

BAKALÁŘSKÁ PRÁCE

**Návrh a implementace řídicího systému pro  
autodráhové vozidlo**

*Martin Lád*



Květen 2014

Vedoucí práce: Ing. Dan Martinec

České vysoké učení technické v Praze  
Fakulta elektrotechnická, Katedra řídicí techniky



České vysoké učení technické v Praze  
Fakulta elektrotechnická

katedra řídicí techniky

## ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Student: **Martin Lád**

Studijní program: Kybernetika a robotika  
Obor: Systémy a řízení

Název tématu: **Návrh a implementace řídicího systému pro autodráhové vozidlo**

Pokyny pro vypracování:


1. Vytvořte matematický model (mechanické) dynamiky autodráhového vozidla včetně základního modelu tření. Získaný model porovnejte s naměřenými daty.
2. Pro měření rychlosti navrhnete způsob fúze dat z různých senzorů vozidla.
3. Navrhnete a implementujete rychlostní regulátor pro autodráhové vozidlo.
4. Očekávaný výsledek je uzavřená rychlostní smyčka.

Seznam odborné literatury:

- [1] Martinec, D.; Sebek, M.; Hurak, Z., "Vehicular platooning experiments with racing slot cars", Control Applications (CCA), 2012 IEEE International Conference on , vol., no., pp.166,171, 3-5 Oct. 2012.
- [2] Chiasson, J., "Modeling and High Performance Control of Electric Machines ", IEEE Press Series on Power Engineering, Wiley-IEEE Press; 1 edition (April 7, 2005)
- [3] Hughes, A.; Drury, B., "Electric Motors and Drives: Fundamentals, Types and Applications", Newnes; 4 edition (May 24, 2013)

Vedoucí: Ing. Dan Martinec

Platnost zadání: do konce letního semestru 2014/2015

  
prof. Ing. Michael Sebek, DrSc.  
vedoucí katedry



  
prof. Ing. Pavel Ripka, CSc.  
děkan

V Praze dne 3. 1. 2014

## Poděkování

Tímto bych chtěl poděkovat vedoucímu práce Ing. Danovi Martincovi za příkladné vedení práce a ochotu při řešení zadání a také Ing. Ivo Hermanovi za věcné rady.

## 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, dne 22. 5. 2014 .....

Lád

Podpis



## Abstrakt

Tento dokument se zabývá modelováním autodráhového autíčka pomocí techniky vazebních grafů. Dále se zabývá návrhem a implementací dvěma způsoby měření rychlosti autíčka, a to pomocí IRC senzoru a měření *back emf*. Obě metody mají své pro a proti, proto byl k filtraci a fúzi obou metod navržen a implementován Kalmanův filtr, který jejich nevýhody potlačuje. Kalmanův filtr je také použit pro odhad třecí síly působící na autíčko. Tento odhad byl také použit pro modelování. V poslední řadě se zabývá návrhem a implementací rychlostního PI regulátoru, který jak se ukazuje funguje velice dobře i při vnějších poruchách a je schopen regulovat rychlost již od  $50 \text{ mms}^{-1}$ . Veškerá implementační úkoly byly realizovány jako program do mikroprocesoru v jazyce C.

## Klíčová slova

Vazební graf, Kalmanův filtr, IRC sensor, back emf, PI regulátor

## **Abstrakt**

This thesis deals with modeling of slotcar by bond graphs. It also deals with design and implementation of two speed measuring methods: a) by IRC sensor and b) by *back emf* measurement. Both methods have pros and cons. A Kalman filter has been designed and implemented to filter and combine these two measurements. Kalman filter is also used to estimate friction used in system modeling. The final result is a discrete PI controller for speed of the slotcar. The measurements show the controller works very well even with external disturbances and it is able to regulate even relatively small speeds such as  $50 \text{ mms}^{-1}$ . All implementation tasks have been implemented as a program in a microprocessor in C language.

## **Keywords**

Bond graph, Kalman filter, IRC sensor, back emf, PI controller

# Obsah

|  |           |
|--|-----------|
| <b>1. Úvod</b>   | <b>1</b>  |
| <b>2. Popis a modelování systému</b>                                 | <b>3</b>  |
| 2.1. Model autodráhového autíčka . . . . .                           | 3         |
| 2.1.1. Parametry modelu . . . . .                                    | 6         |
| 2.1.2. Simulace modelu . . . . .                                     | 7         |
| <b>3. Měření rychlosti</b>   | <b>9</b>  |
| 3.1. Měření rychlosti pomocí IRC senzoru . . . . .                   | 9         |
| 3.1.1. Hardwarová implementace . . . . .                             | 9         |
| 3.1.2. Algoritmus pro měření . . . . .                               | 11        |
| 3.2. Měření rychlosti DC motorku pomocí back EMF . . . . .           | 12        |
| Měření back emf . . . . .  | 12        |
| Výpočet rychlosti z hodnoty back emf . . . . .                       | 14        |
| 3.3. Filtrace a fúze senzorů pomocí Kalmanova filtru . . . . .       | 15        |
| 3.3.1. Obecné rovnice Kalmanova filtru . . . . .                     | 16        |
| 3.3.2. Implementace Kalmanova filtru . . . . .                       | 16        |
| 3.3.3. Ukázka funkčnosti implementovaného Kalmanova filtru . . . . . | 20        |
| <b>4. Regulace - rychlostní regulátor</b>                            | <b>22</b> |
| 4.1. Návrh regulátoru . . . . .                                      | 22        |
| 4.2. Ukázky regulace . . . . .                                       | 23        |
| <b>5. Závěr</b>  | <b>26</b> |
| <b>Přílohy</b>   |           |
| <b>A. Obsah přiloženého CD</b>                                       | <b>27</b> |
| A.1. Obsah složky Platoon_implementation . . . . .                   | 27        |
| A.2. Práce s projektem přiloženým na CD . . . . .                    | 27        |
| <b>Literatura</b>  | <b>28</b> |



# 1. Úvod

Tato bakalářská práce je vedena v rámci projektu Slotcar platooning, který zastřešuje skupina AA4CC. Projekt má za cíl vytvořit platformu pro testování distribuovaného řízení kolony vozidel. Myšlenkou je vytvořit jednoduchý model (Obr. 1), na kterém by se daly testovat různé metody řízení. Základem platformy jsou autodráhová autíčka doplněná o elektroniku, která se stará o řízení. Finálním stavem projektu by měla být kolona až padesáti autodráhových autíček.

Na projektu spolupracuji s Janem Moravcem, který má za úkol měření vzdálenosti, její regulaci a odzkoušení některých způsobů řízení kolony. Dále spolupracuji s Ing. Jaromírem Dvořákem, který má na starosti HW a jádro SW. Projekt se nachází ve stádiu vývoje, kdy je navržen HW. Momentálně se navrhují a implementují principy měření rychlosti a vzdálenosti. Dále se navrhují a testují regulátory pro obě měřené veličiny a s nimi se zkusí základní metody řízení kolony.

Mým úkolem na projektu je řízení rychlosti autíčka. K tomu je potřeba vyřešit následující body (a) modelování systému, (b) měření rychlosti a (c) návrh rychlostního regulátoru.

- (a) Autíčko bude modelováno pomocí techniky vazebních grafů jako nelineární model, kde nelineární složkou bude působící tření. Tření pro jednoduchost aproximujeme konstantním statickým třením a dynamickým třením, které budeme uvažovat jako lineárně závislé na rychlosti. Koeficienty tření odhadneme pomocí Kalmanova filtru. Následně bude provedena linearizace modelu.
- (b) Rychlost bude měřena pomocí dvou na sobě nezávislých metod, a to pomocí



Obrázek 1. Fotografie kolony aut (autor Jan Moravec)

## 1. Úvod

IRC senzoru a měření *back emf* na motoru. Pro filtraci a fúzi dat z obou senzorů bude použit Kalmanův filtr. V práci se budu zabývat zejména jeho použitím a implementací do procesoru v jazyce C. Při návrhu Kalmanova filtru bude uvažován nelineární model a třecí síla bude zavedena jako stav, který se bude na základě měření odhadovat.

- (c) Rychlostní regulátor bude navržen jako diskrétní PI regulátor, který bude implementován v procesoru. Při návrhu musí být bráno v potaz omezení na akční zásah. Rychlostní regulátor bude zapojen v kaskádě za regulátorem vzdálenosti.

Na projektu jsem spolupracoval zejména s Janem Moravcem, a to při návrhu hlavního zdrojového souboru, ovládacího modelu do Matlabu a testování regulace.

## 2. Popis a modelování systému

Vozidla jsou postavena na základě autodráhových autíček Carrera Ford Capri (Obr. 2). Do autíček byla přidána řídicí elektronika, kde je srdcem ARM procesor z rodiny STM32F405, který zajišťuje veškeré výpočetní operace. Vnitřek autíčka i karosérie musely být upraveny tak, aby se dovnitř dala umístit řídicí elektronika i se senzory vzdálenosti. Autíčko obsahuje klasický DC motorek se stacionárním permanentním magnetem, rotorem se třemi smyčkami a komutátorem. Motor je spojen přes převod k zadní nápravě. Vnitřek autíčka je zobrazen na Obr. 3. Z výroby jsou na podvozku umístěny dva permanentní magnety, které slouží pro přitlačování autíčka k dráze. Jeden magnet je umístěn v prostředku autíčka, druhý pod zadní nápravou. Magnet uprostřed byl odebrán. Dráha je tvořena dvěma vodivými koleji, na které je připojen zdroj napětí. Vpředu na podvozku je umístěna otočná vodící lišta, která vede autíčko mezi koleji při pohybu na dráze. Z každé strany této lišty je umístěn sběrač, přes který je autíčko napájeno z kolejí.

### 2.1. Model autodráhového autíčka

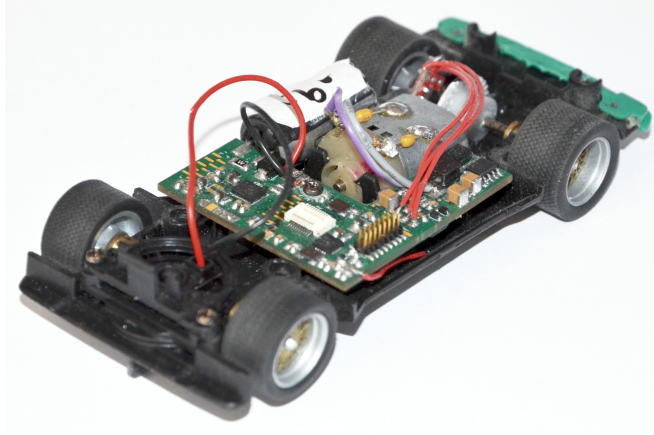
K vytvoření modelu byla použita technika vazebních grafů. Motor byl namodelován jako sériová kombinace elektrického odporu  $R$ , induktoru s indukčností  $L$  a gyrátoru  $G$ . Gyrátor  $G$  s konstantou  $k$  zprostředkovává přenos energie mezi elektrickou a mechanickou částí. Točivý moment motoru je dán jako

$$\tau_m = k_T i \quad (1)$$



Obrázek 2. Autíčko Carrera Ford Capri

## 2. Popis a modelování systému



Obrázek 3. Pohled na vnitřní uspořádání autíčka

a elektromotorické napětí na motoru je dáno následovně

$$\xi = k_b \omega . \quad (2)$$

Ze zákona zachování energie podle [1] lze ukázat, že

$$k_T = k_b = k . \quad (3)$$

Výstupem motoru je moment síly, který se transformuje přes transformátor  $T_1$  s konstantou  $n$  a transformátor  $T_2$  s konstantou  $r$ , kde  $n$  je poměr převodu a  $r$  je poloměr kolečka. Setrvačnost autíčka reprezentujeme inertancí s konstantou  $m$ , která představuje hmotnost autíčka. Rychlost autíčka  $v$  je určena na základě síly dodávané od motoru sníženou o třecí sílu  $F_t$ , která působí proti síle motoru. Předpokládáme, že nedochází k prokluzu mezi koly a vozovkou. Dále zanedbáváme moment setrvačnosti rotoru a hřídelí vedoucích od motoru ke kolečkům, protože mají zanedbatelnou hmotnost vůči autíčku, a také zanedbáváme jejich případnou pružnost, protože uvažujeme, že jsou pro působící síly ideálně pevné. Vazební graf je zobrazen na Obr. 4.

Stavové rovnice modelu určené na základě vazebního grafu jsou následující

$$\dot{P}_1(t) = u(t) - \frac{R}{L}P_1(t) - \frac{k}{rnm}P_2(t) , \quad (4)$$

$$\dot{P}_2(t) = \frac{k}{Lnr}P_1(t) - F_t(P_2(t)) . \quad (5)$$

Dále si z vazebního grafu vyjádříme vztahy pro proud protékající motorem  $i$  a rychlost autíčka  $v$ , které jsou

$$i(t) = \frac{P_1(t)}{L} , \quad (6a)$$

$$v(t) = \frac{P_2(t)}{m} . \quad (6b)$$



Po dosazení (6a), (6b) do (4), (5) získáme diferenciální rovnice modelu

$$\frac{di(t)}{dt} = \frac{u(t)}{L} - \frac{R}{L}i(t) - \frac{k}{rnL}v(t) , \quad (7)$$

$$\frac{dv(t)}{dt} = \frac{k}{mnr}i(t) - \frac{F_t(v(t))}{m} , \quad (8)$$

$$y(t) = 1000v(t) . \quad (9)$$

Z parametrů motoru (Tab. 1) můžeme spočítat časovou konstantu pro sériový RL obvod  $\frac{L}{R} = 400 \mu s$ . Je zřejmé, že dynamika samotného autíčka je mnohokrát vyšší, proto zavedeme předpoklad, že

$$\frac{di(t)}{dt} = 0 . \quad (10)$$

Rovnici (10) dosadíme do (7) a vyjádříme si  $i(t)$  jako

$$i(t) = \frac{u(t)}{R} - \frac{k}{rn}v(t) . \quad (11)$$

Rovnici (11) dosadíme do (8), čímž po úpravě dostaneme

$$\frac{dv(t)}{dt} = \frac{ku(t)}{Rmnr} - \frac{k^2v(t)}{Rmn^2r^2} - \frac{F_t(v(t))}{m} , \quad (12)$$

$$y = 1000v(t) . \quad (13)$$

Touto úpravou jsme získali model, kde máme pouze jednu proměnnou, a to rychlost  $v$  a třecí sílu  $F_t$  funkčně závislou na rychlosti. Třecí sílu  $F_t$  uvažujeme jako součet sil statického a dynamického tření ve tvaru

$$F_t(v(t)) = bv(t) + F_t^s , \quad (14)$$

kde  $b$  je koeficient dynamického tření a  $F_t^s$  je mezní síla statického tření. Jelikož třecí síla obsahuje nelinearitu v podobě přičítání konstanty, tak k získání přenosu je nutno model linearizovat.

Výsledný přenos z napětí  $u$  na rychlost  $v$  po linearizaci má následující tvar

$$H_{u \rightarrow v}(s) = \frac{1000 \frac{k}{Rmnr}}{s + \frac{b}{m} + \frac{k^2}{Rmn^2r^2}} . \quad (15)$$

V našem případě je motor buzen PWM signálem s periodou  $25 \mu s$ , což je podstatně méně než časová konstanta motoru, proto lze uvažovat, že napětí  $u$  je rovno střední hodnotě PWM signálu

$$u = U_m D(t) , \quad (16)$$

## 2. Popis a modelování systému

kde  $D$  je střída PWM signálu a  $U_m$  je napájecí napětí.  $U_m$  je dáno jako

$$U_m = 0.96(U_{\text{zdroj}} - U_{\text{dioda}}) , \quad (17)$$

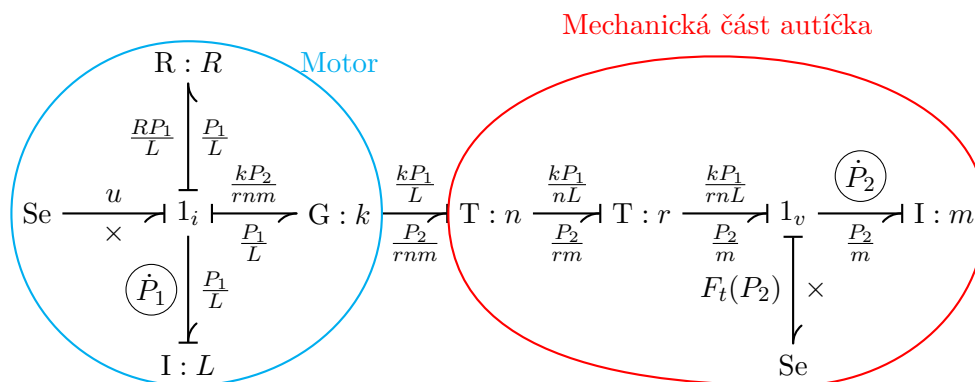
kde  $U_{\text{zdroj}}$  je napětí zdroje,  $U_{\text{dioda}} = 0.3$  V je úbytek napětí na vstupní diodě. Konstanta 0.96 je důsledek úbytku napětí na měřícím odporu pro měření proudu na motoru. Dosazením (16) a (14) do (12) dostaneme výsledný nelineární model ve tvaru

$$\frac{dv(t)}{dt} = \frac{kU_m D(t)}{Rmnr} - \frac{k^2 v(t)}{Rmn^2 r^2} - \frac{bv(t) + F_t^s}{m} . \quad (18)$$

$$y = 1000v(t) . \quad (19)$$

Výsledný přenos ze střídavy na rychlost je

$$H_{D \rightarrow v}(s) = U_m H_{u \rightarrow v}(s) . \quad (20)$$



Obrázek 4. Vazební graf pro model autíčka

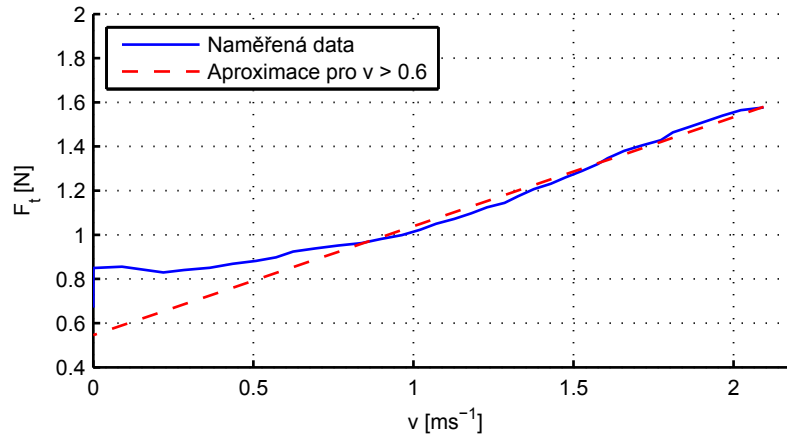
### 2.1.1. Parametry modelu

Hodnoty parametrů modelu jsou zaneseny v Tab. 1. Parametry motoru byly převzaty z [2].

| Název                               | Značka | Hodnota            | Jednotka           |
|-------------------------------------|--------|--------------------|--------------------|
| Elektrický odpor vinutí motoru      | $R$    | 5                  | $\Omega$           |
| Elektrická indukčnost vinutí motoru | $L$    | $2 \cdot 10^{-3}$  | H                  |
| Konstanta motoru                    | $k$    | 0.0061             | $\text{Nm A}^{-1}$ |
| Hmotnost autíčka                    | $m_A$  | 0.094              | kg                 |
| Poloměr kolečka                     | $r$    | $10 \cdot 10^{-3}$ | m                  |
| Konstanta převodu                   | $n$    | 1/3                | -                  |

Tabulka 1. Parametry modelu autíčka

Koeficienty třecí síly byly změřeny pomocí Kalmanova filtru (blíže v sekci Filtrace a fúze senzorů pomocí Kalmanova filtru). Měření (Obr. 5) bylo provedeno



Obrázek 5. Závislost třecí síly na rychlosti

pro celý rozsah vstupní střídy při jízdě na kruhové dráze. Pro tento test bylo vybráno autíčko číslo 5. Bylo zjištěno, že třecí síla nemá předpokládaný lineární charakter, ale do určité rychlosti třecí síla klesá, a poté začne stoupat. Tento jev se nazývá Stribeckův efekt [3]. Model zaměřujeme více na použití ve vyšších rychlostech, kdy již lze třecí sílu aproximovat přímkou. Tuto aproximaci jsme zavedli od hodnoty  $0.5 \text{ ms}^{-1}$  výše, kde již lze závislost považovat za lineární. Aproximace třecí síly je ve tvaru

$$F_t = 0.5v + 0.55 . \quad (21)$$

V modelu je za dobu náběhu primárně odpovědná hmotnost autíčka. Měření ukázala, že simulace se změřenou hmotností neodpovídala reálné době náběhu, proto byl tento parametr upraven na hodnotu

$$m = 0.294 \text{ kg} . \quad (22)$$

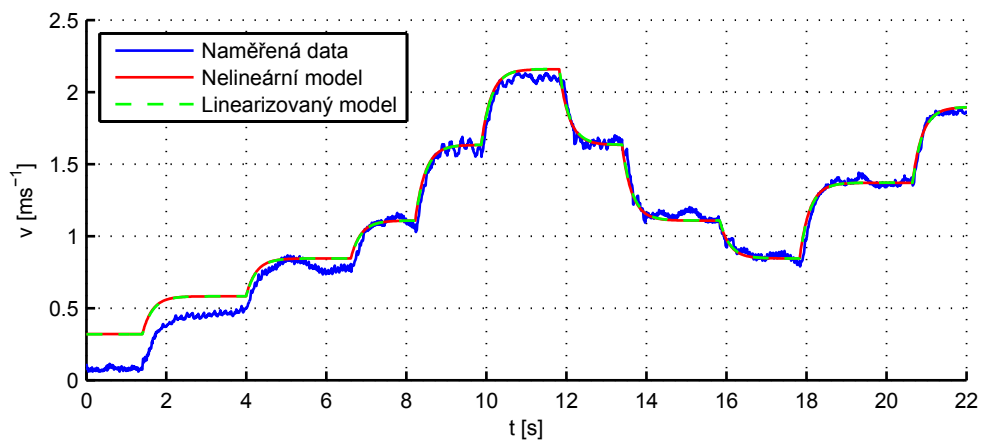
### 2.1.2. Simulace modelu

Pro simulaci uvažujeme napětí  $U_{\text{zdroj}} = 9 \text{ V}$ . Vyčíslený linearizovaný model pro rychlosti větší než  $0.5 \text{ ms}^{-1}$  má přenos ve tvaru

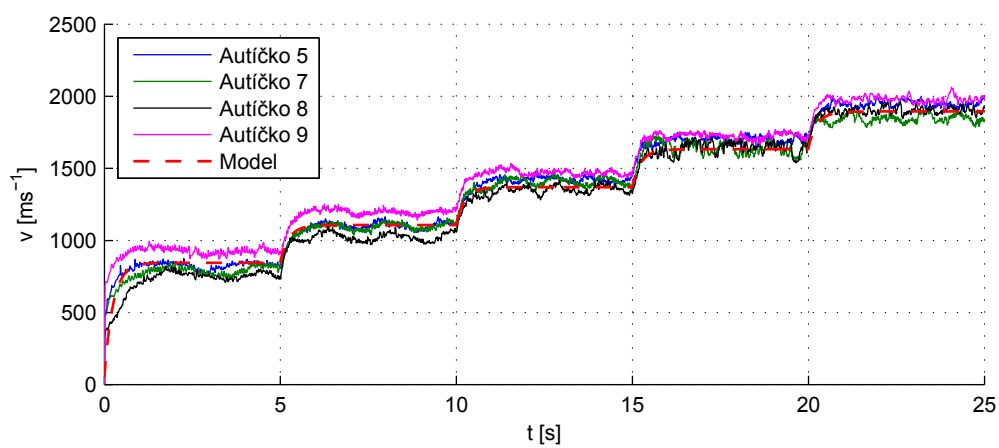
$$H(s) = \frac{10400}{s + 3.96} . \quad (23)$$

Byla provedena měření (Obr. 6), při kterém byl porovnán nelineární model, linearizovaný model a realita. Také byl proveden test (Obr. 7), kdy byla porovnána různá autíčka na stejný vstupní signál. Lze vidět, že se rychlosti autíček liší, což je následek toho, že každé autíčko má jiné vnitřní tření.

## 2. Popis a modelování systému



Obrázek 6. Porovnání rychlosti reality a simulace



Obrázek 7. Porovnání rychlostí různých autíčků na stejný vstup

## 3. Měření rychlosti

V této kapitole jsou popsány dva způsoby měření rychlosti autíčka. První způsob je měření napětí na svorkách otáčejícího se DC motoru, které je lineárně úměrné rychlosti. Druhou možností je použít IRC senzor pro měření otáček. Obě metody měří rychlost otáčení kol, tudíž pro určení výsledné rychlosti autíčka musíme předpokládat, že nedochází k prokluzu mezi koly a vozovkou.

Dalším způsobem měření rychlosti je použití akcelerometru. Výhodou této metody je, že se již neměří rychlost otáčení kol, ale měří se rychlost autíčka jako celku. Nevýhodou tohoto senzoru je, že se měří akcelerace, z níž se rychlost dopočítává integrací. Tím, že se integruje výstup senzoru, tak se integruje také jeho šum a chyba, čímž se poté může dosáhnout chybného měření. Z tohoto důvodu tato varianta měření zatím nebyla použita.

Pro filtraci a fúzi dat z měření rychlosti pomocí IRC senzoru a měření *back emf* je použit Kalmanův filtr.

Měření rychlosti se provádí periodicky, perioda měření  $T_m$  je

$$T_m = 2 \text{ ms} . \quad (24)$$

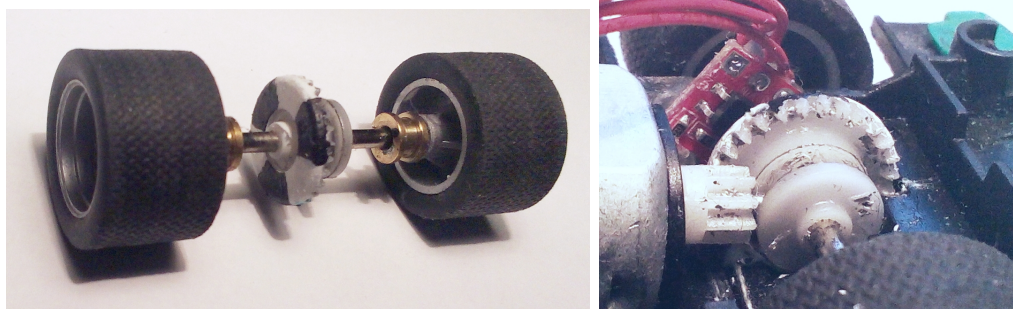
### 3.1. Měření rychlosti pomocí IRC senzoru

IRC senzor, inkrementální rotační enkodér (Incremental Rotary enCoder), se skládá z infračerveného optického vysílače a přijímače a rotačního disku, na kterém jsou střídavě naneseny barvy, v našem případě bílá a černá. Využívá se vlastnosti barevných ploch, které mají různou schopnost pohlcovat světlo. Černá barva pohlcuje více světla, než bílá barva. IRC senzor vysílá kontinuálně infračervené světlo na rotační disk. U odraženého světla se měří intenzita, z které se určí barva plochy před senzorem. Při změně barvy dojde ke změně výstupního signálu, u kterého můžeme měřit periodu nebo frekvenci, z čehož lze dopočítat rychlost.

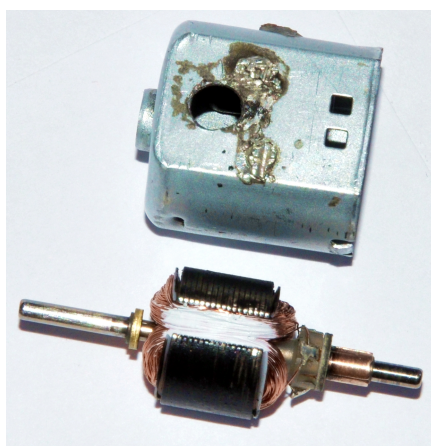
#### 3.1.1. Hardwarová implementace

Základem je infračervený senzor QRE1113 od firmy Sparkfun, který obsahuje vysílač i přijímač v jednom pouzdru. Senzor je umístěn proti rotačnímu disku s barevnými pruhy. Jako rotační disk je využito ozubené kolo v převodu na zadní nápravě, které je dostatečně velké, aby se na něho daly pruhy nanést. Na toto kolo bylo naneseno šest pruhů, tři černé a tři bílé. Konstrukční provedení je zobrazeno na Obr. 8. Z realizačních důvodů nejsou bílé pruhy stejně široké jako černé. Pruhy byly nanášeny podle zubů na ozubeném kole. Zubů je 27, proto byly černé pruhy

### 3. Měření rychlosti



Obrázek 8. Disk s barevnými pruhy a umístění senzoru QRE1113 na zadní nápravě



Obrázek 9. Varianta s vrtáním motoru; otvor v plášti motoru a barevné pruhy na rotoru

naneseny na šířku 5 zubů a bílé na šířku 4 zubů. Výsledná funkce senzoru vypadá tak, že vysílač infračerveného světla kontinuálně vysílá paprsky na rotační disk, kde se světlo odrazí od příslušné barvy, která se zrovna nachází před senzorem. Odražené světlo se přijímá přijímačem, který následně převede úroveň jeho intenzity na výstupní napětí; nízká úroveň napětí odpovídá bílé barvě, vysoká napěťová úroveň odpovídá černé barvě. Toto napětí je následně převedeno A/D převodníkem procesoru na bitovou hodnotu a zpracováno příslušným vyhodnocovacím algoritmem.

V původním návrhu byl jako odrazný rotační objekt použit rotor motoru. Návrh byl takový, že se do pláště motoru vyvrtal otvor, do kterého byl vsazen senzor. Na rotor se nanesly bílé a černé pruhy. Konstrukce je zobrazena na Obr. 9. Tato metoda měla velikou výhodu v tom, že se měřily přímo otáčky motoru, které jsou díky převodu třikrát větší než otáčky kol. Tudíž jsme informaci o stejné rychlosti dostávali častěji než je tomu teď, protože časy přechodů mezi jednotlivými barvami byly kratší. Nevýhodou této metody jsou problémy s vyvrtáním otvoru. Motor se musel nejprve rozebrat, vyvrtat otvor, a poté znovu složit. Po takovém zásahu byl vždy zaznamenán znatelný pokles výkonu motoru, protože se zvýšilo tření uvnitř motoru.

### 3.1.2. Algoritmus pro měření

Algoritmus měření rychlosti funguje na principu měření periody. Perioda je v tomto případě doba přechodu přes bílý a černý pruh dohromady, protože měření (Obr. 10) ukázalo, že přechody přes černou a přes bílou nejsou detekovány jako stejně dlouhé. Přechod přes bílou je detekován jako delší než přes černou, přestože ve skutečnosti jsou bílé pruhy užší než černé. Pro zvýšení přesnosti měření se měří jak perioda z bílé na bílou, tak i z černé na černou. Tím budeme dostávat informaci o rychlosti dvakrát častěji. Tyto dvě měření budou od sebe fázově posunuty. V ideálním případě by toto posunutí bylo o polovinu periody, ale díky nesymetrickým dobám přechodu to tak nebude.

Procesor kontinuálně vzorkuje A/D převodníkem vstupní kanály tak, jak jsou seřazeny ve frontě, ve které je zařazen i kanál, na který je připojen výstup senzoru QRE1113. Samotné měření probíhá tak, že vyčítáme hodnotu převodu A/D převodníku v obsluze přerušeni vyvolaným čítačem na frekvenci  $f_{vf} = 40$  kHz. Frekvence vyčítání dat je řádově nižší, než je frekvence převodu, proto můžeme předpokládat, že vyčtená hodnota je vždy aktuální.

QRE1113 je analogový senzor. Jeho výstup se mění v závislosti na intenzitě odraženého světla. Při přechodech mezi barvami není změna výstupu skoková, ale plynule se mění, proto je potřeba zvolit prahovou hodnotu, kde již končí jedna barva a začíná druhá. K této prahové hodnotě je nutné zvolit i hysterezi, která zabrání detekci falešných přechodů. Hodnota prahu i hystereze byla určena experimentálně. Analogový signál je tímto způsobem převeden na dvoustavový signál, který je vyobrazen modrou barvou na Obr. 10; úrovní 1 odpovídá černá barva, úrovní 0 odpovídá bílá barva. Z grafu lze vidět, že bílé a černé pruhy nejsou z hlediska času přechodu rovnocenné, avšak perioda  $\tau$  je konstantní. Perioda  $\tau$  se měří čítačem s frekvencí  $f_{vf}$  a je dána jako

$$\tau = \tau_{\text{černá}} + \tau_{\text{bílá}} , \quad (25)$$

kde  $\tau_{\text{černá}}$  je doba přechodu nad černou barvou a  $\tau_{\text{bílá}}$  je doba přechodu nad bílou barvou. Dále si definujeme úhel  $\varphi_{b\check{c}}$ , který je dán jako úhel, o který se kolo musí potočit, aby byla detekována sekvence barev bílá-černá-bílá, nebo černá-bílá-černá. Tento úhel je konstantní a je dán jako

$$\varphi_{b\check{c}} = \frac{2\pi}{\frac{A_{\text{bílá}} + A_{\text{černá}}}{2}} , \quad (26)$$

kde  $A_{\text{bílá}} = 3$  je počet bílých pruhů,  $A_{\text{černá}} = 3$  je počet černých pruhů na ozubeném kole.

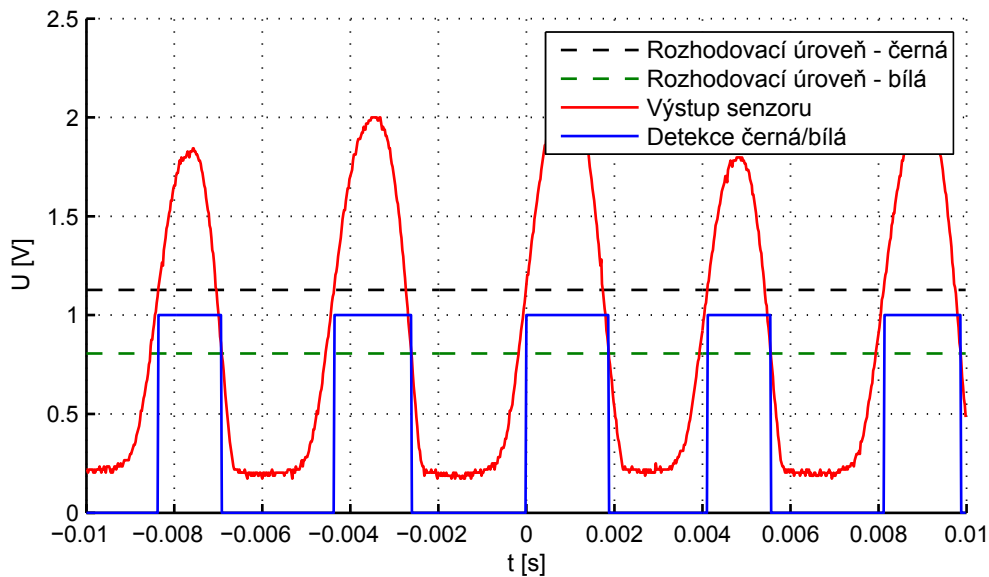
Čas  $\tau$  je dán také jako

$$\tau = \frac{N}{f_{vf}} , \quad (27)$$

kde  $N$  je počet tiků čítače, za kterých se kolo potočí o  $\varphi_{b\check{c}}$ .

Pro výpočet rychlosti využijeme vzorec, že obvodová rychlost se rovná úhlové rychlosti vynásobené poloměrem, kde úhlovou rychlost dostaneme jako úhel poo-

### 3. Měření rychlosti



Obrázek 10. Měření rychlosti pomocí IRC senzoru

točení za čas. Výsledný vzorec je tedy následující

$$v = \frac{\varphi_{b\check{c}r}}{\tau}, \quad (28)$$

kde  $r$  je poloměr kolečka.

Dosažením vzorců (26), (27) do (28) dostaneme finální vztah pro rychlost

$$v = \frac{4\pi r f_{vt}}{(A_{b\acute{i}l\acute{e}} + A_{\check{c}ern\acute{e}}) N}. \quad (29)$$

## 3.2. Měření rychlosti DC motorku pomocí back EMF

Jedna z vlastností DC motoru je, že pokud se rotor otáčí, tak motor na svých svorkách generuje napětí. Toto napětí se nazývá *back emf* (back electromotive force). Pro nás nejdůležitější vlastnost *back emf* je, že je lineárně závislé na rychlosti otáčení rotoru přes konstantu motoru  $k$ .

### Měření back emf

DC motorek je ovládán pomocí PWM přes H-můstek, tudíž piny motoru jsou buď připojovány na napájecí napětí, nebo na zem. Použitý H-můstek podporuje i třetí variantu, a to stav vysoké impedance, který zajistí, že motor je od H-můstku odpojen, tj. všechny výstupní tranzistory H-můstku jsou rozepnuty. Ve stavu vysoké impedance lze měřit *back emf*.

Průběh měření je následující. H-můstek se přepne do stavu vysoké impedance. Dále se musí počkat, než zanikne zbytkový proud v cínce a odezní přechodový



děj napětí na motoru. Ustálená hodnota napětí na motoru je *back emf*, jak jej lze vidět na Obr. 11. Následně přepneme H-můstek zpět. *Back emf* se měří v měřicím oknu dlouhém  $375 \mu\text{s}$  s periodou 2 ms. Šířka tohoto okna je určena experimentálně tak, aby se předešlo komplikacím s jevy popsány dále v textu.

Minimální délka měřicího okna je dána dobou ustálení přechodového děje a dobou zániku zbytkového proudu v cívce motoru. Minimální délka periody je dána délkou měřicího okna, ale musí být dostatečně dlouhá, aby se zbytečně nezkracoval a neovlivňoval čas na napájení motoru. Současně musí být perioda kratší, než je perioda regulátoru rychlosti. Jelikož se doba zániku proudu a doba přechodového děje mění s rychlostí otáčení, respektive s velikostí buzení, tak byla provedena měření (Obr. 11 a Obr. 12.), podle kterých byla určena šířka měřicího okna. Kritérium bylo, aby se za všech možných okolností stihlo napětí *back emf* v tomto časovém okně ustálit. Průběh na Obr. 11 potvrzuje, že napětí po ustálení je závislé na množství energie, které do motoru dodáváme.

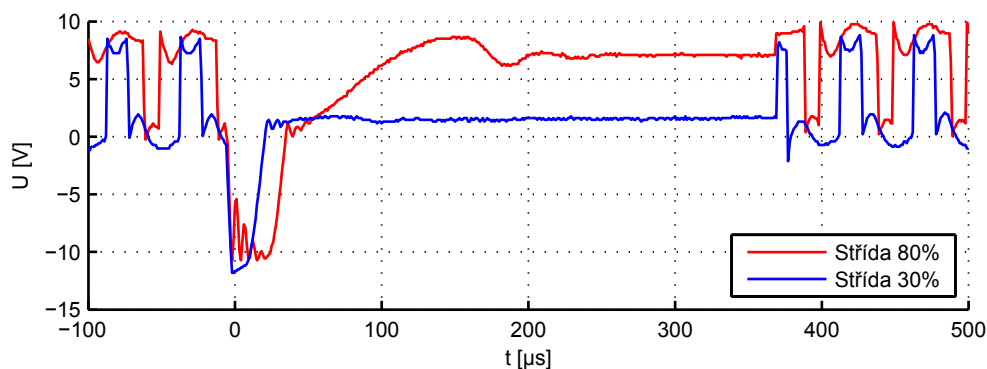
Na Obr. 12 je srovnání odezev v případech, kdy motor je a není mechanicky zabrzděn. Lze vidět, že při mechanickém zabrzdění je samotný přechodový děj posunut oproti nebrzděnému průběhu. Abychom zjistili, co se v motoru děje, tak se musíme podívat, jaký je průběh proudu na motoru před odpojením a po odpojení H-můstku. Na Obr. 13 jsou vyznačeny dvě cesty, kudy teče proud. Červený průběh vyznačuje cestu, kdy jsou otevřeny tranzistory Q1 a Q4, v případě, kdy je motor standardně řízen přes H-můstek; levá svorka motoru je připojena přes Q1 na napájení, pravá svorka motoru je připojena přes Q4 na zem. V situaci, kdy potřebujeme měřit *back emf* musíme všechny tranzistory zavřít. Nicméně motor obsahuje cívku, která má proudovou setrvačnost a musí vybit naakumulovanou energii. K tomuto účelu slouží ochranné diody D1 až D4. Vzhledem k protékajícímu proudu jsou diody D2 a D3 v propustném směru a proud přes ně může odtéci. Tím, že jsou tyto diody otevřené, tak vznikne situace, že vlevo je na motor připojena zem a vpravo napájecí napětí, čímž se směr napětí na motoru změnil oproti původnímu směru. Tento stav trvá do té doby, než hodnota proudu klesne na nulu a diody se opět zavřou. Čím větší proud motorem protéká, tím déle trvá jeho zánik po odpojení od H-můstku.

Z Obr. 12 si lze také všimnout, že přesto, že je motor mechanicky zastaven, tak se zelený průběh neustálil na hodnotě 0 V. Momentálně bohužel není zcela jasné, proč se tento jev děje. Měření ukázala, že v absolutní hodnotě je toto napětí větší čím větším napětím je motor buzen, a tedy čím větší proud motorem procházel. Tento jev je kompenzován tak, že pokud IRC senzor ukazuje nulovou rychlost, tak předpokládáme i nulové *back emf*.

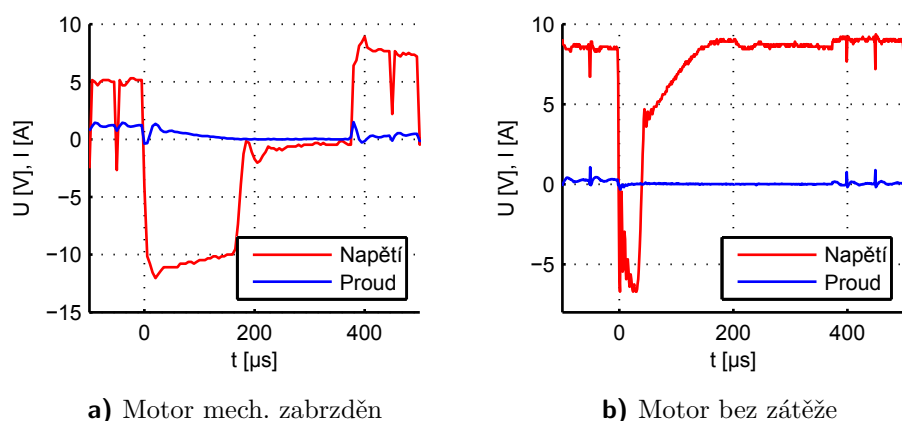
*Back emf* se měří diferenčně měřením napětí na obou svorkách zvlášť. Následně se tyto dvě napětí od sebe odečtou. Schéma měřicího obvodu je ukázáno na Obr. 14. K tomuto obvodu si sestavíme obvodové rovnice pro ustálený stav, které budou vypadat následovně

$$\frac{V_{cc} - u_A}{R_9} + \frac{u_B - u_A + u_{bemf}}{R_4 + R_7} = \frac{u_A}{R_8}, \quad (30)$$

### 3. Měření rychlosti



Obrázek 11. Průběh napětí na motoru při měření back emf, motor bez zátěže



a) Motor mech. zabrzděn

b) Motor bez zátěže

Obrázek 12. Průběh napětí a proudu na motoru při měření back emf

$$\frac{V_{cc} - u_B}{R_6} - \frac{u_B - u_A + u_{bemf}}{R_4 + R_7} = \frac{u_B}{R_5} \quad (31)$$

Dále pokud odečteme druhou rovnici od první, a po dosazení hodnot za odpory

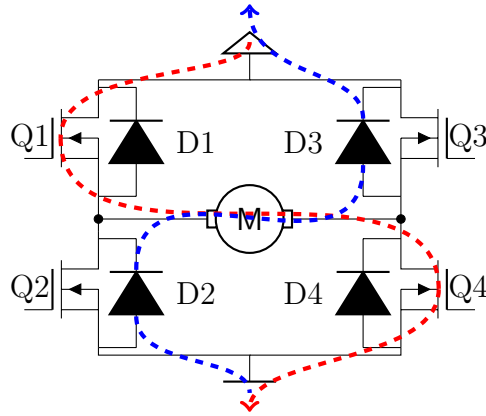
$$R_9 = R_7 = R_4 = R_6 = 10 \text{ k}\Omega, R_8 = R_5 = 1 \text{ k}\Omega, \quad (32)$$

si vyjádříme hodnotu  $u_{bemf}$ , tak dostaneme výsledek

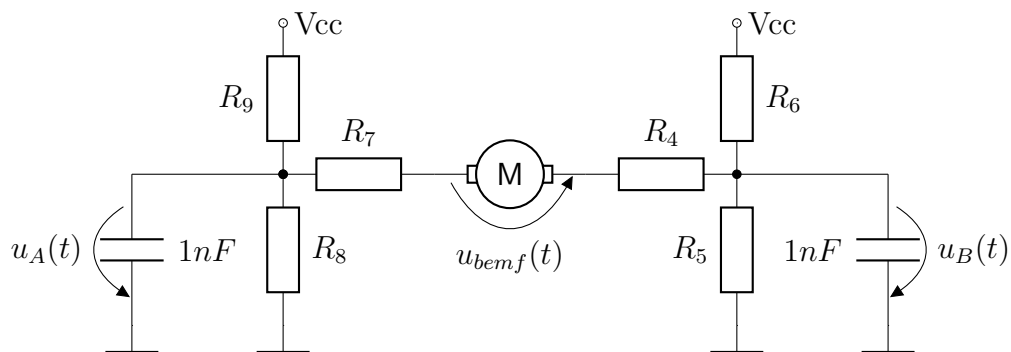
$$u_{bemf} = 12(u_A - u_B) \quad (33)$$

#### Výpočet rychlosti z hodnoty back emf

Z modelu na Obr. 4 lze vidět, že výstupní napětí je součet napětí generované gyrátorem G, napětí na cívce a napětí na rezistoru. Měření provádíme při nulovém budícím napětí tj.  $u = 0 \text{ V}$  a také až po odeznění přechodového děje, což znamená že obvod je v ustáleném stavu a úbytek napětí na cívce je nulový. Dále uvažujeme, že vstup A/D převodníku má nekonečně velký odpor, tudíž v obvodu neteče žádný proud, a proto i úbytek na rezistorech je nulový. Z toho plyne, že napětí *back emf*



Obrázek 13. Schématické připojení H-můstku k motoru a vyznačení cest, kudy teče proud



Obrázek 14. Měřící obvod back emf

je dán jako

$$u_{bemf} = \frac{k}{rn} v . \quad (34)$$

Úpravou předchozího vzorce získáme rychlost

$$v = \frac{rn}{k} u_{bemf} . \quad (35)$$

### 3.3. Filtrace a fúze senzorů pomocí Kalmanova filtru

Kalmanův filtr (KF) [4] je obecně velice známá věc a hojně využívána v řídicí technice, proto se zde nebudu zabývat jeho odvozením. Zaměřím se na jeho aplikaci pro filtraci, fúzi dvou senzorů a pro odhadování tření autíčka. Pro získání informace o rychlosti do KF použijeme IRC senzor a BEMF (měření rychlosti z *back emf*). Pomocí KF provedeme filtraci a fúzi těchto dvou měření. Oba způsoby měření rychlosti jsou zatíženy šumem, který pro jednoduchost budeme uvažovat jako bílý. Tento silný předpoklad se ukázal jako ospravedlnitelný, protože filtrace funguje velice dobře. Implementace KF do procesoru spočívá v implementaci jeho

### 3. Měření rychlosti

rovnice do programového kódu. Pro úplnost zde uvedu základní předpoklady a rovnice KF. Následně se již zaměřím na konkrétní implementaci.

#### 3.3.1. Obecné rovnice Kalmanova filtru

Kalmanův filtr uvažuje lineární systém ve tvaru

$$\mathbf{x}_k = \mathbf{A}\mathbf{x}_{k-1} + \mathbf{B}\mathbf{u}_{k-1} + \mathbf{w}_{k-1} , \quad (36)$$

$$\mathbf{y}_k = \mathbf{C}\mathbf{x}_k + \mathbf{v}_k , \quad (37)$$

kde  $\mathbf{w}$  je procesní šum a  $\mathbf{v}$  je šum měření.

Dále se uvažuje, že  $\mathbf{w}$  a  $\mathbf{v}$  jsou bílé šумы s normálním rozdělením pravděpodobnosti, tedy že

$$p(\mathbf{w}) \sim N(0, \mathbf{Q}) , \quad (38)$$

$$p(\mathbf{v}) \sim N(0, \mathbf{R}) . \quad (39)$$

Diskrétní podoba kalmanova filtru je následovná. Algoritmus je rozdělen na dva kroky predikční a korekční. Predikční krok obsahuje dvě rovnice,

$$\hat{\mathbf{x}}_k^- = \mathbf{A}\hat{\mathbf{x}}_{k-1} + \mathbf{B}\mathbf{u}_{k-1} , \quad (40)$$

$$\mathbf{P}_k^- = \mathbf{A}\mathbf{P}_{k-1}\mathbf{A}^T + \mathbf{Q} . \quad (41)$$

Korekční krok je rozdělen do třech rovnic následovně

$$\mathbf{K}_k = \mathbf{P}_k^- \mathbf{C}^T (\mathbf{C}\mathbf{P}_k^- \mathbf{C}^T + \mathbf{R})^{-1} , \quad (42)$$

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + \mathbf{K}_k (\mathbf{y}_k - \mathbf{C}\hat{\mathbf{x}}_k^-) , \quad (43)$$

$$\mathbf{P}_k = (\mathbf{I} - \mathbf{K}_k \mathbf{C}) \mathbf{P}_k^- . \quad (44)$$

Samotný běh filtru spočívá v tom, že se nastaví počáteční podmínky, z kterých se vstoupí do predikčního kroku a následně se střídají už jen predikční a korekční krok.

#### 3.3.2. Implementace Kalmanova filtru

Na implementaci máme dva požadavky, aby provedl fúzi dvou senzorů i s jejich filtrací, a také aby byl schopen odhadovat tření. V sekci, kde se zabýváme modelováním autíčka, jsme se omezili pouze na lineární případ tření, které se dá uvažovat až u vyšších rychlostí, kde převládá viskózní tření nad statickým. Na nízkých rychlostech tomu tak není a pro KF je potřeba znát model systému, tudíž bychom museli tření změřit. Bohužel statické tření se na různých místech dráhy

lišší, taktéž každé autíčko má jiné vnitřní tření. Abychom nemuseli pro každý případ zvlášť měřit tření, tak můžeme využít KF vedle odhadu rychlosti i pro odhad tření. KF na základě modelu a měření odhaduje stavy systému. Budeme vycházet z modelu (12), kde je třecí síla zavedena jako funkce závislá na rychlosti. Místo této funkce zavedeme tření jako stavovou proměnnou, která nebude mít žádnou dynamiku.

Diferenciální rovnice systému budou vypadat takto

$$\frac{dv_{k-1}}{dt} = \frac{k}{Rrnm} (U_{nap} D_{k-1} - \frac{kv_{k-1}}{rn}) - \frac{F_{tk-1}}{m}, \quad (45)$$

$$\frac{dF_{tk-1}}{dt} = 0. \quad (46)$$

Jedním z kroků KF je odhad nového stavu na základě modelu. Odhad provedeme použitím dopředné Eulerovy metody pro výpočet diferenciálních rovnic následujícím způsobem

$$\begin{bmatrix} v_k \\ b_k \end{bmatrix} = \begin{bmatrix} v_{k-1} \\ b_{tk-1} \end{bmatrix} + \begin{bmatrix} \frac{dv_{k-1}}{dt}(v_{k-1}, F_{tk-1}, u_{k-1}) \\ \frac{dF_{tk-1}}{dt}(v_{k-1}, F_{tk-1}, u_{k-1}) \end{bmatrix} h, \quad (47)$$

kde  $h$  je perioda, na které počítáme filtr. Aktuální nastavení periody je 2 ms. Pro tento systém si vyjádříme matici

$$\mathbf{A} = \begin{bmatrix} -\frac{k^2}{Rmn^2r^2} & -\frac{1}{m} \\ 0 & 0 \end{bmatrix}. \quad (48)$$

Do rovnic KF potřebujeme diskrétní podobu matice  $\mathbf{A}$ . Zde je problém, protože systém obsahuje neřiditelný a nepozorovatelný stav. Z tohoto důvodu nelze použít Tustinovu metodu, protože při tvorbě přenosu nám zmizí stav tření. Proto pro diskretizaci použijeme dopřednou Eulerovu metodu, které nám zachová všechny stavy. Podobně jako v předchozím případě vypočteme nové hodnoty stavů podle vztahu

$$\mathbf{x}_k = \mathbf{x}_{k-1} + \mathbf{A}\mathbf{x}_{k-1}h = (\mathbf{I} + \mathbf{A}h)\mathbf{x}_{k-1}, \quad (49)$$

kde  $\mathbf{I}$  je jednotková matice.

Diskretizovaná matice má tvar

$$\mathbf{A}_d = (\mathbf{I} + \mathbf{A}h) = \begin{bmatrix} 1 - \frac{k^2h}{Rmn^2r^2} & -\frac{h}{m} \\ 0 & 1 \end{bmatrix}. \quad (50)$$

Fúzi senzorů provedeme tak, že matici  $\mathbf{C}$  rozšíříme o nový řádek. Výsledná matice  $\mathbf{C}$  má následující tvar

$$\mathbf{C} = \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix}. \quad (51)$$

### 3. Měření rychlosti

Pro správnou funkci KF, je potřeba vhodně zvolit hodnoty matic  $\mathbf{Q}$  a  $\mathbf{R}$ . Jejich hodnoty jsem zjistil na základě následujícího experimentu. Regulátoru rychlosti nastavujeme referenční rychlost od 350 do 2000 po 50  $\text{mms}^{-1}$ . Pro každou referenční rychlost byla změřena střední hodnota výstupní rychlosti a rozptyl obou metod měření. Výsledná naměřená data jsou na Obr. 16. Vidíme, že závislost rozptylu senzorů na rychlosti nejsou konstantní, ani lineární. Oba naměřené průběhy byly aproximovány polynomem druhého řádu pomocí nástroje Basic fitting v Matlabu. Pro BEMF a IRC senzor vypadají aproximační funkce následovně

$$\sigma_{IRC}^2(v_k) = 0.0062v_k^2 + 0.095v_k + 390 , \quad (52)$$

$$\sigma_{BEMF}^2(v_k) = 0.034v_k^2 - 21v_k + 5900 . \quad (53)$$

Jelikož rozptyly jsou rychlostně závislé, tak hodnoty matice  $R$  přepočítáváme v závislosti na odhadované rychlosti. Přestože změna kovariančních matic v závislosti na odhadovaném stavu se může zdát jako nestandardní krok, tak experimenty ukazují, že to lze použít s dobrými výsledky.

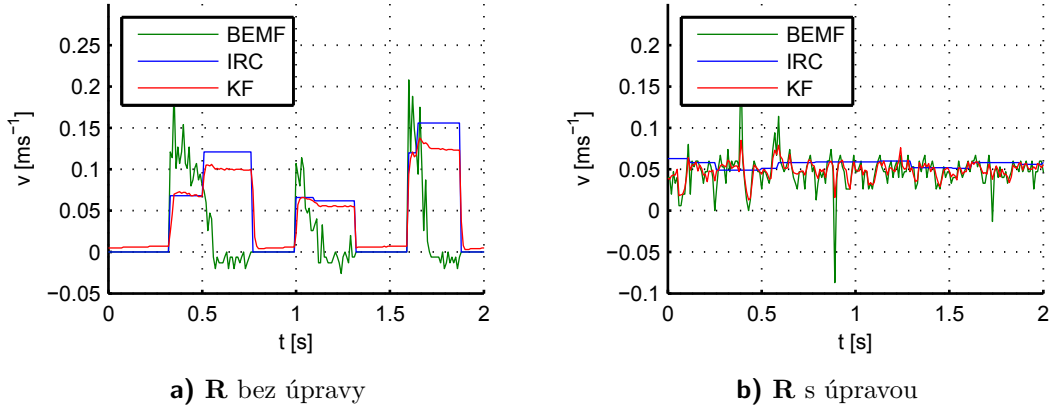
Abychom mohli rovnice (52) a (53) použít v matici  $\mathbf{R}$ , tak je musíme ještě dále upravit.

První úpravou je, že budeme rychlost uvažovat v absolutní hodnotě, aby pro zápornou rychlost byla určena hodnota rozptylu stejná jako pro kladnou. Je to pouze předpoklad, není ho možné zatím ověřit, protože konstrukční charakter autíček nedovoluje dlouhodobější jízdu vzad.

Druhou úpravou je změna hodnot rozptylů na malých rychlostech. Pro oba senzory zvolíme od hraniční rychlosti dolů konstantní hodnotu rozptylu; pro BEMF je hraniční hodnota 330  $\text{mms}^{-1}$  a konst. rozptyl 300, pro IRC je hraniční hodnota 800  $\text{mms}^{-1}$  a konst. rozptyl 4434. Nastavená hodnota konst. rozptylu na malých rychlostech pro BEMF není naměřená, ale byla zvolena, protože chceme, aby se na malých rychlostech odhadovala rychlost více podle měření BEMF, než podle měření IRC senzoru. Obr. 15 ukazuje stejnou situaci s aplikovanou úpravou a bez ní při regulaci rychlosti. Z průběhů vidíme, že pokud úpravu neaplikujeme, tak regulátor není schopen správně regulovat, protože IRC senzor při malých rychlostech není schopen poskytovat aktuální informaci o rychlosti dostatečně často. V takovéto situaci je doba přechodu mezi barvami násobně větší než je perioda měření. Informace o rychlosti se aktualizuje vždy při přechodu z jedné barvy na druhou, tudíž několik period měření nemáme k dispozici aktuální hodnotu rychlosti od IRC senzoru. Při zachování poměrů rozptylů, které byly naměřeny, KF odhaduje rychlost více podle IRC senzoru a tudíž dochází k mylnému odhadu. Naopak, když úpravu matice  $\mathbf{R}$  provedeme, tak se na malých rychlostech odhaduje rychlost více podle BEMF, které poskytuje každou periodu měření vždy aktuální informaci. Tento problém také nastane při skokových změnách rychlosti z vyšších na nižší. Toto řešení přispívá k regulaci na nízkých hodnotách rychlosti.

Po aplikaci těchto dvou úprav budou rozptyly vypadat následovně

$$\sigma_{IRC}^2(v_k) = \begin{array}{ll} 4434 & \text{pro } |v_k| < 800 \\ 0.0062|v_k|^2 + 0.095|v_k| + 390 & \text{pro } |v_k| \geq 800 \end{array} , \quad (54)$$



Obrázek 15. Ukázka odhadu rychlosti pomocí KF v situaci, kdy se provede úprava  $\mathbf{R}$  pro malé rychlosti a kdy ne. Regulace na rychlost  $0.05 \text{ ms}^{-1}$

$$\sigma_{BEMF}^2(v_k) = \begin{cases} 300 & \text{pro } |v_k| < 330 \\ 0.034|v_k|^2 - 21|v_k| + 5900 & \text{pro } |v_k| \geq 330 \end{cases} \quad (55)$$

Výsledná matice  $R$  je diagonální matice následujícího tvaru

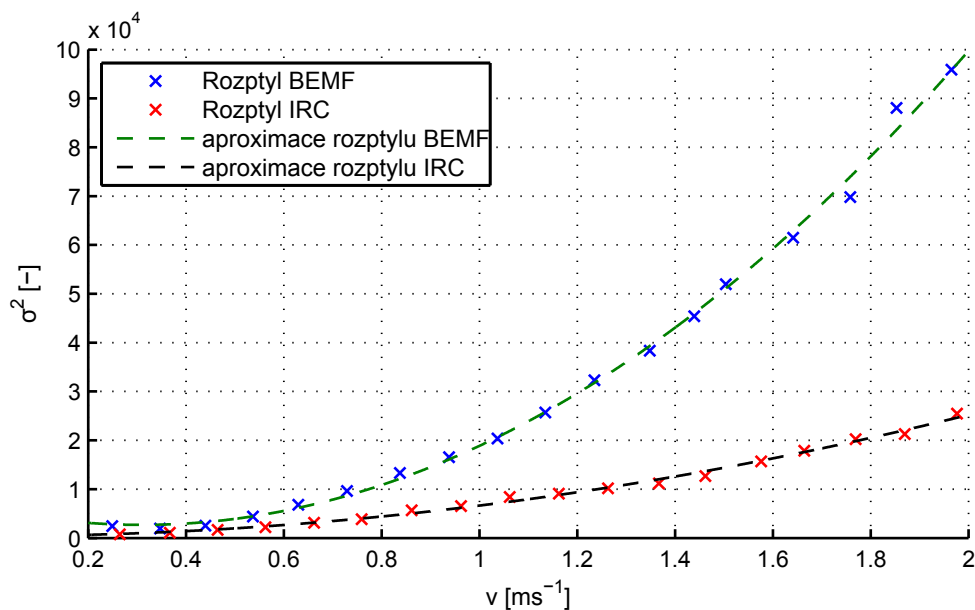
$$\mathbf{R} = \begin{bmatrix} \sigma_{IRC}^2(v_k) & 0 \\ 0 & \sigma_{BEMF}^2(v_k) \end{bmatrix} \quad (56)$$

Poslední matice, kterou musíme navolit je kovarianční matice  $\mathbf{Q}$ . Toto je matice s hodnotami rozptylů procesního šumu. Tyto hodnoty byly zvoleny experimentálně. Maticemi  $\mathbf{Q}$  a  $\mathbf{R}$  nastavujeme, zda budeme věřit více měření, nebo modelu. Nastavením nízkých hodnot matice  $\mathbf{Q}$  dosáhneme toho, že bude signál hodně filtrován, čímž se sníží i rychlost reakce na změnu tření. Požadavek na nastavení matice  $\mathbf{Q}$  je takový, že se musí zachovat míra filtrace, ale zároveň zajistit dostatečně rychlá reakce na změnu tření. Výsledná matice  $\mathbf{Q}$  byla určena následovně

$$\mathbf{Q} = 0.005^2 \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (57)$$

Se znalostí všech těchto hodnot už zbývá přepsat rovnice KF do programovacího jazyka C. Problémem implementace KF je maticové násobení, které je náročné na výpočetní čas. V zájmu zachování rychlosti vykonávání kódu bylo násobení matic a počítání inverze napsáno tak, že se napevno definovalo v kódu násobení a sčítání jednotlivých prvků matic. Pro tento účel lze použít Symbolický toolbox v Matlabu, kde si můžeme spočítat násobení matic v symbolických proměnných, kterými budou označeny prvky matic např.  $Q_{11}$ , což označuje prvek matice  $Q$  v prvním sloupci prvního řádku. Výsledek násobení je potřeba upravit pro programovací jazyk. K tomu lze použít jakýkoliv textový editor, který umí příkaz *vyhledat a nahradit*, pomocí kterého nahradíme symbolické proměnné za prvky polí, které máme nadefinovány v kódu, například  $Q[0][0]$  nahradíme za  $Q_{11}$ . V poslední řadě najdeme v rovnicích opakující se dílčí výpočty, které vy-

### 3. Měření rychlosti



Obrázek 16. Hodnoty rozptylů měření jednotlivých senzorů v závislosti na rychlosti

jmeme, provedeme je zvlášť a do výpočtu dosadíme proměnnou, ve které je uložen výsledek.

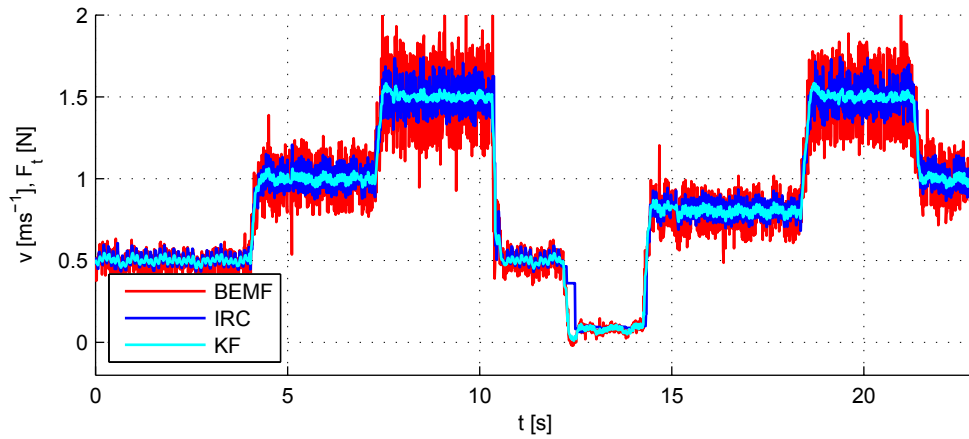
Další možné řešení bylo pomocí algoritmů pro násobení matic a pro inverzi, které jsou řešeny pomocí skokových instrukcí procesoru. Tímto postupem bychom ušetřili paměť programu, ale určitě neušetříme procesorový čas, který je spotřebován na obsluhu skoků.

#### 3.3.3. Ukázka funkčnosti implementovaného Kalmanova filtru

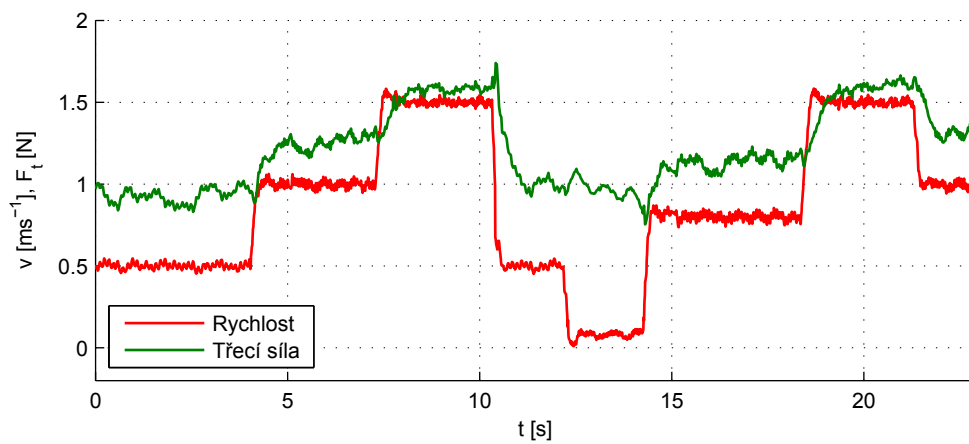
Na Obr. 17 je zobrazen odhad rychlosti na základě údajů od obou senzorů rychlosti. Při tomto pokusu bylo autíčko řízeno regulátorem rychlosti. Lze vidět, jak s rostoucí rychlostí, roste rozptyl šumu senzorů. Mezi časy 10 a 15 sekund lze vidět, že rychlost poklesla na nízkou hodnotu, kdy regulátor autíčko na okamžik téměř zastavil. Zde je názorně vidět problém s měřením rychlosti pomocí IRC senzoru, který není schopen včas zaregistrovat velké změny rychlosti směrem dolů, nicméně měření BEMF je schopno tento skok pojmout, proto se na nízkých rychlostech odhaduje rychlost zejména z něho. Na vysokých rychlostech naopak má BMEF vysoký rozptyl šumu oproti IRC senzoru, proto při odhadu více věříme hodnotě z IRC senzoru.

Na Obr. 18 je zobrazena odhadovaná třecí síla v závislosti na měnící se rychlosti. Lze si všimnout, že při rychlostech vyšších než  $0.5\text{ms}^{-1}$  jsou skoky úměrné skokům rychlosti, ale na nižších rychlostech již nejsou tyto změny tolik patrné. Je to dáno nelinearitou třecí síly na malých rychlostech.





Obrázek 17. Odhadovaná rychlost pomocí Kalmanova filtru



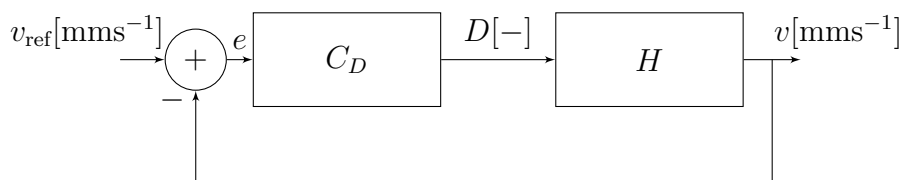
Obrázek 18. Odhadovaná třecí síla v závislosti na rychlosti pomocí Kalmanova filtru

## 4. Regulace - rychlostní regulátor

Pro řízení rychlosti byl implementován disktrétní PI regulátor v mikroprocesoru na desce, který je společně se systémem zapojen ve zpětnovazební smyčce zobrazené na Obr. 19. Perioda regulace rychlosti je

$$T_s = 5 \text{ ms} . \quad (58)$$

PI regulátor byl zvolen pro jednoduchost návrhu a bezproblémové implementaci v procesorovém kódu. Vstupem regulační smyčky je referenční rychlost. Vstupem regulátoru je odchylka měření rychlosti od referenční rychlosti. Výstup systému je rychlost, která je odhadnuta Kalmanovým filtrem. Výstupem regulátoru je velikost střídý PWM signálu, který budí motor.



Obrázek 19. Zapojení regulátoru ve zpětnovazební smyčce

### 4.1. Návrh regulátoru

PI regulátor byl navržen ve spojitě oblasti ve tvaru

$$C = k_p + \frac{k_i}{s} . \quad (59)$$

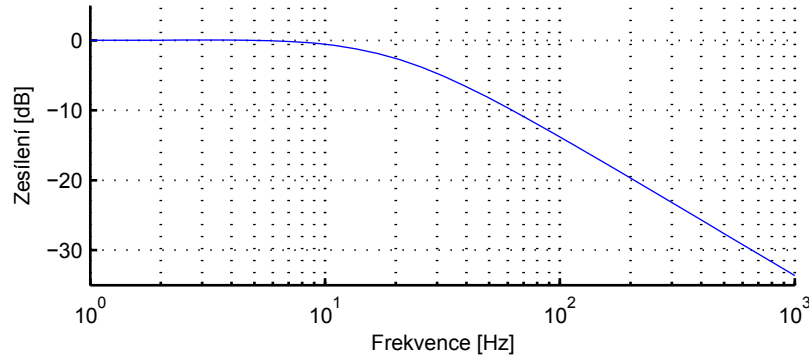
Následně byl diskretizován metodou Zero-Order Hold do tvaru

$$C_D = k_p + \frac{k_i T_s}{z - 1} . \quad (60)$$

Samotné parametry regulátoru byly navrženy následovně

$$k_p = 0.002 , k_i = 0.01 . \quad (61)$$

Parametry byly nastaveny za pomoci nástroje Rltool v Matlabu. Cílem nastavení bylo, aby nedocházelo k překmitu při malých změnách referenční rychlosti, ale aby byly doby náběhu a ustálení dostatečně rychlé, protože v kaskádě bude



Obrázek 20. Frekvenční charakteristika pro uzavřenou rychlostní smyčku

před rychlostním regulátorem implementován vzdálenostní regulátor, tudíž požadavkem je, aby byla rychlost regulace co nejvyšší.

Jelikož střída může nabývat pouze hodnot od 0 do 1 (pro couvání od 0 do -1), tak byla do regulátoru přidána saturace výstupu od -1 do 1, taktéž byl saturován rozsah integrátoru od -1 do 1, čímž předejdeme problému, že by integrátor mohl integrovat nad rozsah střidy a došlo by k windup efektu. Tento zásah má za následek, že při velkých změnách se akční zásah dostane do saturace a dojde k překmitu.

Přenos uzavřené rychlostní smyčky je

$$T(s) = \frac{20.8s^3 + 186.4s^2 + 411.8s}{s^4 + 28.72s^3 + 202s^2 + 411.8s} = \frac{20.8s(s + 5)}{s(s + 19.4)(s + 5.361)} \quad (62)$$

Frekvenční charakteristika pro uzavřenou smyčku je zobrazena na Obr. 20

## 4.2. Ukázky regulace

Porovnání simulace a reality je zobrazeno na Obr. 21 a Obr. 22.

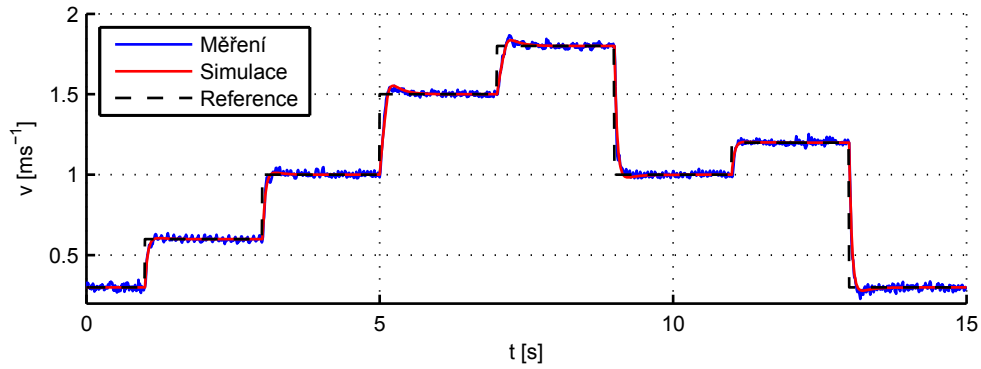
Obr. 23 ukazuje odezvu na skok v případech, kdy akční zásah dosáhl saturace, a kdy nedosáhl. Lze vidět, že když dosáhl saturace, tak došlo k překmitu a prodloužila se doba ustálení. Na levém obrázku lze naopak vidět, že při malých změnách k překmitu téměř nedochází.

Dále byl proveden pokus (Obr. 24), kdy se dvěma autíčkům nastavila stejná referenční rychlost a měřila se vzdálenost mezi nimi. Z průběhu lze vidět, že za 30 s jízdy vzdálenost vzrostla o 5 cm, což odpovídá rychlosti  $1.7 \text{ mms}^{-1}$ . Tato chyba je vzhledem k povaze autíček přijatelná.

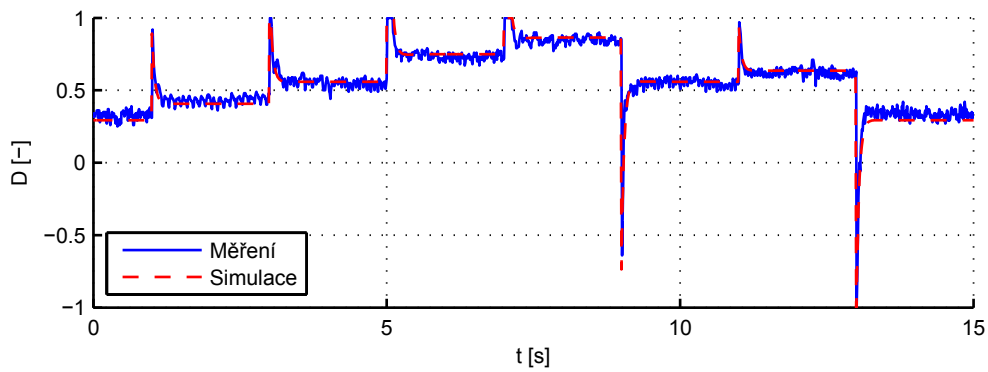
Na Obr. 25 je pro porovnání ukázán průběh regulace s více autíčky, kdy je vidět, že regulací lze chování autíček sjednotit.

Všechny dosavadní průběhy byly pořízeny při testech na kruhové dráze, kde lze očekávat, že se tření nebude tolik lišit. Proto byl proveden test (Obr. 26), kdy byl zvolen složitější tvar, než je kruh. Při jízdě po kruhu lze vidět, že se hodnota odhadovaného tření příliš nemění oproti druhému případu, proto ani regulace nekolísá. V druhém případě při jízdě po nekruhové dráze je vidět, že se tření

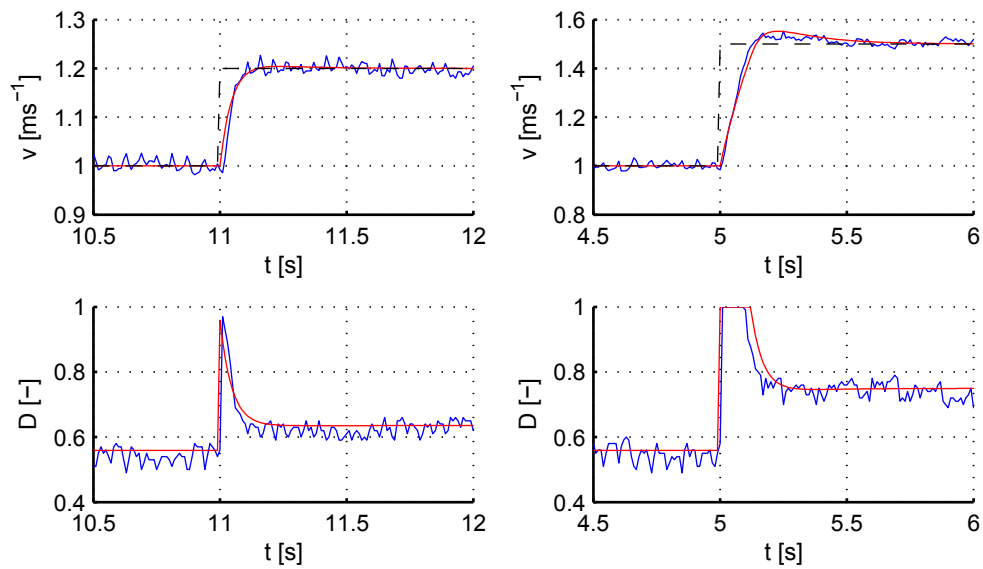
#### 4. Regulace - rychlostní regulátor



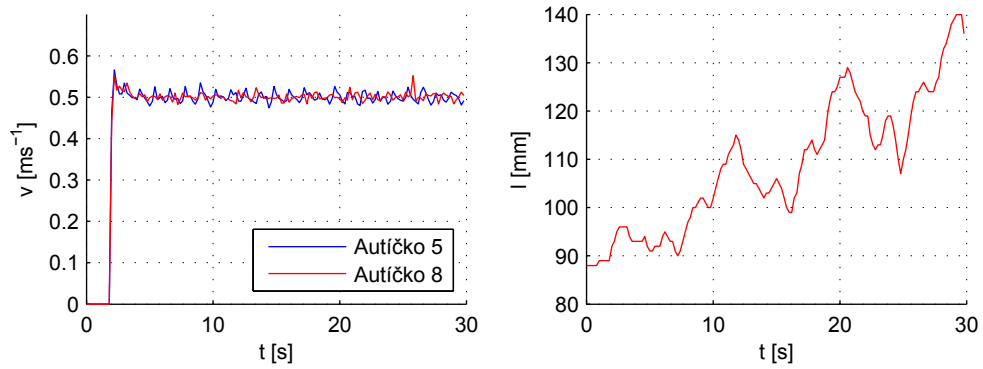
Obrázek 21. Porovnání regulátoru v simulaci a v reálné implementaci - rychlost



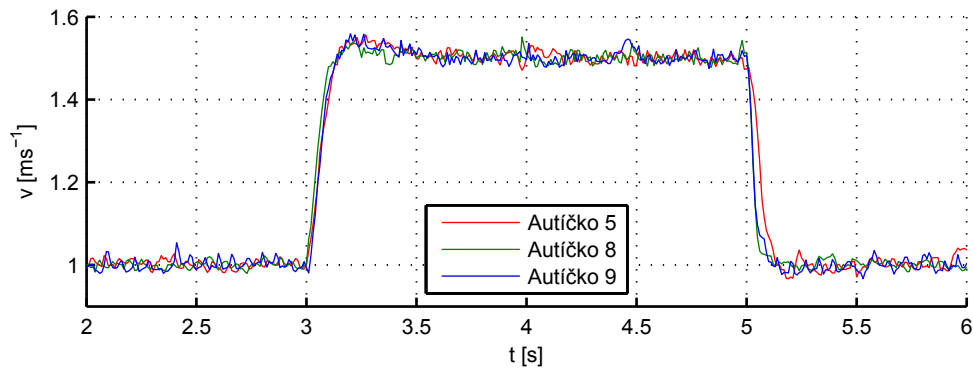
Obrázek 22. Porovnání regulátoru v simulaci a v reálné implementaci - akční zásah



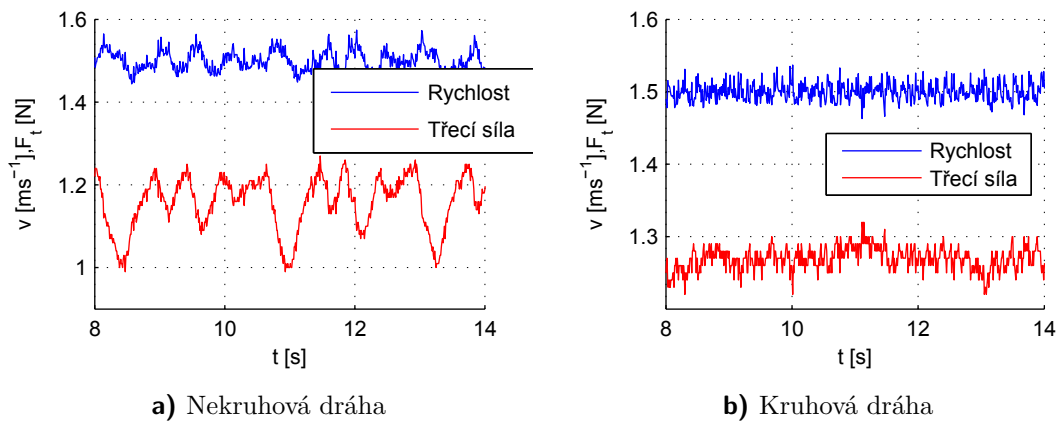
Obrázek 23. Porovnání regulátoru (modrý) se simulací (červený) pro různé skokové změny referenční rychlosti



Obrázek 24. Měření vzdálenosti mezi dvěma autíčky při stejné referenční rychlosti



Obrázek 25. Porovnání regulátor v simulaci a v reálné implementaci - rychlost



Obrázek 26. Regulace rychlosti a odhad tření na různých drahách

měnilo až dvakrát tolik. Lze si také všimnout periodického opakování v průběhu třecí síly, což je také perioda použité dráhy. Změna tření s dráhou vstupuje do soustavy jako poruchová veličina.

## 5. Závěr

Prvním úkolem práce bylo vytvořit matematický model autodráhového autíčka. Ukázalo se, že tření hraje v systému významnou roli a navíc je u každého autíčka viditelně jiné. K odhadu tření byl využit Kalmanův filtr. Z měření vyplynulo, že tření nemá předpokládaný lineární charakter, nicméně při vyšších rychlostech, pro které byl model vytvořen, již lze považovat tření za lineární. Model by se dal zpřesnit i pro malé rychlosti aproximací tření složitější, nebo po částech lineární funkcí.

Druhým úkolem bylo navrhnout a implementovat měření rychlosti. Byly navrženy a implementovány dvě metody, a to měření pomocí IRC senzoru a výpočet rychlosti z *back emf*. Obě metody mají své pro a proti. Výhoda IRC senzoru je, že má menší rozptyl při měření konstantních otáček, než má *back emf*. Nevýhodami jsou složitější konstrukční provedení a frekvence aktualizace informace o rychlosti je závislá právě na rychlosti otáčení, což způsobuje problémy při měření změn na pomalých rychlostech. Výhoda měření pomocí *back emf* je snadné konstrukční provedení, vždy možnost získání aktuální informace o rychlosti a snadný přepočet *back emf* na rychlost. Nevýhodou této metody je, že měření je zatíženo šumem, který roste s rychlostí, a také, že se při měření *back emf* musí motor odpojit od napájení. K určení výsledné rychlosti byl navržen a implementován Kalmanův filtr, který slouží pro filtraci i fúzi obou metod. Byla navržena metoda, jak se pomocí Kalmanova filtru vypořádat s nevýhodami obou měření. Do budoucna by se dalo měření rychlosti rozšířit o použití akcelerometru a gyroskopu k získání informace o pohybu autíčka jako celku. Při filtraci Kalmanovým filtrem by bylo možné použít měření proudu.

Posledním úkolem bylo řízení rychlosti. Byl navržen a implementován diskrétní PI regulátor. Přestože byl regulátor navržen na lineární model, který platí až od vyšších rychlostí, tak si velice dobře poradí i s nízkými. Je schopen regulovat rychlost již od  $50 \text{ mms}^{-1}$ . Byla také testována schopnost regulovat rychlost na složitějším tvaru dráhy, kdy se mění tření s pozicí. V tomto testu regulace obstála také dobře. Regulátor byl použit v kaskádě se vzdálenostním regulátorem. Výsledné testy, které prováděl Jan Moravec, ukázaly, že zapojení v kaskádě fungovalo a bylo možné na něm předvést základní metody řízení kolony vozidel.

# Příloha A.

## Obsah příloženého CD

Na CD je přiložen text práce a celý projekt obsahující veškeré zdrojové kódy.

### A.1. Obsah složky `Platoon_implementation`

#### `/onboard/hw`

Schéma a zapojení desek plošných spojů

#### `/onboard/fw/app`

Hlavní složka projektu, umístění *main.c* a *Makefile* souborů.

#### `/onboard/fw/common`

Nezbytné knihovny pro běh programu, fungování procesoru a komunikace.

#### `/onboard/fw/platoon_lib/measurements`

Knihovny pro měření vzdálenosti a rychlosti.

#### `/onboard/fw/platoon_lib/regulators`

Vzdálenostní a rychlostní regulátory.

#### `/Regulator rychlosti`

Matlabovský model rychlostního regulátoru.

#### `/sw`

Pomocné programy a knihovny pro komunikaci a nahrávání programu do autíčka.

#### `/sw/simulink_sfunction`

Matlabovský model pro komunikaci s autíčky (před spuštěním je nutné zavolat script *nastaveni.m*).

### A.2. Práce s projektem přiloženým na CD

Pomocí příkazové řádky se přesuneme do složky `/onboard/fw/app`.

- Překlad projektu se provádí pomocí *Makefile* zavoláním příkazu `make`.
- Nahrávání programu do autíčka se provádí tak, že přes příkaz `set addr=X`, kde X je číslo auta, vybere příslušné autíčko a zavoláním příkazu `load` se nahraje kód.

Pro překlad kódu je nezbytný kompilátor YAGARTO a WinAVR. Komunikace se provádí přes modul nRF24LU1, pro který je nutné mít nainstalovaný ovladač Libusb. Pro práci s komunikačním modelem v Matlabu jsou potřeba tyto toolboxy: Real-Time Windows Target toolbox, MATLAB Coder.

# Literatura

- [1] John Chiasson. *Modeling and High-Performance Control of Electric Machines*. IEEE Press Series on Power Engineering. Wiley-IEEE Press, 2005.
- [2] Dan Martinec. *Distributed control of platoons of racing slot cars*. 2012. URL: [https://support.dce.felk.cvut.cz/mediawiki/images/7/75/Dp\\_2012\\_martinec\\_dan.pdf](https://support.dce.felk.cvut.cz/mediawiki/images/7/75/Dp_2012_martinec_dan.pdf) (cit. 13.05.2014).
- [3] H. Olsson et al. *Friction Models and Friction Compensation*. 1997. URL: [http://www.gipsa-lab.grenoble-inp.fr/~carlos.canudas-de-wit/publications/friction/dynamic\\_friction\\_EJC\\_98.pdf](http://www.gipsa-lab.grenoble-inp.fr/~carlos.canudas-de-wit/publications/friction/dynamic_friction_EJC_98.pdf) (cit. 13.05.2014).
- [4] Greg Welch a Gary Bishop. *An Introduction to the Kalman Filter*. 2006. URL: [http://www.cs.unc.edu/~welch/media/pdf/kalman\\_intro.pdf](http://www.cs.unc.edu/~welch/media/pdf/kalman_intro.pdf) (cit. 13.05.2014).