



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA STROJNÍHO INŽENÝRSTVÍ

FACULTY OF MECHANICAL ENGINEERING

ÚSTAV MECHANIKY TĚLES, MECHATRONIKY A BIOMECHANIKY

INSTITUTE OF SOLID MECHANICS, MECHATRONICS AND BIOMECHANICS

LABORATORNÍ DEMONSTRÁTOR PRO APLIKACI METOD ODHADU STAVŮ DYNAMICKÝCH SYSTÉMŮ

LABORATORY DEMONSTRATOR FOR THE APPLICATION OF DYNAMIC SYSTEMS STATE ESTIMATION
METHODS

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

Filip Maliňák

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Martin Formánek

BRNO 2024

Zadání bakalářské práce

Ústav:	Ústav mechaniky těles, mechatroniky a biomechaniky
Student:	Filip Maliňák
Studijní program:	Mechatronika
Studijní obor:	bez specializace
Vedoucí práce:	Ing. Martin Formánek
Akademický rok:	2023/24

Ředitel ústavu Vám v souladu se zákonem č.111/1998 o vysokých školách a se Studijním a zkušebním řádem VUT v Brně určuje následující téma bakalářské práce:

Laboratorní demonstrátor pro aplikaci metod odhadu stavů dynamických systémů

Stručná charakteristika problematiky úkolu:

V rámci této práce vzniknou studijní podklady k výukovému modelu, konkrétně k laboratorní úloze zaměřené na aplikaci metod pro odhadování stavů systému. Stávající výukový model obsahuje motor a řadu sensorů. Tyto sensory lze vyměnit za přesnější, ale i méně přesné, co může posloužit pro experimentální analýzu nastavení parametrů metod pro odhad stavů systému, jako je např. Kálmánův filtr.

Cíle bakalářské práce:

1. Simulujte chování Kálmánova filtru na obecných LTI systémech prvního a vyššího řádu – studujte vliv nastavení kovarianční matice šumu měření a procesního šumu.
2. Vytvořte matematický model laboratorní sestavy a experimentálně zjistěte její parametry.
3. Proved'te zadanou sadu experimentů, ve kterých použijete různé přesné sensory pro měření veličin motoru a zhodnoťte vliv nastavení matic Kálmánova filtru.
4. Dle provedených experimentů vytvořte výukový materiál pro použití Kálmánova filtru na reálné soustavě. Zaměřte se především na vliv nastavení kovarianční matice šumu měření a procesního šumu.

Seznam doporučené literatury:

VALÁŠEK, M.: Mechatronika, Vydavatelství ČVUT 1995.

NOSKIEVIČ, Petr. Modelování a identifikace systémů. Ostrava: Montanex, 1999. ISBN 80-722-50-0-2.

NELLES, Oliver. Nonlinear system identification: from classical approaches to neural networks and fuzzy models. Berlin: Springer, 2001, xvii, 785 s. : il. ISBN 3-540-67369-5

LABBE, Roger. Kalman and Bayesian Filters in Python, Dostupné z:
<https://github.com/rllabbe/Kalman-and-Bayesian-Filters-in-Python>

Termín odevzdání bakalářské práce je stanoven časovým plánem akademického roku 2023/24

V Brně, dne

L. S.

prof. Ing. Jindřich Petruška, CSc.
ředitel ústavu

doc. Ing. Jiří Hlinka, Ph.D.
děkan fakulty

Abstrakt

Tato práce se zaměřuje na vytvoření studijních materiálů pro porozumění aplikaci Kalmanova filtru. Obsahuje simulaci Kalmanova filtru na lineárních časově invariantních systémech prvního a druhého řádu. Dále se zabývá aplikací filtru na výukový model stejnosměrného motoru s různými senzory, s cílem lépe porozumět nastavování kovariančních matic Q a R .

Summary

This thesis focuses on the creation of study materials for understanding the application of the Kalman filter. It includes the simulation of the Kalman filter on first and second order linear time-invariant systems. Furthermore, it explores the application of the filter on a teaching model of a DC motor with various sensors, with the aim of better understanding the tuning of the covariance matrices Q and R .

Klíčová slova

Kalmanův filtr, DC motor, Simulink Desktop realtime

Keywords

Kalman filter, DC motor, Simulink Desktop realtime

Bibliografická Citace

MALIŇÁK, Filip *Laboratorní demonstrátor pro aplikaci metod odhadu stavů dynamických systémů*. Brno: Vysoké učení technické v Brně, Fakulta strojního inženýrství, 2024. 46 s., Vedoucí diplomové práce: Ing. Martin Formánek.

Prohlášení

Prohlašuji, že jsem bakalářskou práci na téma **Laboratorní demonstrátor pro aplikaci metod odhadu stavů dynamických systémů** vypracoval samostatně s použitím odborné literatury a pramenů, uvedených v seznamu literatury.

Filip Maliňák

Brno

.

Moc děkuji své rodině, přítelkyni a spolubydlícímu za veškerou podporu a pomoc. Navíc bych chtěl poděkovat vedoucímu práce Ing. Martinu Formánkovi a všem mým kolegům z mechatronické laboratoře.

Filip Maliňák

Obsah

1	Úvod	8
2	Rešerše	9
2.1	Převzatá sestava DC motoru	9
2.1.1	Popis DC motoru: Maxon RE35 273757	10
2.1.2	Popis použitých senzorů	10
2.2	Kalmanův filtr (KF) – Stručný přehled	11
2.2.1	Algoritmus KF	11
2.2.2	Kovarianční Matice v Kalmanově Filtru	12
3	Simulace KF na LTI 1. a 2. řádu	14
3.1	LTI 1.řádu - RC článek	14
3.1.1	Simulace RC článku s KF v Simulinku	15
3.1.2	Nastavování kovariančních matic Q a R a výsledky KF	17
3.2	LTI 2.řádu - DC motor	20
3.2.1	Simulace DC motoru s KF v Simulinku	21
3.2.2	Nastavování kovariančních matic Q a R a výsledky KF	22
4	Realizace KF na sestavě	27
4.1	Matematický model sestavy a zjištění parametrů	27
4.2	Experimentální zjištění R a $c\phi$	27
4.3	Zjištění zbylých parametrů	30
4.4	Aplikování KF	31
5	Výsledky a srovnání	36
6	Závěr	42
	Seznam použitých symbolů a zkratek	43
	Bibliografie	44
	Seznam příloh	46

1 Úvod

V prvním ročníku navazujícího inženýrského studia se studenti mechatroniky seznamují s algoritmem Kalmanova filtru, který je v praxi hojně využíván pro odhadování stavů systému v reálném čase na základě nepřesných a neúplných měření. Je velmi důležité, aby studenti byli schopni snadno porozumět problematice Kalmanova filtru.

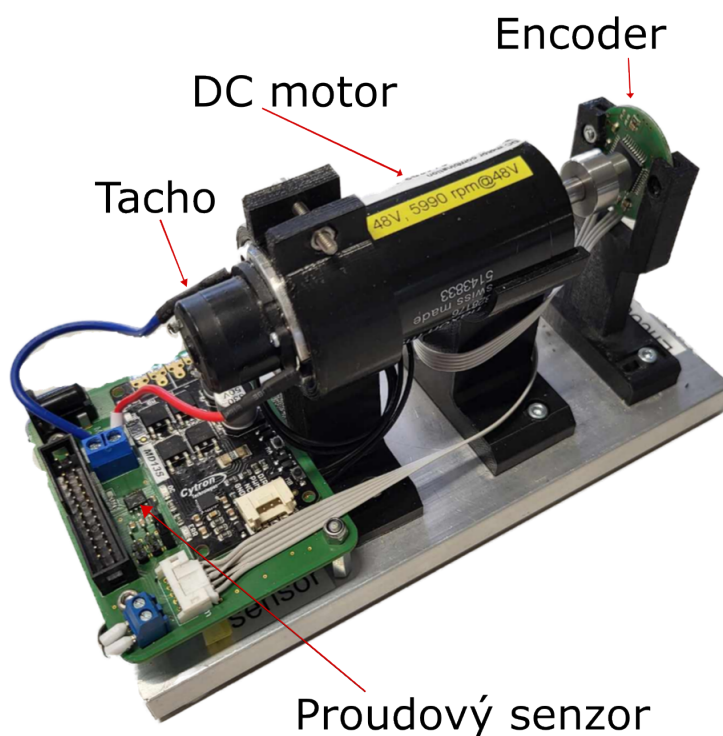
Za tímto účelem jsem dostal k dispozici sestavu, která se používá ve výuce Kalmanova filtru. Mým úkolem je naučit se používat Kalmanův filtr a vytvořit výukové podklady.

Nejdříve musím sám Kalmanovu algoritmu porozumět. Proto nejprve provedu simulaci systému LTI 1. řádu, abych pochopil nastavení kovariančních matic R a Q filtru. Jako druhou fázi seznamování se s filtrem se pak zaměřím na LTI 2. řádu, které odpovídá sestavě DC motoru.

Poté si experimentálně získám parametry pro matematický model převzaté sestavy. Na tuto sestavu pak v reálném čase aplikuji Kalmanův filtr a budu filtrovat otáčky získané derivací polohy z magnetického enkodéru, který použiji ve dvou různých kalibracích. První kalibrace bude s nejmenším možným šumem a druhá s největším možným šumem, čehož dosáhnu úpravou vzdálenosti magnetického enkodéru od magnetu umístěného na hřídeli motoru. Výsledky měření v obou kalibracích pak porovnám.

2 Rešerše

2.1 Převzatá sestava DC motoru



Obrázek 2.1: Fotografie převzaté sestavy s popisem

Sestava, kterou jsem obdržel od vedoucího práce, se skládá z DC motoru Maxon RE 35 273757, Enkodéru RMB30SI13BC1, DC Tacha DCT 22 0.52 V, proudového senzoru ACS712 a budiče DC motoru Cytron MD13S. Veškerá komunikace s PC je zajištěna pomocí multifunkční I/O karty MF 634. Tato karta je v mechatronické laboratoři připojena do PC a sestava je k ní připojena pomocí IDC disketového kabelu.

2.1.1 Popis DC motoru: Maxon RE35 273757

Maxon RE35 273757 je spolehlivý a výkonný DC motor s kartáči z uhlíkových vláken a vysokým počtem komutačních segmentů. Díky tomu má hladký profil otáček a poskytuje stabilitu a kontrolu potřebnou pro úspěšné experimenty v rámci naší sestavy. [6]

- Typ motoru: DC motor s uhlíkovými kartáči.
- Komutační segmenty: Motor je vybaven 13 komutačními segmenty, což umožňuje plynulý a hladký profil otáček.
- Nominální napětí: 48 V
- Nominální otáčky na prázdno: Při nominálním napětí 48 V dosahuje motor otáček na prázdno 5990 ot/min (rpm).

2.1.2 Popis použitých senzorů

- Senzor proudu ACS712: Tento senzor, založený na Hallovu efektu, slouží k monitorování proudu, který prochází motorem během provozu. Poskytuje klíčové informace o aktuálním stavu a zatížení motoru. [2]
- Enkodér RMB30SI13BC1: Jedná se o magnetický enkodér s vysokým rozlišením 8192 CPR, který umožňuje měření s různou mírou zašumění. To je měněno alternováním polohy magnetu uloženého na hřídeli motoru vůči středu čidla. Toto je pro mou práci ideální k porovnání účinnosti Kalmanova filtru na měřeních s rozdílnou velikostí šumu [3]



Obrázek 2.2: Fotografie magnetického enkodéru s uchycením

- DC-Tacho Maxon DCT 22 0.52V: Tachodynamo slouží jako referenční senzor otáček, jejichž stabilní a přesné hodnoty poskytuje, ty jsou pak používány jako cílové pro nastavení Kalmanova filtru [4].

2.2 Kalmanův filtr (KF) – Stručný přehled

Kalmanův filtr je algoritmus, který se používá k odhadu stavu dynamického systému na základě měření zatížených šumem a nepřesnostmi. I když by se mohlo zdát, že přesnější název by byl Kalmanův odhadovač (estimator), termín "filtr" je používán z technických a historických důvodů. V kontextu teorie řízení a zpracování signálů termín "filtr" odkazuje na proces odstraňování šumu z měření, což přesně Kalmanův filtr dělá. Kombinuje měření a model systému, aby odhadl skutečný stav systému s minimální chybou, efektivně "filtrováním" šumu z měření. Tento algoritmus tedy funguje jako filtr tím, že zpracovává signály (měření) a eliminuje jejich šumové složky, aby poskytl čistší a přesnější odhad stavu systému. Přestože je tedy Kalmanův filtr technicky odhadovačem, jeho schopnost filtrovat šum z dat je důvodem, proč se tento termín používá.

Je široce používán v mnoha oblastech, kde je potřeba přesný odhad stavu systému na základě měření zatížených šumem. V robotice slouží k navigaci a lokalizaci robotů, umožňuje jim určit svou polohu a pohyb. V letectví a kosmonautice je klíčový pro sledování trajektorií letadel a kosmických lodí. V ekonomii a finančním inženýrství pomáhá při analýze časových řad a predikci tržních trendů. V inženýrství je využíván pro řízení procesů a systémů, jako jsou autonomní vozidla, které potřebují spolehlivě odhadovat své prostředí a reagovat na něj. KF je také zásadní v oblastech zpracování signálů a telekomunikací pro odstraňování šumu a zlepšování kvality signálů. [10, s. 2]

2.2.1 Algoritmus KF

Část A: Predikce (a priori odhad): V této fázi se používá znalost předchozího stavu systému a dynamiky systému k predikci nového stavu systému v čase k a jeho kovariance $P_{k|k-1}$. To se provádí pomocí stavového modelu a matice šumu procesu Q . Zde jsou jednotlivé kroky:

$$\begin{aligned}\hat{x}_{k|k-1} &= A\hat{x}_{k-1|k-1} + Bu_{k-1} \\ P_{k|k-1} &= AP_{k-1|k-1}A^T + Q\end{aligned}$$

kde:

- $\hat{x}_{k|k-1}$ je predikovaný stav systému v časovém kroku k
- $\hat{x}_{k-1|k-1}$ je predikovaný stav systému v časovém kroku $k - 1$
- A je stavová matice popisující dynamiku systému
- B je matice ovlivňující vstupy
- u_{k-1} je vstup do systému v časovém kroku $k - 1$
- $P_{k|k-1}$ je predikovaná kovariance chyby odhadu v časovém kroku k
- Q je kovariance šumu procesu

Část B: Korekce (a posteriori odhad): V této fázi se na základě nových měření koriguje predikovaný stav, aby se získal optimalizovaný odhad aktuálního stavu systému. To se provádí pomocí matic pozorování C a šumu měření R .

$$\begin{aligned} K_k &= P_{k|k-1} C^T (C P_{k|k-1} C^T + R)^{-1} \\ \hat{x}_k &= \hat{x}_{k|k-1} + K_k (y_k - C \hat{x}_{k|k-1}) \\ P_k &= (I - K_k C) P_{k|k-1} \end{aligned}$$

kde:

K_k je Kalmanovo zesílení (Kalman Gain) pro časový krok k

C je matice pozorování

y_k je pozorování (měření) v čase k

R je kovariance šumu měření

I je jednotková matice

[1]

Intuitivní přístup k vysvětlení fungování Kalmanova filtru je porovnání s tím, jak si lidé odhadují situace ve svém každodenním životě. Představme si, že jdeme ven a sledujeme, jak se počasí mění. Pokud si někdo všimne, že obloha se stává stále tmavší, může odhadnout, že pravděpodobně začne pršet. Tento odhad je založen na informacích o stavu počasí z minulosti (stavu oblohy) a znalostech o tom, jak se počasí mění (dynamice počasí). Pokud se odhad ukáže jako nesprávný, jako například v případě, že se předpověď nevyplní a nezačne pršet, můžeme upravit naši predikci (odhad) na základě skutečných pozorování (měření) - například, že obloha se rozjasňuje.

Podobně Kalmanův filtr kombinuje informace z předchozího stavu systému (část A: Predikce) s novými měřeními (část B: Korekce), aby poskytl odhad stavu systému s minimální chybou. Pokud je odhad založený na předchozích datech blízko skutečnosti, Kalmanův filtr bude dávat větší váhu předchozím odhadům. Naopak, pokud jsou nová měření dostatečně spolehlivá, Kalmanův filtr se přizpůsobí a bude dávat větší váhu novým pozorováním.

Tímto způsobem Kalmanův filtr nejen odhaduje stav systému, ale také se přizpůsobuje a zlepšuje své odhady v průběhu času na základě nových informací. Tato adaptivita a schopnost kombinovat informace z různých zdrojů je to, co činí Kalmanův filtr mocným nástrojem pro odhad stavů dynamických systémů.

2.2.2 Kovarianční Matice v Kalmanově Filtru

V Kalmanově filtru hrají kovarianční matice Q a R klíčovou roli při odhadu stavu systému. Tyto matice se vztahují k nepřesnostem, které vznikají jak v měřeních, tak v samotném modelu systému. Pojďme si detailněji vysvětlit jejich význam a vlastnosti. [9]

Kovarianční Matice Q

Kovarianční matice Q reprezentuje nejistotu spojenou s procesním šumem, tedy nepřesnostmi a chybami, které vznikají v důsledku nedokonalosti odhadovaného modelu systému. Tato nejistota může být způsobena různými faktory, jako jsou nepřesnosti ve fyzi-

kálním modelu, přibližné numerické výpočty nebo neznámé vnější vlivy.

Matice Q má rozměry $n \times n$, kde n je počet stavů systému. Při modelování systému s dvěma stavy (např. pozice a rychlost) bude tedy matice Q rozměru 2×2 .

Kovarianční Matice R

Kovarianční matice R reprezentuje nejistotu spojenou s měřením, tedy nepřesnostmi a chyby, které vznikají při sběru dat z měřících zařízení. Tato nejistota může být způsobena různými faktory, jako jsou šum v senzorech, nepřesnosti měřících přístrojů nebo vnější rušení.

Matice R má rozměry $m \times m$, kde m je počet měření. Při měření dvou veličin (např. pozice a rychlost) bude tedy matice R mít rozměr 2×2 .

Vliv Matic Q a R na Odhad Stavů

Kalmanův filtr kombinuje informace z předchozích stavů systému s novými měřeními a vytváří odhad aktuálního stavu. Kovarianční matice Q a R ovlivňují, jak moc se filtr spoléhá na model systému (predikce) oproti novým měřením (korekce).

- **Velká hodnota v matici Q :** Pokud jsou hodnoty v matici Q velké, znamená to, že máme velkou nejistotu v našem modelu systému. Kalmanův filtr bude dávat větší váhu novým měřením, protože model není považován za příliš přesný.
- **Velká hodnota v matici R :** Pokud jsou hodnoty v matici R velké, znamená to, že měření jsou považována za velmi šumová a nepřesná. Kalmanův filtr bude dávat větší váhu predikci na základě modelu systému a méně váhu novým měřením.
- **Malá hodnota v matici Q :** Pokud jsou hodnoty v matici Q malé, znamená to, že model systému je považován za velmi přesný. Kalmanův filtr bude více spoléhat na predikci na základě modelu.
- **Malá hodnota v matici R :** Pokud jsou hodnoty v matici R malé, znamená to, že měření jsou považována za velmi přesná. Kalmanův filtr bude více spoléhat na nová měření.

Kovarianční matice Q a R jsou základními komponenty Kalmanova filtru, které určují, jak se filtr vyrovnává s nejistotami v modelu systému a v měřeních. Rozměry těchto matic jsou přímo závislé na počtu měřených stavů systému. Kalibrace těchto matic je klíčová pro dosažení optimálního výkonu Kalmanova filtru. Správné nastavení těchto matic umožňuje filtru efektivně kombinovat informace z předchozích stavů a nových měření a poskytovat tak přesné a spolehlivé odhady stavu systému.[9] Postup nastavování těchto matic je podrobně popsán na konci podkapitoly 4.4 Aplikování KF, kde budu nastavovat tyto matice pro zadanou sestavu DC motoru.

3 Simulace KF na LTI 1. a 2. řádu

Pro lepší porozumění Kalmanovu filtru jsem se rozhodl jako první provést simulaci LTI systému prvního řádu. Hlavním účelem této simulace je realizace algoritmu KF v Simulinku, následné porozumění funkci kovariančních matic Q a R a také pochopení nastavování těchto dvou parametrů pro ladění KF. Následně budu provádět simulaci LTI systému druhého řádu, abych lépe pochopil nastavení Q a R pro systémy vyšších řádů. Důvodem je skutečnost, že naše sestava DC motoru je LTI systém druhého řádu.

3.1 LTI 1.řádu - RC článek

Jako první praktickou část své bakalářské práce jsem zahájil seznámením se s Kalmanovým filtrem na co možná nejjednodušším příkladě. Jako LTI systém prvního řádu jsem zvolil RC článek v zapojení low pass filtru, který je popsán diferenciální rovnicí prvního řádu (3.1).

$$\frac{du}{dt} = \frac{u_{in}}{RC} - \frac{u}{RC} \quad (3.1)$$

kde:

u	je výstupní napětí RC článku
$\frac{du}{dt}$	je derivace výstupního napětí RC článku
u_{in}	je vstupní napětí na RC článku
R	je odpor
C	je kapacita kondenzátoru

[6]

Tuto rovnici je možno přepsat do stavového tvaru v této podobě:

$$\begin{aligned} \dot{\mathbf{x}}(t) &= \left[-\frac{1}{RC}\right] \mathbf{x}(t) + \left[\frac{1}{RC}\right] \mathbf{u}(t) \\ \mathbf{y}(t) &= \left[1\right] \mathbf{x}(t) + \left[0\right] \mathbf{u}(t) \end{aligned} \quad (3.2)$$

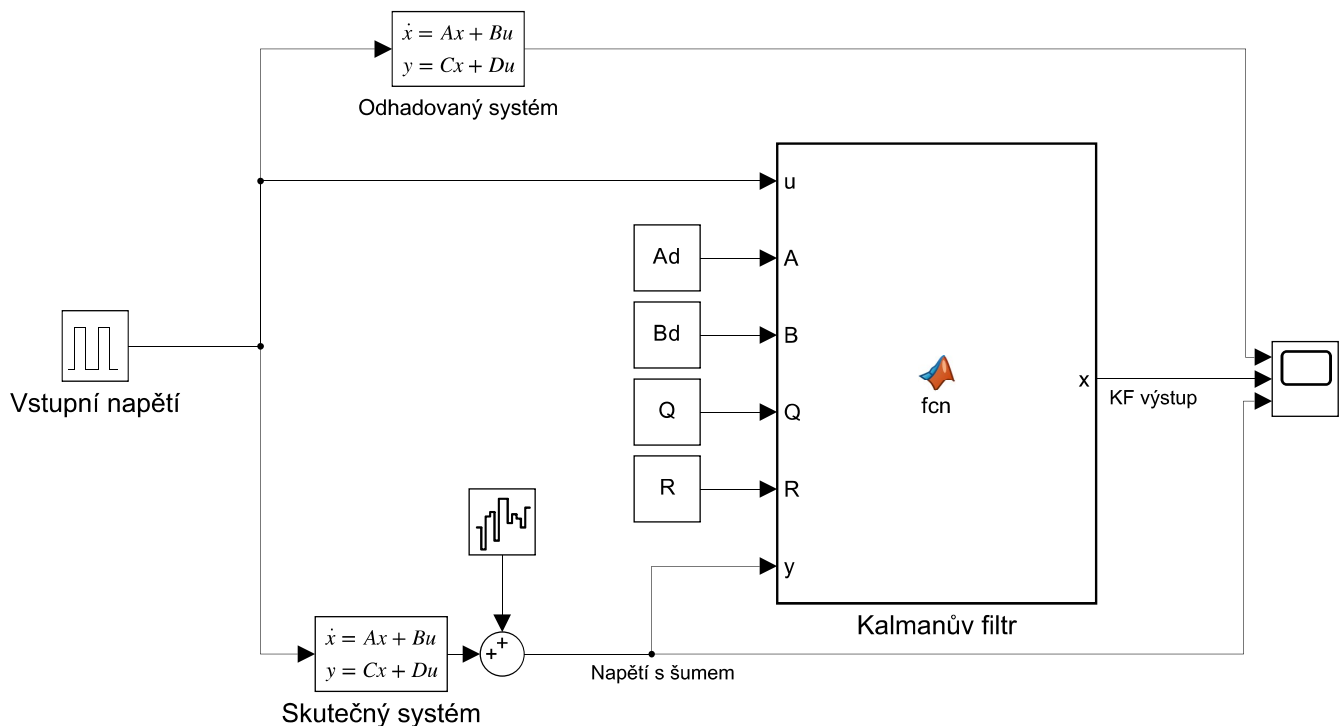
Zápis v matlabu vypadá následovně:

```
A = -1/(R*C); % matice systemove dynamiky
B = 1/(R*C); % matice vstupu
C = 1        % matice vystupu
D = 0        % matice primeho prenosu
```

3.1.1 Simulace RC článku s KF v Simulinku

Pro simulování tohoto systému v simulinku existuje více metod, ale já jsem si zvolil stavový systém, protože bude následně potřebný pro použití Kalmanova filtru. Tento stavový model je potřeba pro použití v Kalmanově filtru převést z kontinuálního času na diskrétní. To se provádí v Matlabu pomocí funkce "c2d(sys, Ts)", kde *sys* je náš stavový systém v kontinuálním čase a *Ts* je časový krok diskrétního systému.

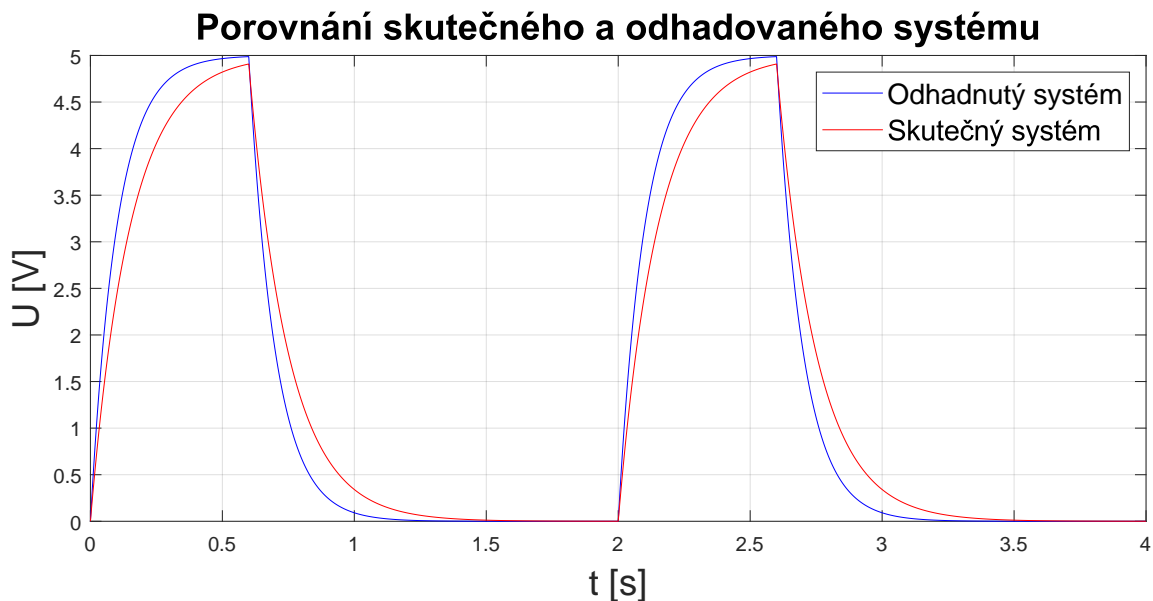
Jak můžete vidět na Obrázku 3.1, tak v modelu máme dva bloky stavového systému (vrchní pro odhadovaný systém a spodní pro skutečný systém), jednu MATLAB funkci (pro aplikování algoritmu KF), dále vstupní signál, který nám skokově přechází mezi napětím 0 V a 5 V každé 2 sekundy, blok s bílým šumem, který je přidán k signálu ze skutečného



Obrázek 3.1: Simulink model RC článku s KF

Dva stavové systémy jsou v modelu z toho důvodu, že při skutečném používání KF se

bude náš model systému lišit od skutečnosti, jeho parametry totiž budeme muset odhadovat z měření. Jejich rozdílné signály lze vidět v grafu 3.2.



Obrázek 3.2: Porovnání odhadovaného systému se skutečným bez přidání šumu

Na grafu můžeme vidět srovnání skutečného modelu bez přidaného šumu z měření (za tento signál budeme v praxi považovat otáčky z dynamu). Jedná se o ideální signál, kterého se snažíme filtrací dosáhnout a odhadovaného systému, který získáváme experimentálním určením parametrů systému.

Samozřejmě se v praxi snažím vytvořit model co nejpřesněji odpovídající skutečnosti, ale musím brát v potaz, že přesnost modelu bude omezena kvalitou senzorů, které použijeme pro získávání dat o systému. To znamená, že i když budeme mít model, který se co nejvíce blíží realitě, nedokážeme zcela eliminovat chyby způsobené nepřesností senzorů.

Kalmanův filtr je následovně proveden jako kód v matlab funkci podle algoritmu z kapitoly 2.2.

```
function X = fcn(u, Ad, Bd, Q, R, ym)

    persistent x_e; % Definice persistentni promenne pro predikovany stav
    if isempty(x_e)
        x_e = 0; % Pocatecni hodnota pro prvni casovy krok
    end
    persistent P; % Definice persistentni promenne pro predikovanou
        % kovarianci chyby odhadu
    if isempty(P)
        P = 1; % Pocatecni hodnota pro prvni casovy krok
    end
    C = 1;

    % Predikce
    x_hat_minus = Ad * x_e + Bd*u;
    P_minus = Ad * P * Ad' + Q;

    % Aktualizace
    K = P_minus*C' / (C * P_minus * C' + R);
    x_e = x_hat_minus + K * (ym - C * x_hat_minus);
    X = x_e;
    P = (eye(size(Ad)) - K * C) * P_minus;
```

3.1.2 Nastavování kovariančních matic Q a R a výsledky KF

V předchozí části jsem popsal samotný model v Simulinku a kód pro Kalmanův filtr. Tudiž se již mohu pustit do nastavování Kalmana pro optimální filtrování signálu. Jak již bylo zmíněno v kapitole 3.1.1, kovarianční matice Q a R jsou maticemi, které určují, jak moc "důvěřujeme" odhadovanému modelu a měřením. V případě RC článku jsou tyto matice 1x1, což znamená, že jsou skalární, protože se jedná o LTI 1. řádu, a tedy máme pouze jeden stav, kterým je výstupní napětí.

Co se týká nastavení těchto dvou veličin, je to hlavní problém, který je potřeba v rámci Kalmanova filtru řešit. Jelikož Kalmanovo zesílení se vypočítá jako kombinace obou veličin, zvyšování jedné n násobně a současně druhé n násobně neovlivní výstup z Kalmana. To má za důsledek, že existuje nekonečně mnoho možností pro nastavení KF. Nejběžněji používaná metoda pro nastavení Q a R v praxi je metoda pokusu a omylu. Zde zvolím obě hodnoty náhodně a následně jednu z hodnot měním podle logaritmické stupnice o

základu deset. Během tohoto procesu pozoruji chování výstupu vůči "Měřené hodnotě" a "Odhadované hodnotě". [9] Vzhledem k tomu, že se jedná o simulaci, kde znám skutečnou charakteristiku systému, můžu také porovnávat výsledky s ní.

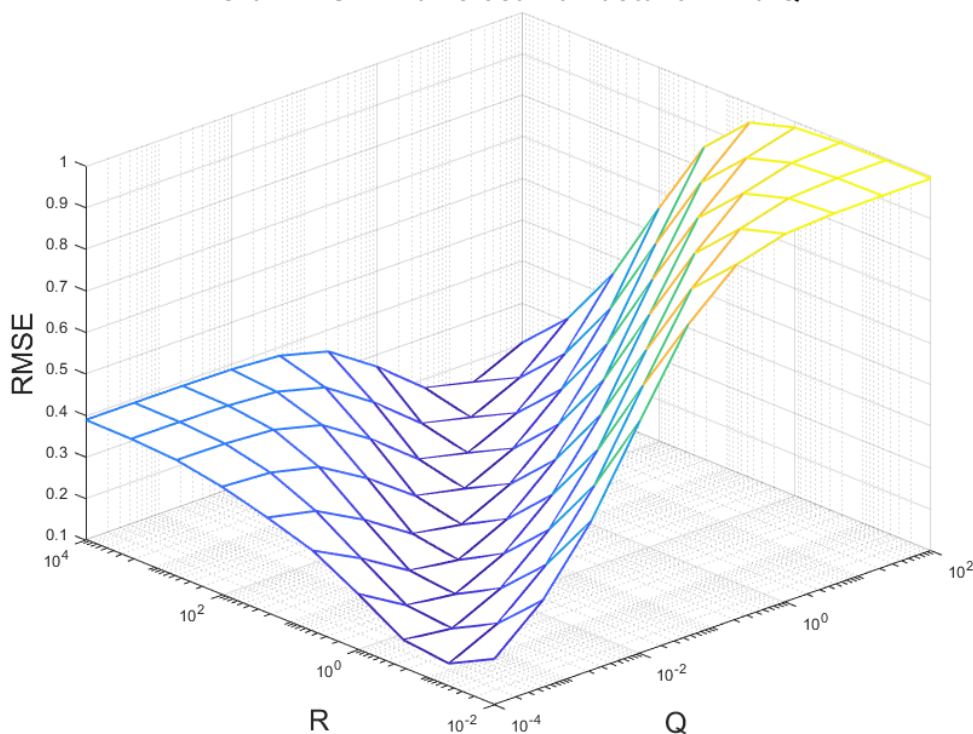
Pro lepší pochopení a představení si nastavování proměnných Q a R jsem se rozhodl provést sadu simulací s různým nastavením těchto veličin. Pro každou simulaci jsem následně vypočítal Root mean square error (RMSE) která se vypočítá následovně:

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^n (\hat{x}_i - x_i)^2}{n}} \quad (3.3)$$

kde \hat{x}_i je skutečná hodnota systému, x_i je hodnota vystupující z KF a n je počet kroků simulace. [7]

Takto jsem schopen kvantifikovat chybu KF vůči skutečnému signálu za dobu celé simulace. Tyto hodnoty jsem následně vykreslil do 3D grafu 3.3, kde na osách X a Y jsou veličiny Q a R s logaritmickým dekadickým rozdělením a na ose Z je hodnota RMSE pro dané nastavení simulace. Z grafu je zřejmé, že pro danou kombinaci Q/R je přesnost téměř stejná. Také můžeme vidět, že pro dostatečně větší R než Q se hodnota RMSE ustálí na konstantní hodnotě a taktéž pro dostatečně větší Q než R se RMSE ustálí na konstantní hodnotě. To vychází z toho, že při $Q \gg R$ Kalman začne brát v potaz pouze model a při $R \gg Q$ pouze měření.

Graf RMSE v závislosti na nastavení R a Q

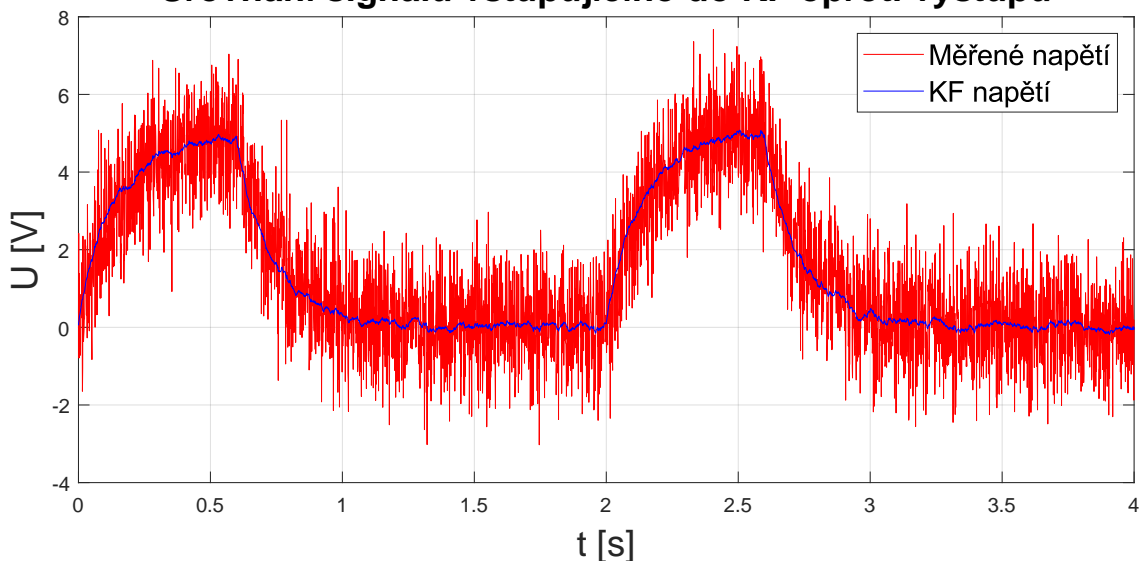


Obrázek 3.3: Graf RMSE simulací s různým Q a R

Po získání tohoto 3D grafu jsem si tedy mohl vybrat jednu dvojici hodnot Q a R

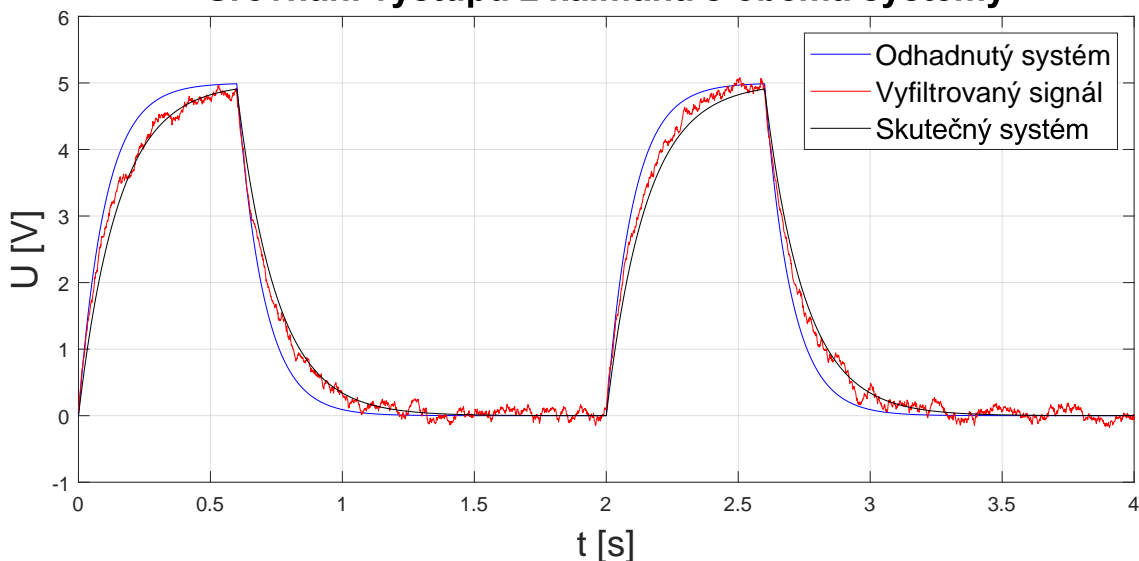
odpovídající nejnižšímu RMSE, což je například $Q = 0.1$ a $R = 100$. S těmito parametry jsem mohl provést simulaci.

Srovnání signálu vstupujícího do KF oproti výstupu



Obrázek 3.4: Srovnání signálu vstupujícího a vystupujícího z KF

Srovnání výstupu z kalmana s oběma systémy

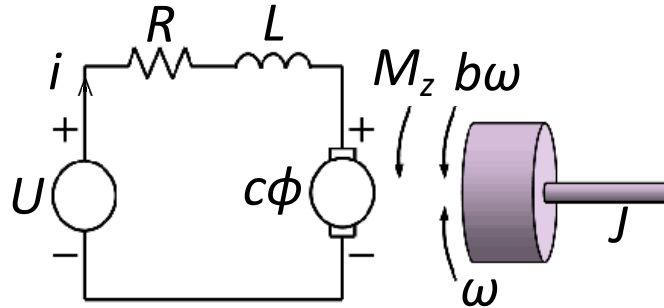


Obrázek 3.5: Srovnání výstupu z KF s oběma systémy

Na grafu 3.4 můžeme vidět zašuměný signál, který vstupuje do KF porovnaný s výstupem z KF, a následně v grafu 3.5 již vyfiltrováný signál, porovnaný vůči odhadovanému systému (v praxi se jedná o systém, jehož parametry experimentálně získám) i skutečnému signálu bez šumu kterému se správným nastavením KF snažíme co nejvíce přiblížit (v praxi tento signál bude výstup z dynama). Z grafů je zřejmé, že při správném nastavení Q a R signál přichází o většinu šumu a zároveň se velice blíží skutečné křivce.

3.2 LTI 2.řádu - DC motor

Na LTI 1. řádu jsem porozuměl KF na základní úrovni a zkusil jsem si, jak lze nastavovat Q a R . Nyní tedy mohu přejít na simulaci složitějšího systému. Pro modelování LTI 2. řádu jsem si zvolil DC motor, jelikož právě na něm budu následně provádět veškeré praktické testy. Model DC motoru můžeme vidět na obrázku 3.6



Obrázek 3.6: Model DC motoru [8]

DC motor je popsán dvojitou rovnicí.

- Rovnice elektrická

$$L \frac{di}{dt} + Ri + c\phi\omega = U \quad (3.4)$$

- Rovnice mechanická

$$J \frac{d\omega}{dt} + b\omega - c\phi i = M_z \quad (3.5)$$

kde L [H] je indukčnost, R [Ω] je odpor motoru, $c\phi$ [$\text{N}\cdot\text{m}/\text{A}$] je konstanta motoru, ω [rad/s] jsou otáčky, U [V] je napětí přiváděné na motor, J [$\text{kg}\cdot\text{m}^2$] je kvadratický moment, b [$\text{Nm}\cdot\text{s}/\text{rad}$] je tlumení motoru, i [A] je proud motoru a M_z [Nm] je zatěžovací moment.

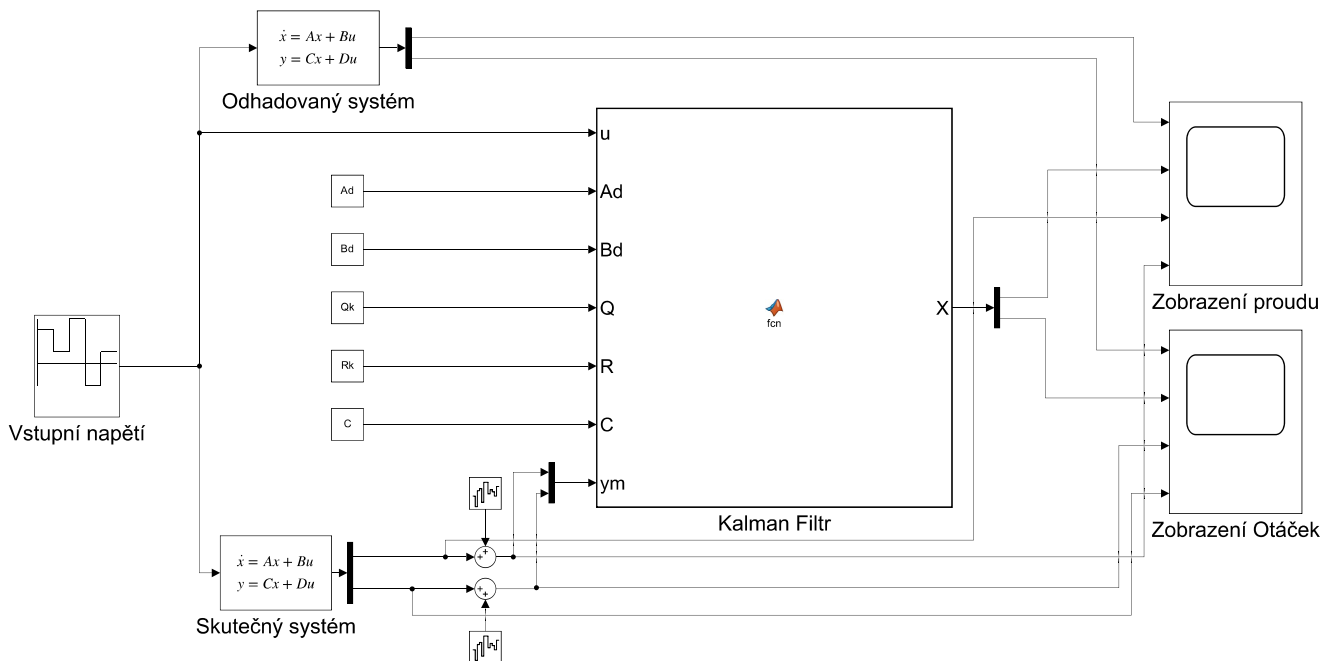
Pro mou simulaci jsem zanedbal zatěžovací moment M_z , což si budu moci dovolit i při měření protože ve skutečnosti nebude na motor působit téměř žádný zatěžovací moment. Tudíž stavový model je v této podobě. [8]

$$\dot{\mathbf{x}}(t) = \begin{bmatrix} -\frac{R}{L} & -\frac{c\phi}{L} \\ \frac{c\phi}{J} & -\frac{b}{J} \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} \frac{1}{L} \\ 0 \end{bmatrix} \mathbf{u}(t)$$

$$\mathbf{y}(t) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} 0 \\ 0 \end{bmatrix} \mathbf{u}(t) \quad (3.6)$$

3.2.1 Simulace DC motoru s KF v Simulinku

Simulace je provedena úplně stejně jako u RC článku; i zde máme dvojici stavových modelů, kde jeden odpovídá skutečnosti a druhý předpokládanému modelu. Na signál ze skutečného modelu přidávám šum pomocí bloku white noise, a to zvláště na signál otáček a zvláště na signál proudu, jelikož ve skutečnosti má každý signál jinou hodnotu zašumění. Tyto dva signály poté spojuji do jednoho dvojnásobného signálu, který přivádím do Kalmanova filtru. Výstup opět rozdělují na proud a otáčky, které následně vykresluji v grafu. Tento model můžeme vidět níže na obrázku 3.7



Obrázek 3.7: Model DC motoru s KF v simulinku

Kalmanův filtr jsem implementoval stejně jako u RC článku, a to prostřednictvím bloku MATLAB funkce. Jediný rozdíl je, že nyní již matice v Kalmanově filtru nejsou jednorozměrné, ale mají rozměr 2×2 , protože DC motor má dva stavy.

Tedy nastavení persistentních proměnných vypadá takto:

```

persistent x_e; % Definice persistentni promenne predikovaneho stavu
if isempty(x_e)
    x_e = zeros(2,1); % Pocatecni hodnota pro prvni casovy krok
end

persistent P; % Definice persistentni promenne pro predikovanou
               % kovarianci chyby
if isempty(P)
    P = [1 1; 1 1]; % Pocatecni hodnota pro prvni casovy krok
end

```

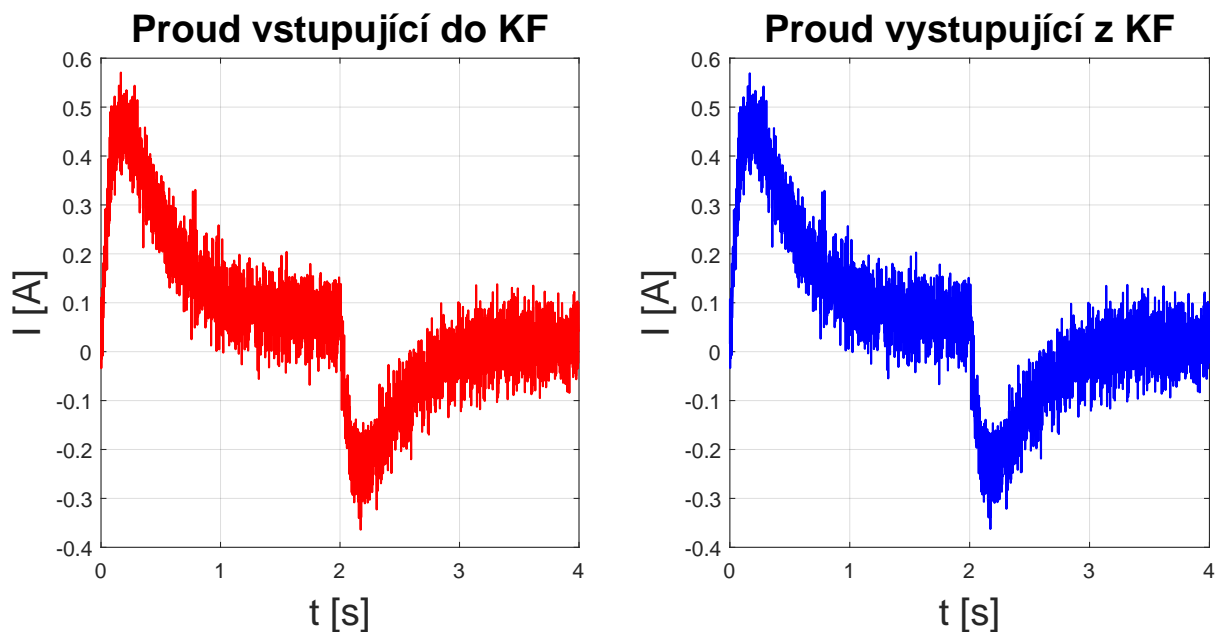
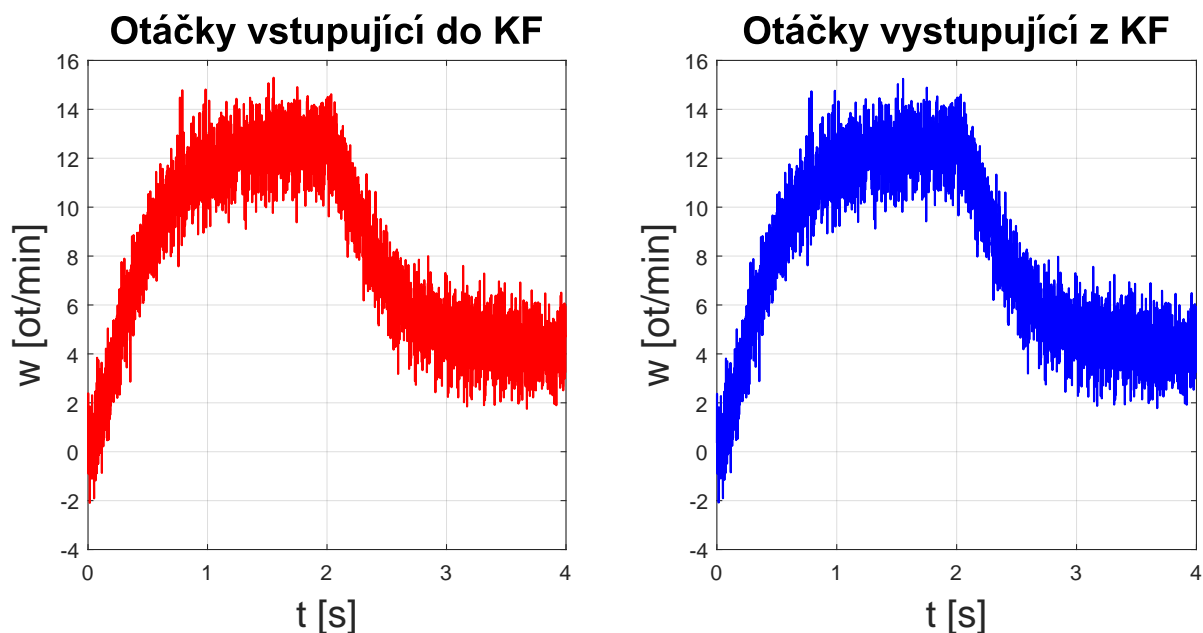
3.2.2 Nastavování kovariančních matic Q a R a výsledky KF

Matice Q a R musí mít rozměr 2×2 , jak již bylo vysvětleno v podkapitole 2.2.2. Prvky na hlavní diagonále matice Q , označené jako Q_{ii} , vyjadřují varianci jednotlivých stavů. Prvky na vedlejší diagonále, označené jako Q_{ij} pro $i \neq j$, představují vzájemné korelace mezi stavy. Obdobně prvky na hlavní diagonále matice R , označené jako R_{ii} , vyjadřují varianci jednotlivých měření a prvky na vedlejší diagonále, označené jako R_{ij} pro $i \neq j$, představují vzájemné korelace mezi měřeními. V našem případě považujeme stavy i měření za nekorelované, a proto prvky na vedlejší diagonále považujeme za nulové, přestože tomu tak není. Stav v našem modelu jsou na sobě závislé, ale jelikož pro nás bude v praktickém měření důležitý pouze jeden ze stavů, mohou si dovolit tuto korelaci zanedbat a Kalmanův filtr nastavit přesně pouze pro jeden stav.

Nastavování prvků Q_{ii} a R_{ii} pokračuje stejně jako u LTI 1. řádu, jenom dvakrát, protože mám dva stavy. Pro každý stav pozoruji výstup zvlášť a měním jeden z odpovídajících prvků Q a R , dokud nedosáhnu dostatečného vyfiltrování. Jako výchozí hodnotu jsem si náhodně zvolil:

$$Q = \begin{bmatrix} 1000 & 0 \\ 0 & 1000 \end{bmatrix} R = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (3.7)$$

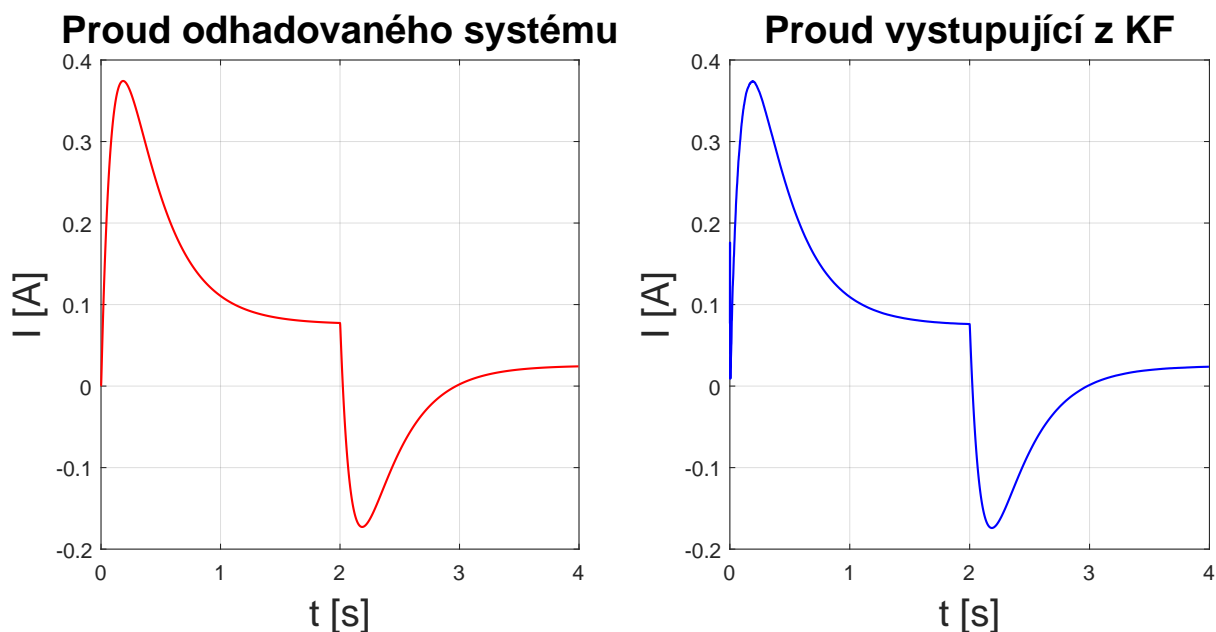
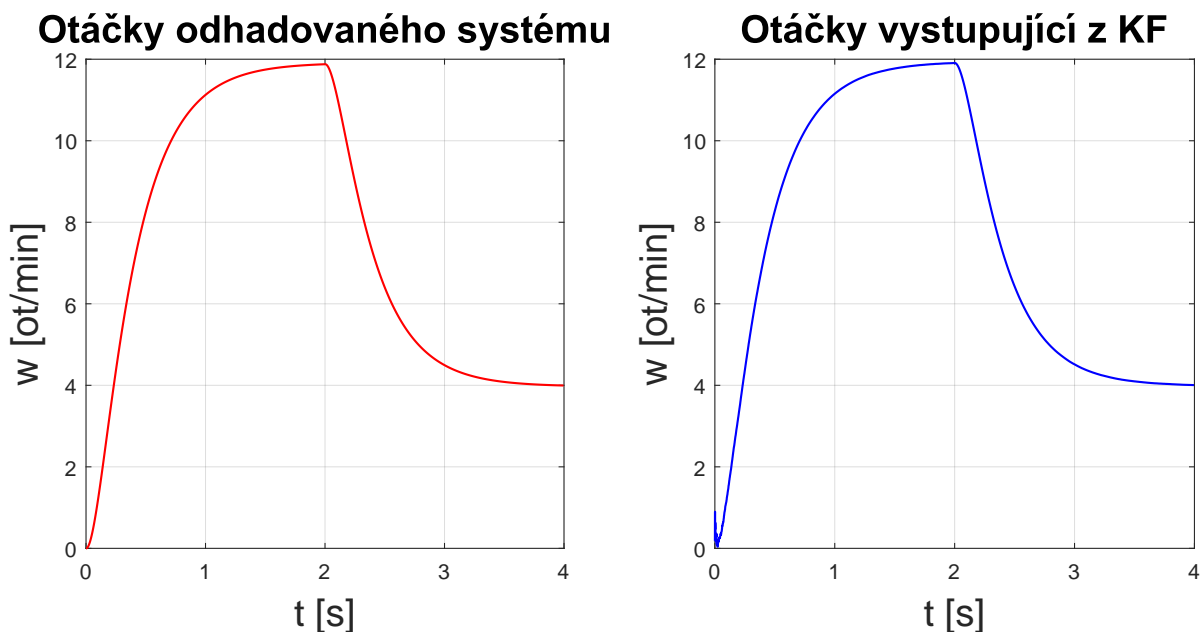
S těmito zvolenými hodnotami jsem spustil simulaci. Porovnával jsem výstupní hodnotu proudu i otáček oproti jejich vstupním hodnotám.

Obrázek 3.8: Srovnání grafů vstupního a výstupního proudu KF při $Q_{ii} = 100$ Obrázek 3.9: Srovnání grafů vstupních a výstupních otáček KF při $Q_{ii} = 100$

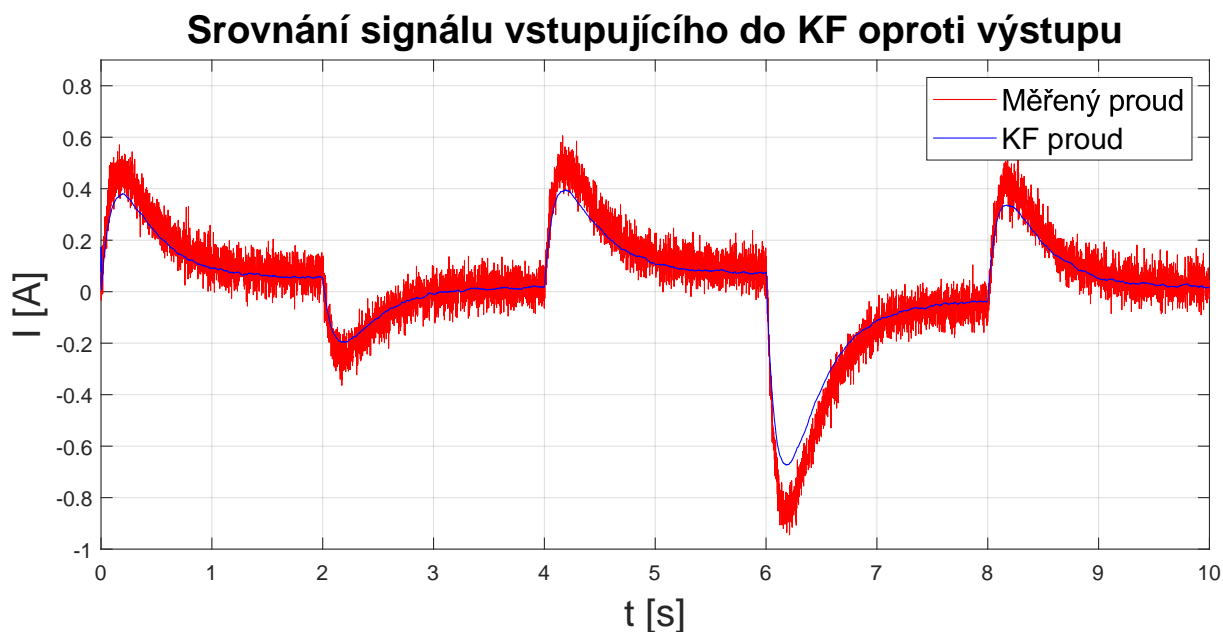
Z těchto grafů je jasně vidět, že při této konfiguraci, kde platí $Q_{ii} \gg R_{ii}$, odpovídá výstupní signál KF vstupnímu signálu, protože KF klade veškerý důraz na měření.

Následně jsem hledal druhý extrém, při kterém Kalmanův filtr klade veškerý důraz na odhadovaný systém. To nastane v případě když platí $Q_{ii} \ll R_{ii}$. Zvolil jsem hodnoty:

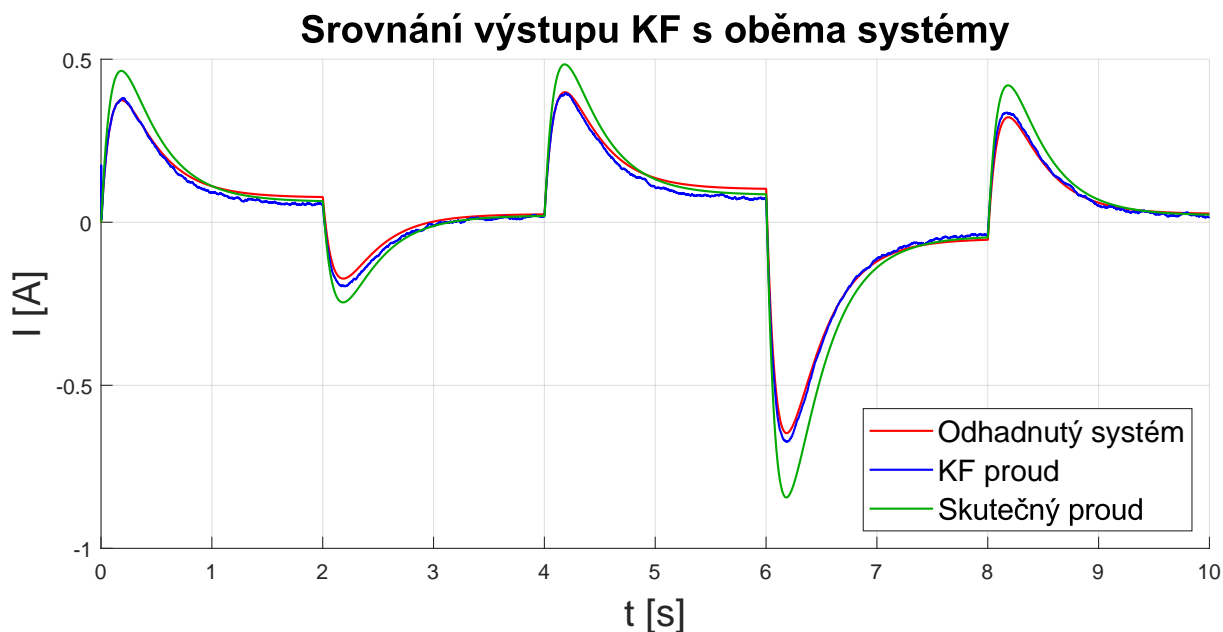
$$Q = \begin{bmatrix} 0.000001 & 0 \\ 0 & 0.000001 \end{bmatrix} R = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (3.8)$$

Obrázek 3.10: Srovnání grafů vstupního a výstupního proudu KF při $Q_{ii} = 0.000001$ Obrázek 3.11: Srovnání grafů vstupních a výstupních otáček KF při $Q_{ii} = 0.000001$

Hodnoty Q_{ii} pro co nejnižší odchylku výstupu Kalmanova filtru od skutečné hodnoty tedy leží v intervalu $(0.000001, 1000)$. Kalibrace KF probíhá následovně: postupně měním hodnotu Q_{11} a sleduji výstup proudu z Kalmanova filtru. Současně měním hodnotu Q_{22} a sleduji výstup otáček z Kalmanova filtru. Jelikož chci dosáhnout co nejlépe vyfiltrovaných otáček, musím se spokojit s nepřesným proudem, protože zanedbávám korelaci mezi i a ω . Z toho důvodu v rámci proudu kladu větší váhu na odhadovaný systém, abych se zbavil šumu z proudu. Výsledné nastavení parametrů je tedy: $Q_{11} = 0.0003$ a $Q_{22} = 0.007$.

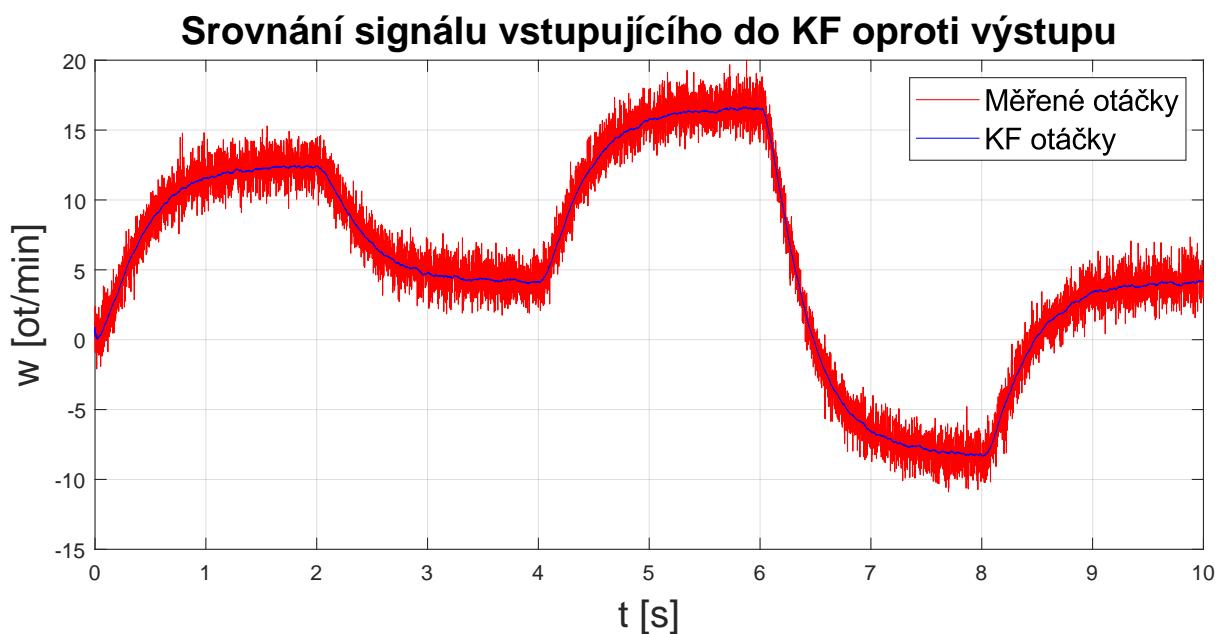


Obrázek 3.12: Graf zašuměného signálu proudu

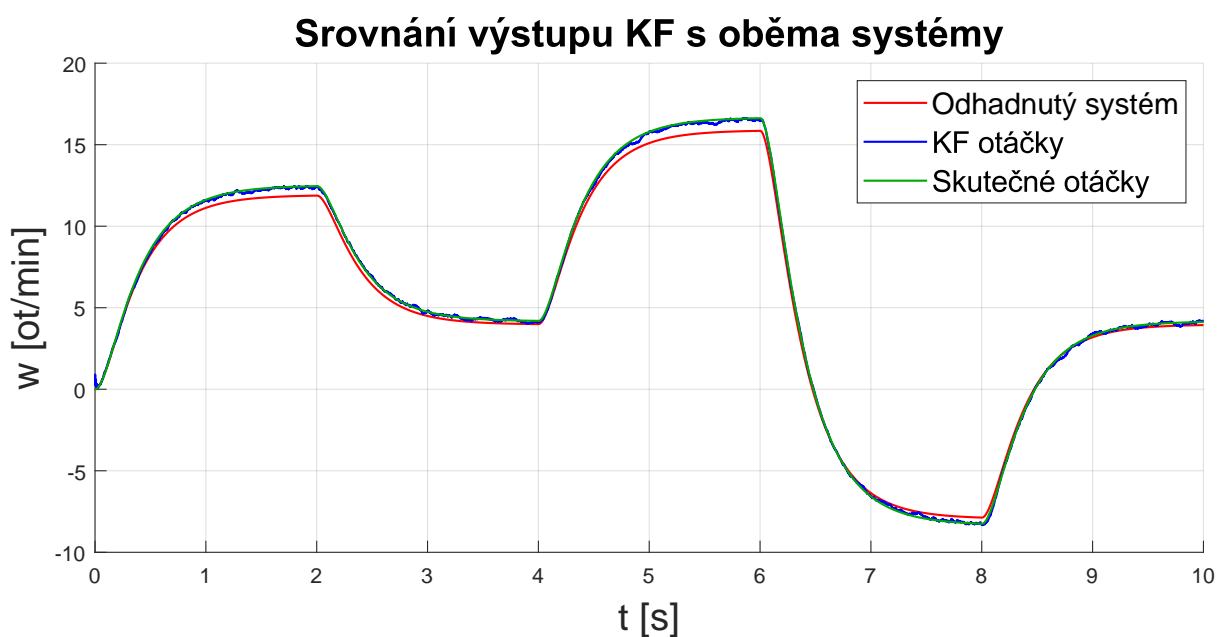


Obrázek 3.13: Srovnání KF s oběma systémy

Na grafu 3.12 můžeme vidět porovnání zašuměného signálu proudu před a po aplikování KF. Následně v grafu 3.13 již vyfiltrovaný signál, porovnaný vůči odhadovanému signálu (opět v praxi systém v KF získaný experimentálně) i skutečnému signálu bez šumu (tento signál v praxi mít nebudu). Z grafu je zřejmé, že při správném nastavení Q a R signál přichází o většinu šumu. Také můžeme vidět, že se signál příliš nepodobá skutečnému systému. To z toho důvodu, že jsem KF nastavoval pro co nejlepší výsledky otáček, jelikož to bude účel KF v praxi, a při tomto nastavení jsem dosáhl nejvyšší přesnosti.



Obrázek 3.14: Graf zašuměného signálu otáček



Obrázek 3.15: Srovnání KF s oběma systémy

Na grafu 3.14 můžeme vidět zašuměný signál otáček, který vstupuje do KF, porovnaný s výstupním signálem již zpracovaným Kalmanovým filtrem. Následně v grafu 3.15 vidíme již vyfiltrovaný signál, porovnaný vůči odhadovanému signálu i skutečnému signálu bez šumu vzniklého měřením (v praxi tento signál budou dynamometr otáčky). Z grafu je zřejmé, že při správném nastavení Q a R signál přichází o většinu šumu a velice přesně opisuje charakteristiku skutečného systému.

4 Realizace KF na sestavě

Po provedení všech simulací, které mi umožnily pochopit chování a práci s Kalmanovým filtrem, jsem se mohl pustit do realizace tohoto filtru na zadané sestavě. Tato realizace zahrnuje několik částí:

1. Použití správného matematického modelu.
2. Zjištění parametrů matematického modelu.
3. Aplikace Kalmanova filtru na sestavu.

4.1 Matematický model sestavy a zjištění parametrů

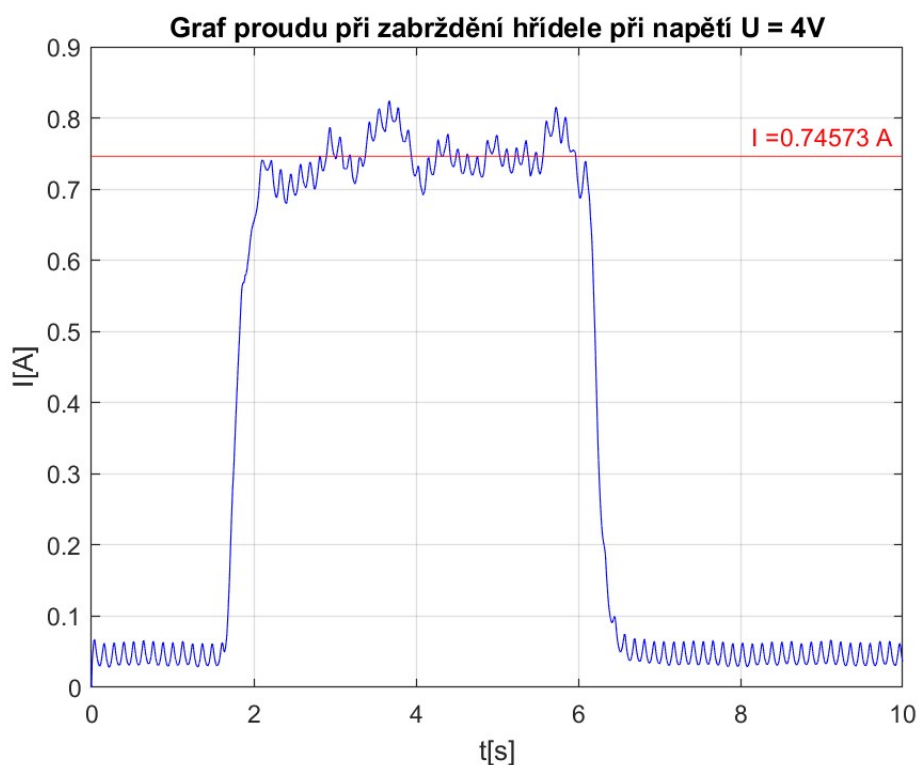
Pro popis zadané sestavy použijeme totožný matematický model jako při simulaci LTI 2. řádu, který můžeme vidět na obrázku 3.6. Diferenciální rovnice pro tento model je totožná s modelem v simulaci. Rce. 3.6

$$\dot{\mathbf{x}}(t) = \begin{bmatrix} -\frac{R}{L} & -\frac{c\phi}{L} \\ \frac{c\phi}{J} & -\frac{b}{J} \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} \frac{1}{L} \\ 0 \end{bmatrix} \mathbf{u}(t)$$

$$\mathbf{y}(t) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} 0 \\ 0 \end{bmatrix} \mathbf{u}(t)$$

4.2 Experimentální zjištění R a $c\phi$

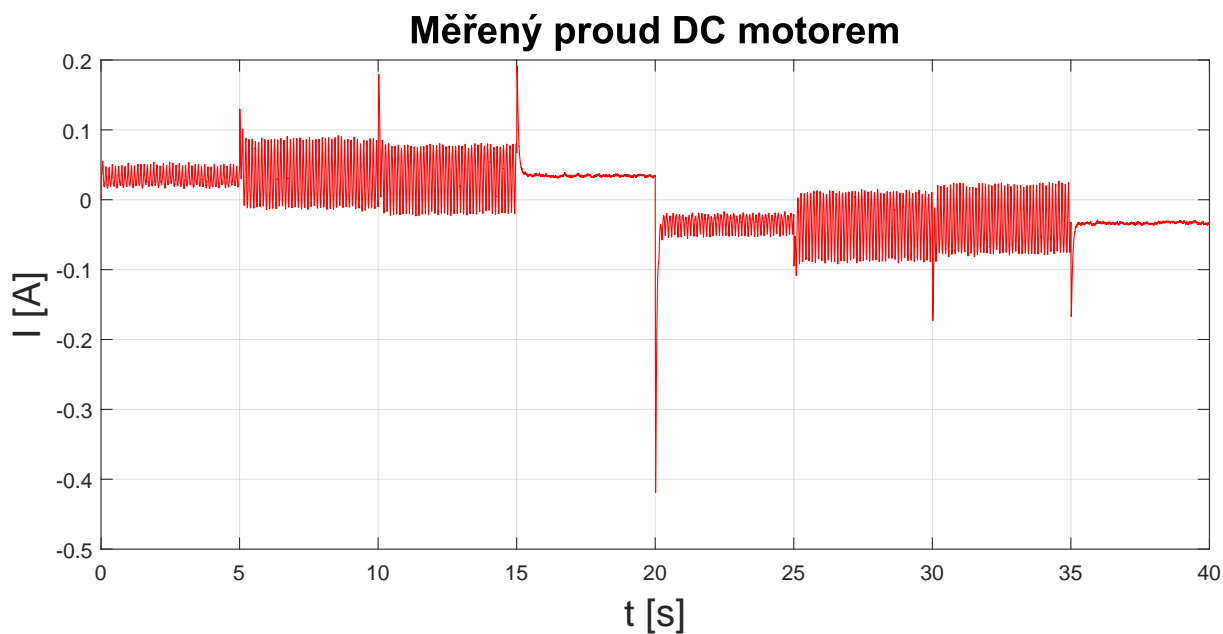
Parametry R a $c\phi$ jsem schopen zjistit experimentálně z rovnice pro elektrickou část DC motoru. Uvažuji ustálený stav, kdy $\frac{di}{dt} = 0$. Když motor zabrzdím, zbyde mi z dané rovnice pouze $Ri = U$, kde U znám, protože ho přivádím na motor pomocí nastavení frekvence PWM a proud i mohu změřit. Abych lépe odčítal proud, nechal jsem ho projít low pass filtrem, který odstraňuje velký šum, a po ustálení jsem schopen odečíst hodnotu proudu. Toto měření jsem provedl s krokem 1V od -6 V do 6 V, vyjma nuly. Bylo by samozřejmě ideální provést měření v celém rozsahu napájení, ale při vyšších otáčkách nejsem schopen motor bezpečně zabrzdít. Pro každé napětí jsem vypočítal odpor a tyto odpory jsem zprůměroval. Takto jsem získal odpor motoru $R = 5.505 \Omega$. V grafu 4.1 lze vidět naměřený signál proudu pro napětí $U = 4V$.



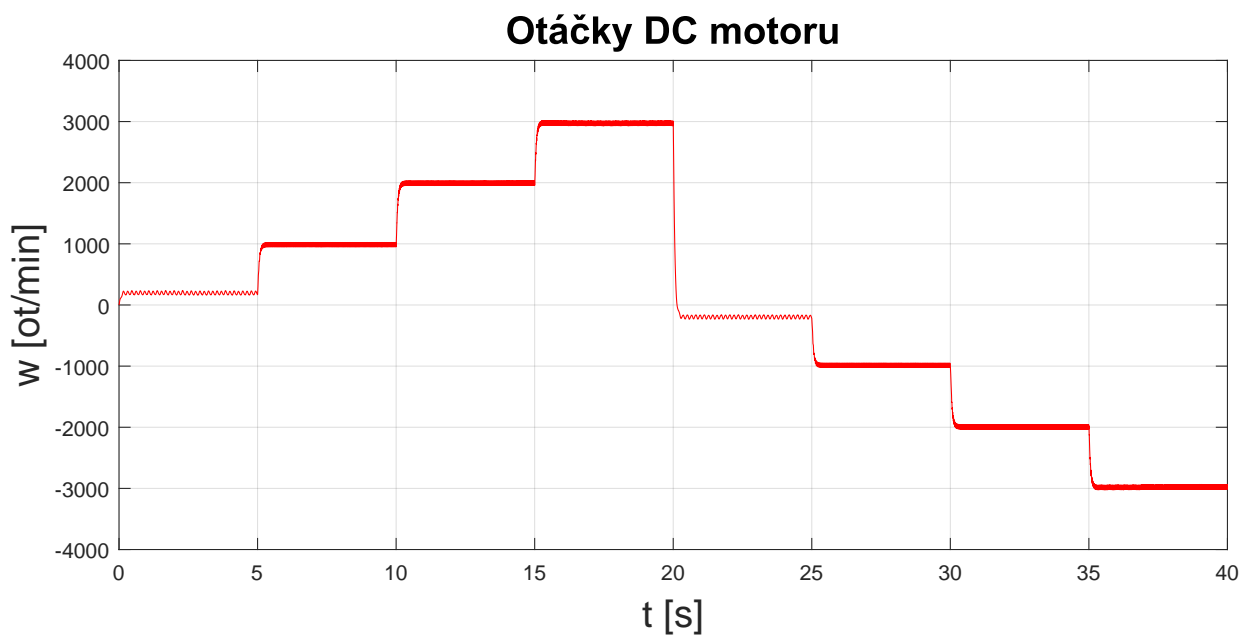
Obrázek 4.1: Měřený proud při zabrzdění motoru s napětím $U = 4V$

Když znám odpor, mohu zjistit konstantu motoru $c\phi$. Toho dosahuji tak, že při daném napětí měřím otáčky i proud, a tudíž po ustálení proudu z rovnice $U = Ri + c\phi \cdot \omega$ zůstane neznámé pouze $c\phi$. Proto jsem provedl měření při napájení skokovým signálem napětí s jednotlivými kroky [2, 8, 16, 24, -2, -8, -16, -24] V, přičemž každý skok trval 5 sekund, aby se systém stihl ustálit. Signál jsem mohl vyfiltrovat pomocí low pass filtru, což usnadnilo vyčítání jednotlivých hodnot. Low pass filtr jsem si mohl dovolit použít, protože jsem měřil hodnoty po ustálení, na které low pass filtr nemá vliv.

Na následující straně lze naléznout grafy 4.2 (proudu) a 4.3 (otáček) použité pro výpočet $c\phi$. Při odhadu parametrů pro měření otáček používám enkodér místo dynama, jelikož v praxi bych používal Kalmanův filtr na sestavu, ve které bych neměl k dispozici přesný senzor. Chci tedy vidět, jak dobře funguje Kalmanův filtr při odhadu systému pomocí senzoru, který následně budu filtrovat. Derivace polohy enkodéru, použitá pro odhad parametrů, je mírně filtrovaná low-pass filtrem pro menší odchylku odhadovaného systému od střední hodnoty měření v ustáleném stavu.



Obrázek 4.2: Graf naměřeného proudu v DC motoru



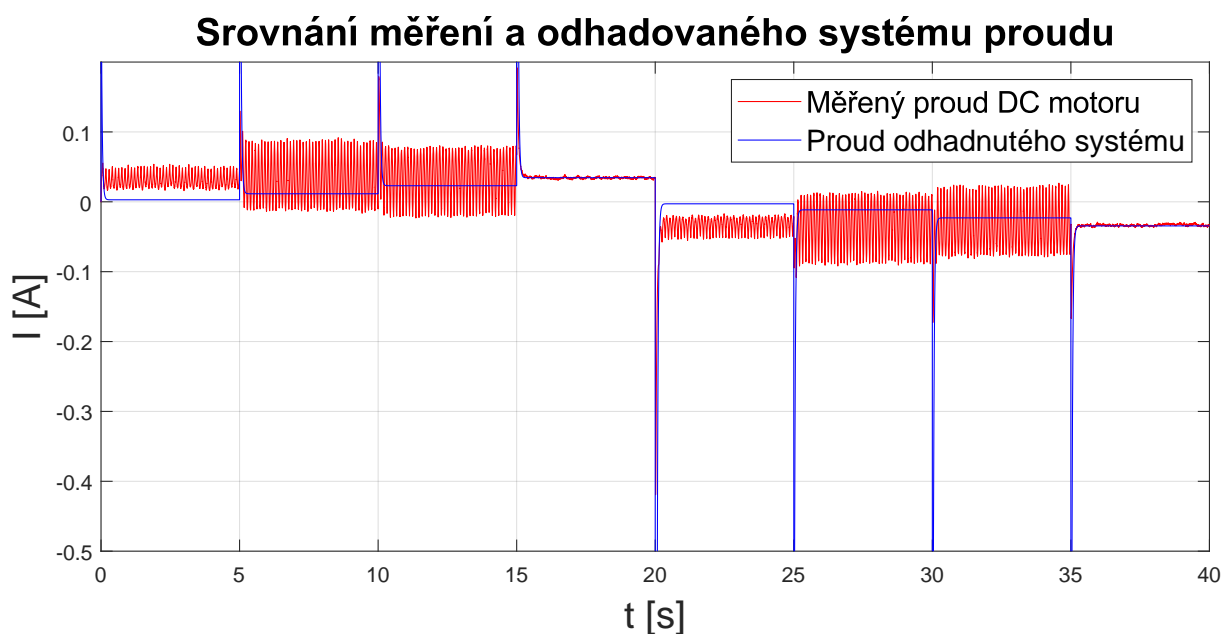
Obrázek 4.3: Graf derivace signálu z magnetického enkodéru

Z těchto dat již pak šlo pomocí zprůměrování každého kroku získat jednu hodnotu proudu i otáček za daný krok, z těchto hodnot jsem dopočítal $c\phi$ pro daný krok a všechny $c\phi$ jsem následovně zprůměroval, čímž jsem dostal hodnotu $c\phi = 0.0083377 \frac{Nm}{A}$

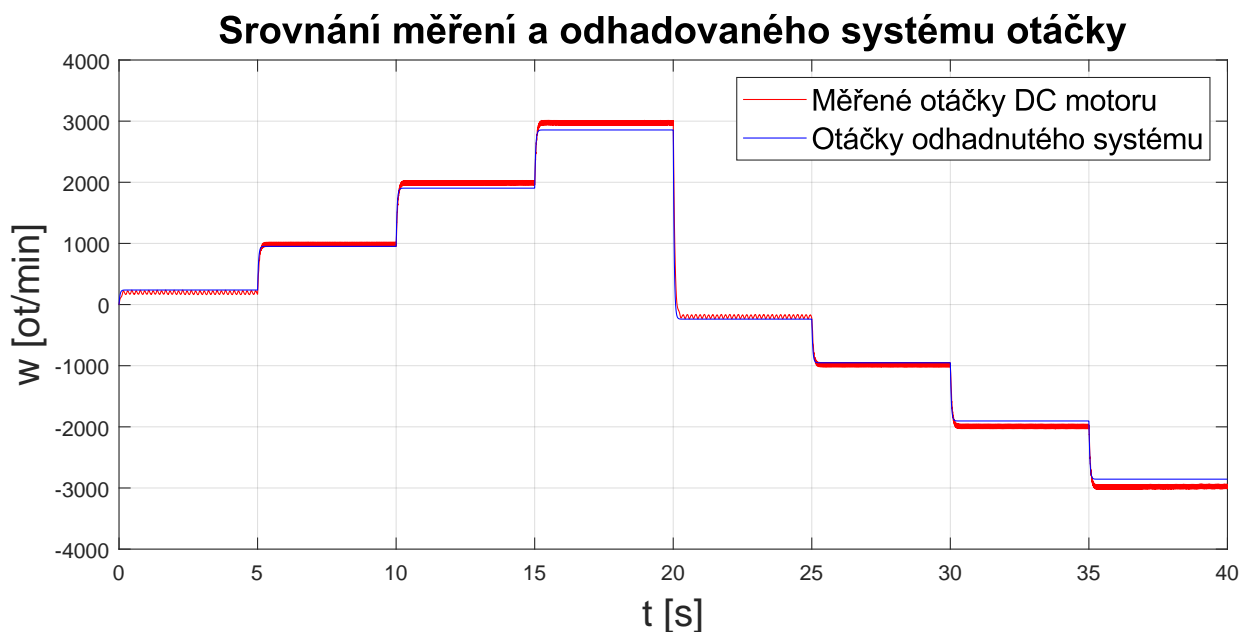
4.3 Zjištění zbylých parametrů

Nyní tedy mám hodnoty $c\phi$ a hodnotu R modelu DC motoru.

Data, která jsem použil pro zjištění hodnoty $c\phi$ DC motoru, nyní použiji pro získání zbylých parametrů L , J a b z rovnic uvedených v kapitole 4.1. Naměřený signál tedy nahraji zpět do Simulinku, kde na něj použiji nástroj Parameter Estimation [11], který mi umožní zbylé parametry odhadnout. Tento algoritmus je již součástí Simulinku a funguje na iterační bázi, kde v každém kroku porovná předpokládaný systém s měřením a pokud si nejsou dostatečně podobné, tak upraví parametry a opět je porovná. Jednoduše přivedu signály z workspace do Simulinku pomocí bloků "From workspace" a vytvořím stavový systém pro DC motor. Následně zapnu aplikaci Parameter Estimation a vyberu, které parametry chci předpovídat a vůči jakému signálu je porovnávat. Zbytek provede algoritmus sám. Po proběhnutí algoritmu dostanu odhadnuté parametry $L = 0.01077$ H, $J = 4.3953 \cdot 10^{-7}$ kg \cdot m² a $b = 1.0071 \cdot 10^{-7}$ Nm \cdot s/rad. Srovnání odhadnutého systému s naměřeným mohu vidět v grafech 4.4 a 4.5



Obrázek 4.4: Graf srovnání měřeného proudu s odhadovaným

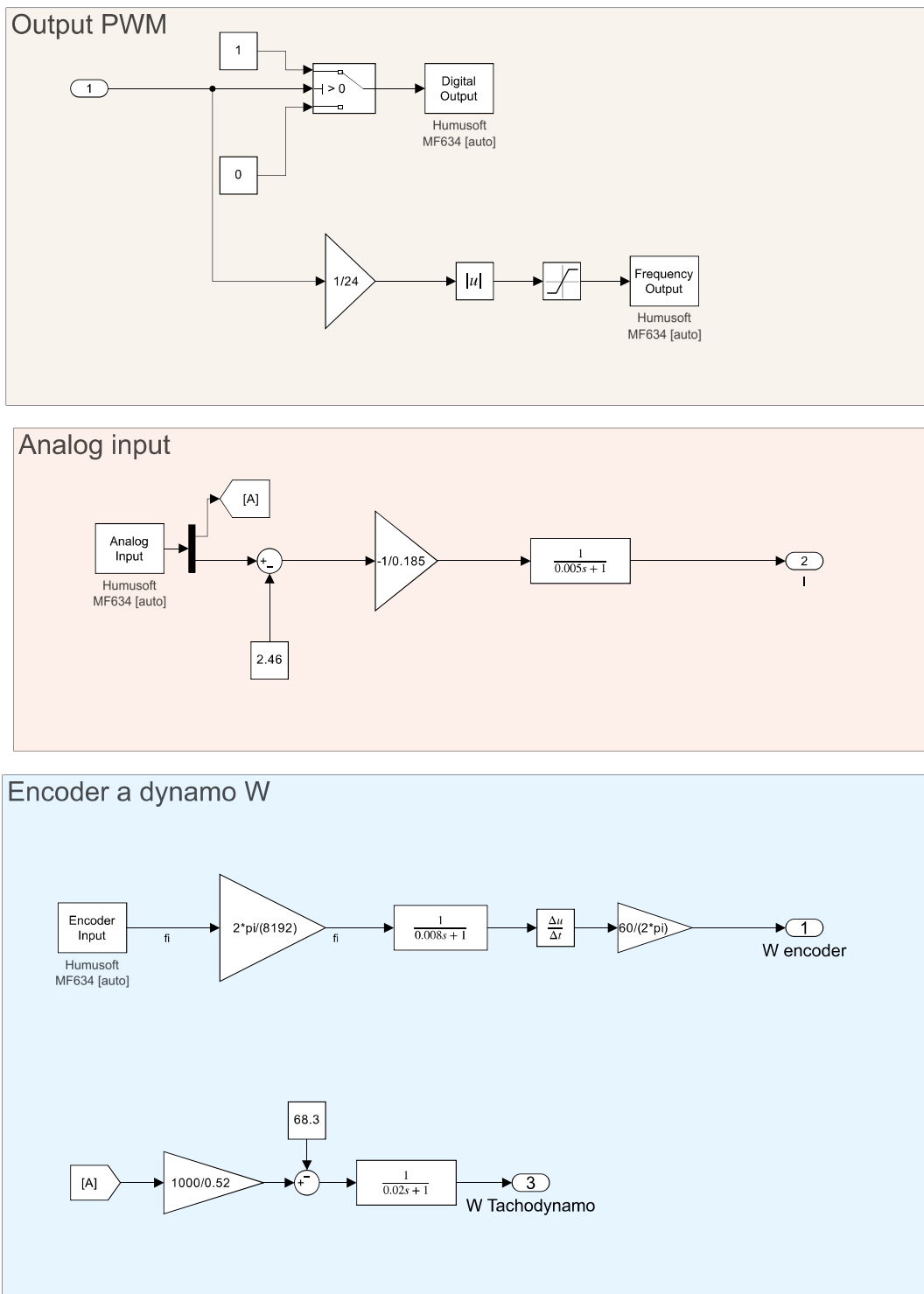


Obrázek 4.5: Graf srovnání měřených otáček s odhadovanými

Takto jsem získal matematický model popisující zadanou sestavu DC motoru. Z grafu je zřejmé, že odhadovaný systém neodpovídá absolutně naměřeným hodnotám. Což bylo očekáváno. Tato nepřesnost je v KF následně zahrnuta v kovarianci Q .

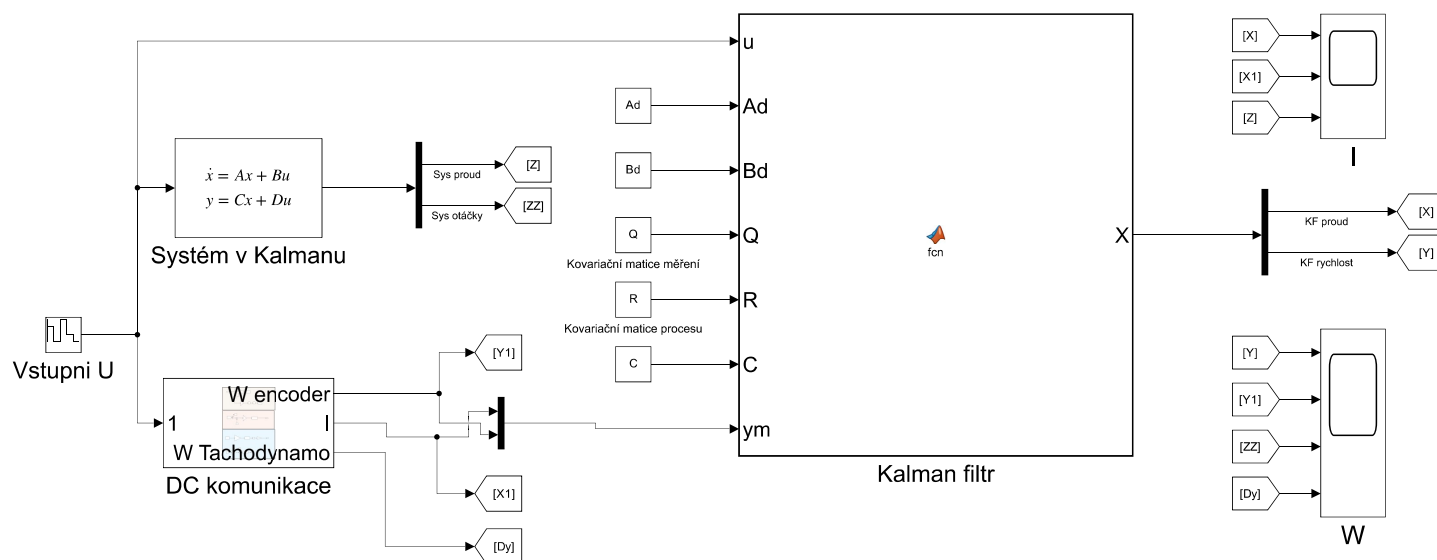
4.4 Aplikování KF

Nyní tedy mohu přejít na konkrétní implementaci Kalmanova filtru na zadanou sestavu. Jak již bylo zmíněno, pro komunikaci s modelem používám realtime funkce Simulinku. Napájení motoru ovládám nastavováním střídavy PWM a směr řídím přepínáním konkrétního pinu, což je vidět v modelu ve vrchní části pod názvem *Output PWM*. Níže čtu analogový vstup, ze kterého získávám signál ze senzoru proudu a signál z tachodynamu. Oba senzory kalibruji pomocí zesílení bločkem *Gain* a pomocí posunutí střední hodnoty, odčítáním konstanty následně oba signály provedu mírným low pass filtrem, abych se zbavil nejsilnějšího šumu. V poslední části zpracovávám signál z enkodéru, který mi po převodu na správné jednotky pomocí bločku *Gain* dává natočení v radiánech. Z natočení dostanu otáčky ω derivací tohoto signálu, jenže tento signál z enkodéru má skokový přírůstek hodnot, tudíž by nešel derivovat. Z toho důvodu signál před derivací prochází low pass filtrem, který mi signál vyhladí a zbaví se skokového charakteru. Všechny signály jsou kalibrovány podle jejich datasheetů [2][3][5]. Celá tato komunikace je následně uzavřena do podsystému *DC komunikace*. Tento podsystém můžeme vidět na obrázku 4.6.



Obrázek 4.6: Podsystem reálnové komunikace v Simulinku

Na vstupní subsystému DC komunikace přivádíme schodovitý signál napětí, který také přivádíme do KF a do stavového systému odpovídajícího odhadovanému systému v KF.



Obrázek 4.7: Simulink model pro aplikaci KF na sestavě

V KF používáme matematický model popsáný stavovým systémem 3.6 s parametry získanými v kapitolách 4.2 a 4.3. Tento model je však potřeba převést do diskretního stavu (stejně jako tomu bylo u simulací), to bylo provedeno v matlabu pomocí funkce "c2d(sys,T)" která náš stavový systém v kontinuálním stavu převede na diskretní systém s časovým krokem T , tento krok musí být totožný s krokem simulace. Veškeré matice vstupující do KF mají rozměry 2×2 , jelikož náš systém má dva stavy (proud a otáčky).

Kód v matlabu pro nastavení veškerých parametrů můžeme vidět níže.

```
Ts = 0.0001;
n = 5;
%Parametry systemu
cfi = 0.0083377;
r = 5.505;
J = 4.3953e-07;
L = 0.01077;
b = 1.0071e-07;

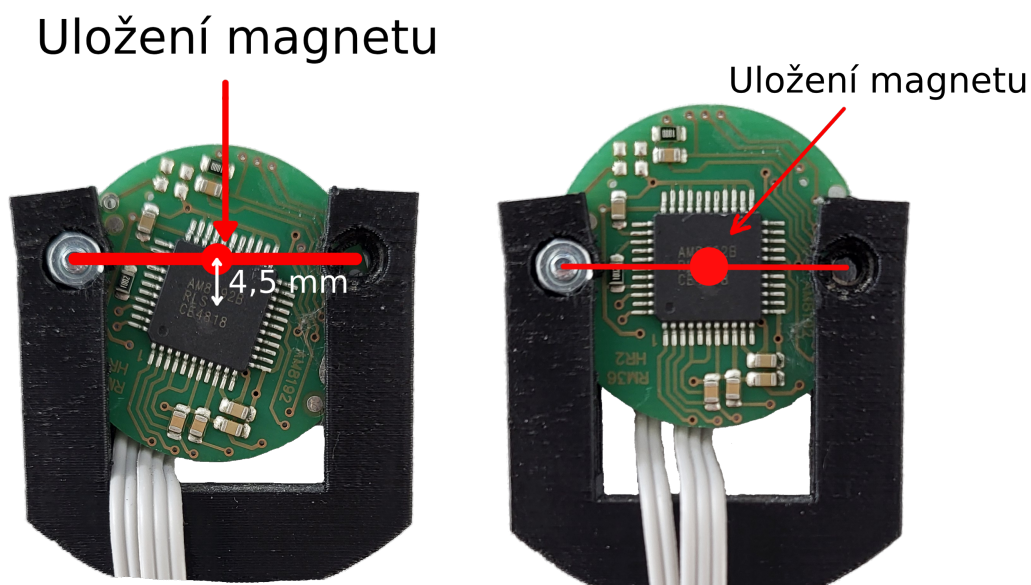
%Kontinualni system DCmotoru
A = [-r/L, -cfi/L; cfi/J, -b/J];
B = [1/L; 0];
C = [1 0; 0 1];
D = [0; 0];

Q = [0.00016, 0; 0, 0.002];
R = [1, 0; 0, 0.1];

sys = ss(A,B,C,D);
sysd = c2d(sys,Ts);

%Diskretni system DCmotoru
[Ad,Bd,C,D] = ssdata(sysd);
```

Jak bylo zmíněno v podkapitole 2.1.2, měněním vzdálenosti uložení středu enkodéru od středu magnetu na hřídeli je možné upravovat míru zašumění signálu. Proto jsem provedl dvě měření: jedno s minimálním zašuměním, kde byl magnetický enkodér umístěn dle datasheetu, tudíž střed enkodéru byl totožný se středem magnetu v hřídeli motoru a druhé s maximálním zašuměním, kde byl enkodér vyosen vůči magnetu o vzdálenost 4.5 mm. Jednalo se o maximální možné vyosení enkodéru, které ovlivňovalo pouze amplitudu šumu a nikoli jeho rozložení. Měření bylo provedeno při konstantních otáčkách, přičemž jsem enkodér měl uchycený jedním šroubem, tudíž jsem jej mohl manuálně vychylovat oproti ideální poloze. Obě uložení můžeme vidět na obrázcích 4.8.



Obrázek 4.8: Fotografie s uložením enkodéru a magnetu pro konfiguraci s maximálním a minimálním šumem

Fotografie nalevo je uložení enkodéru s maximálním šumem a fotografie napravo je s minimálním šumem.

Po zjištění těchto dvou poloh jsem mohl nastavit parametry hlavní diagonály matic Q a R pro maximální i minimální zašumění, stejně jako při simulaci DC motoru v podkapitole 3.2.2.

Na začátku jsem si náhodně zvolil parametry hlavní diagonály matic Q a R z intervalu $(0, +\infty)$.

$$Q = \begin{bmatrix} 1000 & 0 \\ 0 & 1000 \end{bmatrix} R = \begin{bmatrix} 1000 & 0 \\ 0 & 1000 \end{bmatrix} \quad (4.1)$$

S těmito parametry jsem spustil simulaci a sledoval míru zašumění signálů proudu a otáček. Během simulace jsem postupně upravoval jednotlivé parametry. Nejprve jsem začal s parametrem Q_{11} , jehož hodnotu jsem snižoval, dokud jsem nezaznamenal, že signál proudu z KF odpovídá odhadovanému systému. Interval, ve kterém docházelo k přechodu mezi těmito dvěma stavy, je $Q_{11} = (0.00001, 0.001)$. Následně jsem prováděl podobné úpravy i pro Q_{22} , který ovlivňuje otáčky. Zjistil jsem, že přechod mezi stavy, kdy měřené otáčky odpovídají filtrovaným a otáčky odhadovaného systému odpovídají KF otáčkám, nastává v intervalu $Q_{22} = (0.000001, 0.00001)$. Nakonec upravujeme oba parametry současně a sledujeme otáčky výstupující z Kalmanova filtru ve srovnání s otáčkami dynamometru, dokud nedosáhneme minimálního zašumění a nejmenší odchylky amplitud signálu v ustáleném stavu mezi dynamometrem a otáčkami z KF. Takto již získáme konkrétní kalibrované hodnoty matic.

5 Výsledky a srovnání

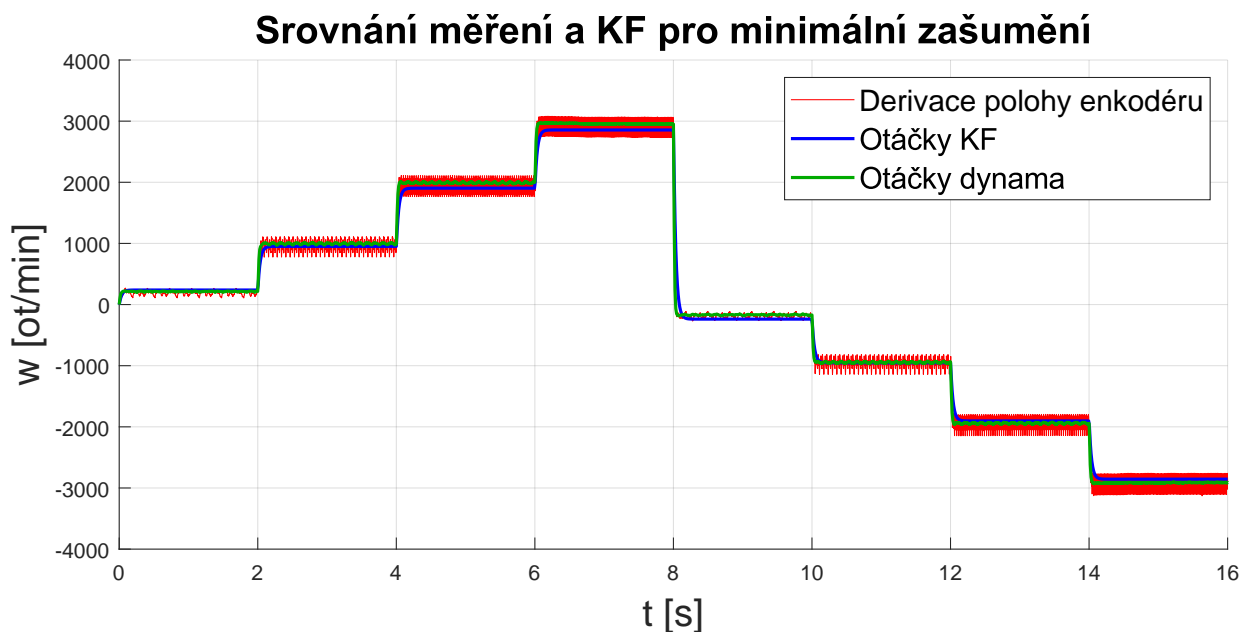
Při nastavení enkodéru do polohy s minimálním zašuměním se ukázalo jako ideální nastavení kovariančních matic Q a R , které zajistilo nejnižší šum a současně nejlépe odpovídaly charakteristice otáček dynamu.

$$Q = \begin{bmatrix} 0.00016 & 0 \\ 0 & 0.0001 \end{bmatrix} R = \begin{bmatrix} 1000 & 0 \\ 0 & 1000 \end{bmatrix} \quad (5.1)$$

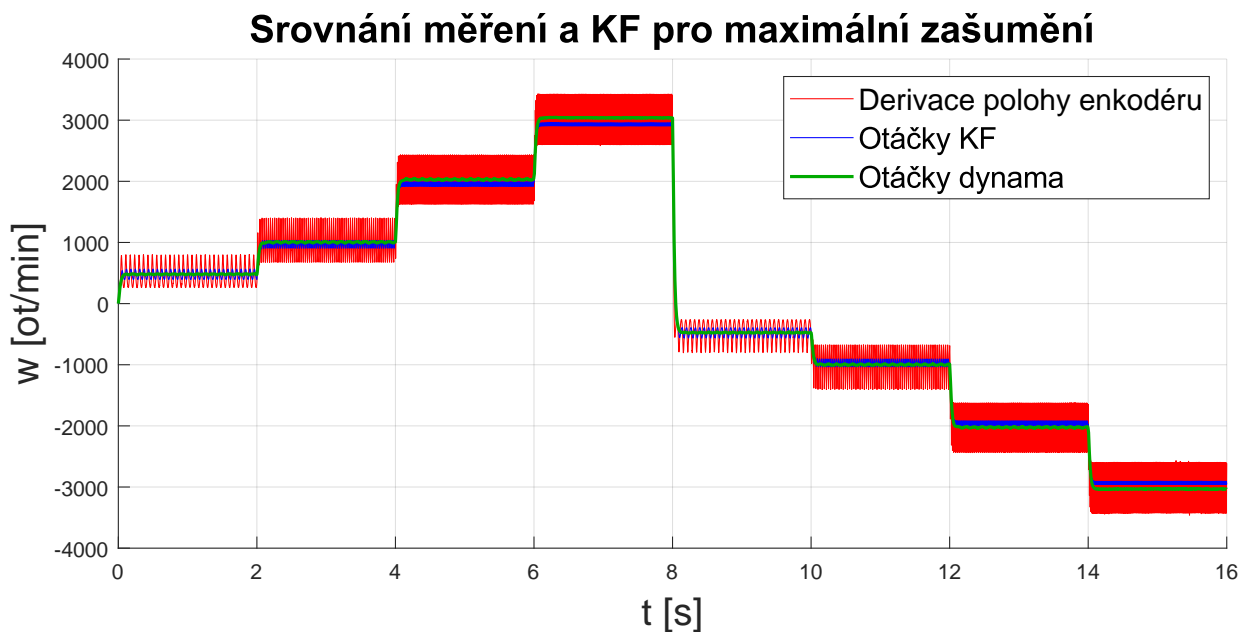
Při nastavení enkodéru do polohy s maximálním zašuměním se ukázalo jako ideální nastavení kovariančních matic Q a R , které zajistilo nejnižší šum a současně nejlépe odpovídaly charakteristice otáček dynamu.

$$Q = \begin{bmatrix} 0.00005 & 0 \\ 0 & 0.00001 \end{bmatrix} R = \begin{bmatrix} 1000 & 0 \\ 0 & 1000 \end{bmatrix} \quad (5.2)$$

5 VÝSLEDKY A SROVNÁNÍ



Obrázek 5.1: Graf srovnání měřených otáček s KF, pro minimální zašumění

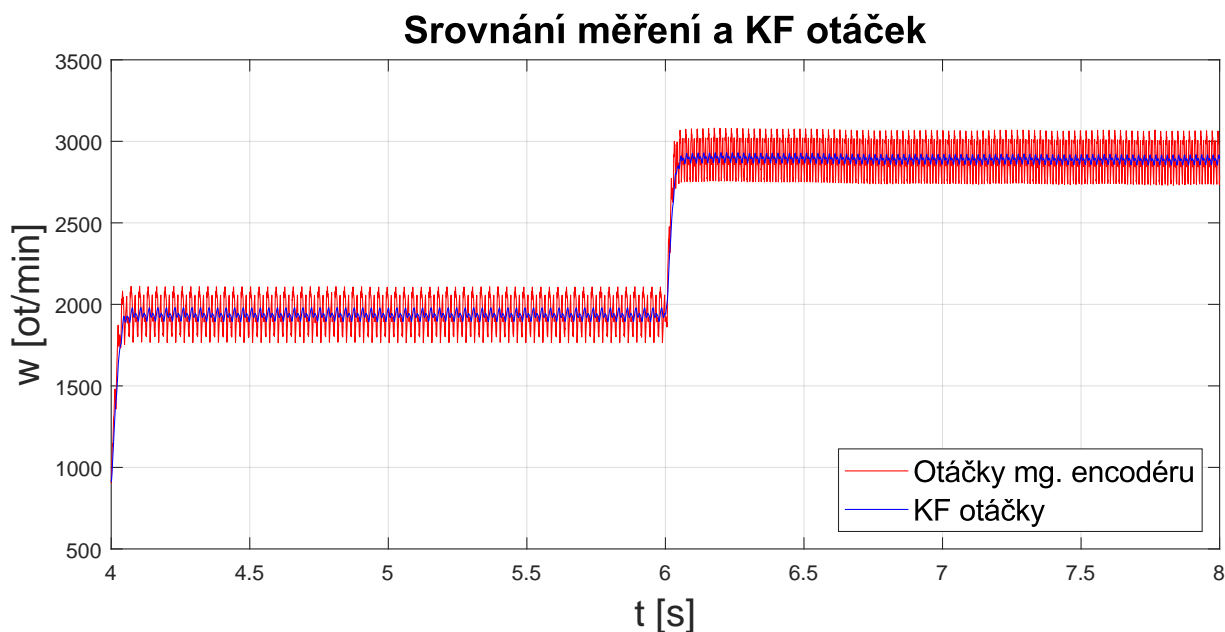


Obrázek 5.2: Graf srovnání měřených otáček s KF, pro maximální zašumění

Na těchto grafech můžeme vidět srovnání derivace polohy magnetického enkodérem při uložení pro minimální šum oproti vyfiltrovaným otáčkám nastaveného KF a otáčkám dynamometru Obr.5.1 a srovnání derivace polohy magnetického enkodérem při uložení pro maximální šum oproti vyfiltrovaným otáčkám nastaveného KF a otáčkám dynamometru na Obr.5.2.

5 VÝSLEDKY A SROVNÁNÍ

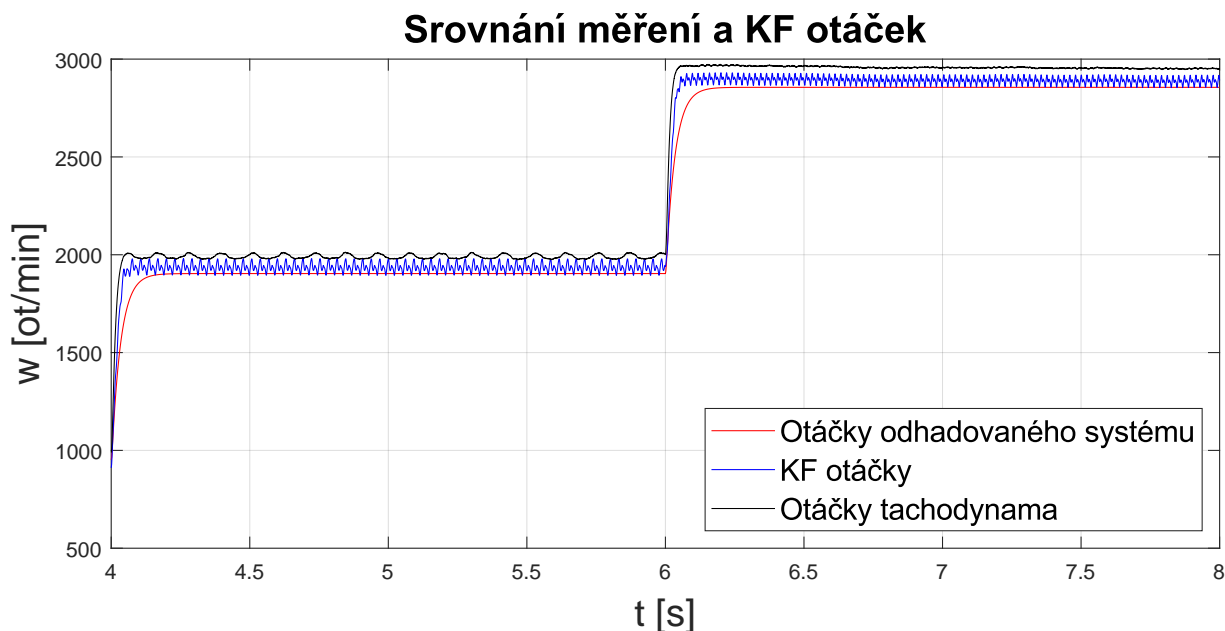
Pro lepší viditelnost zde máme přiblíženou část z grafu 5.1 v rozmezí 4s až 8s



Obrázek 5.3: Přiblížené srovnání měřených otáček s KF, pro nižší šum

Můžeme zde vidět, že velká část šumu je odstraněna a KF funguje velice dobře.

Dále můžeme vidět srovnání získaných KF otáček s odhadovaným systémem a současně s otáčkami tachodynamu, které uvažujeme jako náš přesný signál, kterému se snažíme přiblížit. Pro lepší viditelnost rovnou použijeme přiblíženou část v rozsahu 4s až 8s.



Obrázek 5.4: Přiblížené srovnání otáček z KF, dynamometru a odhad. sys. pro mg. enkoder

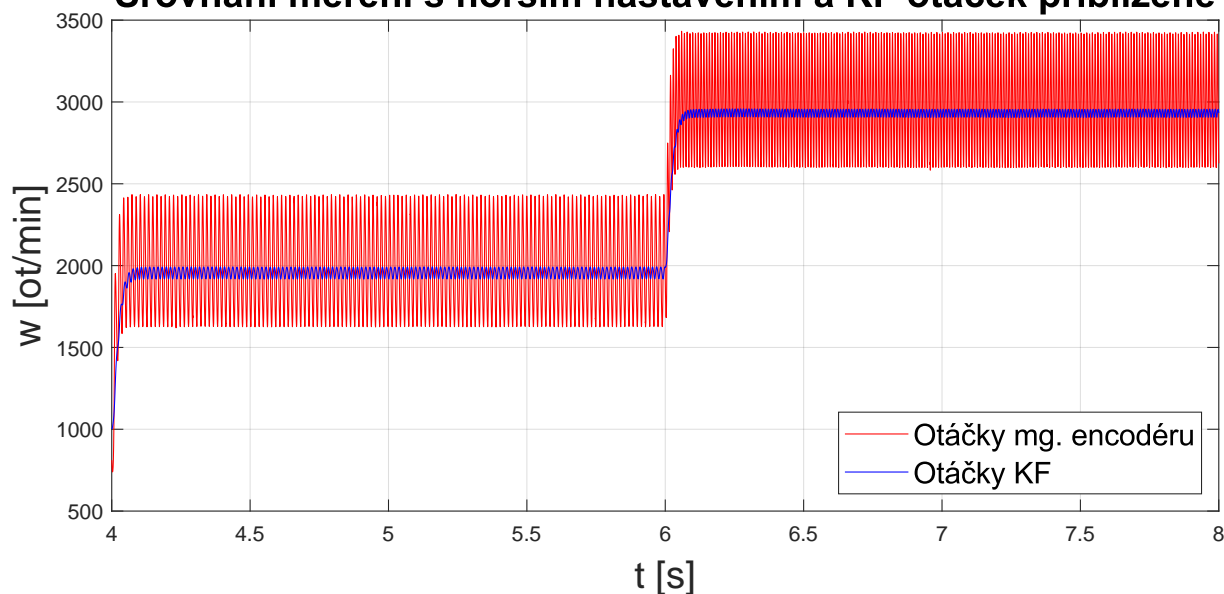
Z tohoto grafu je zřejmé, že otáčky z Kalmanova filtru vykazují rychlejší dynamické chování než odhadovaný systém, avšak nedosahují stejných hodnot jako otáčky z tacho-

5 VÝSLEDKY A SROVNÁNÍ

dynamika. Toto je způsobeno tím, že jak náš odhadovaný systém, tak i derivace polohy enkodéru jsou měřeny stejným senzorem, který oproti tachodynamu ukazuje nižší otáčky. Tudíž tuto nepřesnost Kalmanův filtr nedokáže vyřešit, jelikož kombinace těchto dvou signálů nemůže ve výsledku být vyšší než každý signál zvlášť.

Nyní výsledky pro kalibraci enkodéru na maximální šum. Opět použijeme přiblížení.

Srovnání měření s horším nastavením a KF otáček přiblížené

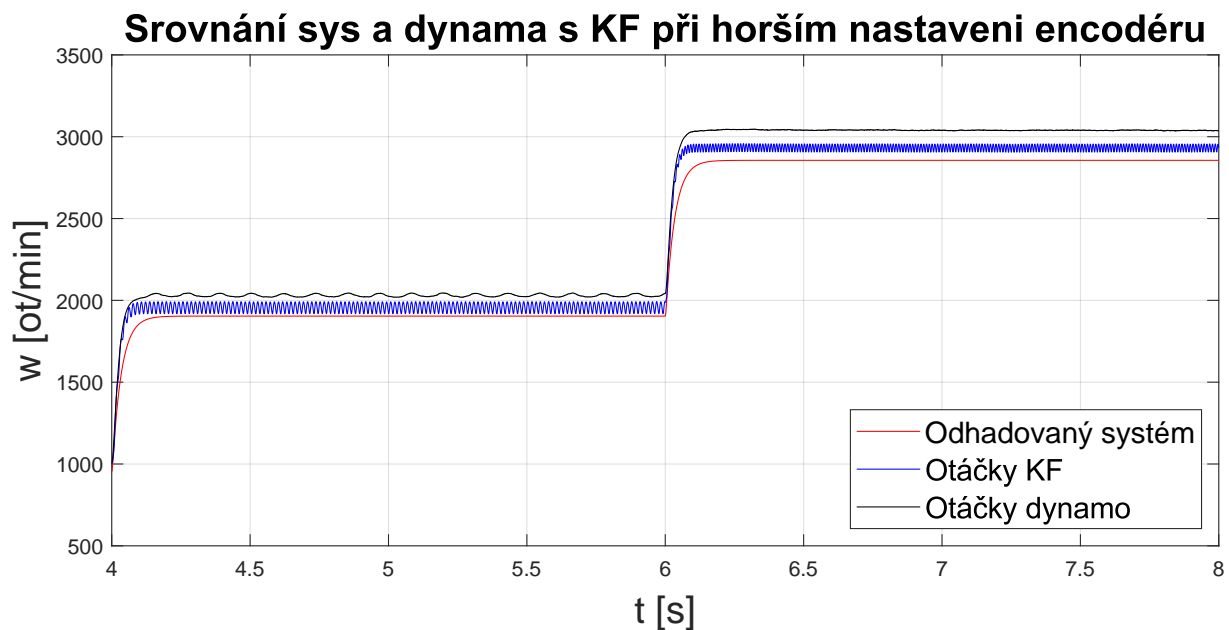


Obrázek 5.5: Přiblížené srovnání měřených otáček (větší šum) s KF

I při zvýšení šumu Kalmanův filtr velice dobře filtruje signál. Při srovnání otáček získaných Kalmanovým filtrem v měřeních s nižší a vyšší mírou zašumění je zřetelné, že oba signály mají velice podobnou přesnost. Tudíž je zřejmé, že použití Kalmanova filtru v sestavě může zaručit přesné výsledky i při použití silně zašuměného signálu, což v praxi může znamenat možnost použití horších (levnějších) senzorů.

5 VÝSLEDKY A SROVNÁNÍ

Dále opět srovnání KF otáček se systémem a tachodynamem.

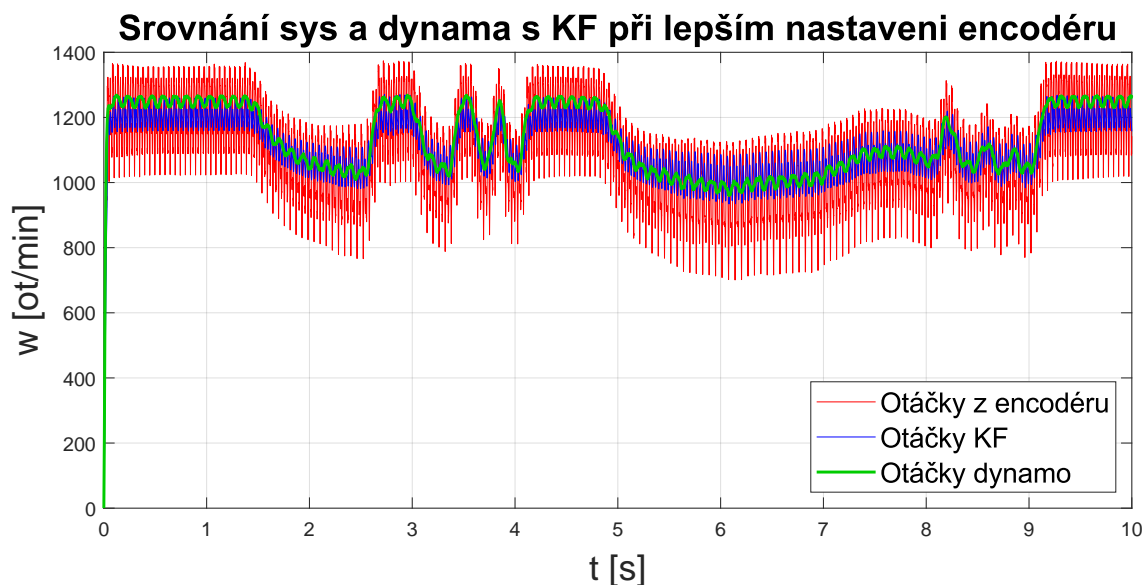


Obrázek 5.6: Přibližné srovnání otáček (větší šum) z KF, dynamometru a odhad. sys.

Zde můžeme vidět totožné výsledky jako u zapojení s nižším šumem. Opět při vyšších otáčkách Kalmanův filtr nedosahuje stejné ustálené hodnoty otáček jako dynamometr a zároveň vykazuje rychlejší dynamické vlastnosti než odhadovaný systém.

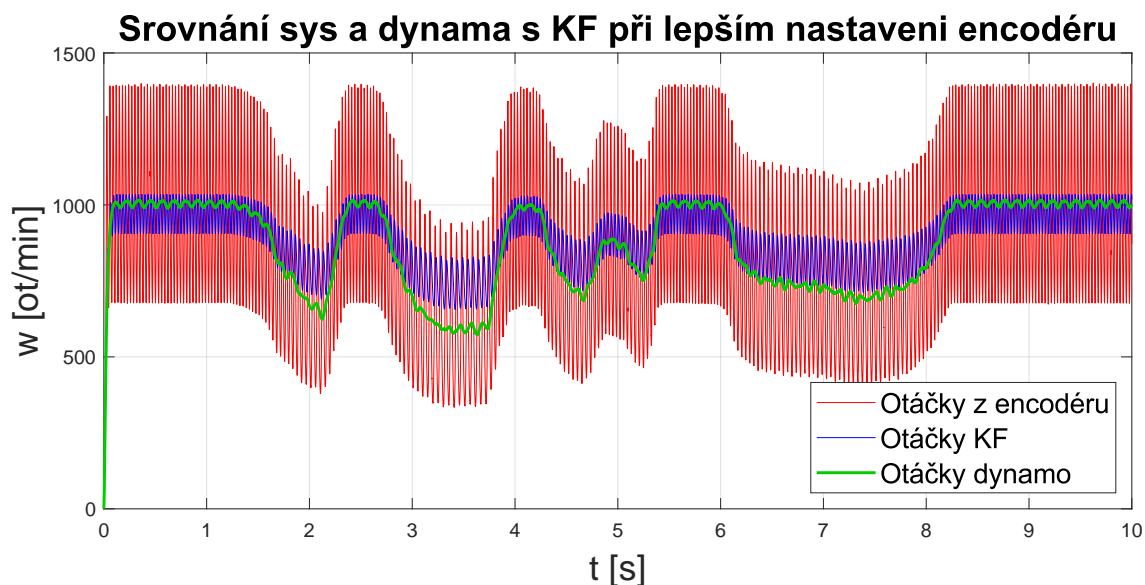
5 VÝSLEDKY A SROVNÁNÍ

Jako další pokus jsem ručně brzdil motor a sledoval, jak KF reaguje na dynamické změny.



Obrázek 5.7: Srovnání měřených otáček při minimálním šumu s KF a dynamometrem při brždění

Z grafu je zřetelné, že při nižších otáčkách a při brždění KF vykazuje velice přesné výsledky. Oproti tachodynamu má jen mírně zašuměný signál a charakteristiku má totožnou.



Obrázek 5.8: Srovnání měřených otáček při maximálním šumu s KF a dynamometrem při brždění

Stejně tak i při zvýšení šumu Kalmanův filtr velice dobře zvládá filtraci. Z těchto pokusů je tedy zřejmé, že při používání Kalmanova filtru je možné v sestavě používat levnější encodéry s větším šumem, aniž by to mělo vliv na výstup.

6 Závěr

Cílem této práce bylo vytvořit podklady k porozumění a naučení se aplikovat Kalmanův filtr na výukovou sestavu.

Provedení simulací a experimentů s Kalmanovým filtrem aplikovaným na LTI systémy prvního a druhého řádu v prostředí Simulink ukázalo, jak významné je správné nastavení kovariančních matic Q a R pro správnou funkci filtru. Experimenty s různými nastaveními těchto matic poskytly hlubší porozumění jejich vlivu na chování filtru, a to zejména ve smyslu kompromisu mezi citlivostí na modelové nepřesnosti a odolností vůči měřicímu šumu. Z výsledků získaných pomocí opakovaného simulování RC článku a vykreslování hodnot RMSE na základě měnění parametrů Q a R je zřejmé, že existuje nekonečně mnoho kombinací ideálního nastavení KF, a tedy metoda ručního nastavování parametrů v závislosti na výstupu z Q a R se ukázala jako dostatečně přesná, i když ne zrovna elegantní.

Následně jsem pomocí experimentálního měření a metody predikce parametrů získal parametry matematického modelu motoru, které jsem využil pro nastavení a kalibraci Kalmanova filtru. Experimenty byly provedeny ve dvou různých nastaveních šumu, kde jsem šum modifikoval změnou vzdálenosti magnetického enkodéru od magnetu na hřídeli motoru.

Výsledky experimentů ukázaly, že Kalmanův filtr efektivně redukuje šum a zlepšuje přesnost odhadu otáček motoru, což bylo demonstrováno ve srovnání s otáčkami měřenými tachodynamem. Filtr však nebyl schopný dosáhnout přesných hodnot v celém rozsahu otáček. Je totiž limitován přesností odhadovaného modelu, který měření nejpřesněji odpovídá v rozsahu $\pm 800rpm - 1600rpm$. Avšak i mimo tento rozsah je výstup výrazně přesnější než měřená data.

Z výzkumu také vyplývá, že Kalmanův filtr umožňuje použití méně přesných a levnějších senzorů bez zásadního ovlivnění kvality výstupních dat. Tato schopnost filtru je zvláště cenná pro aplikace, kde jsou náklady na senzory kritickým faktorem, a přitom je vyžadována vysoká úroveň přesnosti a spolehlivosti.

Další experimenty, kde byl motor ručně brzděn, prokázaly, že Kalmanův filtr byl správně nastaven a spolehlivě reaguje na dynamické změny v systému.

Tato bakalářská práce může sloužit jako studijní materiál studentům navazujícího magisterského studia oboru mechatroniky pro pochopení Kalmanova filtru.

Seznam použitých symbolů a zkratek

KF	Kalman filter
DC	Direct current
PWM	Pulse width modulation
SYS	System
MG	Magnetický
LTI	Linear time invariant
RMSE	Root mean square error

Bibliografie

- [1] WIKIPEDIA CONTRIBUTORS, Kalman filter. In: *Wikipedia, The Free Encyclopedia* [online]. 2024-05-15 [cit. 2024-05-15] Dostupné z: https://en.wikipedia.org/w/index.php?title=Kalman_filter&oldid=1223539406
- [2] ALLEGRO, Senzor proudu ACS712 datasheet [online]. 2006 [cit. 2024-05-11] Dostupné z: <https://www.alldatasheet.com/datasheet-pdf/pdf/168326/ALLEGRO/ACS712.html>
- [3] RLS, Magnetic encoder RMB30SI08BC1 datasheet [online]. 2020, 5 [cit. 2024-05-11] Dostupné z: <https://www.rls.si/eng/fileuploader/download/download/?d=1&id=7&title=Data+sheet%3A+RMB30+angular+magnetic+encoder+module+%28RMB30D01%29>
- [4] MAXONGROUP, DC-Tacho Maxon DCT 22 0.52V datasheet [online]. 2021-03 [cit. 2024-05-11] Dostupné z: https://www.maxongroup.com/medias/sys_master/root/8928344735774/EN-21-495.pdf
- [5] MAXONGROUP, Maxon RE35 273757 datasheet [online]. 2020-09 [cit. 2024-05-11] Dostupné z: https://www.maxongroup.pt/medias/sys_master/root/8992314294302/EN-22-157.pdf
- [6] WIKIPEDIA CONTRIBUTORS, Low-pass filter. In: *Wikipedia, The Free Encyclopedia* [online]. 2024-05-09 [cit. 2024-05-15] Dostupné z: https://en.wikipedia.org/w/index.php?title=Low-pass_filter&oldid=1222999762
- [7] JIM FROST, Root Mean Square Error (RMSE) [online]. 2023-05-09 [cit. 2024-05-11] Dostupné z: <https://statisticsbyjim.com/regression/root-mean-square-error-rmse/>
- [8] MAPLESOFT, DC Motor Control Design. In: *maplesoft.com* [online]. 2024 [cit. 2024-05-11] Dostupné z: <https://www.maplesoft.com/support/help/maple/view.aspx?path=applications%2FDCMotor>
- [9] MOHANADYOUSSEF, How to Tune Process and Measurement Covariances (Kalman Filter) In: *codingcorner.org* [online]. 2023-10-7 [cit. 2024-05-22] Dostupné z: <https://codingcorner.org/kalman-filter-tuning/>
- [10] MOHINDER S. Greval, ANGUS P. Andrews, Kalman filtering: theory and practice using MATLAB [online] John Wiley & Sons, 2001 [cit. 2024-05-23]. ISBN 0-471-26638-8 Dostupné z: https://www.researchgate.net/publication/243786641_Kalman_filtering_theory_and_practice_using_MATLAB

- [11] Mathworks, Parameter Estimation - MATLAB Simulink. In: *Mathworks.com* [online]. 2023 [cit. 2024-05-24] Dostupné z: https://www.mathworks.com/help/sldo/parameter-estimation.html?s_tid=srchtitle_site_search_2_Parameter%2520estimation

Seznam příloh

[1] Použité matlab skripty a Simulink modely