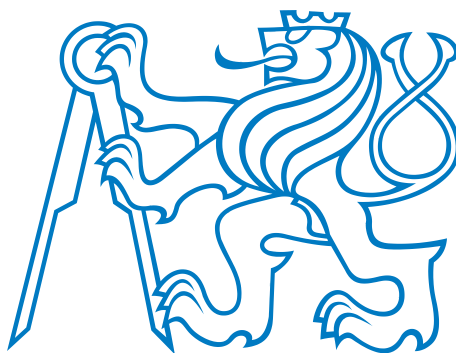


Bakalářská práce

Distribuované řízení kolon vozidel na autodráze

Jan Moravec



Květen 2014

Vedoucí práce: Ing. Ivo Herman

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

Poděkování

Chtěl bych tímto poděkovat zejména panu doktoru Zdeňku Hurákovi, který mne s projektem seznámil, Ivu Hermanovi a Danu Martincovi za jejich rady v teorii a Miroslavu Dvořákovi za pomoc v praxi.

Také bych rád vyjádřil díky rodičům a své přítelkyni Lence Hrabalové za jejich pomoc, podporu a lásku.

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 20. května 2014

.....
Jan Moravec

Abstrakt

Bakalářská práce je součástí projektu Slotcar Platooning Katedry řídicí techniky Elektrotechnické fakulty ČVUT v Praze, jehož cílem je sestavení a zprovoznění modelu kolony autodráhových vozidel pro testování distribuovaného řízení.

Práce se zabývá vytvořením vlastního senzoru měření vzdálenosti, sestavením kolony vozidel a praktickým vyzkoušením algoritmů pro její řízení.

Senzor vzdálenosti je sestaven z dvojice infračervené LED diody a odpovídajícího fototranzistoru. Signál z diody je nutné modulovat pro zlepšení jeho odolnosti proti rušení. Demodulovaný signál je následně přepočítán v závislosti na jeho amplitudě na vzdálenost.

Práce se dále zaměřuje na praktické sestavení kolony, navržení kaskádní regulace vzdálenosti mezi auty, vyzkoušení algoritmů dopředného, obousměrného a váženého obousměrného řízení a vytvoření základního modelu asi deseti vozidel.

Klíčová slova

distribuované řízení; kolona vozidel; implementace; měření vzdálenosti; IR LED; fototranzistor; modulace; demodulace

Abstrakt

The bachelor thesis is a part of Slotcar Platooning project by the Czech Technical University in Prague, Faculty of Electrical Engineering, Department of Control Engineering, whose goal is assembling and getting to work a platoon of slotcars for testing distributed control.

The thesis deals with creating own sensor for measuring distance, assembling slotcar platoon and testing algorithms for its control.

Distance sensor consists of a pair of infrared LED and appropriate fototransistor. The signal from diode is necessary to modulate for improving its resintace to interference. Demodulated signal is then converted to distance according to its amplitude.

Further, the thesis concentrates on practical assembling of a platoon, designing cascade regulation of distance between cars, trying out predecessor following, bidirectional control and asymmetric bidirectional control algorithms and creating basic model of about ten cars.

Keywords

distributed control; platoon; implementation; distance measuring; IR LED; phototransistor; modulation; demodulation

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Student: Jan Moravec
Studijní program: Kybernetika a robotika (bakalářský)
Obor: Robotika
Název tématu: Distribuované řízení kolon vozidel na autodráze

Pokyny pro vypracování:

1. Navrhněte a implementujte HW a SW pro měření vzdálenosti k následujícímu a předchozímu vozidlu.
2. Nastudujte a implementujte vybrané algoritmy pro řízení kolon vozidel. Provedte experimenty na nejméně 10 vozidlech.
3. Bude-li to možné, vyberte kameru vhodnou pro vozidlo. Následně zpracujte obraz pro detekci brzdového světla předchozího vozidla.

Seznam odborné literatury:

- [1] D. Martinec, I. Herman, Z. Hurák, and M. Šebek: Wave-absorbing vehicular platoon controller. arXiv Prepr. arXiv1311.2095, 2013.
- [2] P. Seiler, A. Pant, and K. Hedrick: Disturbance propagation in vehicle strings. Autom. Control. IEEE Trans.vol. 49, no. 10, pp. 1835-1841, 2004.
- [3] F. Tangerman, J. Veerman, and B. Stosic: Asymmetric decentralized flocks. Autom. Control. IEEE Trans., vol. 57,no. 11, pp. 2844-2853, 2012.

Vedoucí bakalářské práce: Ing. Ivo Herman

Platnost zadání: do konce letního semestru 2014/2015

L.S.

doc. Dr. Ing. Jan Kybic
vedoucí katedry

prof. Ing. Pavel Ripka, CSc.
děkan

V Praze dne 10. 1. 2014

Obsah

1. Úvod	1
1.1. Odkaz na přiložené soubory bakalářské práce	1
1.2. Motivace	1
1.3. Projekt Slotcar platooning	2
1.3.1. Lidé za projektem	2
1.3.2. Požadavky pro sestavení modelu kolony	2
1.3.3. Platforma	3
1.3.4. Hardware a senzory	4
1.3.5. Software a propojení s počítačem	5
Nahrání programu do aut	5
Komunikace s auty	5
Vytváření regulátorů a dalších součástí v programu Simulink	5
2. Měření vzdálenosti	6
2.1. Požadavky	6
2.2. Možnosti řešení	6
2.2.1. Komerční senzory	7
Optické senzory	7
2.3. Návrh vlastního optického senzoru	8
2.3.1. Výběr součástí	9
2.3.2. Navržené zapojení	9
2.3.3. Sestavení senzoru, výsledky a motivace pro modulaci	10
2.4. Modulace a demodulace signálu	11
2.5. Vysvětlení kódu modulace a demodulace	11
2.5.1. Modulace, demodulace a nastavení modulační frekvence	12
Modulace a demodulace v programovém kódu	12
Modulační frekvence	12
2.5.2. Ukázka modulace a demodulace v MATLABovském skriptu	13
2.5.3. Diskuze výsledku simulací	15
2.6. Praktická implementace řešení	15
2.6.1. Přepočet amplitudy signálu na vzdálenost	16
2.6.2. Filtrace vypočtené vzdálenosti	17
2.7. Výsledný senzor	18
2.7.1. Vlastnosti senzoru	18
2.7.2. Montáž senzoru	18
Úhlová závislost zabudovaného senzoru	19
2.7.3. Diskuze vlastností výsledného zabudovaného senzoru	19
3. Regulace a teorie řízení kolon vozidel	21
3.1. Regulace pohybu vozidel	21
3.1.1. Model auta	21
3.1.2. Rychlostní smyčka	22
3.1.3. Vzdálenostní smyčka	22
3.1.4. Výhody kaskádní regulace	22
3.1.5. Ladění a použité hodnoty kaskádních regulátorů	22
Rychlostní regulátor	23
Vzdálenostní regulátor	23

Analýza kaskádního regulátoru	24
3.2. Teorie řízení kolon	25
3.2.1. Dopředné řízení (<i>predecessor following</i>)	25
3.2.2. Řízení k následujícímu vozidlu (<i>follower control</i>)	25
3.2.3. Obousměrné řízení (<i>bidirectional control</i>)	26
3.2.4. Obousměrné vážené řízení (<i>asymmetric bidirectional control</i>)	26
4. Pokusy s kolonou vozidel	27
4.1. Škálování se skutečným světem a parametry měření	27
4.2. Experimenty a pokusná měření	28
4.2.1. Reakce kolony na skok	30
Dopředné řízení	30
Obousměrné vážené řízení s váhou 0,5	32
Obousměrné řízení	34
4.2.2. Reakce kolony na rampu	36
Dopředné řízení	36
Obousměrné vážené řízení s váhou 0,5	38
4.2.3. Reakce kolony na poruchu	40
4.2.4. Diskuze výsledků	41
5. Další činnosti, návrhy na vylepšení a na co nezbyl čas	42
5.1. Mechanické sestavení vozidel	42
5.2. Udržení soudržnosti projektu	43
5.2.1. GIT	43
5.2.2. Návod, postupy, problémy a řešení	43
5.2.3. Dokumentace kódu	43
5.3. Návrhy na vylepšení	43
5.3.1. Senzor vzdálenosti	43
5.3.2. Řízení kolon	44
5.4. Na co nezbyl čas	44
6. Závěr	45
Přílohy	
A. Teorie modulace a demodulace signálu	47
A.1. Modulace a demodulace signálu	47
A.1.1. Modulace	48
Grafy	48
A.1.2. Rušení	49
Grafy	49
A.1.3. Demodulace	49
Synchronní detekce	50
Asynchronní detekce	50
Grafy	51
B. Příložené materiály na CD a webu	53
Literatura	54

1. Úvod

V úvodní kapitole bych rád čtenáře seznámil s projektem Slotcar platooning, který se zabývá sestavením platformy pro testování algoritmů řízení kolony. Uvedu zde svoji motivaci pro práci na projektu, představím způsob sestavení platformy a popíši její jednotlivé díly. Nakonec bych se pozastavil u softwaru, nastínil práci s ním a vysvětlil propojení platformy s počítačem.

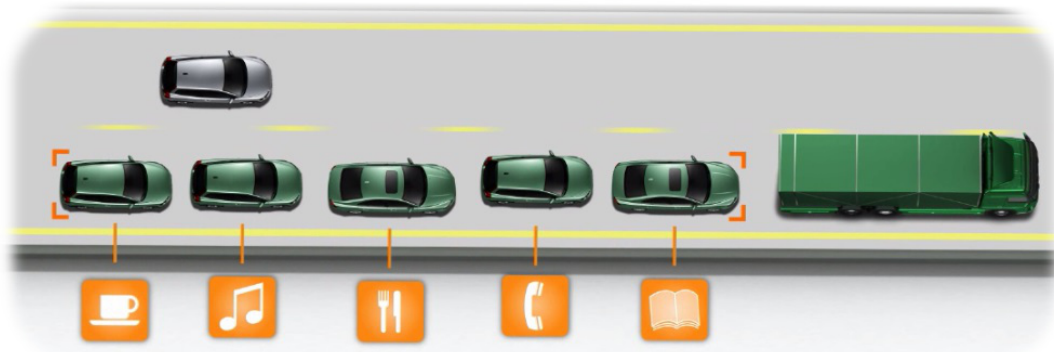
1.1. Odkaz na přiložené soubory bakalářské práce

Veškeré materiály k bakalářské práci (obsah popsán v příloze B) včetně práce samotné jsem umístil krom přiloženého CD i na web, ke stažení zde [1].

1.2. Motivace

Každý z nás se s tímto problémem setkal - dopravní kongesce, lidově řečeno dopravní zácpa. Počet vozidel na silnicích za posledních padesát let zaznamenal raketový růst. Dle serveru AutoSAP¹ je v České Republice k datu 30.6.2012 registrováno 4 638 372 osobních vozidel. Silnice se plní auty a často nezvládají jejich množství. Například v centru Prahy je znát, že při výstavbě magistrály nebylo s takto vysokou dopravou počítáno.

Jak ale tento problém řešit? Existuje více způsobů. Můžeme například dotovat veřejnou dopravu, propagovat cyklistiku, omezit počet osobních vozidel či rozšířit stávající dopravní infrastrukturu. V našem projektu máme poněkud odlišný přístup. Chtěli bychom umožnit testování algoritmů pro řízení dopravních kolon a tak nalézt optimální způsob pro pohyb velkého množství vozidel.



Obrázek 1. Příklad automaticky řízené kolony s vedoucím vozidlem (<http://sartre-project.eu/>)

Není pochyb, že automaticky řízená doprava je naší budoucností. Je jen otázka, jak moc vzdálenou. Člověk, který jede vozidlem v koloně, funguje ve své podstatě jako

¹<http://www.autosap.cz/dalsi-informace/pocty-registrovanych-vozidel-vozidel-v-cr-a-dalsi-udaje/>

1. Úvod

zpětnovazební regulátor rychlosti a vzdálenosti k následujícímu vozidlu. Nedokonalý regulátor, který má omezenou reakční dobu danou jeho biologickými možnostmi. Pokud bychom nahradili člověka počítačem a řídili tak kolonu automaticky, mohli bychom teoreticky optimalizovat pohyb kolony a zvýšit tak průchodnost silnic. Nemluvě o komfortu pro řidiče, který by měl náhle volné ruce. Výhody automaticky řízené dopravy jsou popsány například zde [2].

Cílem řízení kolon je tedy efektivní využití možností silnic, a to zejména díky udržení konstantních a malých rozestupů mezi vozidly.

Náš projekt není zdaleka jediným, který se tímto tématem zabývá. Příkladem může být evropský The SARTRE Project [3], který si dal za cíl vyvinout technologie a strategie pro automatické řízení kolon aut na dálnicích. Dále kalifornský rozsáhlý projekt PATH [4] [5], který hledá řešení pro optimalizaci dopravy. Další podobné výzkumy probíhají i na univerzitě v Eindhovenu [6]. Čím bychom tedy mohli přispět k tomuto tématu?

Chtěli bychom nabídnout nízkorozpočtovou jednoduchou testovací platformu nejen pro naši katedru, ale umožnit testování algoritmů i ostatním školám z celého světa. To bude vyžadovat robustní provedení, přehledný kód, podrobnou dokumentaci a popřípadě i jednoduché uživatelské rozhraní. Důležitým bodem je také počet vozidel v koloně. Dle našich informací bychom byli jedni z prvních, kteří umožní testování dosud jen teoretických algoritmů v praxi na počtu několika desítek aut.

A v neposlední řadě je zejména mou osobní motivací praktická zkušenost. Práce v týmu, projektové plánování, sestavování a zapojování elektronických obvodů, procesorové programování, simulace, užití teoretických znalostí v praxi a mnoho dalšího. To jsou vskutku nenahraditelné zkušenosti, které mi v budoucnosti budou bezpochyby velice užitečné.

1.3. Projekt Slotcar platooning

1.3.1. Lidé za projektem

Projekt vznikl na Katedře řídicí techniky ČVUT. Vedoucím je pan Ing. Zdeněk Hurák Ph.D., studenty pracující na bakalářské práci v rámci projektu mají na starosti doktorandi Ing. Ivo Herman a Ing. Dan Martinec. O hardwarový návrh a implementaci se stará Ing. Jaromír Dvořák.

Spolu se mnou pracují na různých úkolech a částech projektu další dva studenti: Martin Lád a Anastasia Vlasova. Společně jsme si rozdělili pro bakalářské práce základní kroky sestavení, modelace, komunikace a implementace algoritmů.

1.3.2. Požadavky pro sestavení modelu kolony

Cílem projektu Slotcar platooning je z autodráhy a aut sestavit model dopravní kolony, na kterém bude možné zkoušet algoritmy a regulátory pro její řízení - tedy regulátory pro udržení stejných a malých rozestupů mezi vozidly.

Chceme sestavit tento model co nejrealněji vzhledem k dostupným možnostem. To nám přináší řadu požadavků:

- Sestavení modelu pro 50 aut
 - Jeden ze základních požadavků, kvůli kterému projekt vznikl
- Autonomní řízení jednotlivých aut

- Auta by měla fungovat jako v reálném provozu, nezávisle na centrálním počítači
- Co nejvyšší nezávislost aut na vzájemné komunikaci
 - Opět příklad z reálného provozu, kdy auta mezi sebou nekomunikují
 - Pro speciální regulátory však i možnost vzájemné komunikace
- Bezdrátové propojení aut s počítačem
 - Pro nahrávání kódu z počítače do aut
 - Pro přenášení výstupu senzorů, regulátorů a vnitřních proměnných kódu do počítače
- Propojení s MATLABem
 - Pro zobrazení výstupních hodnot v reálném čase
 - Pro řízení vozidel v reálném čase
 - Možnost vytvoření regulátoru v MATLABu, následné přeložení do kódu procesoru a implementace
- Měření vzdálenosti (viz. kapitola 2)
 - Nutno měřit jak vzdálenost před, tak vzdálenost za autem (pro obousměrné řízení)
 - Rozsah od 0 cm do alespoň 20 cm
- Měření rychlosti auta
- Měření proudu motoru
- Měření zrychlení a měření úhlového zrychlení

Další nároky, jako například výkon napájecího zdroje či délka autodráhy, jsou logickým důsledkem těchto základních požadavků.

I vzhledem k požadavkům jsme se rozhodli pro autodráhu. To nám přineslo spoustu zjednodušení. Samotná auta jsou napájena z kolejnic autodráhy, nemusíme řešit zatáčení a můžeme si vybrat nejvhodnější vozidlo z velké nabídky trhu.

1.3.3. Platforma

Vzhledem k předchozí zkušenosti Ing. Dana Martince v jeho diplomové práci [7] jsme se rozhodli pro autodráhu německé společnosti Carrera, auta série Evolution, model Ford Capri RS 1975 (Obrázek 2).



Obrázek 2. Carrera Ford Capri RS 1975 (<http://www.carrera-toys.com>)

1. Úvod

Jedná se o vozidla o velikosti přibližně 20x10 cm. Měla by nám poskytnout dostatečný vnitřní prostor pro umístění desky plošných spojů s procesorem a pro umístění senzorů. Maximální rychlost přesahující tři metry za sekundu by také měla být více než přijatelná.

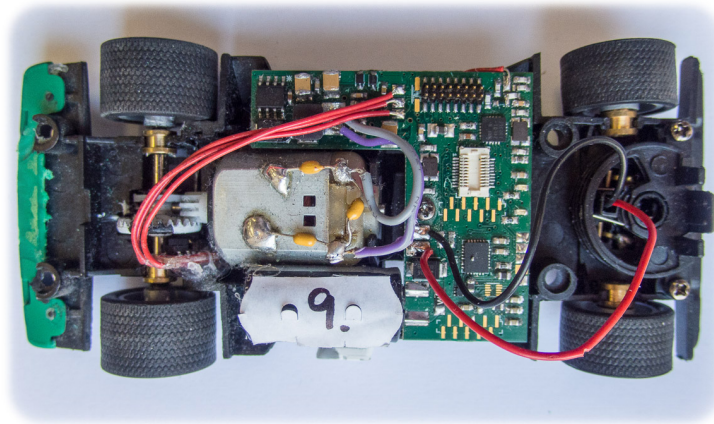
Vzhledem k velikosti aut a k požadované vzdálenosti mezi nimi, která bude dosahovat přes 20 cm, bude potřeba pořídit minimálně 25 m dráhy pro 50 vozidel.

1.3.4. Hardware a senzory

Návrh a implementaci hardwaru má na starost Jaromír Dvořák. Jen pro úplný přehled uvádím důležité použité součástky:

- ARM procesor STM32F40X
- Rádio NRF24L01SMD
- H-můstek pro PWM motoru DRV8816
- Krystal ABM8
- Senzor pro měření proudu INA213AIDCKT
- LDO regulátor TLV70028DSET
- Akcelerometr a gyroskop LSM330DLC
- LED dioda blikající při napájení auta
- Shottkyho dioda chránící proti opačnému zapojení auta
- Analogový senzor QRE1113 pro měření rychlosti
- IR LED dioda VSML3710 a IR fototranzistor VEMT3700F pro měření vzdálenosti (viz. kapitola 2)

Pro popis jednotlivých zapojení a funkcí součástek v této práci bohužel není místo. Tento úvod slouží jen jako stručný přehled. Po dostavění platformy je v plánu sepsat kompletní dokumentaci.



Obrázek 3. Vnitřek auta

Na obrázku 3 vidíme auto bez kapoty. Deska plošných spojů byla navržena tak, aby maximálně využila vnitřní prostory a přitom nebránila nasazení kapoty.

Dvojice červeného a černého drátu vede z napájecích kartáčků pod vozidlem, které odebírají napětí z dráhy. Pod nálepkou s číslem devět se nachází kondenzátor zajišťující napájení při výpadku a chránící proti špičkám. Fialový a šedý drát napájejí motor.

Tři červené dráty, které vedou k levému dolnímu kolu, jsou součástí IRC senzoru rychlosti. Piny konektoru v pravé horní části desky slouží k připojení a napájení dalších periférií, například senzoru vzdálenosti.

1.3.5. Software a propojení s počítačem

Samotný kód procesoru STM32 je psaný v jazyce C. Používáme běžně dostupné i vlastní knihovny. Základní program a systémové funkce sestavil Jaromír Dvořák. Kód zkompilujeme pomocí skriptu Makefile.

Nahrání programu do aut

Pro základní komunikaci s auty slouží program Bootloader, který připravil Jaromír Dvořák. Samotný Bootloader se musí do aut nahrát přes speciální vývojový kit připojený na programovací piny procesoru. Každému autu se určí vlastní číselná adresa. Poté již komunikace s auty probíhá bezdrátově.

Pro nahrání zkompilovaného kódu z počítače tedy pouze stačí spustit připravený program, který celý kód odešle skrz rádiový USB vysílač libovolnému autu podle adresy. Celý proces nezabere více jak deset sekund.

Komunikace s auty

Obrovskou výhodou je komunikace s programem MATLAB. V hlavním programu (main.c) máme připravené pole typu short, do kterého si můžeme ukládat libovolné hodnoty. Ty se poté odešlou rádiově do MATLABu, který data registruje a přijímá díky s-funkci. Dále máme vytvořený model v programu Simulink, ve kterém si přijímané datové pakety z auta přetypováváme, rozdělujeme a zobrazujeme.

Přes s-funkci a simulinkový model si také můžeme odeslat libovolné pole dat do aut, které poté zpracováváme v hlavním kódu.

Další možností komunikace s auty je přes terminál - program Term, který se spouští v příkazové řádce. Obsahuje základní metody pro kontrolu funkčnosti aut a změnu adresy.

V hlavním programu si dále můžeme přidat libovolné funkce a příkazy, které poté lze v terminálu využívat. Takto si například můžeme dynamicky nastavovat rychlost aut, referenční hodnoty, přepínací parametry či dokonce i hodnoty složek regulátorů.

Jedná se o rychlou a nenáročnou cestu, jak s vozidly komunikovat (na rozdíl od simulinkového schématu, které zas nabízí jiné výhody).

Vytváření regulátorů a dalších součástí v programu Simulink

V Simulinku máme dále možnost vytvořit libovolné regulátory a filtry, které za pomocí vestavěného kompilátoru přeložíme do ARM32 kompatibilního C kódu. Ten poté můžeme vložit přímo do hlavního programu. Tak lze snadno a rychle vytvořit i složitější regulátory a další součásti, které bychom jinak museli ručně psát.

Všechny použité regulátory jsme vytvářeli tímto způsobem.

2. Měření vzdálenosti

Pro řízení kolony je měření vzdálenosti k následujícímu a předchozímu vozidlu jedním z elementárních požadavků. Proto mu je také věnovaná velká část mé bakalářské práce. Na první pohled se jedná o jednoduchou záležitost, při realizaci jsem ovšem narazil na řadu problémů.

V této kapitole bych tedy rád čtenáře seznámil se svým řešením. Nejprve zopakuji a rozvedu požadavky, poté budu diskutovat různé možnosti řešení. Uvedu důvody, proč jsme se rozhodli vybudovat vlastní senzor pro měření vzdálenosti. Dále toto řešení detailně popíši teoreticky a uvedu zkušenosti s implementací. Nakonec zhodnotím vlastnosti takto navrženého senzoru.

2.1. Požadavky

Naším cílem je navrhnout a vytvořit hardware i software pro měření vzdálenosti s následujícími požadavky:

- Možnost současného měření vzdálenosti k následujícímu i předchozímu vozidlu
 - Vzdálenost měření k předchozímu vozidlu je nutná pro obousměrné řízení
- Dimenzovat velikost a hmotnost senzoru
 - Čím menší senzor, tím bude lepší výsledný efekt modelu skutečného vozidla
- Rozsah od 0 do alespoň 20 cm
- Možnost měřit vzdálenost pod úhlem alespoň -15° až 15°
 - Autodráha bude postavena v ideálním případě do kruhu
 - Pokud se však vyskytnou zatáčky, je nutno provádět měření k následujícímu a předchozímu autu pod různými úhly
- Měření s frekvencí minimálně 20 Hz
 - Rychlost lidské reakce většinou nesahá pod 100ms, pro strojové regulátory je ovšem reakční doba zásadní
- Přijatelná přesnost v rozmezí 0,25 cm.
 - Cílem samozřejmě bude přesnost maximalizovat, ale odezva má vyšší prioritu
- Odolnost proti případnému rušení z okolí
 - Autodráha s vozidly může být postavena na místech s různými světelnými podmínkami

Díky těmto nárokům jsme značně limitováni při výběru či sestavení senzoru.

2.2. Možnosti řešení

V této sekci bych rád diskutoval jednotlivé možnosti implementace měření vzdálenosti, nad kterými jsme uvažovali.

2.2.1. Komerční senzory

Na trhu se prodává celá řada senzorů pro měření vzdálenosti. Liší se zejména zaměřením, velikostí, cenou, použitou technologií a různými vlastnostmi. Pro naše požadavky jsme vybírali mezi optickou technologií a ultrasonickými senzory.

Ultrasonické senzory jsou ovšem dražší a jejich vlastnosti neodpovídají požadavkům. Mají například příliš velký úhel vysílání, špatný rozsah měřené vzdálenosti a jsou dražší. Při použití vyššího množství ultrasonických senzorů by také docházelo ke vzájemnému rušení.

Optické senzory

V robotice jsou velice oblíbené senzory Sharp¹. Nabízí řadu výrobků určené pro různé vzdálenostní rozsahy. Cena těchto senzorů se pohybuje v řádu stokerun.

Senzor značky Sharp byl použit i v diplomové práci [7], ze které jsme se pro náš projekt často inspirovali. Například senzor Sharp GP2Y0A41SK0F (Obrázek 4) má v datasheetu² uvedeny následující vlastnosti:

- Rozsah 4 až 30 cm
- Odezva 16,5 ms
- Rozměry 29,5 x 13,0 x 13,5 mm
- Analogový výstup



Obrázek 4. Senzor Sharp GP2Y0A41SK0F (<https://www.tinkerforge.com/>)

Optický senzor Sharp měří vzdálenost triangulační technologií [8]. Výhody tohoto měření jsou výrazná nezávislost na odrazivosti povrchu předmětu, ke kterému vzdálenost měříme, a vysoká přesnost.

I přesto tento senzor nespĺňuje všechny naše požadavky. V rozsahu je mrtvé pásmo do 4 cm, kde senzor chybně měří vzdálenost maximální. Rozměry senzoru nejsou úplně kompaktní, museli bychom je složitě mechanicky přidělat na vozidlo. Úhel je také příliš směrový, měří jen vzdálenost k vozidlu, které je přímo před ním.

Tyto všechny nedostatky by se daly nějakým způsobem obejít či vyřešit. Nejzávažnějším je však nemožnost nasměrování dvou senzorů tohoto typu naproti sobě kvůli vzájemnému rušení. Diskutovali jsme následující možná východiska:

1. Mohli bychom auta časově synchronizovat a senzory střídavě spínat. Senzoru ovšem trvá náběh do plné funkce 60 ms, což by způsobilo zpoždění. Také nemáme vyzkoušeno, jakým způsobem by spínání senzoru ovlivnilo jeho životnost.

¹<http://www.sharpsme.com/optoelectronics/sensors/distance-measuring-sensors>

²http://sharp-world.com/products/device/lineup/data/pdf/datasheet/gp2y0a41sk_e.pdf

2. Měření vzdálenosti

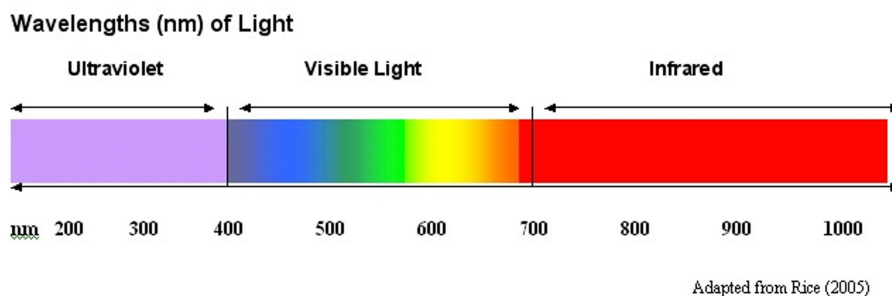
2. Mohli bychom také měřit jen vzdálenost k následujícímu vozidlu a tu posílat k předchozímu bezdrátovou komunikací. To by ovšem opět porušilo požadavek nezávislosti a přineslo zpoždění. V jednom okamžiku může vysílat jen jedno auto. Museli bychom také implementovat algoritmus pro zjištění pořadí v koloně a nesměli bychom poté vyjmout či přidat žádné vozidlo do kolony.

Ani žádný z prodávaných optických senzorů nesplnil naše požadavky. Na první pohled se to nezdá, ale najít senzor, který by vyhovoval specifickým požadavkům a byl cenově přijatelný, je, i přes rozsáhlou nabídku, velmi těžké. Rozhodli jsme se proto vybudovat vlastní optický senzor.

2.3. Návrh vlastního optického senzoru

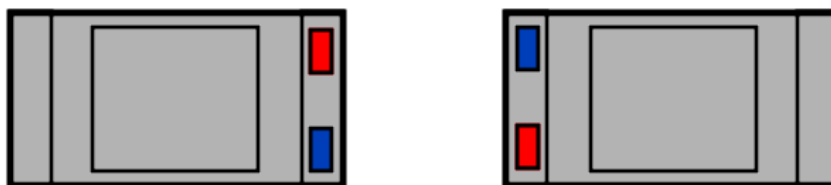
Jelikož nám nestačily komerční senzory, rozhodli jsme se vytvořit vlastní. V podstatě prvním návrhem, u kterého jsme poté i zůstali, byla kombinace infračervené (IR) LED diody a fototranzistoru citlivého na infračervené světlo.

Světlo je druh elektromagnetického vlnění. Viditelné světlo je vlnění o délce v rozmezí přibližně 400 až 700 nm (Obrázek 5). Infračervené světlo má pak vlnovou délku vyšší, sahá až k mikrovlnnému záření. Světlo ze Slunce, zářivek a žárovek obsahuje sice i infračervené světlo, ale je ho méně než viditelného. To nám pomůže omezit okolní rušení.



Obrázek 5. Vlnová délka světla (<http://anders06116diy.blogspot.cz/>)

Fototranzistor umístíme na předek jednoho auta, LED diodu umístíme na zadní část druhého auta. Fotodioda bude vysílat světelný signál, který zachytíme na fototranzistoru. Z amplitudy tohoto signálu získáme vzdálenost prvního auta od druhého. Stejným způsobem změříme i vzdálenost druhého auta od prvního (Obrázek 6).



Obrázek 6. Pohled na auta se senzory shora - modrá je fototranzistor, červená LED dioda

Díky tomuto zapojení měříme nezávisle na odrazu vysílaného signálu (na rozdíl od senzoru Sharp), čímž se zbavíme řady omezení a problémů. Díky odrazu by došlo ke

značnému poklesu amplitudy, která by navíc byla ovlivněna povrchem, od kterého se signál odráží (tedy např. pro různé barvy aut by jsme měřili různé amplitudy, tedy i různé vzdálenosti). Další nevýhodou by byl úhel odrazu, který by se v zatáčkách autodráhy výrazně měnil. Při našem zapojení jsme omezeni jen směrovostí diody a fototranzistoru.

Značným omezením může na druhou stranu být umístění vysílací diody na objektu, ke kterému chceme měřit vzdálenost. To pro naši platformu ovšem není problém.

2.3.1. Výběr součástek

První pokusy ukázaly, že pro měření v rozsahu alespoň 20 centimetrů potřebujeme velmi výkonnou LED diodu a citlivý fototranzistor na stejné vlnové délce. Dále máme požadavek na všesměrovost a pro modulaci signálu musíme LED diodu spínat s frekvencí okolo tisíce hertzů (vysvětleno dále v sekci 2.5.1).

Tyto požadavky nám výběr velmi zjednoduší. Nakonec jsme se rozhodli ve finální verzi použít IR LED diodu Vishay VSML3710³ a NPN fototranzistor Vishay VEMT3700F⁴.

Následující parametry a vlastnosti stojí za zmínku:

- LED dioda Vishay VSML3710
 - Hlavní vlnová délka 940 nm
 - Úhel poloviční intenzity $\pm 60^\circ$
 - Zářivost až $20 \text{ W}\cdot\text{sr}^{-1}$
 - Napájecí proud 100 mA
 - Cena přibližně 11 Kč
- fototranzistor Vishay VEMT3700F
 - Spektrální intenzita se špičkou 950 nm
 - Úhel poloviční intenzity $\pm 60^\circ$
 - Filtry blokující denní světlo
 - Cena přibližně 13 Kč

2.3.2. Navržené zapojení

Navržené zapojení LED diody VSML3710 a fototranzistoru VEMT3700F je na obrázku 7.

LED diodu zapojíme do série s odporem R1, který reguluje proud protékající diodou. Fototranzistor musíme zapojit do série s odporem R2, který nastavuje jeho citlivost. Kondenzátory C1 a C2 slouží pro filtraci a jako lokální zdroj proudu. Spínání LED diody provádíme přes tranzistor.

Měřený signál je napětí na fototranzistoru, vzorkovací obvod tedy připojíme mezi svorky A a B.

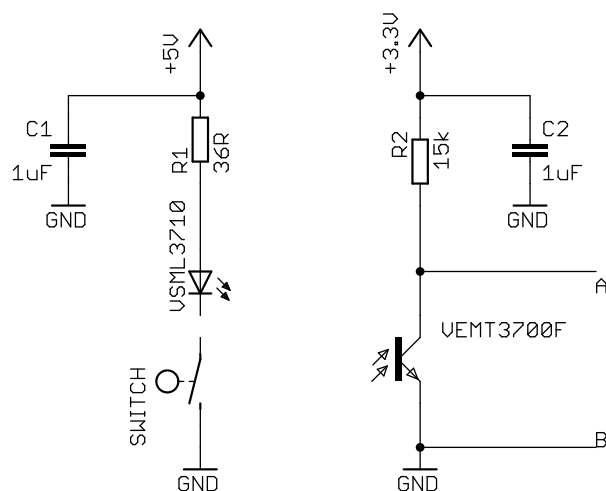
Velikost odporu R1 určíme vztahem:

$$R = \frac{U_{\text{napájecí}} - U_{\text{úbytek}}}{I} [\Omega] \quad (1)$$

kde $U_{\text{napájecí}}$ je napájecí napětí obvodu, $U_{\text{úbytek}}$ je úbytek napětí na diodě a I je napájecí proud diody. Pro nás tedy:

³<http://www.vishay.com/docs/81300/vsml3710.pdf>

⁴<http://www.vishay.com/docs/81584/vemt3700.pdf>



Obrázek 7. Zapojení LED diody a fototranzistoru

$$R = \frac{5 - 1.5}{0,100} = 35\Omega$$

Zjištění odporu R2 je náročnější. Můžeme si změřit, že otevřený fototranzistor, tedy fototranzistor ozářený IR diodou, má vnitřní odpor řádově tisíce ohmů. Abychom na něm tedy mohli vůbec nějaké úbytky napětí změřit, musíme k němu sériově zapojit odpor o hodnotě také tisíce ohmů.

Platí, že čím vyšší bude předřazený odpor R2, tím vyšší bude citlivost fototranzistoru a tím vyšší bude dosah měření vzdálenosti. Na druhou stranu se tím ale zvyšuje i jeho citlivost na šum. Navíc, jak uvedu v sekci 2.7.2, závislost naměřeného napětí na vzdálenosti fototranzistoru od LED diody je velmi nelineární, naším cílem je získat co nejvyšší přesnost v rozsahu okolo 10 centimetrů, kde se budou nejčastěji vozidla pohybovat.

Tedy hodnotu odporu R2 jsem zjistil iterací pokusů. Nejlepší vlastnosti pro měření byly přibližně při 15 000 ohmech.

2.3.3. Sestavení senzoru, výsledky a motivace pro modulaci

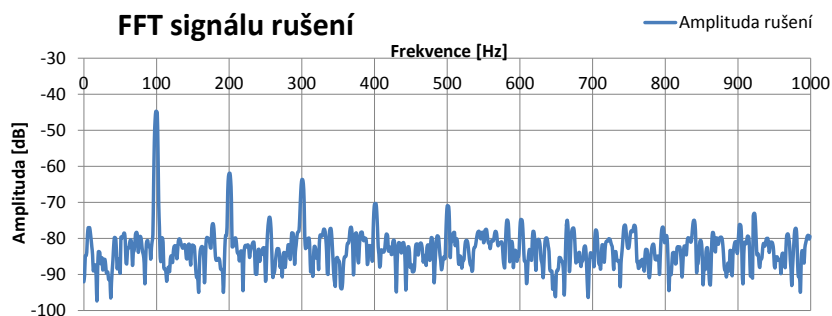
Vzhledem k zdůvodnění důležitosti modulace a demodulace signálu uvedu výsledky takto vytvořeného senzoru. Byly až překvapivě dobré. Umístili jsme obvody do dvou zkušebních autíček a podle potřeb nastavili odpory. Na první pohled se zdálo být toto řešení funkčním a vyhovujícím.

Naše zkušební platforma byla ovšem umístěna mimo přímé sluneční světlo a zářivky byly zhasnuté. Také jsme měřili pouze jednosměrně, neumístili jsme dva takto postavené senzory naproti sobě.

Tím jsme se ovšem vyvarovali všemu vnějšímu rušení.

Takto sestavený senzor je na něj velmi citlivý. V případě rozsvícení zářivek došlo ke kmitání měřeného napětí na frekvenci násobků stovek hertz a Slunce nám stejnosměrně posunulo amplitudu signálu.

Frekvenční charakteristiku okolního rušení, změřenou funkcí Fast Fourier Transformation (FFT) na osciloskopu, zachycuje graf 8. Všimněme si zejména vysoké hodnoty rušení s frekvencí 100 Hz a jejích násobků, tedy zářivek. Měření jsem provedl za dne v místnosti s rozsvícenými světly na fototranzistoru zapojeného dle schématu 7.



Obrázek 8. Graf zachycující amplitudu rušení jednotlivých frekvencí

Pokud zapojíme dva senzory naproti sobě dle obrázku 6, bude se světlo odrážet a výrazně nám také posune měřenou amplitudu.

Naším dalším úkolem tedy bude nějakým způsobem rušení potlačit, vyfiltrovat ho. Elegantním řešením může být modulace vysílaného světelného signálu.

2.4. Modulace a demodulace signálu

Vzhledem k náchylnosti předchozího řešení k rušení je nutné provést modulaci a následnou demodulaci našeho signálu, vysílaného IR LED diodou a snímaného fototranzistorem.

Nejprve je nutné vysílaný infračervený světelný signál modulovat. Nejjednodušší modulace je amplitudová. V našem případě budeme pouze spínat LED diodu několikrát za sekundu. Tak vytvoříme v ideálním případě obdélníkový signál.

Dalším krokem je tento signál přijmout fototranzistorem a demodulovat. V ideálním případě přijmeme opět obdélníky a demodulací z nich vytvoříme spojitý signál. Jediný rozdíl mezi původním a demodulovaným signálem by měl být v amplitudě.

Tímto procesem bychom také měli odfiltrovat veškeré rušení, ať už se jedná o zářivky, které náš signál ruší periodicky na frekvenci 100 Hz, sluneční světlo, které můžeme považovat za stejnosměrné rušení, či prostě náhodný šum.

Teorii ohledně procesu modulace a demodulace uvedu do přílohy A. Pro naši práci není zásadní, slouží spíše pro čtenářovo pochopení celého procesu. Důležitý je zejména závěr, který nám říká, že demodulací za pomoci asynchronní detekce odfiltrujeme veškeré stejnosměrné rušení a značně potlačíme periodické o rozdílných frekvencích, než je ta modulační.

2.5. Vysvětlení kódu modulace a demodulace

Při psaní metod v jazyce c pro modulaci a demodulaci signálu jsem si pro názornost i kontrolu vytvářel MATLABovský skript s kódem shodné funkčnosti. Tento postup se mi bohatě vyplatil. Nalezl a odstranil jsem takto spoustu chyb v indexech polí, různých cyklech a složitějších funkcích. Také mi to umožnilo ujasnit si funkčnost implementace.

Nyní tento kód využiji pro názornou ukázkou, než se však dostanu ke grafům, objasním nejdříve zvolenou modulační frekvenci.

2.5.1. Modulace, demodulace a nastavení modulační frekvence

Stanovení frekvence, se kterou budeme provádět modulaci signálu, je zásadní krok. Nejprve si ale musíme objasnit způsob modulace a demodulace v programovém kódu.

Modulace a demodulace v programovém kódu

Modulaci signálu provádíme střídavým zapínáním a vypínáním napájení LED diody, k čemuž nám slouží jednoduchý příkaz. Procesor pracuje v diskretních krocích, spínání tedy vyřešíme přes čítač, který po stanovený počet kroků drží hodnotu napájení LED diody ve stavu zapnuta či vypnuta.

Pokud si velikost tohoto čítače označíme N , bude počet kroků v jedné periodě modulace roven hodnotě $2 \cdot N$.

Demodulace signálu je složitější. Budeme přitom vycházet z poznatků asynchronní detekce popsané v příloze A. Zejména jsou pro nás důležité rovnice (18)-(20), které si upravíme do diskretního tvaru:

$$S_{demod,sin} = \frac{1}{2 \cdot N} \sum_{n=1}^{2 \cdot N} \left[S_{mod,ruš}(n) \cdot \sin \left(\frac{2\pi}{2 \cdot N} n \right) \right] \quad (2)$$

$$S_{demod,cos} = \frac{1}{2 \cdot N} \sum_{n=1}^{2 \cdot N} \left[S_{mod,ruš}(n) \cdot \cos \left(\frac{2\pi}{2 \cdot N} n \right) \right] \quad (3)$$

$$S_{demod} = \sqrt{(S_{demod,sin}^2 + S_{demod,cos}^2)} \quad (4)$$

Diskretní krok procesoru je označen písmenem n . Po $2 \cdot N$ kroků provádíme demodulaci a načítání signálu $S_{mod,ruš}$ zachyceného na fototranzistoru jak pro sinovou, tak pro cosinovou složku. Jednou po $2 \cdot N$ krocích provedeme celkovou demodulaci odmocněním součtu sinové a kosinové složky demodulovaného signálu a získáme tak demodulovaný signál S_{demod} .

Přesně tímto způsobem provádím demodulaci signálu i v programovém kódu. Pouze jsem mírně optimalizoval tuto metodu pro procesorové výpočty. Hodnoty sinu a kosinu si například při spuštění programu načtu do pole a pak je jen volám, výpočty provádím v celých číslech a podobně.

Určení hodnoty N (velikosti čítače) úzce souvisí s nastavením modulační frekvence.

Modulační frekvence

Vycházet budeme z následujících třech předpokladů:

1. Procesor bude volat funkci, ve které modulujeme a demodulujeme signál, s frekvencí 40 000 Hz
2. Víme, že frekvence rušení zářivek je 100 Hz
3. Budeme potřebovat dvě rozdílné frekvence, zvláště pro měření vpřed a pro měření vzad (kvůli minimalizaci vzájemného rušení při odrazu signálu)

Předpoklady jsou seřazeny podle významnosti. Vzájemné rušení při odrazu má nejnižší prioritu, neboť ho můžeme započíst do přepočtu změřené amplitudy signálu na vzdálenost a tím s ním ve výsledku počítat. Signál LED diod budeme totiž modulovat stále, ale zářivky mohou být vypnuté či zapnuté a tím nám posunout amplitudu

demodulovaného signálu. Zvolíme tedy (kvůli zářivkám) frekvenci výrazně vyšší než 100 hertzů a to takovou, že nebude ani násobkem sta.

Při výběru vhodné modulační frekvence jsme nejvíce omezeni frekvencí volání funkcí pro modulaci a demodulaci procesorem.

Implementace v kódu se řeší přes čítač, který po stanovený počet cyklů drží hodnotu napájení LED diody ve stavu zapnuta či vypnuta.

Výslednou frekvenci modulace získáme vzorcem:

$$f_{mod} = \frac{40000}{2 \cdot N} \quad (5)$$

kde N je velikost čítače. Pro např. $N = 20$ získáme frekvenci modulace 1000 hertzů. Pokud pro demodulaci využijeme rovnic (2)-(4), získáme tak hodnotu demodulovaného signálu diskrétně s frekvencí 1000 hertzů ze 40 měření na fototranzistoru. Zpoždění takto získaného signálu bude samozřejmě 0,001 vteřiny.

Vzhledem k důležitosti těchto vztahů je zapíši zvlášť:

- Velikost čítače modulace: N
- Frekvence modulace: $f_{mod} = \frac{40000}{2 \cdot N}$ [Hz]
- Počet naměřených hodnot za periodu: $2 \cdot N$
- Zpoždění měření: $\frac{1}{f_{mod}}$ [s]

Předpokládejme, že chceme průměrovat s co nejmenším zpožděním, ale chceme také průměrovat z dostatečného počtu naměřených hodnot. Z těchto všech úvah budeme modulovat signál s frekvencí okolo 1000 Hz.

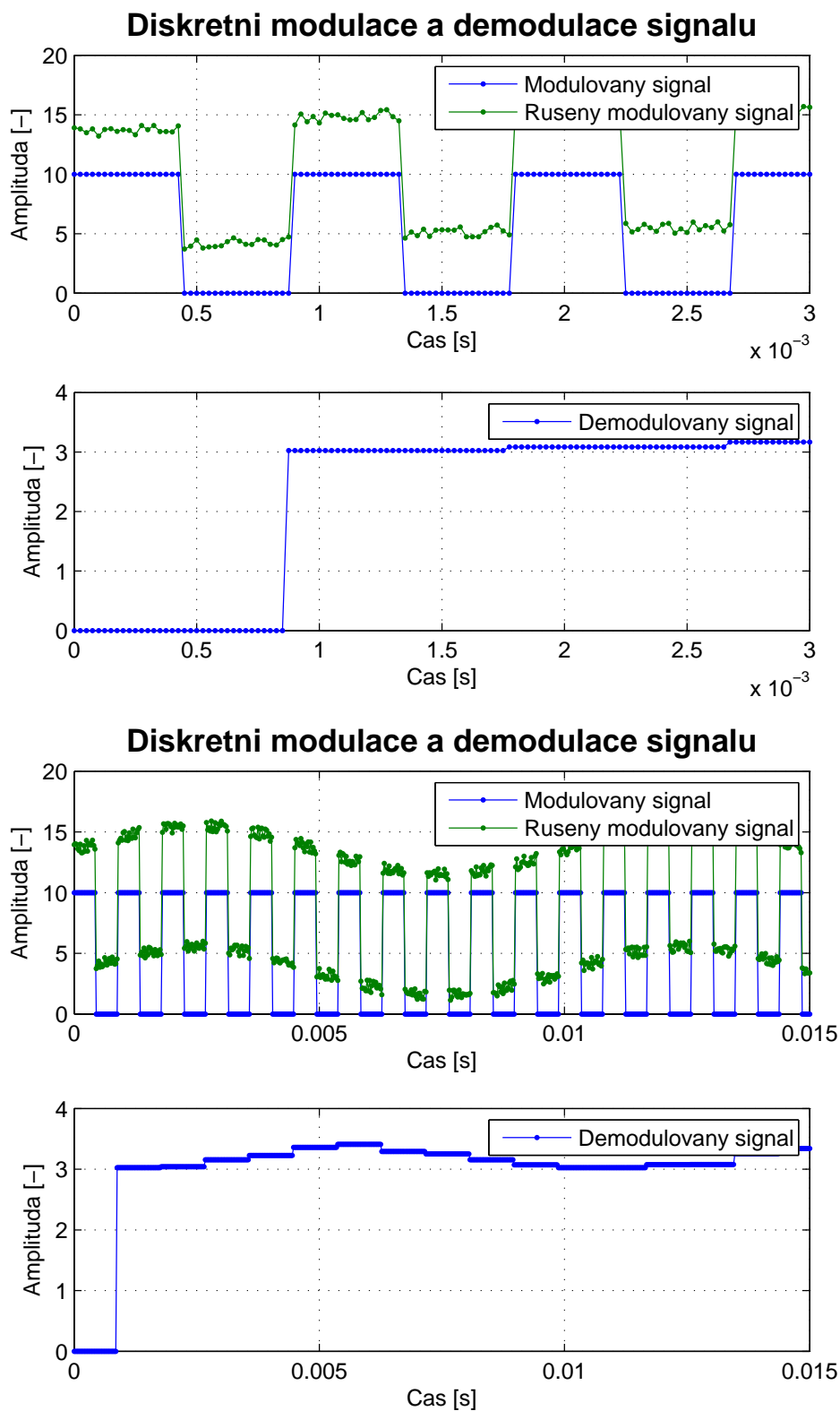
Implementací a následnými pokusy jsem zvolil následující velikosti čítačů a tedy i následující frekvence modulace:

- **Měření vzdálenosti vpřed**
 - Velikost čítače $N = 18$
 - Modulovací frekvence $f_{mod} \approx 1111$ Hz
- **Měření vzdálenosti vzad**
 - Velikost čítače $N = 12$
 - Modulovací frekvence $f_{mod} \approx 1667$ Hz

2.5.2. Ukázka modulace a demodulace v MATLABovském skriptu

Cílem je demonstrovat činnost modulace a demodulace v programu procesoru. Zvolená modulační frekvence byla využita i v implementaci, frekvence harmonického blikání zářivky také odpovídá skutečnosti. Důležité parametry pro grafy jsou:

- Vzorkování 40 000 Hz
- Frekvence modulace 1 111 Hz
- Amplituda nosného signálu 10
- Celkem 3 druhy rušení
 1. Konstantní s amplitudou 3
 2. Zářivky: $3 \cdot \sin(2 \cdot \pi \cdot 100 \cdot t)$
 3. Náhodný šum s amplitudou v rozmezí $\langle 0, 1 \rangle$



Obrázek 9. Diskrétní modulace a demodulace signálu, kratší a delší časový úsek - horní graf zachycuje modulovaný a rušený signál, dolní signál demodulovaný

Na grafech 9 si zejména všimněme vzorkování, které je naznačeno tečkami. Dále vidíme, že aritmetický průměr se spočítá jednou za periodu modulace.

2.5.3. Diskuze výsledku simulací

Úspěšně jsme experimentálně prokázali funkčnost modulace a demodulace infračerveného světelného signálu. Úplně jsme odfiltrovali stejnosměrné rušení a téměř úplně redukovali rušení zářivek.

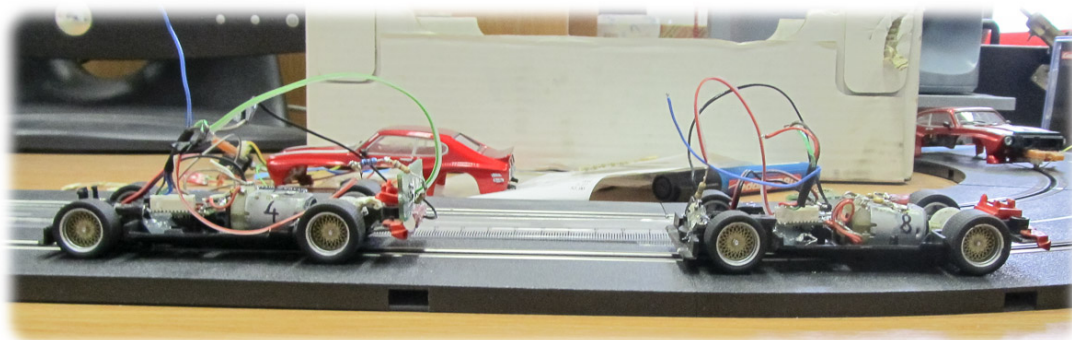
V praxi se nám samozřejmě nepodaří zachytit na fototranzistoru dokonalé obdélníky (obrázek 11). Pokud však zachycený signál bude periodický s námi zvolenou frekvencí, podaří se nám ho úspěšně demodulovat.

V přílohách bakalářské práce se samozřejmě nachází jak model modulace v Simulinku, tak skript simulující diskrétní demodulaci v MATLABu.

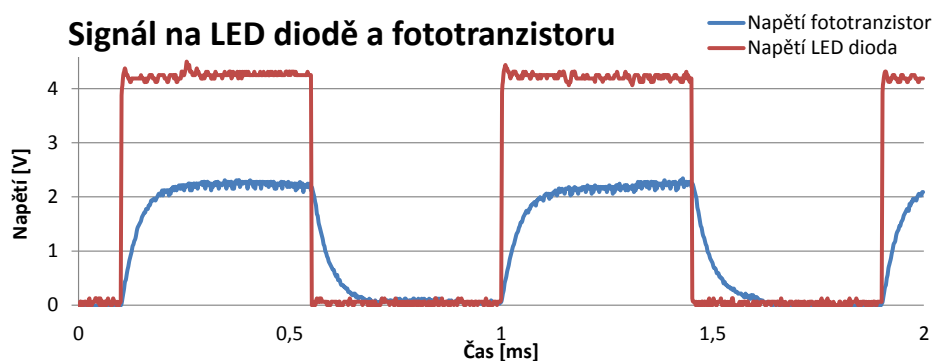
Tímto bych rád ukončil tuto teoretickou část kapitoly a dále bych popsal výsledky skutečných pokusů.

2.6. Praktická implementace řešení

Plošný spoj jsme vytvořili dle schématu zapojení 7. Každý plošný spoj obsahuje LED diodu a fototranzistor, destičku jsem vždy připevnil na přední a zadní část vozidla dle obrázku 6. Sestavená testovací platforma je na obrázku 10.



Obrázek 10. Testovací platforma pro měření vzdálenosti



Obrázek 11. Výstup z osciloskopu - napětí na LED diodě a protilehlého fototranzistoru při vzdálenosti 6 cm

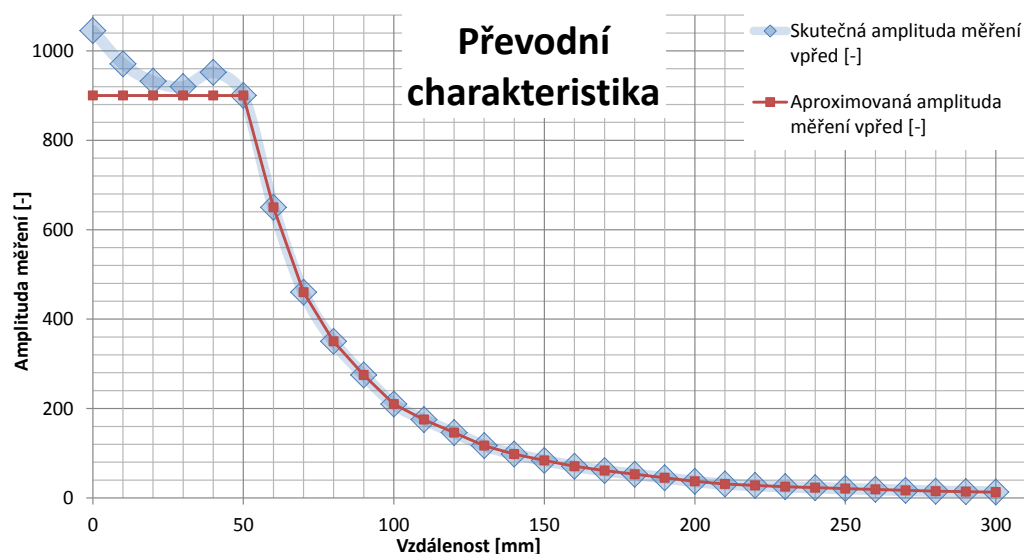
Na grafu 11 s výstupem z osciloskopu je pro představu signál napětí na LED diodě v porovnání s napětím na protilehlém fototranzistoru. Z obrázku je patrné, že zachycený signál fototranzistoru nemá ideální tvar, je však periodický.

Vyladil a odstranil jsem chyby v kódu a provedl řadu měření. Implementace se i přes počáteční potíže ukázala úspěšnou. Demodulovaný signál, který jsme si posílali z vozidel do modelu v Simulinku, byl stabilní a odolný proti rušení. Tím jsme v praxi ověřili princip modulace a demodulace pomocí asynchronní detekce.

2.6.1. Přepočítání amplitudy signálu na vzdálenost

Nyní potřebujeme na základě amplitudy signálu stanovit vzdálenost. Postavil jsem si dvě auta naproti sobě (obrázek 10) a zaznamenával jsem průměrnou amplitudu signálu vždy po centimetru. Tato data jsem poté vynesl do grafu 12, kde světle modrá barva představuje skutečný průběh amplitudy vzhledem ke vzdálenosti senzorů.

Data po demodulaci jsou zašuměná, abych naznačil takovouto nepřesnost, zvolil jsem tlustší čáru průběhu.



Obrázek 12. Převodní charakteristika amplitudy signálu na vzdálenost

Všimněme si nepříjemné vlastnosti charakteristiky, se kterou jsme nepočítali. Do dvou centimetrů nám amplituda signálu plynule klesá, pak ale mezi třetím a čtvrtým centimetrem roste. Dále pak klesá až do třiceti centimetrů, kde jsem kvůli nízkým hodnotám amplitudy měření ukončil.

Z tohoto nepředvídaného trendu nám plyne velice nepříjemná vlastnost konečného senzoru: máme mrtvé pásmo, ve kterém nejsme schopni měřit vzdálenost.

V dalším kroku jsem vytvořil v kódu programu procesoru převodní tabulku, kam jsem zadal přibližně jednotlivé středy naměřených hodnot amplitudy. Byl jsem ovšem nucen při hodnotách amplitudy vyšších než 900 nastavit konstantní vzdálenost 5 centimetrů. Při hodnotách amplitudy menších než 13 jsem nastavil vzdálenost 30 centimetrů. V nezměřených vzdálenostech jsem si hodnoty aproximoval přímkou.

V grafu 12 je tato aproximovaná charakteristika vyznačena červenou tenkou čarou.

Charakteristiku jsem stanovil jak pro měření vzdálenosti vpřed, tak pro měření vzad. Hodnoty jednotlivých amplitud demodulovaného signálu se totiž v obou případech mírně lišily.

Nyní jsme již schopni úspěšně změřit vzdálenost mezi dvěma vozidly.

2.6.2. Filtrace vypočtené vzdálenosti

Pro větší stabilitu a přesnost měření budeme vypočtenou vzdálenost filtrovat. Demodulovaný signál je totiž zašuměný, tedy i vypočtená vzdálenost osciluje kolem střední hodnoty s výchyly okolo 10% amplitudy.

Průběh měřené vzdálenosti také obsahuje náhodné špičky, kdy se nejspíš ztratil krátce signál.

Rozhodl jsem se použít jednoduchý IIR filtr daný vzorcem:

$$y[n] = y[n - 1] + k \cdot (x[n] - y[n - 1]) \quad (6)$$

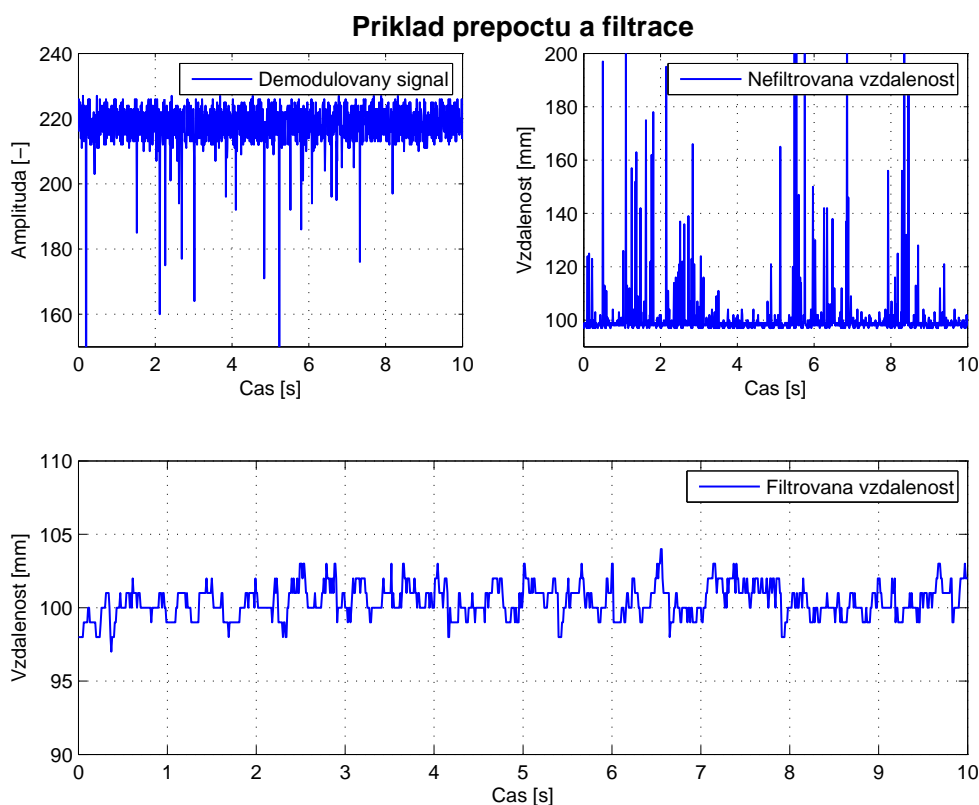
kde $y[n]$ je aktuální filtrovaná hodnota vzdálenosti, $y[n - 1]$ je předešlá filtrovaná hodnota, $x[n]$ je aktuální vypočtená vzdálenost a $k \in (0, 1)$ je konstanta.

Konstantou k tlumíme změny filtrované vzdálenosti. Pro $k = 1$ nedojde k žádnému filtrování, naopak pro velmi malé k se bude měřená vzdálenost měnit jen pomalu.

Při nastavování konstanty musíme mít na mysli zejména frekvenci měření. Náš senzor dává výslednou vzdálenost jednou za necelou milisekundu.

Pro špičky v měření, tedy kdy rozdíl $(x[n] - y[n - 1])$ je větší než 2 cm, jsem nastavil $k = 0,005$. Potřebujeme je totiž co nejvíce zanedbat. Musíme ale pamatovat na případ, kdy následující auto sundáme z dráhy a tím dojde ke chtěné špičce, která se musí projevit na výstupu v rozumném čase.

Pro ostatní případy jsem dal $k = 0, 1$.



Obrázek 13. Příklad přepočtu demodulovaného signálu na vzdálenost a její následná filtrace, měřeno ve vzdálenosti 100 mm

Na grafech 13 jsou vyneseny postupně hodnoty demodulovaného signálu, z nich vypočtené hodnoty nefiltrované vzdálenosti a konečná filtrovaná vzdálenost. Vidíme, že jsme filtrace eliminovali špičky a konečná vzdálenost je velmi stabilní.

2.7. Výsledný senzor

Výsledný senzor měření vzdálenosti není nijak komplikovaný. Je to vlastně jen spínaná LED dioda a fototranzistor, na kterém měříme napětí.

Mnohem komplikovanější byl proces vybrání součástek, vyzkoušení a ověření teorie a napsání a vyladění kódu. Výhodou je, že jakmile se nám podařilo úspěšně měřit vzdálenost pro dvě auta, stačilo jen sestavit a zapojit další senzory dle funkčního prototypu.

2.7.1. Vlastnosti senzoru

Výsledný senzor má následující parametry:

- Rozsah měření 5 až 30 centimetrů
 - Narozdíl od senzoru Sharp jsme však schopni určit, nacházíme-li se v mrtvém pásmu do 5 centimetrů.
- Frekvence měření vyšší než 1000 Hz
 - Konečná filtrace tuto hodnotu sníží, jsme však schopni dle potřeb frekvenci zrychlit či zpomalit.
- Vysoká odolnost proti rušení
- Možnost měření obousměrně
 - Odrazivost je započítána do převodní charakteristiky. Auta jsou schopna také detekovat překážky na trati.
- Schopnost měření vzdálenosti pod úhlem až 60°
 - Čím vyšší úhel, tím se ovšem více zkreslí měřená vzdálenost (při 60° klesne přibližně na polovinu)
 - V konečné implementaci bude senzor ukrytý pod kapotou auta (místo světel), úhlovou závislost tedy změříme až po sestavení (viz. 2.7.2)
- Přesnost až 0,2 cm
 - Přesnost se ovšem liší podle vzdálenosti. Nejvyšší je v rozmezí 5 až 14 cm, pak postupně klesá a od 20 cm je přibližně 2 cm.
- Cena přibližně 30 Kč
- Malá velikost
 - Všechny součástky jsou v SMD pouzdrů, největší část je tedy deska plošných spojů

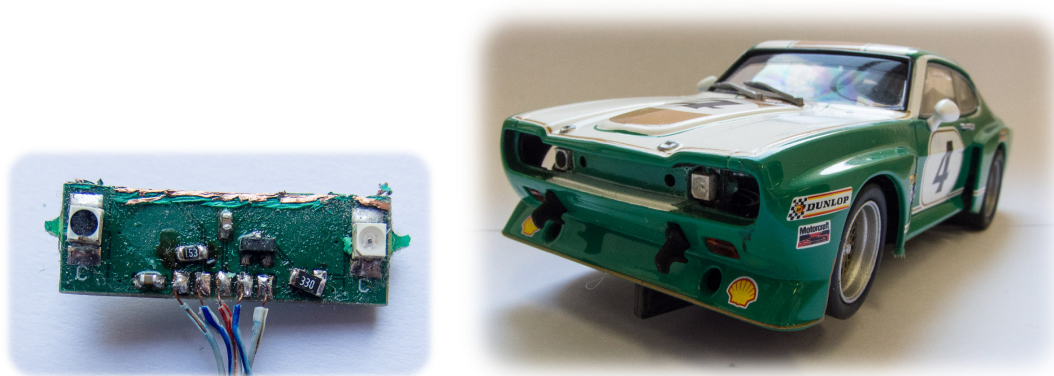
Pro nás jsou tyto vlastnosti více než slibné. V jiných aplikacích může být velkou překážkou **nutnost dvou senzorů namířených proti sobě**, namísto funkčnosti například odrazem signálu.

K jeho provozu je také **nutný výkonný kontrolér**, neboť proces demodulace a následného přepočtu amplitudy na vzdálenost je výpočetně náročný.

2.7.2. Montáž senzoru

Rozhodli jsme se umístit výsledný senzor pod kapotu vozidla. Tím skryjeme veškerou elektroniku a vylepšíme tak celkový estetický dojem.

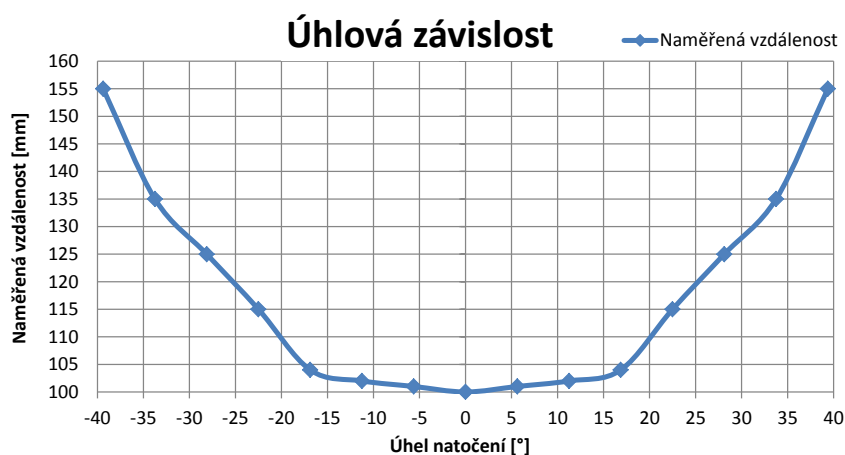
Bylo jen nutné mechanicky upravit kapotu i desku plošného spoje tak, aby se senzor vešel a optické součástky byly umístěny v dírách kapoty po světlech. Deska a její následné zabudování je zachyceno na obrázcích pod číslem 14.



Obrázek 14. Senzor pro měření vzdálenosti: vlevo osazená deska plošného spoje, vpravo deska zabudovaná ve vozidlu

Úhlová závislost zabudovaného senzoru

Pro takto dokončené vozidlo jsem změřil úhlovou závislost, tedy jak úhel natočení vozidel vůči sobě ovlivní měřenou vzdálenost. Výsledek jsem vynesl do grafu 15.



Obrázek 15. Úhlová charakteristika - dvě auta postupně vychylována ve vzdálenosti deset centimetrů

Měření jsem provedl následovně: Na rovnou trať jsem umístil dvě auta se senzory naproti sobě, jedno vůči druhému jsem postupně vychyloval a změřené hodnoty nanášel do grafu. Skutečnou vzdálenost mezi auty jsem udržoval konstantní.

2.7.3. Diskuze vlastností výsledného zabudovaného senzoru

Pokud porovnáme vlastnosti našeho senzoru s požadavky, zjistíme, že jsme je téměř všechny splnily. Senzor je malý, levný, v požadovaném pásmu přesný, odolný proti rušení a nepříliš závislý na úhlu měření.

Nejprudší zatáčka autodráhy, kterou v konečné implementaci použijeme, způsobí úhlově vychýlení maximálně 15° . Tento vliv na měřenou vzdálenost je (z grafu 15) zanedbatelný.

Rozsah měření není ideální díky limitní spodní hranici 5 centimetrů. Tento výsledek nás skutečně v konečné implementaci překvapil. Myslíme si, že by se ovšem dal vylep-

2. Měření vzdálenosti

šit jiným zapojením, kdy například nebudeme měřit napětí na fototranzistoru, ale na předřadném odporu. Věříme, že lepším návrhem schématu bychom mohli výsledky ještě zdokonalit, neměli jsme ovšem čas tyto úvahy potvrdit.

Důležité je, že náš senzor překonal (pro naše kritéria) všechny komerční senzory měření vzdálenosti, nad kterými jsme uvažovali. Jeho použití je však velice specifické.

3. Regulace a teorie řízení kolon vozidel

V této kapitole bych nejdříve objasnil způsob regulace pohybu autodráhových vozidel, dále bych uvedl navržené regulátory a nakonec se krátce zaměřil na základní teorii řízení kolon.

Chtěl bych zde objasnit pojmy, metody a regulátory využitě při pokusech v následující kapitole 4.

3.1. Regulace pohybu vozidel

Regulaci pohybu vozidel si můžeme přiblížit pomocí příkladu s řidičem jedoucím na dálnici. Předpokládejme, že nebude přejíždět mezi pruhy.

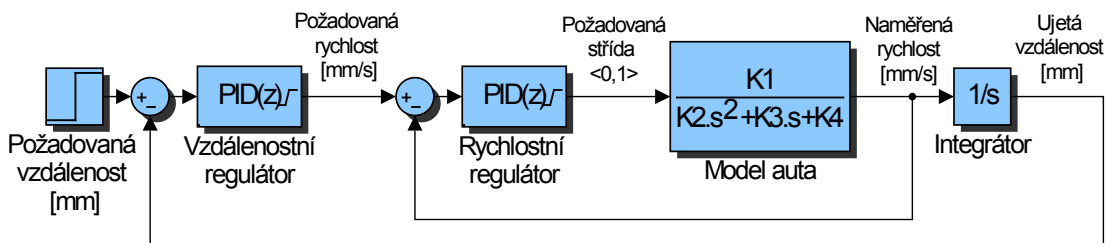
Řidič sleduje zejména dvě věci: rychlost vozidla a vzdálenost k následujícímu dopravnímu prostředku. Rychlost auta ovládá pedálem.

V případě, že před sebou má volnou silnici, rozhodne se zrychlit a s naučeným reflexem sešlápně pedál. Pokud dožene vozidlo, vyhodnotí jeho vzdálenost od sebe a rozhodne se snížit rychlost uvolněním pedálu.

Regulace tedy probíhá v každém okamžiku následovně ve třech krocích: Zjištění vzdálenosti, určení potřebné změny rychlosti, určení tlaku na pedál.

Takto přesně budeme regulovat pohyb vozidla i my.

Regulaci zachycuje kaskádní model zobrazený na schématu 16. Kaskádní proto, že obsahuje dva regulátory zapojené sériově do kaskády.



Obrázek 16. Kaskádní model řízení

Tento kaskádní model se skládá ze tří základních částí: model auta, rychlostní smyčka a vzdálenostní smyčka.

3.1.1. Model auta

Model je pro nás identifikovaná dynamika auta. Ta bude samozřejmě nelineární, pro návrh regulátorů a pro analýzu se nám ovšem hodí linearizovaná ve formě přenosu.

Identifikace a linearizace je uvedena v bakalářské práci [9]. Pro naše účely použijeme přenosovou funkci:

$$H(s) = \frac{93,88 \cdot 10^6}{s^2 + 2503s + 34,72 \cdot 10^3} \tag{7}$$

3. Regulace a teorie řízení kolon vozidel

Vstupem do modelu je střída napětí s rozsahem $\langle 0, 1 \rangle$, výstup je rychlost vozidla. Pro náš příklad s řidičem to odpovídá sešlápnutí pedálu a následné rychlosti jeho auta.

Identifikaci můžeme k příkladu s řidičem přirovnat následovně: u různých aut je výstupní rychlost při stejném sešlápnutí různá, musíme proto nejprve poznat jeho vlastnosti.

Závislost rychlosti na střídě je nelineární. Výslednou rychlost ovlivňuje také napájecí napětí. Pro představu, při napětí 8 V dokáže auto vyvinout rychlost od 200 mm/s, při střídě 0,4 do 2500 mm/s při střídě 1.

3.1.2. Rychlostní smyčka

Rychlostní smyčka v sobě zahrnuje regulátor rychlosti, model auta a zpětnou vazbu. Vstupem do ní je požadovaná rychlost, výstupem je rychlost skutečná, tedy ideálně i naměřená

Regulátor rychlosti nám tedy slouží k udržení rychlosti požadované, jeho vstupem je rychlostní odchylka (tedy rychlost požadovaná mínus rychlost naměřená), výstupem je střída.

V běžném autě je tímto regulátorem řidič, který na základě požadované rychlosti nastaví nutný tlak na pedál.

3.1.3. Vzdálenostní smyčka

Součástí vzdálenostní smyčky je regulátor vzdálenosti, příslušná zpětná vazba a celá rychlostní smyčka. Vstupem je požadovaná vzdálenost, výstupem vzdálenost skutečná.

Regulátor vzdálenosti tedy ve výsledku udržuje požadovanou vzdálenost od následujícího (či předchozího) vozidla. Na vstup přivádíme odchylku vzdáleností (požadovaná vzdálenost mínus vzdálenost ujetá) a jeho výstupem je požadovaná rychlost. Odchylku vzdálenosti je vlastně vzdálenost od následujícího vozidla.

Analogie s řidičem je stejná jako v úvodu sekce 3.1.

3.1.4. Výhody kaskádní regulace

Vidíme tedy, že vzdálenost se reguluje ve dvou krocích. Nejprve se snažíme vzdálenostním regulátorem určit požadovanou rychlost, poté vypočítáme rychlostním regulátorem požadovanou střídu napětí.

Mohli bychom také použít jen jeden regulátor vzdálenosti, jehož výstupem by byla rovnou střída pro motor. Vyvarovali bychom se tak práci při ladění dvou regulátorů a výpočetně pro procesor by to také bylo snazší.

Kaskádní regulátor má ovšem lepší stabilitu, je robustnější a spolehlivější. Můžeme také lépe modifikovat jeho vlastnosti tím, že jeden z regulátorů uděláme agresivnější (rychlostní) a druhý pomalejší a robustnější (vzdálenostní).

3.1.5. Ladění a použité hodnoty kaskádních regulátorů

Kaskádní regulátory musíme vytvořit ve dvou krocích. Nejprve navrhujeme rychlostní regulátor s odpojeným vzdálenostním, následně až připojíme vzdálenostní a naladíme ho pro výsledný přenos.

Model auta je spojitý systém, regulátory pro procesor musí být ovšem navrženy diskrétně. Regulační krok můžeme nastavit libovolně dle možností procesoru a senzorů.

V obou případech použijeme z hlediska nejlepších vlastností PI regulátor. Mohli bychom navrhnout i PID, jeho vlastnosti se ovšem v praxi neosvědčily, jeho výhody nad PI regulátorem byly zanedbatelné.

Obecný diskrétní PI regulátor budeme uvažovat ve tvaru:

$$R(z) = P + I \cdot T_s \frac{1}{z - 1} \quad (8)$$

kde P je proporcionální složka, I je integrální složka a T_s je perioda vzorkování, tedy perioda provádění výpočtů regulátoru.

Další důležité vlastnosti jsou anti-windup a saturace regulátorů.

Rychlostní regulátor

Rychlostní regulátor by vyladěn s ohledem na rychlost a co nejmenší překmit. Jeho návrh byl komplikovaný zejména vysokým statickým třením při rozjíždění a nízkých rychlostech autíček.

O jeho návrh se postaral Martin Lád v práci [9]. Tření při nízkých rychlostech vyřešil podmíněným dopředným regulátorem, tím se zde však zabývat nebudu.

Výsledné hodnoty regulátoru jsou:

- Proporcionální složka $P = 0,002$
- Integrální složka $I = 0,01$
- Vzorkování $T_s = 0,001$
- Možné výstupní hodnoty v rozsahu $\langle -1, 1 \rangle$ (saturace)

Vzdálenostní regulátor

Regulátor jsem nejprve navrhl s pomocí MATLABovské funkce PID Tuner, poté jsem ho vyzkoušel a popřípadě pozměnil ve vlastní implementaci.

Regulace vzdálenosti by neměla probíhat tak často, jako regulace rychlosti. Je potřeba umožnit autům, aby stihla na požadovanou rychlost vzdálenostního regulátoru zareagovat.

Dále je zde spíše požadavek na stabilitu, než na rychlost. S ohledem na tyto skutečnosti jsme zvolili následující hodnoty:

- Proporcionální složka $P = 10$
- Integrální složka $I = 2$
- Vzorkování $T_s = 0,05$
- Možné výstupní hodnoty v rozsahu $\langle -2500, 2500 \rangle$ (maximální povolené rychlosti)

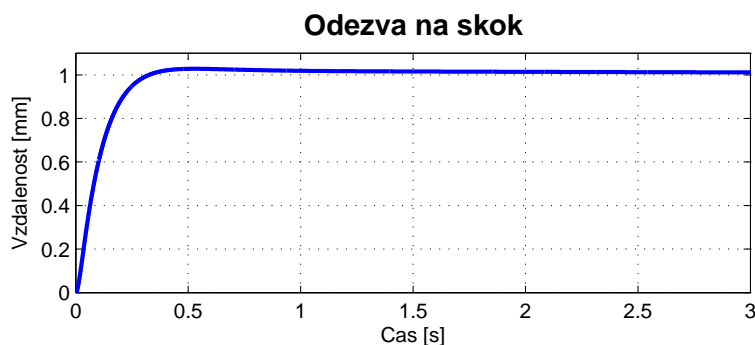
Další výraznou vlastností je tzv. Dead zone. Ten drží vstup (vzdálenost) do regulátoru na stejné hodnotě, dokud se nezmění o 5 mm. Tím zabráníme popojíždění vozidel kvůli chybě měření.

Takto nastavený vzdálenostní regulátor je kompromisem mezi stabilitou a rychlostí. Agresivnější regulátory fungovaly dobře při vyšších rychlostech (>1000 mm/s) a pro méně aut v koloně, při nižších rychlostech docházelo ke srážkám díky příliš silným reakcím a velkému překmitu. Naopak pomalejší regulátor zvládá skvěle nízké rychlosti, ale jeho reakce jsou pomalé a trvá mu velmi dlouho, než zrychlí na vysokou rychlost.

Tento regulátor jsem vyzkoušel pro kolonu až deseti vozidel při rychlostech mezi 200 až 1500 mm/s.

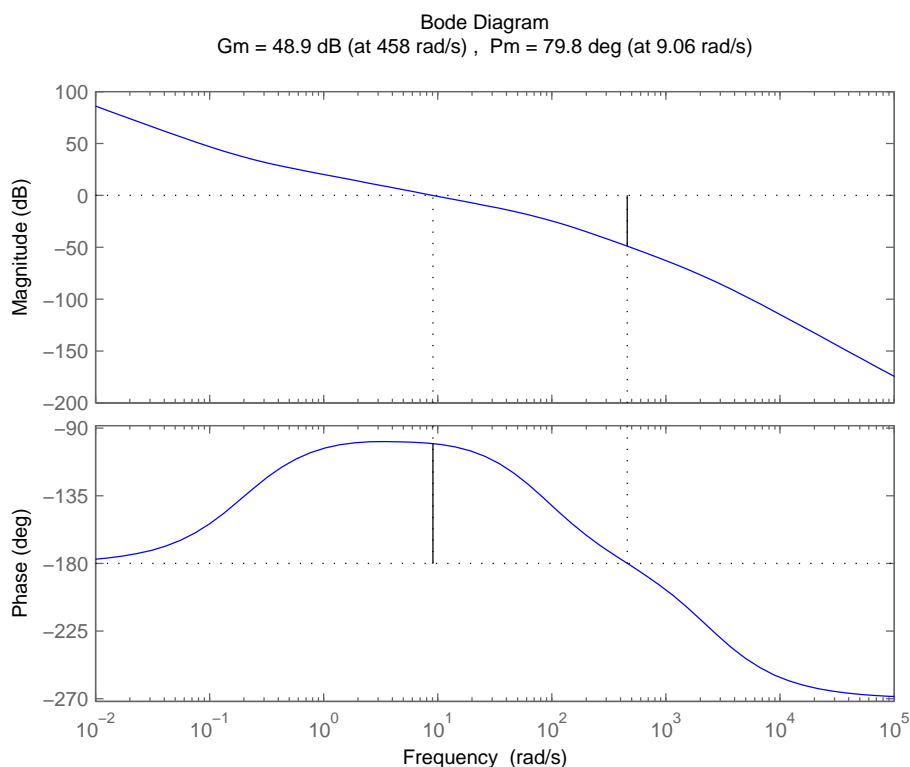
Analýza kaskádního regulátoru

Pro zajímavost umístuji některé vlastnosti vzdálenostní smyčky. Na grafu 17 vidíme odezvu na skok uzavřené smyčky. Snažil jsem se o co nejmenší překmit, který je 2,22%.



Obrázek 17. Odezva na skok uzavřené vzdálenostní smyčky

Graf 18 zobrazuje vyznačenou amplitudovou bezpečnost (*gain margin*), tedy rezervu v zesílení, a fázovou bezpečnost (*phase margin*), která udává bezpečnou velikost zpoždění. Tentokrát počítáme Bodeho diagram otevřené vzdálenostní smyčky.



Obrázek 18. Bodeho diagram otevřené vzdálenostní smyčky s vyznačenou fázovou (Pm) a amplitudovou (Gm) bezpečností

Nakonec jsem ještě vypočítal póly uzavřené smyčky. Jedná se o systém 5. řádu, všechny póly jsou reálné.

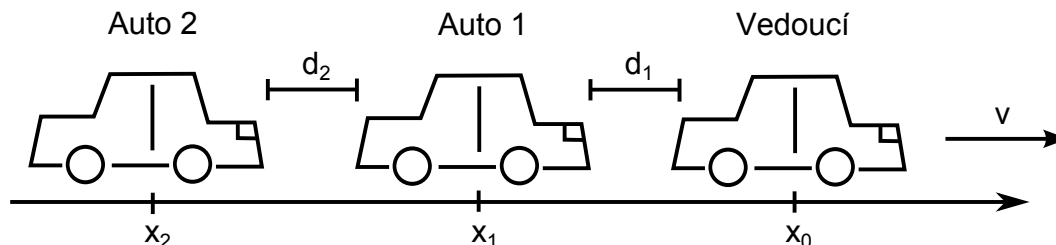
Zaokrouhlené póly: -0,02; -0,61; -0,81; -7,74; -2411.

3.2. Teorie řízení kolon

O teorii řízení automobilových kolon bylo napsáno mnoho, například zde [10]. Rád bych tuto práci soustředil zejména na implementaci, zmíním tedy pouze základní způsoby pro přehled a objasnění pokusů v následující kapitole 4.

V této práci se budeme zabývat pouze metodami pro řízení kolon vozidel, které spolu nikterak nekomunikují.

Příklad takovéto kolony s použitým značením je na obrázku 19.



Obrázek 19. Kolona s vedoucím vozidlem a dále používaným indexováním vzdáleností a aut

Vedoucí vozidlo jede rychlostí v zcela nezávisle na N autech za ním. Každé vozidlo se nachází ve vzdálenosti x_j od vedoucího vozidla, $j \in 0, 1, \dots, N$. Vzdálenost mezi sousedními auty je d_i , $i \in 1, 2, \dots, N$.

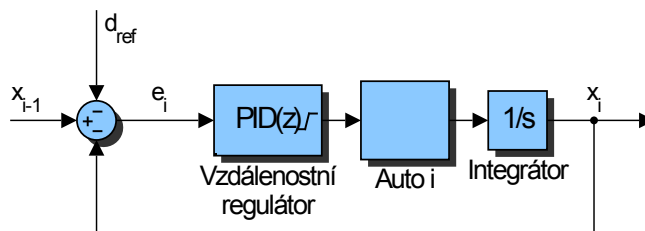
Pro vzdálenost auta i platí:

$$d_i = x_{i-1} - x_i \quad (9)$$

Auta $1, 2, \dots, N$ se snaží vedoucí auto následovat s nastavenou referenční vzdáleností.

3.2.1. Dopředné řízení (*predecessor following*)

Nejpoužívanější a i řídiči nejbližší je metoda dopředného řízení, kde řídíme vzdálenost pouze k vozidlu před námi. Jednoduché schéma takového řízení je na obrázku 20, kde máme skrytou rychlostní smyčku. Zajímá nás zejména odchylka e_i , tedy vstup do regulátoru vzdálenosti.



Obrázek 20. Schéma dopředně řízeného vozidla zařazeného v koloně

Regulační odchylku dopředného řízení popisuje rovnice:

$$e_i = x_{i-1} - x_i - d_{ref} = d_i - d_{ref} \quad (10)$$

3.2.2. Řízení k následujícímu vozidlu (*follower control*)

Narozdíl od dopředného měření zde regulujeme vzdálenost pouze k následujícímu vozidlu. Regulační odchylka je:

$$e_i = x_i - x_{i+1} - d_{ref} = d_{i+1} - d_{ref} \quad (11)$$

Pro řízení naší kolony tento způsob nevyužijeme, neboť potřebujeme řídit kolonu vedoucím vozidlem. Při řízení k předchůdci by ovšem vedoucí vozidlo muselo být umístěno vzadu kolony.

3.2.3. Obousměrné řízení (*bidirectional control*)

Obousměrné řízení využívá jak dopředného řízení k předchozímu, tak řízení k následujícímu vozidlu. Regulační odchylku vypočteme dle následujícího vztahu:

$$e_i = (x_{i-1} - x_i - d_{ref}) - (x_i - x_{i+1} - d_{ref}) = (x_{i-1} - x_i) - (x_i - x_{i+1}) = \quad (12)$$

$$= d_i - d_{i+1} \quad (13)$$

Zajímavostí je, že u všech vozidel kromě vedoucího a posledního se nám odečte referenční vzdálenost. Jejich cílem je tedy udržet stejnou vzdálenost k vozidlu vpřed a k vozidlu vzad.

Poslední vozidlo kolony se nemůže řídit dle předchůdce, využijeme tedy pouze dopředného řízení.

3.2.4. Obousměrné vážené řízení (*asymmetric bidirectional control*)

Jedná se o speciální případ obousměrného řízení, pouze vážíme účinek měření vzdálenosti k následujícímu a k předchozímu vozidlu. Regulační odchylku si vypočteme následujícím vztahem:

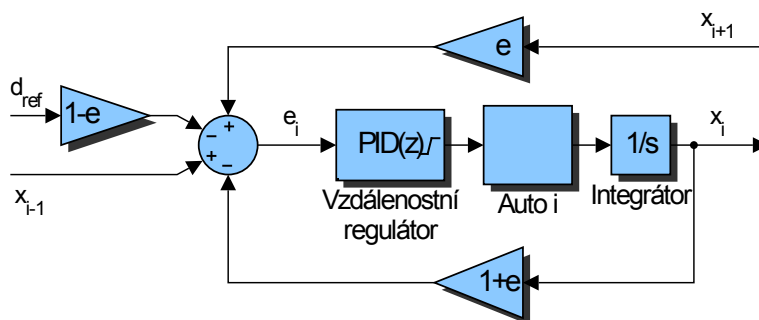
$$e_i = (x_{i-1} - x_i - d_{ref}) - \epsilon(x_i - x_{i+1} - d_{ref}) = \quad (14)$$

$$= x_{i-1} - x_i(1 + \epsilon) - d_{ref}(1 - \epsilon) + \epsilon \cdot x_{i+1} \quad (15)$$

$$= d_i - d_{i+1} - d_{ref}(1 - \epsilon) \quad (16)$$

Váha ϵ může teoreticky nabývat hodnot v intervalu $\langle 0, \infty \rangle$, kde pro $\epsilon = 0$ se jedná o dopředné řízení a pro $\epsilon = 1$ se jedná o řízení obousměrné.

Dle vzorců pro výpočet regulační odchylky můžeme sestavit schéma 21 pro vozidlo i .



Obrázek 21. Schéma obousměrně váženě řízeného vozidla zařazeného v koloně

Obousměrné vážené řízení má na rozdíl od dopředného výhodu v tom, že auta berou ohled na stav vozidel za sebou. Díky váze ϵ navíc můžeme určit, jak moc na nich mají záviset. Takto řízené kolony mají teoreticky v některých případech lepší chování, jak ukáží v pokusech kapitoly 4.

4. Pokusy s kolonou vozidel

Tato kapitola obsahuje výstupy z pokusů s kolonou vozidel. Jedná se vlastně o prezentaci dosavadního výsledku celého projektu.

Cílem projektu je vytvořit kolonu vozidel, na kterých se budou moci vyzkoušet různé algoritmy řízení a různé regulátory. Proto bych spíše než na navržení vhodného regulátoru kladl důraz na funkčnost celé platformy.



Obrázek 22. Kolona autodráhových vozidel

4.1. Škálování se skutečným světem a parametry měření

Před samotnými pokusy je nutné se nejprve zamyslet nad měřítkem vozidel. V našem projektu používáme autodráhová vozidla od společnosti Carrera. Přesněji se jedná o model Ford Capri provedený v měřítku 1:32. Chtěl bych nyní obhájit parametry provedených pokusů, k čemuž využiji porovnání našeho zmenšeného modelu kolony s kolonou v reálné velikosti.

Změřil jsem si například, že délka modelu auta je 13 centimetrů. Z měřítka mohu určit délku skutečného auta jednoduše vynásobením délky modelu $13 \cdot 32$, tedy skutečné vozidlo je dlouhé něco přes 4 metry.

V následujících tabulkách takto porovnáám pár zásadních hodnot pro naše měření.

V tabulce 1 jsou nejdůležitější parametry měření vzdálenosti, které vyplývají z vlastností senzoru. Jedná se o minimální a maximální možné hodnoty měření a zvolenou referenční vzdálenost.

4. Pokusy s kolonou vozidel

	Model [mm]	Skutečnost [m]
Minimum	50	1.6
Reference	150	4.8
Maximum	300	9.6

Tabulka 1. Důležité parametry měřené vzdálenosti

Hodnotu referenční vzdálenosti 15 centimetrů jsem zvolil s ohledem na rozsah možného měření.

Skutečnost [km/h]	Model [mm/s]
10	87
20	174
50	434
100	868
200	1736

Tabulka 2. Porovnání rychlostí modelu a skutečnosti (přibližné hodnoty)

Z tabulky 2 vyčteme, že rychlosti autodráhových vozidel ve skutečnosti přesahují rychlostní možnosti kdejakého závodního vozidla.

Pro parametry měření je však důležité porovnání rychlostí a referenční vzdálenosti. Pokud bychom například učinili pokus, ve kterém by auta měla následovat vedoucí vozidlo jedoucí rychlostí 2 m/s ve zvolené referenční vzdálenosti 15 cm, odpovídalo by to ve skutečnosti koloně, která by se snažila následovat auto jedoucí přes 200 km/h s rozestupem mezi vozidly 4,8 m!

To jsou samozřejmě již extrémní hodnoty. Dle útvaru Ministerstva dopravy BE-SIP [11] je bezpečný rozestup mezi vozidly roven vzdálenosti, kterou auto ujede za 2 sekundy při dané rychlosti. To by pro naši referenční vzdálenost znamenalo rychlost skutečného vozidla přibližně 10 km/h a tomu odpovídající modelovou 87 mm/s.

To je ovšem nereálné, neboť motor autodráhových vozidel se při malých rychlostech potýká s velkým statickým třením. Pro pokusy jsem proto zvolil rychlosti mezi 500 mm/s a 1000 mm/s, tedy tomu odpovídající skutečné rychlosti přibližně 60 km/h a 120 km/h.

Tyto úvahy zde uvádím zejména pro uvědomění si, že člověk by ani nebyl sto v tak vysokých rychlostech a malých rozestupech mezi vozidly rychle reagovat na změny a nenabourat. Dobře navržený regulátor by toho ovšem měl být schopen.

4.2. Experimenty a pokusná měření

Nyní bych rád uvedl výsledky některých měření. Připravil jsem deset aut, dostatečně velkou dráhu, program s pokusy a vyřešil jsem hromadné ukládání naměřených dat a jejich stažení do MATLABu.

Chtěl bych uvést dva základní experimenty používané v teorii řízení: odezvu na skok a na rampu. Měříme a regulujeme dle vzdálenosti, takže se bude jednat o skok s určitou vzdáleností a o rampu vzdálenosti, tedy o rychlost.

Pro následující pokusy jsem použil hodnoty regulátorů uvedené v sekci (3.1.5). Jak jsem již napsal, použitý regulátor vzdálenosti je navržen co nejuniverzálněji.

Pokusy jsem provedl se šesti nebo sedmi z deseti sestavených aut, jejichž jízdní vlastnosti byly nejlepší. Některá vozidla jsme totiž postavili ze zbytků minulého projektu

a měla vysoké tření či neměřila správně rychlost. Navíc bylo potřeba nechat nějaké vozidlo jako záložní a některá auta byla rozpůjčována kolegyně.

Pro skok i rampu uvedu vždy dopředné a pro porovnání i obousměrné vážené řízení. Pro skok jsem se rozhodl uvést i čisté obousměrné řízení, ačkoli bylo nestabilní a kmitalo.

Uvedu vždy teoretický průběh a skutečné naměřené hodnoty. Pro simulace jsem použil linearizovanou přenosovou funkci v sekci 3.1.1.

Přesnost porovnání teoretického průběhu s naměřeným ovšem velmi závisí na kvalitě identifikace modelu. Ta je velmi náročná díky nelinearitám, tření a možnostem motoru. Hodnoty teoretického průběhu jsou tudíž pouze orientační, rozhodně neplatí, že regulátory pracující výborně v teorii musí stejně dobře pracovat i ve skutečnosti.

Další velká odlišnost od ideálního průběhu je chyba měření jak rychlosti, tak vzdálenosti. Snažili jsme se maximálně zpřesnit tyto hodnoty, avšak stále zůstává drobná odchylka od skutečnosti. V případě měření vzdálenosti se pohybuje kolem 5 procent.

Vzdálenost jsme schopni měřit pouze v rozsahu 5 až 30 centimetrů. Tento jev jsme se to v počítačové simulaci zohlednit, nicméně jeho výskyt je vysoce nežádoucí. Proto jsem se snažil pokusy navrhnout tak, aby se měřená vzdálenost pohybovala vždy v tomto rozmezí.

Co se stane v případě přesažení maximální měřitelné vzdálenosti mezi auty? Do regulátoru vstupuje regulační odchylka, tedy měřená vzdálenost mínus referenční. Čím vyšší je tento rozdíl, tím vyšší bude reakce regulátoru. Díky omezení v měření ale vznikne i omezení v reakci regulátoru, kdy je akční zásah nižší, než by teoreticky měl být.

Tato nevýhoda je obzvláště patrná při obousměrném řízení, kde záleží na rozdílu měření vpřed a vzad. V nejhorsím případě, kdy bude vzdálenost nad maximální hodnotou v obou směrech, bude vstup do regulátoru nulový a auto se zastaví.

V pokusech se mi obousměrné řízení neosvědčilo, auta se z neznámého důvodu rozkmitaly. Zobrazil jsem proto jen odezvu na skok a rozhodl jsem se uvést obousměrné vážené řízení, které fungovalo rozumně.

Ukládal a vykreslil jsem dvě následující hodnoty: vzdálenost k předchozímu vozidlu a rychlost vozidla.

Pro názornost by bylo dobré uvést i vzdálenost, kterou vozidla ujedou. Tuto hodnotu jsem počítal z rychlosti, ale ukázala se jako velice nepřesná, proto jsem se rozhodl ji neuvádět.

Poznámka ke grafické úpravě: formátování následujících stránek jsem zvolil pro maximální přehlednost tištěné verze.

4.2.1. Reakce kolony na skok

Nejdříve bych rád uvedl reakci kolony na náhlou změnu vzdálenosti vedoucího vozidla, tedy skok.

Měření jsem provedl následovně: 7 aut jsem rozmístil v referenční vzdálenosti 150 milimetrů od sebe. Poté jsem první vozidlo (vedoucí) posunul o dalších 75 nebo 150 milimetrů vpřed. Zahájil jsem pokus v čase 1 sekunda a sledoval 6 následujících vozidel, jak se snaží srovnat za vedoucím vozidlem do referenční vzdálenosti.

Dopředné řízení

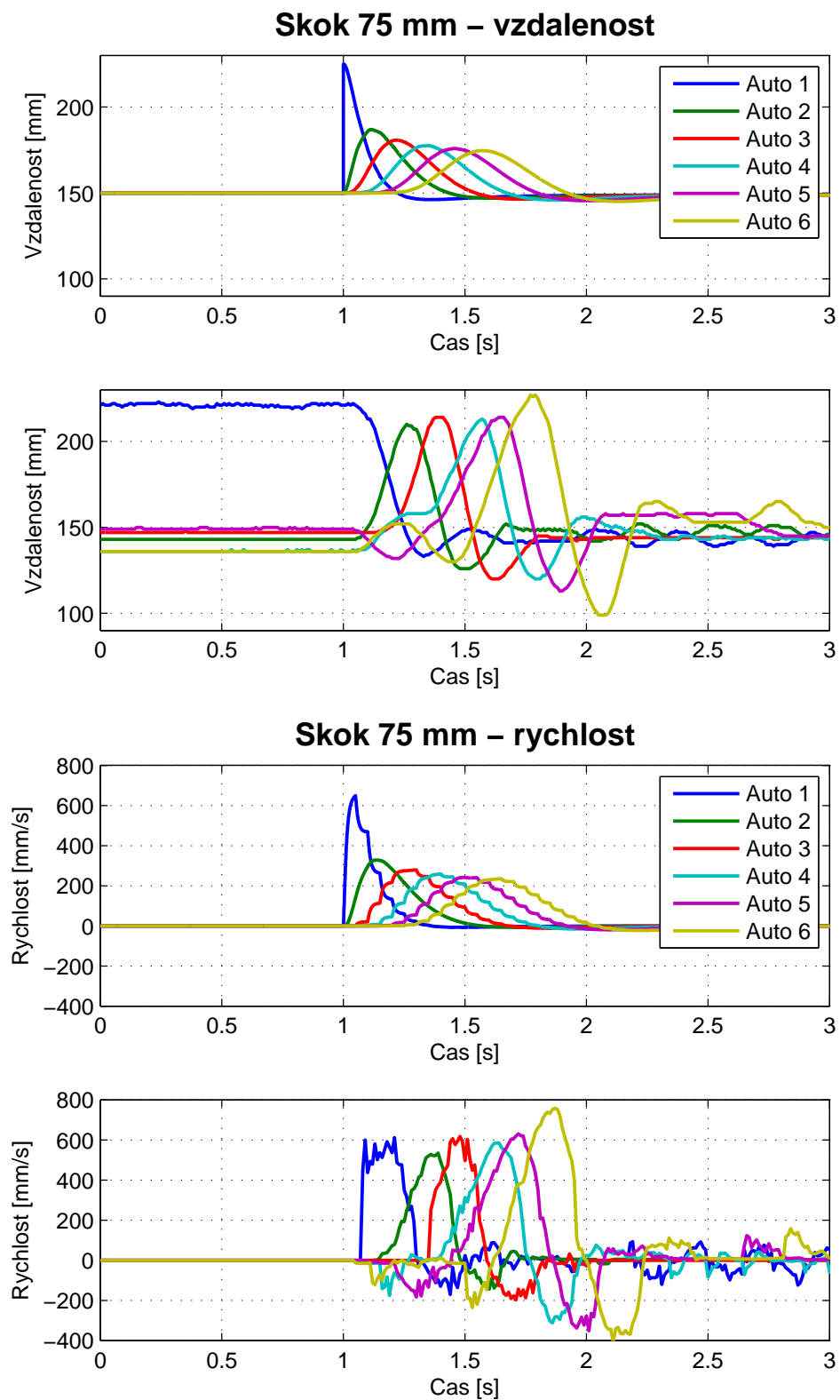
U dopředného řízení jsem nastavil skok na 75 milimetrů. Byla to maximální hodnota, při které ještě nedošlo k srážce vozidel kvůli překmitům.

V grafech 23 vidíme porovnání teoretického a pokusného průběhu pro měřenou rychlost a vzdálenost k předchozímu vozidlu. Zatímco simulovaný průběh je stabilní a bez překmitu, skutečný díky zpoždění a rozdílné dynamice překmit má.

Všimněme si zejména, že překmit se u každého dalšího auta mění - postupně se šíří a zvyšuje. První vozidlo dorazilo k referenční vzdálenosti od vedoucího, ale nestihlo kvůli agresivitě regulátoru zastavit, tak se muselo vracet. Zatím druhé vozidlo vyrazilo za prvním, také přejelo, ale musí nyní vyrovnat překmity dva - svůj a překmit prvního vozidla, které právě couvá. Tento úkaz se postupně zesiluje a kdybychom měli v koloně více aut, došlo by pravděpodobně ke srážce.

Tento jev se nazývá stringová nestabilita ([12]) a cílem řízení kolon je ho odstranit, nebo alespoň co nejvíce potlačit.

V teoretickém průběhu se také u rychlosti vyskytují drobné zuby, vlnky. Je to dáno frekvencí vzdálenostního regulátoru, který je aplikován dvacetkrát za sekundu. V praxi totiž potřebuje rychlostní regulátor, který je volán častěji, čas na zareagování.

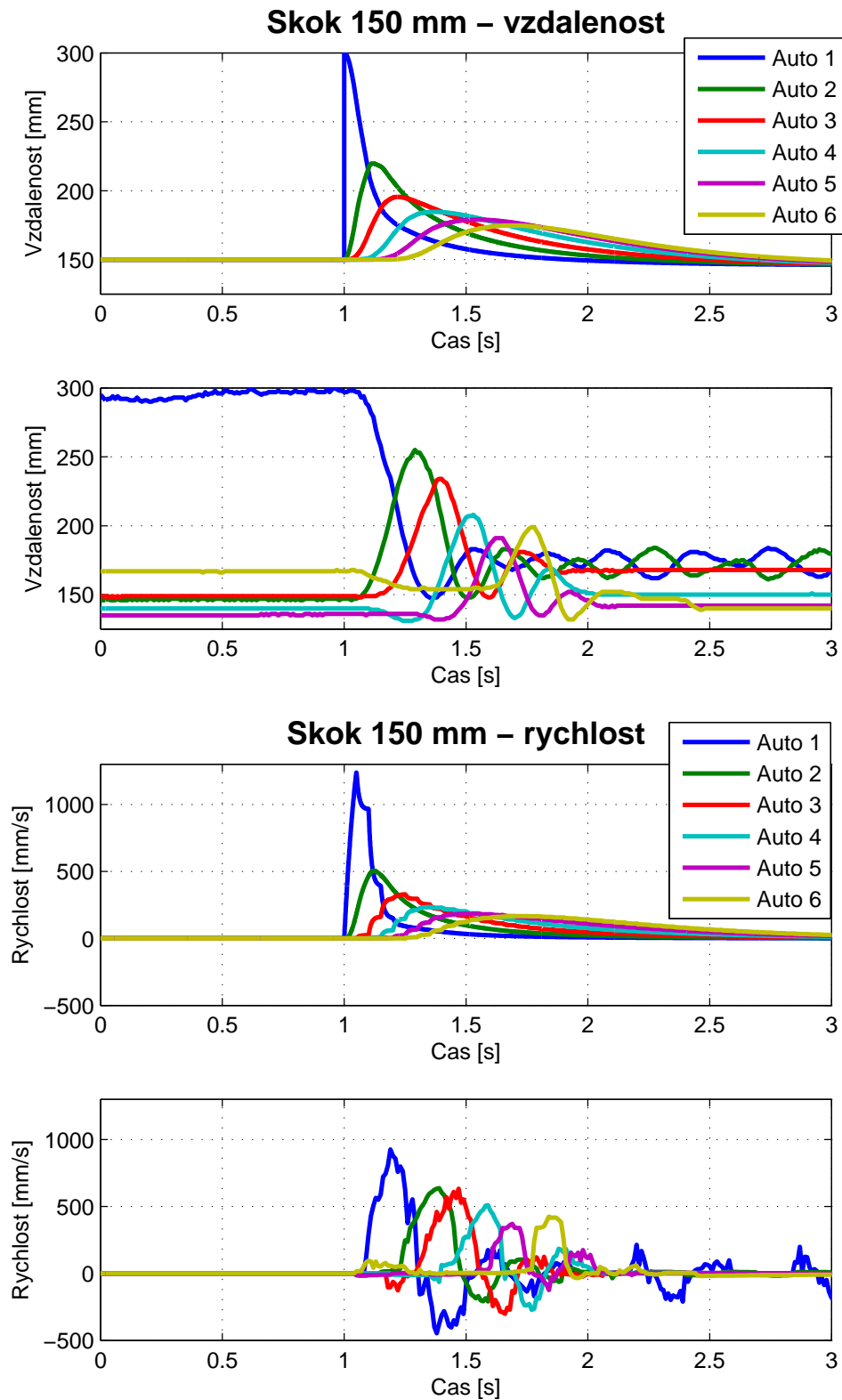


Obrázek 23. Dopředné řízení, reakce kolony na skok 75 mm, grafy zachycují měřenou rychlost vozidel a vzdálenost k předchozímu vozidlu; nahoře teoretický průběh, dole změřený

Obousměrné vážené řízení s váhou 0,5

Z grafů 24 lze pozorovat výhodu obousměrného řízení. Tím, že bereme v úvahu i vzdálenost následujícího vozidla v koloně, zmenšíme překmit. Na druhou stranu tím ale zpomalíme celkovou rychlost reakce kolony na změnu.

Pro své pokusy jsem nastavil maximální možný měřitelný skok 150 milimetrů, neboť u menších hodnot nebylo ve výsledku nic vidět a kolona se téměř ani nepohnula. Při takto vysokém skoku by při dopředném řízení došlo ke srážce aut, obousměrné řízení zafungovalo velice dobře. Naopak pro malý skok by reakce obousměrného řízení byla nedostatečná, alespoň v praxi, kde naše auta nejsou schopna velmi malých rychlostí.



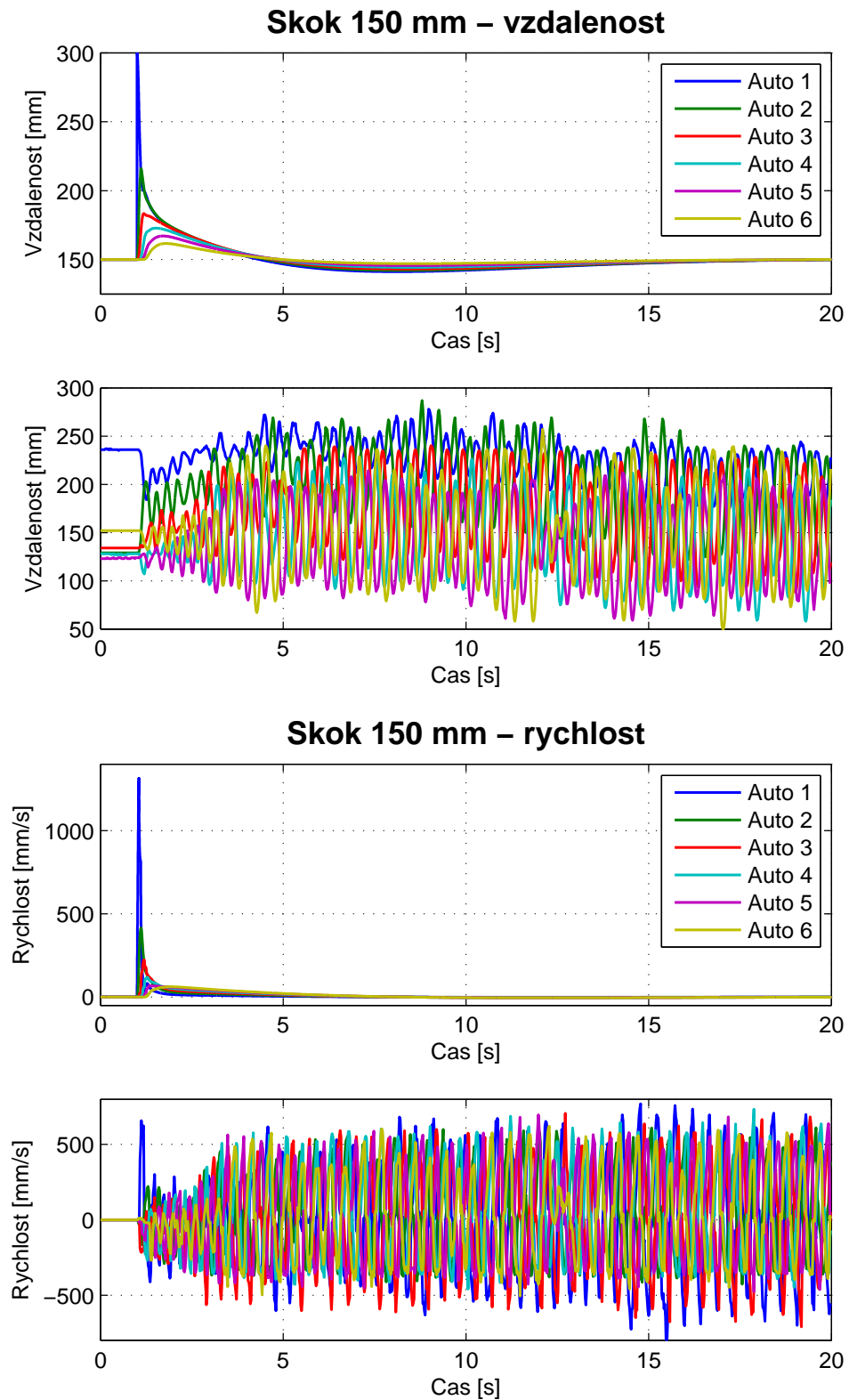
Obrázek 24. Obousměrné vážené řízení s váhou 0,5, reakce kolony na skok 150 mm, grafy zachycují měřenou rychlost vozidel a vzdálenost k předchozímu vozidlu; nahore teoretický průběh, dole změřený

Obousměrné řízení

Obousměrné vážené řízení se mi bohužel nepodařilo stabilizovat. Vždy po zahájení pokusu se auta rozkmitala, jako by k sobě byla připevněna pružinou. Naměřené hodnoty jsem vynesl do grafů 25.

Snažil jsem se najít příčinu tohoto kmitání, ale nebyl jsem úspěšný. Kontroloval jsem zejména měření vzdálenosti, ale vše bylo v pořádku.

Teoreticky by kmitání mohlo vznikat neschopností autíček vyvinou velmi malé rychlosti (menší než 100 mm/s). Tím nejsou s to dorovnávat drobné vzdálenostní rozdíly a vždy přejedou požadovanou vzdálenost. Tento problém se nejspíš bude muset vyřešit v implementaci regulátoru.



Obrázek 25. Obousměrné řízení, reakce kolony na skok 150 mm, grafy zachycují měřenou rychlost vozidel a vzdálenost k předchozímu vozidlu; nahoře teoretický průběh, dole změřený

4.2.2. Reakce kolony na rampu

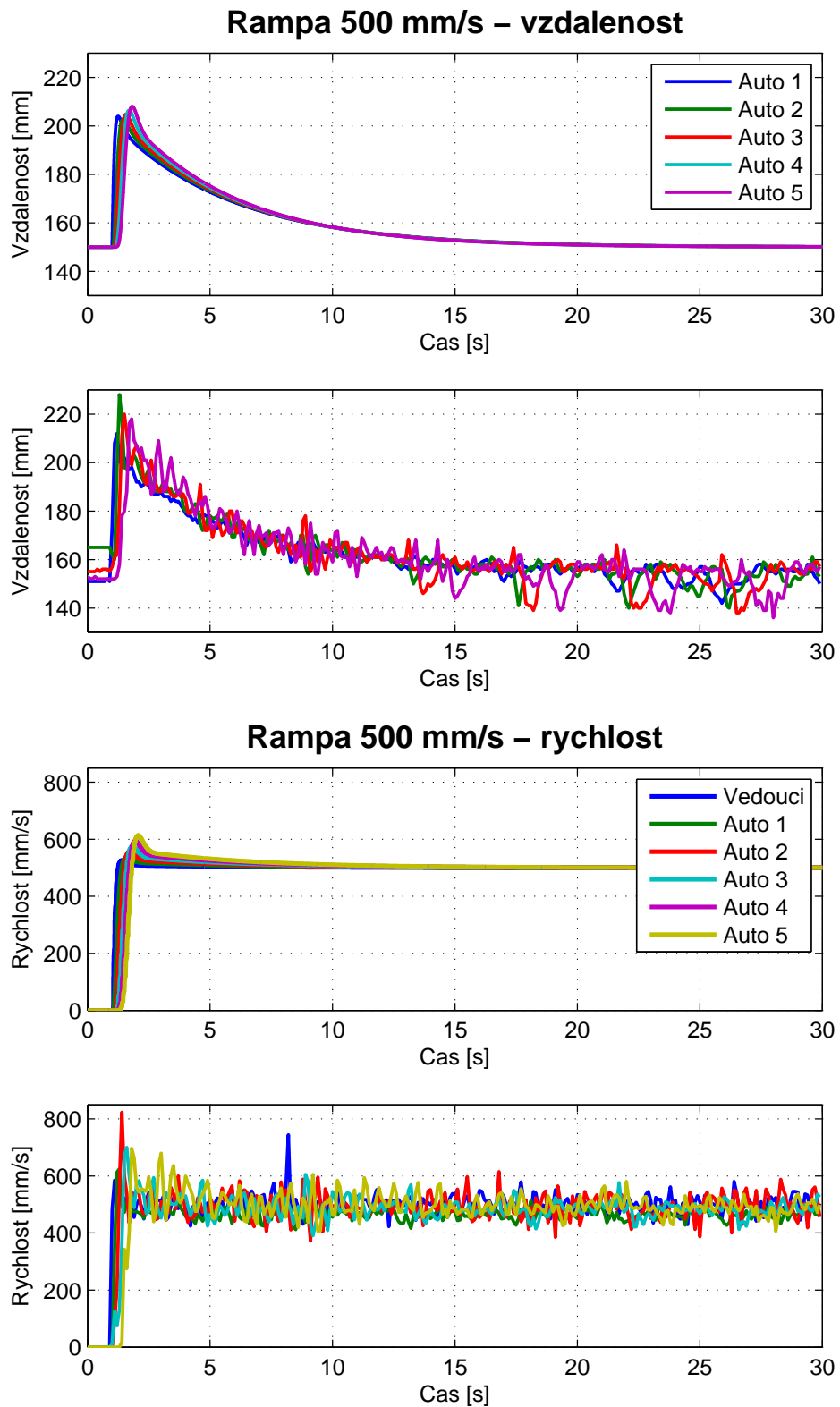
Nyní bych rád uvedl schopnost kolony sledovat jedoucí vozidlo konstantní rychlostí, tedy rampu vzdálenosti.

Měření jsem provedl následovně: 6 vozidel jsem rozmístil v referenční vzdálenosti 150 milimetrů od sebe. V čase 1 sekunda jsem vedoucímu vozidlu nastavil referenční rychlost 500 milimetrů za sekundu a sledoval pohyb kolony.

Dopředné řízení

Při porovnání teoretického a pokusného průběhu z grafů 26 si všimneme zejména rozkmitanosti skutečných průběhů. Zde je to dáno zejména nelinearitou skutečnosti, kde je například trochu rozdílné tření v každém místě kulaté dráhy a podobně. Možná je i chyba měření.

Při bližším prozkoumání ale zjistíme, že střední hodnoty průběhů se velice blíží těm teoretickým. Doba, kterou kaskádnímu regulátoru zabere dohnání rampy (asi 15 sekund) je dána agresivitou regulátoru. Při vyšší P a I složce by dohnání rampy trvalo kratší dobu, ale byla by tím ohrožena stabilita při poruchách.



Obrázek 26. Dopředné řízení, reakce kolony na rampu 500 mm/s, grafy zachycují měřenou rychlost vozidel a vzdálenost k předchozímu vozidlu; nahoře teoretický průběh, dole změřený

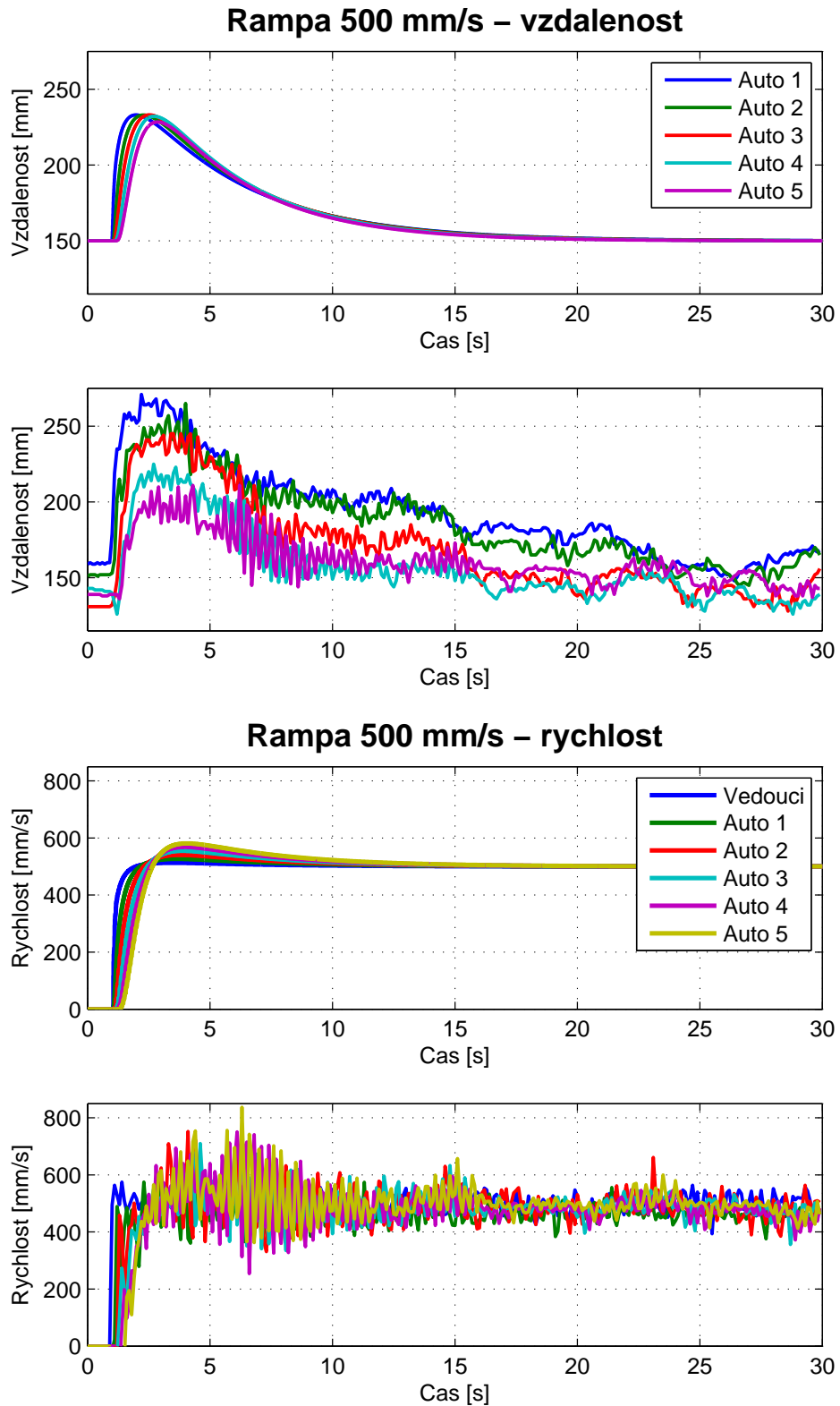
Obousměrné vážené řízení s váhou 0,5

Obousměrné řízení z grafů 27 na rozdíl od dopředného dává na první pohled horší výsledky. Mnohem více oscilují kolem střední hodnoty. Tohoto jevu jsem si všiml zejména u posledních dvou vozidel.

Důvod může být jak v nerovnostech, tření autodráhy či nelinearitě motoru, tak v nepřesnosti měření vzdáleností vpřed a vzad, kdy jedna hodnota se mírně liší od druhé.

Poslední vozidlo je řízeno čistě dle předchozího. Předchozí se však řídí jak podle svého předchůdce, tak i s poloviční intenzitou (váha 0,5) dle posledního vozidla. Pokud se jen mírně liší měřená vzdálenost těchto dvou vozidel mezi sebou, budou se „přetahovat“. Zejména poslední vozidlo pak při rozkmitání předposledního tento jev ještě zesílí.

Toto kmitání naštěstí postupně ustává. Střední hodnota průběhu je opět velmi podobná tomu simulovanému.



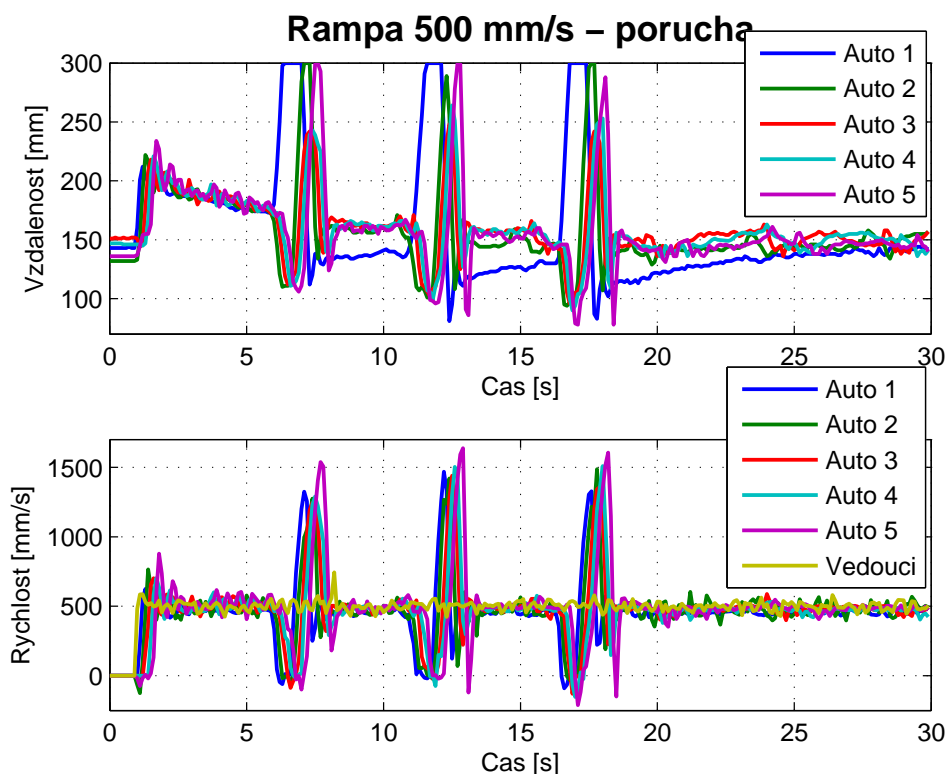
Obrázek 27. Obousměrné vážené řízení s váhou 0,5, reakce kolony na rampu 500 mm/s, grafy zachycují měřenou rychlost vozidel a vzdálenost k předchozímu vozidlu; nahore teoretický průběh, dole změřený

4.2.3. Reakce kolony na poruchu

Nakonec ještě uvedu pokus, ve kterém jsem uměle vyvolal v koloně poruchu. Nastavil jsem dopředné řízení a vedoucí vozidlo jsem nechal jet rychlostí 500 milimetrů za sekundu. Poruchu jsem způsobil krátkým zastavením vozidla, které následuje vedoucího. Průběh je zachycen na grafu 28.

Celkem jsem takto během pokusu kolonu vyrušil třikrát, poprvé přibližně v šesté, podruhé v jedenácté a potřetí v šestnácté sekundě. Reakce kolony byla pokaždé podobná.

Zastavené auto (modrá barva) postupně ztratilo z dohledu vedoucí vozidlo. Zbytek kolony zastavil přibližně 100 milimetrů za ním. Po opětovném rozjetí během necelé sekundy dohonila kolona vedoucí vozidlo, došlo k překmitu a následuje postupné dohnání na referenční vzdálenost.



Obrázek 28. Dopředné řízení, rampa 500 mm/s, celkem vytvořeny tři poruchy (6, 11 a 16 sekunda), grafy zachycují měřenou vzdálenost a rychlost vozidel

4.2.4. Diskuze výsledků

Pro lepší představu jsem nahrál kolonu sledující vedoucí vozidlo, kterému jsem změnil rychlost. Video jsem přiložil do materiálů na CD a na web [1].

Výsledky provedených pokusů jsou dobré. Mnou navržený vzdálenostní regulátor při dopředném a obousměrném váženém řízení pracoval dle předpokladů.

Obousměrné vážené řízení se mi bohužel kvůli kmitání nepodařilo zprovoznit.

Co je ale mnohem důležitější, prokázali jsme funkčnost celého konceptu. Při dopředném řízení auta jezdí dle mého subjektivního názoru více než dobře. Nyní nic nebrání sestavení kolony o více vozidlech.

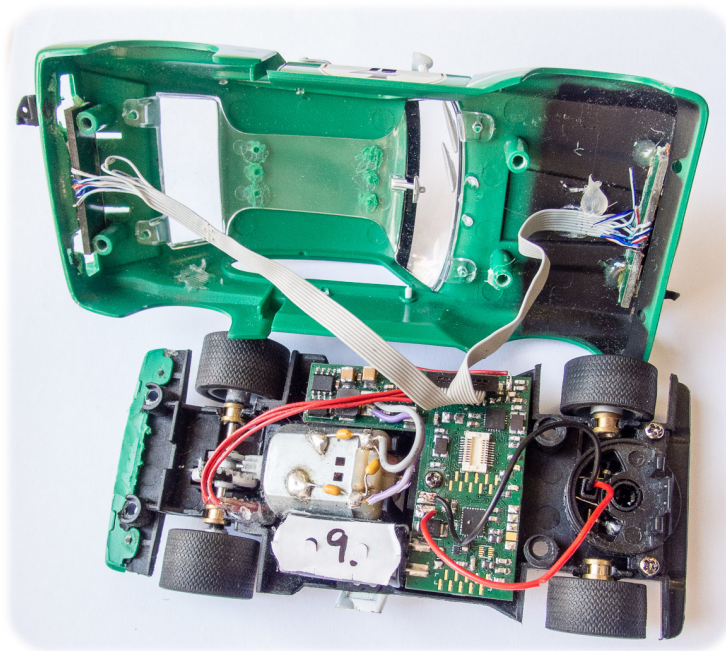
5. Další činnosti, návrhy na vylepšení a na co nezbyl čas

V této kapitole bych rád krátce čtenáře seznámil s dalšími časově náročnými činnostmi, které jsem v rámci bakalářské práce vykonal, ale které svým obsahem nezapadají do zbylých částí.

Také bych uvedl některé náměty, které by mohly v budoucnu projektu prospět. Nakonec bych se zmínil o cílech, na které nezbyl čas.

5.1. Mechanické sestavení vozidel

S prvním funkčním prototypem senzoru pro měření vzdálenosti jsem uzavřel dlouhou a zejména teoretickou část bakalářské práce. Druhá velká kapitola se poté týkala chování kolon aut. Důležitým krokem mezi těmito kapitolami bylo ovšem mechanické sestavení dostatečného počtu senzorů a samotných vozidel.



Obrázek 29. Výsledné upravené auto

Základní deska plošného spoje s procesorem nám byla dodána již plně zhotovená, upravili jsme tedy jen auta a připevnili ji na podvozek. Bylo nutné jen připájet vstup z dráhy a výstup na motůrek auta. Dále jsme museli do všech aut postupně nahrát kód a vykoušet funkčnost. U počtu deseti aut se často vyskytovaly chyby, které práci velmi prodlužovaly.

Deska plošného spoje pro fototranzistor a LED diodu nám byla navržena, mým úkolem bylo ji osadit namontovat pod kapotu vozidla tak, aby se celá skryla a jen optoelektronika měla volný výhled otvory pro světla autíčka.

Pro celkový počet deseti aut bylo potřeba osadit a vyzkoušet dvacet senzorových desek. Auta i kapotu bylo pro montáž nutné mechanicky upravit - upilovat překážející části, rozšířit otvory pro LED a fototranzistor atd.

5.2. Udržení soudržnosti projektu

Do projektu je zapojena spousta lidí, jejichž práce se postupně prolíná. Je potřeba sjednocovat programový kód, sdílet veškeré materiály pro software i hardware vozidel, zveřejňovat instalační instrukce, návody a postupy.

5.2.1. GIT

Pro sdílení souborů k projektu jsme využívali freewarový program GIT [13]. Jedná se o open source aplikaci, která zajišťuje distribuovaný systém správy verzí. Původně byl GIT vyvinut pro vývoj jádra operačního systému Linux.

Git podporuje všechny základní operační systémy (Linux, Windows, Mac OS, Solaris) a nabízí rozsáhlé možnosti nastavení.

V projektu ho využíváme zejména pro zálohování, správu a sjednocování kódu, schémat, knihoven atd.

5.2.2. Návody, postupy, problémy a řešení

V rámci projektu jsme si založili webovou stránku ve stylu Wikipedie podpůrného webu Katedry řídicí techniky [14], kam jsme postupně doplňovali veškeré postupy, návody, nalezené problémy a jejich řešení.

Tyto informace zajišťují manuál pro nově příchozí, stránka je v rámci mezinárodnosti vedena v angličtině.

5.2.3. Dokumentace kódu

Pro pochopení kódu je nutné psát podrobné komentáře. Rozhodl jsem se využít těchto komentářů a rovnou je psát v syntaxi nástroje Doxygen [15].

Doxygen je oblíbený a často využívaný nástroj pro generování dokumentace z komentovaného programového kódu. Podporuje řadu programovacích jazyků (C, C++, C#, PHP, Java, Python, IDL, ...). Výstupní dokumentace může být v HTML či v LaTeXu.

Samotný kód píše v IDE Eclipse [16], které umožňuje díky pluginu vytvářet dokumentaci přímo ve svém prostředí. Snažil jsem se vytvořit základní schéma pro následné vytvoření kompletní dokumentace.

5.3. Návrhy na vylepšení

5.3.1. Senzor vzdálenosti

Komplikace byly zejména s výběrem vhodných součástek, páru infračervené LED diody a fototranzistoru. Zde si myslím, že při troše zkoušení lze dosáhnout ještě vyššího dosahu, popřípadě vyšší přesnosti.

5. Další činnosti, návrhy na vylepšení a na co nezbyl čas

Nečekanou nevýhodou senzoru bylo v konečné implementaci mrtvé pásmo. Zde mám zato, že změnou zapojení (například měřením napětí na rezistorech místo na fototranzistoru) bychom tento jev odstranili.

Dalším vylepšením by bylo přizpůsobení rozměrů desky plošných spojů senzoru tak, aby přesně zapadly do auta a nebylo potřeba (tak moc) mechanických úprav.

5.3.2. Řízení kolon

Při zkoušení kolon jsem narazil na řadu problémů. Prvním z nich byla komunikace s auty, kde jsme schopni v jedné chvíli navázat spojení pouze s jedním autem. To poněkud komplikovalo hromadné spouštění skriptů pro pokusy a bylo potřeba aktivovat jedno auto za druhým. Stálo by za to vytvořit program, který by zjistil všechny aktivní vozidla a udržoval s nimi spojení.

Dalším omezením pro mne byl nedostatek aut. V zadání bakalářské práce jsem měl úkol vytvořit kolonu deseti vozidel, což se mi podařilo, ale v době, kdy jsem prováděl měření, jich již část nebyla funkční.auta jsou poměrně složitá a byla vytvářena více lidmi v časovém nedostatku. Jejich porucha je běžnou záležitostí a i během pokusů jsem musel pár aut opravovat. Nakonec jsem byl rád, když jsem jednu sadu experimentů dodělal s pouhými šesti vozidly.

Doporučoval bych také vytvoření nějaké GUI aplikace pro pokusy s kolonou. Určitě by to zjednodušilo mnohé jinak zdlouhavé procesy, kdy jsem jednotlivé parametry před pokusem vždy upravoval v kódu a pak je nahrával postupně do všech aut. Pro mé experimenty to stačilo, ale pokročilá zpráva by se určitě bohatě vyplatila.

5.4. Na co nezbyl čas

Práce na projektu byla časově náročnější, než se původně předpokládalo. Často jsme byli závislí na jiných lidech, spoustu času zabralo jen čekání např. na dokončení návrhu a zhotovení desek plošných spojů. To vedlo k časovému skluzu, takže jsme část věcí, které jsme na začátku plánovali, museli odložit a v rámci bakalářské práce se již nestihnou.

Nezbyl například čas na vyzkoušení pokročilých algoritmů řízení kolon. Například se pokusit potlačit zmíněnou stringovou nestabilitu.

Další zajímavé pokusy by mohly být s vzájemnou komunikací vozidel. Zde jsme však ztratili člena týmu a tím se tato část projektu také odložila.

Také jsme nestihli ani začít řešit kameru a snímání s následnou analýzou obrazu. Pouze jsme prosadili umístění konektorů kamery na desku plošných spojů auta.

6. Závěr

Bakalářská práce se skládá ze dvou velkých částí a spousty drobných doprovodných prací. Rád bych každou část stručně zhodnotil a obhájil výsledky porovnané se zadáním.

První část bakalářské práce, a stejně tak první bod zadání, se týká implementování měření vzdálenosti pro naši platformu. Toto téma celé pokrývá kapitola 2. Nejprve jsem krátce obhájil použitou cestu, poté jsem navrhl vlastní senzor a ukázal způsob modulace a demodulace. Na simulaci jsem dokázal funkčnost konceptu. Teorii okolo modulace signálu a následné demodulace za pomoci asynchronní detekce jsem vysvětlil i v dodatku A.

Výsledný senzor svými vlastnostmi, i přes některé nedostatky, splnil naše požadavky. Vlastnosti jsou podrobně uvedeny v sekci 2.7.1 a okomentovány v sekci 2.7.3. Zde bych jen zopakoval to nejdůležitější. Senzor se skládá ze dvou oddělených částí - vysílací LED dioda a přijímací fototranzistor. Vzdálenost počítáme z amplitudy napětí na fototranzistoru. Odolnost proti okolnímu rušení jsme si zajistili modulací a následnou demodulací signálu.

Senzor je schopen měřit v rozsahu 5 až 30 centimetrů. Největší výhody senzoru obecně jsou: nízká cena, úhlová nezávislost, rychlá odezva, malé rozměry a možnost obousměrného měření. Naopak nevýhody jsou: mrtvé pásmo do 5 centimetrů, nutnost výkonného procesoru pro demodulaci, nutnost mít vysílač a přijímač.

Zatímco výhody jsou pro nás zásadní, nevýhody, krom mrtvého pásma, nás neomezuji.

Druhá velká část práce spočívala v sestavení, nastavení a rozběhnutí kolony vozidel. Základní teorii kolem regulace pohybu vozidel a řízení kolon jsem nastínil v kapitole 3. Samotnou implementaci a pokusy jsem rozvedl v následné kapitole 4. O mechanickém sestavení a ostatních vedlejších činnostech jsem se zmiňoval v kapitole 5.

Ve výsledku se nám podařilo sestavit funkční a (subjektivně) dobře jezdící kolonu deseti autodráhových vozidel, kvůli poruchovosti a jiným důvodům jsem pokusy provedl asi na sedmi z nich. Navrhl jsem a vyladil regulátor vzdálenosti, jeho složky P a I jsou nicméně v kódu snadno změnitelné pro budoucí úpravu. Vyzkoušel jsem tři druhy řízení: dopředné, obousměrné a obousměrné vážené.

Dopředné řízení fungovalo velmi dobře. Na přiložené CD a na web [1] jsem umístil video s dopředně řízenou kolonou. Obousměrné řízení se mi v praktických pokusech neosvědčilo, auta v koloně vždy začala kmitat. Obousměrné vážené řízení naopak pracovalo dle předpokladů.

Kvůli časovému nedostatku způsobenému zčásti i vnějšími vlivy, jak jsem popsal v sekci 5.4, jsme nevyřešili kameru pro vozidla a tudíž jsem se ani nezabýval zpracováním obrazu.

Osobně doufám v další rozvoj projektu. Tím, že jsme úspěšně vyzkoušeli kolonu o deseti vozidlech, jsme připravili vše potřebné pro sériovou výrobu aut dalších. Principy jsme ověřili, nyní je nutno dotáhnout projekt do zdárného konce.

6. Závěr

Naše kolona je malá a moc velký zájem tedy nepřitáhne. Rozšířením však teprve získá na významu. Cílovou představou je dvacet metrů dlouhá dráha s padesáti seřazenými auty. Osobně bych moc rád takovouto kolonu viděl, proto projektu a všem lidem v něm zapojených přeji co nejvíce úspěchů!

Příloha A.

Teorie modulace a demodulace signálu

V rámci zachování plynulosti textu umísťuji teoretickou část týkající se modulace, rušení a následné demodulace signálu do přílohy.

Zde je objasněna a názorně ukázána funkčnost synchronní a asynchronní detekce na příkladech spojitého signálu v programu Simulink.

A.1. Modulace a demodulace signálu

Vzhledem k náchylnosti předchozího řešení k rušení je nutné provést modulaci a následnou demodulaci našeho signálu, vysílaného IR LED diodou a snímaného fototranzistorem.

Pro úplnou představu tohoto procesu ho nejprve uvedu souhrnně, následně jednotlivé body rozvedu do podrobností a nakonec budu diskutovat praktickou implementaci.

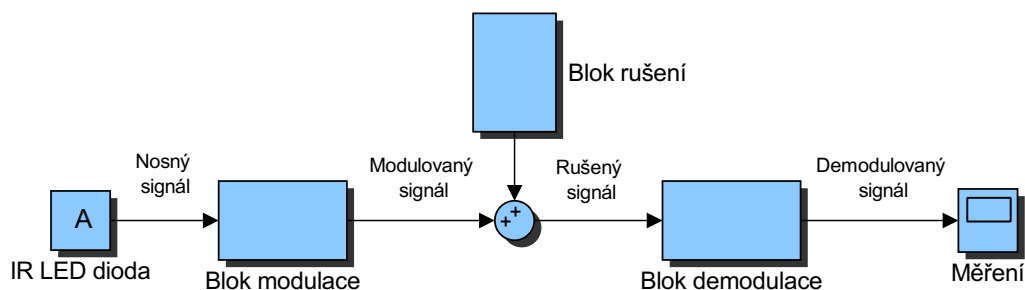
Nejprve je nutné vysílaný infračervený světelný signál modulovat. Nejjednodušší modulace je amplitudová. V našem případě budeme pouze spínat LED diodu několikrát za sekundu. Tak vytvoříme v ideálním případě obdélníkový signál.

Dalším krokem je tento signál přijmout fototranzistorem a demodulovat. V ideálním případě přijmeme opět obdélníky a demodulací z nich vytvoříme spojitý signál. Jediný rozdíl mezi původním a demodulovaným signálem by měl být v amplitudě.

Tímto procesem bychom také měli odfiltrovat veškeré rušení, ať už se jedná o zářivky, které náš signál ruší periodicky na frekvenci 100Hz, sluneční světlo, které můžeme považovat za stejnosměrné rušení, či prostě náhodný šum.

Pro přehlednost celého procesu bych rád nyní uvedl simulinkový model, ze kterého budu vycházet a ke kterému se budu vracet (Obrázek 30). Skládá se ze tří základních částí - modulace, rušení a demodulace.

V dalších kapitolách rozeberu podrobněji jednotlivé bloky. Cílem je vysvětlit princip a demonstrovat funkčnost zvolené modulovací metody.



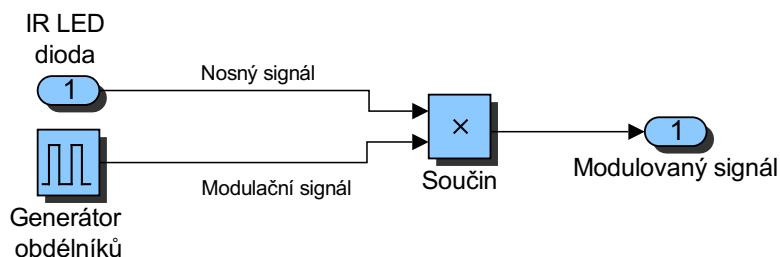
Obrázek 30. Schéma procesu modulace, rušení a demodulace

A.1.1. Modulace

Nejprve musíme náš signál IR LED diody modulovat. To provedeme jednoduše, prostě budeme naši LED diodu spínat s určitou frekvencí¹. Vznikne nám tak obdélníkový signál.

Princip amplitudové modulace je jednoduchý. Světelný signál z LED diody o nějaké amplitudě je pro nás nosným signálem. Obdélníkový signál s amplitudou 1 o zvolené frekvenci je signál modulační. Pokud vynásobíme modulační a nosný signál, vznikne nám signál modulovaný.

Tento proces zachycuje schéma 31.

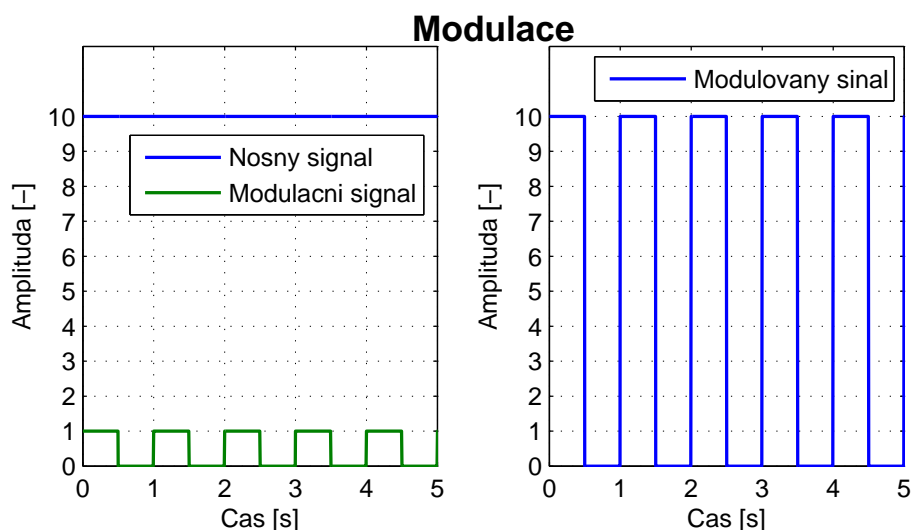


Obrázek 31. Simulinkový model zachycující modulaci signálu

Grafy

Pro názornost je v grafu 32 použit spojitý signál, modulace má následující parametry:

- Amplituda nosného signálu 10
- Frekvence modulace 1 Hz



Obrázek 32. Modulace signálu - nahoře nosný a modulační signál, dole signál modulovaný

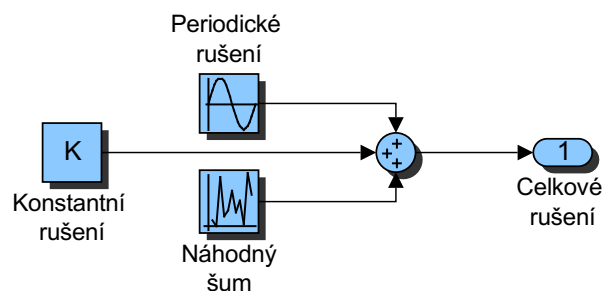
¹Zvolení vhodné frekvence je popsáno v sekci 2.5

A.1.2. Rušení

Modulaci a demodulaci provádíme zejména z důvodu náchylnosti původního konceptu k rušení. Přidáme ho tedy i do naší simulace a budeme pozorovat, jak ovlivňuje výsledný demodulovaný signál.

V simulaci budeme předpokládat, že se jedná o ideální případ, kdy rušení je lineární proces. Pouze ho tedy přičteme k modulovanému signálu.

Blok rušení je znázorněn na schématu 33.



Obrázek 33. Simulinkový model rušícího signálu

Zvolil jsem tři druhy rušení, se kterými se setkáme i v praxi. První představuje okolní světlo a je konstantní. Druhé jsou například zářivky, které periodicky blikají. Poslední reprezentuje náhodný šum, který vzniká ve vedení či na senzorech.

Grafy

Pro naši jednoduchou ukázkovou simulaci budeme předpokládat následující hodnoty amplitudy rušení:

- Periodické rušení: $\sin\left(\frac{2\pi \cdot t}{10}\right)$
- Konstantní rušení: 3
- Náhodný šum: rozmezí $\langle 0, 0.8 \rangle$

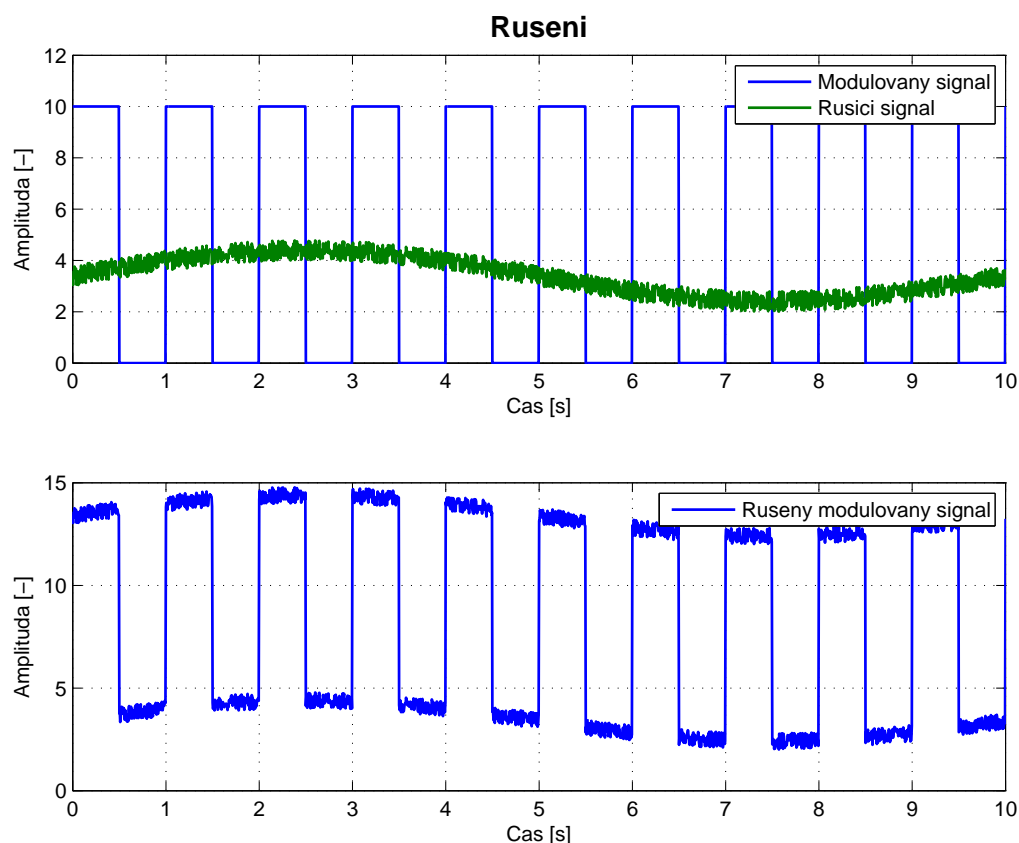
A.1.3. Demodulace

Posledním krokem celého procesu je demodulace. Ta je poněkud složitější než předchozí dva kroky. Cílem je odfiltrout rušení a získat z modulovaného signálu (obdélníků) původní konstantní signál.

Využijeme zde znalost frekvence modulovaného signálu.

Jednou z možností by bylo použít pásmovou propust. Chceme odfiltrout stejnosměrnou složku signálu a všechny frekvence krom modulační. Vytvořit takto úzkofrekvenční pásmovou propust by bylo však zbytečně složité a implementace v diskrétních výpočtech procesoru příliš náročná. Při větší vzdálenosti LED diody a fototranzistoru odstup signálu, rušení a šumu znatelně klesá a my potřebujeme detekovat i takto slabý a zašuměný signál.

Využijeme toho, že nás zajímá skutečně jen specifické frekvenční spektrum - jedna námi zvolená frekvence, zbytek chceme odfiltrout. Pro tento případ můžeme použít synchronní nebo asynchronní detekci.



Obrázek 34. Rušení - nahoře signál rušení a modulovaný signál, dole rušený modulovaný signál

Synchronní detekce

Synchronní detekce (popsaná například zde [17]) je metoda využívaná pro velmi zašuměné signály. Vychází ze znalosti **frekvence** a **fáze** demodulovaného signálu. Měřený signál přičítáme a odečítáme se stejnou periodou a fází.

Náhodný šum a konstantní rušení bude v přičítací i odečítací fázi poměrně stejné, proto se ve výsledku vyruší. Taktéž signál s velmi odlišnou frekvencí se podaří částečně odfiltrvat.

V praxi se pro tuto činnost používá přístroj zvaný synchronní detektor. Nevýhodou může být, že výsledný signál získáme pouze jednou za periodu modulovaného signálu.

Tento demodulační proces můžeme vyjádřit například s pomocí sinové funkce:

$$S_{demod} = \frac{1}{T_{mod}} \int_{t'}^{t'+T} [S_{mod,ruš}(t) \cdot \sin(2\pi f_{mod}t + \varphi_{mod})] dt \quad (17)$$

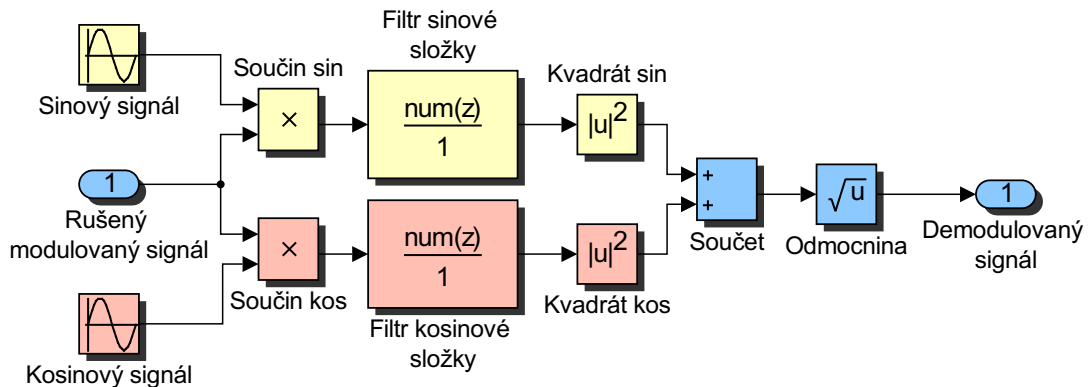
S_{demod} je amplituda demodulovaného signálu, kterou získáme jednou za periodu modulovaného signálu T_{mod} . $S_{mod,ruš}(t)$ je modulovaný rušený signál, f_{mod} a φ_{mod} jsou frekvence a fáze modulovaného signálu.

Asynchronní detekce

V našem případě však nemůžeme využít synchronní detekce, neboť neznáme fázi modulovaného signálu. Pokud však provedeme zvlášť synchronní demodulaci se sinovou

a kosinovou funkcí, umocníme je druhou mocninou, sečteme a tento součet odmocníme, získáme tak demodulovaný signál i bez znalosti fáze.

Tento proces se nazývá asynchronní detekce. Vychází ze základní vlastnosti sinu a kosinu, kdy $\sin^2 x + \cos^2 x = 1$ a kdy jedna funkce je posunutá od druhé a půl fáze. Odpadá nám tím podmínka znalosti fáze modulovaného signálu a vystačíme si pouze se znalostí jeho **frekvence**. Celý proces demodulace je znázorněn na schématu 35.



Obrázek 35. Simulinkový model zachycující demodulaci signálu provedenou asynchronní detekcí

Vztah pro získání demodulovaného signálu asynchronní detekcí může vypadat například následujícím způsobem:

$$S_{demod,sin} = \frac{1}{T_{mod}} \int_{t'}^{t'+T} [S_{mod,ruš}(t) \cdot \sin(2\pi f_{mod}t)] dt \quad (18)$$

$$S_{demod,cos} = \frac{1}{T_{mod}} \int_{t'}^{t'+T} [S_{mod,ruš}(t) \cdot \cos(2\pi f_{mod}t)] dt \quad (19)$$

$$S_{demod} = \sqrt{(S_{demod,sin}^2 + S_{demod,cos}^2)} \quad (20)$$

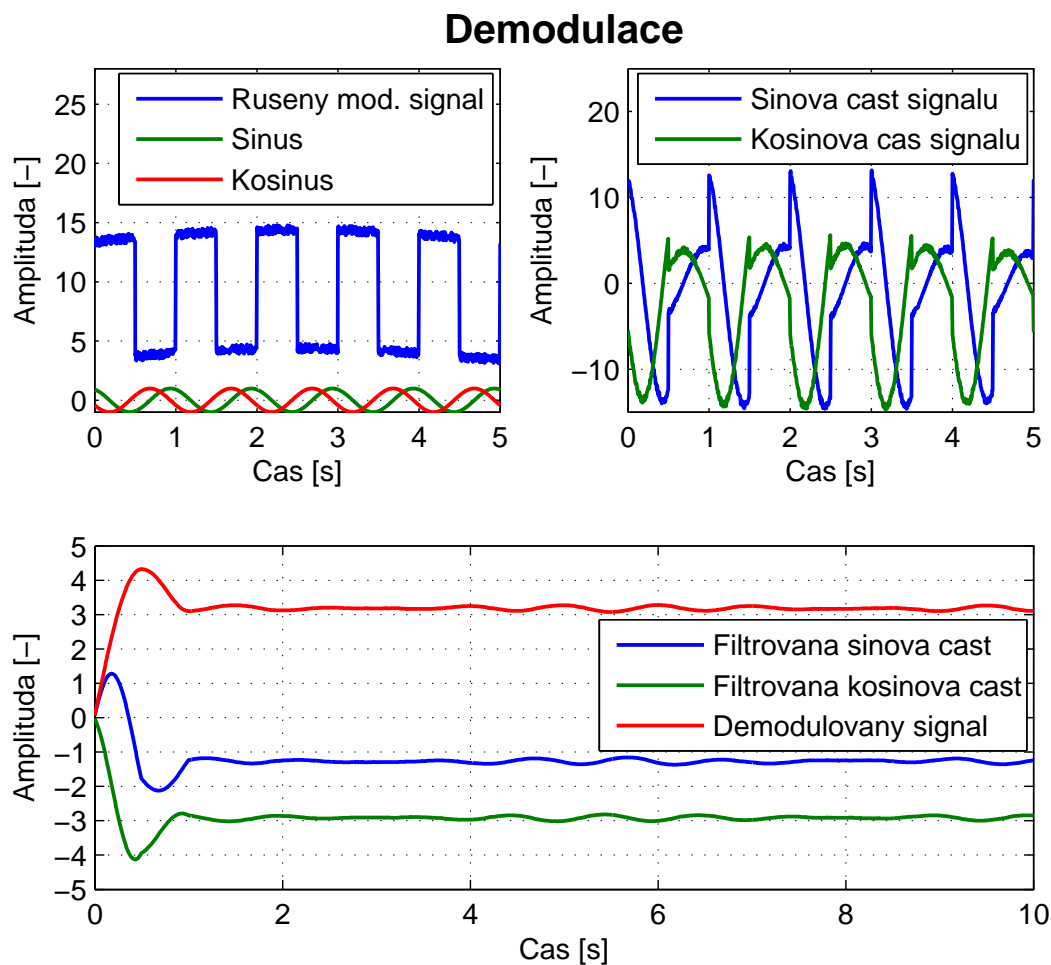
Demodulovaný signál takto získáme opět jednou za periodu T_{mod} .

V simulinkovém schématu místo průměrování jednou za periodu používáme klouzavý průměr přes periodu. Je to jen prostá náhrada, princip zůstává stejný. Jsme tím však schopni simulovat demodulaci signálu pro "spojitý"² signál.

Grafy

Demodulovat budeme rušený signál z minulého příkladu. Proces zachycuje graf 36. Vidíme zde postup od rušeného signálu se sinovou a kosinovou funkcí (všimněme si, že jejich fáze je odlišná od fáze rušeného signálu), jejich násobení a následnou filtraci, až po přepočítání na výsledný demodulovaný signál. Úplně jsme vyfiltrovali stejnosměrné rušení a šum, ostatní alespoň částečně.

²O spojité signál se však ve skutečnosti samozřejmě nejedná, Simulink jej pouze aproximuje častým vzorkováním



Obrázek 36. Demodulace signálu - horní grafy zachycují rušený modulovaný signál a sinovou a kosinovou funkci použitou pro demodulaci (vlevo), rušený signál vynásobený sinovou a kosinovou funkcí (vpravo), dolní graf zobrazuje demodulaci sinové a kosinové složky signálu a výslednou demodulaci

Výsledný demodulovaný signál ale není naprosto ideální. V případě vyšších výkyvů můžeme použít dodatečné filtry, neboť, jak vysvětlím v následující sekci 2.5, získáme tento demodulovaný signál s dostatečnou frekvencí.

Další nevýhodou je ztráta amplitudy signálu po modulaci. Je to dáno tím, že pro demodulaci využíváme sinové a kosinové funkce³.

Ukázali jsme, že princip této demodulace je funkční.

³Za povšimnutí stojí, že amplituda klesne π -krát, což lze dokázat z rovnic 18 až 20

Příloha B.

Přiložené materiály na CD a webu

Nedílnou součástí práce jsou materiály na přiloženém CD a na webu [1]. Stručný obsah jednotlivých adresářů s popisem je následující:

- **Fotky a videa:** obsahuje mnou zachycené fotky aut a dvě videa s ukázkou kolony řízené dopředným řízením s referenční vzdáleností 150 milimetrů
- **Regulace:** obsahuje materiály důležité k regulaci, zejména modely v programu Simulink, na kterých jsem vytvářel a testoval vzdálenostní regulátor
- **Projekt_Slotcar_Platooning:** složka obsahuje námi vytvářené sdílené adresáře k projektu Slotcar platooning
 - **onboard:** veškeré soubory k implementaci v autech. Za shlédnutí stojí zejména:
 - * *fw->platoon_lib->measurements->distance.c*: obsahuje metody pro modulaci a demodulaci
 - * *fw->app->main.c*: hlavní soubor programu, implementuje metody z *distance.c*
 - **sw:** veškeré programy a nástroje pro komunikaci s vozidly z počítače
- **Simulace:** obsahuje skript vytvořený v programu MATLAB, který simuluje modulaci a demodulaci signálu v kódu programu, a model v programu Simulink, který ukazuje princip modulace a demodulace použitý v A

Podrobnější obsah a elektronická verze bakalářské práce jsou přiloženy přímo k adresářům.

Literatura

- [1] Příložené materiály k bakalářské práci, 2014. URL https://dl.dropboxusercontent.com/u/28453913/BP_Jan_Moravec_2014.zip.
- [2] E. Coelingh and S. Solyom. All aboard the robotic road train. *IEEE Transactions on Vehicular Technology*, 40(11):34–39, 2012.
- [3] Project Sartre, 2014. URL <http://www.sartre-project.eu/>.
- [4] Project PATH, 2014. URL <http://www.path.berkeley.edu/>.
- [5] S. Shladover, C. Desoer, J. Hedrick, M. Tomizuka, J. Walrand, W. Zhang, D. McMahon, H. Peng, S. Sheikholeslam, and N. McKeown. Automated vehicle control developments in the PATH program. *IEEE Spectrum*, 49(1):114–130, 1991.
- [6] Eindhoven VEHL, 2014. URL <http://www.tue.nl/en/publication/ep/p/d/ep-uid/270656/>.
- [7] Dan Martinec. Distributed control of platoons of racing slot cars. Master's thesis, ČVUT v Praze, 2012. URL http://support.dce.felk.cvut.cz/mediawiki/images/7/75/Dp_2012_martinec_dan.pdf.
- [8] Materiály k předmětu A3B38SME, 2013. URL http://fieldbus.feld.cvut.cz/system/files/files/cs/vyuka/predmety/A3B38SME/Opticky_senzor_polohy_10a.pdf.
- [9] Martin Lád. Návrh a implementace řídicího systému pro autodráhové vozidlo. Master's thesis, ČVUT v Praze, 2014.
- [10] P. Seiler, A. Pant, and K. Hedrick. Disturbance propagation in vehicle strings. *IEEE Transactions on Vehicular Technology*, 49(10):1835–1842, 2004.
- [11] Bezpečná vzdálenost mezi vozidly dle útvaru BESIP, 2014. URL <http://www.ibesip.cz/cz/ridic/bezpecne-rizeni-vozidla/bezpecna-vzdalenost>.
- [12] D. Swaroop and J.K. Hedrick. String stability of interconnected systems. *IEEE Transactions on Vehicular Technology*, 41(3):349–357, 1996.
- [13] GIT, 2014. URL <http://git-scm.com/>.
- [14] Podpůrný web Katedry řídicí techniky, fakulty elektrotechnické, ČVUT v Praze, 2014. URL https://support.dce.felk.cvut.cz/mediawiki/index.php/Hlavn%C3%AD_strana.
- [15] Nástroj Doxygen, 2014. URL <http://www.stack.nl/~dimitri/doxygen/index.html>.
- [16] Eclipse. <http://eclipse.org/>, 2014. URL <http://eclipse.org/>.

- [17] Juraj Poliak. Synchronní detekce modulovaných optických signálů, 2009. URL https://dspace.vutbr.cz/bitstream/handle/11012/2975/BBCE_Poliak.pdf?sequence=1.