



# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

## FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

## ÚSTAV AUTOMATIZACE A MĚŘICÍ TECHNIKY

DEPARTMENT OF CONTROL AND INSTRUMENTATION

## OPTIMALIZACE FUZZY REGULÁTORU V PLC

FUZZY CONTROLLER OPTIMIZATION IN PLC

### DIPLOMOVÁ PRÁCE

MASTER'S THESIS

### AUTOR PRÁCE

AUTHOR

Bc. Dominik Piskoř

### VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Jakub Arm, Ph.D.

BRNO 2022

# Diplomová práce

magisterský navazující studijní program **Kybernetika, automatizace a měření**

Ústav automatizace a měřicí techniky

**Student:** Bc. Dominik Piskoř

**ID:** 203322

**Ročník:** 2

**Akademický rok:** 2021/22

**NÁZEV TÉMATU:**

## Optimalizace fuzzy regulátoru v PLC

### POKYNY PRO VYPRACOVÁNÍ:

Práce se zabývá vytvořením fuzzy regulátoru v PLC zařízení regulující otáčky větrného tunelu (modelovaného pomocí fuzzy logiky). Předpokládá se vytvoření regulátoru a funkce optimalizace v prostředí Simulink s následnou simulací a vygenerováním kódu pro PLC pomocí toolboxu Matlab PLC Coder.

- 1) Seznamte se s metodami optimalizace fuzzy regulátorů.
- 2) Navrhněte manuálně fuzzy regulátor pro danou soustavu v prostředí Simulink.
- 3) Implementujte funkci offline optimalizace parametrů regulátoru na reálném zařízení.
- 4) Dosažené výsledky srovnajte s lineárním regulátorem optimálním podle daného kritéria.

### DOPORUČENÁ LITERATURA:

Abonyi, J. Fuzzy model identification for control. Birkhauser, 2003. ISBN: 3-7643-4238-2

**Termín zadání:** 7.2.2022

**Termín odevzdání:** 18.5.2022

**Vedoucí práce:** Ing. Jakub Arm, Ph.D.

**doc. Ing. Petr Fiedler, Ph.D.**  
předseda rady studijního programu

### UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

## **ABSTRAKT**

Diplomová práce se zabývá řešením optimalizačních metod vhodných k optimalizaci fuzzy regulátorů. Dále se zaměřuje na manuální návrh fuzzy regulátoru typu Mamdani, off-line optimalizaci navrženého regulátoru a otestování optimalizovaného regulátoru řízením reálné soustavy. Tato reálná soustava má podobu modelu větrného tunelu skládajícího se z ventilátoru, trubky a turbíny umístěné na hřídeli rotačního enkóderu sloužícího jako snímač. Poslední částí je porovnání dosažených výsledků s lineárním regulátorem, který je optimalizován podle daného kritéria.

## **KLÍČOVÁ SLOVA**

fuzzy regulátor, Mamdani, optimalizace, Nelder-Mead, model větrného tunelu, ITAE kritérium, PLC

## **ABSTRACT**

This diploma thesis deals with the search for optimization methods suitable for fuzzy controller optimization. The next goal of this thesis is the manual design of the Mamdani fuzzy controller, off-line optimization of the designed fuzzy controller, and the testing of the optimized controller by controlling a real system. The real system is a model of the wind tunnel, which is consisted of a fan, a tube, and a small turbine placed on a shaft of a rotary encoder. This rotary encoder is used as a sensor. The last part of this thesis is a comparison of the achieved results with a linear controller, which is optimized according to the given criterion.

## **KEYWORDS**

fuzzy controller, Mamdani, optimization, Nelder-Mead, wind tunnel model, ITAE criterion, PLC

PISKOŘ, Dominik. *Optimalizace fuzzy regulátoru v PLC*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav automatizace a měřicí techniky, 2022, 86 s. Diplomová práce. Vedoucí práce: Ing. Jakub Arm, Ph.D.



## Prohlášení autora o původnosti díla

<b>Jméno a příjmení autora:</b>	Bc. Dominik Piskoř
<b>VUT ID autora:</b>	203322
<b>Typ práce:</b>	Diplomová práce
<b>Akademický rok:</b>	2021/22
<b>Téma závěrečné práce:</b>	Optimalizace fuzzy regulátoru v PLC

Prohlašuji, že svou závěrečnou práci jsem vypracoval samostatně pod vedením vedoucí/ho závěrečné práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené závěrečné práce dále prohlašuji, že v souvislosti s vytvořením této závěrečné práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno .....

.....

podpis autora\*

---

\*Autor podepisuje pouze v tištěné verzi.

## PODĚKOVÁNÍ

Rád bych poděkoval vedoucímu své diplomové práce, panu Ing. Jakubu Armovi, Ph.D., za pedagogickou a odbornou pomoc i za další cenné a podnětné rady při zpracování mé diplomové práce. Také bych zde velice rád poděkoval panu Ing. Ondreji Mihálikovi a panu Ing. Tomáši Sýkorovi za veškerou jejich pomoc a trpělivost, kterou se mnou měli při vypracování této diplomové práce.

# Obsah

Úvod	12
<b>1 Fuzzy systémy</b>	<b>13</b>
1.1 Úvod k fuzzy problematice	13
1.2 Popis fuzzy systémů	15
1.2.1 Fuzzifikace	16
1.2.2 Defuzzifikace	16
1.3 Fuzzy systém typu Mamdani	17
1.4 Fuzzy systém typu Takagi-Sugeno	18
1.5 Fuzzy regulátory	19
1.5.1 Fuzzy P regulátor	20
1.5.2 Fuzzy PI regulátor	20
1.5.3 Fuzzy PD regulátor	21
<b>2 Popis modelu větrného tunelu</b>	<b>22</b>
2.1 Tubus	22
2.2 Ventilátor	22
2.3 Turbína a inkrementální snímač otáček	24
<b>3 PLC Siemens S7-1200</b>	<b>26</b>
<b>4 Metody optimalizace</b>	<b>27</b>
4.1 Metoda Nelder-Mead	27
4.2 Newtonova metoda a line search	29
4.3 Metoda zlatého řezu	32
<b>5 Identifikace větrného tunelu</b>	<b>35</b>
5.1 TIA Portal	35
5.2 MATLAB a System Identification Toolbox	38
5.3 Nová identifikace větrného tunelu	42
<b>6 Manuální návrh fuzzy regulátoru</b>	<b>49</b>
6.1 Návrh konstant regulátoru	49
6.2 Návrh fuzzy regulátoru ve Fuzzy Toolboxu	51
6.3 Zapojení fuzzy regulátoru v Simulinku	53
6.4 Dosažené výsledky s manuálně navrženým fuzzy PI regulátorem	54

<b>7</b>	<b>Dodatečné úpravy regulátoru a zapojení</b>	<b>56</b>
7.1	Rozšíření navrženého fuzzy PI regulátoru . . . . .	56
7.2	Změna v zapojení regulačního obvodu . . . . .	58
<b>8</b>	<b>Optimalizace fuzzy PI regulátoru</b>	<b>60</b>
8.1	Optimalizační parametry fuzzy regulátoru . . . . .	60
8.2	Optimalizační funkce . . . . .	61
8.3	Dosažené výsledky optimalizace v Simulinku . . . . .	62
<b>9</b>	<b>Řízení reálného systému fuzzy regulátorem v PLC</b>	<b>66</b>
9.1	Přenos kódu ze Simulinku do PLC . . . . .	66
9.2	Program v prostředí TIA Portal . . . . .	67
9.3	Dosažené výsledky při řízení reálného modelu . . . . .	68
	<b>Závěr</b>	<b>74</b>
	<b>Literatura</b>	<b>77</b>
	<b>Seznam příloh</b>	<b>79</b>
<b>A</b>	<b>Odezvy regulačního obvodu po optimalizaci regulátoru</b>	<b>80</b>
<b>B</b>	<b>Obsah elektronické přílohy</b>	<b>86</b>

# Seznam obrázků

1.1	Příklad pro fuzzy logiku . . . . .	13
1.2	Vybrané typy funkcí příslušnosti (převzato z [1]) . . . . .	14
1.3	Struktura fuzzy systému (převzato z [1]) . . . . .	15
1.4	Fuzzifikace singletonem a fuzzy množinou (převzato z [1]) . . . . .	16
1.5	Defuzzifikace metodou středu plochy-těžiště (převzato z [1]) . . . . .	17
1.6	Struktura fuzzy regulátoru (převzato z [1]) . . . . .	19
2.1	Model větrného tunelu . . . . .	22
2.2	HT trubka tvořící tubus tunelu (ilustrační obrázek) [5] . . . . .	22
2.3	Ventilátor Sunon Maglev ME80252V3-000U-G99 [4] . . . . .	23
2.4	Inkrementální enkodér LPD3806 600BM G5 24C [6] . . . . .	24
3.1	SIMATIC S7-1200 CPU 1214C 6ES7214-1AG40-0XB0 [10] . . . . .	26
5.1	Schéma zapojení větrného tunelu pro identifikaci . . . . .	35
5.2	Časový průběh signálu pro identifikaci . . . . .	36
5.3	Časový průběh signálu pro verifikaci . . . . .	37
5.4	Odezva větrného tunelu na identifikační signál . . . . .	37
5.5	Odezva větrného tunelu na verifikační signál . . . . .	38
5.6	Struktura systému typu Hammerstein [15] . . . . .	38
5.7	Porovnání skutečné odezvy a odezvy identifikovaného modelu . . . . .	39
5.8	Porovnání skutečné odezvy a odezvy identifikovaného modelu - validační . . . . .	40
5.9	Identifikovaná nelinearita větrného tunelu . . . . .	41
5.10	Přechodová charakteristika identifikované lineární části modelu . . . . .	41
5.11	Časový průběh signálu pro identifikaci . . . . .	42
5.12	Časový průběh signálu pro verifikaci . . . . .	43
5.13	Odezva větrného tunelu na identifikační signál . . . . .	44
5.14	Odezva větrného tunelu na verifikační signál . . . . .	44
5.15	Porovnání skutečné odezvy a odezvy identifikovaného modelu . . . . .	45
5.16	Porovnání skutečné odezvy a odezvy identifikovaného modelu v detailu . . . . .	46
5.17	Porovnání skutečné odezvy a odezvy identifikovaného modelu - validační . . . . .	46
5.18	Identifikovaná nelinearita větrného tunelu . . . . .	47
5.19	Přechodová charakteristika identifikované lineární části modelu . . . . .	48
6.1	Přenos řízení obvodu s konstantami regulátoru podle ITAE kritéria . . . . .	49
6.2	Regulační obvod pro určení konstant regulátoru . . . . .	50
6.3	Odezva obvodu s PID Controllerem na jednotkový skok . . . . .	50
6.4	Hlavní okno Fuzzy Toolboxu . . . . .	51
6.5	Definované funkce příslušnosti obou vstupů . . . . .	52
6.6	Definované funkce příslušnosti výstupu (akční zásah) . . . . .	52
6.7	Pravidla pro fuzzy regulátor . . . . .	53

6.8	Schéma zapojení v Simulinku . . . . .	53
6.9	Odezva regulačního obvodu na skokovou změnu požadované hodnoty .	55
6.10	Porovnání průběhů akčního zásahu fuzzy a lineárního regulátoru . . .	55
7.1	Funkce příslušnosti na vstupech a výstupu fuzzy regulátoru . . . . .	57
7.2	Báze pravidel fuzzy regulátoru . . . . .	57
7.3	Upravené schéma zapojení regulačního obvodu . . . . .	58
8.1	Optimalizované parametry . . . . .	60
8.2	Odezva regulačního obvodu na požadovanou hodnotu po optimalizaci	63
8.3	Porovnání akčních zásahů obou regulátorů . . . . .	63
8.4	Vývoj hodnoty ITAE kritéria při optimalizaci fuzzy regulátoru . . . .	64
8.5	Optimalizované funkce příslušnosti pro požadovanou hodnotu 300 RPM	64
9.1	Upravený obvod pro přenesení do PLC . . . . .	66
9.2	Odezva reálného reg. obvodu s fuzzy regulátorem pro 300 RPM . . .	68
9.3	Odezva reálného reg. obvodu s lineárním regulátorem pro 300 RPM .	69
9.4	Porovnání odezev reálného regulačního obvodu pro 300 RPM . . . . .	70
9.5	Porovnání akčních zásahů obou regulátorů pro 300 RPM . . . . .	70
9.6	Porovnání odezev reálného regulačního obvodu pro 600 RPM . . . . .	71
9.7	Testování fuzzy regulátoru na neočekávanou chybu . . . . .	72
9.8	Reakce regulátorů po skončení působení chyby . . . . .	73
9.9	Akční zásah regulátorů po skončení působení chyby . . . . .	73
A.1	Odezva obvodu na požadovanou hodnotu 350 RPM . . . . .	80
A.2	Odezva obvodu na požadovanou hodnotu 400 RPM . . . . .	81
A.3	Odezva obvodu na požadovanou hodnotu 400 RPM v detailu . . . . .	81
A.4	Odezva obvodu na požadovanou hodnotu 450 RPM . . . . .	82
A.5	Odezva obvodu na požadovanou hodnotu 450 RPM v detailu . . . . .	82
A.6	Odezva obvodu na požadovanou hodnotu 500 RPM . . . . .	83
A.7	Odezva obvodu na požadovanou hodnotu 500 RPM v detailu . . . . .	83
A.8	Odezva obvodu na požadovanou hodnotu 550 RPM . . . . .	84
A.9	Odezva obvodu na požadovanou hodnotu 550 RPM v detailu . . . . .	84
A.10	Odezva obvodu na požadovanou hodnotu 600 RPM . . . . .	85
A.11	Odezva obvodu na požadovanou hodnotu 600 RPM v detailu . . . . .	85

# Seznam tabulek

2.1	Technické parametry ventilátoru (převzato z [3]) . . . . .	23
2.2	Technické parametry inkrementálního enkodéru (převzato z [7]) . . .	25
3.1	Vybrané parametry PLC Siemens S7-1200 1214C (převzato z [9]) . . .	26
8.1	Hodnoty ITAE kritéria pro optimalizované regulátory . . . . .	65

# Úvod

Cílem této diplomové práce je návrh fuzzy regulátoru a jeho následná optimalizace pro reálnou nelineární soustavu. Jako součást výstupu této diplomové práce bude také porovnání výsledků fuzzy regulátoru s běžným lineárním regulátorem.

Řízenou nelineární soustavou bude model větrného tunelu, který je složen z ventilátoru, tubusu a rotačního enkodéru s malou turbínou na jeho hřídeli. Výstup enkodéru bude sloužit jako zpětná vazba pro řízení. Fuzzy regulátor může být vhodnou alternativou k běžným lineárním regulátorům, obzvláště pokud se v regulované soustavě nachází nelinearita.

Dalším cílem této práce je rešerše ohledně optimalizačních metod vhodných pro optimalizaci fuzzy regulátoru. Optimalizační metody napomáhají modifikovat fuzzy regulátor tak, aby lépe vyhovoval aktuálním podmínkám. Použitím těchto metod lze dosáhnout adekvátnějšího průběhu regulačního děje. Bude rovněž vytvořena optimalizační funkce pro manuálně navržený fuzzy regulátor. Tato funkce následně zajistí optimalizaci jednotlivých parametrů a poté i samotnou úpravu regulátoru. Diplomová práce bude rozdělena na dvě hlavní části.

První část se zaměří převážně na teorii. Nejprve bude uveden princip fuzzy logiky, který přiblíží, jakým způsobem fuzzy logika funguje. Budou zde také definovány a vysvětleny základní pojmy související s fuzzy oblastí (např. funkce příslušnosti, universum, defuzzifikace apod.). Další kapitolou bude popis struktury, typů a funkce obecného fuzzy regulátoru. Rovněž zde bude uveden detailnější popis použité reálné soustavy a programovatelného logického automatu (PLC). Tuto první část diplomové práce zakončí kapitola, jež bude věnována teorii, popisu a principu vybraných metod pro optimalizaci fuzzy regulátoru.

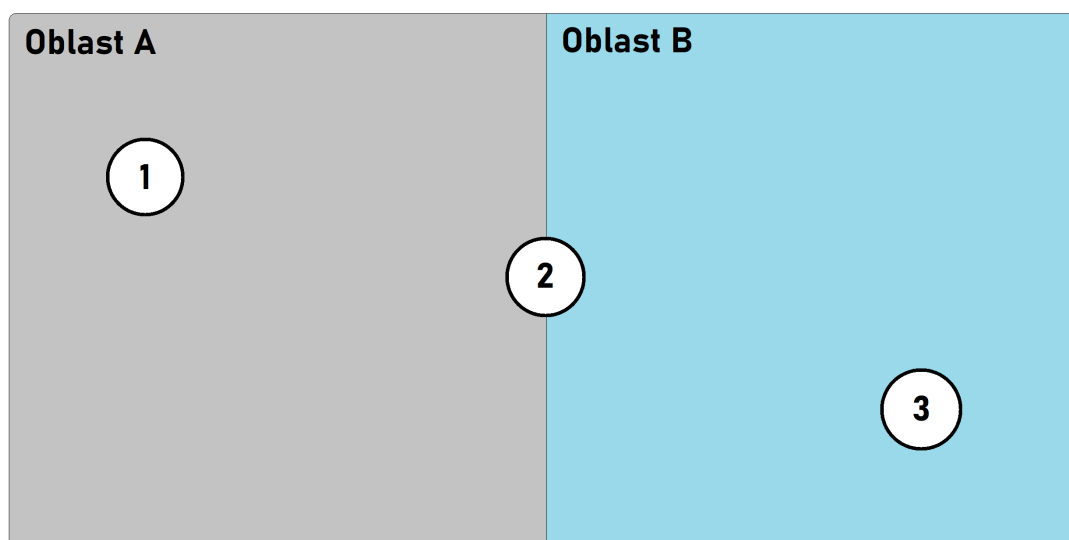
Druhá část bude praktická a přiblíží postup při identifikaci dané reálné soustavy za pomoci programů TIA Portal a MATLAB. Rovněž zde bude nastíněn průběh manuálního návrhu fuzzy regulátoru v prostředích MATLAB a Simulink. Součástí této praktické části diplomové práce bude také kapitola popisující vytvořenou optimalizační funkci a také způsob optimalizace navrženého fuzzy regulátoru. Poslední kapitola se zaměří na transformaci optimalizovaného regulátoru z programu MATLAB do PLC Siemens S7-1200 pomocí toolboxu MATLAB Coder a jeho otestování na reálné soustavě v podobě modelu větrného tunelu. Nedílnou součástí bude i prezentace dosažených výsledků.



# 1 Fuzzy systémy

## 1.1 Úvod k fuzzy problematice

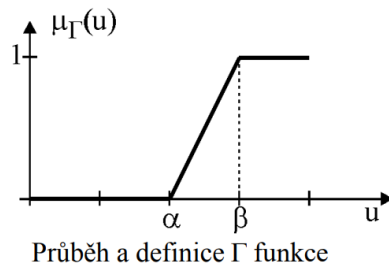
Pojem fuzzy pochází z angličtiny a v překladu znamená neurčitý, mlhavý či neostrý [1]. V tomto kontextu pak pracuje tzv. fuzzy logika, jejíž princip bude přiblížen na příkladu z obrázku 1.1.



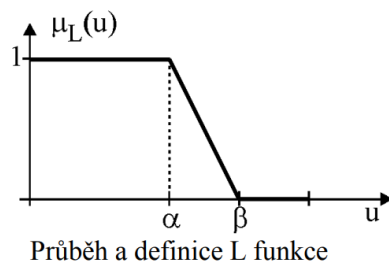
Obr. 1.1: Příklad pro fuzzy logiku

Výše uvedený obrázek znázorňuje situaci, kdy jsou dány dvě oblasti A a B. V těchto oblastech se nacházejí tři prvky označené čísly 1, 2 a 3. V rámci klasické dvoustavové logiky ANO/NE, PLATÍ/NEPLATÍ nebo 0/1 lze o prvku s číslem 1 říci, že leží v oblasti A. Obdobný úsudek je možné vyjádřit také o prvku s číslem 3, jen s tím rozdílem, že tento prvek náleží do oblasti B. U prvku číslo 2 již nelze říci jednoznačně, zdali se nachází v oblasti A nebo v oblasti B, jako tomu bylo u dvou předešlých prvků. V rámci klasické logiky je možné o prvku s číslem 2 říci, že patří do oblasti A, ale také že náleží do oblasti B. Obě tyto tvrzení jsou z pohledu klasické logiky správná, avšak ne zcela přesná. V této situaci nachází uplatnění právě fuzzy logika, jejímž výstupem je funkce příslušnosti. V rámci fuzzy logiky je možné o prvku s číslem 2 říci, že náleží do oblasti A s funkcí příslušnosti 0,5 a zároveň náleží do oblasti B s funkcí příslušnosti taktéž 0,5. O prvcích s číslem 1 a 3 lze usoudit, že náleží do oblasti A, respektive do oblasti B s funkcí příslušnosti 1. Fuzzy logika vystihuje výše popsanou skutečnost mnohem lépe a přesněji, než klasická dvoustavová logika.

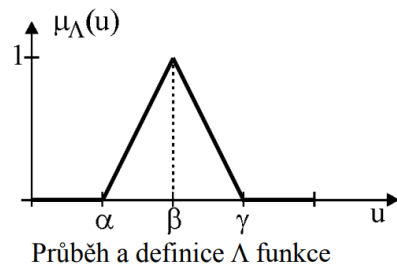
V rámci příkladu uvedeném v předchozím odstavci byl zmíněný pojem funkce příslušnosti. Funkce příslušnosti definuje, jakou měrou přísluší daný prvek dané oblasti ve fuzzy logice. V případě klasické logiky se jedná o charakteristickou funkci, tj. funkci, která mapuje universum (rozsah přípustných hodnot) do množiny se dvěma prvky (např.  $\{0,1\}$ ). Naopak funkce příslušnosti ve fuzzy logice mapuje universum na celý interval (např.  $\langle 0,1 \rangle$ ). Tvar funkce příslušnosti může být různý. Nejčastěji používané tvary jsou  $\Gamma$ -funkce, L-funkce,  $\Lambda$ -funkce a  $\Pi$ -funkce. Jejich průběhy jsou zobrazeny na obrázku 1.2.



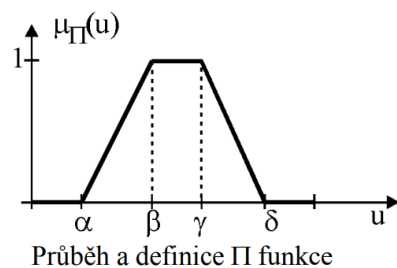
$$\Gamma(u, \alpha, \beta) = \begin{cases} 0 & u < \alpha \\ (u - \alpha) / (\beta - \alpha) & \alpha \leq u \leq \beta \\ 1 & u > \beta \end{cases}$$



$$L(u, \alpha, \beta) = \begin{cases} 1 & u < \alpha \\ (\beta - u) / (\beta - \alpha) & \alpha \leq u \leq \beta \\ 0 & u > \beta \end{cases}$$



$$\Lambda(u, \alpha, \beta, \gamma) = \begin{cases} 0 & u < \alpha \\ (u - \alpha) / (\beta - \alpha) & \alpha \leq u \leq \beta \\ (\gamma - u) / (\gamma - \beta) & \beta \leq u \leq \gamma \\ 0 & u > \gamma \end{cases}$$

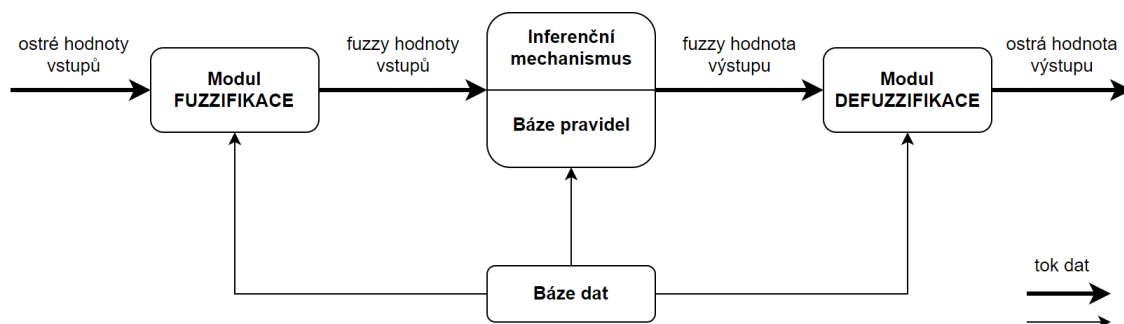


$$\Pi(u, \alpha, \beta, \gamma, \delta) = \begin{cases} 0 & u < \alpha \\ (u - \alpha) / (\beta - \alpha) & \alpha \leq u \leq \beta \\ 1 & \beta \leq u \leq \gamma \\ (\delta - u) / (\gamma - \delta) & \gamma \leq u \leq \delta \\ 0 & u > \delta \end{cases}$$

Obr. 1.2: Vybrané typy funkcí příslušnosti (převzato z [1])

## 1.2 Popis fuzzy systémů

Pod termínem fuzzy systém si lze představit takový systém, jehož proměnné jsou definovány slovními hodnotami, tj. fuzzy množinami. Tyto slovní hodnoty se nazývají jazykové proměnné a jsou definovány na daném universu. Velmi často bývá toto universum tvořeno intervalem reálných čísel. Základní strukturu fuzzy systému ilustruje obrázek 1.3. Uvedenou ostrou hodnotou se rozumí hodnota náležící do klasické množiny [1].



Obr. 1.3: Struktura fuzzy systému (převzato z [1])

Báze dat zahrnuje data o fuzzy množinách všech proměnných daného systému. Obsahuje informace o tvaru fuzzy množiny a její poloze v universu. Báze pravidel začleňuje veškerá pravidla fuzzy systému. Podle těchto pravidel se řídí výstup systému. Obě uvedené báze bývají nazývány jednotně jako znalostní báze [1]. Inferenční mechanismus vyhodnocuje jednotlivá pravidla podle daného pořadí a slučuje jejich výsledky do jedné výsledné fuzzy množiny. Jednotlivá pravidla mají například následující tvar [1]:

$$IF \text{ (fuzzy výrok) } THEN \text{ (fuzzy výrok)}$$

První fuzzy výrok v pravidle se nazývá *antecedent* a může být tvořen dílčími fuzzy výroky propojenými pomocí logických spojek AND nebo OR. Samotné pravidlo představuje fuzzy implikaci a jeho výsledek se nazývá *konsekvent* [1]. Uvedené pravidlo náleží fuzzy systému typu Mamdani, který je popsán v kapitole 1.3. Pro bližší představu lze jako příklad uvést následující pravidlo:

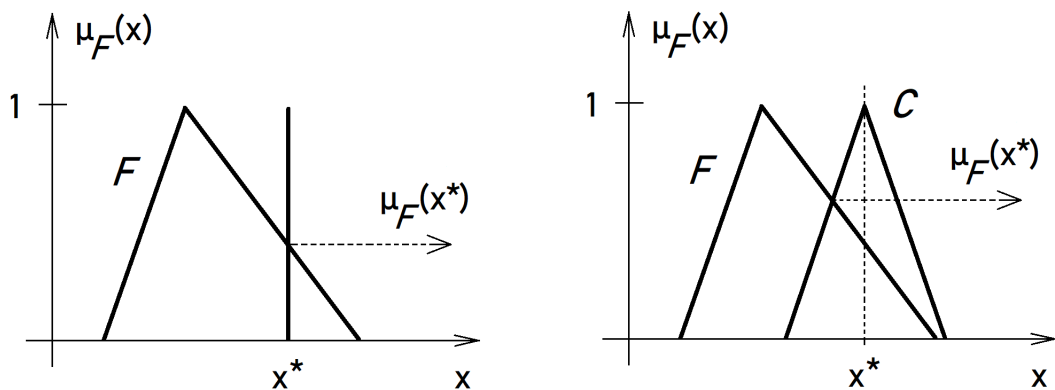
$$IF \text{ (} v \text{ je malá) } AND \text{ (} d \text{ je velká) } THEN \text{ (} F \text{ je malá)}$$

Pravidlo ilustruje situaci auta jedoucího za jiným autem. Pokud je rychlost (označena jako  $v$ ) daného auta malá a vzdálenost (označena jako  $d$ ) mezi auty je velká, pak brzdná síla (označena jako  $F$ ) daného auta zabraňující srážce obou aut bude

malá. Proměnné rychlost a vzdálenost lze nazvat jako jazykové proměnné. Výrazy malá a velká pak je možno definovat jako hodnoty těchto jazykových proměnných [1].

### 1.2.1 Fuzzifikace

Fuzzifikace má za úkol transformovat ostrou hodnotu vstupní proměnné na fuzzy množinu. Pro svou nízkou výpočetní náročnost se nejčastěji používá fuzzifikace singletonem. Singleton je jeden z typů funkcí příslušnosti, pro jehož tvar je typická jedna ostrá čára na dané pozici slovní hodnoty [1]. V některých případech nemusí být číslo vstupující do fuzzifikace ostré, ale může se jednat o číslo vyjádřené fuzzy množinou, která představuje kupříkladu neurčitost vstupní hodnoty [1]. Oba výše uvedené případy fuzzifikace jsou znázorněny na obrázku 1.4.



Obr. 1.4: Fuzzifikace singletonem a fuzzy množinou (převzato z [1])

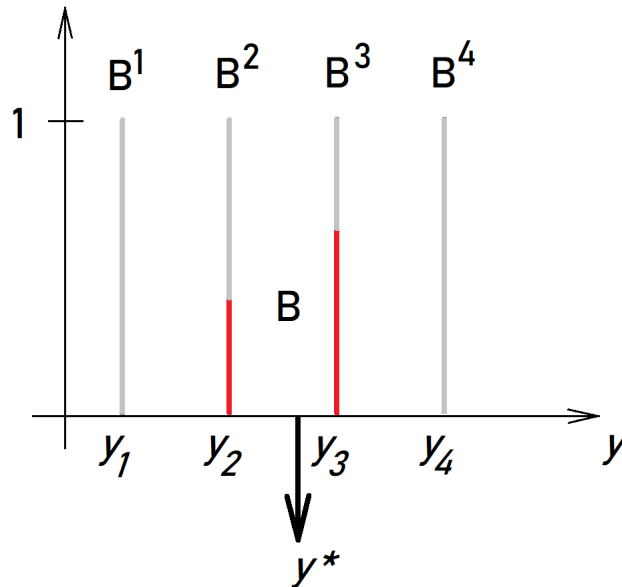
### 1.2.2 Defuzzifikace

Defuzzifikace má přesně opačný úkol než fuzzifikace. Z fuzzy množiny (jazykové proměnné) získává ostrou hodnotu dané veličiny na výstupu. Opět se zde využívá funkce příslušnosti typu singleton kvůli své nízké výpočetní náročnosti. Existuje několik typů defuzzifikačních metod. Mezi ty nejčastěji používané patří [1]:

- metoda středu plochy-těžiště
- metoda středu součtů
- metoda průměru středů

Obrázek 1.5 zobrazuje defuzzifikaci metodou středu plochy (také nazývanou jako metoda středu těžiště) na diskretním universu.  $B^1$  až  $B^4$  jsou fuzzy množiny a  $y_1$  až  $y_4$  pak značí jednotlivé jazykové proměnné. Výstupem je ostrá hodnota  $y^*$ , která je

určena jako souřadnice těžiště fuzzy množin  $B^2$  a  $B^3$ . Zbývající fuzzy množiny se při defuzzifikaci neuplatní, jelikož mají nulovou hodnotu funkce příslušnosti.



Obr. 1.5: Defuzzifikace metodou středu plochy-těžiště (převzato z [1])

### 1.3 Fuzzy systém typu Mamdani

V této podkapitole je popsán jeden z nejznámějších a nejpoužívanějších fuzzy systémů, a to fuzzy systém typu Mamdani. Je dána nelineární funkce více proměnných  $y = f(x_1, x_2, \dots, x_n)$ , která se má aproximovat fuzzy systémem. Výstupní proměnná je definována na universu  $Y$ , které je pokryto souborem fuzzy množin  $B^i$ . Vstupní proměnné  $x_i$  jsou definovány na univerzech  $X_i$ , která jsou pokryta soubory fuzzy množin  $A_i^j$ . Nelineární funkci  $f$  je možné tedy aproximovat fuzzy systémem se souborem  $r$  pravidel následujícího typu:

$$IF (x_1 = A_1^{jk}) AND \dots AND (x_n = A_n^{jk}) THEN (y = B^{jk}) \quad (1.1)$$

kde  $k$  je číslo pravidla ( $k = 1, 2, \dots, r$ ) a  $r$  je celkový počet všech pravidel. Tento druh fuzzy systému se nazývá fuzzy systém typu Mamdani. V některých literaturách nebo aplikacích se tento typ uvádí jako fuzzy systém s fuzzy závěry. Typickým znakem tohoto typu fuzzy systému je fuzzy výrok v konsekventu každého pravidla. Tento typ fuzzy systému nachází široké využití například v lékařské diagnostice [1, 2].

Mezi přednosti fuzzy systému typu Mamdani patří [2]:

- intuitivnost

- široké možnosti použití
- srozumitelná báze pravidel
- snadné zpracování kvalitativní informace od obsluhy

## 1.4 Fuzzy systém typu Takagi-Sugeno

V této podkapitole je popsán další z neznámějších a nepoužívanějších fuzzy systémů, a to fuzzy systém typu Takagi-Sugeno. Velice často je jeho název zkracován pouze na fuzzy systém typu Sugeno. Tento typ fuzzy systému se od fuzzy systému typu Mamdani odlišuje v konsekventu pravidla. Systém Mamdani má v konsekventu fuzzy výrok, zatímco systém Sugeno má v konsekventu obecně funkci více proměnných. Tvar pravidel je pak následující:

$$IF (x_1 = A_1^{j_k}) AND \dots AND (x_n = A_n^{j_k}) THEN y = f_k(x_1, \dots, x_n) \quad (1.2)$$

kde  $k$  je číslo pravidla ( $k = 1, 2, \dots, r$ ) a  $r$  je celkový počet všech pravidel. Funkce v konsekventu pravidla bývá ve většině případů prostá kombinace vstupních proměnných:

$$f_k(x_1, \dots, x_n) = a_{k_0} + a_{k_1}x_1 + a_{k_2}x_2 + \dots + a_{k_n}x_n \quad k = 1, 2, \dots, r$$

Ostrá výstupní hodnota se získá pomocí váženého průměru funkcí  $f_k$  pro jednotlivá pravidla podle následujícího vzorce:

$$y = \frac{\sum_{k=1}^r w_k f_k(x_1, \dots, x_n)}{\sum_{k=1}^r w_k} \quad (1.3)$$

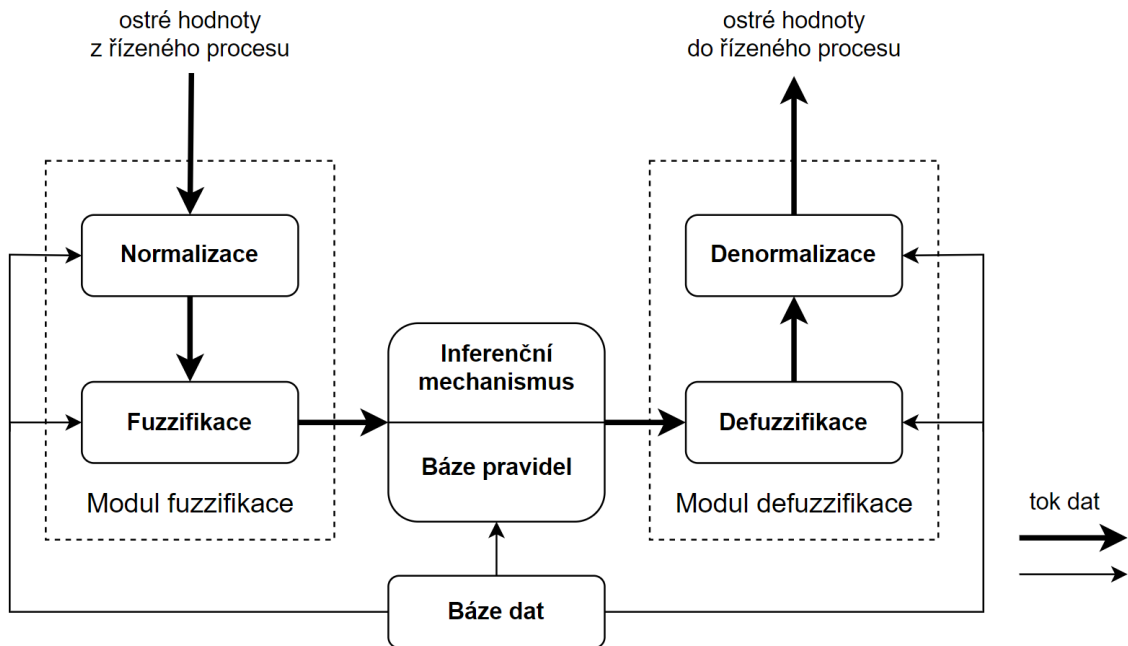
kde váha  $w_k$  je výsledná hodnota antecedentu  $k$ -tého pravidla. Tento systém tak realizuje plynulý přepínač mezi funkcemi  $f_k$  [1].

Mezi výhody fuzzy systému typu Sugeno patří [2]:

- nízká výpočetní náročnost
- umožňuje použití lineárních přístupů (např. PID regulace)
- vhodný pro adaptivní a optimalizační techniky
- výstupní plocha je na spojitém universu spojitá
- vhodný pro matematickou analýzu
- nižší počet pravidel než systém Mamdani

## 1.5 Fuzzy regulátory

Fuzzy regulátory mohou být vhodnou alternativou ke klasickým lineárním regulátorům. Své využití naleznou zejména při regulaci soustav obsahujících nelinearitu. V této kapitole jsou uvedeny regulátory založené na fuzzy systému typu Mamdani. Strukturu fuzzy regulátoru ilustruje následující obrázek 1.6.



Obr. 1.6: Struktura fuzzy regulátoru (převzato z [1])

Jednotlivé bloky byly již popsány v kapitole 1.2 v rámci fuzzy systémů. Zde se navíc nacházejí bloky normalizace a denormalizace. V bloku normalizace dochází na základě tzv. měřítka k přepočtu ostré vstupní hodnoty fyzikální veličiny do normalizovaného univerza. V bloku denormalizace pak dochází přesně k opačné operaci, tj. k převodu do ostré hodnoty fyzikální veličiny [1].

Vstupními veličinami fuzzy regulátoru mohou být [1]:

- odchylka  $e(k) = w - y(k)$
- změna odchylky  $\Delta e(k) = e(k) - e(k - 1)$
- součet odchylky  $\delta e(k) = \sum_{i=1}^k e(k)$

kde  $w$  je požadovaná hodnota a  $y(k)$  je skutečná hodnota na výstupu řízeného regulačního obvodu v kroku  $k$ .

Výstupními veličinami fuzzy regulátoru mohou být [1]:

- akční zásah  $u(k)$
- změna akčního zásahu  $\Delta u(k) = u(k) - u(k - 1)$

Jazyková proměnná (např. odchylka) nabývá hodnot z množiny slovních hodnot:

$$\{NB, NM, NS, ZO, PS, PM, PB\}$$

kde NB = negative big (záporná velká)

NM = negative medium (záporná střední)

NS = negative small (záporná malá)

ZO = zero (nula)

PS = positive small (kladná malá)

PM = positive medium (kladná střední)

PB = positive big (kladná velká)

V následujících podkapitolách jsou popsány základní typy fuzzy regulátorů. Konkrétně se jedná o P, PI a PD regulátor. PID regulátor není zahrnut, jelikož jeho matice pravidel je kubickou maticí a její naplnění je obtížné. Z tohoto důvodu se fuzzy PID regulátor realizuje jako součtově pracující fuzzy PI a fuzzy PD regulátor [1].

### 1.5.1 Fuzzy P regulátor

Fuzzy P regulátor je nejjednodušším typem fuzzy regulátoru. Klasický P regulátor je možné popsat rovnicí

$$u(k) = K_R e(k) \quad (1.4)$$

kde  $K_R$  je zesílení regulátoru. Pravidla regulátoru pak budou v následujícím tvaru [1]:

$$IF (e(k) = \langle hodnota \rangle) THEN (e(k) = \langle hodnota \rangle) \quad (1.5)$$

### 1.5.2 Fuzzy PI regulátor

Fuzzy PI regulátor obsahuje oproti fuzzy P regulátoru navíc integrační složku. Výsledkem diference rovnice klasického diskrétního PI regulátoru je:

$$\Delta u = K_R \Delta e + K_I e \quad (1.6)$$

kde  $K_R$  je zesílení regulátoru a  $K_I$  je integrační zisk. Pravidla regulátoru pak budou v následujícím tvaru [1]:

$$IF (e = \langle hodnota \rangle) AND (\Delta e = \langle hodnota \rangle) THEN (\Delta u(k) = \langle hodnota \rangle) \quad (1.7)$$

Na výstupu fuzzy PI regulátoru je pouze změna akčního zásahu  $\Delta u^*$ . Celkový akční zásah je proto nutné vypočítat zvlášť mimo strukturu regulátoru a to pomocí sumátoru podle následující rovnice [1]:

$$u(k)^* = u(k-1)^* + \Delta u(k)^* \quad (1.8)$$



### 1.5.3 Fuzzy PD regulátor

Fuzzy PD regulátor zahrnuje oproti fuzzy P regulátoru navíc derivační složku. Rovnice klasického diskrétního PD regulátoru je:

$$u(k) = K_R e(k) + K_D \Delta e(k) \quad (1.9)$$

kde  $K_R$  je zesílení regulátoru a  $K_D$  je derivační časová konstanta. Pravidla fuzzy PD regulátoru pak budou v tomto tvaru [1]:

$$IF (e(k) = \langle hodnota \rangle) AND (\Delta e(k) = \langle hodnota \rangle) THEN (u(k) = \langle hodnota \rangle) \quad (1.10)$$

## 2 Popis modelu větrného tunelu

Tato kapitola se zabývá podrobnějším popisem jednotlivých částí modelu větrného tunelu. Ten se skládá z ventilátoru, tubusu a turbíny spojené se snímačem otáček.



Obr. 2.1: Model větrného tunelu

### 2.1 Tubus

Stěny modelu větrného tunelu jsou tvořeny běžnou HT trubkou s vnitřním průměrem DN 75 mm a HT spojkou. Celková délka tunelu je 50 cm.



Obr. 2.2: HT trubka tvořící tubus tunelu (ilustrační obrázek) [5]

### 2.2 Ventilátor

Jako akční člen, který je řízený výstupem fuzzy regulátoru, byl zvolen ventilátor SUNON MagLev ME80252V3-000U-G99 s čtyřpólovým bezkartáčovým DC motorem. Jedná se o jednoduchý ventilátor, jehož parametry jsou uvedeny v tabulce 2.1. Součástí tohoto ventilátoru je i integrovaný snímač otáček.

Tab. 2.1: Technické parametry ventilátoru (převzato z [3])

Parametr	Hodnota
Napájecí napětí	24 V DC
Jmenovitý proud	49 mA
Jmenovitý výkon	1,2 W
Rozsah pracovního napětí	10 V až 27,6 V
Startovací napětí	10 V
Jmenovité otáčky	2 600 otáček/min $\pm 10$ % při napájecím napětí
Rozsah pracovní teploty	-10 °C až 70 °C
Maximální průtok vzduchu	56,07 $m^3 \cdot h^{-1}$
Hlučnost	28 dB(A)
Životnost	70 000 hodin při 40 °C a relativní vlhkosti 65 %
Rozměry	80x80x25 mm
Hmotnost	75 g



Obr. 2.3: Ventilátor Sunon Maglev ME80252V3-000U-G99 [4]

Podobu ventilátoru přibližuje obrázek 2.3. Červený vodič slouží k připojení napájecího napětí 24 V, černý vodič se připojuje k uzemnění (GND) a žlutý vodič je výstup integrovaného snímače otáček.

## 2.3 Turbína a inkrementální snímač otáček

Zpětnou vazbou pro fuzzy regulátor je turbína usazená na hřídeli inkrementálního enkodéru, který slouží jako snímač otáček. V závislosti na počtu otáček turbíny, která je roztáčena proudícím vzduchem, se na výstupu snímače generují pulzy.

Zvoleným snímačem je LPD3806 600BM G5 24C Rotary Encoder. Tento snímač lze využít nejen pro snímání otáček, ale také pro snímání směru otáčení, úhlu natočení a polohy. Jedna otáčka hřídele enkodéru odpovídá 600 pulzům na jeho výstupu. Pro rozlišení směru otáčení snímač disponuje dvěma kanály A a B, jejichž signály jsou fázově posunuty o 90°. Technické parametry inkrementálního enkodéru jsou uvedeny v tabulce 2.2 [6].

Vzhled inkrementálního enkodéru ilustruje obrázek 2.4. Enkodér se připojuje pomocí čtyř vodičů. Červený vodič slouží k připojení napájecího napětí a černý vodič pak k připojení uzemnění (GND). Zelený vodič vede signál kanálu A a bílý vodič pak kanálu B [6].



Obr. 2.4: Inkrementální enkodér LPD3806 600BM G5 24C [6]

Tab. 2.2: Technické parametry inkrementálního enkodéru (převzato z [7])

<b>Parametr</b>	<b>Hodnota</b>
Napájecí napětí	5 V až 24 V DC
Napětová logická úroveň	5 V až 24 V DC
Maximální proud	80 mA
Maximální frekvence signálu	20 kHz
Maximální počet otáček hřídele	5000 otáček/min
Rozlišení	600 pulzů/otáčka
Typ výstupu	NPN
Rozměry enkodéru	38x36 mm
Rozměry hřídele	13x6 mm %

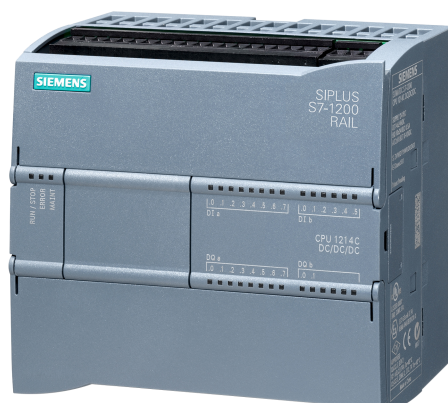
### 3 PLC Siemens S7-1200

Pro proces identifikace modelu větrného tunelu, jehož postup je popsán v kapitole 5, byl využit kompaktní programovatelný logický automat (zkráceně PLC z anglických slov programmable logic controller) od německé firmy Siemens. Byl vybrán model SIMATIC S7-1200 s označením CPU 1214C DC/DC/DC 6ES7214-1AG40-0XB0. Tento typ PLC je vhodný pro realizaci malých a středních automatizačních úloh [8], a proto je pro proces identifikace zcela dostačující. Základní parametry jsou uvedeny v tabulce 3.1.

Tab. 3.1: Vybrané parametry PLC Siemens S7-1200 1214C (převzato z [9])

Parametr	Hodnota
Počet digitálních vstupů	14, z toho 6 pro High Speed Counting (HSC)
Počet digitálních výstupů	10, z toho 4 pro 100 kHz Pulse Train Output
Počet analogových vstupů	2 (10 bitů, rozsah 0 až 10 V DC)
Napájecí napětí	20,4 až 28,8 V DC
Pracovní paměť	100 kB
Rozhraní	Ethernet, PROFINET
Protokol	TCP/IP, PROFIBUS, Modbus, ASI
Firmware	verze 4.4
Čas vykonání bitové operace	0,08 $\mu$ s / operace
Provozní teplota	-20 až 60 °C

Tento automat lze rovněž doplnit o signální, technologické a komunikační moduly, které umožní efektivnější využití celého systému [8].



Obr. 3.1: SIMATIC S7-1200 CPU 1214C 6ES7214-1AG40-0XB0 [10]

## 4 Metody optimalizace

V této kapitole jsou uvedeny vybrané optimalizační metody, jejich popis a algoritmus výpočtu.

### 4.1 Metoda Nelder-Mead

Jedná se o numerickou metodu používanou k nalezení maxima nebo minima účelové funkce ve vícerozměrném prostoru. Byla vytvořena v roce 1965 britskými statistiky Johnem Nelderem a Rogerem Meadem [11].

Metoda Nelder-Mead se řadí mezi metody přímého vyhledávání, jenž jsou založeny na porovnávání funkcí. Je využívána pro nelineární optimalizační problémy, pro které nemusí být známy jejich derivace. Jelikož se však jedná o metodu heuristického vyhledávání, může tato metoda konvergovat i k nestacionárním bodům [11].

Výpočetní algoritmus metody Nelder-Mead používá simplex  $n + 1$  bodů pro  $n$ -rozměrný vektor  $x$ . Simplex je  $n$ -rozměrným zobecněním trojúhelníku, přičemž se jedná o konvexní obal  $n + 1$  afinně nezávislých bodů. Tento obal je pak umístěn v euklidovském prostoru rozměru  $n$  nebo vyšším. V simplexu se metoda pokouší nahradit bod s největší funkční hodnotou novým bodem s menší funkční hodnotou. Nový bod lze dostat pomocí zrcadlení, prodloužení či zkrácení simplexu podél přímky dané vrcholem s největší funkční hodnotou a těžištěm ostatních bodů simplexu. V případě, že se tímto způsobem nenalezne bod s menší funkční hodnotou, zachová se bod s nejmenší funkční hodnotou a celý simplex se zúží (srazí) posunutím všech ostatních vrcholů simplexu směrem k tomuto bodu. Během iterace algoritmu můžou tedy nastat čtyři stavy [12, 13, 14]:

- zrcadlení (angl. reflection)
- prodloužení (angl. expansion)
- zkrácení (angl. contraction)
- sražení (angl. shrink)

Výpočetní algoritmus metody Nelder-Mead je pro větší přehlednost uveden na následující straně (převzat z [11]).

Shrnutí metody [13]:

- efektivní (v každé iteraci se provede 1 až 2 vyhodnocení kritériální funkce)
- jednoduchá implementace
- nepočítá derivace
- nestará se o hladkost funkce
- může proběhnout velký počet iterací
- občas může vyžadovat restart

## CÍL

Nalézt lokální minimum optimalizačního problému

$$\min_{x \in \mathbb{R}} f(x)$$

## VSTUP

Spojité diferencovatelná funkce  $f: \mathbb{R}^n \rightarrow \mathbb{R}$ .

Afinně nezávislé vektory  $x_1^0, \dots, x_{n+1}^0$  z  $\mathbb{R}^n$ , pro které platí, že  $f(x_1^0) \leq f(x_{i+1}^0)$ , kde  $i = 1, 2, \dots, n$ .

Požadovaná přesnost  $\varepsilon \in \mathbb{R}$ ,  $\varepsilon > 0$ .

## VÝSTUP

Aproximace optimálního řešení  $x^* \in \mathbb{R}$ .

## INICIALIZACE

$k := 0$

## CYKLUS

$$x_c := \frac{1}{n} \sum_{i=1}^n x_i^k$$

Urči  $d_k := x_c - x_{n+1}^k$

$$x_r = x_{n+1}^k + 2d_k = 2x_c - x_{n+1}^k$$

**IF**  $f(x_r) < f(x_1^k)$  **THEN** hledej za  $x_r$

$$\dots x_e = x_{n+1}^k + 3d_k = 2x_r - x_c$$

**IF**  $f(x_e) < f(x_r)$  **THEN**  $\hat{x} := x_c$

**ELSE**  $\hat{x} := x_r$

**IF**  $f(x_n^k) > f(x_r) \geq f(x_1^k)$  **THEN**  $\hat{x} := x_r$

**IF**  $f(x_r) \geq f(x_n^k)$  **THEN** hledej před  $x_r$

$$\dots \text{IF } f(x_r) \geq f(x_{n+1}^k) \text{ THEN } \hat{x} := x_{n+1}^k + \frac{1}{2}d_k = \frac{1}{2}(x_{n+1}^k + x_c)$$

$$\dots \text{ELSE } \hat{x} := x_{n+1}^k + \frac{3}{2}d_k = \frac{1}{2}(x_r + x_c)$$

$$x_{n+1}^{k+1} := \hat{x}$$

$$x_i^{k+1} := x_i^k, i = 1, \dots, n$$

$k := k + 1$

Přečísli, aby bylo získáno  $f(x_i^k) \leq f(x_{i+1}^k)$ .  $i = 1, \dots, n$

## CYKLUS OPAKUJ, DOKUD $\|d_k\| \leq \varepsilon$

$$x^* = x_1^k$$



## 4.2 Newtonova metoda a line search

Jedná se o spojení dvou optimalizačních metod – Newtonovy metody a metody line search. Aby bylo možné tuto metodu použít, musí být daná funkce alespoň dvakrát diferencovatelná.

Spojení těchto dvou metod do jednoho výpočetního algoritmu vylepšuje chování metody line search a zároveň potlačuje nežádoucí stavy Newtonovy metody (zejména konvergenci k nesprávnému minimu optimalizované funkce). Výpočetní algoritmus využívá tzv. Choleského rozklad, který umožňuje nahradit Hessián jinou pozitivně definitní maticí. Dochází zde k rozložení matice  $A$  na součin

$$A = LL^T$$

kde  $L$  je regulární dolní trojúhelníková matice s kladnými diagonálními prvky. Způsob vyčíslení faktoru  $L$  se pak nazývá Choleského faktorizace, jehož výpočetní algoritmus vypadá následovně [11, 14]:

### CÍL

Upravit danou matici tak, aby byla pozitivně definitní.

### VSTUP

Symetrická matice  $A \in \mathbb{R}^{n \times n}$ .

### VÝSTUP

Dolní trojúhelníková matice  $L$  a  $\tau \geq 0$  tak, aby  $A + \tau I = LL^T$  byla pozitivně definitní.

### INICIALIZACE

$k := 0$

**IF**  $\min_i a_{ii} > 0$  **THEN**  $\tau_k := 0$

**ELSE**  $\tau_k := \frac{1}{2} \|A\|_F$

### CYKLUS

Vypočítej Choleského faktorizaci  $LL^T$  z  $A + \tau I$ .

**IF** faktorizace není úspěšná **THEN**  $\tau_{k+1} := \max(2\tau_k, \frac{1}{2} \|A\|_F)$

$k := k + 1$

**CYKLUS OPAKUJ, DOKUD není faktorizace úspěšná**

Ještě než bude uveden algoritmus pro výpočet metody line search, je nutné definovat dvě následující podmínky [11]:

$$f(x_k + \alpha_k d_k) \leq f(x_k) + \alpha_k \beta_1 \nabla f(x_k)^T d_k \quad (4.1)$$

kde  $f : \mathbb{R}^n \rightarrow \mathbb{R}$ , bod  $x_k \in \mathbb{R}^n$ , sestupný směr  $d_k \in \mathbb{R}^n$  tak, aby  $\nabla f(x_k)^T d_k < 0$  a krok  $\alpha_k \in \mathbb{R}$ , přičemž  $\alpha_k > 0$  a  $0 < \beta_1 < 1$ . Tato podmínka bývá nazývána jako první Wolfova podmínka nebo také jako Armijovo pravidlo. Pokud platí tato podmínka, je možné říci, že funkce  $f$  klesá v bodě  $x_k + \alpha_k d_k$  oproti bodu  $x_k$ .

$$\nabla f(x_k + \alpha_k d_k)^T d_k \geq \beta_2 \nabla f(x_k)^T d_k \quad (4.2)$$

kde  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  je diferencovatelná funkce, bod  $x_k \in \mathbb{R}^n$ , sestupný směr  $d_k \in \mathbb{R}^n$  tak, aby  $\nabla f(x_k)^T d_k < 0$  a krok  $\alpha_k \in \mathbb{R}$ , přičemž  $\alpha_k > 0$  a  $0 < \beta_2 < 1$ . Tato podmínka bývá nazývána jako druhá Wolfova podmínka. Pokud platí tato podmínka, je možné říci, že funkce  $f$  roste v bodě  $x_k + \alpha_k d_k$  oproti bodu  $x_k$ .

Algoritmus výpočtu pro metodu line search vypadá následovně [11]:

## CÍL

Nalézt takový krok  $\alpha^*$ , který bude vyhovovat podmínkám 4.1 a 4.2.

## VSTUP

Spojitě diferencovatelná funkce  $f : \mathbb{R}^n \rightarrow \mathbb{R}$ .

Gradient funkce  $\nabla f : \mathbb{R}^n \rightarrow \mathbb{R}^n$ .

Vektor  $x \in \mathbb{R}^n$ .

Sestupný směr  $d$  takový, aby platilo  $\nabla f(x)^T d < 0$ .

Inicializační řešení  $\alpha_0 > 0$  (např.  $\alpha_0 = 1$ ).

Parametry  $\beta_1$  a  $\beta_2$  takové,

aby platilo  $0 < \beta_1 < \beta_2 < 1$  (např.  $\beta_1 = 10^{-4}$  a  $\beta_2 = 0,99$ ).

Parametr  $\lambda > 1$  (např.  $\lambda = 2$ ).

## VÝSTUP

Krok  $\alpha^*$ , který bude vyhovovat podmínkám 4.1 a 4.2.

## INICIALIZACE

$i := 0$

$\alpha_i := 0$

$$\alpha_r := +\infty$$

## CYKLUS

**IF**  $\alpha_i$  porušuje podmínku 4.1 **THEN** krok je příliš dlouhý

$$\dots \alpha_r := \alpha_i$$

$$\dots \alpha_{i+1} := \frac{\alpha_l + \alpha_r}{2}$$

**IF**  $\alpha_i$  splňuje podmínku 4.1, ale porušuje 4.2 **THEN** krok je příliš krátký

$$\dots \alpha_l := \alpha_i$$

$$\mathbf{IF} \alpha_r < +\infty \mathbf{THEN} \alpha_{i+1} := \frac{\alpha_l + \alpha_r}{2}$$

$$\mathbf{ELSE} \alpha_{i+1} := \lambda \alpha_i$$

$$i := i + 1$$

## CYKLUS OPAKUJ, DOKUD $\alpha_i$ nesplňuje podmínky 4.1 a 4.2

$$\alpha^* = \alpha^i$$

Nyní již následuje hlavní výpočetní algoritmus této optimalizační metody vzniklé spojením Newtonovy metody a metody line search [11]:

## CÍL

Nalézt aproximaci lokálního minima optimalizačního problému

$$\min_{x \in \mathbb{R}^n} f(x)$$

## VSTUP

Dvkrát diferencovatelná funkce  $f : \mathbb{R}^n \rightarrow \mathbb{R}$ .

Gradient funkce  $\nabla f : \mathbb{R}^n \rightarrow \mathbb{R}$ .

Hessián funkce  $\nabla^2 f : \mathbb{R}^n \rightarrow \mathbb{R}^{n \times n}$ .

Počáteční řešení  $x_0 \in \mathbb{R}^n$

Požadovaná přesnost  $\varepsilon \in \mathbb{R}$ ,  $\varepsilon > 0$ .

## VÝSTUP

Aproximace optimálního řešení  $x^* \in \mathbb{R}$ .

## INICIALIZACE

$$k := 0$$

## CYKLUS

Pomocí Choleského faktorizace vypočti dolní trojúhelníkovou matici  $L_k$  a  $\tau$  tak, aby platilo:  $L_k K_k^T = \nabla^2 f(x_k) + \tau I$

Najdi  $z_k$  řešením trojúhelníkové soustavy  $L_k z_k = \nabla f(x_k)$ .

Najdi  $d_k$  řešením trojúhelníkové soustavy  $L_k^T d_k = -z_k$ .

Urči  $\alpha_k$  pomocí algoritmu metody line search s počáteční hodnotou  $\alpha_0 = 1$ .

$$x_{k+1} := x_k + \alpha_k d_k$$

$$k := k + 1$$

**CYKLUS OPAKUJ, DOKUD  $\|\nabla f(x_k)\| \leq \varepsilon$**

$$x^* = x^k$$

### 4.3 Metoda zlatého řezu

Metoda zlatého řezu hledá minimum funkce jedné proměnné na daném intervalu. Tato funkce musí být unimodální, což znamená, že funkce má na daném intervalu právě jeden lokální extrém [14].

Jedná se o metodu, která nevyžaduje výpočet derivací. Metoda zlatého řezu je založena na rozdělení intervalu na dva nové intervaly tak, aby poměr větší části k menší části byl shodný jako poměr původního intervalu k větší části. Metoda tak vytváří posloupnost intervalů:

$$I_0 \supset I_1 \supset \dots \supset I_{k-1} \supset I_k \supset \dots, \quad I_k = \langle a_k, b_k \rangle.$$

Hledané minimum zadané funkce se vyskytuje ve všech vytvořených intervalech [14].

Pokud platí, že funkce  $f$  je unimodální v  $\langle a_{k-1}, b_{k-1} \rangle$  a nabývá svého minima v  $x^* \in \langle a_{k-1}, b_{k-1} \rangle$ , pak pro libovolné  $x_1$  a  $x_2$ , kdy  $a_{k-1} \leq x_1 \leq x_2 \leq b_{k-1}$ , platí některé z těchto tvrzení:

- Pokud je  $f(x_1) < f(x_2)$ , pak  $x^* < x_2$ .
- Pokud je  $f(x_1) > f(x_2)$ , pak  $x^* > x_2$ .
- Pokud je  $f(x_1) = f(x_2)$ , pak  $x_1 < x^* < x_2$ .

Budou-li poté zvoleny v intervalu  $I_{k-1} = \langle a_{k-1}, b_{k-1} \rangle$  body:

$$d_{k-1} := \frac{b_{k-1} - a_{k-1}}{\frac{1+\sqrt{5}}{2}} + a_{k-1}$$

$$c_{k-1} := c_{k-1} + b_{k-1} - d_{k-1}$$

lze následně po porovnání jejich funkčních hodnot a užití některého z výše uvedených tvrzení získat nový interval  $I_k$ . Algoritmus končí ve chvíli, kdy je splněna podmínka, že velikost intervalu  $I_k$  je menší než žádaná hodnota. Výsledné minimum dané funkce se získá jako střed tohoto intervalu [14].

Algoritmus metody zlatého řezu vypadá následovně [11]:

## CÍL

Nalézt aproximaci globálního minima funkce  $f(x)$  na intervalu  $\langle a, b \rangle$ .

## VSTUP

Interval  $[a, b] \supset \mathbb{R}$ .

Funkce  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  unimodální na intervalu  $[a, b]$ .

Požadovaná přesnost  $\varepsilon > 0$ .

## VÝSTUP

Aproximace globálního minima  $x^*$  funkce  $f(x)$  na intervalu  $\langle a, b \rangle$ .

## INICIALIZACE

$k := 1$

$a_1 := a$

$b_1 := b$

$\rho := \frac{3-\sqrt{5}}{2}$

$x_1^1 := a_1 + \rho(b_1 - a_1)$

$x_2^1 := b_1 - \rho(b_1 - a_1)$

## CYKLUS

**IF**  $f(x_1^k) = f(x_2^k)$  **THEN**

...  $a_{k+1} := x_1^k$

...  $b_{k+1} := x_2^k$

...  $a_1^{k+1} := a_{k+1} + \rho(b_{k+1} - a_{k+1})$

...  $a_2^{k+1} := b_{k+1} - \rho(b_{k+1} - a_{k+1})$

**ELSEIF**  $f(x_1^k) > f(x_2^k)$  **THEN**

...  $a_{k+1} := x_1^k$

...  $b_{k+1} := b_k$

...  $a_1^{k+1} := x_2^k$

...  $a_2^{k+1} := b_{k+1} - \rho(b_{k+1} - a_{k+1})$

**ELSE**

...  $a_{k+1} := a_k$

...  $b_{k+1} := x_2^k$

...  $a_1^{k+1} := a_{k+1} + \rho(b_{k+1} - a_{k+1})$

...  $a_2^{k+1} := x_1^k$

$$k := k + 1$$

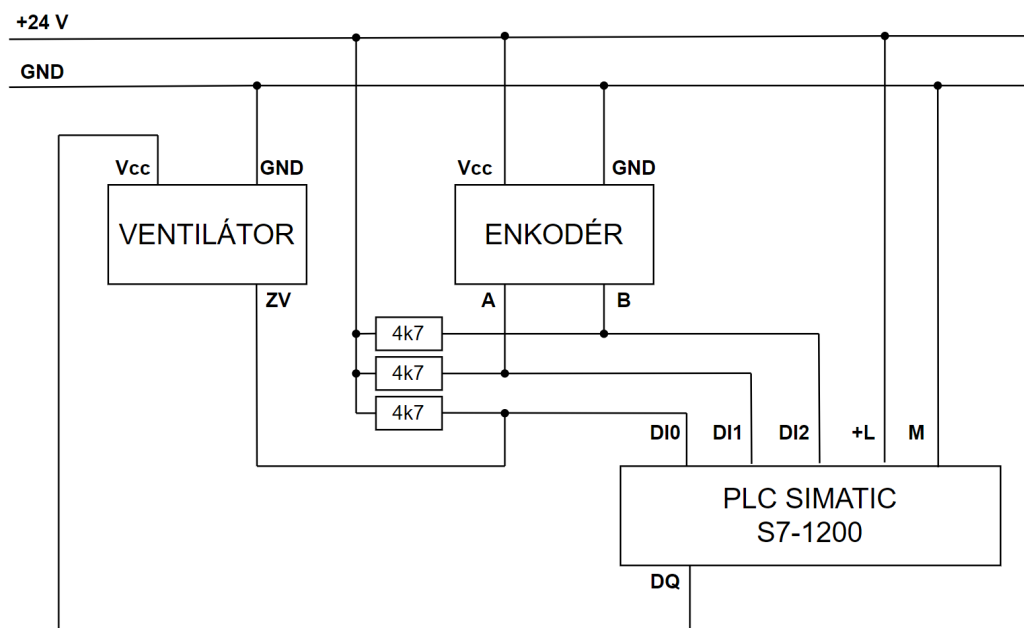
**CYKLUS OPAKUJ, DOKUD  $b_k - a_k \leq \varepsilon$**

$$x^* = \frac{a_k + b_k}{2}$$

## 5 Identifikace větrného tunelu

Tato kapitola přibližuje postup při identifikaci větrného tunelu prostřednictvím programovatelného logického automatu Siemens SIMATIC S7-1200. Dále byly využity programy TIA Portal a MATLAB.

Proces identifikace byl započat propojením jednotlivých částí větrného tunelu s programovatelným automatem. Schéma zapojení je znázorněno na obrázku 5.1.

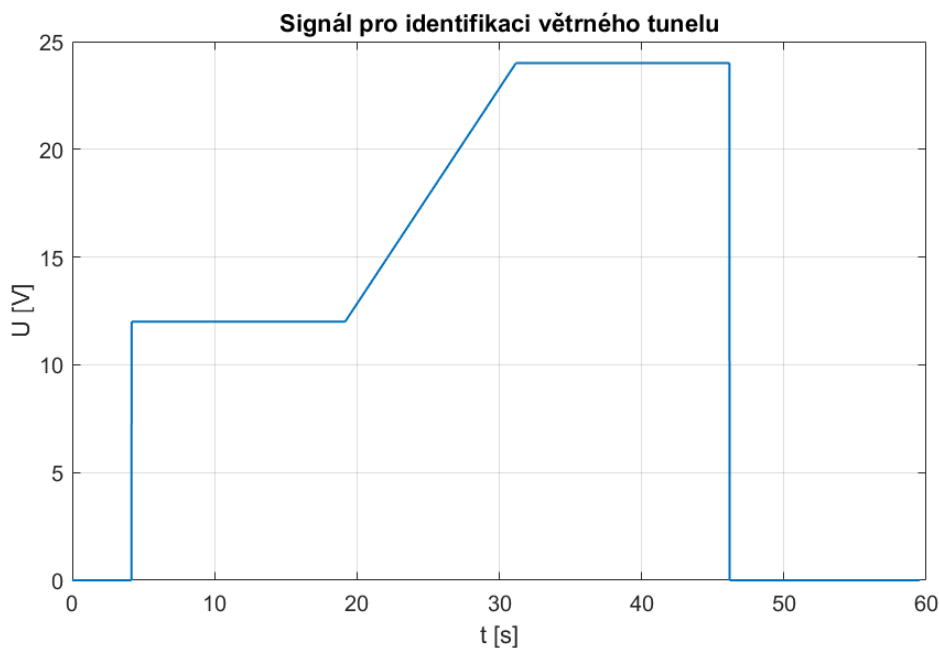


Obr. 5.1: Schéma zapojení větrného tunelu pro identifikaci

Trojice rezistorů s nominální hodnotou elektrického odporu  $4\,700\ \Omega$  slouží k potlačení vlivu nežádoucího šumu. Výstup ZV u ventilátoru je integrovaný snímač otáček a výstupy A a B u enkodéru jsou jeho výstupní kanály vzájemně posunuté o  $90^\circ$ .

### 5.1 TIA Portal

Po propojení větrného tunelu s PLC se proces identifikace přesunul do programu TIA Portal verze 15.1. Zde byly vygenerovány dva signály. První z nich byl určen pro samotnou identifikaci a druhý pak pro následnou validaci získaného modelu. Vzorkovací perioda byla zvolena na 10 ms. Časový průběh identifikačního signálu je znázorněn na obrázku 5.2.



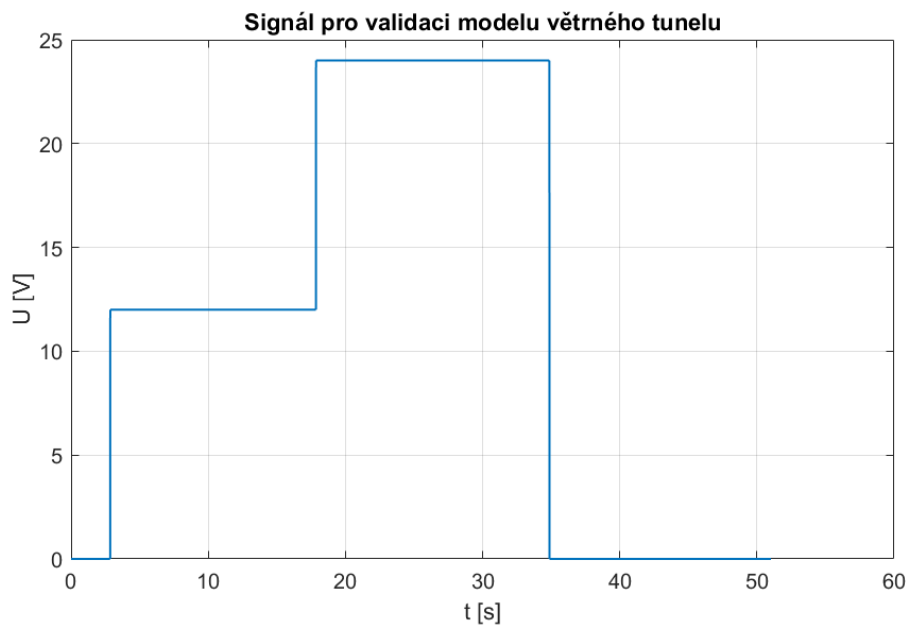
Obr. 5.2: Časový průběh signálu pro identifikaci

Signál obsahuje rostoucí lineární část, díky níž bylo možné vhodně určit nelinearitu, která je obsažena v modelu. Tato rostoucí lineární část signálu začíná na hodnotě napětí 12 V a končí na hodnotě 24 V. Důvodem volby tohoto intervalu je skutečnost, že se inkrementální enkodér začíná samovolně roztáčet až od hodnoty napětí 12 V na ventilátoru. Při nižších hodnotách se enkodér neroztočí bez vnější pomoci. Horní hranice intervalu je pak určena výrobcem jako hodnota napájecího napětí ventilátoru.

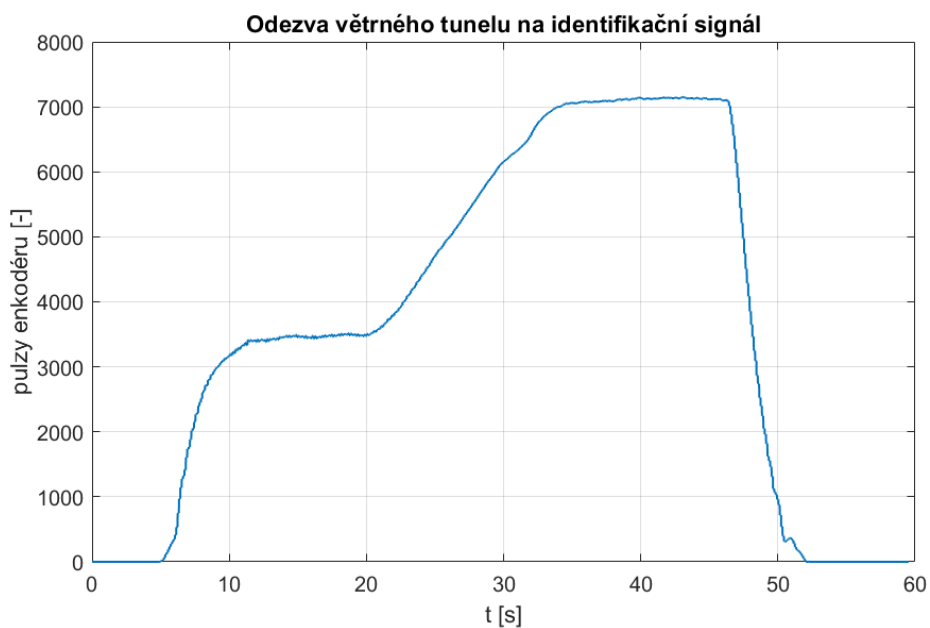
Časový průběh validačního signálu zachycuje obrázek 5.3. Tento signál obsahuje skokovou změnu hodnoty z 12 V na 24 V. Důvod volby těchto hodnot je stejný jako tomu bylo u předešlého signálu.

Následně byla za pomoci funkce Traces, jež je integrována přímo v programu TIA Portal, získána odezva větrného tunelu na vstupní identifikační signál. Vzorovací perioda byla opět zvolena na 10 ms. Časový průběh takto získané odezvy je zobrazen na obrázku 5.4.





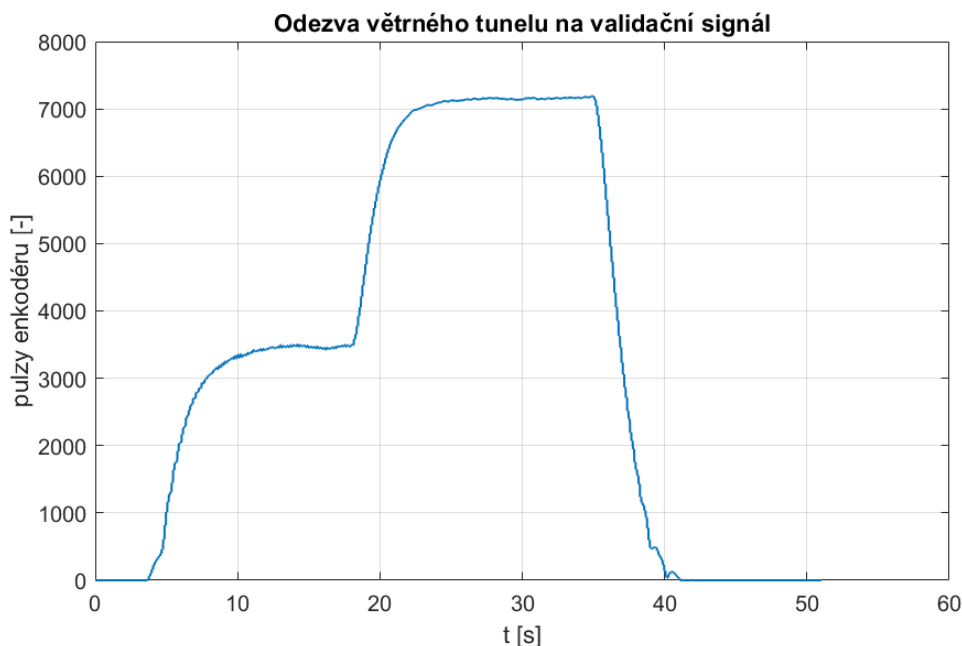
Obr. 5.3: Časový průběh signálu pro verifikaci



Obr. 5.4: Odezva větrného tunelu na identifikační signál

Hodnoty této odezvy jsou dány počtem pulzů inkrementálního enkodéru. Jedna celá otáčka hřídele enkodéru je definována jako 600 pulzů na jeho výstupu.

Dále byla také změřena odezva větrného tunelu na validační signál. Průběh této odezvy zachycuje obrázek 5.5.



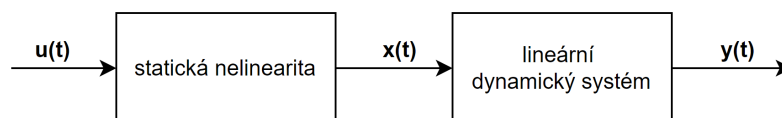
Obr. 5.5: Odezva větrného tunelu na verifikační signál

Tato data byla poskytnuta panem Ing. Tomášem Sýkorou.

## 5.2 MATLAB a System Identification Toolbox

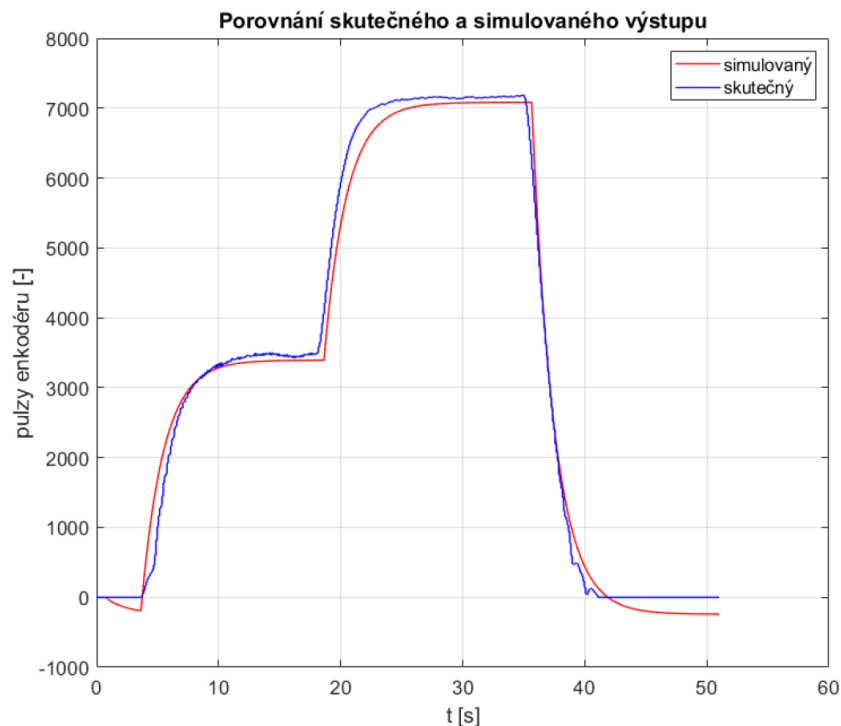
Následně se proces identifikace přesunul do programu MATLAB verze 2021a. Zde se prvně načetla a předzpracovala data získaná v programu TIA Portal. Předzpracování spočívalo v rozřazení dat do příslušných vektorů, s nimiž bylo následně pracováno.

Ještě před samotnou identifikací je vhodné si ujasnit, k jakému výsledku by identifikace měla vést. Větrný tunel se skládá z ventilátoru a otočného inkrementálního enkodéru, přičemž jsou tyto prvky od sebe vzdáleny přibližně 45 cm. Na základě této skutečnosti lze říci, že systém bude druhého řádu s dopravním zpožděním. Větrný tunel se tak skládá z nelineární a lineární části. Tento typ systému se nazývá Hammersteinův model. Struktura tohoto modelu je zobrazena na obrázku 5.6, kde  $x(t)$  je vnitřní proměnná stejné dimenze jako vstupní proměnná  $u(t)$ .



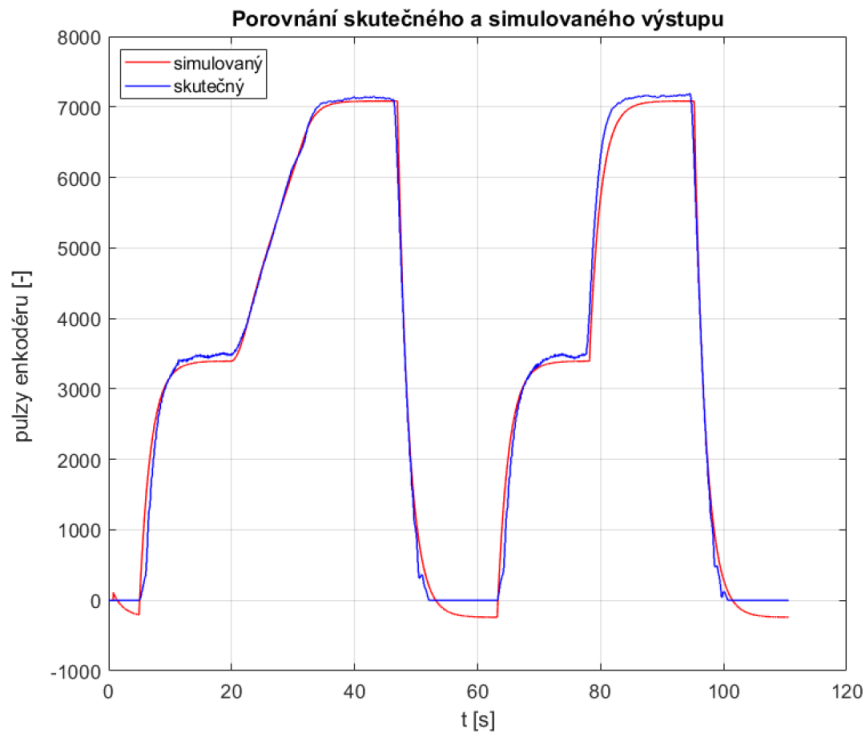
Obr. 5.6: Struktura systému typu Hammerstein [15]

Samotná identifikace probíhala v System Identification Toolboxu, který právě umožňuje identifikaci systému typu Hammerstein. Prvním krokem v identifikačním toolboxu byl import dat z Workspace. Jako učící data byl zvolen identifikační signál s rostoucí lineární částí, jehož průběh byl ilustrován na obrázku 5.2. Jako validační data byl vybrán druhý signál se skokovou změnou, jehož průběh byl zachycen na obrázku 5.3. Následně byly nastaveny parametry estimátoru. Nelinearita byla zvolena jako po částech lineární se třemi body zlomu, tj. s čtyřmi lineárními částmi. Dále byl nastaven počet nul lineární části větrného tunelu na hodnotu 1, počet pólů na hodnotu 2 a velikost dopravního zpoždění na 79 vzorků. Velikost dopravního zpoždění byla určena na základě průběhu vstupního a výstupního signálu. S tímto nastavením byl získán model, jehož přesnost odezvy v porovnání se skutečnou odezvou dosáhla 91,2 %. Průběh obou odezvy je zobrazen na obrázku 5.7.



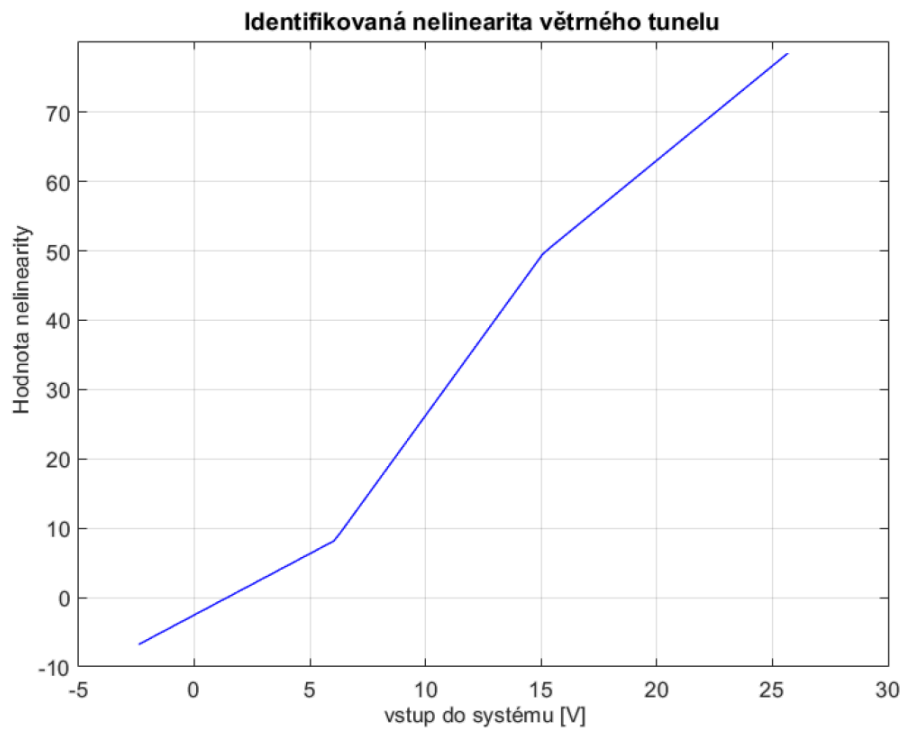
Obr. 5.7: Porovnání skutečné odezvy a odezvy identifikovaného modelu

Dále byl model otestován na signálu vzniklém sjednocením identifikačního a validačního signálu do jednoho. Tento způsob testování však není zcela validní, jelikož část signálu tvoří data, jež byla použita pro učení modelu. Z tohoto důvodu bylo toto otestování provedeno jen pro zajímavost. Bylo zde dosaženo přesnosti 91,99 %. Průběh odezvy je zachycen na obrázku 5.8.

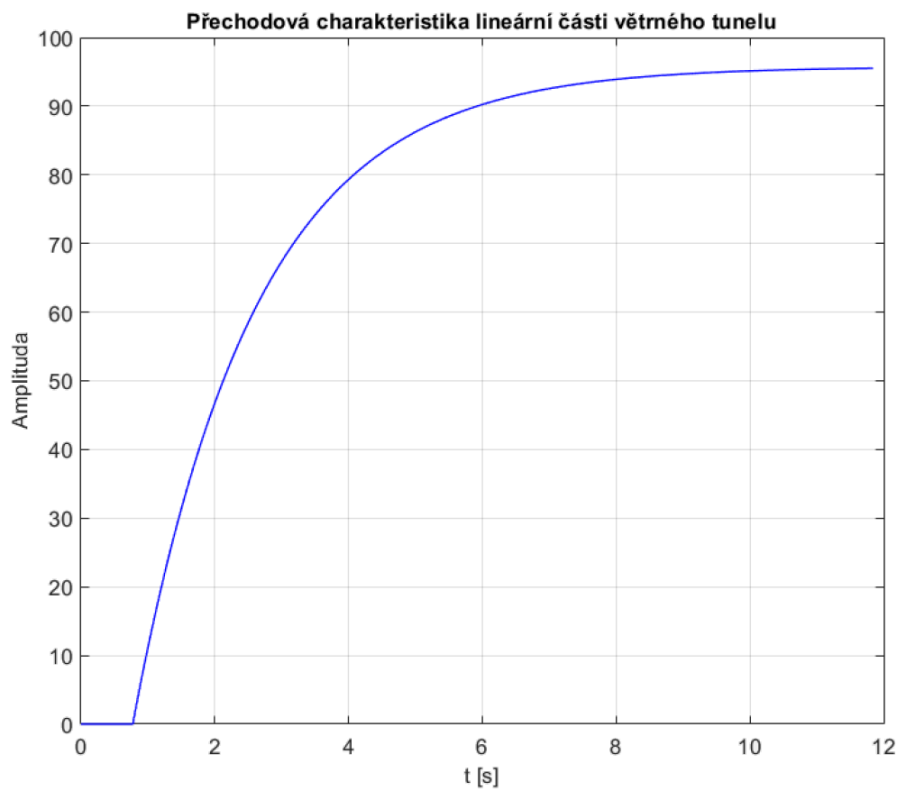


Obr. 5.8: Porovnání skutečné odezvy a odezvy identifikovaného modelu - validační

Průběh identifikované nelinearity modelu větrného tunelu je zachycen na obrázku 5.9. Vodorovná osa odpovídá hodnotě napětí vstupujícího do systému, svislá osa pak udává hodnotu nelinearity. Přechodovou charakteristiku identifikované lineární části větrného tunelu zachycuje obrázek 5.10.



Obr. 5.9: Identifikovaná nelinearita větrného tunelu

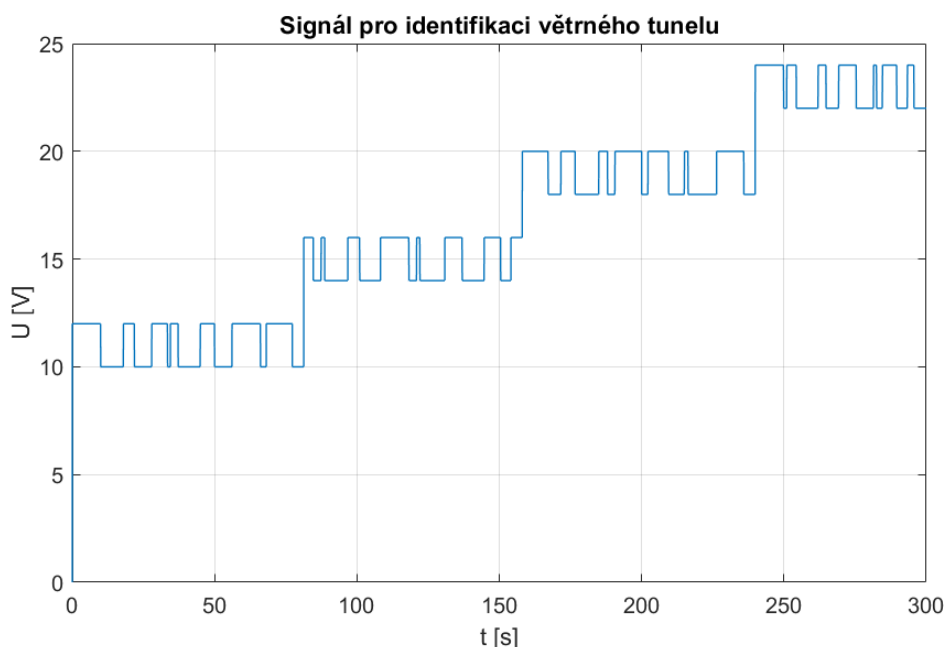


Obr. 5.10: Přechodová charakteristika identifikované lineární části modelu

### 5.3 Nová identifikace větrného tunelu

Při pozdější implementaci optimalizační funkce pro fuzzy regulátor a jeho následného testování v Simulinku byla provedená identifikace větrného tunelu shledána jako nedostatečná. Vystala potřeba reálnou soustavu identifikovat ve více pracovních bodech a zároveň dosáhnout větší shody mezi identifikovaným virtuálním modelem a reálnou soustavou. Z tohoto důvodu bylo přistoupeno k nové identifikaci pomocí PRBS signálu. Zapojení obvodu zůstalo totožné jako při předešlé identifikaci (viz obrázek 5.1).

V programu TIA Portal verze 15.1 byly vygenerovány dva signály - identifikační a validační. Hodnota vzorkovací periody zůstala 10 ms. Časový průběh identifikačního signálu je znázorněn na obrázku 5.11.

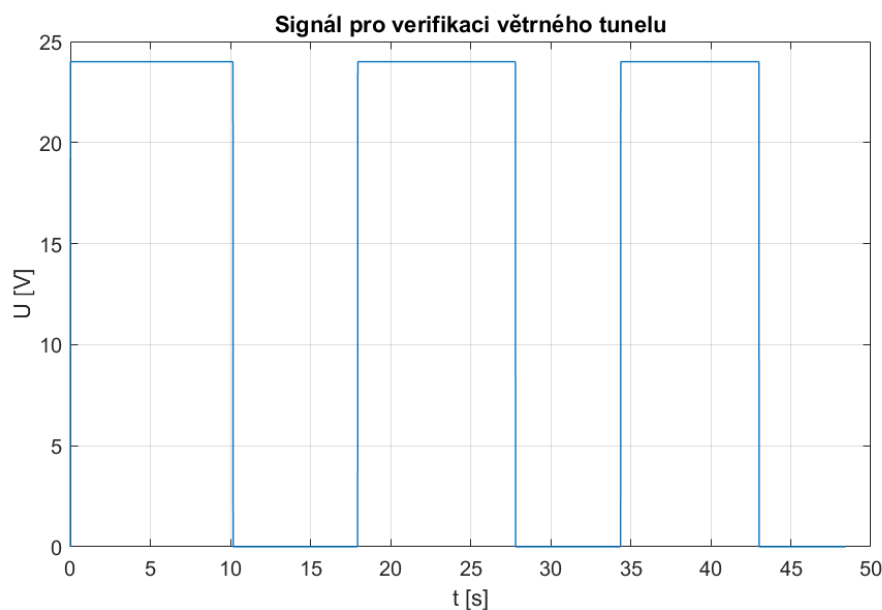


Obr. 5.11: Časový průběh signálu pro identifikaci

Signál má průběh binární pseudonáhodné posloupnosti (často označována pod zkratkou **PRBS** vycházející z anglických slov Pseudo Random Binary Sequence). Aby bylo možné vhodně určit danou nelinearitu, která je obsažena v modelu větrného tunelu, byl tento signál vygenerován ve čtyřech různých napěťových úrovních (10 až 12 V, 14 až 16 V, 18 až 20 V a 22 až 24 V). Odstup mezi mezními hodnotami jednotlivých napěťových úrovní byl zvolen s ohledem na úroveň šumu. Mezní hodnoty nejnižší napěťové úrovně byly nastaveny na 10 a 12 V. Důvodem volby těchto hodnot byla skutečnost, že se enkodér začíná samovolně točit až od hodnoty napětí na ventilátoru 8 V. Při nižších hodnotách se enkodér neroztočí bez další vnější

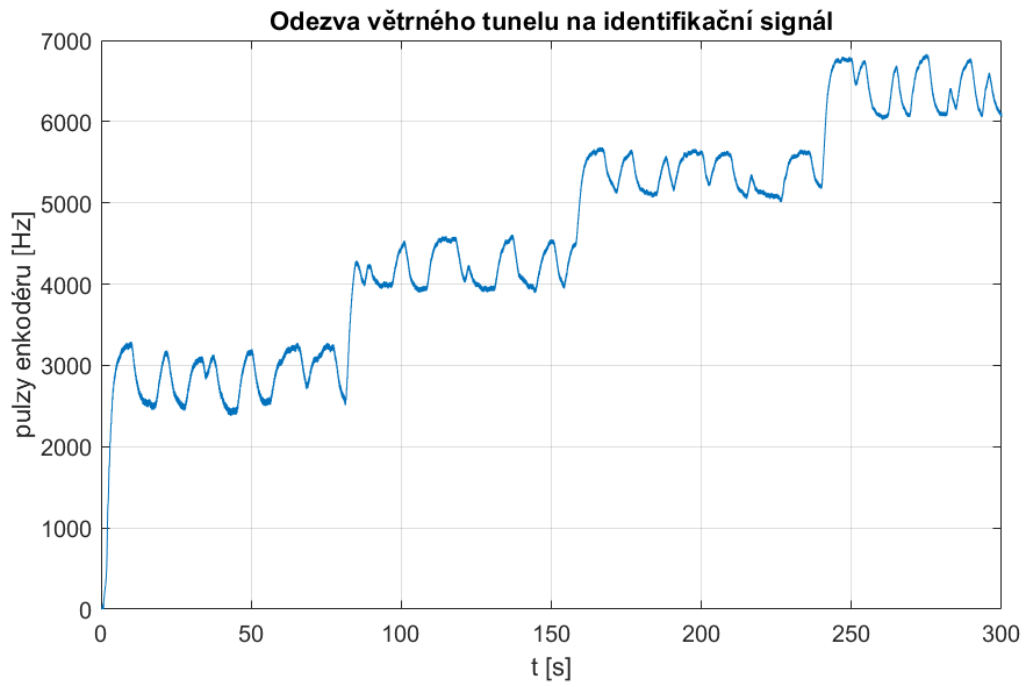
pomoci. Dalším důvod, proč nebyla zvolena jako dolní mez hodnota 8 V, byl ten, že při hodnotách napětí 8 až 10 V docházelo ke značnému kolísání výstupní hodnoty enkodéru způsobené fyzickou realizací samotné hřídele. Horní hranice nejvyšší napěťové úrovně pak byla určena výrobcem ventilátoru jako hodnota napájecího napětí. Ačkoliv je ventilátor schopný pracovat až do napětí 27,6 V, bylo z bezpečnostních důvodů zvoleno nižší napětí 24 V.

Časový průběh validačního signálu zachycuje obrázek 5.12. Tento signál obsahuje skokové změny hodnoty z 0 V na 24 V. Důvod volby těchto hodnot bylo zjištění chování modelu při maximální změně vstupní hodnoty.

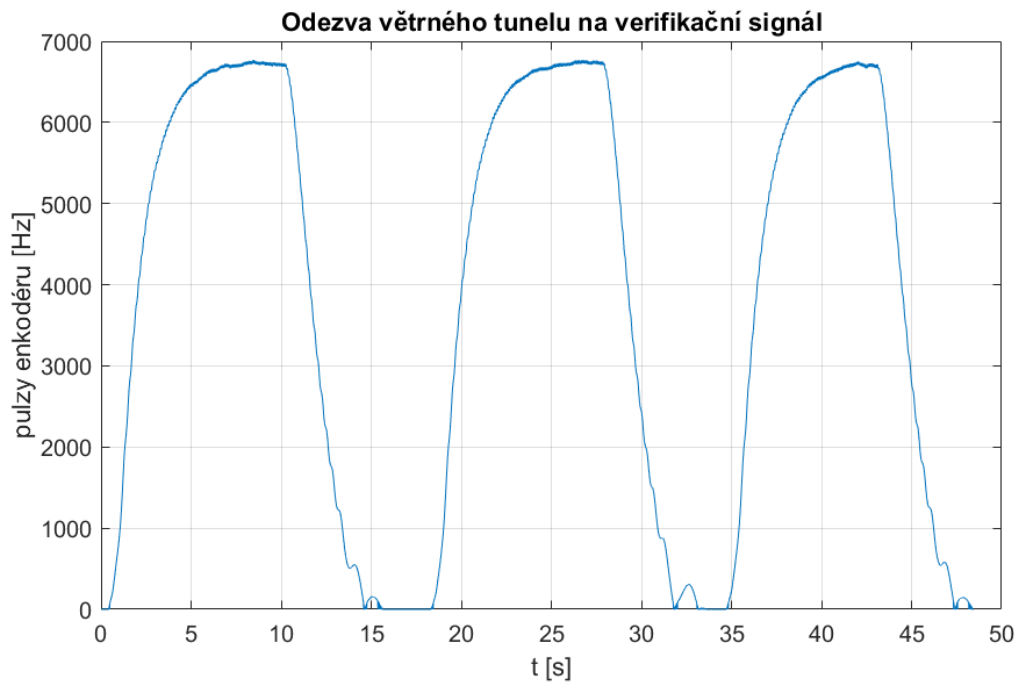


Obr. 5.12: Časový průběh signálu pro verifikaci

Následně byla za pomoci funkce Traces získána odezva větrného tunelu na vstupní identifikační signál. Data byla vzorkována opět s periodou 10 ms. Časový průběh takto získané odezvy je zobrazen na obrázku 5.13. Dále byla také změřena odezva větrného tunelu na validační signál. Průběh této odezvy zachycuje obrázek 5.14. Veškerá vygenerovaná a naměřená data byla uložena do CSV souborů.



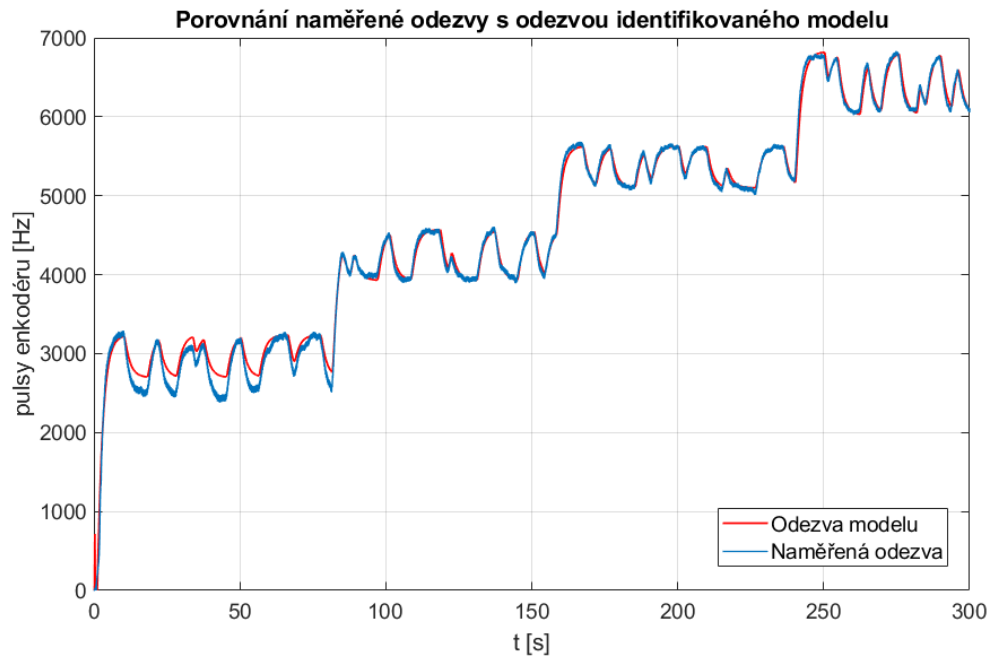
Obr. 5.13: Odezva větrného tunelu na identifikační signál



Obr. 5.14: Odezva větrného tunelu na verifikační signál

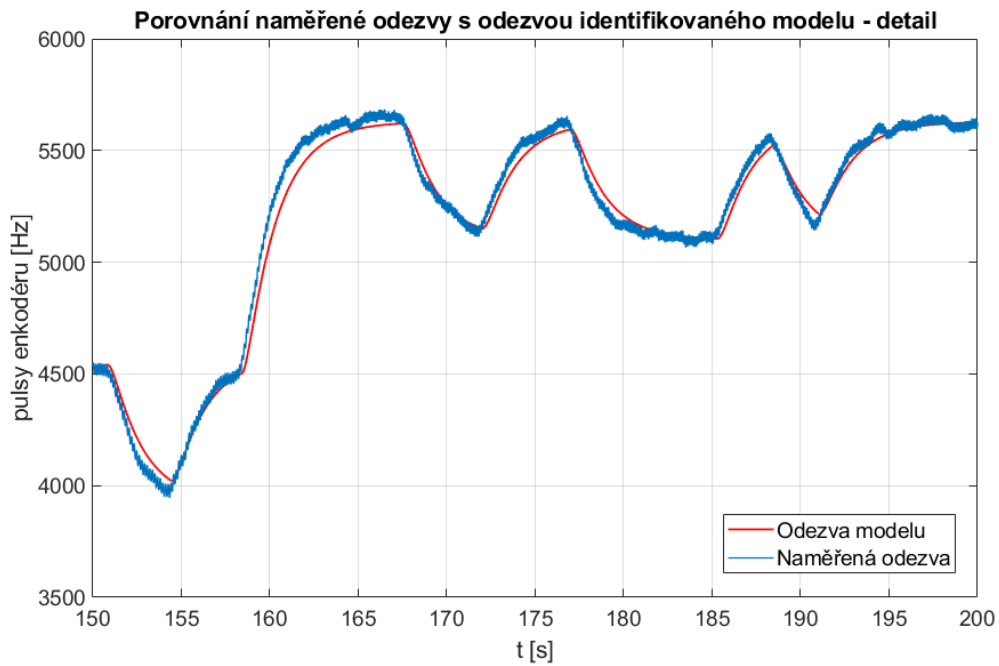


V System Identification Toolboxu v MATLABu byl jako učící data zvolen identifikační PRBS signál se čtyřmi napěťovými úrovněmi, jehož průběh byl ilustrován na obrázku 5.11. Jako validační data byl vybrán druhý signál se skokovou změnou, jehož průběh byl zachycen na obrázku 5.12. Následně byly nastaveny parametry estimátoru. Nelinearita byla zvolena jako po částech lineární se třemi body zlomu, tj. s čtyřmi lineárními částmi. Dále byl nastaven počet nul lineární části větrného tunelu na hodnotu 1, počet pólů na hodnotu 2 a velikost dopravního zpoždění na 35 vzorků. Velikost dopravního zpoždění byla určena na základě průběhu vstupního a výstupního signálu. S tímto nastavením byl získán model, jehož přesnost odezvy v porovnání se skutečnou odezvou dosáhla 93,2 %. Tato hodnota byla získána při porovnání na identifikačním signálu. Průběh obou odezev je zobrazen na obrázku 5.15.



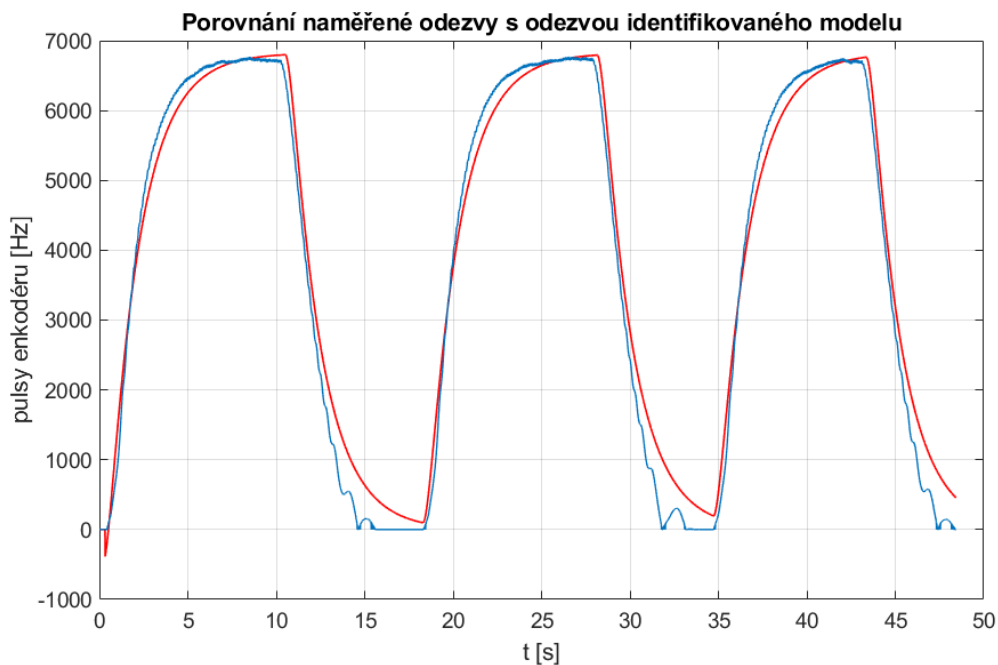
Obr. 5.15: Porovnání skutečné odezvy a odezvy identifikovaného modelu

Z obrázku 5.15 je patrné, že nejvyšší nepřesnosti dosáhl model v první napěťové úrovni 10 až 12 V. Detailnější pohled je pak zachycen na obrázku 5.16. Zde jsou vykresleny průběhy pro třetí napěťovou úroveň 18 až 20 V.



Obr. 5.16: Porovnání skutečné odezvy a odezvy identifikovaného modelu v detailu

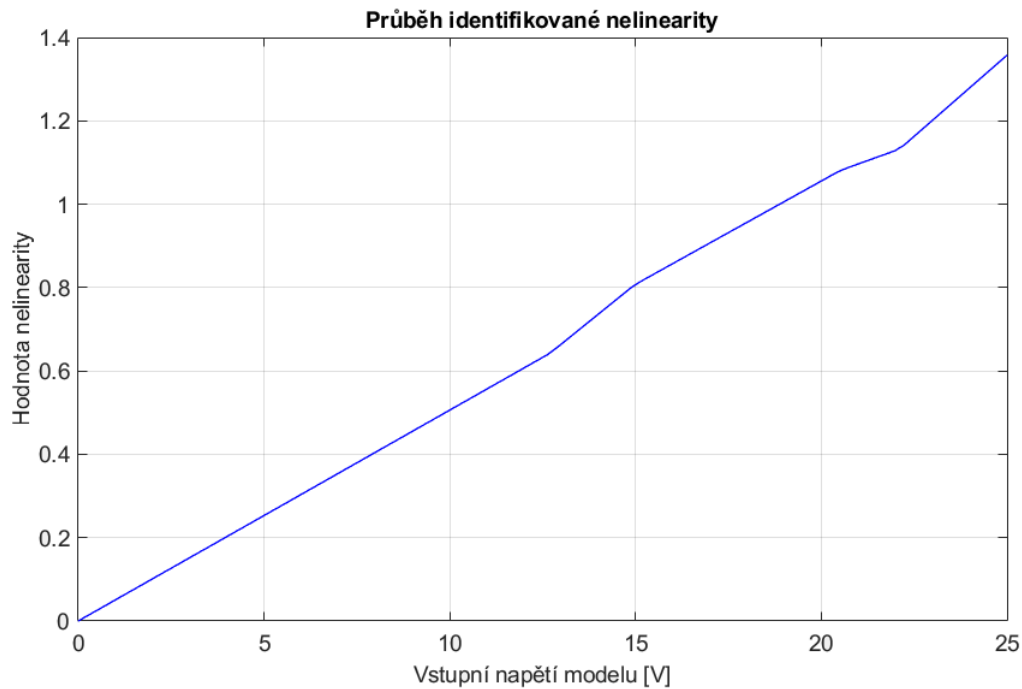
Na obrázku 5.17 je zobrazen průběh odezev identifikovaného modelu a reálného systému na validační signál.



Obr. 5.17: Porovnání skutečné odezvy a odezvy identifikovaného modelu - validační

Zde bylo dosaženo přesnosti 87,0 %. Hodnota je oproti předešlé hodnotě o přibližně 6 % nižší, jelikož předešlá hodnota byla získána na signálu, který byl použit i pro učení modelu. Dalším aspektem, který mohl ovlivnit výslednou hodnotu přesnosti identifikovaného modelu, je také samotný průběh validačního signálu. Ten má skokový průběh z 0 V na 24 V, přičemž identifikovaný model byl učen až od úrovně 10 V.

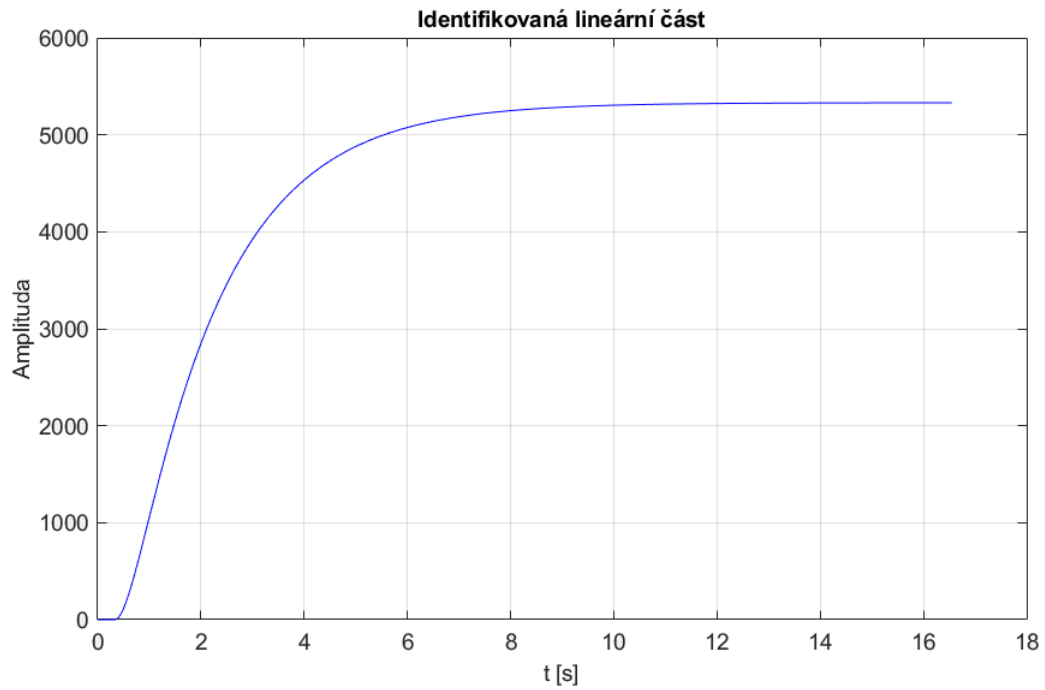
Průběh a tvar identifikované nelinearity modelu větrného tunelu je zachycen na obrázku 5.18.



Obr. 5.18: Identifikovaná nelinearita větrného tunelu

Do identifikované nelinearity byl ručně umístěn bod  $[0;0]$ , a to z důvodu malé nepřesnosti identifikace v tomto bodě, kdy identifikovaný průběh nelinearity dosahoval pro vstupní napětí 0 V hodnoty  $-0,2$ . Tímto krokem nedošlo k prohřešku vzhledem k reálnému systému, jelikož při nulovém napájecím napětí ventilátoru nedochází k otáčení hřídele enkodéru.

Přechodovou charakteristiku identifikované lineární části větrného tunelu zachycuje obrázek 5.19.



Obr. 5.19: Přejchodová charakteristika identifikované lineární části modelu

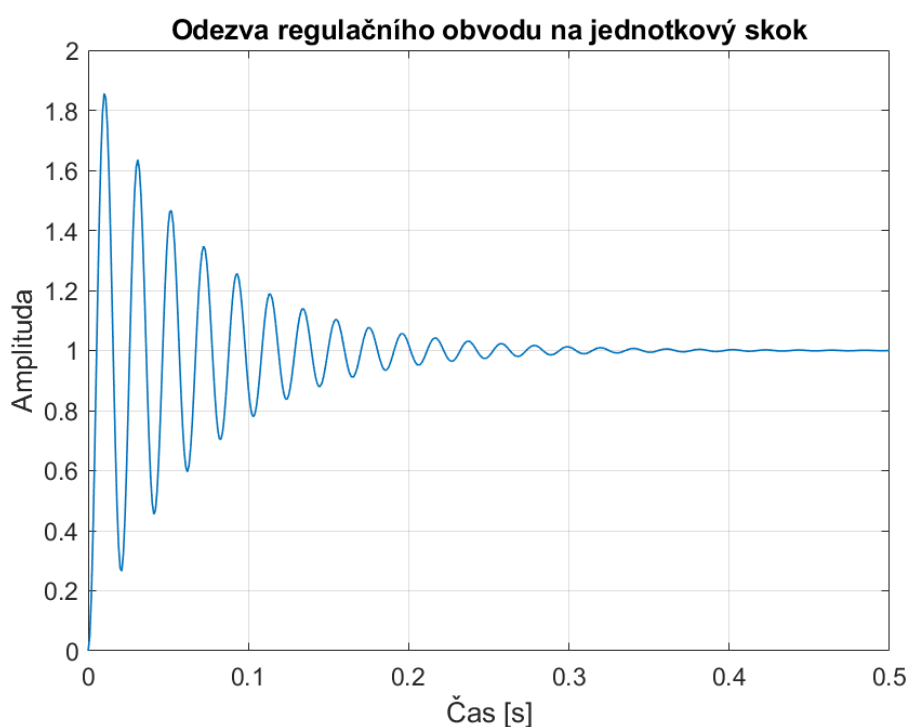
Vyšší přesnosti identifikace větrného tunelu by bylo možné dosáhnout delším časem identifikačního signálu, kdy by v každé ze čtyř napěťových úrovní došlo k zopakování periody signálu. Toto však nebylo možné zrealizovat z důvodu omezené paměti funkce Trace v programu TIA Portal, která je pro vzorkovací periodu 0,01 s omezena na 32 768 vzorků, tj. přibližně 327 sekund identifikačního signálu. Navíc s rostoucím počtem zaznamenávaných signálů se tato doba zmenšuje.

## 6 Manuální návrh fuzzy regulátoru

Tato kapitola přibližuje postup při návrhu fuzzy regulátoru pro řízení modelu větrného tunelu. Návrh probíhal v prostředích MATLAB a Simulink. V programu MATLAB byl využit Fuzzy Toolbox. Byl zvolen fuzzy PI regulátor, jelikož je zcela dostačující a zároveň je i jednodušší na implementaci než fuzzy PID regulátor. S jednodušším fuzzy P regulátorem nebylo při regulaci dosaženo požadované hodnoty.

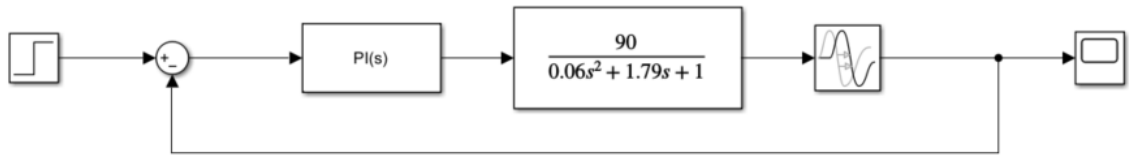
### 6.1 Návrh konstant regulátoru

Návrh započal určením konstant fuzzy regulátoru – zesílením  $K$  a integrační časové konstanty  $T_I$ . Bylo zvoleno ITAE kritérium, jehož výpočet byl implementován ve skriptu. Snahou byla minimalizace regulační odchylky. Výsledná získaná hodnota zesílení byla přibližně 35, což však zapříčinilo tlumené kmity v regulačním ději regulačního obvodu. Tento průběh je zobrazen na obrázku 6.1.



Obr. 6.1: Přenos řízení obvodu s konstantami regulátoru podle ITAE kritéria

Z tohoto důvodu bylo přistoupeno k určení konstant fuzzy regulátoru pomocí Simulinku. Zde byl nakreslen regulační obvod s uzavřenou smyčkou se zjednodušeným modelem identifikované soustavy (s pouze lineární částí), jak je znázorněno na obrázku 6.2.



Obr. 6.2: Regulační obvod pro určení konstant regulátoru

Konstanty byly poté určeny za pomoci bloku PID Controller, kde byl kladen důraz rychlost regulace a současně na minimální překmit nad požadovanou hodnotu. Navržené konstanty jsou:

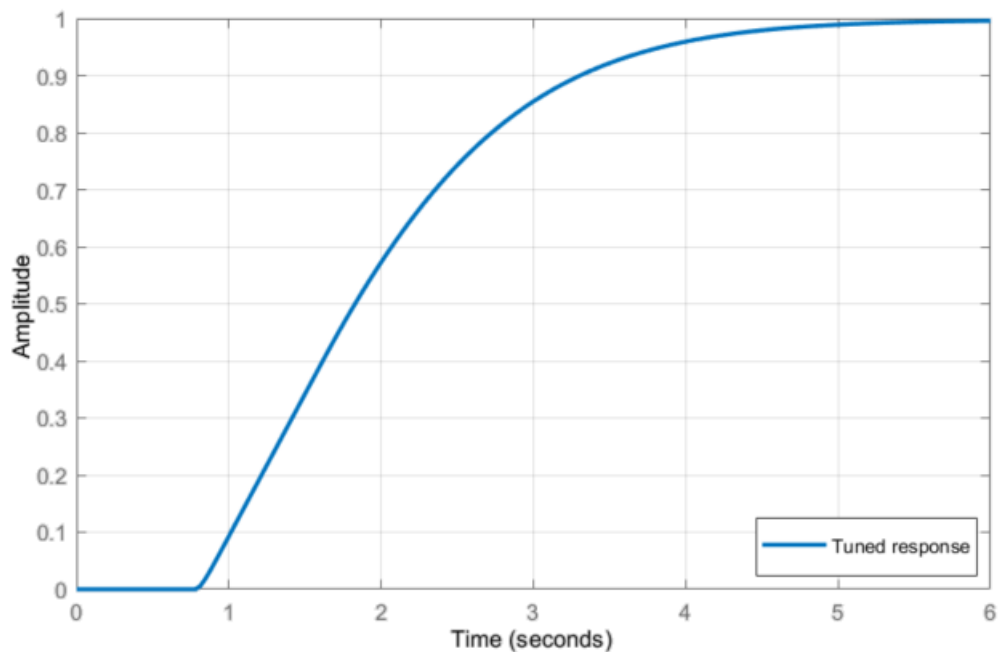
$$K = 0,005451$$

$$T_I = 1,793$$

Tyto konstanty jsou platné jen pro operátorový přenos ideálního PI regulátoru ve tvaru:

$$F_R(p) = K \frac{T_I p + 1}{p}$$

Výsledná odezva tohoto regulačního obvodu na jednotkový skok je zobrazena na obrázku 6.3.

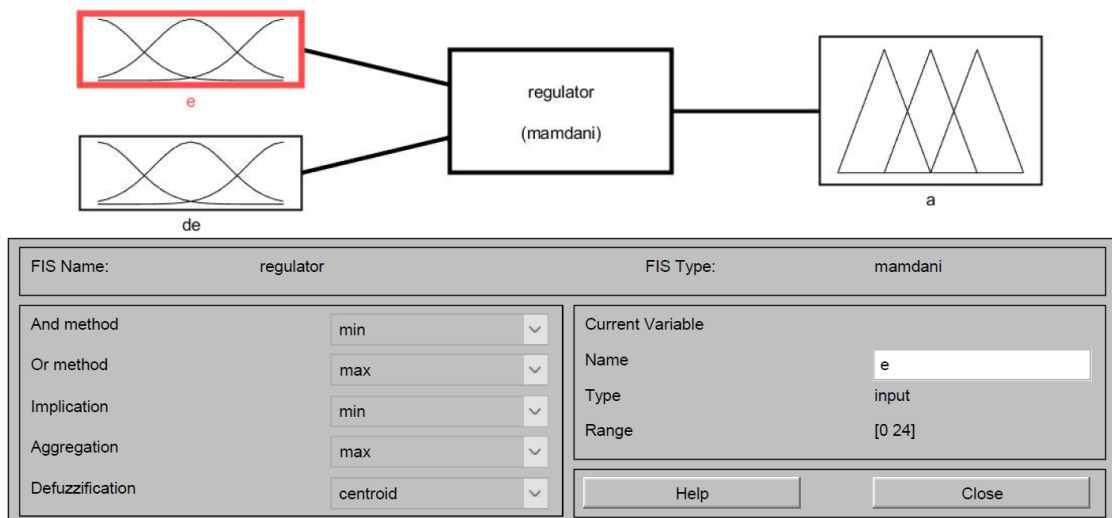


Obr. 6.3: Odezva obvodu s PID Controllerem na jednotkový skok

## 6.2 Návrh fuzzy regulátoru ve Fuzzy Toolboxu

Dalším krokem při návrhu bylo vytvoření samotného regulátoru pomocí Fuzzy Toolboxu v programu MATLAB.

Byl zvolen typ Mamdani se dvěma vstupy (odchylkou a diferencí odchylky) a jedním výstupem (akční zásah). Vstupní universum bylo nastaveno na rozsah 0 až 24. Zvolené typy implikace, defuzzifikace a agregace jsou uvedené na obrázku 6.4, který zobrazuje hlavní úvodní okno Fuzzy Toolboxu.

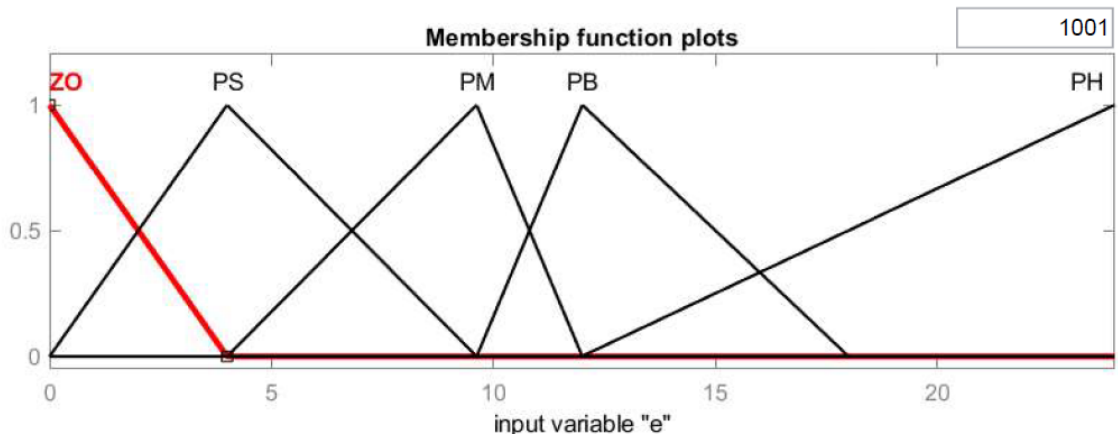


Obr. 6.4: Hlavní okno Fuzzy Toolboxu

Pro oba vstupy byly zvoleny funkce příslušnosti ve tvaru  $\Lambda$ -funkce. Pozice vrcholů těchto funkcí příslušnosti v universu byly zvoleny na základě průběhu identifikované nelinearity modelu větrného tunelu. Rozmístění funkcí příslušnosti je v obou vstupech shodné. Oba vstupy obsahují celkem pět funkcí příslušnosti nazvaných následovně:

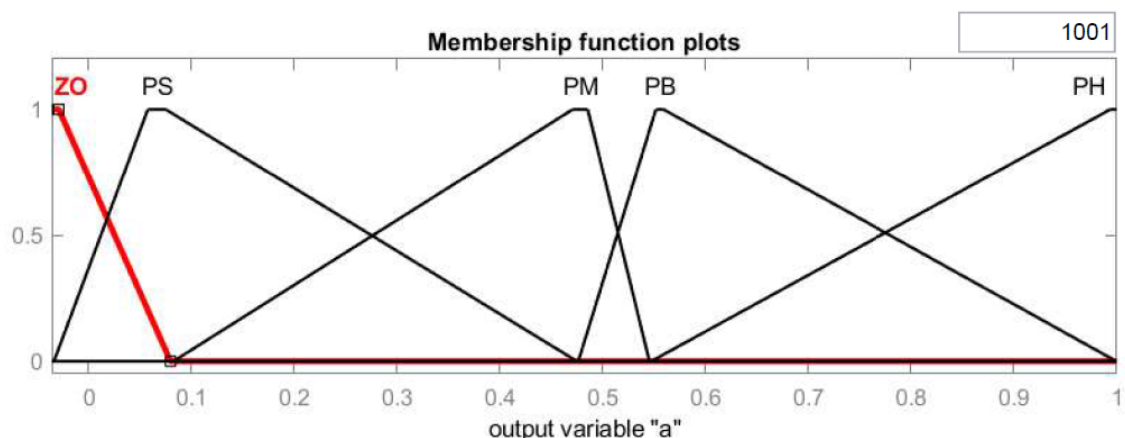
- ZO (zero)
- PS (positive small)
- PM (positive medium)
- PB (positive big)
- PH (positive high)

Zvolené rozmístění funkcí příslušností obou vstupů fuzzy regulátoru je znázorněno na obrázku 6.5.



Obr. 6.5: Definované funkce příslušnosti obou vstupů

Obdobným způsobem byly vytvořeny funkce příslušnosti pro výstup fuzzy regulátoru. Rozdíl spočíval ve volbě jiného typu funkce příslušnosti a rozsahu universa. Zde byl zvolen typ II-funkce (důvodem je skutečnost, že při prvním návrhu byl tento typ použit pro vytvoření funkce příslušnosti typu singleton, kde však byl akční zásah regulátoru příliš skokový, a proto došlo k roztažení částí jednotlivých funkcí do současné podoby). Prvotní rozmístění funkcí příslušnosti v universu odpovídalo hodnotě identifikované nelinearity modelu větrného tunelu v daném bodě. Vyskytl se však problém s velkým akčním zásahem, díky němuž nebylo možné adekvátně regulovat na požadovanou hodnotu. Z tohoto důvodu byl rozsah universa zvolen od  $-0,03545$  do 1. Dolní hranice intervalu byla posléze ve fázi testování vytvořeného regulátoru doladěna experimentálně na základě průběhu regulačního děje. Definované tvary funkcí příslušnosti a jejich rozmístění v universu je zobrazeno na obrázku 6.6.



Obr. 6.6: Definované funkce příslušnosti výstupu (akční zásah)



Poslední částí návrhu ve Fuzzy Toolboxu bylo nadefinování báze pravidel, podle nichž se bude výsledný fuzzy regulátor řídit. Jednotlivá pravidla byla sestavena na základě tabulky zachycené na obrázku 6.7.

		<b>e</b>				
		<b>ZO</b>	<b>PS</b>	<b>PM</b>	<b>PB</b>	<b>PH</b>
<b>de</b>	<b>ZO</b>	ZO	ZO	ZO	PS	PM
	<b>PS</b>	ZO	ZO	PS	PM	PB
	<b>PM</b>	ZO	PS	PM	PB	PH
	<b>PB</b>	PS	PM	PB	PH	PH
	<b>PH</b>	PM	PB	PH	PH	PH

Obr. 6.7: Pravidla pro fuzzy regulátor

Pravidla se pak sestavují následujícím způsobem:

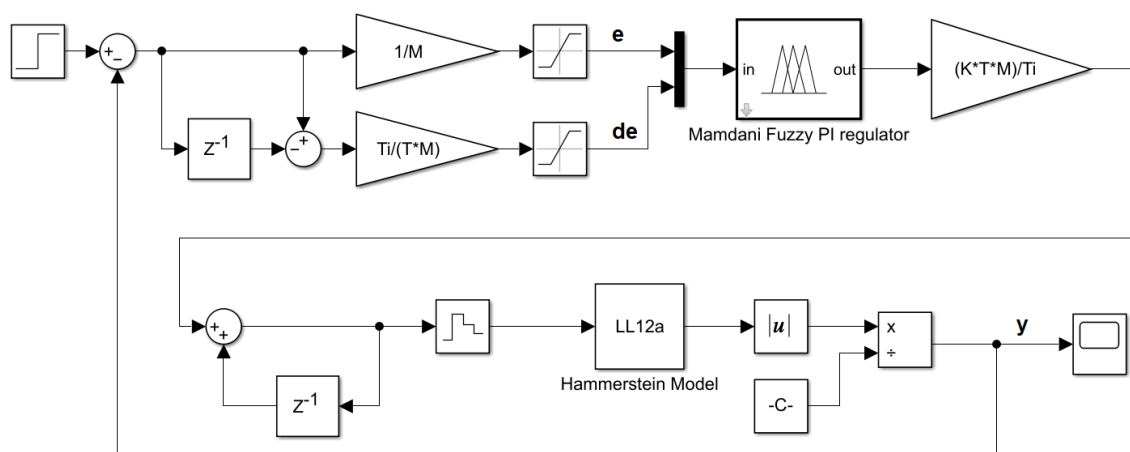
*IF (e je PS) AND (de je PB) THEN (a je PM)*

kde  $e$  je regulační odchylka,  $de$  je difference regulační odchylky a  $a$  je akční zásah.

### 6.3 Zapojení fuzzy regulátoru v Simulinku

Poslední částí návrhu regulátoru je jeho zapojení do regulačního obvodu. K tomuto účelu byl použit program Simulink.

Nakreslené schéma v prostředí Simulink je znázorněno na obrázku 6.8.



Obr. 6.8: Schéma zapojení v Simulinku

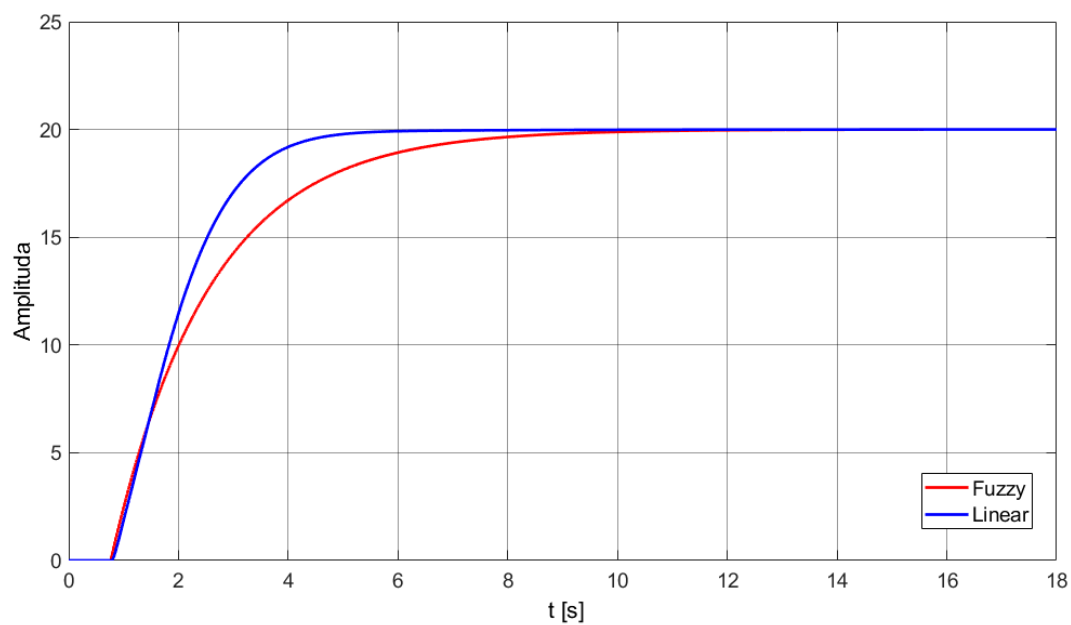
Vstupem do regulačního obvodu je skoková změna požadované hodnoty. Pomocí prvního zpožďovacího členu a sčítacího členu se vytváří diference regulační odchylky, která je následně násobena integrační časovou konstantou a podělena součinem vzorkovací periody a nastaveného měřítka. Regulační odchylka je podělena pouze nastaveným měřítkem. Dále následují saturační členy, jenž jsou nastaveny na hodnotu 24, která odpovídá horní hranici intervalu universa pro vstupy. Výstup fuzzy regulátoru je násoben součinem zesílení, vzorkovací periody a měřítka. Zároveň je podělen hodnotou integrační časové konstanty. Následně dochází k sumaci výstupu pomocí sumátoru tvořeného jedním zpožďovacím blokem. Výstup sumátoru prochází blokem Zero-Order Hold, který uchová hodnotu svého vstupu na svém výstupu po dobu jedné vzorkovací periody (během jedné vzorkovací periody se tedy hodnota nezmění). Pak už signál vstupuje do identifikovaného modelu větrného tunelu. Jelikož signál vystupující z modelu je v záporných hodnotách, prochází blokem, který na svém výstupu navrácí svůj vstup v absolutní hodnotě. Následně je ještě upraveno zesílení na základě požadované hodnoty.

## 6.4 Dosažené výsledky s manuálně navrženým fuzzy PI regulátorem

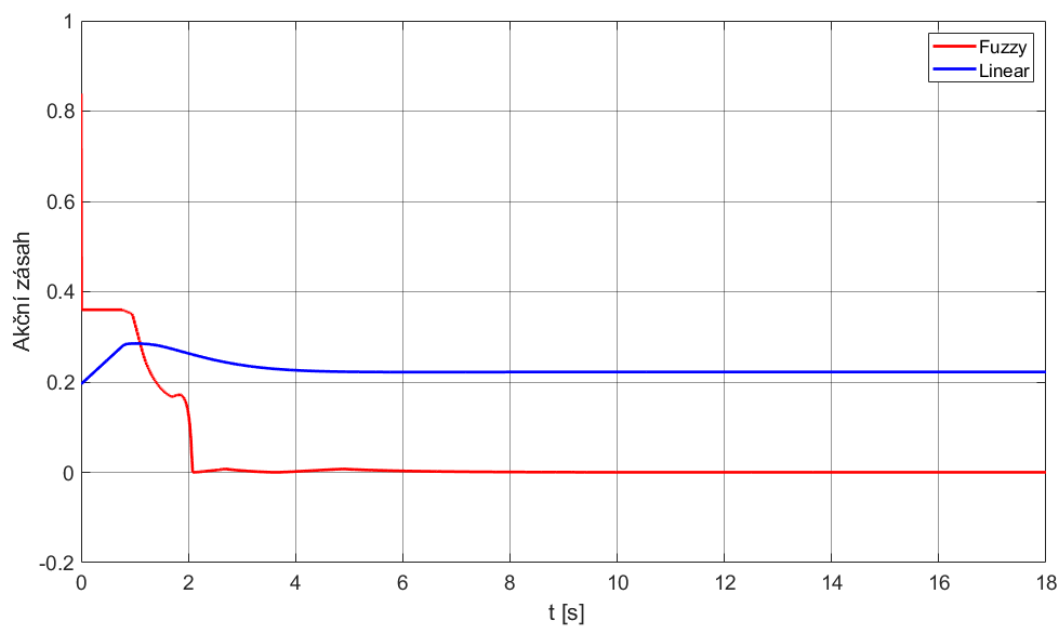
Tato kapitola představí dosažené výsledky manuálně navrženého fuzzy PI regulátoru řídicího identifikovanou reálnou soustavu. Tato soustava byla identifikována pomocí původní identifikace popsané v kapitolách 5.1 a 5.2.

Pro nastavenou požadovanou hodnotu například 20, dojde k jejímu ustálení na výstupu regulačního obvodu přibližně za 11 sekund. Během regulačního děje nedojde k žádnému překmitu ani podkmitu požadované hodnoty. Odezva regulačního obvodu na požadovanou hodnotu 20 je znázorněna na obrázku 6.9. Pro porovnání je zakreslena i odezva lineárního PI regulátoru. Konstanty  $K$  a  $T_I$  tohoto regulátoru jsou stejné jako u fuzzy regulátoru. Z obrázku je zřetelné, že lineární PI regulátor dosáhne žádané hodnoty dříve než navržený fuzzy PI regulátor. To mohlo být zapříčiněno manuálním nastavením fuzzy regulátoru, které nemusí být optimální, a proto je výsledný průběh v porovnání s průběhem lineárního regulátoru horší.

Průběh akčního zásahu fuzzy regulátoru je pak zachycen na obrázku 6.10. Opět je zde pro porovnání zakreslen průběh akčního zásahu lineárního PI regulátoru.



Obr. 6.9: Odezva regulačního obvodu na skokovou změnu požadované hodnoty



Obr. 6.10: Porovnání průběhů akčního zásahu fuzzy a lineárního regulátoru

## 7 Dodatečné úpravy regulátoru a zapojení

Tato kapitola popisuje provedené dodatečné úpravy v návrhu fuzzy regulátoru a zapojení regulačního obvodu. Potřeba těchto úprav vyvstala při implementaci optimalizační funkce a při následných simulacích v Simulinku. Pro větší přehlednost je tato kapitola zařazena před kapitolou pojednávající o optimalizaci fuzzy regulátoru.

### 7.1 Rozšíření navrženého fuzzy PI regulátoru

Došlo rovněž k úpravě samotného fuzzy PI regulátoru. Změnily se počty a tvary funkcí příslušnosti jak na vstupech, tak také na výstupu. S tímto krokem byla pozměněna i báze pravidel. Poslední úpravou v regulátoru bylo nahrazení proporcionální a integrační konstanty hodnotou 1.

Počet funkcí příslušnosti na vstupech regulátoru byl zredukován na čtyři funkce příslušnosti:

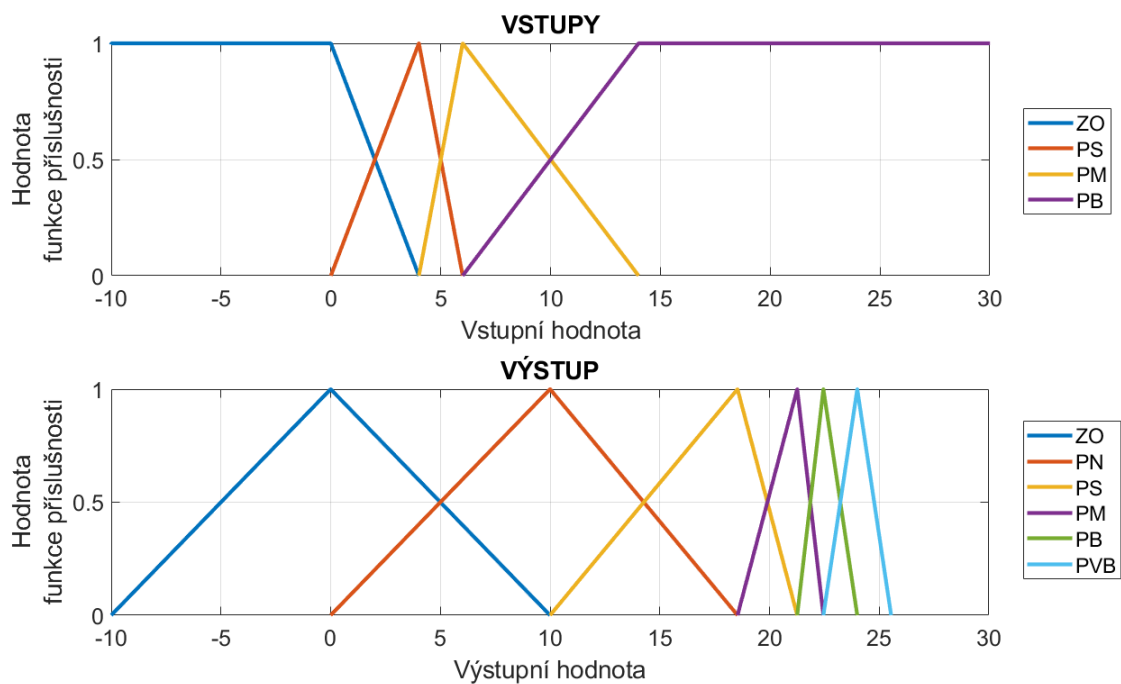
- ZO - zero
- PS - positive small
- PM - positive medium
- PB - positive big

Důvod snížení počtu funkcí bylo omezení počtu laditelných parametrů pro následnou optimalizaci regulátoru. Funkce příslušnosti nacházející se nejbližší krajním hodnotám universa byly změněny na typ  $\Pi$ -funkce. Ostatní zůstaly typu  $\Lambda$ -funkce. Dolní hranice universa byla pevně nastavena na -10. Horní hranice pak byla změněna na variabilní, aby si ji optimalizační funkce mohla upravovat podle své potřeby. Oba vstupy jsou opět totožné.

Naopak z důvodu jemnějšího akčního zásahu se počet funkcí příslušnosti na výstupu fuzzy regulátoru zvýšil na šest:

- ZO - zero
- PN - positive negligible
- PS - positive small
- PM - positive medium
- PB - positive big
- PVB - positive very big

Všechny funkce příslušnosti jsou typu  $\Lambda$ -funkce. Dolní hranice universa byla pevně nastavena na 0, horní hranice pak na 24. Na obrázku 7.1 lze vidět, že funkce příslušnosti svým průběhem zasahují také mimo nastavené universum. Důvodem je zvolený typ defuzzifikace jako těžiště, kdy je nutné, aby pro požadovanou výstupní hodnotu (např. 0) byl vrchol funkce v tomto bodě a tvar funkce byl v podobě rovnoramenného trojúhelníku.



Obr. 7.1: Funkce příslušnosti na vstupech a výstupu fuzzy regulátoru

Společně s úpravami vstupů a výstupu bylo přistoupeno rovněž ke změně báze pravidel regulátoru, která je zobrazena na obrázku 7.2. Bylo také opraveno značení druhého vstupu (suma odchylky).

		<b>e</b>			
		<b>ZO</b>	<b>PS</b>	<b>PM</b>	<b>PB</b>
$\Sigma e$	<b>ZO</b>	ZO	PM	PB	PVB
	<b>PS</b>	PN	PS	PM	PVB
	<b>PM</b>	PS	PM	PB	PVB
	<b>PB</b>	PM	PB	PVB	PVB

Obr. 7.2: Báze pravidel fuzzy regulátoru

Způsob sestavování pravidel zůstal stejný.

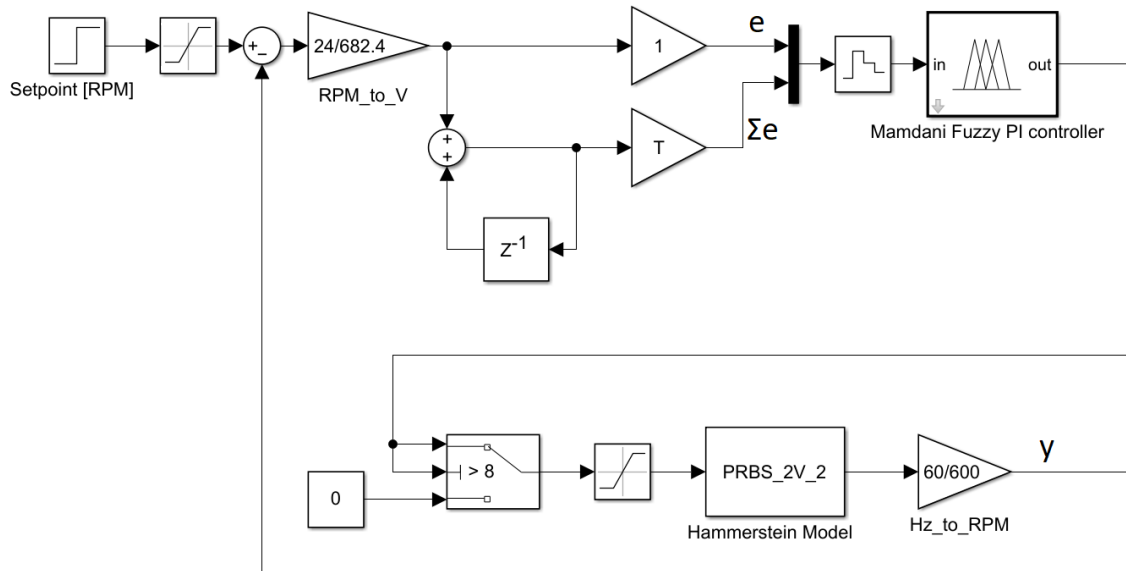
$$IF (e \text{ je } \mathbf{PS}) \text{ AND } (\Sigma e \text{ je } \mathbf{PB}) \text{ THEN } (a \text{ je } \mathbf{PM})$$

kde  $e$  je regulační odchylka,  $\Sigma e$  je suma regulační odchylky a  $a$  je akční zásah.

Poslední významnou úpravou související s fuzzy regulátorem bylo nahrazení konstanty proporcionálního zesílení a integrační konstanty hodnotou 1. Důvodem byla snaha snížit počet laditelných parametrů pro optimalizaci.

## 7.2 Změna v zapojení regulačního obvodu

Několik úprav bylo provedeno také v samotném zapojení regulačního obvodu. Byly implementovány převody mezi jednotlivými veličinami v obvodu, omezení a saturace akčního zásahu a druhý vstup fuzzy regulátoru byl změněn na sumu odchylky. Schéma zapojení obvodu po úpravách je zobrazeno na obrázku 7.3.



Obr. 7.3: Upravené schéma zapojení regulačního obvodu

Vstupem je skoková změna požadované hodnoty udávaná v otáčkách za minutu (RPM). Poté signál prochází saturačním blokem, který zajišťuje, aby nebyla překročena maximální možná hodnota. Následně dochází k přepočtu z otáček za minutu na volty podle následující rovnice:

$$e_V = e_{RPM} \cdot \frac{24}{682,4} \quad (7.1)$$

kde  $e_V$  je regulační odchylka ve voltech,  $e_{RPM}$  je regulační odchylka v RPM, hodnota 24 odpovídá maximálnímu napájecímu napětí (tj. také maximálnímu možnému akčnímu zásahu) a hodnota 682,4 odpovídá maximálním otáčkám na výstupu obvodu při napětí 24 V. Poté se signál rozdělí do dvou větví. První je samotná hodnota regulační odchylky, která je násobena jedničkou (před provedenými úpravami násobena konstantou proporcionálního zesílení). Druhou větví je sumace odchylky násobena vzorkovací periodou. Následně jsou obě větve multiplexovány do jedné. Poté signál prochází blokem Zero-Order Hold, který uchová hodnotu svého vstupu na svém výstupu po dobu jedné vzorkovací periody (během jedné vzorkovací periody se tedy hodnota nezmění) a vstupuje do samotného fuzzy regulátoru. Výstup regulátoru

(tj. akční zásah) je pomocí bloku switch omezen zdola na minimální hodnotu 8 V. Pokud je hodnota menší, bude na výstupu tohoto bloku hodnota 0 V. Toto omezení je dáno fyzickou realizací modelu větrného tunelu, kdy se enkodér při nižších hodnotách napětí na ventilátoru neroztočí. Dalším blokem je saturace na maximální možnou hodnotu 24 V odpovídající maximálnímu napájecímu napětí ventilátoru. Následně signál vstupuje do identifikovaného modelu větrného tunelu, jehož výstup, daný počtem pulsů enkodéru, je přepočten na otáčky za minutu. Tento přepočet se provádí pomocí rovnice:

$$y_{RPM} = y_{Hz} \cdot \frac{60}{600} \quad (7.2)$$

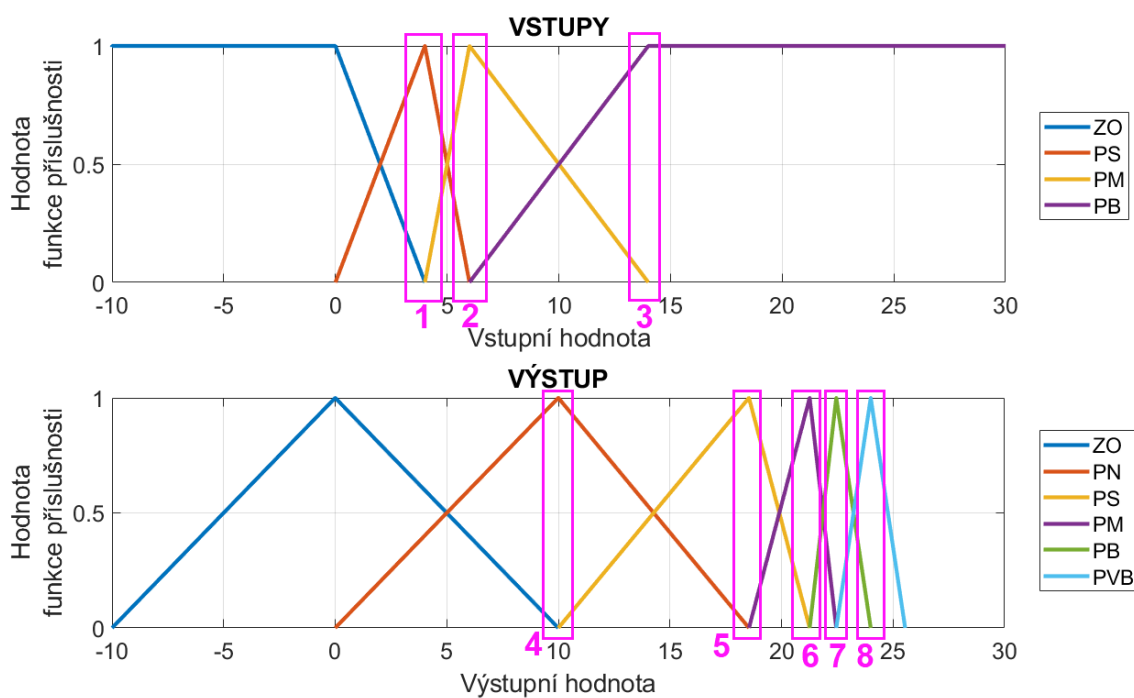
kde  $y_{RPM}$  je výstup v otáčkách za minutu,  $y_{Hz}$  je výstup v počtech pulsů za sekundu, hodnota 60 odpovídá 60 sekundám za 1 minutu a hodnota 600 udává počet pulsů na 1 otáčku. Následně výstup pokračuje do součtového členu, kde je hodnota výstupu odečtena od požadované hodnoty.

## 8 Optimalizace fuzzy PI regulátoru

Tato kapitola popisuje způsob implementace optimalizační funkce pro optimalizaci parametrů navrženého fuzzy PI regulátoru v prostředích MATLAB a Simulink. Součástí kapitoly je rovněž ukázka dosažených výsledků při optimalizaci regulátoru zapojeného v regulačním obvodu s identifikovaným modelem větrného tunelu.

### 8.1 Optimalizační parametry fuzzy regulátoru

Pro optimalizaci bylo zvoleno osm parametrů. Tyto parametry ovlivňují rozložení funkcí příslušnosti v universu. První tři parametry určují rozložení funkcí v obou vstupech regulátoru, zbylé ostatní pak na výstupu. Vybrané optimalizační parametry jsou zvýrazněny růžovými obdélníky na obrázku 8.1.



Obr. 8.1: Optimalizované parametry

Každý parametr ovlivňuje umístění vždy jedné souřadnice tří sousedících funkcí příslušnosti v daném universu (výjimkou jsou parametry 3 a 8, které se nachází na kraji universa a ovlivňují pouze dvě funkce příslušnosti). Pro lepší představu zvoleného ladění regulátoru je uveden následující příklad pro parametr 1. Třetí souřadnice funkce příslušnosti ZO, druhá souřadnice funkce příslušnosti PS a první souřadnice funkce příslušnosti PM budou mít stejnou hodnotu a tato hodnota bude



rovna hodnotě parametru 1. Důvodem tohoto seskupení, kdy jeden parametr ovlivňuje hned tři souřadnice, je menší počet možných nastavení regulátoru. Bylo by samozřejmě možné pomocí optimalizační funkce ladit každou souřadnici všech funkcí příslušnosti zvlášť. To by však znamenalo mnohem vyšší časovou náročnost samotné optimalizace, kdy by bylo nutné vykonat nesrovnatelně větší množství výpočtů a simulací. Seskupení více souřadnic do jednoho parametru se tedy jeví jako vhodnější řešení celého problému a to i za cenu, že nebude nalezeno nejoptimálnější řešení, k němuž kvůli zmíněnému seskupení nemůže dojít.

Ostatní souřadnice, které nepodléhají žádnému z parametrů (např. první a druhá souřadnice funkce příslušnost ZO na výstupu regulátoru) jsou nastaveny na předem dané hodnoty a během procesu optimalizace jsou invariantní.

## 8.2 Optimalizační funkce

Pro optimalizaci fuzzy regulátoru byla vytvořena optimalizační funkce skládající se ze tří hlavních částí - optimalizační metody, zápisu parametrů a vyhodnocení dané kritériální funkce.

Jako optimalizační metoda byla zvolena metoda Nelder-Mead (byla popsána v kapitole 4.1). Důvodem volby této metody byla její efektivita, snadná implementace a také výpočty bez derivací. Jedinou vyzpozorovanou nevýhodou byla nutnost nového spuštění výpočtu poté, co metoda dospěla k lokálnímu minimu funkce namísto hledaného globálního minima. Při následné implementaci v MATLABu byla využita funkce *fminsearch*, která právě metodu Nelder-Mead využívá.

Další dvě metody popsané v kapitole 4 nebyly implementovány. Newtonova metoda ve spojení s metodou linesearch nebyla realizována z důvodu vyšší implementační náročnosti (v kontextu porovnání obdobného výsledku jako u metody Nelder-Mead a tedy nižší přidané hodnoty oproti její vyšší náročnosti) a metoda zlatého řezu, která neumožňuje optimalizaci s více parametry.

Druhou částí optimalizační funkce je zápis nových parametrů do daných jednotlivých souřadnic funkcí příslušnosti fuzzy regulátoru. Tento zápis byl proveden pomocí přímého přístupu k datům fuzzy regulátoru prostřednictvím strukturálního členění dat. Samotný zápis byl implementován pomocí skriptu v programu MATLAB. V rámci tohoto skriptu byly ošetřeny i další náležitosti související se samotnou podobou jednotlivých funkcí příslušnosti (např. parametr 2 nemůže být větší než parametr 1 nebo funkce příslušnosti PS se musí nacházet na nižších hodnotách universa než funkce příslušnosti PB, apod.).

Poslední částí optimalizační funkce je vyhodnocení fuzzy regulátoru s nově nastavenými parametry pomocí kritériální funkce. Bylo zvoleno ITAE kritérium a to z důvodu vyšší, s časem narůstající, penalizace odchylky od požadované hodnoty.

Kritérium bylo vypočteno podle následující rovnice:

$$I = \int_0^{\infty} |e(t)| \cdot t dt \quad (8.1)$$

V programu MATLAB byla k výpočtu ITAE kritéria použita funkce *trapz*, která vy počítává aproximaci integrálu pomocí lichoběžníkové metody [16].

### 8.3 Dosažené výsledky optimalizace v Simulinku

Po implementaci optimalizační funkce bylo přistoupeno k samotné optimalizaci navrženého fuzzy PI regulátoru. Optimalizace probíhala v prostředích MATLAB a Simulink. Regulátor byl zapojen v regulačním obvodu s identifikovaným modelem větrného tunelu podle obrázku 7.3.

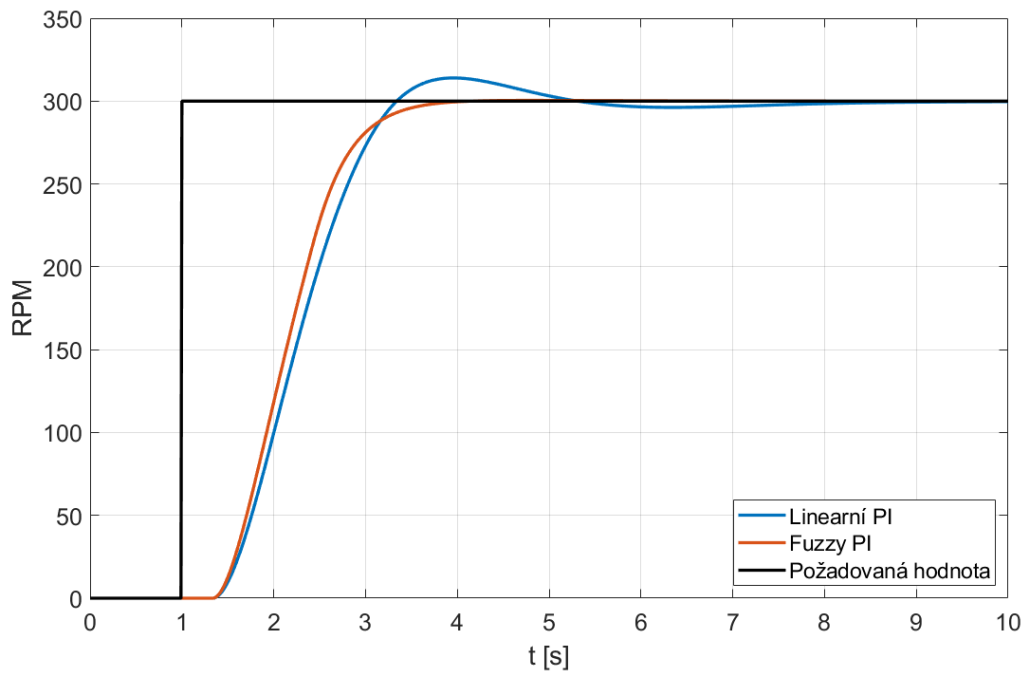
Nejnižší možná požadovaná hodnota, pro níž lze provést optimalizaci je 300 RPM. Tato hodnota je dána opět fyzikálními vlastnostmi reálného modelu větrného tunelu, kdy akční zásah regulátoru nemůže mít menší hodnotu než 8 V (mimo nulový akční zásah, který je povolen). Hodnoty nižší než je 300 RPM tuto podmínku nesplňují. Naopak maximální požadovaná hodnota byla stanovena na 600 RPM. Tato hodnota vychází z otáček enkodéru při maximálním napětí 24 V na ventilátoru. Ve skutečnosti se maximální hodnota otáček pohybuje v intervalu 620 až 660 RPM. Z důvodu bezpečnosti a také s přihlédnutím na jistou nepřesnost identifikovaného modelu byla zvolena jako maximální přípustná požadovaná hodnota 600 RPM.

Obrázek 8.2 zachycuje průběh odezvy regulačního obvodu na skokovou změnu požadované hodnoty z 0 na 300 RPM. Pro porovnání je také vykreslena odezva regulačního obvodu s lineárním PI regulátorem. Z obrázku je patrné, že s fuzzy regulátorem bylo dosaženo lepšího regulačního děje než s lineárním regulátorem. Fuzzy regulátor reguloval bez překmitu a dosáhl požadované hodnoty přibližně po 3 sekundách. Lineární regulátor zaznamenal 7% překmit požadované hodnoty a následně malý podkmit. K ustálení na požadované hodnotě došlo za 8 sekund. Hodnota ITAE kritéria byla pro fuzzy regulátor vyčíslena na 245,86 a pro lineární regulátor na 385,10. Lineární regulátor byl rovněž optimalizován optimalizační funkcí, která byla popsána v kapitole 8.2. Na obrázku 8.3 je zachycen průběh akčních zásahů obou regulátorů. Ukončovacími podmínkami procesu optimalizace byly:

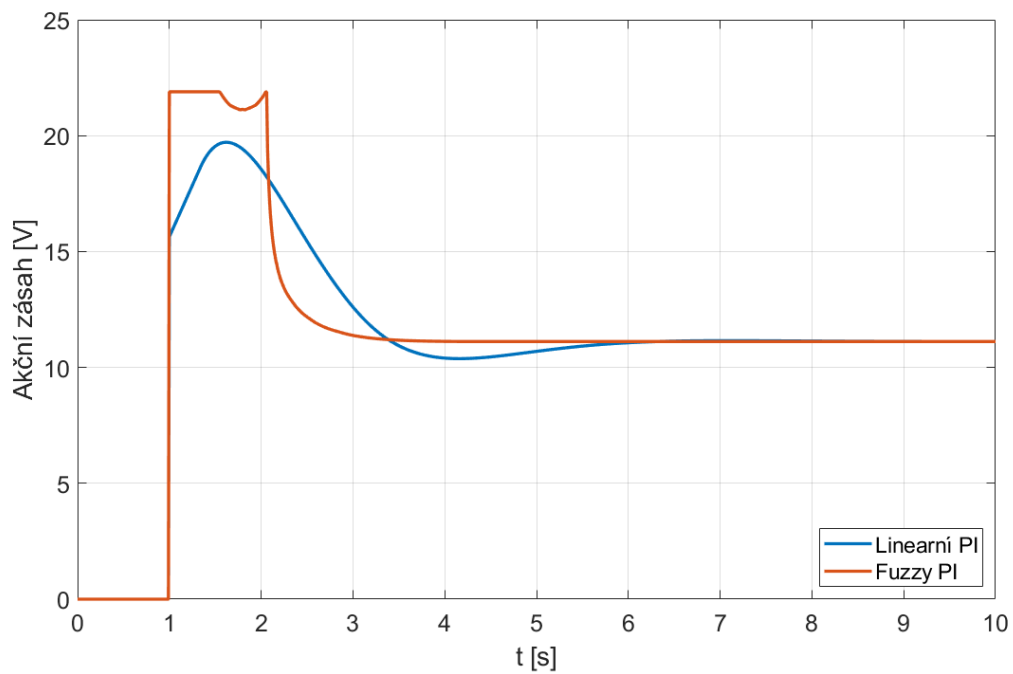
- velice malá změna hodnoty ITAE kritéria za posledních 100 iterací (v řádech tisícín)
- situace, kdy změna žádného z parametrů nevedla k minimalizaci hodnoty ITAE kritéria
- vyčerpání všech možných řešení

Občas bylo vhodné spustit proces optimalizace znovu, a to v případech, kdy se hodnota ITAE kritéria za posledních 50 iterací nezměnila. Tato situace mohla nastat,

když metoda Nelder-Mead konvergovala k lokálnímu minimumu namísto globálního minima.

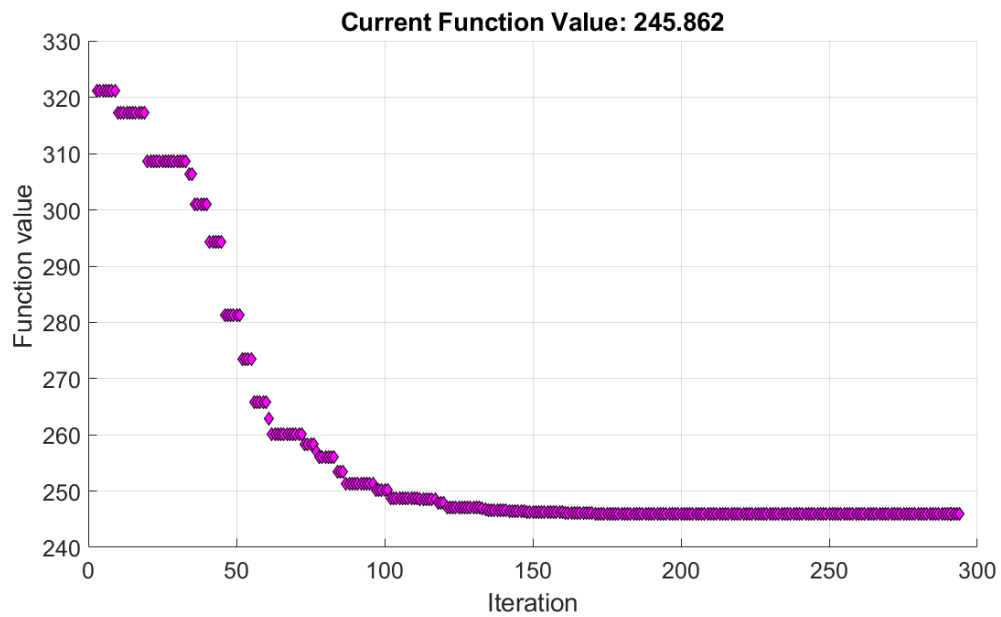


Obr. 8.2: Odezva regulačního obvodu na požadovanou hodnotu po optimalizaci

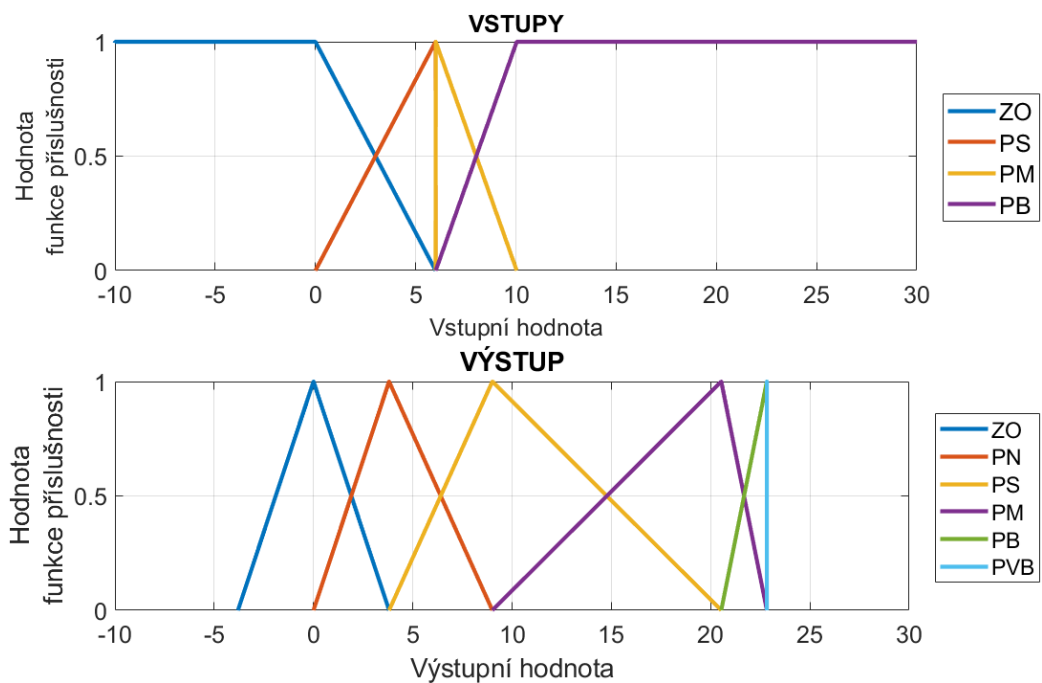


Obr. 8.3: Porovnání akčních zásahů obou regulátorů

Obrázek 8.4 ilustruje vývoj hodnoty ITAE kritéria v průběhu optimalizace fuzzy regulátoru pro požadovanou hodnotu 300 RPM. Na obrázku 8.5 jsou pak vykresleny optimalizované funkce příslušnosti pro oba vstupy i výstup regulátoru.



Obr. 8.4: Vývoj hodnoty ITAE kritéria při optimalizaci fuzzy regulátoru



Obr. 8.5: Optimalizované funkce příslušnosti pro požadovanou hodnotu 300 RPM

Na obrázku 8.5 lze spatřit, že funkce příslušnosti PVB na výstupu regulátoru má zdánlivě jiný tvar než zbylé ostatní. Ve skutečnosti se jedná rovněž o typ  $\Lambda$ -funkce, akorát šířka základny je tak velmi malá, že se tvarem tato funkce příslušnosti podobá spíše funkci příslušnosti typu singleton.

Dále byly oba regulátory optimalizovány pro požadované hodnoty 350, 400, 450, 500, 550 a 600 RPM. Jednotlivé vypočtené hodnoty ITAE kritéria jsou uvedeny v tabulce 8.1. Obrázky dalších odezev regulačních obvodů s optimalizovanými regulátory pro jednotlivé požadované hodnoty jsou uvedeny v příloze A.

Tab. 8.1: Hodnoty ITAE kritéria pro optimalizované regulátory

<b>RPM</b>	<b>ITAE fuzzy</b>	<b>ITAE lineární</b>
300	245,86	385,10
350	303,41	458,02
400	442,37	495,70
450	565,53	608,11
500	743,49	778,90
550	1015,47	1023,21
600	1367,46	1400,66

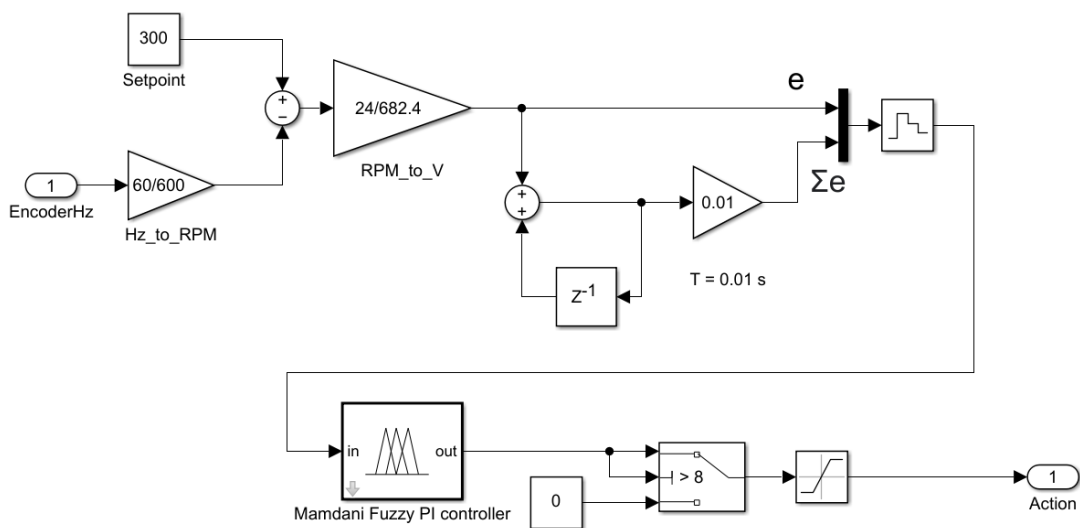
Z hodnot ITAE kritéria v tabulce 8.1 je patrné, že u všech optimalizovaných fuzzy PI regulátorů bylo dosaženo lepšího přechodového děje (tj. menšího překmitu, rychlejšího ustálení na požadované hodnotě, apod.) než u klasického lineárního PI regulátoru.

## 9 Řízení reálného systému fuzzy regulátorem v PLC

Tato kapitola přiblíží postup při přenesení vytvořeného fuzzy PI regulátoru z prostředí Simulink do programu TIA Portal V15.1 pomocí toolboxu MATLAB Simulink PLC Coder. Dále pak zde budou představeny dosažené výsledky optimalizovaného regulátoru při řízení reálného modelu větrného tunelu.

### 9.1 Přenos kódu ze Simulinku do PLC

Prvním krokem byla modifikace vytvořeného regulačního obvodu v Simulinku tak, aby bylo možné jej přetransformovat do jazyka SCL. Na obrázku 9.1 je zakreslen regulační obvod po provedených úpravách.



Obr. 9.1: Upravený obvod pro přenesení do PLC

Oproti regulačnímu obvodu zakreslenému na obrázku 7.3 bylo přistoupeno k náhradě vstupu pro požadovanou hodnotu. Nyní je tato hodnota reprezentována nastavenou konstantou. Bylo by samozřejmě možné tento vstup ponechat a nastavovat požadovanou hodnotu prostřednictvím programu TIA Portal přímo v PLC. Tato možnost se však jeví jako zbytečná, jelikož při změně požadované hodnoty musí být změněny i optimalizované parametry fuzzy regulátoru, které jsou pro každou požadovanou hodnotu rozdílné. Dále byl také odstraněn identifikovaný model nelineární soustavy.

Obvod má nyní jeden vstup a jeden výstup. Vstupem je výstupní frekvence enkodéru (tj., zpětná vazba), která je následně přepočtena na otáčky za minutu. Tyto otáčky jsou posléze v součtovém členu odečteny od nastavené požadované hodnoty. Výstupem je pak akční zásah, jehož hodnota odpovídá napětí přivedenému na ventilátor na vstupu modelu větrného tunelu.

Takto upravený obvod byl následně zahrnut do jednoho subsystému, aby bylo možné z něj vygenerovat SCL kód. Před samotným vygenerováním kódu bylo ještě zapotřebí označit celý subsystém jako atomickou jednotku a nastavit parametry pro generování. Zde bylo vybráno cílové IDE jako TIA Portal: Double Precision. Volitelným parametrem pak bylo vygenerování dokumentace ke kódu.

## 9.2 Program v prostředí TIA Portal

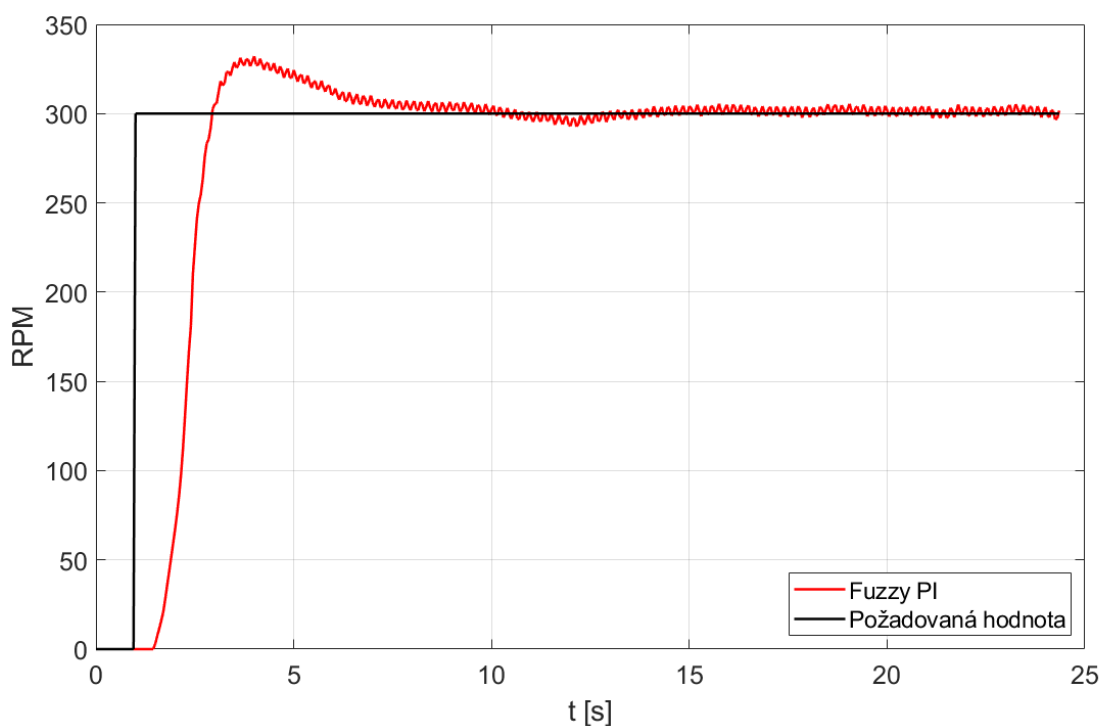
V programu TIA Portal verze 15.1 byl vytvořen nový projekt a proběhla základní konfigurace PLC (vybrání typu PLC, verze firmware, nastavení síťového připojení, apod.). Kód vygenerovaný z programu Simulink byl rozdělen do jednotlivých funkčních bloků (FB) a funkcí (FC). K funkčním blokům byly vytvořeny i jejich datové bloky (DB) pro uchovávání dat. Pro fuzzy regulátor byly vytvořeny dva funkční bloky (jeden pomocný pro funkci příslušnosti typu  $\Lambda$ -funkce a jeden pro samotné zapojení regulátoru) a dvě funkce (pro výpočty funkcí příslušnosti typu  $\Pi$ -funkce a  $\Lambda$ -funkce). Dále byl funkční blok s regulátorem doplněn o ochrannou povolovací podmínku, která umožňovala uživateli povolit spuštění regulace. Samotná regulace se spouštěla nastavením příslušné proměnné svázané se strukturou case, v níž byla funkce regulátoru implementována. Vstupem funkčního bloku s regulátorem byl výstup rotačního enkodéru, výstupem pak akční zásah. Hodnota akčního zásahu byla následně přivedena na PWM výstup PLC. Vytvořené bloky funkcí tramf a trimf jsou určeny k výpočtu výstupní hodnoty obou použitých typů funkcí příslušnosti.

V hlavním organizačním bloku Main (OB1) bylo implementováno nastavení čítání rychlých vstupů PLC (High Speed Counter) a rovněž zde byla nastavena PWM na výstupu PLC.

Volání funkčního bloku s regulátorem by implementováno v organizačním bloku Cyclic interrupt (OB30), jehož perioda přerušení byla nastavena na 50 ms. Samotné měření odezvy reálného systému bylo provedeno pomocí integrované funkce programu TIA Portal s názvem Trace. Naměřené hodnoty byly následně uloženy do souboru typu CSV pro pozdější zpracování v programu MATLAB a vykreslení grafických závislostí.

### 9.3 Dosažené výsledky při řízení reálného modelu

Na obrázku 9.2 je zachycena změřená odezva reálného nelineárního systému řízeného optimalizovaným fuzzy PI regulátorem na skokovou změnu požadované hodnoty z 0 na 300 RPM. Oproti odezvě vykreslené na obrázku 8.2 obsahuje tato odezva menší překmit požadované hodnoty a k dosažení ustálené požadované hodnoty dojde později. Hodnota překmitu v jeho nejvyšším bodě dosahuje přibližně 10 % požadované hodnoty. K dosažení ustálené požadované hodnoty došlo přibližně po 9 sekundách (tj. přibližně o 5 s později, než v případě odezvy v Simulinku). Následně došlo ještě i k malému podkmitu požadované hodnoty. Tento podkmit však pravděpodobně nesouvisí s přechodovým dějem, nýbrž se samotnou stavbou reálného systému a jeho fyzikálními vlastnostmi. Naměřená hodnota dopravního zpoždění dosahuje přibližně 0,5 s. Viditelné zvlnění průběhu odezvy bylo způsobeno nevyváženou turbínou umístěnou na hřídeli rotačního enkodéru, kdy při rotaci opisovaly jednotlivé konce listů turbíny trajektorii ve tvaru elipsy.

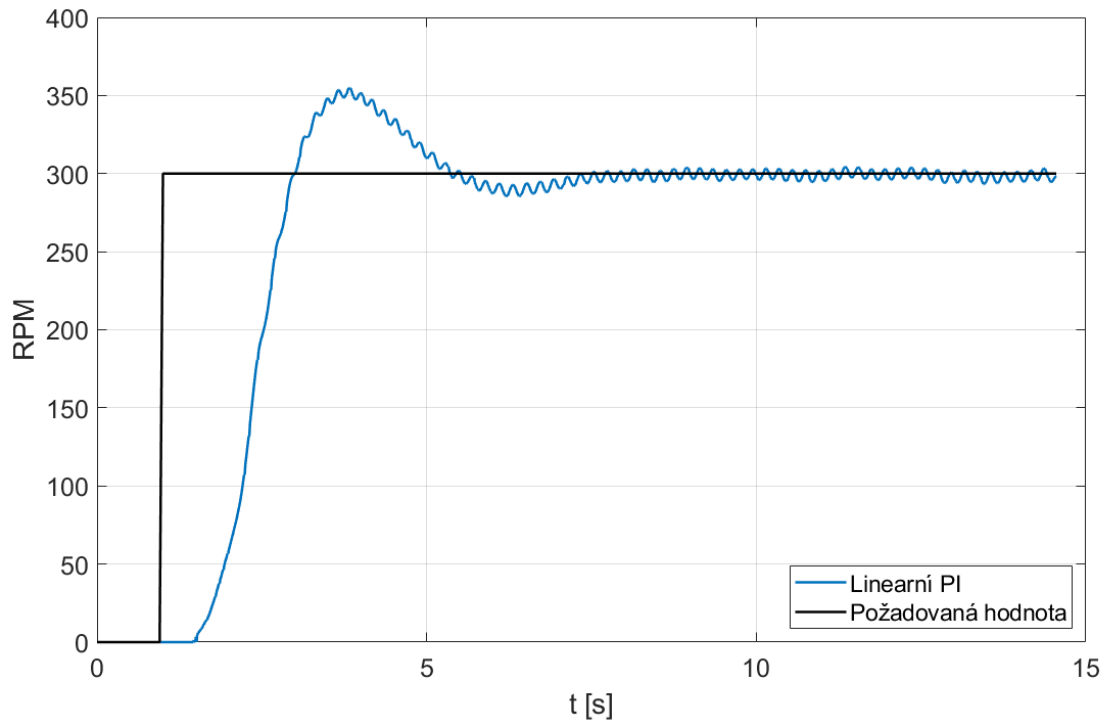


Obr. 9.2: Odezva reálného reg. obvodu s fuzzy regulátorem pro 300 RPM

Obrázek 9.3 znázorňuje průběh odezvy reálného nelineárního systému řízeného optimalizovaným lineárním PI regulátorem na skokovou změnu požadované hodnoty z 0 na 300 RPM. Zde bylo dosaženo většího překmitu i podkmitu než při odezvě na obrázku 8.2. Hodnota překmitu v jeho nejvyšším bodě činí přibližně 17 % požado-



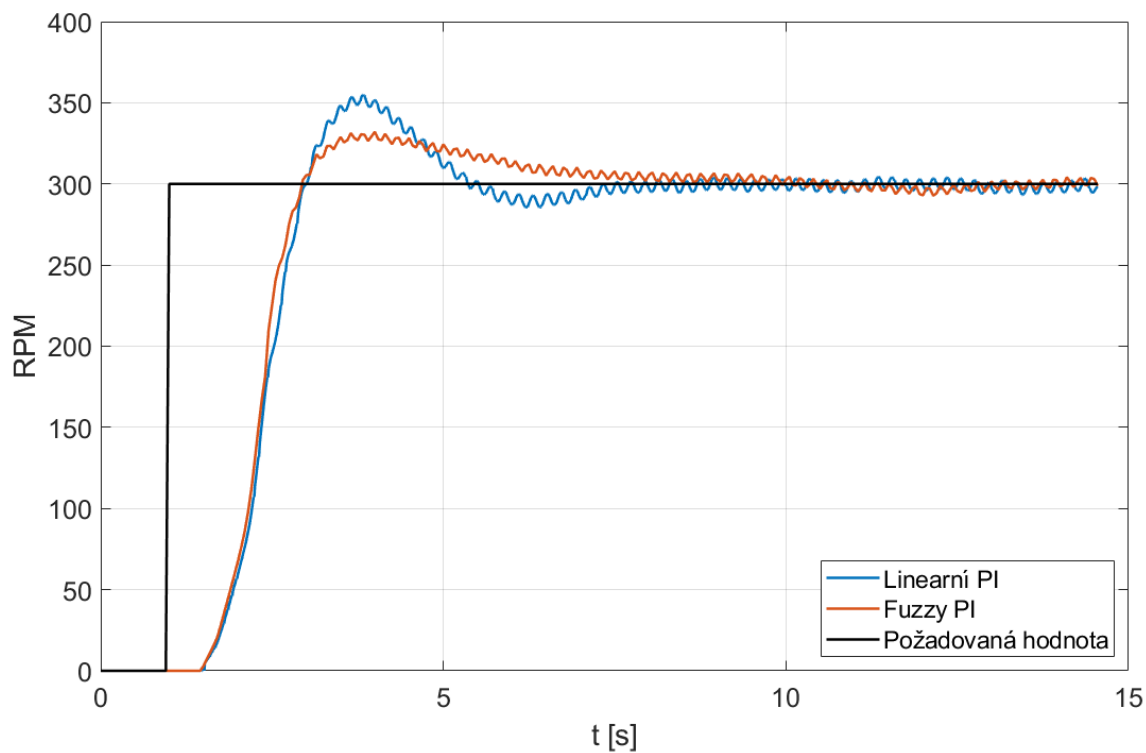
vané hodnoty. Ustálené požadované hodnoty bylo dosaženo po přibližně 8 sekundách, což zhruba odpovídá době dosažené při simulaci. Viditelné zvlnění signálu bylo opět způsobeno nevyváženou turbínou na enkodéru.



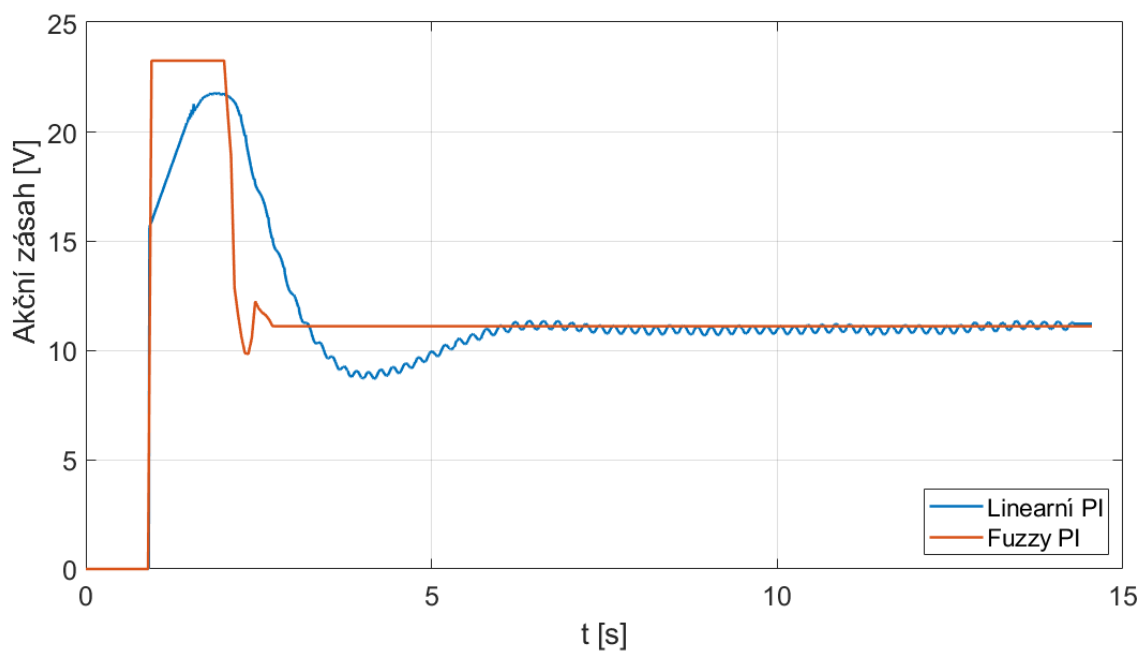
Obr. 9.3: Odezva reálného reg. obvodu s lineárním regulátorem pro 300 RPM

Na obrázku 9.4 je znázorněno porovnání odezev regulačních obvodů s fuzzy PI a lineárním PI regulátorem. V případě fuzzy regulátoru bylo dosaženo menšího překmitu požadované hodnoty. Oba regulátory dosáhly požadované hodnoty přibližně ve stejný okamžik (případný rozdíl není příliš významný). Hodnota ITAE kritéria pro obvod s fuzzy regulátorem je 11 338, pro obvod s lineárním regulátorem pak 11 735.

Obrázek 9.5 znázorňuje průběh akčních zásahů obou regulátorů při regulaci na požadovanou hodnotu 300 RPM.

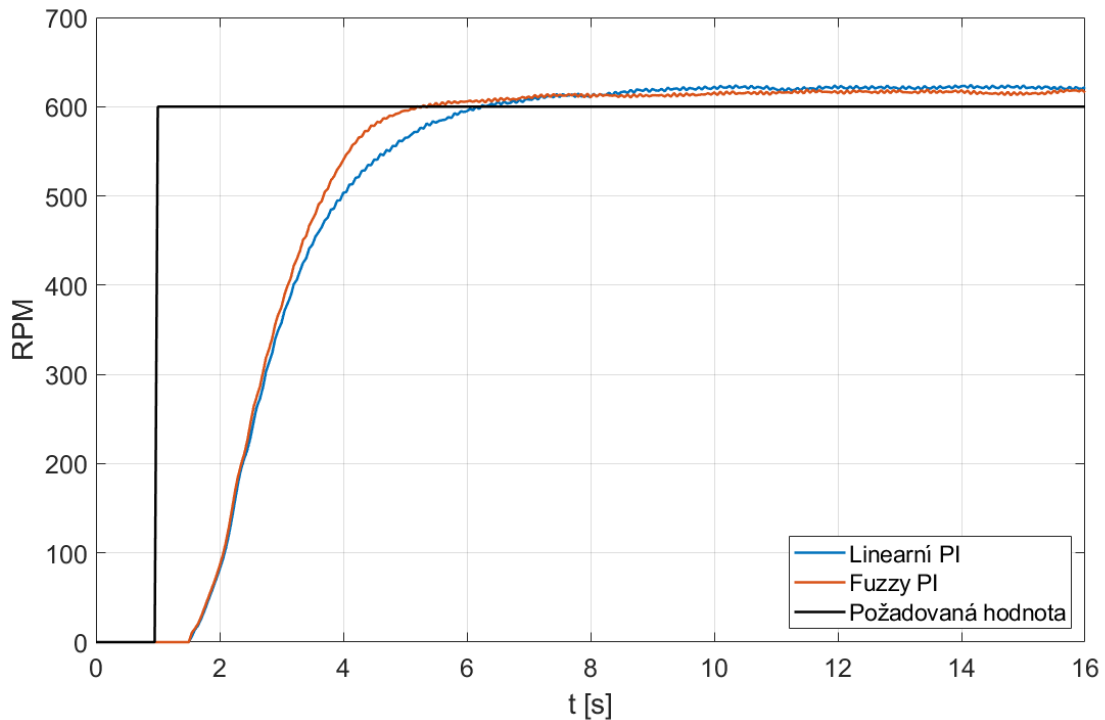


Obr. 9.4: Porovnání odezev reálného regulačního obvodu pro 300 RPM



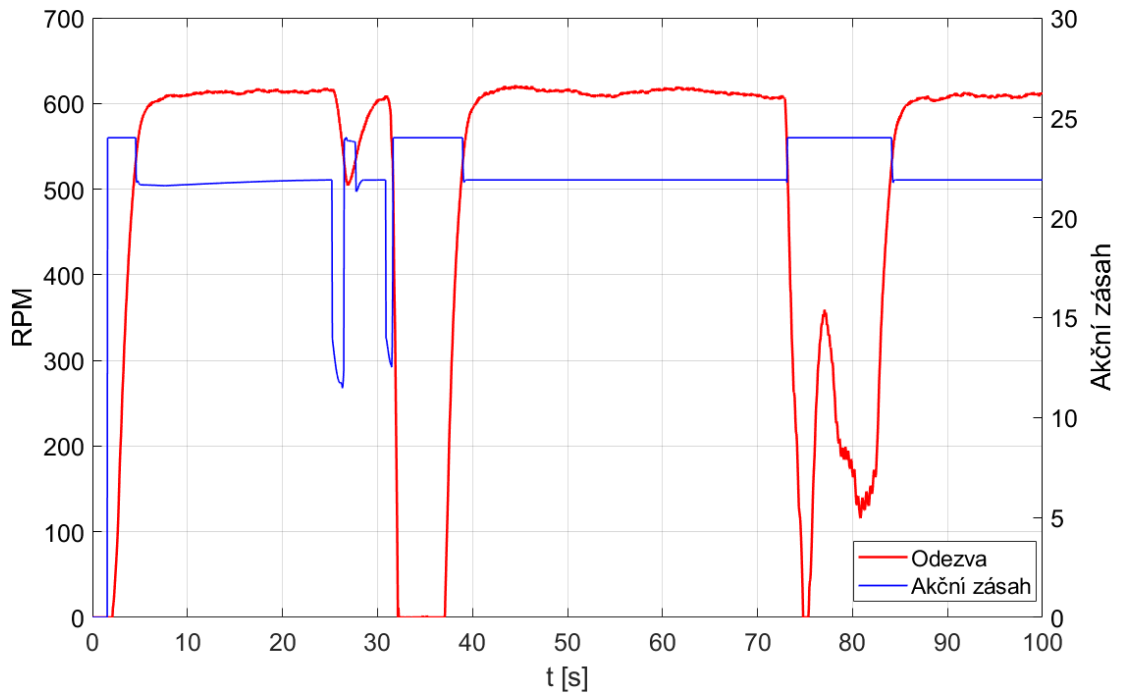
Obr. 9.5: Porovnání akčních zásahů obou regulátorů pro 300 RPM

Dále byly oba regulátory vyzkoušeny pro požadovanou hodnotu 600 RPM. Porovnání získaných odezev je zakresleno na obrázku 9.6. Zde se již projevila nepřesnost identifikované soustavy modelu větrného tunelu. Oba regulátory se ustálily nad požadovanou hodnotou 600 RPM. Vypočtená hodnota ITAE kritéria pro obvod s fuzzy regulátorem činí 20 120 a pro obvod s lineárním regulátorem 22 203. Z vykreslených odezev je rovněž patrné, že oba regulátory začaly s velkým akčním zásahem, přičemž lineární regulátor začal tento akční zásah snižovat dříve než fuzzy regulátor.



Obr. 9.6: Porovnání odezev reálného regulačního obvodu pro 600 RPM

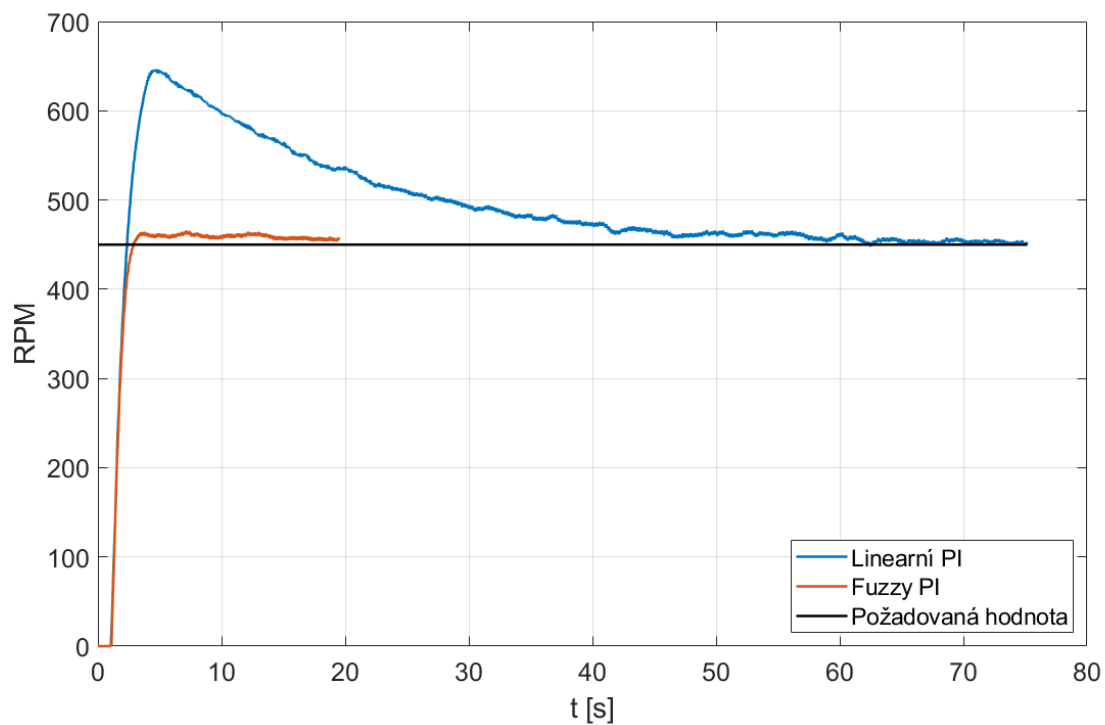
V rámci testování regulátorů na reálném systému bylo také vyzkoušeno, jak se oba regulátory dokáží vypořádat s neočekávanou chybou. Tuto chybu představovalo ruční zpomalení nebo úplné zastavení rotace turbíny na enkodéru. Na obrázku 9.7 je zachycena odezva reálného obvodu s fuzzy regulátorem pro požadovanou hodnotu 600 RPM. Součástí grafu je i průběh akčního zásahu regulátoru. V čase 25 sekund došlo ke krátkému přibrzdění rotace enkodéru, přičemž po opětovném uvolnění se regulátor vrátil zpět na požadovanou hodnotu. V čase 31 sekund byl enkodér na 6 sekund zcela ručně zastaven. Po opětovném uvolnění došlo k vyregulování na požadovanou hodnotu. V čase 73 sekund byl enkodér na sekundu úplně zastaven a následně zcela uvolněn. Vzápětí došlo k jeho přibrzdění po dobu přibližně 8 sekund. V této fázi byl akční zásah regulátoru maximální. Po následném odbrzdění se regulovaná veličina opět dostala na požadovanou hodnotu.



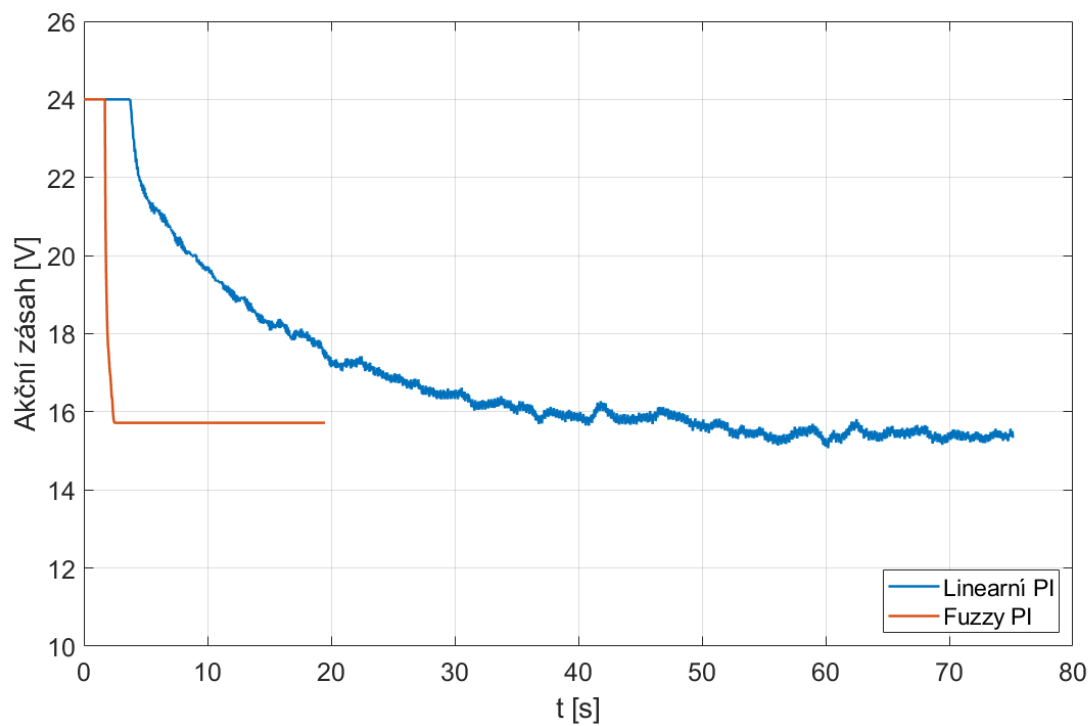
Obr. 9.7: Testování fuzzy regulátoru na neočekávanou chybu

V průběhu tohoto testování bylo zjištěno, že se fuzzy regulátor dokázal vždy po přibrzdění nebo úplném zastavení rotace enkodéru vrátit na požadovanou hodnotu bez jakýchkoliv problémů, zatímco u lineárního regulátoru byla tato situace jiná. Tuto skutečnost zachycuje obrázek 9.8. Zde je požadovaná hodnota nastavena na 450 RPM. Obrázek znázorňuje situaci po odbrzdění zcela zastaveného enkodéru, kdy je akční zásah obou regulátorů maximální. Lineární regulátor dosáhne velkého překmitu (v tomto případě 42 %) a pozvolna začne klesat k požadované hodnotě. Naopak fuzzy regulátor se vrátí na požadovanou hodnotu bez jakýchkoliv problémů. Obrázek 9.9 znázorňuje průběh akčního zásahu obou regulátorů. Zde je patrné, že fuzzy regulátor sníží hodnotu akčního zásahu mnohem dříve než lineární regulátor. V důsledku toho se fuzzy regulátor nedostane do velkého překmitu požadované hodnoty. Rovněž je také z obrázku zjevná vyšší odolnost fuzzy regulátoru proti šumu. Malé výchylinky od požadované hodnoty nezpůsobí výraznou změnu hodnoty akčního zásahu, jako tomu je u lineárního regulátoru.

Veškeré dosažené výsledky prezentované v této kapitole byly získány v univerzitní laboratoři při teplotě okolního vzduchu přibližně 24 °C. Velký vliv na dosažené výsledky mají fyzikální parametry vzduchu protékajícího větrným tunelem a to zejména jeho teplota a procento nasycení vodními parami. Bylo zjištěno, že při měření stejné závislosti ve dvou různých dnech lze získat významně odlišné výsledky. Tyto vlivy by bylo pravděpodobně možné potlačit zvětšením průměru větrného tunelu.



Obr. 9.8: Reakce regulátorů po skončení působení chyby



Obr. 9.9: Akční zásah regulátorů po skončení působení chyby

# Závěr

Cílem této diplomové práce byl návrh fuzzy regulátoru a jeho následná optimalizace pro reálnou nelineární soustavu. Dále se práce zaměřila na rešerši optimalizačních metod a porovnání výsledků vytvořeného fuzzy regulátoru s klasickým lineárním regulátorem při řízení reálného modelu větrného tunelu.

V kapitole 4 byly popsány tři optimalizační metody. Jedná se o metodu Nelder-Mead, Newtonovu metodu ve spojení s metodou line search a metodu zlatého řezu. Pro optimalizaci parametrů fuzzy regulátoru byla použita metoda Nelder-Mead. Důvodem volby této metody byla její efektivita, snadná implementace a také výpočty bez derivací. Jedinou nevýhodou byla nutnost opětovného spuštění výpočtu poté, co metoda dospěla k lokálnímu minimu funkce namísto hledaného globálního minima. Newtonova metoda ve spojení s metodou linesearch nebyla použita z důvodu vyšší implementační náročnosti (v kontextu porovnání obdobného výsledku jako u metody Nelder-Mead a tedy nižší přidané hodnoty oproti její vyšší náročnosti). Metoda zlatého řezu zase neumožňuje optimalizaci s více parametry.

Kapitola 5 se zabírala identifikací modelu větrného tunelu pomocí PLC a programu TIA Portal. Poprvé byl model identifikován pomocí signálu s rostoucí lineární částí (obrázek 5.2). Přesnost identifikace dosáhla 91,99 %. V průběhu implementace optimalizační funkce však bylo přistoupeno k nové identifikaci a to pomocí PRBS signálu ve čtyřech různých napěťových úrovních (obrázek 5.11). Při této identifikaci bylo dosaženo přesnosti 93,2 %.

Kapitola 6 přiblížila postup při manuálním návrhu fuzzy regulátoru. Byl zvolen fuzzy Mamdani PI regulátor, jelikož je zcela dostačující a zároveň je i jednodušší na implementaci než fuzzy PID regulátor. Tento regulátor má pět funkcí příslušnosti v obou vstupech i výstupu. Při jeho návrhu byl kladen důraz na regulaci bez překmitu. Při prvotním návrhu byly konstanty získány pomocí ITAE kritéria, které však zapříčinilo tlumené kmity v regulačním ději. Proto byly konstanty získány pomocí simulace v Simulinku za pomoci bloku PID Controller. Navržený fuzzy regulátor reguluje na požadovanou hodnotu. V porovnání odezev na obrázku 6.9 je zřetelné, že lineární PI regulátor dosáhne žádané hodnoty dříve než navržený fuzzy PI regulátor. To mohlo být způsobeno manuálním nastavením parametrů fuzzy regulátoru. Toto nastavení nemusí být optimální, a proto je výsledný průběh v porovnání s průběhem lineárního regulátoru horší.

Při implementaci optimalizační funkce došlo k úpravám navrženého fuzzy PI regulátoru. Počet funkcí příslušnosti na obou vstupech byl snížen z pěti na čtyři. Naopak počet funkcí na výstupu byl zvýšen na šest. Dále byly hodnoty proporcionálního zesílení a integrační časové konstanty nahrazeny hodnotou 1. Hlavním důvodem těchto změn bylo omezení počtu optimalizovaných parametrů fuzzy regu-

látoru, jelikož se zvyšujícím se počtem těchto parametrů se zvyšoval i potřebný čas na celý proces optimalizace.

V kapitole 8 byla popsána vytvořená optimalizační funkce pro navržený fuzzy PI regulátor. Jak již bylo zmíněno výše, k optimalizaci byla použita metoda Nelder-Mead a jako kritériální funkce bylo zvoleno ITAE kritérium. Důvodem volby tohoto kritéria byla skutečnost, že ITAE kritérium násobí odchylku výstupu od požadované hodnoty uplynulým časem, čímž více penalizuje odchylky v pokročilejší fázi regulačního děje. V této kapitole byly rovněž prezentovány dosažené výsledky optimalizovaného regulátoru při řízení identifikovaného modelu reálné soustavy v prostředí Simulink. Z tabulky 8.1 vyplývá, že pro každou testovanou požadovanou hodnotu dosáhl fuzzy regulátor nižší hodnoty ITAE kritéria v porovnání s klasickým lineárním regulátorem, který byl také optimalizován stejnou optimalizační funkcí jako fuzzy regulátor. Na obrázku 8.2 je zobrazeno porovnání odezev regulačních obvodů s fuzzy a lineárním regulátorem na požadovanou hodnotu 300 RPM. Fuzzy regulátor vyreguloval na požadovanou hodnotu bez překmitu a její ustálené hodnoty dosáhl po přibližně 3 sekundách. Lineární regulátor zaznamenal 7% překmit požadované hodnoty a následně malý podkmit. K ustálení na požadované hodnotě došlo za 8 sekund. Hodnota ITAE kritéria byla pro fuzzy regulátor vyčíslena na 245,86 a pro lineární regulátor na 385,10.

Kapitola 9 přiblížila postup přenesení vygenerovaného kódu z prostředí Simulink do programu TIA Portal v15.1. K vygenerování kódu byl využit toolbox MATLAB Simulink PLC Coder. V této kapitole byly rovněž prezentovány dosažené výsledky navrženého a optimalizovaného fuzzy regulátoru při řízení reálné soustavy v podobě modelu větrného tunelu. Na obrázku 9.4 je znázorněno porovnání naměřených odezev regulačních obvodů s fuzzy a lineárním regulátorem na požadovanou hodnotu 300 RPM. Z obrázku vyplývá, že při porovnání obou regulátorů dosáhl lepšího výsledku regulace fuzzy regulátor, který reguloval s menším překmitem požadované hodnoty. Ustálenou požadovanou hodnotu dosáhly oba regulátory přibližně ve stejný čas. Hodnota ITAE kritéria byla pro fuzzy regulátor vyčíslena na 11 338 a pro lineární regulátor na 11 735. Při porovnání s odezvami na obrázku 8.2 získanými v Simulinku, je viditelný rozdíl v průběhu odezvy. Tento rozdíl byl způsoben převážně identifikací modelu větrného tunelu, kdy její přesnost dosáhla 93,2 %. Navíc v této oblasti požadovaných otáček má model silné nelineární závislosti mezi vstupem a výstupem, a proto je i identifikace v této oblasti obtížná. Oba regulátory byly optimalizovány pro identifikovanou soustavu a po přenesení do PLC nebyly upravovány. Pro lepší výsledky při řízení reálné soustavy by bylo zapotřebí optimalizovat parametry regulátoru až přímo v PLC (tzv. online optimalizace). Toto řešení však není zcela jednoduché na implementaci. V rámci testování byly oba regulátory (tj., fuzzy i lineární) vyzkoušeny na schopnost vypořádání se s neočekávanou

chybou. Tuto chybu představovalo ruční zpomalení nebo úplné zastavení rotace inkrementálního enkodéru. Z tohoto testování vyplynulo, že se s neočekávanou chybou lépe vypořádá fuzzy regulátor. Při řízení reálné soustavy bylo rovněž vyzorováno, že významný vliv na průběh regulace mají i fyzikální vlastnosti vzduchu protékajícího větrným tunelem a to zejména jeho teplota a procento nasycení vodními parami. Bylo zjištěno, že při měření stejné závislosti ve dvou různých dnech lze získat významně odlišné výsledky. Tyto vlivy by bylo pravděpodobně možné potlačit zvětšením průměru větrného tunelu.



# Literatura

- [1] JURA, Pavel. *Základy fuzzy logiky pro řízení a modelování*. Brno, 2002. Skriptum. Vysoké učení technické v Brně.
- [2] Mamdani and Sugeno Fuzzy Inference Systems: MATLAB and Simulink. *Help Center for MATLAB, Simulink and other MathWorks products* [online]. [cit. 2021-11-28]. Dostupné z: <https://www.mathworks.com/help/fuzzy/types-of-fuzzy-inference-systems.html>
- [3] *Sunon: Specification for approval*. Portál [www.official.cz](http://www.official.cz) [online]. [cit. 2021-11-21]. Dostupné z: [https://www.official.cz/static/\\_dokumenty/4/4/3/ME80252V3-0000-G99-D08031290G-00-0.pdf](https://www.official.cz/static/_dokumenty/4/4/3/ME80252V3-0000-G99-D08031290G-00-0.pdf)
- [4] Ventilátor 24V/49mA, 80x80x25mm, 2600RPM, 28dB, MagLev Vapo ME80252V3-000U-G9. *GM electronic | elektronické součástky, komponenty . / GM electronic, spol. s.r.o.* [online]. [cit. 2021-11-21]. Dostupné z: <https://www.gme.cz/ventilator-sunon-me80252v3-g99>
- [5] Odpadní trubka s hrdlem HTEM, DN 75, délka 150 mm. *AAA-radiátory - komplexní systém topení ze nejlepší ceny na trhu* [online]. [cit. 2021-11-24]. Dostupné z: <https://www.aaaradiatory.cz/odpadni-trubka-s-hrdlem-htem-dn-75-delka-150-mm-p7466/>
- [6] Inkrementální enkodér 600 P/R. *Dratek.cz: VELKOOBCHOD, MALOOBCHOD S ARDUINEM* [online]. [cit. 2021-11-21]. Dostupné z: <https://dratek.cz/arduino/51385-inkrementalni-enkoder-360-p-r-2.html>
- [7] Inkrementální enkodér 600 P/R. *Portál www.dratek.cz* [online]. [cit. 2021-11-21]. Dostupné z: <https://dratek.cz/docs/produkty/1/1274/1609837184.pdf>
- [8] SIMATIC S7-1200 | Řídicí systémy SIMATIC PLCs | Siemens Czech Republic: Přizpůsobí se veškerým vašim požadavkům. *Siemens Česká republika* [online]. [cit. 2021-11-21]. Dostupné z: <https://new.siemens.com/cz/cs/products/automation/systems/industrial/plc/s7-1200.html>
- [9] 6ES7214-1AG40-0XB0 Siemens - Datasheet PDF and Technical Specs *All About Circuits - Electrical Engineering and Electronics Community* [online]. [cit. 2022-04-10]. Dostupné z: <https://www.allaboutcircuits.com/electronic-components/datasheet/6ES72141AG400XB0--Siemens/>
- [10] 6ES7214-1AG40-0XB0. *Industry Mall - Siemens Česká republika* [online]. [cit. 2021-11-21]. Dostupné z: <https://mall.industry.siemens.com/mall/cs/cz/Catalog/Product/6ES7214-1AG40-0XB0>

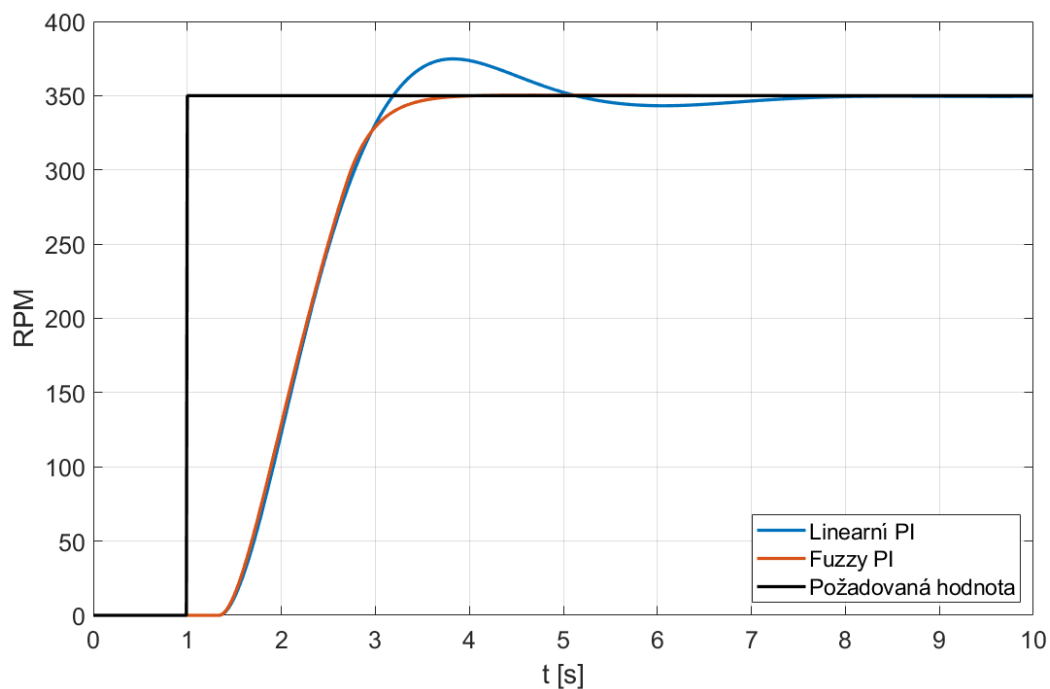
- [11] BIERLAIRE, Michel. *Optimization: principles and algorithms* Second edition. Lausanne: EPFL Press, 2018. ISBN 9782889152797.
- [12] Optimizing Nonlinear Functions - MATLAB and Simulink. *Help Center for MATLAB, Simulink and other MathWorks products*[online]. [cit. 2021-12-12]. Dostupné z: <https://www.mathworks.com/help/matlab/math/optimizing-nonlinear-functions.html>
- [13] BLAHA, Petr. *Další přístupy k identifikaci* [online]. Brno [cit. 2021-12-12]. Přednášková prezentace předmětu MPC-MID. Vysoké učení technické v Brně. Dostupné z: <https://moodle.vut.cz/pluginfile.php/12095/course/section/21729/cli.pdf>
- [14] DOSTÁL, Zdeněk a Petr BEREMLIJSKI. *Metody optimalizace* [online]. Ostrava [cit. 2021-12-21]. Přednášková prezentace předmětu MPC-MID. Vysoké učení technické v Brně. Dostupné z: [https://mi21.vsb.cz/sites/mi21.vsb.cz/files/metody\\_optimalizace.pdf](https://mi21.vsb.cz/sites/mi21.vsb.cz/files/metody_optimalizace.pdf). Skriptum. Vysoká škola báňská - Technická univerzita Ostrava.
- [15] What are Hammerstein-Wiener Models?. *Center for MATLAB, Simulink and other MathWorks products* [online]. [cit. 2021-12-30]. Dostupné z: <https://www.mathworks.com/help/ident/ug/what-are-hammerstein-wiener-models.html>
- [16] Trapezoidal numerical integration - MATLAB trapz. *Center for MATLAB, Simulink and other MathWorks products* [online]. [cit. 2022-04-10]. Dostupné z: <https://www.mathworks.com/help/matlab/ref/trapz.html>

# Seznam příloh

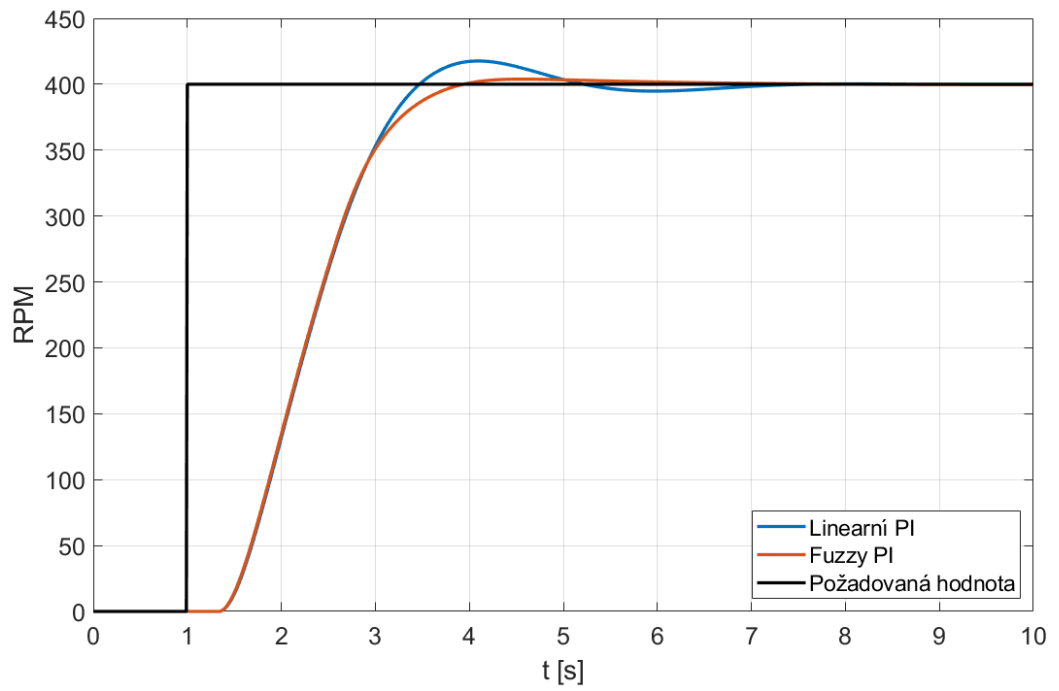
A	Odezvy regulačního obvodu po optimalizaci regulátoru	80
B	Obsah elektronické přílohy	86

## A Odezvy regulačního obvodu po optimalizaci regulátoru

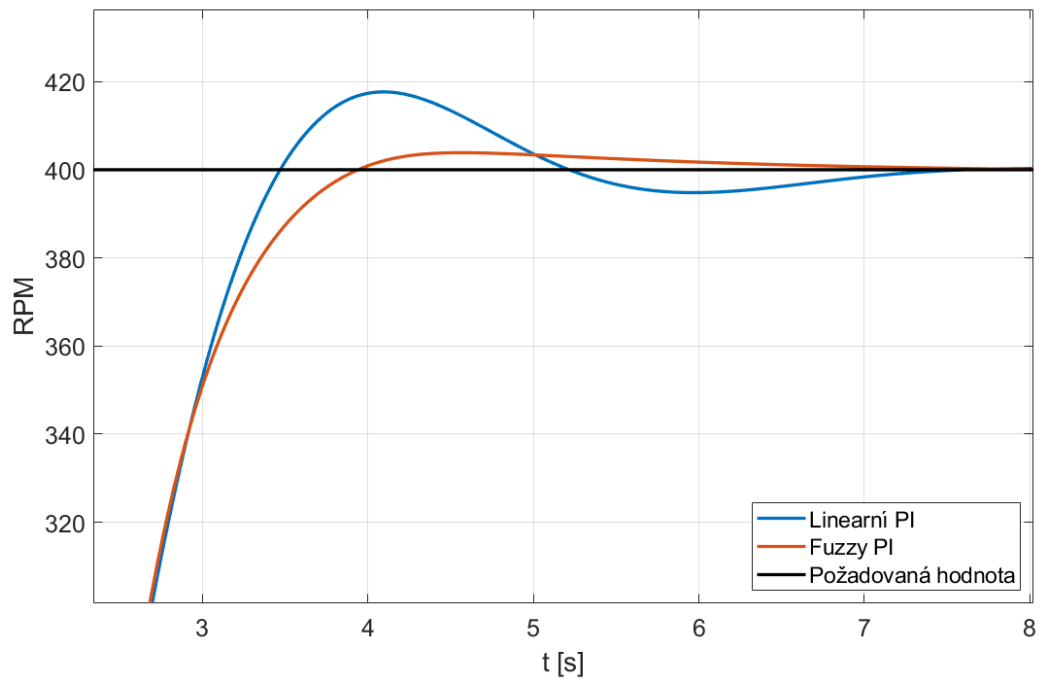
V této příloze jsou uvedeny obrázky zachycující odezvy regulačního obvodu simulovaného v Simulinku s fuzzy PI regulátorem pro jednotlivé požadované hodnoty. Pro porovnání je také vykreslena odezva regulačního obvodu s lineárním PI regulátorem.



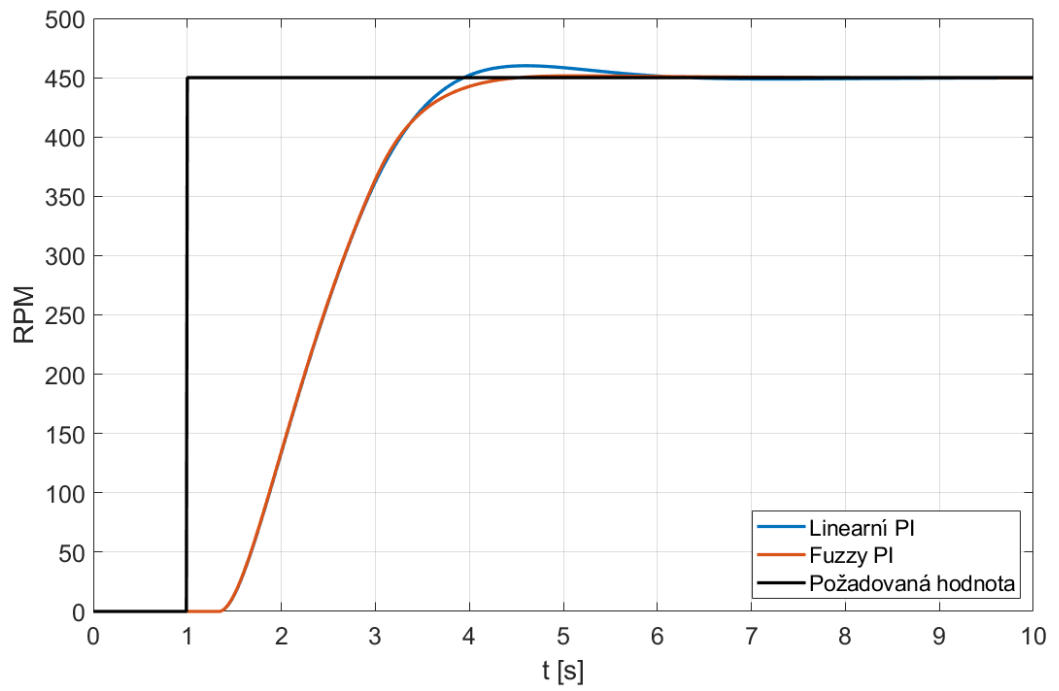
Obr. A.1: Odezva obvodu na požadovanou hodnotu 350 RPM



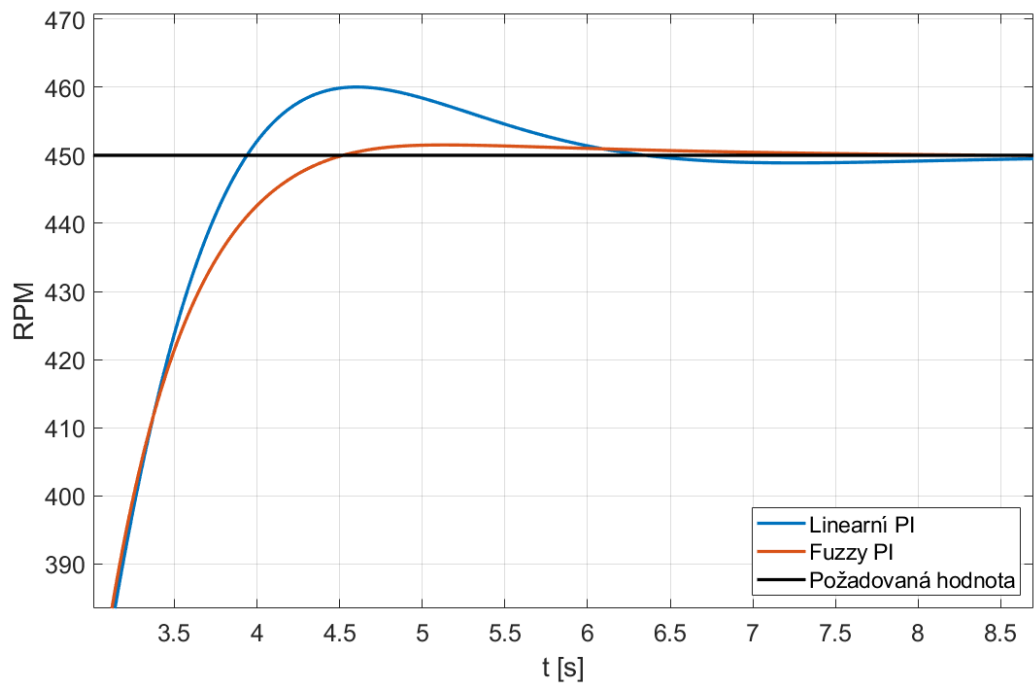
Obr. A.2: Odezva obvodu na požadovanou hodnotu 400 RPM



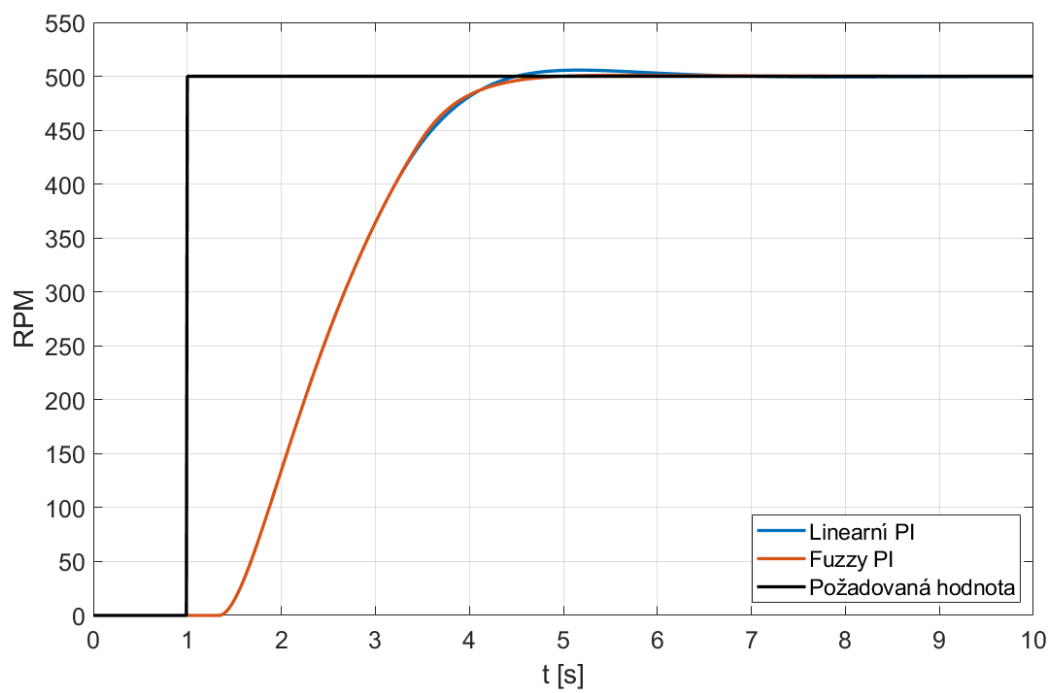
Obr. A.3: Odezva obvodu na požadovanou hodnotu 400 RPM v detailu



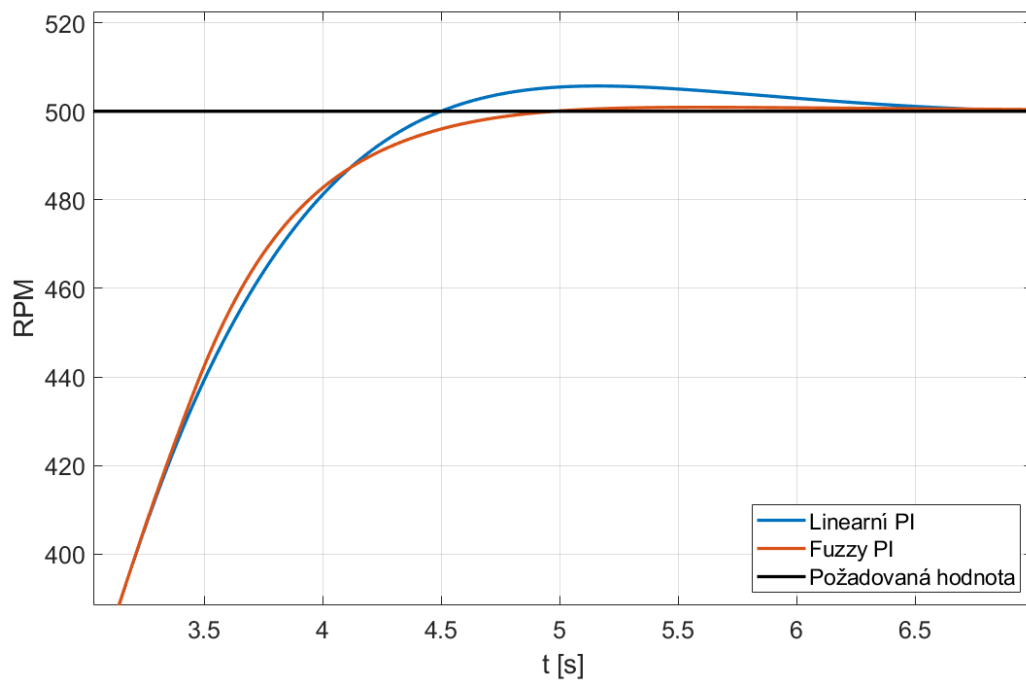
Obr. A.4: Odezva obvodu na požadovanou hodnotu 450 RPM



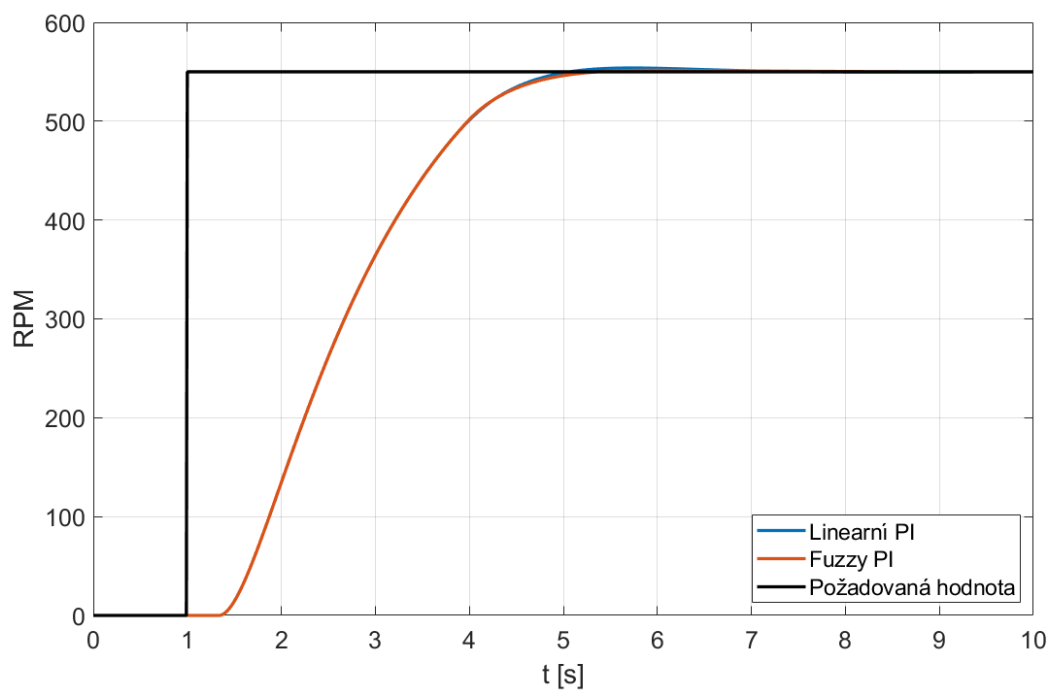
Obr. A.5: Odezva obvodu na požadovanou hodnotu 450 RPM v detailu



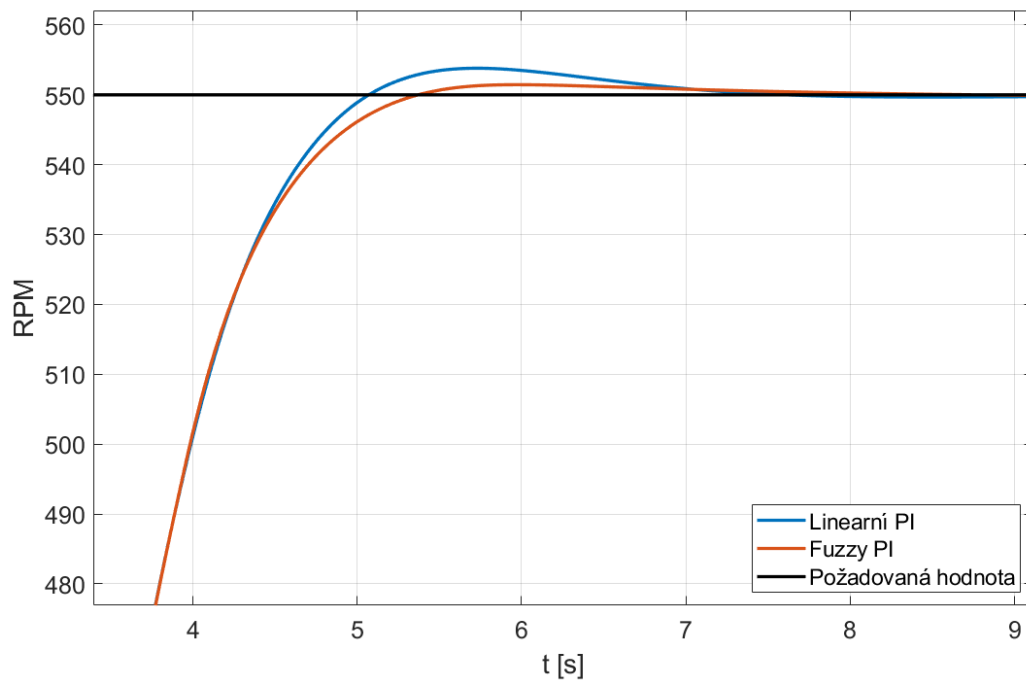
Obr. A.6: Odezva obvodu na požadovanou hodnotu 500 RPM



Obr. A.7: Odezva obvodu na požadovanou hodnotu 500 RPM v detailu

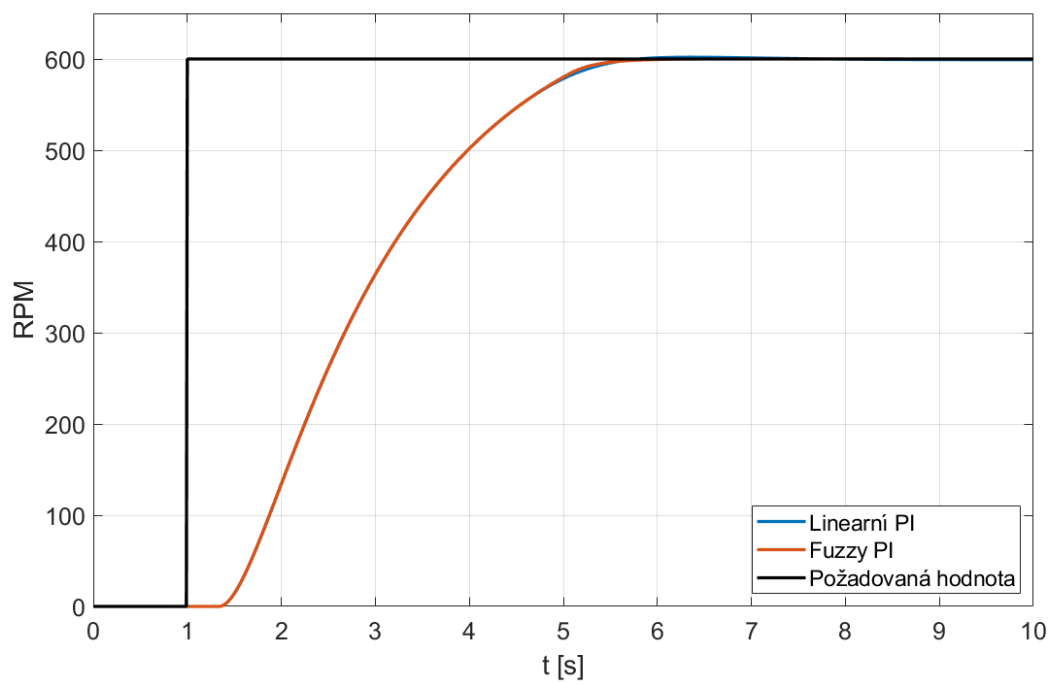


Obr. A.8: Odezva obvodu na požadovanou hodnotu 550 RPM

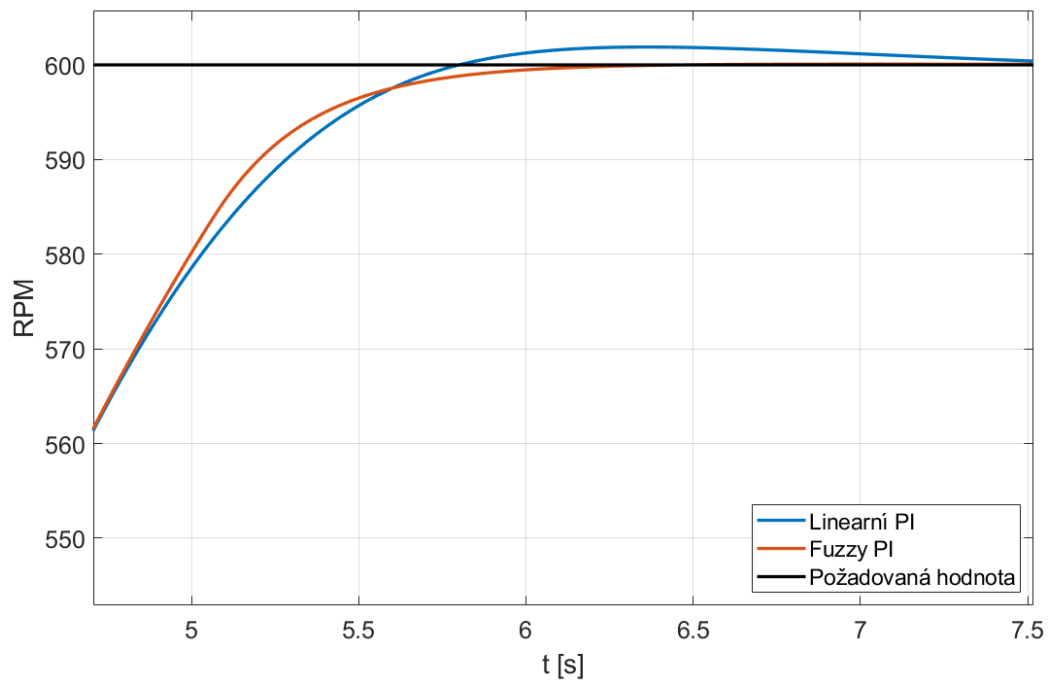


Obr. A.9: Odezva obvodu na požadovanou hodnotu 550 RPM v detailu





Obr. A.10: Odezva obvodu na požadovanou hodnotu 600 RPM



Obr. A.11: Odezva obvodu na požadovanou hodnotu 600 RPM v detailu

## B Obsah elektronické přílohy

Seznam přiložených souborů:

```
/.....kořenový adresář přiloženého archivu
├── elektronicka-verze-prace ..... elektronická verze diplomové práce
│   └── xpisko02-dp.pdf
├── matlab ..... zdrojové soubory pro MATLAB 2021a a novější
│   ├── main.m ..... hlavní soubor
│   ├── hammerstein.slx ..... regulační obvod v Simulinku
│   ├── PRBS-2V0-2.mat ..... identifikovaná soustava
│   ├── fuzzy-template.mat ..... šablona fuzzy PI regulátoru
│   ├── evaluate-itae-fuzzy.m ..... skript výpočtu ITAE kritéria pro fuzzy
│   ├── evaluate-itae-linear.m ..... skript výpočtu ITAE kritéria pro lineární
│   ├── fuzzywrite.m ..... skript pro zápis parametrů do regulátoru
│   ├── optimized.m ..... skript s optimalizovanými parametry
│   ├── hammerstein-plc.slx ..... obvod pro generování kódu do PLC
│   ├── hammerstein-plc-linear.slx ..... obvod pro generování kódu do PLC
│   └── real-data.mat ..... naměřená data z reálného systému
├── tiaportal ..... soubory projektu pro TIA Portal v15.1
│   └── tunel.ap15-1 ..... hlavní soubor projektu
```