K_T^2}, \quad (2.19)$$

$$c_7 = \frac{RJ^2}{K_T^2}, \quad (2.20)$$

$$c_8 = \frac{J}{K_T} \left(\frac{2RD_f}{K_T} + K_E \right), \quad (2.21)$$

$$c_9 = \frac{2RJ\kappa_\tau}{K_T^2}. \quad (2.22)$$

Kapitola 3

Odhad spotřeby energie pro danou trajektorii

Se znalostí v předchozí kapitole uvedeného matematického modelu spotřeby energie kvadrokoptéry jsem v Matlabu vytvořil program *energyConsumed.m*, který určí pro předem známou trajektorii její energetickou spotřebu. Tento program může sloužit k porovnání energetické náročnosti různých trajektorií.

Aby mohl program určit spotřebu konkrétní kvadrokoptéry, musí znát její specifické parametry vyplývající z její geometrie, vlastností motorů a aerodynamických vlastností. Já jsem si pro představení funkce programu vybral komerčně dostupnou, dnes již ovšem nevyráběnou kvadrokoptéru *DJI Phantom 2* [2], která se ovšem v parametrech příliš neliší od svých následovníků s pořadovými čísly 3 a 4. Tento dron jsem si vybral protože je velmi rozšířený a jsou dobře známy jeho technické parametry. Program by samozřejmě fungoval i pro jiné kvadrokoptéry, bylo by ovšem zapotřebí upravit hodnoty parametrů tak, aby odpovídaly konkrétnímu stroji. Pro kvadrokoptéru *DJI Phantom 2* jsou hodnoty parametrů uvedeny v následující tabulce (3.1).

S pomocí konstant z tabulky (3.1) lze určit hodnoty dalších parametrů, které figurují v konstantách c_1, \dots, c_9 (2.14-2.22) zmíněných v předchozí kapitole. Hodnoty těchto parametrů jsou uvedeny v tabulce (3.2). Hodnoty všech výše zmíněných parametrů jsou zadány přímo v programu.

| Název parametru | Zkratka | Hodnota | Jednotka |
|---------------------------------------|------------------|------------|-------------------------|
| hmotnost | m | 1.3 | kg |
| vzdálenost motoru od těžiště | ℓ | 0.175 | m |
| setrvačnost ve směru osy x | I_x | 0.081 | $kg \cdot m^2$ |
| setrvačnost ve směru osy y | I_y | 0.081 | $kg \cdot m^2$ |
| setrvačnost ve směru osy z | I_z | 0.142 | $kg \cdot m^2$ |
| maximální rychlost rotoru | ω_{\max} | 1047.197 | rad/s |
| konstanta rychlosti motoru | K_V | 920 | $ot./min/V$ |
| napěťová konstanta motoru | K_E | 0.0104 | $V \cdot s/rad$ |
| třecí moment motoru | T_f | 0.04 | $N \cdot m$ |
| koefficient viskózního tlumení motoru | D_f | 0.0002 | $N \cdot m \cdot s/rad$ |
| odpor fázového vinutí | R | 0.2 | Ω |
| počet listů vrtule | n_B | 2 | - |
| hmotnost listu vrtule | m_B | 0.0055 | kg |
| poloměr vrtule | r | 0.12 | m |
| poloměr osy vrtule | ϵ | 0.004 | m |
| koefficient tahu vrtule | C_T | 0.0048 | - |
| točivý moment vrtule | C_Q | 0.00023515 | - |
| hustota vzduchu | ρ | 1.225 | kg/m^3 |
| hmotnost tenkého válcového pláště | m_{rot} | 0.025 | kg |
| poloměr tenkého válcového pláště | r_{rot} | 0.014 | m |

Tabulka 3.1: Hodnoty parametrů pro kvadrokoptéru *DJI Phantom 2*

Aby byl program schopen odhadnout energetickou náročnost nějaké trajektorie, musí jako vstupní argument dostat nějakou informaci o této trajektorii. Ze vzorce pro spotřebu energie (2) je patrné, že pro odhad spotřeby konkrétní trajektorie není potřeba kompletní informace o dané trajektorii, ale stačí pouze znát hodnoty rychlostí a zrychlení jednotlivých rotorů v daných časech. A protože zrychlení je derivací rychlosti a celá trajektorie je předem známá (tudíž jsou k dispozici všechny hodnoty rychlostí rotorů), tak není zapotřebí znát předem ani jednotlivá zrychlení, protože ta se dají získat derivacemi rychlostí podle času. Pro odhad spotřeby tedy není zapotřebí znát jednotlivé polohy a orientace dronu, není zapotřebí vědět kde a s jakými rychlostmi a zrychleními v jednotlivých osách se dron pohyboval v jednotlivých okamžicích letu, stačí znát jen informaci o tom, jak rychle se otáčely jednotlivé rotory v jednotlivých okamžicích letu.

Trajektorie letu kvadrokoptéry je spojitá, ale počítače nedokážou se spojitými veličinami pracovat. Každá počítačem navržená nebo počítačem (či nějakými elektronickými senzory) zaznamenaná trajektorie je tedy diskrétní. Program odhadující spotřebu energie dané trajektorie tedy potřebuje jako vstupní argument hodnoty úhlových rychlostí rotorů v diskrétních okamžicích letu. Aby mohl program korektně odhadnout spotřebu energie, potřebuje rovněž znát vzorkovací frekvenci (nebo délku časového kroku - vzorkovací

| Název parametru | Zkratka | Hodnota | Jednotka |
|-----------------------------------|---------------|------------------------|----------------|
| konstanta točivého momentu motoru | K_T | 0.0104 | $N \cdot m/A$ |
| plocha otáček vrtule | A | 0.0452 | m^2 |
| faktor tahu rotoru | κ_b | $3.8305 \cdot 10^{-6}$ | $kg \cdot m$ |
| aerodynamický odpor vrtulí | κ_τ | $2.2518 \cdot 10^{-8}$ | $kg \cdot m^2$ |
| moment setrvačnosti motoru | J_m | $4.9 \cdot 10^{-6}$ | $kg \cdot m^2$ |
| zátěžový moment setrvačnosti | J_L | $3.7004 \cdot 10^{-5}$ | $kg \cdot m^2$ |
| celková setrvačnost motoru | J | $4.1904 \cdot 10^{-5}$ | $kg \cdot m^2$ |

Tabulka 3.2: Hodnoty parametrů figurujících v konstantách c_1, \dots, c_9 (2.14-2.22)

periodu). Čím vyšší je vzorkovací frekvence (čím kratší je délka časového kroku), tím jemnější je diskretizovaná trajektorie. Čím jemnější je diskretní trajektorie, tím více se blíží trajektorii spojitě a odhad její spotřeby je tudíž realističtější než odhad spotřeby hrubší trajektorie (s nižší vzorkovací frekvencí - s delším časovým krokem). Program tedy na vstupu vyžaduje hodnoty úhlových rychlostí všech čtyřech rotorů ve všech diskretních okamžicích letu a délku časového kroku mezi každými dvěma vzorky.

Pro vyzkoušení funkce programu jsem potřeboval nějaký generátor trajektorií. Využil jsem výstupu programu, který pro zadané body průletu navrhne pomocí iterativní gradientní metody časově optimální trajektorii pro účely závodů autonomních kvadrokoptér[10]. Výstupem tohoto programu nejsou přímo jednotlivé rychlosti rotorů $\omega(t)$, ale tahy rotorů $T(t)$. Tah rotoru se ovšem dá snadno přepočítat na rychlost dle následujícího vzorce.

$$T_j(t) = \kappa_b \omega_j^2(t) \quad \Rightarrow \quad \omega_j(t) = \sqrt{\frac{T_j(t)}{\kappa_b}}, \quad (3.1)$$

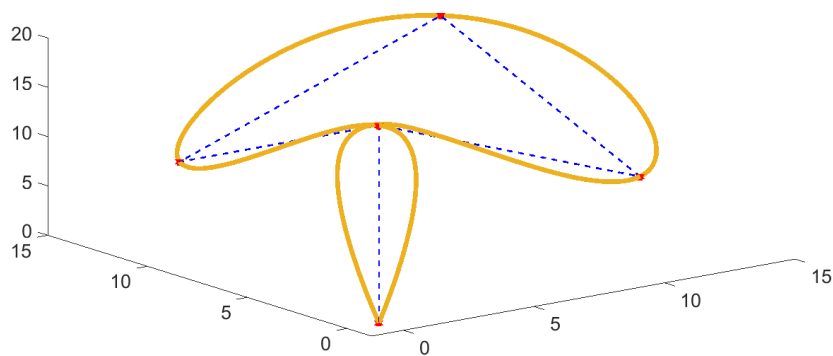
kde j je index rotoru, t je konkrétní časový okamžik a κ_b je faktor tahu rotoru.

Na následujícím obrázku (3.1) můžete vidět konkrétní vygenerovanou trajektorii a na dalších obrázcích (3.2-3.5) grafy popisující tuto trajektorii z pohledu spotřeby energie. Trasa letu prochází body $B_1, \dots, B_7 \in \mathbb{R}^3$, kde každý bod určuje x -ovou, y -ovou a z -ovou souřadnici v metrech. Tyto body jsou:

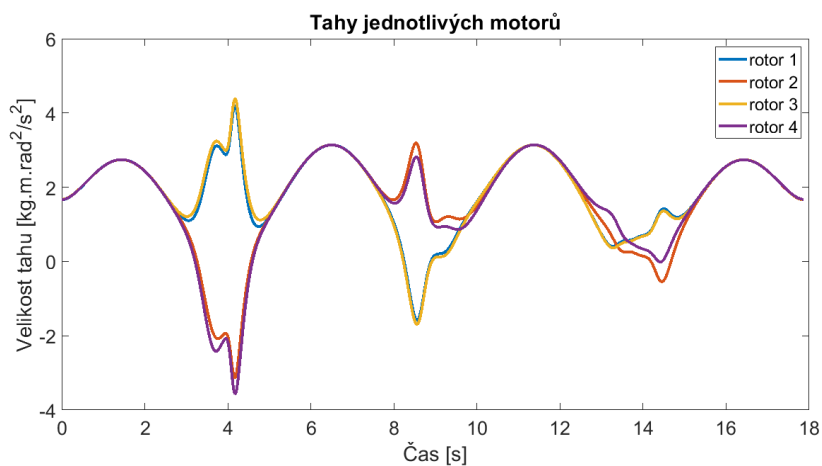
$$B_1 = [0, 0, 0]^T, B_2 = [0, 0, 20]^T, B_3 = [10, 0, 10]^T, B_4 = [10, 10, 20]^T,$$

$$B_5 = [0, 10, 10]^T, B_6 = [0, 0, 20]^T, B_7 = [0, 0, 0]^T.$$

Délka časového kroku mezi každými dvěma vzorky trajektorie je rovna 0.01 sekundy, tedy 10 milisekund. Vzorkovací frekvence je tedy 100 Hz.



Obrázek 3.1: Trasa letu kvadroptéry

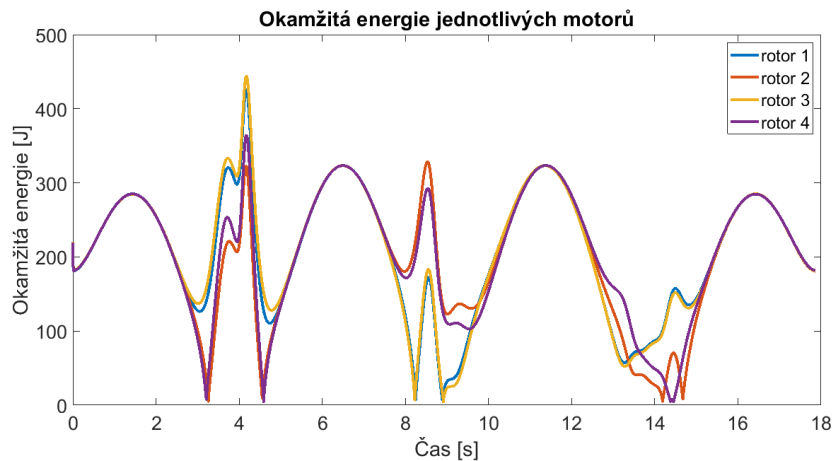


Obrázek 3.2: Tahy jednotlivých rotorů

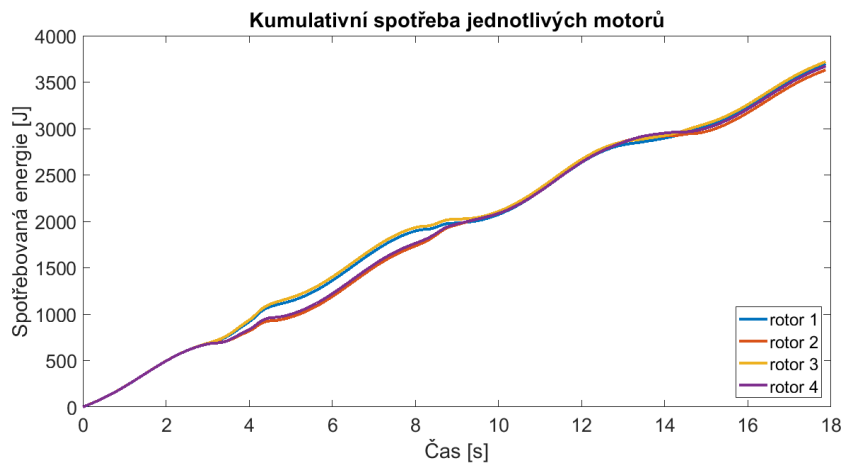
Na obrázku (3.2) jsou vyobrazeny tahy jednotlivých rotorů. Protože se jedná o trajektorii vygenerovanou pro účely závodu dronů, tak model umožňuje i záporné tahy rotorů, tedy že se vrtule točí opačným směrem. Z tohoto grafu je patrné, že když má dron stoupat vzhůru, tak zabírají všechny rotory téměř stejně, když má ovšem dron klesat a zároveň zatáčet, dva rotory vždy zabírají více než zbylé dva.

Na obrázku (3.3) je graf ukazující okamžitou spotřebu jednotlivých rotorů. Při porovnání s grafem na obrázku (3.2) je patrné, že spotřeba energie je tím vyšší, čím více rotor zabírá. Spotřeba energie je vždy nezáporná.

Na obrázku (3.4) je vidět kumulativní spotřeba energie jednotlivých rotorů, tedy kolik energie již daný rotor spotřeboval od začátku letu do konkrétního časového okamžiku. Všechny rotory mají téměř stejnou spotřebu energie, jen



Obrázek 3.3: Okamžitá spotřeba energie jednotlivých rotorů

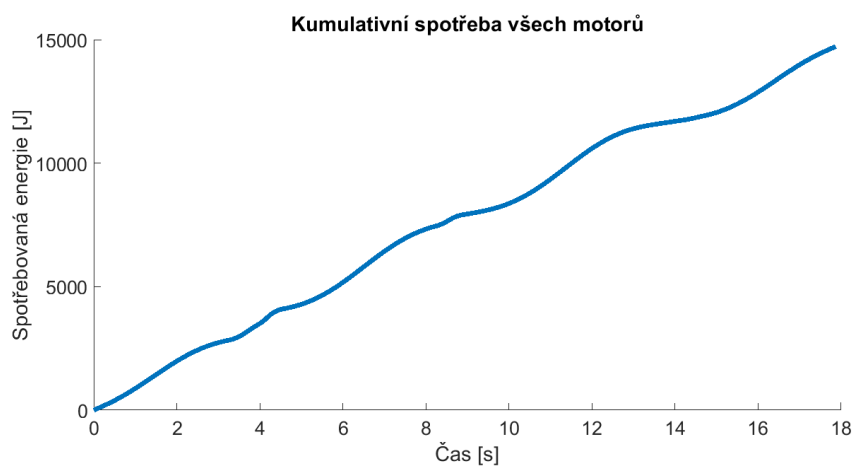


Obrázek 3.4: Kumulativní spotřeba energie jednotlivých rotorů

mezi třetí a devátou sekundou letu je patrné, že celková spotřeba prvního a třetího rotoru je o něco vyšší než u zbylého druhého a čtvrtého rotoru, protože tyto rotory prováděly mezi třetí a pátou sekundou letu otáčecí manévr. Nicméně spotřeba energie všech rotorů se po deváté sekundě letu srovná, neboť obdobný otáčecí manévr provádějí druhý a čtvrtý rotor mezi sedmou a devátou sekundou letu.

Na obrázku (3.5) je již vidět celková kumulativní spotřeba energie všech rotorů dohromady. Z grafu je patrné, že průběh celkové spotřeby je téměř lineární, s tím, že při stoupacích manévrech tempo nárůstu spotřeby zrychluje, naopak při těch okamžicích letu, kdy stroj klesá, tempo nárůstu spotřeby zpomaluje. To je dáno tím, že při stoupání vzhůru musí stroj překonávat gravitační sílu, která jej táhne dolů k zemi, takže rotory se musejí více otáčet, aby tuto sílu překonaly. Naopak při klesání stačí aby rotory zpomalily, neboť při klesání stroji gravitační síla pomáhá.

3. Odhad spotřeby energie pro danou trajektorii



Obrázek 3.5: Celková spotřeba energie během letu

Celková odhadnutá spotřeba energie pro tuto trajektorii je 14 722 joulů.

Kapitola 4

Optimalizace trajektorie letu UAV

4.1 Formulace optimalizační úlohy

Hlavním cílem této práce je navrhnout způsob, jak nalézt pro zadanou trasu (posloupnost souřadnic, jimiž má UAV proletět) takovou trajektorii, která bude optimální vůči spotřebě elektrické energie na baterii. Jinými slovy úkol je pro zadané body trasy najít takový způsob jak je proletět, aby při průletu UAV touto trasou bylo spotřebováno co nejméně energie, tedy aby spotřeba na baterii byla minimální. Úkolem tedy je najít takové hodnoty řídicích proměnných, pro které bude hodnota účelové funkce (odhad spotřeby) za daných omezujících podmínek minimální. Jedná se tedy o úlohu optimálního řízení (zkratkou OCP, z anglického *Optimal Control Problem*). Pro účely formulace této úlohy je potřeba určit stavové a řídicí veličiny.

Stavové veličiny vyjádřím vektorem $\mathbf{x} = [x_1, x_2, \dots, x_{16}]^T \in \mathbb{R}^{16}$ a řídicí vstupy $\alpha = [\alpha_1, \alpha_2, \alpha_3, \alpha_4]^T \in \mathbb{R}^4$ a definuji je takto[5]:

$$\begin{aligned} x_1 &= x, & x_2 &= \dot{x}_1 = \dot{x}, & x_3 &= y, & x_4 &= \dot{x}_3 = \dot{y}, \\ x_5 &= z, & x_6 &= \dot{x}_5 = \dot{z}, & x_7 &= \phi, & x_8 &= \dot{x}_7 = \dot{\phi}, \\ x_9 &= \theta, & x_{10} &= \dot{x}_9 = \dot{\theta}, & x_{11} &= \psi, & x_{12} &= \dot{x}_{11} = \dot{\psi}, \\ x_{13} &= \omega_1, & \dot{x}_{13} &= \alpha_1, & x_{14} &= \omega_2, & \dot{x}_{14} &= \alpha_2, \\ x_{15} &= \omega_3, & \dot{x}_{15} &= \alpha_3, & x_{16} &= \omega_4, & \dot{x}_{16} &= \alpha_4. \end{aligned} \quad (3)$$

S takto definovanými proměnnými se získá následující systém diferenciálních rovnic prvního řádu[5]:

$$\left\{ \begin{array}{l} \dot{x}_1 = x_2, \\ \dot{x}_2 = \frac{\kappa_b}{m} (\sin x_7 \sin x_{11} + \cos x_7 \cos x_{11} \sin x_9) \sum_{k=13}^{16} x_k^2, \\ \dot{x}_3 = x_4, \\ \dot{x}_4 = \frac{\kappa_b}{m} (\cos x_7 \sin x_9 \sin x_{11} - \cos x_{11} \sin x_7) \sum_{k=13}^{16} x_k^2, \\ \dot{x}_5 = x_6, \\ \dot{x}_6 = \frac{\kappa_b}{m} (\cos x_9 \cos x_7) \sum_{k=13}^{16} x_k^2 - g, \\ \dot{x}_7 = x_8, \\ \dot{x}_8 = \left(\frac{I_y - I_z}{I_x} \right) x_{10} x_{12} + \frac{\ell \kappa_b}{I_x} (x_{14}^2 - x_{16}^2) - \frac{J}{I_x} x_{10} (x_{13} - x_{14} + x_{15} - x_{16}), \\ \dot{x}_9 = x_{10}, \\ \dot{x}_{10} = \left(\frac{I_z - I_x}{I_y} \right) x_8 x_{12} + \frac{\ell \kappa_b}{I_y} (x_{15}^2 - x_{13}^2) + \frac{J}{I_y} x_8 (x_{13} - x_{14} + x_{15} - x_{16}), \\ \dot{x}_{11} = x_{12}, \\ \dot{x}_{12} = \left(\frac{I_x - I_y}{I_z} \right) x_8 x_{10} + \frac{\kappa_\tau}{I_z} (x_{13}^2 - x_{14}^2 + x_{15}^2 - x_{16}^2), \\ \dot{x}_{13} = \alpha_1, \\ \dot{x}_{14} = \alpha_2, \\ \dot{x}_{15} = \alpha_3, \\ \dot{x}_{16} = \alpha_4. \end{array} \right. \quad (4)$$

Nyní je k dispozici jak účelová funkce (2) kterou je třeba minimalizovat podle řídicích vstupů $\alpha_1, \dots, \alpha_4$, tak i soubor omezujících podmínek (4). Je tedy k dispozici vše potřebné k formulaci optimalizační úlohy[5]:

$$\begin{aligned} \min_{\substack{\alpha_1, \alpha_2 \\ \alpha_3, \alpha_4}} E_r &= \int_{t_0}^{t_f} \left[\sum_{k=13}^{16} (c_1 + c_2 x_k(t) + c_3 x_k^2(t) + c_4 x_k^3(t) \right. \\ &\quad \left. + c_5 x_k^4(t) + c_6 \alpha_{k-12}(t) + c_7 \alpha_{k-12}^2(t) \right. \\ &\quad \left. + c_8 x_k(t) \alpha_{k-12}(t) + c_9 x_k^2(t) \alpha_{k-12}(t) \right] dt \\ \text{z.p.} &\quad \text{podmínky(4), } \mathbf{x}(t_0) = \mathbf{x}_{t_0}, \mathbf{x}(t_f) = \mathbf{x}_{t_f}, \\ &\quad 0 \leq x_{13} \leq \omega_{\max}, \quad 0 \leq x_{14} \leq \omega_{\max}, \\ &\quad 0 \leq x_{15} \leq \omega_{\max}, \quad 0 \leq x_{16} \leq \omega_{\max}. \end{aligned} \quad (5)$$

4.2 Numerická optimalizace v Matlabu

Takto definovanou úlohu optimálního řízení jsem se pokusil vyřešit pomocí nástroje *FALCON.m*[11] v Matlabu. Tento nástroj byl vyvinut výzkumníky z Institutu dynamiky letových systémů na Technické univerzitě v Mnichově (TUM). *FALCON.m* využívá přímé diskretizační metody v kombinaci s gradientní numerickou optimalizací a automatickou analytickou diferenciací k řešení matematických optimálních řídicích problémů (OCP). Tento program dokáže analyticky spočítat Jakobiány (matice parciálních derivací funkcí podle proměnných figurujících v těchto funkcích) pro účelovou funkci, omezující podmínky i pro model systému. Umožňuje řešit problémy pomocí nelineárního programování (NLP), k čemuž používá softwarovou knihovnu IPOPT (zkratka z anglického *Interior Point OPTimizer*). Program umožňuje zadávat jednotlivé body trasy tak, aby vygenerované trajektorie procházely těmito body ve stanoveném pořadí.

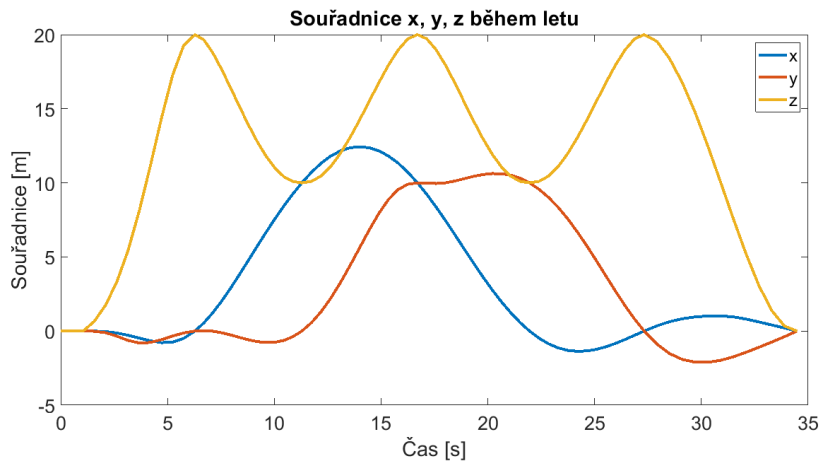
Pro vyřešení konkrétního problému je potřeba vytvořit nový projekt, čímž se vygeneruje několik souborů v Matlabu, které jsou předdefinované pro zadání konkrétních údajů. Do předem vygenerovaných souborů se zadají jednotlivé stavové veličiny, řídicí veličiny, případně nějaké další parametry či výstupní veličiny a stanoví se účelová funkce a model dynamického systému. Pro každou veličinu se musí určit její název a horní a dolní limity hodnot těchto veličin, dále je možné volitelně určit škálování veličin nebo jejich offset. V dalším souboru se určí parametry letu, tedy hodnoty stavových veličin pro jednotlivé etapy letu, tedy například souřadnice bodů trasy. V tomto souboru se určují i parametry pro samotnou optimalizaci, tedy vzorkovací frekvence nebo tolerance odchylek od zadaných hodnot.

Do předem připravených souborů jsem tedy zadal hodnoty veličin, dynamický model a účelovou funkci dle údajů v předchozí sekci této kapitoly. Dále jsem zadal stejné body trasy jako v předchozí kapitole, abych mohl porovnat odhadnutou spotřebu vygenerované energeticky optimální trajektorie se spotřebou neoptimální trajektorie z předchozí kapitoly. Tyto body jsou:

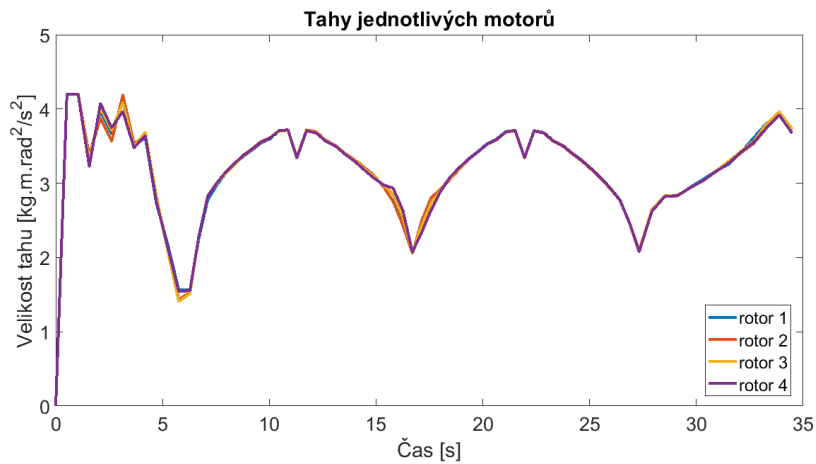
$$B_1 = [0, 0, 0]^T, B_2 = [0, 0, 20]^T, B_3 = [10, 0, 10]^T, B_4 = [10, 10, 20]^T, \\ B_5 = [0, 10, 10]^T, B_6 = [0, 0, 20]^T, B_7 = [0, 0, 0]^T.$$

Vzorkovací frekvenci jsem zvolil 15 Hz.

Vygenerovanou trajektorii můžete vidět na obrázku (4.1).



Obrázek 4.1: Trajektorie vygenerovaná programem *FALCON.m*



Obrázek 4.2: Tahy rotorů pro trajektorii vygenerovanou *FALCON.m*

Jak je vidět, trajektorie přesně prochází stanovenými body trasy B_1, \dots, B_7 . V porovnání s obrázkem trajektorie z předchozí kapitoly je ovšem na první pohled zřejmé, že doba letu je téměř dvakrát delší. Což je podivuhodné, neboť obvykle platí, že čím déle UAV letí, tím vyšší má spotřebu energie. Je však možné, že nějakým způsobem spoří energii tím, že se tolik netočí rotory.

Na obrázku (4.2) jsou vidět tahy jednotlivých rotorů. Je patrné, že předpoklad, že stroj spoří nějakým způsobem energii tím, že by omezoval tahy rotorů, je mylný. Rotory se točí po celou dobu letu poměrně vysokou rychlostí. Spotřeba energie tedy zcela určitě nebude optimální, neboť se točí velmi rychle a to po dobu dvakrát delší, než v případě trajektorie z předchozí kapitoly. Celková programem *FALCON.m* odhadnutá spotřeba energie je 45 477 joulů, což je více než třikrát tolik co v případě trajektorie z předchozí kapitoly. To znamená, že program *FALCON.m* nefunguje tak jak by měl.

Tento problém jsem se snažil vyřešit několika způsoby. Měnil jsem vzorkovací frekvenci, nicméně pro vyšší frekvence program generoval nesmyslné údaje a pro nižší frekvence byla trajektorie příliš hrubá. Obvykle program zhlásil, že optimalizace dosáhla do bodu místní neproveditelnosti (uvázla v nějakém lokálním extrému). Dalším způsobem, kterým jsem se snažil tento problém vyřešit, bylo měnit (tedy zjednodušovat) účelovou funkci nebo dynamický model. Ani tento způsob nevedl ke zlepšení. Zkoušel jsem různé trasy letu a výsledky se velmi lišily. Někdy našel program údajně optimální řešení, ale většinou zhlásil, že řešení nelze nalézt. Zkoušel jsem pomocí tohoto programu hledat řešení optimální vůči době letu a v tomto případě program vždy nějaké údajně optimální řešení nalezl. Problém může být způsoben numerickou nestabilitou nějakých operací potřebných k výpočtu této úlohy. Vzhledem k tomu, že se jedná o tzv. *blackbox* optimalizaci, tak je obtížné zjistit, čím konkrétně je problém způsoben. Mě se to bohužel zjistit nepodařilo.

4.3 Optimalizace v reálném čase

Pro účely optimalizace trajektorie za letu, tedy v reálném čase, není možné psát algoritmus v Matlabu. Matlab totiž vyžaduje vysoký výpočetní výkon, kterým řídicí počítače UAV obvykle nedisponují. Je tedy potřeba najít jiný způsob, který bude proveditelný i na řídicích počítačích s nižším výpočetním výkonem a který bude dostatečně rychlý. Místo Matlabu se hodí použít nějaký nízkoúrovňový programovací jazyk, například jazyk C. Takový způsob optimalizace umožňuje program GRAMPC[12] vyvinutý výzkumníky z Erlangensko-norimberské univerzity v Bavorsku. Zkratka GRAMPC je složenina z anglických slov *GRAdient-based Model Predictive Control*. Přestože má program v názvu zkratku MPC (*Model Predictive Control*), je schopen řešit i úlohy optimálního řízení (OCP, z anglického *Optimal Control Problem*). GRAMPC umožňuje vypořádat se s (vestavěnými) MPC problémy nelineárních a vysoce dynamických systémů s časy vzorkování v (sub)milisekundovém rozsahu. Tento způsob optimalizace je postaven na znalosti modelu nelineárního systému se spojitým časem podléhající stavovým a řídicím omezením. Optimalizační algoritmus, který je základem GRAMPC, se skládá z rozšířeného Lagrangeova schématu ve spojení s metodou přizpůsobeného gradientu. GRAMPC je implementován jako kód v jazyce C s dodatečným uživatelským rozhraním do C++, Matlabu nebo Simulinku. Samotná optimalizační úloha je tedy formulována v jazyce C a uživatelské rozhraní, ve kterém se nastavují parametry letu (například body průletu), je dostupné z prostředí Matlabu, ve kterém je možné si zobrazit i vygenerovanou optimální trajektorii či údaje o výpočtu v reálném čase.

Tento program má ovšem oproti optimalizačním programům v Matlabu jistou nevýhodu. Tou nevýhodou je, že není schopen naplánovat trajektorii pro více bodů trasy, ale jen od jednoho bodu k druhému. To je daň za rychlost výpočtu. V praxi to ovšem žádný problém nepředstavuje, neboť autonomní drony snímají mnohdy zcela neznámé prostředí, takže v celé řadě úloh, pro které se UAV využívají, stejně nelze body průletu deterministicky určit předem. Autonomní létající drony mají celou řadu senzorů (radary, lidary, kamery, termokamery...) které toto neznámé prostředí monitorují a v reálném čase určují podle stanovených kritérií možné směry pohybu. A ve stanoveném směru pohybu určí konkrétní bod, kterého chtějí dosáhnout. Tento jeden bod může sloužit jako konečný bod pro funkci určující optimální trajektorii vůči spotřebě energie. A protože je tato funkce rychlá, tak může být volána opakovaně během letu a původní konečný bod trasy může být po jeho dosažení při dalším volání této funkce považován za počáteční bod, přičemž bude vybrán nový konečný bod a tak pořád znovu po celou dobu letu.

Já jsem se snažil implementovat v tomto programu optimalizační úlohu (5) formulovanou v první části této kapitoly, nicméně program nefungoval dobře. V uživatelském rozhraní v Matlabu jsem určil, že kvadroptéra má letět z bodu $B_1 = [0, 0, 0]^T$ do bodu $B_2 = [1, 2, 3]^T$, časový krok jsem nastavil na 2 milisekundy. Ovšem místo aby kvadroptéra letěla jeden metr ve směru osy x , dva metry ve směru osy y a tři metry ve směru osy z , tedy vzhůru, tak neletěla dle údajů o poloze vůbec ve směrech os x a y , zato se neustále propadala dolů v záporném směru osy z , jako by padala volným pádem. Usoudil jsem tedy, že by to mohlo být způsobeno gravitačním zrychlením působícím v záporném směru osy z , což ovšem nijak nevysvětluje, proč se stroj vůbec nepohyboval ve směrech zbývajících os. Zkoušel jsem různé modifikovat model systému nebo účelovou funkci, nicméně kýženého výsledku, tedy aby stroj dosáhl požadované konečné polohy, jsem nedosáhl. Zkoušel jsem různé hodnoty stavových či řídicích veličin, nicméně program stále generoval nesmyslné hodnoty.

Rozhodl jsem se nakonec, že využiji demonstračního příkladu přímo od vývojářů GRAMPC, který pojmenovali *Quadrotor*. V tomto příkladu použili podobný, nicméně zjednodušený dynamický model kvadroptéry. Jako řídicí proměnné nevyužili úhlová zrychlení rotorů, ale řídicí vstupy U_1, \dots, U_4 definovány vztahy (2.1-2.4), tedy celkový tah rotorů a úhly *roll*, *pitch* a *yaw*. Stavové veličiny $\mathbf{x} = [x_1, x_2, \dots, x_9]^T \in \mathbb{R}^9$ určili takto:

$$\begin{aligned} x_1 &= x, & x_2 &= \dot{x}_1 = \dot{x}, & x_3 &= y, \\ x_4 &= \dot{x}_3 = \dot{y}, & x_5 &= z, & x_6 &= \dot{x}_5 = \dot{z}, \\ x_7 &= \phi, & x_8 &= \theta, & x_9 &= \psi. \end{aligned} \quad (6)$$

S takto definovanými proměnnými určili následující systém diferenciálních rovnic prvního řádu:

$$\begin{cases} \dot{x}_1 &= x_2, \\ \dot{x}_2 &= (\sin x_7 \sin x_9 + \cos x_7 \cos x_9 \sin x_8)U_1, \\ \dot{x}_3 &= x_4, \\ \dot{x}_4 &= (\cos x_7 \sin x_8 \sin x_9 - \cos x_9 \sin x_7)U_1, \\ \dot{x}_5 &= x_6, \\ \dot{x}_6 &= (\cos x_8 \cos x_7)U_1 - g, \\ \dot{x}_7 &= \frac{U_2 \cos x_7 + U_3 \sin x_7}{\cos x_8}, \\ \dot{x}_8 &= U_3 \cos x_7 - U_2 \sin x_7, \\ \dot{x}_9 &= U_2 \frac{\cos x_7 \sin x_8}{\cos x_8} + U_3 \frac{\sin x_7 \sin x_8}{\cos x_8} + U_4. \end{cases} \quad (7)$$

Jako účelovou funkci zvolili minimalizaci odchylek aktuálních hodnot stavových i řídicích proměnných od požadovaných hodnot odpovídajících koncových

proměnných. Tím dosáhli toho, že kvadroptéra se skutečně dostane z počátečního do koncového bodu. Tato účelová funkce by šla rozšířit o prvek minimalizující spotřebu energie, to se mi ovšem nepodařilo do data odevzdání této práce realizovat. Tento úkol tedy bude předmětem mého dalšího studia dané problematiky a práce na tomto projektu.

Kapitola 5

Závěr

V této práci byl představen obecný matematický model kvadrokoptéry s ohledem na její aerodynamické vlastnosti a vlastnosti motorů pohánějících rotory. Rovněž byla uvedena účelová funkce optimalizační úlohy, tedy funkce odhadu spotřeby energie letu UAV.

Dále byl představen program *energyConsumed.m* implementovaný v Matlabu, který pro předem známou trajektorii letu odhadne její energetickou náročnost pro konkrétní typ kvadrokoptéry. Funkce tohoto programu byla demonstrována na příkladu komerčně dostupné kvadrokoptéry *DJI Phantom 2* pro trajektorii vygenerovanou programem určujícím časově optimální trajektorii pro účely závodů autonomních kvadrokoptér. Program *energyConsumed.m* zároveň ilustruje energetickou náročnost jednotlivých letových manévrů, neboť kromě finálního odhadu spotřeby energie na konci letu jsou jeho výstupy i grafy ukazující tahy jednotlivých rotorů v každém okamžiku letu, okamžité spotřeby jednotlivých rotorů a kumulativní spotřeby jednotlivých rotorů i stroje jako celku v každém okamžiku letu.

Poté byla formulována optimalizační úloha. Jednalo se o úlohu optimálního řízení, takže bylo potřeba definovat stavové a řídicí proměnné. Tyto proměnné byly shrnuty v systému diferenciálních rovnic prvního řádu a následně byla představena samotná formulace minimalizace účelové funkce za podmínek stanovených touto soustavou rovnic. Tuto optimalizační úlohu jsem se pokoušel vyřešit pomocí nástroje zvaného *FALCON.m* v Matlabu. Ve snaze najít optimální trajektorii vůči spotřebě energie pro danou trasu letu pomocí tohoto nástroje jsem nebyl zcela úspěšný. Program sice generoval proveditelné trajektorie, nicméně tyto trajektorie zcela určitě nebyly vždy

optimální, neboť trajektorie vygenerované jiným programem, jehož účelem nebylo hledat energeticky optimální trajektorii, mnohdy vygeneroval pro tutéž trasu trajektorii s nižší energetickou spotřebou (což jsem mohl ověřit pomocí dříve uvedeného programu odhadujícího spotřebu energie dané trajektorie). Nakonec byl uveden způsob optimalizace trajektorie v reálném čase, který by mohl být použit přímo při letu kvadrokoptéry jejím palubním počítačem. Tento způsob optimalizace má nevýhodu v tom, že dokáže určit optimální trajektorii jen z jednoho místa na druhé, nikoliv pro nějakou komplexnější trasu. Jeho nespornou výhodou ovšem je vysoká rychlost optimalizačního algoritmu díky implementaci v nízkoúrovňovém programovacím jazyce C, díky čemuž může být optimalizační funkce spotřeby energie volána opakovaně během letu kvadrokoptéry.

V práci na tomto projektu by se dalo zcela určitě pokračovat. Šlo by vylepšit optimalizaci v reálném čase v programu GRAMPC a následně ji implementovat do MRS UAV systému[13] pro použití v praxi. Dále by bylo dobré zkusit naměřit spotřebu energie na baterii UAV při reálném letu a porovnat naměřenou spotřebu s odhadem z programu *energyConsumed.m*, abychom věděli, jak realistický je odhad spotřeby tímto programem. Pokud by se ukázalo, že tento program odhaduje spotřebu realisticky, tak by se dala například vytvořit databáze energeticky úsporných letových manévřů pro specifické úkoly UAV.



Literatura

- [1] “Amazon Prime Air Drone Delivery.” <https://www.amazon.com/Amazon-Prime-Air/b?ie=UTF8&node=8037720011> [online], Naposledy navštíveno 14. 5. 2022.
- [2] “Phantom 2 - DJI.” <https://www.dji.com/cz/phantom-2> [online], Naposledy navštíveno 14. 5. 2022.
- [3] Y. Fouad, N. Rizoug, O. Bouhali, and M. Hamerlain, “Optimization of Energy Consumption for Quadrotor UAV,” in *International Micro Air Vehicle Conference and Flight Competition (IMAV) 2017*, pp. 215–222, 2017.
- [4] S. Bouabdallah, P. Murrieri, and R. Siegwart, “Towards Autonomous Indoor Micro VTOL,” *Autonomous Robots*, vol. 18, p. 171–183, 2005.
- [5] F. Morbidi, R. Cano, and D. Lara, “Minimum-energy path generation for a quadrotor UAV,” in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1492–1498, 2016.
- [6] I. Mademlis, V. Mygdalis, N. Nikolaidis, M. Montagnuolo, F. Negro, A. Messina, and I. Pitas, “High-Level Multiple-UAV Cinematography Tools for Covering Outdoor Events,” *IEEE Transactions on Broadcasting*, vol. 65, no. 3, pp. 627–635, 2019.
- [7] S. M. Shavarani, M. G. Nejad, F. Rismanchian, and G. Izbirak, “Application of hierarchical facility location problem for optimization of a drone delivery system: a case study of Amazon prime air in the city of San Francisco,” *The International Journal of Advanced Manufacturing Technology*, vol. 95, p. 3141–3153, 2018.

- [8] R. Ribeiro, J. Ramos, D. Safadinho, and A. M. de Jesus Pereira, “UAV for Everyone : An Intuitive Control Alternative for Drone Racing Competitions,” in *2018 2nd International Conference on Technology and Innovation in Sports, Health and Wellbeing (TISHW)*, pp. 1–8, 2018.
- [9] T. Chang and H. Yu, “Improving Electric Powered UAVs’ Endurance by Incorporating Battery Dumping Concept,” *Procedia Engineering*, vol. 99, pp. 168–179, 2015. 2014 Asia-Pacific International Symposium on Aerospace Technology, APISAT2014 September 24-26, 2014 Shanghai, China.
- [10] F. Nekovář, “Optimal Trajectory Planning for Quadrotor UAV for Autonomous Drone Race,” Master’s thesis, České vysoké učení technické v Praze, [Dostupné online] <https://dspace.cvut.cz/handle/10467/80355>, 2019.
- [11] M. Rieck, M. Bittner, B. Grüter, J. Diepolder, P. Piprek, C. Göttlicher, F. Schwaiger, B. Hosseini, F. Schweighofer, T. Akman, and F. Holzapfel, “Falcon.m User Guide version 1.27,” *Institute of Flight System Dynamics, Technical University of Munich*, 2022.
- [12] T. Englert, A. Völz, F. Mesmer, S. Rhein, and K. Graichen, “A software framework for embedded nonlinear model predictive control using a gradient-based augmented Lagrangian approach (GRAMPC),” *Optimization and Engineering*, vol. 20, p. 769–809, 2019.
- [13] T. Báča, M. Petrlík, M. Vrba, V. Spurný, R. Pěnička, D. Heřt, and M. Saska, “The MRS UAV System: Pushing the Frontiers of Reproducible Research, Real-world Deployment, and Education with Autonomous Unmanned Aerial Vehicles,” *Journal of Intelligent & Robotic Systems*, vol. 102, no. 26, 2021.

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Kovář** Jméno: **Miroslav** Osobní číslo: **487035**
Fakulta/ústav: **Fakulta elektrotechnická**
Zadávající katedra/ústav: **Katedra kybernetiky**
Studijní program: **Kybernetika a robotika**

II. ÚDAJE K BAKALÁŘSKÉ PRÁCI

Název bakalářské práce:

Odhad a optimalizace spotřeby energie během letu UAV

Název bakalářské práce anglicky:

Estimation and Optimization of Power Consumption during UAV Flight

Pokyny pro vypracování:

Cílem práce je navrhnout způsob minimalizace spotřeby baterie letu bezpilotní kvadrokoptéry (UAV). Úloha se skládá z několika dílčích částí. V první řadě je potřeba spotřebu baterie v závislosti na letové trajektorii výpočtem určit pomocí matematického modelu spotřeby, který může být závislý i na odhadu aerodynamických vlastností UAV. Poté je možné trajektorii podle kritéria spotřeby optimalizovat. Pro potřeby použití optimalizace za letu UAV tzn. provádět výpočet v reálném čase bude také potřeba najít vhodnou aproximaci odhadu spotřeby za letu i bez použití kompletního dynamického či aerodynamického modelu. Výsledky práce mohou posloužit pro prodloužení letového času UAV používajících MRS systém a pro experimentální odhad spotřeby v závislosti na konstrukci UAV a jeho aerodynamických vlastnostech.

1. Student vypracuje matematický model spotřeby baterie za letu v závislosti na trajektorii kvadrokoptéry [2].
2. Student numericky optimalizuje trajektorii k minimalizaci spotřeby baterie [3].
3. Student navrhne aproximační metodu pro odhad spotřeby a optimalizaci trajektorie za letu.
4. Student implementuje svou aproximaci v C++ pro použití v MRS UAV systému [1].

Seznam doporučené literatury:

- [1] Baca, T., Petrlík, M., Vrba, M., Spurný, V., Penicka, R., Hert, D., and Saska, M., "The MRS UAV System: Pushing the Frontiers of Reproducible Research, Real-world Deployment, and Education with Autonomous Unmanned Aerial Vehicles", J Intell Robot Syst 102, 26 (2021).
- [2] Fouad, Yacef & Rizoug, Nassim & Bouhali, Omar & Hamerlain, Mustapha., "Optimization of Energy Consumption for Quadrotor UAV", International Micro Air Vehicle Conference and Flight (2017).
- [3] F. Morbidi, R. Cano and D. Lara, "Minimum-energy path generation for a quadrotor UAV", IEEE International Conference on Robotics and Automation (2016).

Jméno a pracoviště vedoucí(ho) bakalářské práce:

Ing. František Nekovář Multirobotické systémy FEL

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

Datum zadání bakalářské práce: **24.01.2022**

Termín odevzdání bakalářské práce: **20.05.2022**

Platnost zadání bakalářské práce: **30.09.2023**

Ing. František Nekovář
podpis vedoucí(ho) práce

prof. Ing. Tomáš Svoboda, Ph.D.
podpis vedoucí(ho) ústavu/katedry

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

III. PŘEVZETÍ ZADÁNÍ

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

Datum převzetí zadání

Podpis studenta