

**KATEDRA ELEKTRICKÝCH POHONŮ A
TRAKCE**

**ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ
V PRAZE**



FAKULTA ELEKTROTECHNICKÁ
ODHAD PARAMETRŮ ASYNCHRONNÍHO
MOTORU S VYUŽITÍM UMĚLÝCH
NEURONOVÝCH SÍTÍ

DIPLOMOVÁ PRÁCE

Bc. TOMÁŠ ŘEHOŘEK

KVĚTEN 2022

VEDOUcí PRÁCE: Ing. ONDŘEJ LIPČÁK

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: Řehořek Jméno: Tomáš Osobní číslo: 466098
Fakulta/ústav: Fakulta elektrotechnická
Zadávající katedra/ústav: Katedra elektrických pohonů a trakce
Studijní program: Elektrotechnika, energetika a management
Specializace: Elektrické pohony

II. ÚDAJE K DIPLOMOVÉ PRÁCI

Název diplomové práce:

Odhad parametrů asynchronního motoru s využitím umělých neuronových sítí

Název diplomové práce anglicky:

Estimation of induction motor parameters using artificial neural networks

Pokyny pro vypracování:

1. Uveďte základní vektorové rovnice asynchronního motoru odvozené za zjednodušujících předpokladů.
2. Popište princip nejpoužívanějšího vektorového řízení orientovaného na vektor rotorového toku včetně používaných matematických modelů.
3. Proveďte rešerši o možnostech využití umělých neuronových sítí (ANN) pro odhad parametrů vektorově řízeného asynchronního motoru.
4. Implementujte model vektorového řízení orientovaného na vektor rotorového toku v prostředí MATLAB/Simulink.
5. Do vytvořeného modelu řízení implementujte alespoň dva vybrané přístupy odhadu parametrů asynchronního motoru založené na ANN.

Seznam doporučené literatury:

- [1] B.K. Bose, Modern power electronics and AC drives, Upper Saddle River, NJ: Prentice Hall, 2002.
- [2] B. Karanayil, M. F. Rahman and C. Grantham, "Online Stator and Rotor Resistance Estimation Scheme Using Artificial Neural Networks for Vector Controlled Speed Sensorless Induction Motor Drive," in IEEE Transactions on Industrial Electronics, vol. 54, no. 1, pp. 167-176, Feb. 2007.
- [3] S. Haykin, Neural networks and learning machines, 3rd. Upper Saddle River: Pearson Education, 2009.

Jméno a pracoviště vedoucí(ho) diplomové práce:

Ing. Ondřej Lipčák katedra elektrických pohonů a trakce FEL

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) diplomové práce:

Datum zadání diplomové práce: 09.02.2022

Termín odevzdání diplomové práce: 20.05.2022

Platnost zadání diplomové práce: 30.09.2023

Ing. Ondřej Lipčák
podpis vedoucí(ho) práce

podpis vedoucí(ho) ústavu/katedry

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

III. PŘEVZETÍ ZADÁNÍ

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

Datum převzetí zadání

Podpis studenta

PODĚKOVÁNÍ

Zde bych velice rád poděkoval svému vedoucímu Ing. Ondřeji Lipčákovi za cenné rady, trpělivost a čas, který mi věnoval při vypracovávání této práce.

PROHLÁŠENÍ

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

V Praze dne 19. května 2022

.....

ABSTRAKT

Tato diplomová práce má za cíl prozkoumat možnost použití umělých neuronových sítí jako součásti tzv. adaptivního systému s referenčním modelem pro odhad parametrů asynchronního motoru za účelem zlepšení kvality vektorové regulace. Cílem je především odhad rotorového a statorového odporu motoru a případně i mechanické úhlové rychlosti.

V první kapitole je představen stručný úvod do neuronových sítí. Následuje kapitola zaměřená především na matematický popis asynchronního motoru. Na tu pak navazuje teorie vektorové regulace. Dále je popsána poslední část pohonu, a to napěťový střídač s modulací prostorového vektoru. Poté je v další kapitole vysvětlen model celého pohonu pro odhad parametrů v prostředí MATLAB/Simulink. V závěru práce jsou pak prezentovány simulované výsledky včetně jejich kritického zhodnocení.

Klíčová slova: asynchronní motor, vektorová řízení, umělé neuronové sítě, adaptivní systém s referenčním modelem, MATLAB, Simulink, odhad parametrů, odhad rychlosti

ABSTRACT

This thesis aims to investigate the possibility of using artificial neural networks as part of a model reference adaptive system for estimation of the induction motor parameters to improve the quality of vector control. The main objective is to estimate the rotor and stator resistance of the motor and possibly the mechanical angular velocity.

In the first chapter, a brief introduction to neural networks is presented. This is followed by a chapter mainly focused on the mathematical description of an induction motor. Then, the theory of vector control follows. Next, the last part of the drive is described, namely the voltage-source inverter with space vector modulation. Then, in the next section, the model of the whole drive for the parameter estimation in MATLAB/Simulink environment is explained. The thesis is then concluded by a presentation of the simulated results, including their critical evaluation.

Keywords: induction motor, vector control, artificial neural network, model reference adaptive system, MATLAB, Simulink, parameter estimation, speed estimation

OBSAH

ÚVOD	1
KAPITOLA 1: UMĚLÉ NEURONOVÉ SÍTĚ	2
1.1 REÁLNÝ NEURON	3
1.2 UMĚLÝ NEURON	4
1.2.1 Aktivační funkce	5
1.2.1.1 Lineární aktivační funkce.....	5
1.2.1.2 Binární aktivační funkce.....	6
1.2.1.3 Nelineární aktivační funkce.....	7
1.3 ARCHITEKTURA SÍTÍ	8
1.3.1 Dopředné sítě	8
1.3.2 Rekurentní sítě.....	10
1.4 PROCES UČENÍ UMĚLÝCH NEURONOVÝCH SÍTÍ	10
1.4.1 Učení s učitelem.....	11
1.4.2 Učení bez učitele.....	11
1.4.3 Pravidla učení.....	12
1.4.4 Algoritmus zpětného šíření.....	12
KAPITOLA 2: ASYNHRONNÍ MOTOR	16
2.1 KONSTRUKCE A PRINCIP ČINNOSTI	16
2.2 MATEMATICKÝ MODEL	18
KAPITOLA 3: VEKTOROVÉ ŘÍZENÍ ASNCHRONNÍHO MOTORU	21
3.1 PRINCIP VEKTOROVÉ REGULACE	21
3.2 MATEMATICKÉ MODEL Y PRO VEKTOROVÉ ŘÍZENÍ ASYNCHRONNÍHO MOTORU	22
3.2.1 Proudový I-n model	22
3.2.2 Napěťový U-I model.....	23
3.2.3 Transformační úhel.....	23
3.3 FREKVENČNÍ MĚNIČ A JEHO ŘÍZENÍ	24
3.3.1 Třífázový napěťový střídač.....	24
3.3.2 Pulzně-šířkové řízení modulací prostorového vektoru	26
KAPITOLA 4: MODEL REFERENCE ADAPTIVE SYSTEM – MRAS	30
KAPITOLA 5: ODHAD PARAMETRŮ ASYNCHRONNÍHO MOTORU POMOCÍ UMĚLÝCH NEURONOVÝCH SÍTÍ	31
5.1 ODHAD ROTOROVÉHO ODPORU A ELEKTRICKÉ ÚHLOVÉ RYCHLOSTI ..	31
5.2 ODHAD STATOROVÉHO ODPORU	34
KAPITOLA 6: MODEL POHONU S ASYNCHRONNÍM MOTOREM V PROSTŘEDÍ MATLAB SIMULINK	37
6.1 MODEL ASYNCHRONNÍHO MOTORU A STŘÍDAČE	39
6.2 MODEL REGULÁTORU S ODVAZBENÍM	42
6.3 MODEL PROUDOVÉHO I-N MODELU	44

6.4	MODEL PRO ODHAD ROTOROVÉHO ODPORU A OTÁČEK POMOCÍ KOMBINACE MRAS A ANN	44
6.5	MODEL PRO ODHAD STATOROVÉHO ODPORU POMOCÍ KOMBINACE MRAS A ANN.....	49
	KAPITOLA 7: SIMULOVANÉ PRŮBĚHY MODELU	51
7.1	ZÁKLADNÍ PRŮBĚHY	51
7.2	PRŮBĚHY ODHADU ROTOROVÉHO ODPORU S POTLAČENOU VAHOU w_3 54	
7.3	PRŮBĚHY ODHADU ROTOROVÉHO ODPORU BEZ POTLAČENÍ VÁHY w_3 60	
7.4	ODHAD STATOROVÉHO ODPORU	61
7.5	ODHAD ELEKTRICKÉ ÚHLOVÉ RYCHLOSTI ROTORU.....	64
	ZÁVĚR	68

SEZNAM OBRÁZKŮ

Obr. 1-1 Blokový diagram znázorňující nervový systém.....	3
Obr. 1-2 Stavba neuronu.	4
Obr. 1-3 Blokový model umělého neuronu označeného jako „k”.....	4
Obr. 1-4 Průběh lineární aktivační funkce s omezením na ± 1 a bez omezení.....	6
Obr. 1-5 Průběhy pro různé binární aktivační funkce.....	7
Obr. 1-6 Průběh nelineárních aktivačních funkcí.....	7
Obr. 1-7 Průběh ReLU aktivační funkce	8
Obr. 1-8 Architektura jednovrstvého perceptronu a vícevrstvého perceptronu.....	9
Obr. 1-9 Architektura rekurentní sítě.....	10
Obr. 1-10 Blokový diagram učení s učitelem.....	11
Obr. 1-11 Minimalizace kvadratické chyby pomocí metody poklesu gradientu.	15
Obr. 2-1 Konstrukce asynchronního motoru s klecovým rotorem.....	17
Obr. 3-1 Transformační úhel mezi souřadnicovým systémem $\alpha\beta$ a dq	24
Obr. 3-2 Schéma dvouúrovňového střídače napětí.....	25
Obr. 3-3 Vektorový prostor daný základními vektory napětí střídače a jeho rozdělení na kvadranty a sektory.	26
Obr. 3-4 Příklad realizace žádaného vektoru napětí pomocí dvou krajních vektorů pro sektor $S1$	27
Obr. 3-5 Vzory pro spínání tranzistorů v jednotlivých sektorech.....	28
Obr. 4-1 Obecné nejjednodušší schéma MRAS.	30
Obr. 5-1 Proudový I-n model pomocí umělé neuronové sítě pro výpočet složek vektoru magnetického toku rotoru.....	32
Obr. 5-2 Schéma kombinace MRAS a ANN pro odhad $R1$, $R2$ a ω	33
Obr. 5-3 Proudový I-n model pomocí umělé neuronové sítě pro výpočet složek vektoru magnetického toku rotoru.....	36
Obr. 6-1 Celkový model pohonu.....	38
Obr. 6-2 Detail celého bloku “Triggered subsystem”.	38
Obr. 6-3 Celkový model asynchronního motoru.....	39
Obr. 6-4 Model statoru.....	40
Obr. 6-5 Model rotoru.....	40
Obr. 6-6 Schéma modelu pro výpočet hnacího momentu a otáček.	41
Obr. 6-7 Model měniče.....	41
Obr. 6-8 Schéma modelu s regulátorem, odvazbením a transformacemi.....	42
Obr. 6-9 Model regulátoru.....	42

Obr. 6-10 Model s Clarkové transformací proudu, proudovým modelem a Parkovou transformací.	44
Obr. 6-11 Proudový I-n model s výpočtem transformačního úhlu a modulu vektoru magnetického toku rotoru.	44
Obr. 6-12 Část spínaného bloku pro odhad R_2 a ω pomocí MRAS s adaptivní ANN.	45
Obr. 6-13 Schéma pro kompenzaci DC offsetu.	45
Obr. 6-14 Napěťový U-I model s kompenzací DC offsetu.	46
Obr. 6-15 Proudový I-n model pomocí ANN.	47
Obr. 6-16 Úprava vah W_1, W_2 a W_3 podle algoritmu zpětného šíření a odhad R_2 a ω	47
Obr. 6-17 Část spínaného bloku pro odhad R_1 pomocí MRAS a ANN.	49
Obr. 6-18 Model umělé neuronové sítě pro odhad složek $i_{1\alpha}$ a $i_{1\beta}$	50
Obr. 6-19 Úprava vah W_4, W_5 a W_6 podle algoritmu zpětného šíření a odhad R_1	50
Obr. 7-1 Průběh hnacího momentu při rozběhu a následném zatížení v čase $t=1,5$ s.	52
Obr. 7-2 Průběh skutečných a žádaných otáček při rozběhu a následném zatížení v čase $t=1,5$ s.	52
Obr. 7-3 Průběh složek a modulu vektoru statorového proudu v souřadnicovém sstému $\alpha\beta$	53
Obr. 7-4 Průběh složek a modulu vektoru magnetického toku rotoru v souřadnicovém sstému $\alpha\beta$	53
Obr. 7-5 Průběh $\psi_{2\alpha}$ pro referenční U-I a adaptivní ANN model a chyba odhadu daná jejich rozdílem.	55
Obr. 7-6 Průběh velikosti ψ_2 pro referenční U-I a adaptivní ANN model.	55
Obr. 7-7 Průběh úpravy vah W_1, W_2 a W_3	56
Obr. 7-8 Průběh referenční a odhadnuté složky $\psi_{2\alpha}$ pro koeficient učení $\eta_1 = 0,001$	57
Obr. 7-9 Průběh W_1 s koeficientem učení $\eta_1 = 0,001$	57
Obr. 7-10 Průběh odhadu R_2 pomocí W_1 a W_3 při postupném navýšení odporu o 30, 60 a 100 %.	58
Obr. 7-11 Průběh odhadu R_2 pomocí W_1 pro vyšší koeficient učení $\eta_1 = 0,001$ a při postupném navýšení odporu o 30, 60 a 100 %.	59
Obr. 7-12 Průběhy odhadu R_2 pomocí W_1 při zatížení 65 Nm a 20 Nm a postupném navýšení odporu o 30, 60 a 100 %.	59
Obr. 7-13 Průběh $\psi_{2\alpha}$ pro referenční U-I a adaptivní ANN model a chyba odhadu daná jejich rozdílem bez potlačení váhy W_3	60
Obr. 7-14 Průběhy odhadu R_2 pomocí W_1 a W_3 při postupném navýšení odporu o 30, 60 a 100 % bez potlačení váhy W_3	61

Obr. 7-15 Průběhy odhadu $R2$ pomocí $W1$ při zatížení 65 Nm a 20 Nm a postupném navýšení odporu o 30, 60 a 100 %.	62
Obr. 7-16 Průběhy odhadu $R2$ pomocí $W1$ při zatížení 65 Nm a 20 Nm a postupném navýšení odporu o 30, 60 a 100 %.	62
Obr. 7-17 Průběhy odhadu $R1$ pomocí $W4$ při zatížení 65 Nm a 20 Nm a postupném navýšení odporu o 30, 60 a 100 %.	63
Obr. 7-18 Průběhy odhadu $R1$ pomocí $W4$ při postupném navýšení rotorového odporu o 30, 60 a 100 % a se jmenovitou hodnotou statorového odporu.	64
Obr. 7-19 Průběh $\psi2\alpha$ pro referenční U-I a adaptivní ANN model a chyba odhadu daná jejich rozdílem při odhadu rychlosti.	65
Obr. 7-20 Průběh velikosti vektoru $\psi2$ pro referenční U-I a adaptivní ANN model a chyba odhadu daná jejich rozdílem při odhadu rychlosti.	65
Obr. 7-21 Průběh žádané, skutečné a odhadnuté mechanické úhlové rychlosti při postupné změně otáček na 100, 80, 50, 30, 100, 120 a 153 $\text{rad}\cdot\text{s}^{-1}$.	66
Obr. 7-22 Průběhy odhadu $R2$ pomocí $W1$ při postupné změně žádané rychlosti na 100, 80, 50, 30, 100, 120 a 153 $\text{rad}\cdot\text{s}^{-1}$.	67
Obr. 7-23 Průběhy odhadu $R1$ pomocí $W4$ při postupné změně žádané rychlosti na 100, 80, 50, 30, 100, 120 a 153 $\text{rad}\cdot\text{s}^{-1}$.	67

SEZNAM TABULEK

Tab. 3-1 Základní vektory napětí a jejich příslušné spínací kombinace	25
Tab. 3-2 Výpočet krajních vektorů napětí pro určitý kvadrant a sektor.	29
Tab. 3-3 Stanovení aktuálního kvadrantu a sektoru pomocí složek $u_{1\beta}$, $u_{1\alpha}$ a členu l	29
Tab. 6-1 Parametry modelovaného motoru.....	37
Tab. 6-2 Konstanty pro jednotlivé PI regulatory.....	43
Tab. 7-1 Nastavení koeficientů učení a momentových faktorů pro odhad R_2 s potlačenou vahou W_3	54
Tab. 7-2 Nastavení koeficientů učení a momentových faktorů pro odhad R_2	60
Tab. 7-3 Nastavení koeficientů učení a momentových faktorů pro odhad R_1	62
Tab. 7-4 Nastavení koeficientů učení a momentových faktorů pro odhad elektrické úhlové rychlosti.....	64

ÚVOD

Asynchronní motory jsou v praxi hojně používány více jak sto let, a to díky jejich jednoduché konstrukci, robustnosti, spolehlivosti, účinnosti, nízkým pořizovacím nákladům, ekonomickému provozu, kompaktnosti či snadné velkovýrobě především v případě typů s klecovým rotorem. Ač jejich výzkum a vývoj prošel za tu dobu velkým pokrokem a dále pokračuje, tak rychlejší pokrok probíhá v oblasti výkonové elektroniky. Tradičně byl tento typ motoru používán v aplikacích s konstantní rychlostí, zatímco stejnosměrné motory byly preferovány pro pohony s proměnou rychlostí. S rozvojem výkonové elektroniky ale vznikla široká škála asynchronních motorů pro aplikace s proměnnými otáčkami. Vyžadují však komplexnější řídicí systém oproti například cize buzeným stejnosměrným motorům. Vyšší náročnost řídicího systému asynchronních motorů sebou zpočátku přinášela především vyšší cenu a také obtížnější vývoj. Tyto nedostatky ale postupem času vymizely díky výkonnějším polovodičovým součástkám, modulárnějším provedením a silnějším a rychlejším mikroprocesorům, které umožnily realizovat řízení uceleným softwarem namísto drahého hardwaru [1], [2].

Pro řízení vzniklo mnoho metod, avšak v dnešní době se v náročných aplikacích nejvíce využívá především vektorové řízení a přímé řízení momentu. Princip obou těchto variant je stejný, a to odděleně řídit magnetický tok a moment stroje. V této práci je použita první zmíněná metoda. Ta pracuje s matematickými modely, závislými na elektrických parametrech motoru, které se mohou za chodu měnit s teplotou, vlivem skin efektu či úrovní sycení. Některé z nich jako je třeba rotorový odpor nebo magnetický tok je nemožné měřit, nebo jsou tato měření drahá a těžko realizovatelná. Odpory se tak v základních variantách vektorové regulace uvažují jako konstantní a především vlivem zahřívání může docházet k jejich zvýšení až na dvojnásobek, což vede ke zhoršení kvality regulace [1], [2].

Tato diplomová práce si klade za cíl ověřit některé z možností odhadu statorového a rotorového odporu, a to především algoritmy založené na neuronových sítích a adaptivní systému s referenčním modelem (MRAS). Neuronové sítě by v této struktuře měly sloužit jako adaptivní systém schopný reagovat na změny odporů. Navíc bude prezentována i možnost pomocí stejné neuronové sítě bezsenzorově odhadovat i mechanické otáčky rotoru. Jejich měření je sice technicky dobře zvládnuté, avšak takto by mohly být dražší senzory teoreticky nahrazeny jednou neuronovou sítí, která zároveň odhaduje více veličin.

V práci je nejdříve probrána základní teorie umělých neuronových sítí, jejich struktura, způsoby učení či možnosti jejich využití. Část s teorií neuronových sítí má sloužit jako jejich základní přehled pro všechny čtenáře. Následně je pak popsán model asynchronního motoru a vektorového řízení s potřebným matematickým popisem.

Model celého pohonu je sestaven v prostředí MATLAB/Simulink a jednotlivé části jsou blíže popsány v konkrétních kapitolách. Kvalita odhadu je pak demonstrována na simulacích pro různé provozní podmínky.

KAPITOLA 1: UMĚLÉ NEURONOVÉ SÍTĚ

Umělé neuronové sítě (anglicky artificial neural network, zkráceně ANN), jak jejich název napovídá, jsou inspirovány reálným chováním neuronů v lidském mozku. Jejich historie sahá až do 40. let minulého století, kdy americký neurofyziolog Warren Sturgish McCulloch a jeho kolega, logik a matematik, Walter Pitts v roce 1943 představily práci *A logical calculus of the ideas immanent in nervous activity*. Zde představili výpočtový elektronický model, postavený na faktu, že jelikož neurony v lidském mozku vyšlou signál jenom v případě, že dojde k dosažení nějaké prahové hodnoty, tak by takovéto chování bylo možné simulovat jednoduchými logickými obvody. Takovýto model vykonával pouze logické operace a všechny koeficienty vah a prahů byly pevně nastavené. Nebyla zde tedy žádná schopnost „učit se.“ Důležitou myšlenkou bylo to, že jednotlivé části tohoto obvodu (jednotlivé „neurony“) jsou samy o sobě velice jednoduché, ale při jejich propojení do řetězců a smyček, vznikne komplexní celek, schopný vykonávat i složité operace. I přes jednoduchost jejich modelu, který měl daleko k napodobení mozku, to byla dobrá myšlenka, jenž následně vedla k dalšímu vývoji a výzkumu. Zároveň se ukázala možnost nahlížet na mozek jako na počítač složený ze základních logických operací [3], [4].

Ještě v roce 1949 přišel kanadský psycholog Donald Hebb s prací *The Organization of Behavior*, kde popisuje hypotézu, že síla synaptických spojení mezi neurony je proměnná a záleží na jejich používání. Tato variabilita je základem pro učení a paměť. Výzkum samozřejmě pokračoval i v 50. letech a zlom přišel již v roce 1958, kdy Frank Rosenblatt poprvé publikoval práci o takzvaném perceptronu. To byl první model neuronové sítě, který je platný i v dnešní době a v jistých modifikacích se stále používá. Propojil vlastně model McCullocha a Pittse se schopností „učit se“ změnou vah synapsí a prahů. Perceptron a především jeho vylepšení, takzvaný vícevrstvý perceptron je velice důležitý model jakožto obecně pro umělé neuronové sítě, tak i pro tuto práci, a proto bude ještě blíže probrán [5], [6].

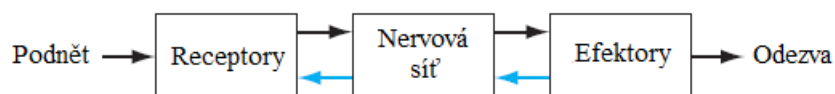
Vývoj a výzkum umělých neuronových sítí probíhal i v dalších letech velice aktivně a s jejich složitostí se rozšiřoval i rozsah jejich praktického využití. Z důvodu zaměření této práce je zbytečné zde rozebírat další historická fakta, avšak výše uvedené milníky jsou uvedeny, aby nastínili počátky jejich výzkumu. V dnešní době najdou neuronové sítě uplatnění v mnoha oborech jako jsou například [5]:

- medicína,
- ekonomie,
- kontrolní systémy,
- přeprava a doprava,
- psychologie,
- překlady textu a
- předpověď vývoje komplexních systémů.

Základní přehled o fungování umělých neuronových sítí, jejich struktura, způsoby učení či výhody a nevýhody jsou shrnuty v následujících podkapitolách.

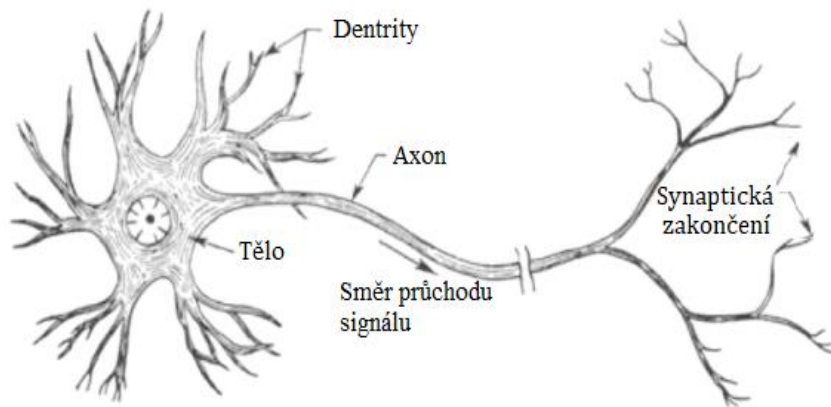
1.1 Reálný neuron

Nervové buňky neboli neurony jsou základním stavebním kamenem lidského nervového systému, který je rozvětvený po celém těle a lze na něj nahlížet jako na tříúrovňový systém znázorněný na Obr. 1-1. Řídicím centrem celého tohoto systému, a i celého těla, je mozek, který je složen z obrovské sítě neuronů. Tato síť v každém okamžiku přijímá veliké množství informací, zpracovává je a vytváří adekvátní odezvu na tyto podněty. Šipky na Obr. 1-1 vedoucí zleva doprava znázorňují dopředný přenos informací systémem a šipky jdoucí opačným směrem, znázorňují jistou zpětnou vazbu. Receptory přeměňují podněty na elektrické impulsy, které přenáší informaci do nervové sítě (mozku) a ten provádí emisi elektrických impulsů k ovládnutí efektorů. Ty nakonec přemění elektrický signál na viditelnou odezvu jakožto výstup celého systému [7], [8].



Obr. 1-1 Blokový diagram znázorňující nervový systém [7] (upraveno).

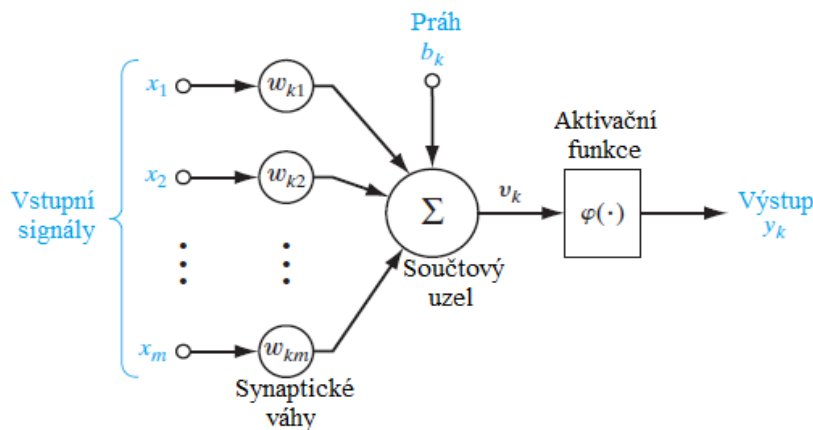
Rychlost přenosu informací v neuronu je v řádu milisekund, zatímco například rychlost křemíkových logických hradel je v řádu nanosekund. Neurony se tak dají považovat za poměrně pomalé, avšak tento nedostatek vyvažují svým obrovským množstvím, a ještě větším množstvím jejich vzájemných propojení. Odhaduje se, že v mozkové kůře je asi 100 miliard neuronů. Spojení neuronů na konci rozvětvené terminální části axonu jsou takzvané synapse. Ty zprostředkovávají přenos informace z neuronu na neuron. Axon je pak výběžek z těla neuronu, který nese vždy informaci směrem od těla k synaptickým spojením a může být dlouhý až několik metrů. Posledním důležitou částí pro přenos informace skrz neuron jsou dendrity. Jsou to rozvětvené výběžky těla neuronu, které přijímají signály a vedou je vždy směrem k jeho tělu. Stavbu neuronu lze vidět na Obr. 1-2, přičemž přenos signálu skrz neuron lze zjednodušeně popsat tak, že skrz dendrity přichází informace, ty jsou vysílány do těla a následně axonem až k jednotlivým synapsím, kde dochází k předání informace dendritům dalšího neuronu. Synaptická spojení se zároveň mohou posilovat nebo naopak oslabovat a například schopnost učit se, tak nemusí probíhat tvorbou nových neuronů, ale právě zesílením zmíněných vazeb. Toto je důležitá vlastnost napodobovaná i umělými sítěmi [1], [9], [10].



Obr. 1-2 Stavba neuronu [10] (upraveno).

1.2 Umělý neuron

Na základě vlastností reálných neuronů uvedených v předchozí kapitole vznikl základní model neuronu umělého. Na Obr. 1-3 je blokový diagram modelu neuronu, který slouží jako základ k dalšímu zkoumání složitějších modelů umělých neuronových sítí [7].



Obr. 1-3 Blokový model umělého neuronu označeného jako „k” [7] (upraveno).

Vstupní signály x_1, x_2, \dots, x_m jsou typicky spojité hodnoty, avšak může se jednat i o diskrétní pulsy. Následně je signál vynásoben synaptickými vahami $w_{k1}, w_{k2}, \dots, w_{km}$, přičemž každému vstupu náleží vlastní synaptická váha. Zde je důležité dodržet pořadí indexace. Pro synaptickou váhu w_{k1} značí první index, že se jedná o neuron k a druhý index označuje spojení se vstupním signálem 1. Hodnoty synaptických vah mohou být jak kladné, tak záporné. Všechny lineární kombinace vstupů a vah jsou pak sečteny v součtovém uzlu, což je vyjádřeno členem u_k . Do toho uzlu vstupuje ještě takzvaný práh (anglicky bias), označený jako b_k . Ten ovlivňuje hodnotu aktivačního potenciálu v_k a to buďto kladně nebo záporně. Nakonec je na aktivační potenciál aplikována aktivační funkce $\varphi(\cdot)$ a tím je dán výstup neuronu. To může být výstup celého systému, nebo pouze vstup do dalšího neuronu. Matematicky lze neuron k na Obr. 1-3 vyjádřit následujícími rovnicemi [7], [1], [11], [12]

$$u_k = \sum_{j=1}^m w_{kj} x_j, \quad (1-1)$$

$$v_k = u_k + b_k, \quad (1-2)$$

$$y_k = \varphi\left(\sum_{j=1}^m (w_{kj} x_j) + b_k\right) = \varphi(v_k). \quad (1-3)$$

Práh můžeme nahradit novým vstupem x_0 jehož hodnota je pevně dána jako +1 a synaptickou vahou $w_{k0} = b_k$. Potom lze rovnice (1-2) upravit do tvaru dané rovnicí (1-4) [7]

$$v_k = \sum_{j=0}^m w_{kj} x_j. \quad (1-4)$$

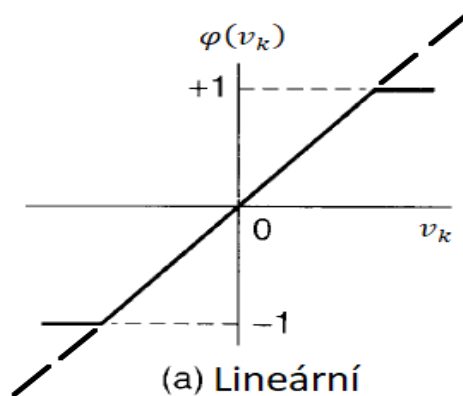
Výstup daný rovnicí (1-3) je naprosto ekvivalentní ať jsou použity rovnice (1-1) a (1-2) k vyjádření aktivačního potenciálu v_k , nebo je použita rovnice (1-4). Významnou roli hraje výběr aktivační funkce. Těch je více druhů a podrobněji jsou popsány v následující kapitole [7].

1.2.1 Aktivační funkce

Aktivační funkce (nebo také přenosová či prahová funkce) $\varphi(\cdot)$ mění výstupní hodnotu v závislosti na hodnotě aktivačního potenciálu v_k a na Obr. 1-3 je vidět, že je to poslední člen před výstupem neuronu, který přímo určuje jeho hodnotu. Základní rozdělení těchto funkcí může být na binární, lineární a nelineární. Obecně existuje veliké množství aktivačních funkcí, avšak ne všechny jsou vhodné pro všechny aplikace a jsou zde tedy uvedeny především ty obecně nejpoužívanější [13].

1.2.1.1 Lineární aktivační funkce

Lineární aktivační funkce je dána předpisem $\varphi(v_k) = av_k$ a tedy na výstupu je přímo hodnota aktivačního potenciálu. Hodnota a je dána uživatelem jako konstanta, která určuje sklon. Někdy se tato funkce používá s omezením na hodnotě ± 1 , jak je znázorněno plnou čarou na Obr. 1-4, čárkovaný průběh znázorňuje lineární funkci bez omezení. Derivace této funkce je konstantní a závisí na hodnotě a , gradient je tedy také konstantní a nezávislý na vstupní hodnotě v_k . Například při použití učení pomocí metody zpětného šíření (anglicky back propagation) tak může docházet k pomalejšímu zmenšování chyby na výstupu. Používá se především jako výstupní aktivační funkce pro účely regrese (určení číselné hodnoty) a to v jednodušších aplikacích [1], [14].



Obr. 1-4 Průběh lineární aktivační funkce s omezením na ± 1 (plná čára) a bez omezení (čárkovaná čára) [1] (upraveno).

1.2.1.2 Binární aktivační funkce

Jedná se o takzvanou prahovou funkci, která je čistě binární a může tedy nabývat hodnoty 0 nebo 1. Z ní vycházející funkce signum již není binární, ale je navíc rozšířena o hodnotu -1.

- Prahová funkce: nebo také Heavisidova funkce je dána předpisem

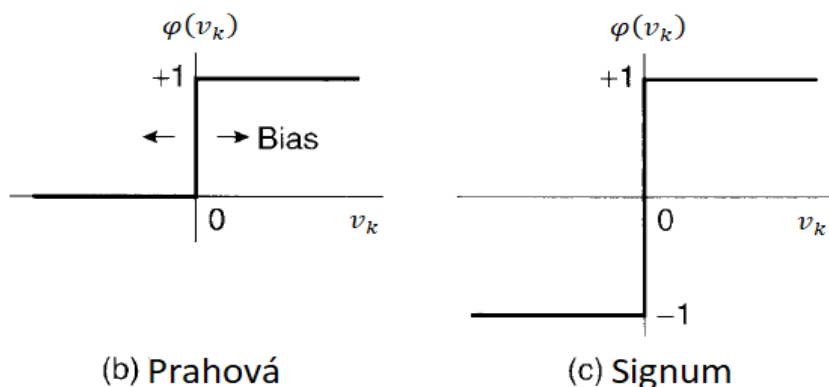
$$\varphi(v_k) = \begin{cases} 1 & \text{pokud } v_k \geq 0 \\ 0 & \text{pokud } v_k < 0 \end{cases} \quad (1-5)$$

Je to nejzákladnější a nejjednodušší aktivační funkce, která slouží k základní klasifikaci do dvou skupin, 1 nebo 0. Kvůli nulové derivaci ji nelze použít pro učení pomocí zpětného šíření, to je hlavní nevýhoda této funkce. Průběh této funkce je na Obr. 1-5. Pokud je tato funkce použita pro model s jedním umělým neuronem, pak se jedná o takzvaný perceptron [1], [14].

- Funkce signum: tuto funkce lze dostat pouze rozšířením předchozí prahové funkce na předpis

$$\varphi(v_k) = \begin{cases} 1 & \text{pokud } v_k > 0 \\ 0 & \text{pokud } v_k = 0 \\ -1 & \text{pokud } v_k < 0 \end{cases} \quad (1-6)$$

Klasifikace je tak rozšířena ještě o jednu hodnotu, avšak základní problém v podobě nulového gradientu je stejný. Průběh je dán na Obr. 1-5



Obr. 1-5 Průběhy pro různé binární aktivační funkce [1](upraveno).

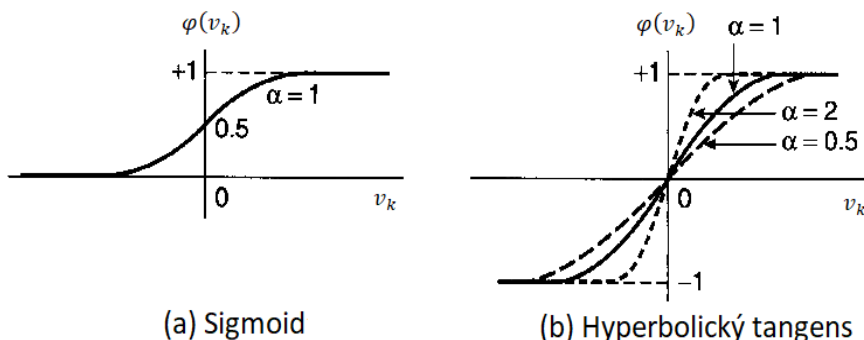
1.2.1.3 Nelineární aktivační funkce

Nelineární aktivační funkce je používanější častěji, neboť obecně jsou řešené problémy pomocí neuronových sítí nelineární. Do výpočtu pak nelinearitu vnáší právě aktivační funkce [15].

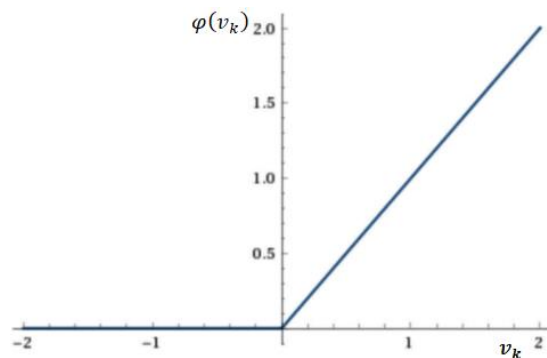
- Funkce sigmoid: předpis této funkce je $\varphi(v_k) = 1/(1 + e^{-\alpha v_k})$, pro záporné hodnoty se blíží k nule, pro $\varphi(0) = 0,5$ a pro kladné hodnoty se blíží k jedné. Jedná se o nejpoužívanější a velice rozšířenou funkci. Díky nesymetričnosti okolo nuly jsou také všechny výstupní hodnoty kladné. Průběh je na Obr. 1-6 [1], [13], [15].
- Funkce hyperbolický tangens (tanh): předpis je dán jako $\varphi(v_k) = (e^{\alpha v_k} - e^{-\alpha v_k})/(e^{\alpha v_k} + e^{-\alpha v_k})$. Je velice podobná předchozí funkci, avšak její rozsah se pro záporné hodnoty blíží k minus jedné, v nule má hodnotu nula a pro kladné hodnoty se pak blíží k jedné. Je tedy symetrická okolo nuly a oproti předchozí funkci nabývá výstup obou hodnot. Zároveň její gradient je vyšší. Průběh této funkce je na Obr. 1-6 [1], [14], [15].
- Rektifikovaná lineární aktivační funkce (ReLU): ta je dána jako

$$\varphi(v_k) = \begin{cases} v_k & \text{pokud } v_k > 0 \\ 0 & \text{pokud } v_k \leq 0 \end{cases} \quad (1-7)$$

Jedná se o jednu z nejpoužívanějších funkcí pro skryté vrstvy a její průběh je na Obr. 1-7. Její hlavní výhoda oproti výše zmíněným je v tom, že ne všechny neurony jsou aktivní současně, což zajišťuje vyšší efektivnost [13], [14].



Obr. 1-6 Průběh nelineárních aktivačních funkcí [1] (upraveno).



Obr. 1-7 Průběh ReLU aktivační funkce [16] (upraveno)

Člen α u funkce sigmoid a tanh upravuje jejich sklon, nebo se dá také říct citlivost. Neboť čím rychleji se dostávají k hraničním hodnotám, tak tím citlivější či přesnější jsou [1].

Výběr správné aktivační funkce může být v praxi problémový. Pro modely umělých neuronových sítí, které jsou složeny z více vrstev neuronů, jsou většinou aktivační funkce ve skrytých vrstvách jiné než ty na výstupu. Pro skryté vrstvy se dříve volili především funkce sigmoid a tanh, avšak v současné době je to spíše ReLU. U poslední zmíněné funkce tolik nedochází k takzvanému mizení gradientu, zatímco u zbylé dvě jsou k tomuto více náchylné. Dochází k tomu, že se gradient pro úpravu vah blíží nule a jejich další úprava proto probíhá pomalu, nebo vůbec. V takovém případě se síť při svém učení může k požadovanému výsledku přibližovat velice pomalu, nebo se do něho ani nedostane. Pro výstupní funkci záleží, zda se řeší problém regrese (odhad hodnoty) nebo klasifikace. Pro první skupinu se používá funkce lineární a pro tu druhou nelineární, a to především sigmoid. Nakonec však výběr funkce záleží na dané aplikaci a na použitém modelu neuronové sítě a neexistuje jedna univerzální volba [13], [17].

1.3 Architektura sítí

V části 1.2 byla probrána základní podoba umělého neuronu od kterého se odvíjí všechny ostatní složitější architektury. Těch je v dnešní době velké množství různých typů a vyvíjeli se postupem času spolu s aplikacemi, na které jsou neuronové sítě používány. Tyto různé modely jsou úzce spojeny i s danými algoritmy učení, které jsou dále popsány v kapitole 1.4. Základní rozdělení je na takzvané [7]:

- dopředné sítě a
- rekurentní sítě.

1.3.1 Dopředné sítě

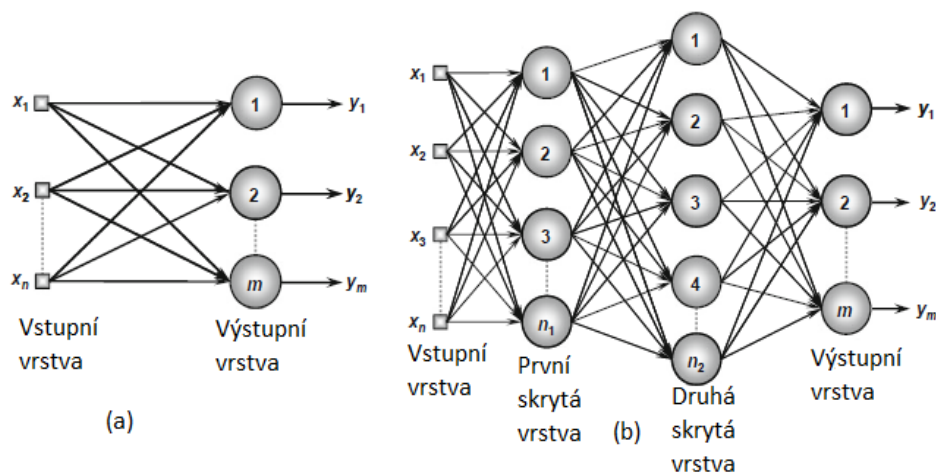
Název dopředná síť (anglicky feedforward network) je dána tím, že signál putuje od vstupu na výstup skrz jednotlivé neurony pouze jedním směrem, tedy dopředu. Tyto sítě mohou být jak jednovrstvé, tak i vícevrstvé. Do této skupiny spadá mnoho různých modelů jako jsou například [1]:

- perceptron,
- vícevrstvý perceptron (anglicky multilayer perceptron),

- adaline a madaline,
- síť s radiální bází (anglicky radial basis function network – RBFN),
- fuzzy neuronové síť (anglicky fuzzy neural network – FNN).

Existují i další, avšak tyto zmíněné patří mezi nejpoužívanější. V mezinárodní literatuře je však možné potkat se s různými výrazy, pro stejnou architekturu sítě, je proto nutné vyjasnit používané pojmy. Perceptron, jak již zbylo zmíněno, je nejjednodušší model s jedním neuronem a s prahovou aktivační funkcí. V literatuře se však tento název prolíná s jednovrstvou dopřednou sítí a často se tak lze setkat s pojmenováním jednovrstvý perceptron (anglicky single-layer perceptron). V této práci se tedy pod pojmem jednovrstvý perceptron myslí jednovrstvá dopředná síť bez ohledu na původní správný význam perceptronu, neboť je to v mezinárodní literatuře běžně používaný termín. Další jednovrstvou dopřednou sítí je i výše zmíněný adaline [1].

Jednovrstvý perceptron se tedy skládá z jednotlivých vrstev, z vrstvy vstupní a výstupní a ty jsou mezi sebou propojeny vahami. Architekturu takovéto sítě je možné vidět na Obr. 1-8 (a). To sice dává dvě vrstvy, avšak vstupní vrstva se nepočítá, neboť v ní nedochází k žádnému výpočtu. Zde je potřeba opět upozornit na nepřesnosti v literatuře, jelikož je možné se setkat s tím, že se jako vrstva počítají i vstupy a taková to struktura bývá pojmenována jako dvouvrstvá dopředná síť. Tento typ sítě je velice jednoduchý a jeho použití je tedy omezené na jednodušší úkoly. Aktivační funkce nemusí být prahová, ale může být libovolná [7], [18].

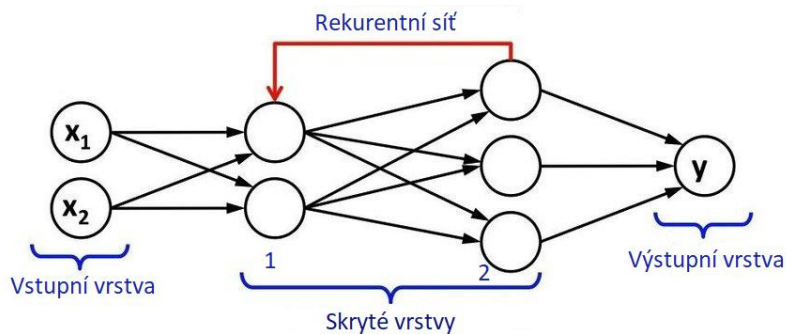


Obr. 1-8 Architektura jednovrstvého perceptronu (a) a vícevrstvého perceptronu (b) [19] (upraveno).

Vícevrstvý perceptron na rozdíl od jednovrstvého obsahuje ještě takzvané skryté vrstvy, které se nacházejí mezi vstupní a výstupní vrstvou a sestávají ze skrytých neuronů. Takovýchto skrytých vrstev může být i více. Architektura této sítě je na Obr. 1-8 (b) a stejně jako u jednovrstvého perceptronu, kde jsou vstupy a výstup spojeny pomocí vah, tak i všechny spojení do, anebo ze skrytých vrstev, jsou dány vahami. Zároveň výstup každé skryté vrstvy prochází přes aktivační funkci a ty se většinou volí stejné pro všechny skryté vrstvy. Aktivační funkce bývají ve skrytých vrstvách nelineární, neboť tato nelinearita zajišťuje možnost modelovat složitější nelineární systémy [20], [2].

1.3.2 Rekurentní síť

Tento typ má na rozdíl od dopředných sítí alespoň jednu zpětnou vazbu. Síť ani nemusí obsahovat skryté vrstvy a zpětná vazba může být čistě mezi vstupem a výstupem. Jednoduchá rekurentní síť tak může být vytvořena z obyčejné dopředné sítě pouhým přidáním zpětné vazby. Obecně se dá říct, že k výstupu systému v čase t jsou použity jak aktuální vstupy, tak i vstupy v čase $t+1$. Odezva sítě je tak dynamická a používají se k predikci nelineárních systémů, jejich identifikaci a ke klasifikaci časových vzorů. Zjednodušenou podobu architektury takové sítě lze vidět na Obr. 1-9 [7], [21], [22].



Obr. 1-9 Architektura rekurentní sítě [23] (upraveno).

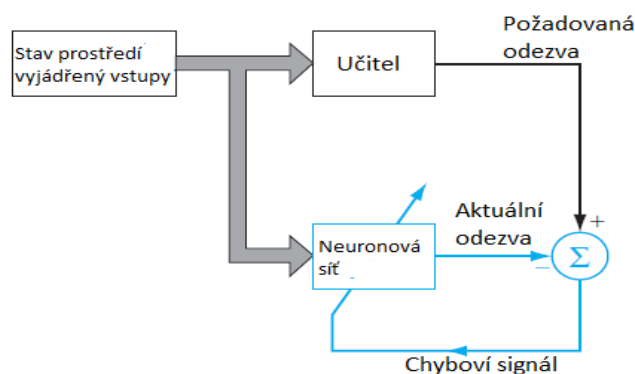
1.4 Proces učení umělých neuronových sítí

V části 1.1 bylo popsáno, jak se struktury reálných neuronů mohou učit tím, že se posilují vazby mezi nimi bez nutnosti tvorby nových neuronů. Stejně tak se umělé neuronové sítě učí změnou hodnot jednotlivých vah mezi vstupem a výstupem, či mezi jednotlivými skrytými vrstvami. Stejně jako se člověk učí od svého okolí mnoha různými způsoby, tak stejně tak se umělé sítě učí pomocí rozličných algoritmů. Jinak řečeno umělá síť není přímo naprogramována na danou aplikaci, ale je nutné ji na ni naučit, tak aby byla schopná plnit daný úkol s dostatečnou přesností. Metody učení je v zásadě možné rozdělit na učení s učitelem a učení bez učitele. Druhou zmíněnou metodu lze ještě rozdělit na učení bez dohledu (anglicky unsupervised learning) a zesílené učení (anglicky reinforcement learning) [7], [24].

Druh učení neuronových sítí lze rozdělit ještě podle toho, zda učení probíhá off-line nebo on-line. Při on-line učení jsou upravovány hodnoty vah a prahů po každém tréninkovém vzorku. Po úpravě již není vzorek potřeba a je možné ho zapomenout. To je užitečné, pokud se systémy rychle mění a s ním i vstupní vzorky, neboť pro off-line typ učení by předložené vzorky z jednoho okamžiku nemuseli být vypovídající o chování systému v pozdějších okamžicích. Výhodou také je, že není potřeba uchovávat velké množství vzorků pro učení. Opakem tohoto přístupu je pak již zmíněné off-line učení, nebo také dávkové učení (anglicky batch learning), kdy je najednou předložena celá trénovací sada. Pro jeden krok upravení vah a prahových hodnot vyžaduje tento přístup alespoň jednu tréninkovou epochu. Všechny vzorky tak musí být k dispozici po celou dobu učení [19].

1.4.1 Učení s učitelem

Také může být nazýváno jako učení s dohledem (anglicky supervised learning) a spočívá v tom, že pro žádané výstupy je k dispozici sada vstupních signálů. To znamená, že každý tréninkový vzorek sestává ze vstupu a jemu odpovídajícímu výstupu. Pokud se tedy na učitele i síť aplikuje sada vstupů, tak výstup učitele lze chápat jako požadovanou odezvu. Tato odezva je chápána jako optimální a teoreticky žádaná od neuronové sítě. Parametry sítě jsou upravovány pomocí sady trénovacích signálů a signálu chybového. Chybová odezva je dána jako rozdíl žádané hodnoty a té skutečné. Tato úprava se provádí iteračně krok po kroku, dokud síť nenapodobí učitele s nějakou žádanou statistickou přesností. Blokový diagram učení s učitelem je na Obr. 1-10. Tato forma učení je základem takzvaného učení s korekcí chyby. To znamená, že jako ukazatel výkonu systému je použita střední kvadratická chyba odchylky od žádané hodnoty. Snaha je tuto hodnotu minimalizovat s co největším poklesem, tedy s co největším gradientem. Žádané minimum pak může být lokální nebo globální. Tato metoda učení je použita i v této diplomové práci [7], [19].



Obr. 1-10 Blokový diagram učení s učitelem [7] (upraveno).

1.4.2 Učení bez učitele

Jak bylo popsáno výše, tento typ učení lze rozdělit na učení bez dohledu a na zesílené učení. Pro první zmíněný typ je hlavní rozdíl od učení s dohledem ten, že zde není žádný učitel, který dohlíží nad učením a nevyžaduje jakoukoliv znalost o požadovaných výstupech. Síť se tedy musí sama organizovat podle určitých zvláštností mezi prvky, které tvoří celý soubor vzorků, a identifikovat podmnožiny vykazující podobnost [7], [19].

Zesílené učení probíhá za stálé interakce s okolím, přičemž není potřeba sada vstupů s požadovanými výstupy. Vnitřní parametry sítě jsou upravovány na základě kvalitativních či kvantitativních informací získaných interakcí s okolím a tuto informaci využívá k hodnocení výkonu učení. Učící proces většinou probíhá formou pokus-omyl, neboť systém má zpětnou vazbu pouze v podobně toho, zda byl daný výstup uspokojivý či nikoliv [19].

1.4.3 Pravidla učení

V předchozích kapitolách byli představeny základní metody učení, které sice udávají jak a jaké vstupní informace neuronová síť dostává, avšak již neříkají nic o tom, jakým způsobem by měli být samotné prahové váhy sítě upravovány. Učení neuronových sítí se běžně provádí pomocí adaptivního postupu, známého právě jako pravidlo nebo algoritmus učení, pomocí něhož jsou váhy postupně upravovány tak, aby se v průběhu času zlepšila předem definovaná míra výkonu. Zároveň je důležité určit, v jakém okamžiku zastavit učení. Tato pravidla se rovněž liší podle použité metody učení, podle stavby sítě a podle použití [25].

Pro učení s dohledem jsou nejpoužívanější [24]:

- algoritmus nejmenších čtverců (anglicky least mean square algorithm – LMS)
- perceptronové pravidlo,
- delta pravidlo,
- algoritmus zpětného šíření.

Jako kritérium pro tyto algoritmy se používá minimalizace chybové funkce, nebo kvadratická chybová funkce. Obě jsou blíže popsány v kapitole 1.4.4. zabývající se algoritmem zpětného šíření. Pro učení bez dohledu se používají například následující algoritmy [25]:

- pravidlo Hebbian,
- Hassounovo pravidlo,
- standardní pravidlo konkurenčního učení.

1.4.4 Algoritmus zpětného šíření

Algoritmus zpětného šíření neboli anglicky back-propagation algorithm je nejvíce populární metoda pro učení pod dohledem u dopředných vícevrstvých sítí. V podstatě se jedná o zobecnění delta pravidla. Jak bylo uvedeno v předchozí kapitole, tak tento algoritmus se používá pro učení s dohledem. Máme tedy nějakou sadu vstupních a výstupních vzorků pro neuron j danou rovnicí (1-8), která je použita k trénování sítě neboli k úpravě prahových vah [1]

$$T = \{\underline{x}(k), d_j(k)\}_{n=1}^N, \quad (1-8)$$

kde

$$\vec{x}(k) = [x_1(k), x_2, \dots, x_M(k)]^T, \quad (1-9)$$

přičemž \vec{x} je vektor vstupů, M je celkový počet vstupů, d_j je požadovaný výstup a index k značí, o kolikátý trénovací vzorek se jedná. Aktivační potenciál a výstup neuronu j pro $ktý$ vzorek je pak dán rovnicemi (1-10) a (1-11) [7]:

$$v_j(k) = \sum_{k=0}^M w_{jk}(k) x_k(k) = \vec{x}^T(k) w_j(k), \quad (1-10)$$

$$y_j(k) = \varphi_j(v_j(k)), \quad (1-11)$$

kde

$$\vec{w}_j(k) = [w_{j1}(k), w_{j2}, \dots, w_{jM}(k)]^T. \quad (1-12)$$

V rovnici (1-10) je již započítán i práh aplikovaný na neuron j , čemuž odpovídá synaptická váha w_{j0} (odpovídající fixovanému vstupu $y_0 = 1$). Odpovídající chybový signál mezi požadovaným výstupem a skutečným výstupem je dán jako

$$e_j(k) = d_j(k) - y_j(k), \quad (1-13)$$

avšak častěji se pracuje s kvadratickou chybou, která je pro neuron j dána rovnicí (1-14) a nazývá se taky jako okamžitá chybová energie [7]:

$$E_j(k) = \frac{1}{2} e_j^2(k). \quad (1-14)$$

Shrnutím příspěvků chybových energií přes všechny neurony ve výstupní vrstvě pak lze dostat celkovou okamžitou chybovou energii rovnicí (1-15)

$$E(k) = \frac{1}{2} \sum_{j \in C} e_j^2(k). \quad (1-15)$$

Pokud se uvažuje funkce $E(\vec{w})$, která je spojitě diferencovatelná podle nějakého známého váhového vektoru \vec{w} , tak v optimálním případě se hledá optimální řešení dané vektorem \vec{w}^* , které splňuje podmínku, že

$$E(\vec{w}^*) \leq E(\vec{w}). \quad (1-16)$$

Cílem tedy je minimalizovat funkci $E(\vec{w})$ s ohledem na váhový vektor \vec{w} při splnění podmínky dané rovnicí (1-17) [7]

$$\nabla E(\vec{w}^*) = 0, \quad (1-17)$$

kde $\nabla E(\vec{w})$ je gradient vektor daný jako

$$\nabla E(\vec{w}) = \left[\frac{\partial E}{\partial w_1}, \frac{\partial E}{\partial w_2}, \dots, \frac{\partial E}{\partial w_M} \right]^T. \quad (1-18)$$

Pokud se tedy začne s počátečním odhadem $\vec{w}(0)$ a následně se postupně generuje sekvence $\vec{w}(1)$, $\vec{w}(2)$, ..., tak funkce $E(\vec{w})$ je každým iteračním krokem redukována podle rovnice

$$E(\vec{w}(k+1)) \leq E(\vec{w}(k)). \quad (1-19)$$

Hodnota $\vec{w}(n+1)$ je nová aktualizovaná hodnota a $\vec{w}(n)$ je hodnota stará. Algoritmus zpětného šíření chyby pracuje s metodou nejstrmějšího klesání, kdy úprava hodnot je aplikována ve směru opačném gradientu $\nabla E(\vec{w})$. Algoritmus nejstrmějšího klesání (nebo také delta pravidlo), je formálně vyjádřen jako

$$\vec{w}(k+1) = \vec{w}(k) - \eta \nabla E(\vec{w}(k)), \quad (1-20)$$

kde η je kladná hodnota, která se nazývá jako konstanta rychlosti učení či koeficient učení. Při přechodu z iterace n na $n+1$ aplikuje algoritmus následující korekci [7]

$$\Delta \vec{w}(k) = \vec{w}(k+1) - \vec{w}(k) = -\eta \nabla E(\vec{w}(k)). \quad (1-21)$$

Rychlost učení nám udává rychlost konvergence neboli jak rychle je dosaženo nějaké optimální hodnoty dané velikostí kvadratické chyby. Obecně čím menší η je, tak tím je menší změna synaptických vah pro jednu iteraci a o to jemnější bude trajektorie poklesu v prostoru prahových vah. Tento fakt ovšem snižuje rychlost učení. Na druhou stranu příliš velká rychlost učení může způsobit oscilaci, či

dokonce divergenci systémů. Je proto potřeba optimalizace. Proto se na začátku rychlost učení většinou nastaví větší a postupně se snižuje. Pokud se dále pro zjednodušení bude uvažovat úprava jedné prahové váhy $w_{ji}(n)$ mezi vstupem i a výstupním neuronem j , tak korekce $\Delta w_{ji}(n)$ aplikována algoritmem zpětného šíření chyby na tuto váhu, je podle rovnice (1-21) úměrná parciální derivaci $\partial E(n)/\partial w_{ji}(n)$. Tuto parciální derivaci lze podle pravidla o derivaci složené funkce rozepsat následovně [7], [1]

$$\frac{\delta E(k)}{\delta w_{ji}(k)} = \frac{\delta E(k)}{\delta e_j(k)} \frac{\delta e_j(k)}{\delta y_j(k)} \frac{\delta y_j(k)}{\delta v_j(k)} \frac{\delta v_j(k)}{\delta w_j(k)}. \quad (1-22)$$

Postupným vyjádřením jednotlivých parciálních derivací na pravé straně rovnice (19) se dostane sada rovnic (1-23) až (1-26).

$$\frac{\delta E(k)}{\delta e_j(k)} = e_j(k), \quad (1-23)$$

$$\frac{\delta e_j(k)}{\delta y_j(k)} = -1, \quad (1-24)$$

$$\frac{\delta y_j(k)}{\delta v_j(k)} = \varphi'_j(v_j(k)), \quad (1-25)$$

$$\frac{\delta v_j(k)}{\delta w_j(k)} = x_i(k). \quad (1-26)$$

Dosazením vztahů (1-23) až (1-26) do rovnice (1-22) získáme

$$\frac{\delta E(k)}{\delta w_{ji}(k)} = -e_j(k) \varphi'_j(v_j(k)) x_i(k). \quad (1-27)$$

Korekce $\Delta w_{ji}(k)$ aplikovaná na prahovou váhu $w_{ji}(k)$ je dána algoritmem nejstrmějšího klesání podle rovnice (1-21) a do té se dosadí rovnice (1-27). Tím je dána korekce jako

$$\Delta w_{ji}(k) = -\eta \frac{\delta E(k)}{\delta w_{ji}(k)} = \eta \delta_j(k) x_i(k), \quad (1-28)$$

kde $\delta_j(n)$ je lokální gradient definovaný jako [7]

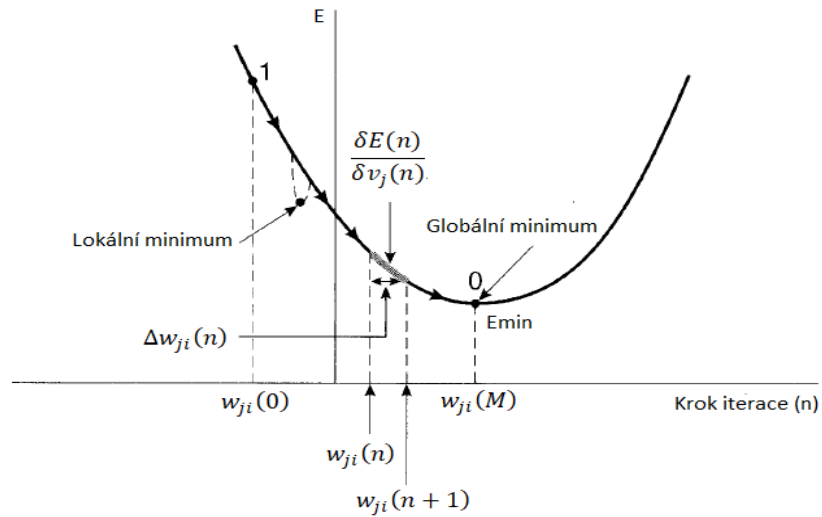
$$\delta_j(k) = \frac{\delta E(k)}{\delta v_j(k)} = e_j(k) \varphi'_j(v_j(k)). \quad (1-29)$$

Korekce chyby je tedy dána rovnicí (1-28) a hodnotu prahové váhy pro iteraci $n + 1$ lze zapsat následovně

$$w_{ji}(k + 1) = w_{ji}(k) - \eta \frac{\delta E(k)}{\delta w_{ji}(k)} + \mu \Delta w_{ji}(k - 1), \quad (1-30)$$

kde μ je momentový faktor. Ten se stejně jako rychlost učení nastaví na začátku a slouží k tomu, aby se učení nezastavilo na lokálním minimu, namísto minima

globálního. Na Obr. 1-11 je znázorněna minimalizace kvadratické chyby pomocí klesajícího gradientu pro váhu w_{ji} , přičemž lokální minimum je znázorněno čárkovanou křivkou [7], [1].



Obr. 1-11 Minimalizace kvadratické chyby pomocí metody poklesu gradientu [1] (upraveno).

KAPITOLA 2: ASYNHRONNÍ MOTOR

Asynchronní motor (zkráceně AM) je typ motoru na střídavý elektrický proud a jedná se o typ elektrického motoru na světě. Jeho historie sahá až do konce 19. století a za jeho prvotním výzkumem stojí především Galileo Ferraris a Nikola Tesla. Ferraris v roce 1888 představil jako první princip střídavého AM, který ale dosahoval nízké účinnosti. Ten stejný rok byly Teslovy schváleny patenty asynchronního motoru, které vykazovali vyšší účinnost, než tomu bylo u motoru Ferrarise. Tesla navíc na rozdíl od Ferrarise věřil v jejich praktické využití. Patenty následně koupil George Westinghouse, který pracoval na jejich zdokonalení pro praktické využití. To se podařilo v roce 1892, kdy byl tento typ motoru poprvé použit v průmyslu. Od té doby prošel asynchronní motor velikým pokrokem a v dnešní době se jedná o hojně využívaný typ motoru [26], [27].

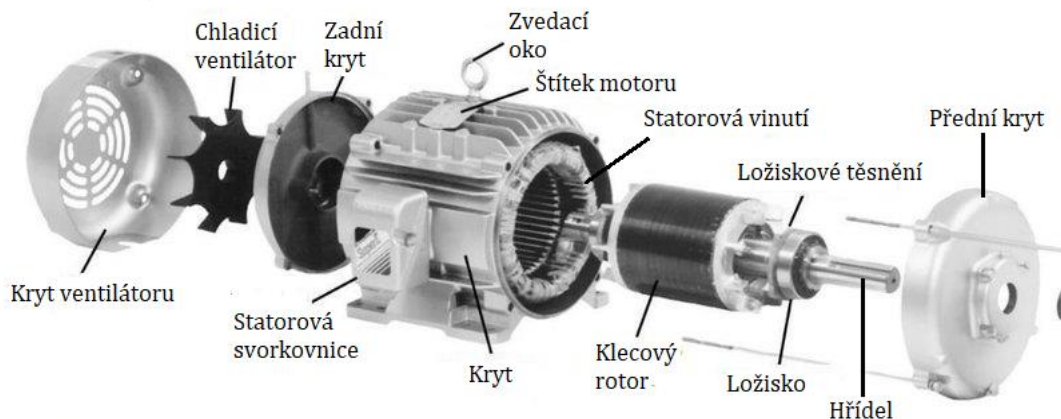
Využití naleznou v celé řadě aplikací od domácích spotřebičů, ručního nářadí, přes ventilátory, kompresory, pumpy až po trakční pohony. V těch největších aplikacích se uplatní například v textilních strojích. Hlavními výhodami tohoto typu elektrického motoru je jeho jednoduchá konstrukce, čímž je dána i snadná výroba, nízká pořizovací cena a vysoká provozní spolehlivost. Mají samozřejmě i nevýhody a oproti synchronním motorům s permanentními magnety nebo bezkartáčovým motorům, mají menší hustotu výkonu a větší ztráty v rotoru, čímž může docházet k jeho zahřívání. Mezi jednu z hlavních nevýhod dříve patřilo i obtížné řízení rychlosti, ta však byla eliminována s rozvojem měničů výkonové elektroniky [28]–[30].

V následujících kapitolách jsou stručně probrány konstrukce, princip činnosti, matematický model a způsoby řízení třífázového AM.

2.1 Konstrukce a princip činnosti

Asynchronní motor se podobně jako ostatní elektrické motory skládá ze statoru, a z rotoru, který se otáčí. Stator je složen z tenkých plechů oceli pro elektrotechniku z důvodu snížení ztrát především vířivými proudy. Na vnitřní straně plechů jsou uspořádány drážky, ve kterých je rovnoměrně rozloženo vinutí s izolací po celé délce statoru. Vinutí je zpravidla třífázové, ale může být i jedno, nebo dvou fázové, či naopak více fázové. Začátky a konce fází jsou nakonec vyvedeny na svorkovnici [30], [31].

Rotor je od statoru oddělen tenkou vzduchovou mezerou a může být buďto klecový, nebo kroužkový. Kroužkový typ je, stejně jako stator, složen z tenkých plechů, které však mají drážkování na vnější straně, kde je pak uloženo vinutí. To je vyvedeno v zapojení do hvězdy na kroužky, které jsou umístěny na hřídeli. Přes ně je možné pomocí sběracích kartáčů připojit do obvodu rotoru rozběhový odpor. Klecový typ se skládá z tyčí z mědi nebo hliníku, které jsou v čelech spojeny do kruhu a tím jsou zkratovány. Prostor mezi jednotlivými tyčemi je opět vyplněn tenkými plechy pro elektrotechniku. Pro jednoduchost a robustnost se klecová varianta používá výrazně častěji. V obou případech bývá motor ještě osazen ventilátorem umístěným přímo na hřídeli a celý stroj je chráněn krytem. U větších motorů může být ventilátor napájen z externího zdroje, aby se dosáhlo požadovaného chlazení i při nižších otáčkách. Konstrukce asynchronního motoru s klecovým rotorem je na Obr. 2-1 [31], [32].



Obr. 2-1 Konstrukce asynchronního motoru s klecovým rotorem [33] (upraveno).

Při uvažování nejčastějšího třífázového vinutí, kde všechny cívky mají stejné uspořádání, mají stejný počet závitů a jsou vůči sobě natočeny o úhel, který odpovídá fázovému posuvu třífázového souměrného napájecího harmonického napětí, tedy o $2\pi/3$ radiánů, je princip následující. Po připojení na toto napájecí napětí protékají vinutími proudy, které dohromady vytvoří točivé magnetické pole, které se otáčí synchronní elektrickou úhlovou rychlostí ω_1 podle následující rovnice [30]

$$\omega_1 = 2\pi f, \quad (2-1)$$

kde f je frekvence napájecího zdroje. V rotoru se díky tomu indukují napětí, které vyvolávají proudy. Ty interagují s polem statoru a vzniká moment. Ten otáčí rotorem ve směru otáčení pole statoru a rotor se tak snaží zmenšit napětí a proudy, které se v něm indukují. Při tomto chodu pracuje jako motor a magnetické pole rotoru se vlastně snaží dohnat pole statoru. K tomu však nikdy nedojde, neboť v tu chvíli by byl moment nulový a zároveň jsou na rotoru i při chodu naprázdno vždy alespoň minimální ztráty dané třením. Motor tak pracuje s trvalým skluzem s . Ten je dán rovnicí (2-2) a vyjadřuje jakou rychlost má rotor vůči statoru [31], [34]

$$s = \frac{\omega_1 - \omega}{\omega_1}, \quad (2-2)$$

přičemž ω je úhlová rychlost otáčení rotoru. Tuto rychlost lze získat z mechanické úhlové rychlosti Ω jako [30].

$$\omega = p_p \Omega, \quad (2-3)$$

kde p_p je počet pólových dvojic a Ω je mechanická úhlová rychlost. Často se uvádí ještě takzvaná elektrická skluzová úhlová rychlost ω_s , což je úhlová rychlost rotoru vůči poli a lze ji vyjádřit jako [30]

$$\omega_s = \omega_1 - \omega = s\omega_1. \quad (2-4)$$

Z předchozí rovnice lze dostat i jiné vyjádření pro elektrickou úhlovou rychlost rotoru ve tvaru

$$\omega = (1 - s)\omega_1. \quad (2-5)$$

Převod elektrických veličin na mechanické podle rovnice (2-5) lze aplikovat i na ostatní výše zmíněné vztahy.

2.2 Matematický model

Při odvozování základních rovnic a modelů asynchronního stroje se uvažuje kroužkový motor s třífázovým vinutím na statoru a rotoru. Dále jsou přijata některé zjednodušující předpoklady pro usnadnění výpočtu a odvození [35]:

- napájecí soustava je třífázová souměrná a všechna napětí jsou harmonická,
- vinutí statoru jsou sinusově rozložena po celém obvodu vzduchové mezery a jednotlivá vinutí jsou vůči sobě natočena o 120° ,
- vzduchová mezera mezi státorem a rotorem se uvažuje konstantní,
- odpory a indukčnosti jednotlivých fází statoru nebo rotoru jsou shodné,
- ztráty v železe jsou zanedbatelné,
- motor pracuje v lineární části magnetizační křivky, kde již nedochází k sycení magnetického obvodu.

Za předpokladu, že vinutí statoru ani rotoru nemá vyvedený uzel platí

$$i_a + i_b + i_c = 0, \quad (2-6)$$

$$i_A + i_B + i_C = 0, \quad (2-7)$$

kde indexy malými písmeny označují stator a velkými písmeny rotor. Postupnými úpravami a pomocí takzvané Clarkovy a Parkovy transformace je pak možné dospět k soustavě rovnic (2-12) popisující asynchronní stroj ve složkách v obecném souřadnicovém systému otáčejícím se rychlostí ω_k vůči statoru a rychlostí otáčení $\omega_k - \omega$ vůči rotoru. Pro úplnost je nejdříve stručně popsán význam použití zmíněných transformací [35].

1. **Clarkové transformace.** Ta převádí třífázovou soustavu abc do dvoufázové ortogonální soustavy $\alpha\beta$, přičemž osa a je totožná s osou α . Transformační matice je obecně dána následovně [35], [36]

$$\begin{pmatrix} x_\alpha \\ x_\beta \end{pmatrix} = K \begin{pmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{pmatrix} \begin{pmatrix} x_a \\ x_b \\ x_c \end{pmatrix}, \quad (2-8)$$

kde K je transformační konstanta, která může být v zásadě libovolná, avšak pro účely regulace střídavých elektrických pohonů se volí $K = 2/3$. Tato hodnota je použita i v této práci. Potom pokud platí podmínka podle rovnice (2-6) a (2-7), tak pro transformaci proudů lze přímo psát, že $i_\alpha = i_a$. Zpětná transformační matice, pro inverzní Clarkovu transformaci s $K = 2/3$ potom je [35]

$$\begin{pmatrix} x_a \\ x_b \\ x_c \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ -\frac{1}{2} & \frac{\sqrt{3}}{2} \\ -\frac{1}{2} & -\frac{\sqrt{3}}{2} \end{pmatrix} \begin{pmatrix} x_\alpha \\ x_\beta \end{pmatrix}. \quad (2-9)$$

2. **Parkova transformace.** Tato transformace slouží k převodu do soustavy xy , která rotuje libovolnou rychlostí vůči původnímu systému os $\alpha\beta$, neboť tento systém je pevně svázán se státorem (v případě rotorového vinutí s rotorem).

Parkova transformační matice ze systému $\alpha\beta$ do obecného xy , je pak následující [35]

$$\begin{pmatrix} x_x \\ x_y \end{pmatrix} = \begin{pmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{pmatrix} \begin{pmatrix} x_\alpha \\ x_\beta \end{pmatrix}, \quad (2-10)$$

kde θ je okamžitý úhel mezi osami α a x . Pro zpětnou Parkovu transformaci pak lze psát [36]

$$\begin{pmatrix} x_\alpha \\ x_\beta \end{pmatrix} = \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix} \begin{pmatrix} x_x \\ x_y \end{pmatrix}, \quad (2-11)$$

Aplikováním transformací na napěťové a tokové rovnice sestavené pro třífázový systém na statoru a rotoru získáváme základní rovnice AM ve složkách ve tvaru

$$\begin{aligned} u_{1x} &= R_1 i_{1x} + \frac{d\psi_{1x}}{dt} - \omega_k \psi_{1y} \\ u_{1y} &= R_1 i_{1y} + \frac{d\psi_{1y}}{dt} + \omega_k \psi_{1x} \\ u_{2x} &= R_2 i_{2x} + \frac{d\psi_{2x}}{dt} - (\omega_k - \omega) \psi_{2y} \\ u_{2y} &= R_2 i_{2y} + \frac{d\psi_{2y}}{dt} + (\omega_k - \omega) \psi_{2x} \\ \psi_{1x} &= L_1 i_{1x} + L_m i_{2x} \\ \psi_{1y} &= L_1 i_{1y} + L_m i_{2y} \\ \psi_{2x} &= L_2 i_{2x} + L_m i_{1x} \\ \psi_{2y} &= L_2 i_{2y} + L_m i_{1y}, \end{aligned} \quad (2-12)$$

kde u_1 je napětí na statoru, u_2 je napětí na rotoru, R_1 respektive R_2 je statorový, respektive rotorový odpor, L_1 respektive L_2 je statorová, respektive rotorová indukčnost, L_m je magnetizační indukčnost a ψ_1 , respektive ψ_2 je statorový, respektive rotorový magnetický spřažený tok. Takovýto obecný souřadnicový systém se v elektrických pohonech následně převádí do některého z následujících dvou souřadnicových systému, podle toho, jakého zjednodušení chceme dosáhnout a které veličiny sledovat [37], [35].

- **Souřadnicový systém svázaný se statorem.** Tento systém se značí $\alpha\beta$ a platí, že $\omega_k = 0$. Soustava rovnic (2-12) tak přejde do tvaru [35]

$$\begin{aligned} u_{1\alpha} &= R_1 i_{1\alpha} + \frac{d\psi_{1\alpha}}{dt} \\ u_{1\beta} &= R_1 i_{1\beta} + \frac{d\psi_{1\beta}}{dt} \\ u_{2\alpha} &= R_2 i_{2\alpha} + \frac{d\psi_{2\alpha}}{dt} + \omega \psi_{2\beta} \\ u_{2\beta} &= R_2 i_{2\beta} + \frac{d\psi_{2\beta}}{dt} - \omega \psi_{2\alpha} \\ \psi_{1\alpha} &= L_1 i_{1\alpha} + L_m i_{2\alpha} \\ \psi_{1\beta} &= L_1 i_{1\beta} + L_m i_{2\beta} \\ \psi_{2\alpha} &= L_2 i_{2\alpha} + L_m i_{1\alpha} \\ \psi_{2\beta} &= L_2 i_{2\beta} + L_m i_{1\beta}. \end{aligned} \quad (2-13)$$

- **Souřadnicový systém rotující synchronní rychlostí magnetického pole.** Tento systém se značí dq a platí, že $\omega_k = \omega_1$. Rovnice (2-13) tak přejde do tvaru [35]

$$\begin{aligned}
 u_{1d} &= R_1 i_{1d} + \frac{d\psi_{1d}}{dt} - \omega_1 \psi_{1q} \\
 u_{1q} &= R_1 i_{1q} + \frac{d\psi_{1q}}{dt} + \omega_1 \psi_{1d} \\
 u_{2d} &= R_2 i_{2d} + \frac{d\psi_{2d}}{dt} - (\omega_1 - \omega) \psi_{2q} \\
 u_{2q} &= R_2 i_{2q} + \frac{d\psi_{2q}}{dt} + (\omega_1 - \omega) \psi_{2d} \\
 \psi_{1d} &= L_1 i_{1d} + L_m i_{2d} \\
 \psi_{1q} &= L_1 i_{1q} + L_m i_{2q} \\
 \psi_{2d} &= L_2 i_{2d} + L_m i_{1d} \\
 \psi_{2q} &= L_2 i_{2q} + L_m i_{1q}.
 \end{aligned} \tag{2-14}$$

KAPITOLA 3: VEKTOROVÉ ŘÍZENÍ ASNCHRONNÍHO MOTORU

V této kapitole je stručně probrána jedna ze dvou nejpoužívanějších metod regulace pro pohony s asynchronními motory a tou je takzvaná vektorově orientovaná regulace. Cílem této metody je efektivně řídit zvláště jak moment motoru, tak i magnetický tok, podobně jako je tomu u stejnosměrných cize buzených motorů. Tímto způsobem lze eliminovat nedostatky tzv. skalárního řízení, jako je třeba přesnost a rychlost regulace. Hlavní uplatnění našlo především v náročnějších aplikacích (např. trakční vozidla, servopohony apod.), přičemž ho lze aplikovat jak na asynchronní, tak i synchronní motory [1].

3.1 Princip vektorové regulace

Jak již bylo řečeno, tak v ideálním případě se asynchronní motor při vektorové regulaci chová jako stejnosměrný cize buzený motor. Tato analogie vychází z upravené rovnice (3-1) pro moment stejnosměrného stroje se zanedbáním reakce kotvy a saturace magnetického obvodu [1]

$$M_e = k' i_a i_f, \quad (3-1)$$

kde i_a je proud kotvy, i_f je proud buzení a k' je konstanta stroje. Z konstrukce stejnosměrného stroje je pak budicí tok ψ_f vyvolaný proudem i_f kolmý na rotorový tok ψ_a vyvolaný proudem i_a . Tyto prostorové vektory jsou stacionární v prostoru, ortogonální a oddělené již z podstaty. Moment pak lze řídit proudem i_a bez ovlivnění toku ψ_f a ten naopak lze řídit proudem i_f bez ovlivnění momentu [1].

Této podoby lze dosáhnout uvažováním základních rovnic asynchronního motoru v souřadnicovém systému dq rotujícím synchronní rychlostí. Sinusové proměnné motoru pak vystupují jako stejnosměrné veličiny v ustáleném stavu. Dalšími faktory pro řízení asynchronního motoru jako motoru stejnosměrného jsou [1], [38]:

- oddělené řízení magnetického toku a momentu,
- při konstantním magnetickém toku je moment lineárně úměrný momentotvorné složce proudu a
- moment by měl být úměrný skluzu.

Splnění podmínky lineární závislosti momentu na momentotvorné složce proudu je dosaženo položením osy d souřadnicového systému do směru vektoru magnetického toku. Zbylé podmínky jsou splněny nejlépe, pokud je vektorové řízení orientované na rotorový magnetický tok a platí tedy, že $|\vec{\psi}_2| = \psi_{2d}$ a $\psi_{2q} = 0$. Navíc v ustáleném stavu platí, že vektor rotorového proudu je kolmý na vektor rotorového toku a platí tedy, že $i_{2d} = 0$. Za těchto podmínek lze ze soustavy rovnic (2-14) vyjádřit rotorový tok rovnicí (3-2) a moment pomocí rovnice (3-3) [38], [39].

$$\psi_2 = L_m i_{1d}, \quad (3-2)$$

$$m = \frac{3}{2} p_p \frac{L_m}{L_2} \psi_2 i_{1q}. \quad (3-3)$$

S využitím (3-2) a zavedením konstanty $k = (3/2) p_p (L_m^2 / L_2)$ lze (3-3) zjednodušit na

$$m = k i_{1d} i_{1q}. \quad (3-4)$$

Vztah (3-4) dává analogii k stejnosměrnému stroji, z čehož i plyne, že proudem i_{1d} lze přímo řídit tok, a proto se nazývá **tokotvorná složka** satorového proudu a proudem i_{1q} lze přímo řídit moment a jedná se tak o **momentotvornou složku** satorového proudu [1].

3.2 Matematické modely pro vektorové řízení asynchronního motoru

Jelikož je pro vektorové řízení nutné, aby byli regulované veličiny v souřadnicovém systému dq , který rotuje synchronní rychlostí, tak je nutné potřebné veličiny do toho souřadnicového systému transformovat. Nejdříve se provede Clarková transformace z třífázového souřadnicového systému do systému $\alpha\beta$ pomocí rovnice (2-8). Poté je nutné tento souřadnicový systém transformovat do systému dq rotujícím synchronní rychlostí podle Parkovy transformace dané rovnicí (2-10). Jak je ale z tohoto vztahu zjevné, tak je potřeba znát okamžitý transformační úhel θ mezi osou α a d . Dále je nutné v každém okamžiku znát velikost vektoru rotorového toku, a to jak kvůli regulaci, tak v některých případech i k výpočtu transformačního úhlu. Pro zjišťování těchto hodnot se v zásadě používají dva modely [1]

- I-n model (anglicky current model),
- U-I model (anglicky voltage model),

a dvě metody:

- přímá metoda (anglicky direct, nebo také feedback vector control),
- nepřímá metoda (anglicky indirect, nebo také feedforward vector control).

Co se týče modelů, tak ty lze odvodit z obecné soustavy rovnic (2-12), tu lze pro motor s kotvou nakrátko přepsat ve vektorovém tvaru jako

$$\begin{aligned}\vec{u}_1 &= R_1 \vec{i}_1 + \frac{d\vec{\psi}_1}{dt} + j\omega_k \vec{\psi}_1 \\ 0 &= R_2 \vec{i}_2 + \frac{d\vec{\psi}_2}{dt} + j(\omega_k - \omega) \vec{\psi}_2 \\ \vec{\psi}_1 &= L_1 \vec{i}_1 + L_m \vec{i}_2 \\ \vec{\psi}_2 &= L_2 \vec{i}_2 + L_m \vec{i}_1.\end{aligned}\tag{3-5}$$

3.2.1 Proudový I-n model

Jak napovídá název, tak „I-n“ označuje, že výsledný vektor rotorového magnetického toku je získán pomocí vektoru satorového proudu a otáček motoru, nebo přesněji jeho elektrické úhlové rychlosti. Pro přímou metodu se proudový model uvádí v souřadnicovém systému $\alpha\beta$ svázaným se statorem (platí tedy $\omega_k = 0$) a je dán následujícími rovnicemi [1]

$$\begin{aligned}\frac{d\psi_{2\alpha}}{dt} &= \frac{L_m}{T_r} i_{1\alpha} - \omega \psi_{2\beta} - \frac{1}{T_r} \psi_{2\alpha} \\ \frac{d\psi_{2\beta}}{dt} &= \frac{L_m}{T_r} i_{1\beta} + \omega \psi_{2\alpha} - \frac{1}{T_r} \psi_{2\beta},\end{aligned}\tag{3-6}$$

kde $T_r = L_2/R_2$ je rotorová časová konstanta. Jak bylo řečeno, tak výhodou je použitelnost v celém rozsahu otáček (především i v oblasti nízkých otáček). Zároveň z rovnic (3-6) je vidět závislost na elektrických parametrech motoru. Především zvyšující se elektrický odpor rotoru vlivem teploty, případně skin efektu, má dominantní vliv na výpočet rotorového toku, a tedy i na kvalitu regulace. Kompenzace je zároveň složitá, neboť rotorový odpor se nedá přímo měřit [1].

U nepřímého modelu se I-n model používá v souřadnicovém systému rotujícím synchronní rychlostí, platí tedy $\omega_k = \omega_1$ a zároveň lze provést nahrazení $\omega_s = \omega_1 - \omega$. Potom lze úpravami dostat následující rovnice pro výpočet vektoru rotorového magnetického toku a skluzové rychlosti [1]

$$\frac{d\psi_{2d}}{dt} = \frac{L_m}{T_r} i_{1d} - \frac{1}{T_r} \psi_{2d}, \quad (3-7)$$

$$\psi_{2d} \omega_s = \frac{L_m}{T_r} i_{1q}. \quad (3-8)$$

3.2.2 Napěťový U-I model

Tento model používá jako vstupní veličiny vektor statorového proudu a vektor statorového napětí. Výstupem je stejně jako u proudového modelu vektor rotorového toku. Model se stejně jako předchozí proudový model uvažuje v souřadnicovém systému $\alpha\beta$ svázaného se statorem a je dán následujícími rovnicemi [1]

$$\psi_{1\alpha} = \int (u_{1\alpha} - R_1 i_{1\alpha}) dt,$$

$$\psi_{1\beta} = \int (u_{1\beta} - R_1 i_{1\beta}) dt,$$

(3-9)

$$\psi_{2\alpha} = \frac{L_2}{L_m} (\psi_{1\alpha} - \sigma L_1 i_{1\alpha}),$$

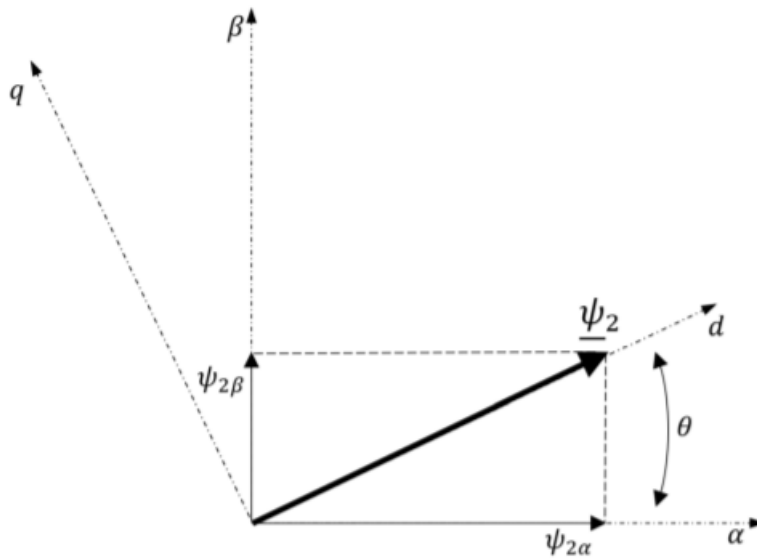
$$\psi_{2\beta} = \frac{L_2}{L_m} (\psi_{1\beta} - \sigma L_1 i_{1\beta}),$$

kde $\sigma = 1 - L_m^2/(L_1 L_2)$ je takzvaný rozptyl. Problémy tohoto modelu jsou především v tom, že běžná jednoduchá integrace je obtížná vlivem akumulace DC offsetu na výstupu integrátoru. Dalším problémem je vliv elektrických parametrů při nižších otáčkách, a to především statorového odporu R_1 , který se mění s teplotou. Tento problém se však dá kompenzovat. V průmyslových aplikacích je však většinou vyžadováno, aby pohony s vektorovou regulací operovaly z nulových otáček a v takovém případě je použití tohoto typu modelu složitější [1].

3.2.3 Transformační úhel

Modely v předchozích kapitolách slouží k výpočtu vektoru rotorového magnetického rotorového toku. Jak již ale bylo zmíněno, tak je potřeba znát ještě okamžitý úhel θ pro Parkovu transformaci. Z Obr. 3-1 je vidět, že pro přímou metodu lze vyjít přímo z vypočtených složek rotorového toku v systému $\alpha\beta$ a s použitím goniometrických funkcí podle rovnice(3-10) [1].

$$\theta = \tan^{-1} \frac{\psi_{2\beta}}{\psi_{2\alpha}}. \quad (3-10)$$



Obr. 3-1 Transformační úhel mezi souřadnicovým systémem $\alpha\beta$ a dq [40].

U nepřímé metody se transformační úhel počítá pomocí integrace synchronní úhlové rychlosti rotoru podle

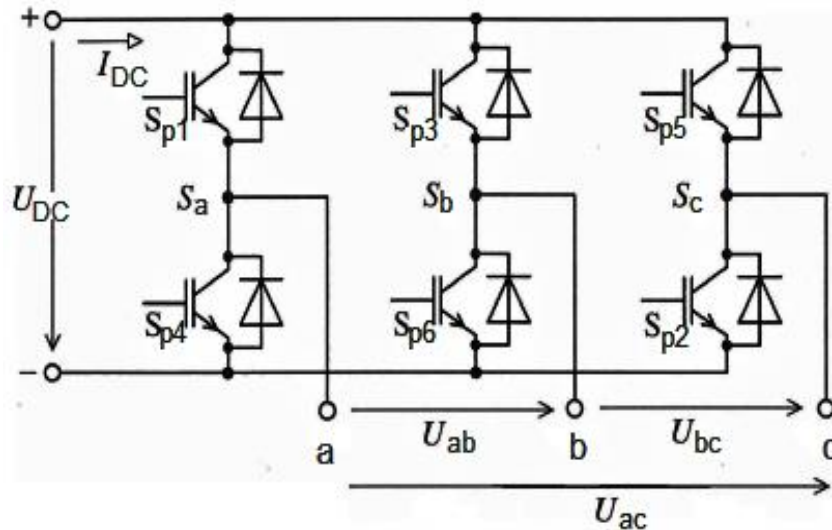
$$\theta = \int \omega_1 dt = \int (\omega_s + \omega) dt. \quad (3-11)$$

3.3 Frekvenční měnič a jeho řízení

3.3.1 Třífázový napěťový střídač

Třífázový můstkový střídač je hojně využíván k napájení střídavých motorů. Jelikož asynchronní motor není schopen sám dodávat napětí pro zajištění komutace polovodičových součástek, tak je nutné použít součástky, které lze vypínat. Většinou se proto využívají GTO, IGBT, MOSFET nebo IGCT tranzistory. Obvodové schéma třífázového napěťového střídače s IGBT tranzistory je na Obr. 3-2, přičemž tranzistory jsou ještě osazeny antiparalelně zapojenými diodami [35].

Obecně se tento typ střídače skládá ze tří paralelních větví, přičemž v každé větvi jsou sériově zapojeny dva spínače (vypínatelná součástka s antiparalelní diodou) označeny jako S_1 až S_6 . Tyto spínače jsou poté buďto spínány nebo vypínány pulzy z vnějšího zdroje řídicích signálů. Jednotlivé větve jsou napájeny napětím U_{DC} ze stejnosměrného napěťového meziobvodu. Tento meziobvod bývá ještě doplněn o vyhlazovací kondenzátor. Vývody označené jako S_a , S_b a S_c mezi každými dvěma spínači v jedné větvi, jsou vyvedeny na jednotlivé fáze třífázového vinutí statoru [35], [41].



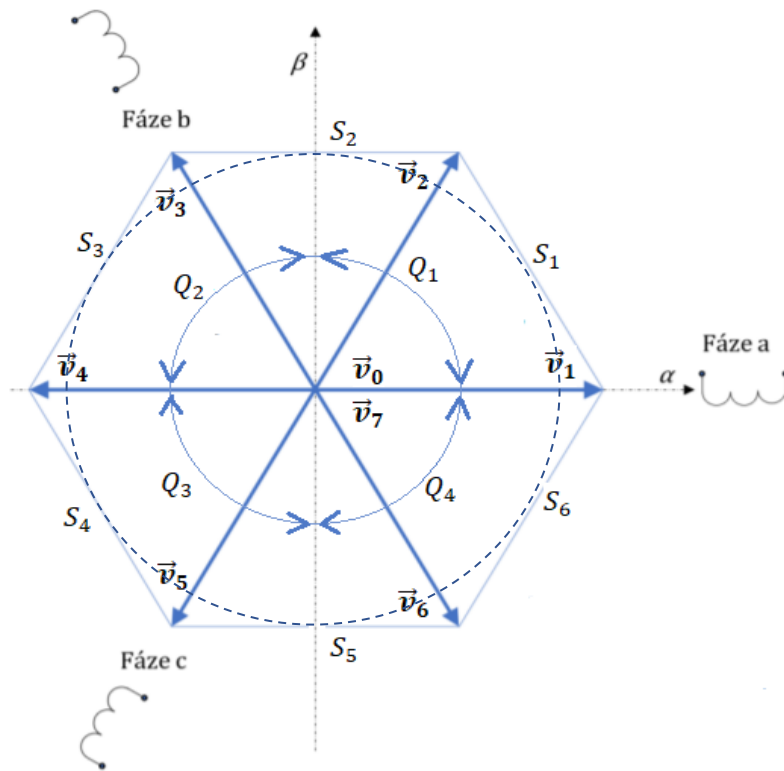
Obr. 3-2 Schéma dvouúrovňového střídače napětí [35] (upraveno).

U tohoto typu řízeného střídače je možné dosáhnout celkem osmi různých stavů sepnutí jednotlivých tranzistorů, přičemž nikdy nesmí dojít k sepnutí dvou spínačů v jedné větvi. Došlo by tím ke zkratování stejnosměrného obvodu, což by mohlo vést až ke zničení tranzistorů. Řídicím obvodem musí být tedy zajištěno, aby k takovému stavu nedošlo [30], [38], [41].

Povolené stavy sepnutí jsou uvedeny v Tab. 3-1 a každá tato kombinace vytváří prostorový napěťový vektor \vec{v}_0 až \vec{v}_7 . Pokud se tyto vektory vyjádří v souřadnicovém systému $\alpha\beta$ svázaného se statorem, tak reálná a imaginární osa rozdělují tuto rovinu na čtyři kvadranty Q_1 až Q_4 a napěťové vektory jí dělí na šest sektorů S_1 až S_6 , jak je znázorněno na Obr. 3-3. Přičemž vektor \vec{v}_0 a \vec{v}_7 odpovídá sepnutí všech dolních, respektive všech horních tranzistorů a jedná se o takzvané nulové vektory, jejichž velikost je nulová. Sepnutí tranzistorů v horní řadě je dáno logickou 1 a vinutí je tak připojeno na kladnou polaritu napětí, naopak sepnutí tranzistorů v dolní řadě je charakterizováno logickou 0 a na vinutí se objeví záporná polarita napětí [38], [41].

Tab. 3-1 Základní vektory napětí a jejich příslušné spínací kombinace

	\vec{v}_0	\vec{v}_1	\vec{v}_2	\vec{v}_3	\vec{v}_4	\vec{v}_5	\vec{v}_6	\vec{v}_7
a	0	1	1	0	0	0	1	1
b	0	0	1	1	1	0	0	1
c	0	0	0	0	1	1	1	1



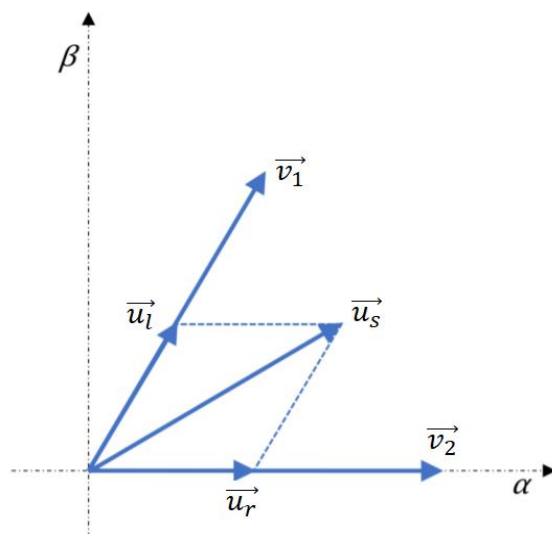
Obr. 3-3 Vektorový prostor daný základními vektory napětí střídače a jeho rozdělení na kvadranty (Q) a sektory (S) [40] (upraveno).

3.3.2 Pulzně-šířkové řízení modulací prostorového vektoru

Pulzně-šířkové řízení (zkráceně PWM) je jedna z metod, jak spínat tranzistory střídače. Její hlavní výhodou je možnost řídit jak amplitudu výstupního napětí, tak i frekvenci. Oproti jedné z dalších metod, takzvanému obdélníkovému řízení, probíhá spínání tranzistorů mnohonásobně častěji, přičemž se mění šířka jednotlivých pulzů a tím je možné měnit střední hodnotu napětí za jednu modulační periodu. Z toho plyne i název „pulzně-šířková.“ Vyšší spínací frekvence s sebou ovšem přináší vyšší spínací ztráty. Výhody však jsou například malé zkreslení výstupního napětí a proudu a tím i nižší obsah vyšších harmonických, lepší účinnost a vyšší výkon [1], [35], [42], [41].

U zmíněné obdélníkové metody je výstupní napětí dáno pouze jedním ze šesti aktivních a dvou pasivních stavů, které odpovídají jednomu ze základních prostorových vektorů. Při modulaci prostorového vektoru je ale teoreticky možné pomocí vhodného spínání vytvořit jakýkoliv napěťový vektor omezený hexagonem, který spojuje vrcholy vektorů na Obr. 3-3. Při uvažování požadovaného vektoru napětí \vec{u}_s , který se nachází například v sektoru S_1 , je podle Obr. 3-4 možné tento vektor získat vektorovým součtem dvou okrajových vektorů \vec{u}_r a \vec{u}_l . Dolní indexy r a l označují pravý, respektive levý vektor. Ty mají stejný směr jako základní vektory \vec{v}_1 a \vec{v}_2 , ale mohou mít jinou velikost. Maximální velikost vektoru \vec{u}_s je pak stejná jako velikost základních prostorových vektorů napětí a to [1], [38], [41]

$$|\vec{u}_{smax}| = |\vec{v}_1| = |\vec{v}_2| = |\vec{v}_3| = |\vec{v}_4| = |\vec{v}_5| = |\vec{v}_6| = \frac{2}{3} U_{DC}. \quad (3-12)$$



Obr. 3-4 Příklad realizace žádaného vektoru napětí pomocí dvou krajních vektorů pro sektor S_1 [40] (upraveno).

V některých případech v praxi, a i v této práci, je z důvodu omezení vyšších harmonických ve výstupním napětí omezen žádaný vektor napětí na $1/\sqrt{3}U_{DC}$. Na Obr. 3-3 je toto omezení znázorněno čárkovanou kružnicí vepsanou do hexagonu. Maximální velikost žádané napětí pak je dáno jako [38]

$$|\vec{u}_{smax}| = \frac{1}{\sqrt{3}} U_{DC}. \quad (3-13)$$

Pro získání výsledného žádaného vektoru je třeba určit, jak dlouho z doby jedné periody T_p má být sepnutý pravý či levý vektor. Tato spínací doba je dána jako [38]

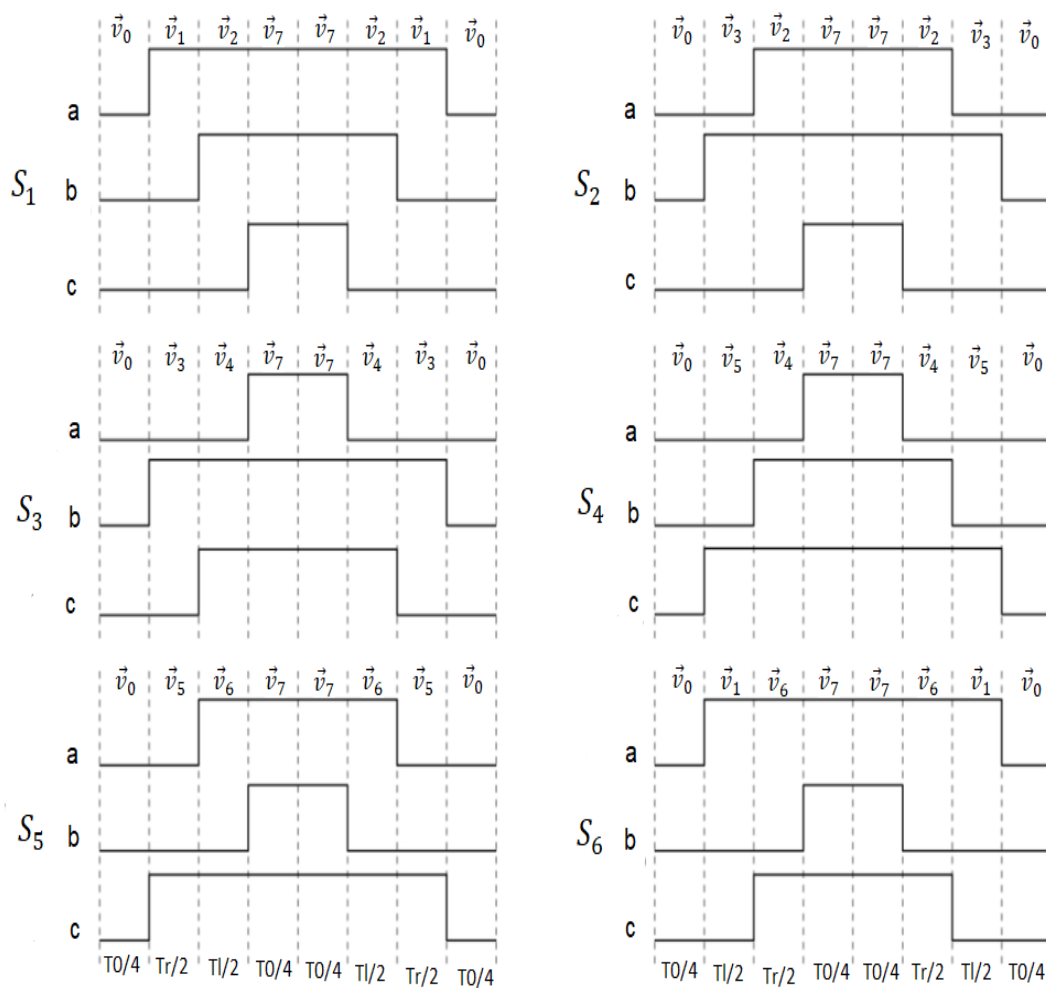
$$T_r = T_p \frac{|\vec{u}_r|}{|\vec{u}_{smax}|} \quad (3-14)$$

$$T_l = T_p \frac{|\vec{u}_l|}{|\vec{u}_{smax}|}.$$

Zbytek času, tedy $T_p - (T_r + T_l)$, kdy není sepnut pravý ani levý vektor, je sepnut jeden z nulových vektorů, tedy \vec{v}_0 nebo \vec{v}_7 . Žádaný napěťový vektor \vec{u}_s je tak dán rovnicí jako [38]

$$\begin{aligned} \vec{u}_s &= \vec{u}_r + \vec{u}_l + \vec{v}_0 \text{ (nebo } \vec{v}_7) = \\ &= \frac{T_r}{T_p} \vec{v}_1 + \frac{T_l}{T_p} \vec{v}_2 + \frac{T_p - (T_r + T_l)}{T_p} \vec{v}_0 \text{ (nebo } \vec{v}_7) \end{aligned} \quad (3-15)$$

Pro ostatní sektory se postupuje obdobně a mění se pouze okrajové vektory. Jsou různé vzory pro spínání jednotlivých tranzistorů podle toho, v jakém sektoru se žádaný vektor napětí nachází. Spínání je možné realizovat například podle vzorů na Obr. 3-5 [38].



Obr. 3-5 Vzory pro spínání tranzistorů v jednotlivých sektorech [43] (upraveno).

Z rovnice (3-14) plyne, že pro určení spínacích časů je potřeba znát velikost okrajových vektorů \vec{u}_r a \vec{u}_l . Existuje několik možností, jak určit okrajové vektory a konkrétní kvadrant a sektor. Jedna z možností je zavedení členů k, l a m , podle rovnice (3-16). Ty udávají velikost pravého či levého vektoru v závislosti na aktuálním sektoru podle Tab. 3-2.

$$\begin{aligned}
 k &= |u_{1\alpha}| + \frac{1}{\sqrt{3}} |u_{1\beta}| \\
 l &= |u_{1\alpha}| - \frac{1}{\sqrt{3}} |u_{1\beta}| \\
 m &= \frac{2}{\sqrt{3}} |u_{1\beta}|
 \end{aligned}
 \tag{3-16}$$

Tab. 3-2 Výpočet krajních vektorů napětí pro určitý kvadrant a sektor.

Sektor	Kvadrant	$ \vec{u}_r $	$ \vec{u}_l $
S_1	Q_1	l	m
S_2	Q_1	k	$-l$
	Q_2	$-l$	k
S_3	Q_2	m	l
S_4	Q_3	l	m
S_5	Q_3	k	$-l$
	Q_4	$-l$	k
S_6	Q_4	m	l

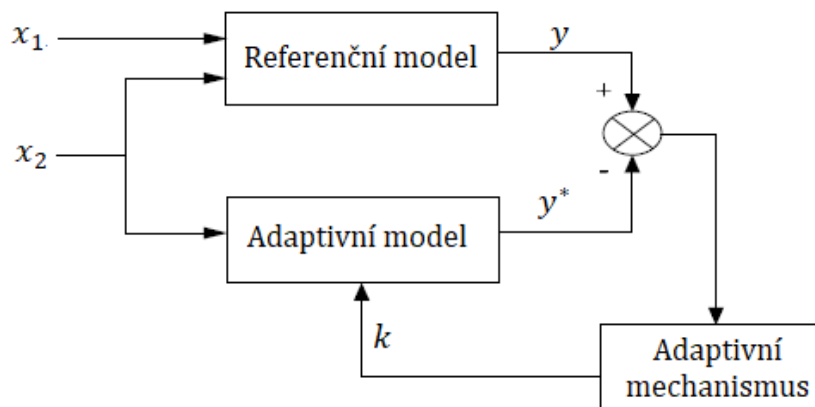
Nakonec je potřeba ještě určit daný sektor a kvadrant pro vybrání správné spínací sekvence podle předchozího obrázku. Podle znamének složek $u_{1\alpha}, u_{1\beta}$ je možné určit daný kvadrant a podle znaménka členu l lze určit sektor, neboť jeho znaménko se mění s každým průchodem sektoru. Přehledně je určení kvadrantu a sektoru znázorněno v následující tabulce [38].

Tab. 3-3 Stanovení aktuálního kvadrantu a sektoru pomocí složek $u_{1\beta}, u_{1\alpha}$ a členu l .

	$u_{1\beta} > 0$				$u_{1\beta} < 0$			
	$u_{1\alpha} > 0$		$u_{1\alpha} < 0$		$u_{1\alpha} < 0$		$u_{1\alpha} > 0$	
Kvadrant	Q_1		Q_2		Q_3		Q_4	
	$l > 0$	$l < 0$	$l < 0$	$l > 0$	$l > 0$	$l < 0$	$l < 0$	$l > 0$
Sektor	S_1	S_2	S_2	S_3	S_4	S_5	S_5	S_6

KAPITOLA 4: MODEL REFERENCE ADAPTIVE SYSTEM – MRAS

Model reference adaptive system, zkráceně MRAS, je metoda založená na uzavřené smyčce, ve které jsou upravovány určité parametry za použití dvou rozdílných dílčích modelů, jmenovitě referenčního a adaptivního. Upravován je adaptivní model a to tak, aby chyba mezi výstupní hodnotou z tohoto modelu a modelu referenčního byla co nejlépe nulová. Referenční systém neobsahuje proměnou, která je upravována a je tak na ní nezávislá. Naopak adaptivní model se používá k výpočtu stejných stavových proměnných využívající jiné sady a vstupů, které zahrnují parametr, který se má odhadnout. Obecné nejjednodušší schéma takového systému je pak na následujícím obrázku [44].



Obr. 4-1 Obecné nejjednodušší schéma MRAS [45] (upraveno).

Při porovnání s modelem pro učení umělé neuronové sítě pomocí učitele z kapitoly 1.4.1 je vidět, že se principiálně jedná o stejnou metodu.

V elektrických pohonech se MRAS používá většinou pro bez senzorový odhad otáček a adaptivní mechanismus je realizován pomocí PI regulátoru. Při tomto použití jsou eliminovány problémy mechanických a optických senzorů pro měření rychlosti jako jsou vyšší hardwarová náročnost, vyšší cena a větší velikost pohonného systému [44], [46].

V této práci je jako adaptivní použit model na bázi umělé neuronové sítě, jejíž adaptivní mechanismus je dán úpravou prahových vah algoritmem zpětného šíření chyby.

KAPITOLA 5: ODHAD PARAMETRŮ ASYNCHORNÍHO MOTORU POMOCÍ UMĚLÝCH NEURONOVÝCH SÍTÍ

V této kapitole je popsáno využití kombinace MRAS a umělých neuronových sítí pro odhad rotorového a statorového odporu. Zároveň je tento způsob odhadu možné jednoduše použít i pro určení elektrické úhlové rychlosti rotoru, proto i tato možnost zde bude diskutována.

5.1 Odhad rotorového odporu a elektrické úhlové rychlosti

Jak již bylo lehce nastíněno v kapitole 3.2.1, tak zvyšující se elektrický odpor rotoru, především vlivem teploty, může mít za následek zhoršenou kvalitu regulace. Je to dáno tím, že vektor magnetického rotorového toku počítaný z proudového modelu již neodpovídá při změně rotorového odporu skutečné hodnotě a velikost tohoto odporu se může zvýšit až dvojnásobně. Regulátor pak reguluje podle hodnoty, která již neodpovídá skutečnosti. Z tohoto důvodu vzniklo mnoho, více či méně úspěšných způsobů právě na odhad rotorového odporu. Jednou z možností je MRAS ve spojení s umělými neuronovými sítěmi. Tento přístup byl již diskutován například v [47]–[49]. V takovém případě by pro odhad měla stačit jednoduchá jednovrstvá síť s lineární aktivační funkcí. V uvedených referencích je uváděna jako síť dvouvrstvá, avšak jak již bylo uvedeno v kapitole 1.3.1, tak v této práci je takováto síť pojmenována jako jednovrstvá, neboť obsahuje pouze vstupy a výstupní vrstvu.

Pro použití typu odhadu postaveném na MRAS je potřeba mít referenční model. V tomto případě bude k tomuto účelu sloužit napěťový U-I model, popsáný v kapitole 3.2.2, neboť ten není závislý na rotorovém odporu. Jako adaptivní model pak slouží proudový I-n model, probíraný v kapitole 3.2.1, namodelovaný právě pomocí zmíněné neuronové sítě. Výstupní veličinou obou těchto modelů jsou složky vektoru magnetického toku rotoru, které jsou porovnávány a adaptivní model je upravován tak, aby jejich rozdíl byl nulový. K tomu účelu lze soustavu rovnic (3-6) diskretizovat nahrazením derivace dopřednou diferencí jako

$$\begin{aligned} \frac{\psi_{2\alpha}^{adapt}(k+1) - \psi_{2\alpha}^{adapt}(k)}{T_s} &= \frac{L_m}{T_r} i_{1\alpha}(k) - \omega \psi_{2\beta}^{adapt}(k) - \frac{1}{T_r} \psi_{2\alpha}^{adapt}(k) \\ \frac{\psi_{2\beta}^{adapt}(k+1) - \psi_{2\beta}^{adapt}(k)}{T_s} &= \frac{L_m}{T_r} i_{1\beta}(k) + \omega \psi_{2\alpha}^{adapt}(k) - \frac{1}{T_r} \psi_{2\beta}^{adapt}(k), \end{aligned} \quad (5-1)$$

kde horní indexy adapt a ref značí adaptivní, respektive referenční model. Konstanta T_s je vzorkovací perioda, která je v této práci spojena s periodou PWM střídače, tedy $T_s = T_{pwm} = 100 \mu s$. Rotorový magnetický tok v čase $k+1$ pak lze dostat upravením rovnice (5-1) do tvaru

$$\begin{aligned} \psi_{2\alpha}^{adapt}(k+1) &= \left(1 - \frac{T_s}{T_r}\right) \psi_{2\alpha}^{adapt}(k) - T_s \omega \psi_{2\beta}^{adapt}(k) + \frac{T_s L_m}{T_r} i_{1\alpha}(k) \\ \psi_{2\beta}^{adapt}(k+1) &= \left(1 - \frac{T_s}{T_r}\right) \psi_{2\beta}^{adapt}(k) + T_s \omega \psi_{2\alpha}^{adapt}(k) + \frac{T_s L_m}{T_r} i_{1\beta}(k). \end{aligned} \quad (5-2)$$

Následně lze provést nahrazení $W_1 = 1 - \frac{T_s}{T_r}$, $W_2 = T_s \omega$ a $W_3 = \frac{T_s L_m}{T_r}$. Soustava rovnic (5-2) pak lze upravit do výsledného tvaru jako [2], [47], [48]

$$\psi_{2\alpha}^{adapt}(k+1) = W_1 \psi_{2\alpha}^{adapt}(k) - W_2 \psi_{2\beta}^{adapt}(k) + W_3 i_{1\alpha}(k) \quad (5-3)$$

$$\psi_{2\beta}^{adapt}(k+1) = W_1\psi_{2\beta}^{adapt}(k) + W_2\psi_{2\alpha}(k) + W_3i_{1\beta}(k).$$

Po zavedení následujících matic

$$X_1 = \begin{pmatrix} \psi_{2\alpha}^{adapt}(k) \\ \psi_{2\beta}^{adapt}(k) \end{pmatrix}, \quad (5-4)$$

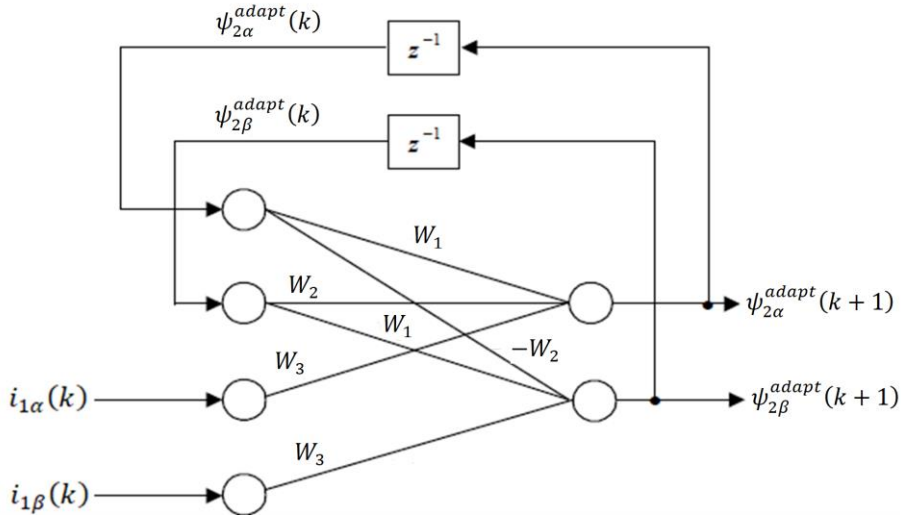
$$X_2 = \begin{pmatrix} -\psi_{2\beta}^{adapt}(k) \\ \psi_{2\alpha}^{adapt}(k) \end{pmatrix}, \quad (5-5)$$

$$X_3 = \begin{pmatrix} i_{1\alpha}(k) \\ i_{1\beta}(k) \end{pmatrix}, \quad (5-6)$$

se může soustava rovnic (5-3) přepsat do vektorového tvaru v obecném vektorovém prostoru jako [2], [47], [48]

$$\overrightarrow{\psi_{2\alpha\beta}^{adapt}}(k+1) = W_1X_1 - W_2X_2 + W_3X_3, \quad (5-7)$$

kde členy W_1, W_2 a W_3 lze chápat jako váhy neuronové sítě a členy $\psi_{2\alpha}, \psi_{2\beta}, i_{1\alpha}$ a $i_{1\beta}$ jako její vstupy. Takto daná umělá neuronová síť je pak na Obr. 5-1.



Obr. 5-1 Proudový I-n model pomocí umělé neuronové sítě pro výpočet složek vektoru magnetického toku rotoru [50] (upraveno).

Všechny váhy jsou upravovány pomocí algoritmu zpětného šíření, který je popsán v kapitole 1.4.4. Chyba odpovídající rozdílu žádané a skutečné hodnoty neboli rozdíl mezi referenčním a adaptivním modelem, je dán jako

$$\vec{e}_1(k) = \left(\overrightarrow{\psi_2^{ref}}(k) - \overrightarrow{\psi_2^{adapt}}(k) \right). \quad (5-8)$$

Tomu odpovídající kvadratická chyba je pak dána jako

$$\vec{E}_1(k) = \frac{1}{2} (\overrightarrow{\psi_2^{ref}}(k) - \overrightarrow{\psi_2^{adapt}}(k))^2. \quad (5-9)$$

Pro připomenutí je úprava vah pomocí algoritmu zpětného šíření obecně dána následujícími rovnicemi

$$\Delta w_{ji}(k) = -\eta \frac{\delta E(k)}{\delta w_{ji}(k)} = \eta \delta_j(k) x_i(k), \quad (5-10)$$

$$\delta_j(k) = \frac{\delta E(k)}{\delta v_j(k)} = e_j(k) \varphi_j'(v_j(k)), \quad (5-11)$$

$$w_{ji}(k+1) = w_{ji}(k) - \eta \frac{\delta E(k)}{\delta w_{ji}(k)} + \mu \Delta w_{ji}(k-1). \quad (5-12)$$

Jelikož výstupní aktivační funkce je lineární, tak platí, že $\varphi_j'(v_j(k)) = 1$. Po dosazení rovnice (5-11) a (5-8) do rovnice (5-10) a použití členů X_1, X_2 a X_3 je pak úprava vah pro W_1, W_2 a W_3 dána jako

$$W_1(k+1) = W_1(k) + \eta_1 (\overline{\psi_2^{ref}}(k) - \overline{\psi_2^{adapt}}(k))^T X_1 + \mu_1 \Delta W_1(k-1), \quad (5-13)$$

$$W_2(k+1) = W_2(k) + \eta_2 (\overline{\psi_2^{ref}}(k) - \overline{\psi_2^{adapt}}(k))^T X_2 + \mu_2 \Delta W_2(k-1), \quad (5-14)$$

$$W_3(k+1) = W_3(k) + \eta_3 (\overline{\psi_2^{ref}}(k) - \overline{\psi_2^{adapt}}(k))^T X_3 + \mu_3 \Delta W_3(k-1). \quad (5-15)$$

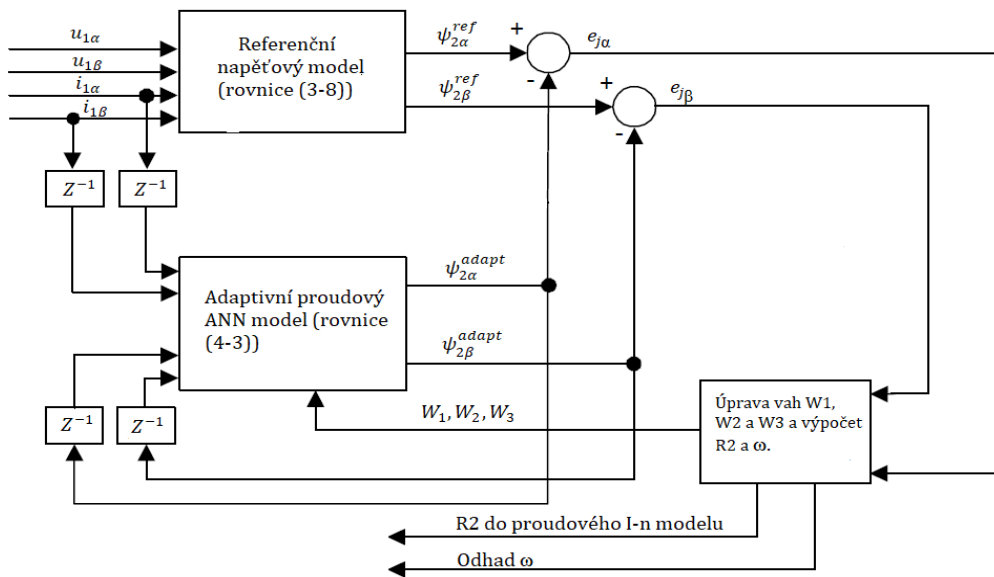
Je vidět, že pro každou váhu je uvažován jiný koeficient učení η a jiný momentový faktor μ . Z úprav vah W_1 nebo W_3 je pak možné spočítat rotorový odpor a z váhy W_2 elektrickou úhlovou rychlost jako

$$R_2 = \frac{L_2(1 - W_1)}{T_s}, \quad (5-16)$$

$$R_2 = \frac{L_2 W_3}{L_m T_s}, \quad (5-17)$$

$$\omega = \frac{W_2}{T_s}. \quad (5-18)$$

Blokový diagram MRAS využívající adaptivní model na bázi umělých neuronových sítí je pak na následujícím obrázku.



Obr. 5-2 Schéma kombinace MRAS a ANN pro odhad R_1, R_2 a ω [50] (upraveno).

Koeficient učení se prozatím uvažovali konstantní, teoreticky je ale ideální, aby se snižoval spolu s kvadratickou chybou a tím pádem by měla být síť přesnější při malé chybě. V této práci tedy byla vyzkoušena úprava koeficientu učení pomocí funkce členu ζ_i závislého na kvadratické chybě. Ten je dán jako

$$\zeta_i(k) = \Delta w_{ji}(k) \Delta w_{ji}(k-1). \quad (5-19)$$

Následně se zavede funkce $f(\zeta_i)$, která na základě zmíněného parametru mění hodnotu koeficientů učení jako

$$f(\zeta_i) = \text{sign}(\zeta_i) \frac{\alpha_0}{1 + e^{-\zeta_i \text{sign}(\zeta_i)}},$$

kde α_0 je kladný člen. Koeficient učení by teď pro k -tý krok měl být dán jako

$$\eta_i(k) = \eta_i(k-1)(1 + f(\zeta_i(k-1))).$$

Při testování modelu se ovšem ukázalo, že s takovou úpravou koeficientů učení se nedosahuje lepších výsledků. Do modelu se tím ale vnáší další člen, který je potřeba pro každou váhu nastavit jinak a tím roste počet možných kombinací a tím i náročnost správného naladění sítě. Tím pádem bylo od této metody ustoupeno.

5.2 Odhad statorového odporu

U statorového odporu může docházet ke změně jeho hodnoty podobně jako u rotorového odporu. Při použití proudového I-n modelu za účelem regulace, nemá změna statorového odporu na regulaci vliv. Vystupuje však v napětovém U-I modelu pro dohad rotorového odporu a ten je tak na změně statorového odporu závislý. Jeho změna pak může mít vliv na kvalitu odhadu.

Stejně jako v předchozí kapitole, i zde je použita jednoduchá jednovrstvá dopředná neuronová síť, diskutována v těchto zdrojích [47]–[49], sloužící jako adaptivní model pro metodu MRAS. V tomto případě síť odhaduje hodnotu složek statorového proudu v souřadnicovém systému $\alpha\beta$. Jako referenční hodnota slouží přímo měřený statorový proud transformovaný pomocí Clarkovy transformací do příslušného systému.

Úpravou soustav rovnic (3-6) a (3-9) lze dostat rovnice pro vyjádření složek statorového proudu jako

$$\begin{aligned} \frac{di_{1\alpha}^{adapt}}{dt} = & -\frac{R_1}{\sigma L_1} i_{1\alpha}^{adapt} - \frac{L_m^2}{\sigma L_1 L_2 T_r} i_{1\alpha}^{adapt} + \frac{L_m}{\sigma L_1 L_2 T_r} \psi_{2\alpha} + \frac{L_m}{\sigma L_1 L_2} \omega \psi_{2\beta} \\ & + \frac{1}{\sigma L_1} u_{1\alpha}, \end{aligned} \quad (5-20)$$

$$\begin{aligned} \frac{di_{1\beta}^{adapt}}{dt} = & -\frac{R_1}{\sigma L_1} i_{1\beta}^{adapt} - \frac{L_m^2}{\sigma L_1 L_2 T_r} i_{1\beta}^{adapt} + \frac{L_m}{\sigma L_1 L_2 T_r} \psi_{2\beta} - \frac{L_m}{\sigma L_1 L_2} \omega \psi_{2\alpha} \\ & + \frac{1}{\sigma L_1} u_{1\beta}, \end{aligned}$$

kde horní indexy ref a adapt opět odkazují na referenční, respektive adaptivní model. Po úpravě do diskrétního tvaru pak lze tuto soustavu přepsat do tvaru

$$\begin{aligned}
i_{1\alpha}^{adapt}(k+1) &= \left(1 - \frac{R_1 T_s}{\sigma L_1} - \frac{L_m^2 T_s}{\sigma L_1 L_2 T_r}\right) i_{1\alpha}^{adapt}(k) + \frac{L_m T_s}{\sigma L_1 L_2 T_r} \psi_{2\alpha}(k) \\
&\quad + \frac{L_m T_s}{\sigma L_1 L_2} \omega \psi_{2\beta}(k) + \frac{T_s}{\sigma L_1} u_{1\alpha}(k), \\
i_{1\beta}^{adapt}(k+1) &= \left(1 - \frac{R_1 T_s}{\sigma L_1} - \frac{L_m^2 T_s}{\sigma L_1 L_2 T_r}\right) i_{1\beta}^{adapt}(k) + \frac{L_m T_s}{\sigma L_1 L_2 T_r} \psi_{2\beta}(k) \\
&\quad - \frac{L_m T_s}{\sigma L_1 L_2} \omega \psi_{2\alpha}(k) + \frac{T_s}{\sigma L_1} u_{1\beta}(k).
\end{aligned} \tag{5-21}$$

Stejně jako u odhadu rotorového odporu lze zavést váhy a provést nahrazení

$$W_4 = \left(1 - \frac{R_1 T_s}{\sigma L_1} - \frac{L_m^2 T_s}{\sigma L_1 L_2 T_r}\right), \tag{5-22}$$

$$W_5 = \frac{L_m T_s}{\sigma L_1 L_2 T_r}, \tag{5-23}$$

$$W_6 = \frac{L_m T_s}{\sigma L_1 L_2} \omega, \tag{5-24}$$

$$W_7 = \frac{T_s}{\sigma L_1}, \tag{5-25}$$

a vstupní veličiny se vyjádří maticově jako

$$X_4 = \begin{pmatrix} i_{1\alpha}^{adapt}(k) \\ i_{1\beta}^{adapt}(k) \end{pmatrix}, \tag{5-26}$$

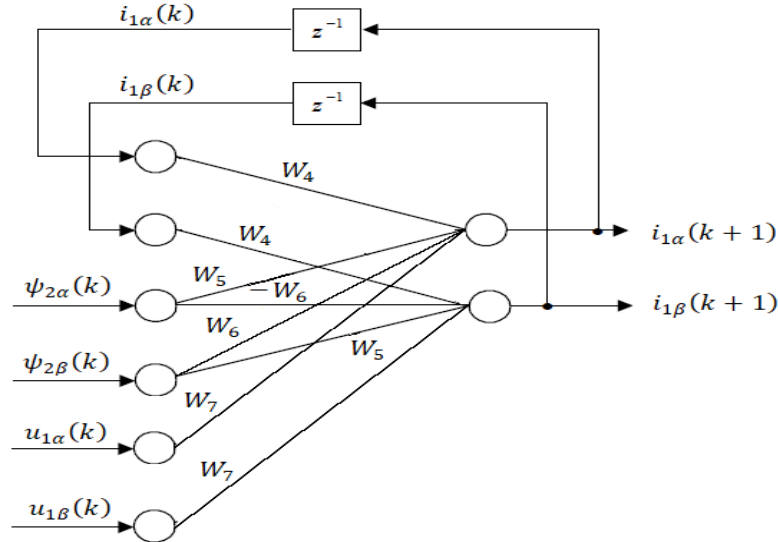
$$X_5 = \begin{pmatrix} \psi_{2\alpha}(k) \\ \psi_{2\beta}(k) \end{pmatrix}, \tag{5-27}$$

$$X_6 = \begin{pmatrix} \psi_{2\beta}(k) \\ -\psi_{2\alpha}(k) \end{pmatrix}, \tag{5-28}$$

$$X_7 = \begin{pmatrix} u_{1\alpha}(k) \\ u_{1\beta}(k) \end{pmatrix}. \tag{5-29}$$

Rovnici (5-21) pak lze upravit do výsledného vektorového tvaru daného rovnicí (5-30) a schéma znázorňující umělou neuronovou síť danou touto rovnicí je na Obr. 5-3.

$$\overrightarrow{i_{1\alpha\beta}^{adapt}}(k+1) = W_4 X_4 - W_5 X_5 + W_6 X_6 + W_7 X_7. \tag{5-30}$$



Obr. 5-3 Proudový I-n model pomocí umělé neuronové sítě pro výpočet složek vektoru magnetického toku rotoru [50] (upraveno).

V tomto modelu nejsou upravovány všechny váhy jako u odhadu statorového odporu, ale pouze hodnota W_4 , W_5 a W_6 . Váha W_7 je konstantní. V uvedených zdrojích je sice jako dostačující uváděna úprava pouze váhy W_4 , to se ale v praxi ukázalo jako nefunkční. Proto jsou zde upravovány ještě další dvě váhy, aby se zlepšila kvalita odhadu.

Úprava je u adaptivních vah prováděna opět pomocí algoritmu zpětného šíření, přičemž chyba mezi referenčním a adaptivním modelem, a tomu odpovídající kvadratická chyba, jsou dány jako

$$\vec{e}_2(k) = \left(\vec{i}_1^{ref}(k) - \vec{i}_1^{adapt}(k) \right), \quad (5-31)$$

$$\vec{E}_2(k) = \frac{1}{2} (\vec{i}_1^{ref}(k) - \vec{i}_1^{adapt}(k))^2. \quad (5-32)$$

Úprava vah W_4 , W_5 a je W_6 potom následující

$$W_4(k+1) = W_4(k) + \eta_4 (\vec{i}_1^{ref}(k) - \vec{i}_1^{adapt}(k))^T X_4 + \mu_4 \Delta W_4(k-1), \quad (5-33)$$

$$W_5(k+1) = W_5(k) + \eta_5 (\vec{i}_1^{ref}(k) - \vec{i}_1^{adapt}(k))^T X_5 + \mu_5 \Delta W_5(k-1), \quad (5-34)$$

$$W_6(k+1) = W_6(k) + \eta_6 (\vec{i}_1^{ref}(k) - \vec{i}_1^{adapt}(k))^T X_6 + \mu_6 \Delta W_6(k-1). \quad (5-35)$$

Stejně jako pro úpravu vah v předešlé kapitole, tak i zde byla vyzkoušena metoda pro úpravu koeficientu učení v průběhu výpočtu. Ani zde však nepřinesla zlepšení, a proto se koeficienty učení a momentové faktory berou jako konstantní. Statorový odpor lze potom přepočítat z úpravy W_4 jako

$$R_1 = \frac{\sigma L_1 (1 - W_4 - \frac{L_m^2 R_2 T_s}{\sigma L_1 L_2^2})}{T_s}. \quad (5-36)$$

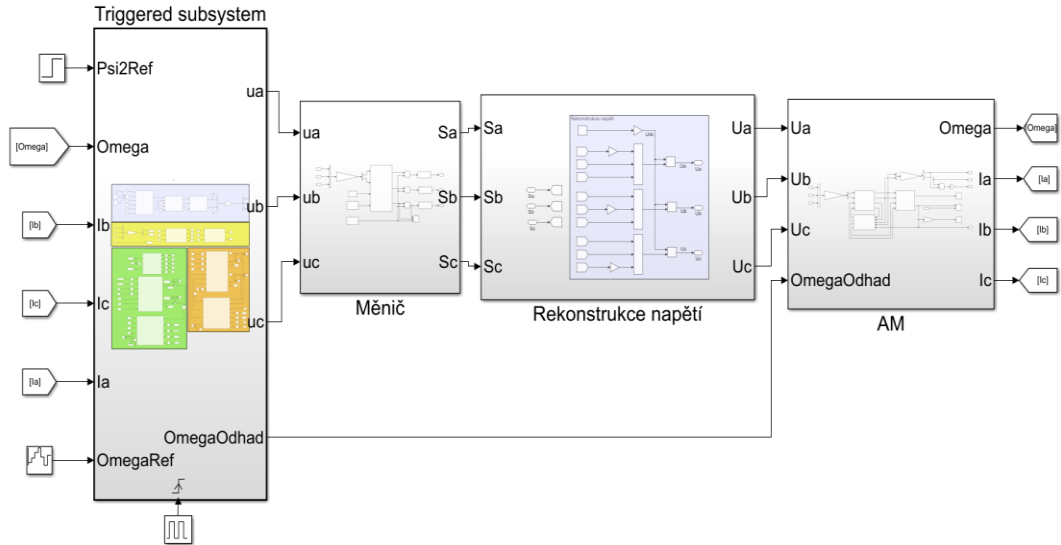
KAPITOLA 6: MODEL POHONU S ASYNCHRONNÍM MOTOREM V PROSTŘEDÍ MATLAB SIMULINK

V této kapitole jsou postupně stručně popsány jednotlivé části modelovaného pohonu. Jmenovitě se postupně jedná o model asynchronního motoru, střídače, regulátoru a o modely neuronových sítí pro odhad jednotlivých veličin. Bloky asynchronního motoru a měniče jsou realizovány jako spojité, zatímco zbytek modelu je realizován disktrétně pomocí bloku „Triggered Subsystem“. Spínací perioda tohoto bloku je stejná, jako perioda PWM střídače, a to konkrétně 100 μ s. Tímto řešením se lze více přiblížit chování v reálné situaci, kdy jsou jednotlivé algoritmy počítány disktrétně v procesoru.

Jako řešiče byly použity typy „ode4 (Runge-Kutta)“ s konstantním krokem o velikosti 5 μ s, případně „ode8“ (Dormand-Prince) pro dosažení lepší přesnosti. Tyto řešiče spolu se zvolenou velikostí kroku zajišťují dostatečnou přesnost a zároveň poměrně slušnou rychlost, kdy se výpočet modelu pohybuje v řádu desítek sekund, až jednotek minut pro „ode8“, v závislosti na délce simulace. Nejlepší kvality simulací bylo dosaženo s krokem o jeden řád nižší, tedy 0,5 μ s a to z důvodu menšího zakmitání a přesnějších výsledků. Obecně je však doba simulace s takto nízkým krokem výrazně delší a jedná se o jednotky až desítky minut podle délky celé simulace. Parametry nutné pro matematický model asynchronního motoru jsou v Tab. 6-1 a schéma celého modelu je na Obr. 6-1. Prezentovaný model je dostupný v příloze k této práci.

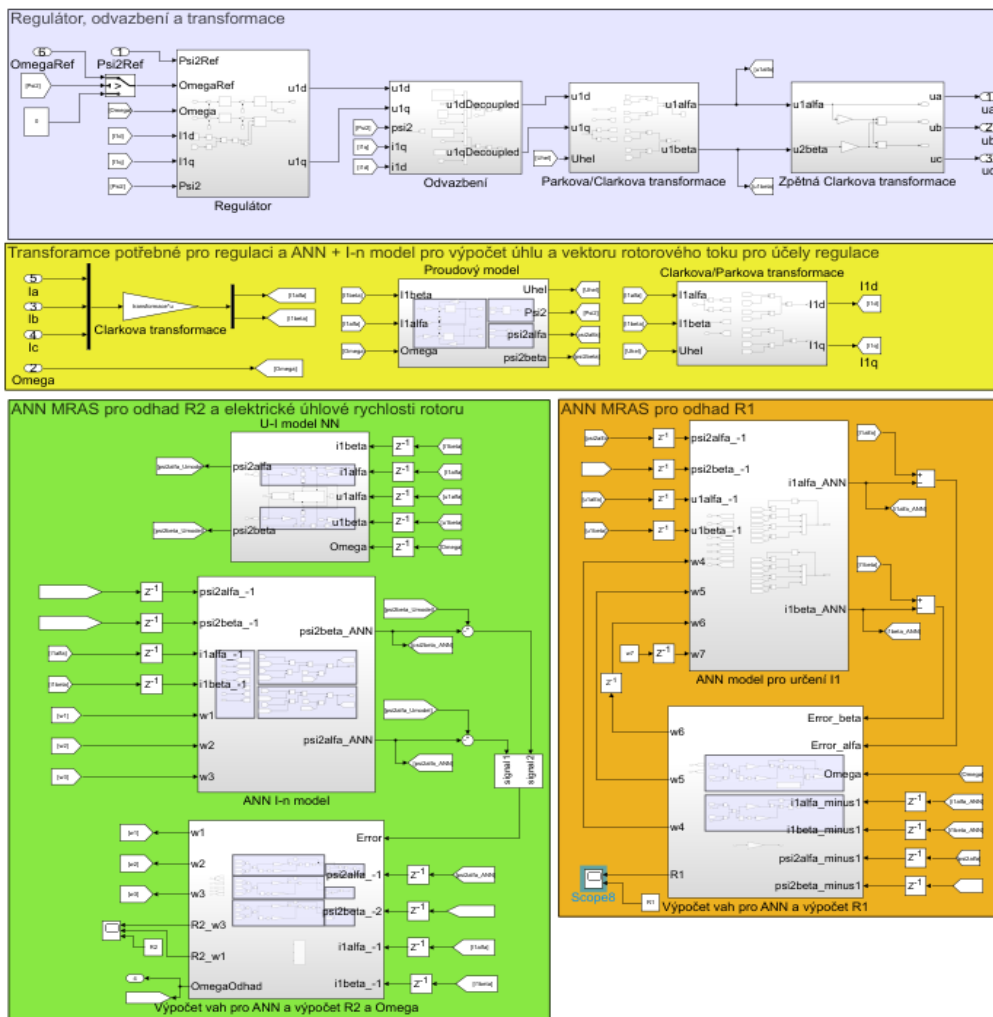
Tab. 6-1 Parametry modelovaného motoru.

Štítkové parametry modelovaného motoru		Ostatní měřitelné parametry potřebné pro matematický model	
P_n	12 kW	R_1	370 m Ω
I_n	22 A	R_2	370 m Ω
U_n	380 V	$L_{1\sigma}$	2,27 mH
f_n	50 Hz	$L_{2\sigma}$	2,27 mH
$\cos \varphi_n$	0,8	L_m	82,5 mH
n_n	1460 min	L_1	84,77 mH
p_p	2	L_2	84,77 mH
		J	0,4 kg·m ⁻²



Obr. 6-1 Celkový model pohonu.

Jelikož blok řešený diskrétně obsahuje ještě další oddělené části s mnoha bloky, jeho detail je na následujícím obrázku a jednotlivé části jsou pak blíže probrány v konkrétních kapitolách.



Obr. 6-2 Detail celého bloku "Triggered subsystem".

6.1 Model asynchronního motoru a střídače

Pro modelování asynchronního motoru je možné vycházet z několika možných matematických popisů. V této práci je použit matematický model pro výpočet složek vektorů statorového proudu \vec{i}_1 a rotorového proudu \vec{i}_2 v souřadnicovém systému $\alpha\beta$ spojeného se statorem a platí tedy, že $\omega_k = 0$. Dále se uvažuje motor s kotvou nakrátko, tedy $\vec{u}_2 = 0$ a transformační konstanta jako $K = 2/3$. S těmito předpoklady je možné soustavu rovnic (2-13), po dosažení rovnic pro magnetický tok do rovnic pro napětí a vyjádření nejvyšší derivace, přepsat do následujícího tvaru:

$$\begin{aligned} \frac{di_{1\alpha}}{dt} &= \frac{1}{L_m} [u_{1\alpha} - R_1 i_{1\alpha} - L_m \frac{di_{2\alpha}}{dt}] \\ \frac{di_{1\beta}}{dt} &= \frac{1}{L_m} [u_{1\beta} - R_1 i_{1\beta} - L_m \frac{di_{2\beta}}{dt}] \\ \frac{di_{2\alpha}}{dt} &= \frac{1}{L_2} [-R_2 i_{2\alpha} - L_m \frac{di_{1\alpha}}{dt} - \omega(L_2 i_{2\beta} + L_m i_{1\beta})] \\ \frac{di_{2\beta}}{dt} &= \frac{1}{L_2} [-R_2 i_{2\beta} - L_m \frac{di_{1\beta}}{dt} - \omega(L_2 i_{2\alpha} + L_m i_{1\alpha})]. \end{aligned} \quad (6-1)$$

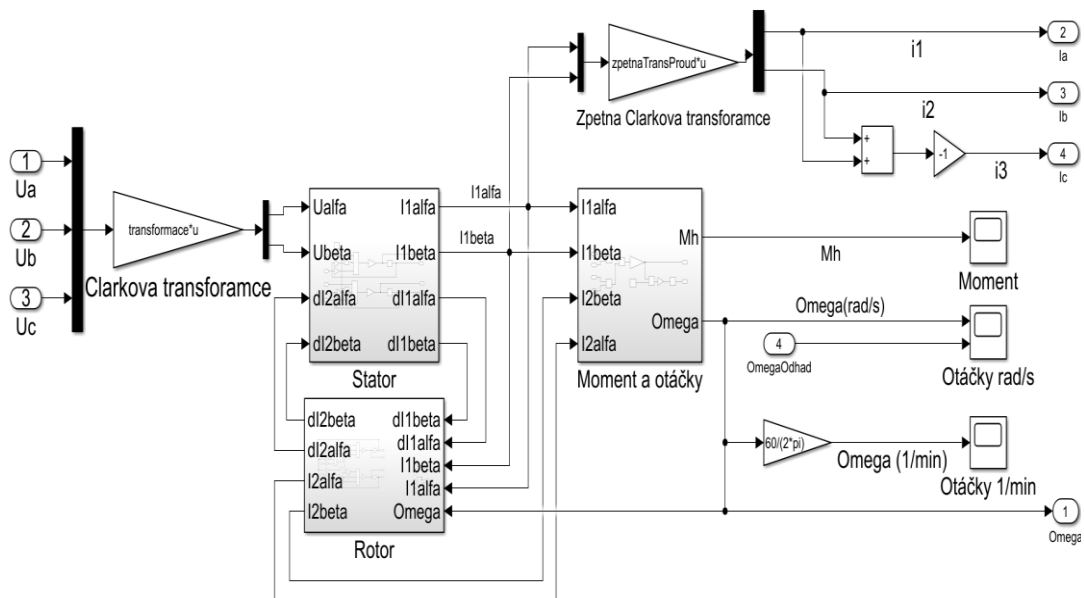
Dále jsou pro výpočet momentu a otáček ještě použity následující rovnice.

$$M = \frac{3}{2} p_p L_m (i_{2\alpha} i_{1\beta} - i_{2\beta} i_{1\alpha}), \quad (6-2)$$

$$M - M_z = J \frac{d\Omega}{dt}, \quad (6-3)$$

$$\omega = p_p \Omega, \quad (6-4)$$

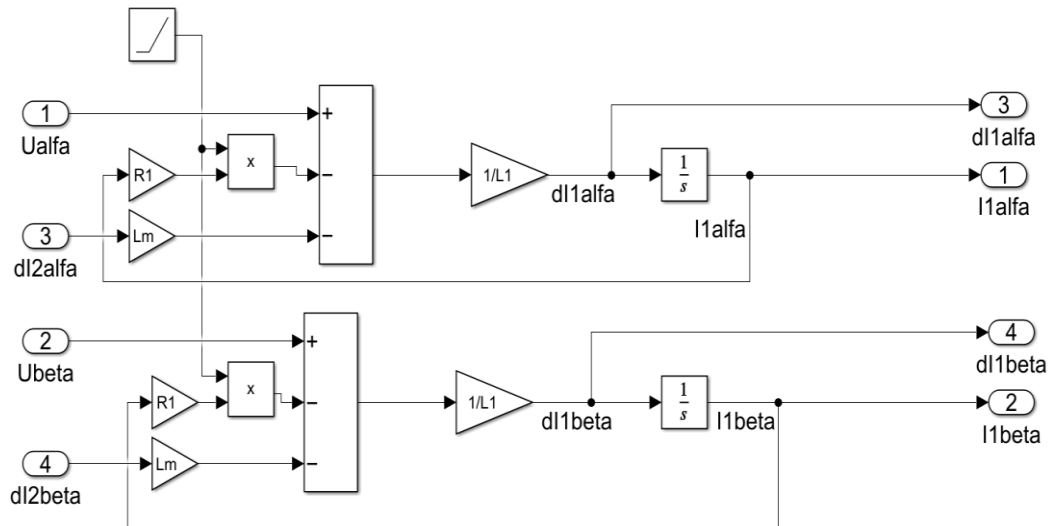
Kde M je hnací moment motoru, M_z je zátěžný moment a J je moment setrvačnosti motoru. Výsledný model asynchronního motoru je pak na následujícím obrázku.



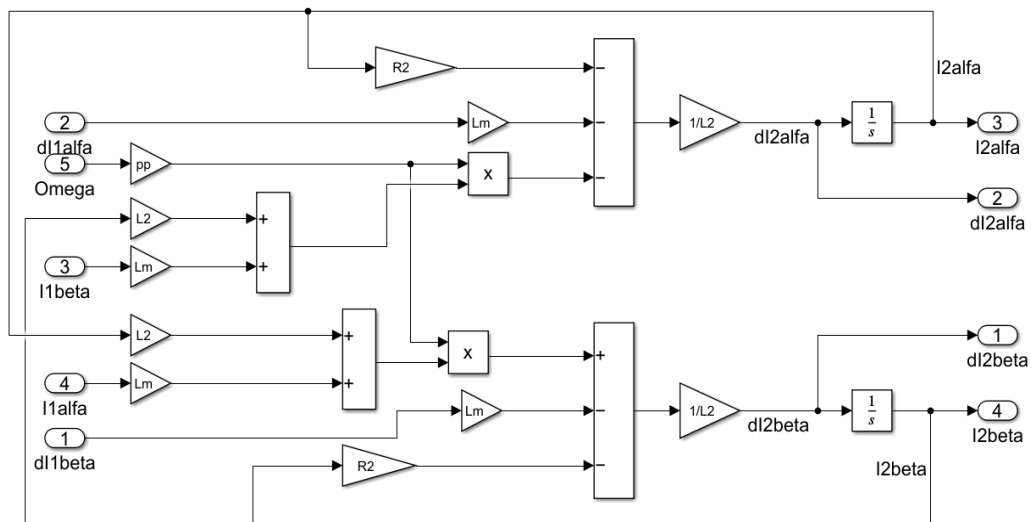
Obr. 6-3 Celkový model asynchronního motoru.

Vstupní veličiny jsou jednotlivé fáze statorového napětí, které jsou následně pomocí Clarkovy transformace dané rovnicí (2-8) převedeny do souřadnicového systému $\alpha\beta$. Ty pak vstupují do bloku „Stator“, který je spolu s blokem „Rotor“

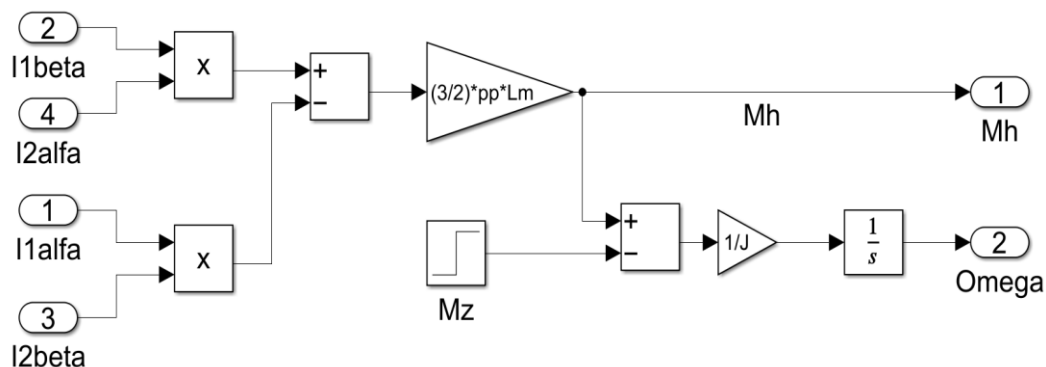
definován soustavou rovnic (6-1). Dále se zde nachází blok „Moment a otáčky“ právě pro výpočet momentu a otáček podle soustavy rovnic (6-4). Výstupem z celého bloku asynchronního motoru je mechanická úhlová rychlost rotoru „Omega“ a proudy jednotlivých fází statoru. Ty jsou předtím ještě pomocí zpětné Clarkové transformace dané rovnicí (2-9) převedeny do třífázového souřadnicového systému *abc*. Detaily ostatních bloků jsou na následujících obrázcích.



Obr. 6-4 Model statoru.

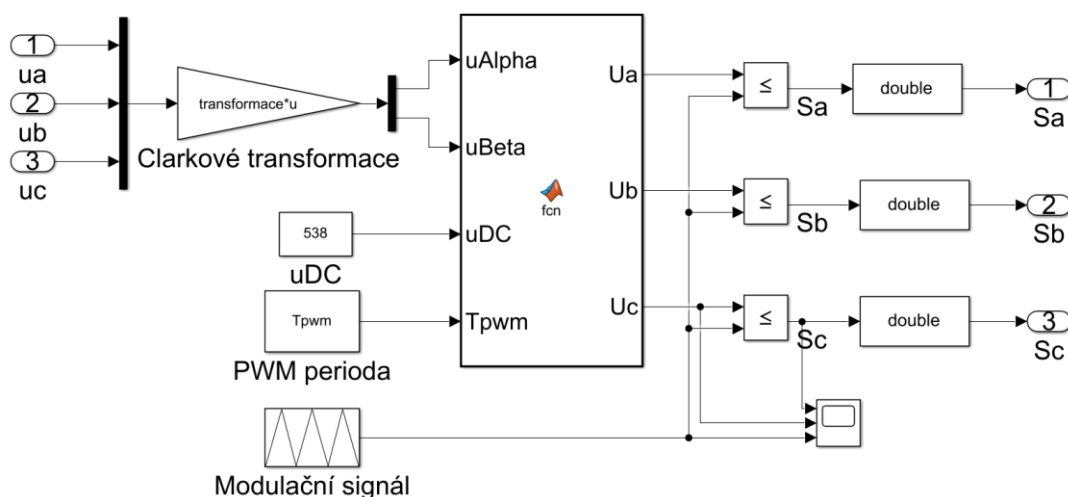


Obr. 6-5 Model rotoru.



Obr. 6-6 Schéma modelu pro výpočet hnacího momentu a otáček.

Na Obr. 6-1 jsou vidět bloky „Měnič“ a „Rekonstrukce napětí“, které simulují třífázový dvouúrovňový střídač napětí řízený pomocí modulace prostorového vektoru. Tento princip je blíže popsán v kapitole 3.3. Detail bloku „Měnič“ je vidět na následujícím obrázku.



Obr. 6-7 Model měniče.

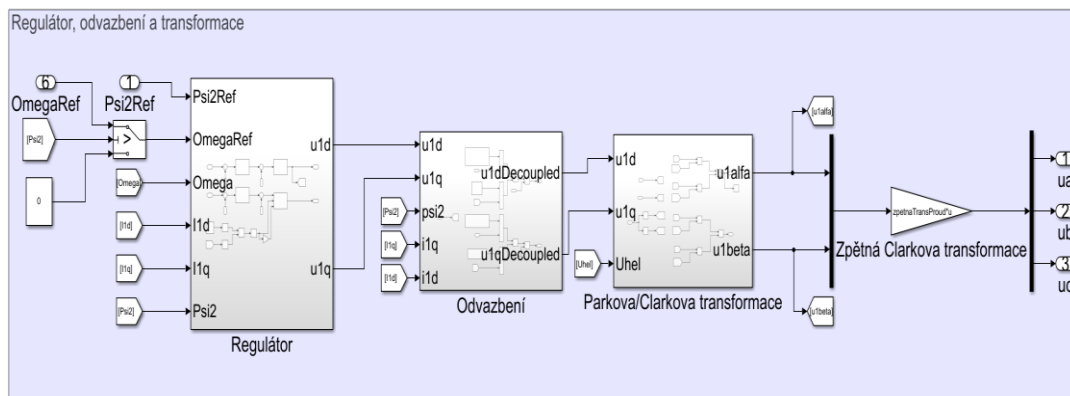
Nejdůležitější částí je prvek „MATLAB function“ ve kterém je napsaný kód pro zjištění aktuálního kvadrantu a sektoru, výpočet spínacích časů pro okrajové vektory a vektor nulový, a nakonec jsou vypočteny komparační hodnoty pro jednotlivé fáze. Tyto hodnoty jsou dány jako poměr jednotlivých spínacích časů ze spínacích vzorů na Obr. 3-5. Komparační hodnoty jsou následně porovnány s trojúhelníkovým nosným signálem o periodě 100 μ s. Pokud je komparační hodnota větší než pilovitý signál, tak výstupní signál nabývá hodnoty 0 a dojde k teoretickému sepnutí některého z tranzistorů v dolní řadě, v opačném případě je hodnota signálu 1 což simuluje sepnutí tranzistoru v horní řadě. Tyto binární signály jsou v dalším již zmíněném bloku „Rekonstrukce napětí“ pouze přepočítány na žádanou hodnotu napětí podle následující rovnice [51]

$$\begin{pmatrix} u_a \\ u_b \\ u_c \end{pmatrix} = \frac{1}{3} U_{DC} \begin{pmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{pmatrix} \begin{pmatrix} S_a \\ S_b \\ S_c \end{pmatrix}. \quad (6-5)$$

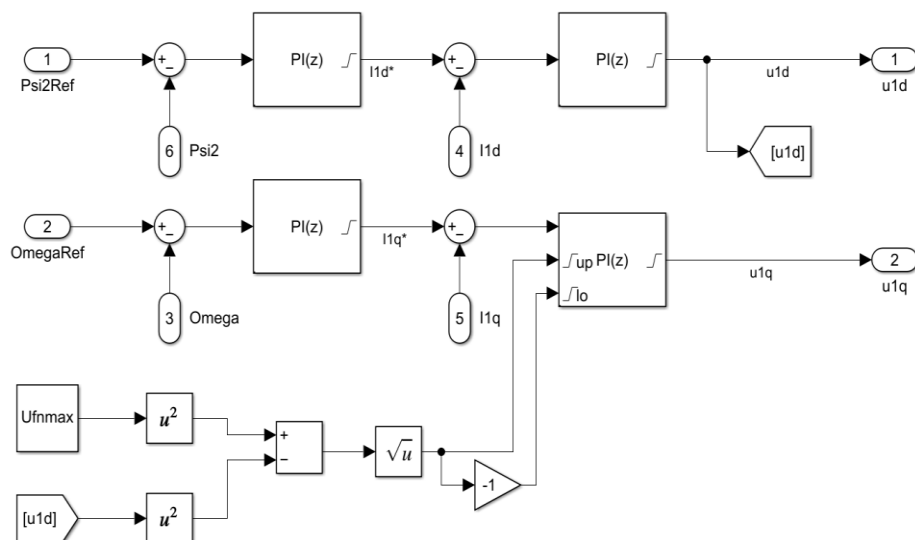
U stejnosměrného meziobvodu se uvažuje, že je napájený ze šesti-pulzního diodového usměrňovače připojeného na síť 400 V a 50 Hz. Tomu odpovídá střední hodnota napětí v meziobvodu 538 V. Stejnosměrný obvod se tedy uvažuje ideální bez kolísání napětí a jeho napěťová hladina tak do modelu vstupuje jako konstanta.

6.2 Model regulátoru s odvazbením

Jak již bylo řečeno výše, tak všechny ostatní části modelu, kromě těch zmíněných v předchozí kapitole, jsou realizovány diskrétně a spínány s periodou 100 μ s. Jednou z takto spínaných částí je i část s regulátorem, odvazbením a transformacemi, jehož schéma je na Obr. 6-8. Detail samotného bloku „Regulátor“ je na pak k vidění na Obr. 6-9.



Obr. 6-8 Schéma modelu s regulátorem, odvazbením a transformacemi.



Obr. 6-9 Model regulátoru.

Vstupní veličiny jsou žádané hodnoty rotorového toku a mechanické úhlové rychlosti rotoru. Ty jsou porovnávány se skutečnými hodnotami rotorového toku, respektive rychlosti a jednotlivé rozdíly těchto veličin vstupují v podobě regulační odchylky do regulátoru rotorového toku, respektive do regulátoru otáček. Jak bylo popsáno v kapitole 3.1, tak jednotlivými složkami vektoru statorového proudu v souřadnicovém systému dq je možné řídit magnetický tok rotoru a hnací moment. Výstupem z tokového regulátoru je tak žádaná velikost proudu i_{1d} , jehož rozdíl

oproti skutečné hodnotě vstupuje do regulátoru tokotvorné složky. Obdobně je tomu u regulátoru otáček, kdy výstupní žádaná hodnota je velikost proudu i_{1q} a regulační odchylka vstupuje do regulátoru momentotvorné složky. Výstupem z obou těchto regulátorů je žádaná hodnota statorového napětí v souřadnicovém systému dq . Jak je patrné z Obr. 6-9, tak všechny regulátory jsou typu PI, jejich výstupy jsou omezeny saturací a jako anti-windup zapojení je použita metoda clamping, kterou je možné zapnout přímo v jednotlivých blocích regulátorů. Regulátory byly naladěny tak, že se nejdříve naladil samotný regulátor tokotvorné složky, poté jeho nadřazený regulátor rotorového toku, následně se připojil regulátor momentotvorné složky, který má stejné hodnoty, jako regulátor tokotvorné složky a jako poslední byl naladěn regulátor otáček. Hodnoty konstant jednotlivých regulátorů jsou uvedeny v Tab. 6-2.

Tab. 6-2 Konstanty pro jednotlivé PI regulatory.

Regulátor	r_0	r_{-1}
Tokotvorné složky	20	500
Momentotvorné složky	20	500
Rotorového toku	100	1000
Otáček	100	3

Omezení výstupních hodnot pro vnitřní regulátory statorového proudu, je dáno amplitudou jmenovitého fázového napětí statoru $U_{fn(max)}$, neboť jednotlivé složky tohoto napětí jsou žádané hodnoty na výstupu těchto regulátorů. Regulátor tokotvorné složky je tak omezen na $\pm U_{fn(max)}$ a druhý vnitřní regulátor je omezován dynamicky na $\pm U_{mez}$ podle rovnice (6-6), tak aby amplituda výsledného žádaného vektoru napětí nepřesáhla právě $U_{fn(max)}$.

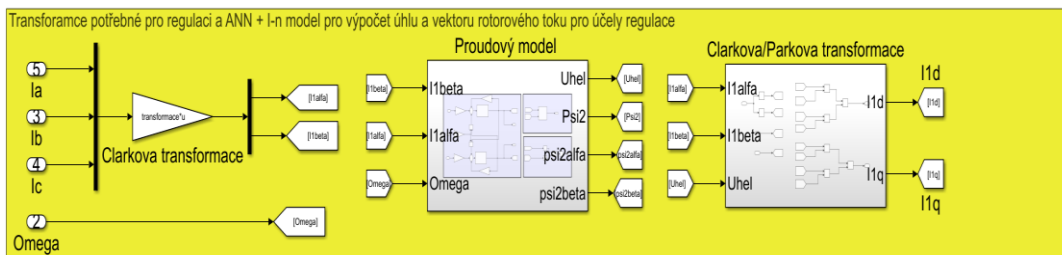
$$U_{mez} = \sqrt{(U_{fn(max)})^2 - (u_{1d})^2} \quad (6-6)$$

Výstup z regulátoru rotorového magnetického toku je ze shora omezen na $1,5i_{1dn}$, kde i_{1dn} je jmenovitá hodnota tokotvorné složky statorového proudu, a ze spodu je pak omezen na 0. Výstup posledního regulátoru, tedy otáčkového, je pak omezen na $\pm i_{1qn}$, kde i_{1qn} je jmenovitá hodnota momentotvorné složky statorového proudu.

Po výstupu z regulátoru vstupují žádané hodnoty napětí u_{1d} a u_{1q} do bloku „Odvazbení“, který slouží k úpravě žádaného napětí, kvůli provázanosti některých rovnic. Odvazbení však není nutné, neboť toto provázání kompenzují regulátory proudu. Nakonec je napětí převedeno pomocí zpětné Parkovy transformace do souřadnicového systému $\alpha\beta$ a následně pomocí zpětné Clarkové transformace do souřadnicového systému abc .

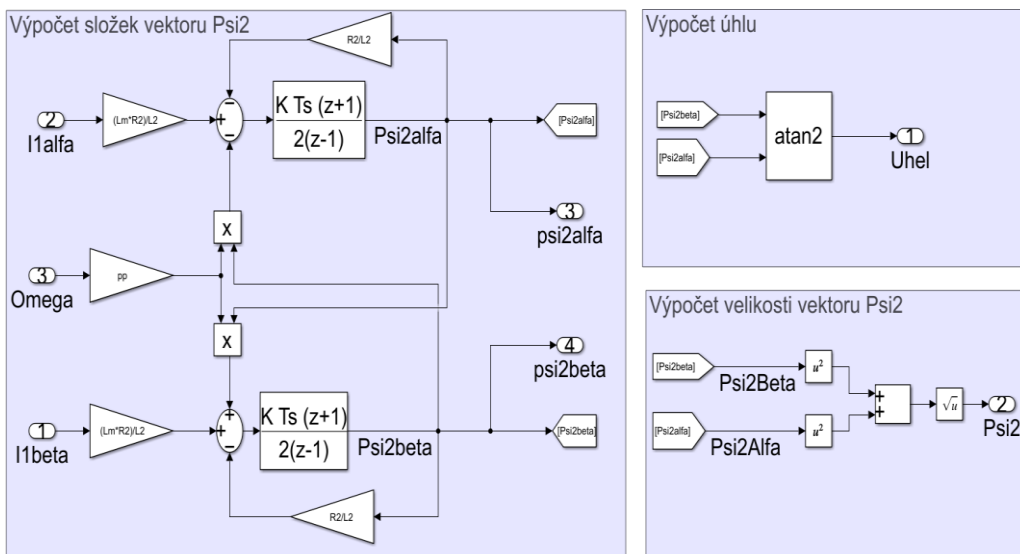
6.3 Model proudového I-n modelu

Stejně jako v předchozí kapitole, je i tato část řešena diskrétně. Proudový model asynchronního motoru, jak bylo popsáno v části 3.2.1, slouží k zjištění transformačního úhlu pro Parkovu transformaci a k výpočtu vektoru rotorového magnetického toku $\vec{\psi}_2$. Tudíž jak je patrné z Obr. 6-10, tak vstupní třífázový satorový proud je pomocí Clarkové transformace převeden do souřadnicového systému $\alpha\beta$ a spolu s elektrickou úhlovou rychlostí rotoru „Omega“ vstupuje to proudového I-n modelu. Výstupní vektor $\vec{\psi}_2$ je posílán do regulátoru a transformační úhel je v dalším bloku použit k transformaci složek $i_{1\alpha}$ a $i_{1\beta}$ na složky i_{1d} a i_{1q} . V tomto tvaru vstupují složky satorového proudu do regulátoru.



Obr. 6-10 Model s Clarkové transformací proudu, proudovým modelem a Parkovou transformací.

Vnitřek samotného bloku „Proudový model“ je pak možné vidět na následujícím obrázku, kde pro integraci jsou použity diskrétní integrátory a je použita lichoběžníková integrační metoda.

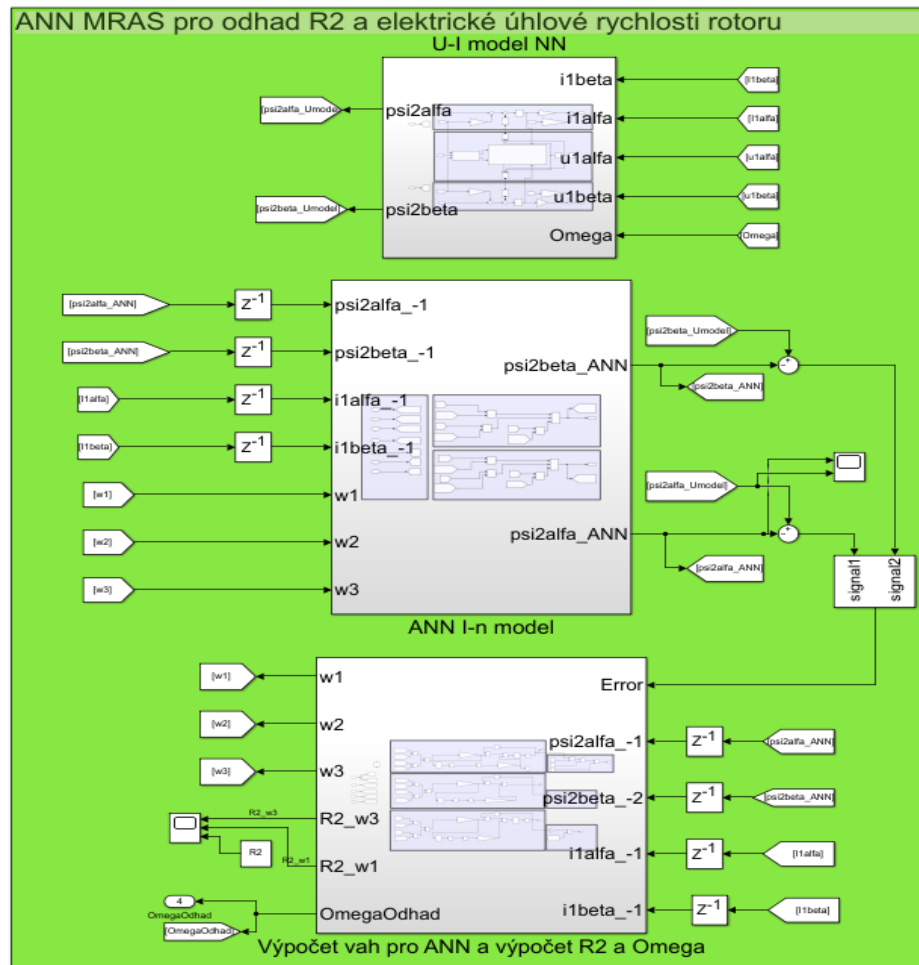


Obr. 6-11 Proudový I-n model s výpočtem transformačního úhlu a modulu vektoru magnetického toku rotoru.

6.4 Model pro odhad rotorového odporu a otáček pomocí kombinace MRAS a ANN

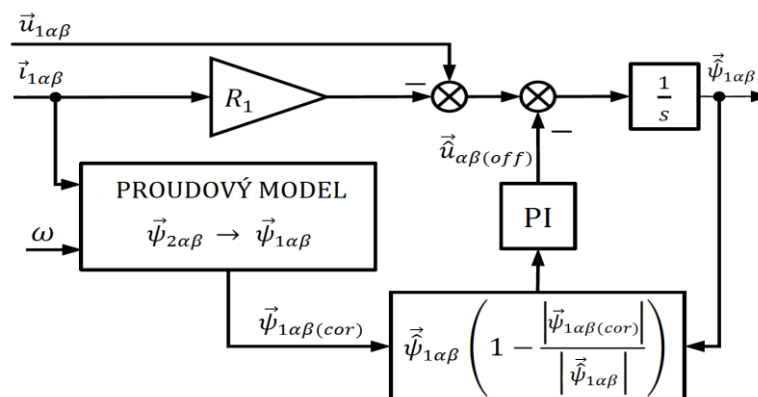
V této kapitole je aplikován model MRAS s adaptivní umělou neuronovou sítí pro odhad rotorového odporu a elektrické úhlové rychlosti diskutovaný v kapitole 5.1. Tato část je také řešena diskrétně a na Obr. 6-2 se jedná o zelenou oblast, která je detailněji vidět na Obr. 6-12. Skládá se především ze tří bloků, kterými jsou

napětový U-I model sloužící jako referenční, model neuronové sítě sloužící jako adaptivní, a nakonec blok pro aplikaci algoritmu zpětného šíření chyby a výpočtu odhadovaných parametrů.



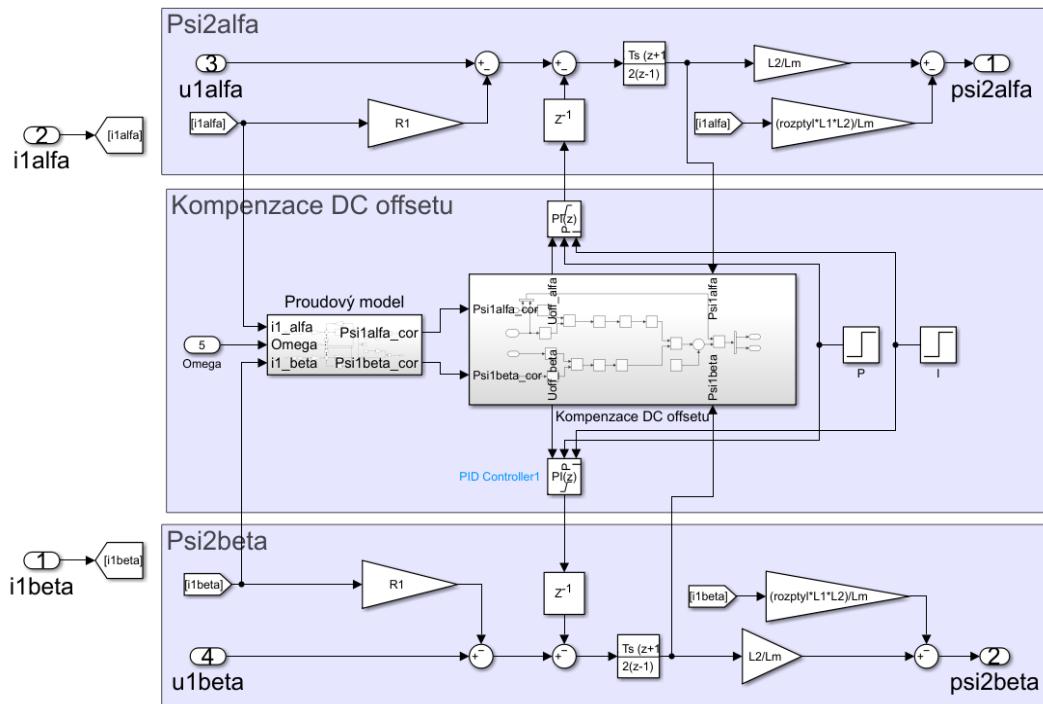
Obr. 6-12 Část spínaného bloku pro odhad R_2 a ω pomocí MRAS s adaptivní ANN.

Při aplikaci napětového U-I modelu pomocí soustavy rovnic (3-9) se výrazně projevoval stejnosměrný offset na výstupu integrátoru, z toho důvodu byl doplněn o další adaptivní model, který tento offset kompenzuje. Detail vnitřku tohoto modelu je na Obr. 6-13.



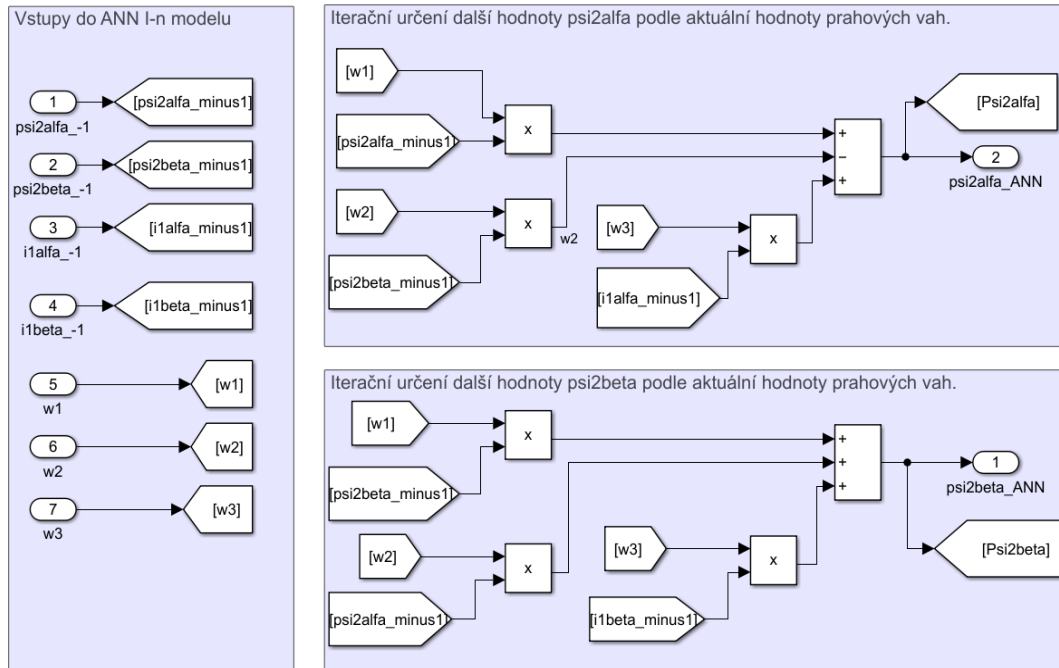
Obr. 6-13 Schéma pro kompenzaci DC offsetu.

Kompenzace offsetu je upravována pomocí PI regulátorů, jejichž konstanty jsou na začátku simulace $r_0 = 20$ a $r_{-1} = 20$ a v čase $t = 0,8$ s dojde k přepnutí na hodnoty $r_0 = 600$ a $r_{-1} = 5000$. Regulátory sice jsou schopny offset dobře eliminovat, avšak již dochází k drobným nepřesnostem, které především s větším krokem simulace mají vliv na odhad neuronové sítě. Integrátory musí být opět použity diskrétní a je použita lichoběžníková integrační metoda. Schéma zapojení samotného bloku s U-I modelem a kompenzací DC offsetu je na následujícím obrázku.

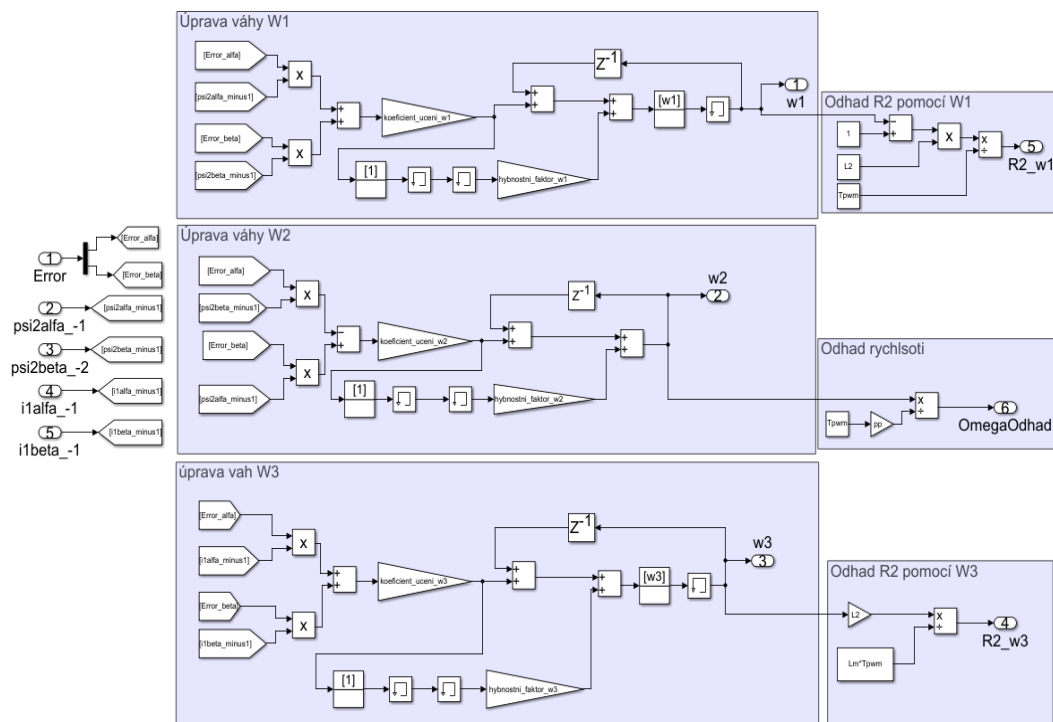


Obr. 6-14 Napěťový U-I model s kompenzací DC offsetu.

Blok s umělou neuronovou sítí je čistě aplikací rovnice (5-7). Podobně i úprava vah a odhad statorového odporu R_2 a elektrické úhlové rychlosti rotoru ω jsou dány aplikací rovnic (5-13) až (5-18). Detail obou těchto bloků je postupně na Obr. 6-15 a Obr. 6-16.



Obr. 6-15 Proudový I-n model pomocí ANN.



Obr. 6-16 Úprava vah W_1 , W_2 a W_3 podle algoritmu zpětného šíření a odhad R_2 a ω .

Kvůli problémům s algebraickými smyčkami jsou v celém spínaném bloku na některých místech signály úmyslně zpožděny o jeden výpočetní krok. Kvůli těmto zpožděním začíná spínaný blok počítat až při druhé iteraci. S periodou spínání $100 \mu\text{s}$ jsou tak první hodnoty ve spínaném bloku počítány až v čase $t = 0,0001$. To obecně nezpůsobuje problémy s modelací, avšak v bloku pro úpravu vah, který je na posledním obrázku, je potřeba, aby váhy měly nějakou počáteční hodnotu. K tomu

je použit blok „IC“ spolu s blokem „Memory“. První zmíněný blok slouží pouze k zadání počáteční hodnoty při první iteraci, ta se však nespočítá a při druhé již tento blok nemá vliv. Blok „Memory“ si tuto hodnoty při první iteraci zapamatuje a zpozdí ji o právě jeden krok a počáteční hodnota se tak již projeví ve druhém výpočetním kroku tak, jak má. Počáteční hodnoty pro dané prahové váhy a jmenovité elektrické hodnoty motoru pak jsou

$$W_1 = 1 - \frac{R_2 T_s}{L_2} = 1 - \frac{0,225 * 0,0001}{0,08477} = 0,9997, \quad (6-7)$$

$$W_3 = \frac{R_2 L_m T_s}{L_2} = \frac{0,225 * 0,0825 * 0,0001}{0,08477} = 0,0000219. \quad (6-8)$$

Váha W_2 má počáteční hodnotu nulovou, protože rozběh motoru je simulován z klidu.

Úprava všech tří vah probíhá podle rovnic (5-13) až (5-15), přičemž koeficienty učení i momentové faktory jsou konstantní. V kapitole 5.1 byla ještě uvedena varianta pro průběžnou úpravu koeficientu učení, jak již ale bylo zmíněno, tak tato možnost se neukázala jako efektivnější. Pro správnou funkci této umělé neuronové sítě je tak nutné především dobře nastavit právě koeficienty učení a momentové faktory. Koeficienty učení se obecně dávají menší než 1. V této práci byly nastavovány ručně a ukázalo se, že nejlepších výsledků je dosahováno, pokud pro W_1 a W_2 jsou koeficienty učení v rozsahu zhruba 0,001 – 0,00001 a momentové faktory jsou pak zhruba o řád menší, tedy 0,01 – 0,0001. Pro váhu W_3 jsou potom hodnoty obou koeficientů výrazně nižší a pohybují se v rozsahu 0,000001 – 0,0000000001 pro koeficient učení a 0,00001 – 0,000000001 pro momentový faktor. Záleží ale vždy na kombinaci všech tří koeficientů dohromady.

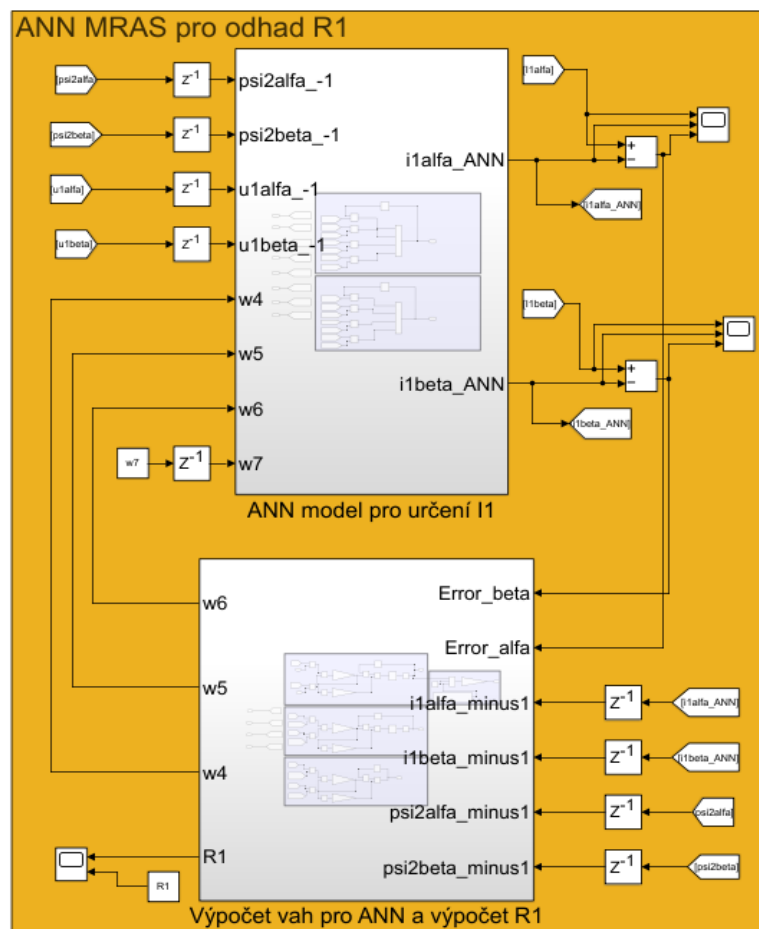
Hodnoty pro W_1 a W_3 se mezi sebou navzájem ovlivňují a pokud je například koeficient učení u váhy W_1 navýšen, je dobré naopak adekvátně snížit koeficient učení u W_3 . Obdobně je tomu pro momentový faktor. Jako nejlepší se ukázala kombinace s vyšší hodnotou koeficientů v řádu tisícín pro W_1 . To zajišťuje velice dobrý hrubý odhad. Pro co nejmenší kvadratickou chybu je ale potřeba ještě správně nakombinovat koeficienty pro zbylé dvě váhy. Změny hodnot koeficientů pro W_2 sice ovlivňují především kvalitu odhadu elektrické úhlové rychlosti a nemají takový vliv na odhad složek rotorového toku jako u předchozích dvou vah, ale právě spolu s koeficienty pro W_3 jsou důležité pro konečnou přesnost odhadu. Nejlepších výsledků pro odhad magnetického toku je sice docíleno pro koeficienty váhy W_1 v řádu desetín, tedy o řád vyšší než uvedené, ale pro takovéto hodnoty pak dochází k postupnému nárůstu hodnoty váhy W_1 do tak vysoké hodnoty, že dojde k selhání výpočtu.

Prakticky se zároveň osvědčilo použití zmíněného momentového faktoru μ , neboť se stávalo, že síť se ustálila na nějakém průběhu, který však neodpovídal žádané hodnotě. V těchto případech se po zvýšení momentového faktoru síť již neustalovala na nechtěné hodnotě, ale po nějaké době se ustálila na hodnotě žádané. Konkrétní hodnoty koeficientů učení a momentových faktorů jsou uvedeny v kapitole Kapitola 7:. Je to z toho důvodu, že pro některé kombinace je odhad

rotorového odporu lepší na úkor odhadu elektrické úhlové rychlosti a naopak. Zároveň má na odhad velký vliv i rychlost rotoru a zatížení stroje.

6.5 Model pro odhad statorového odporu pomocí kombinace MRAS a ANN

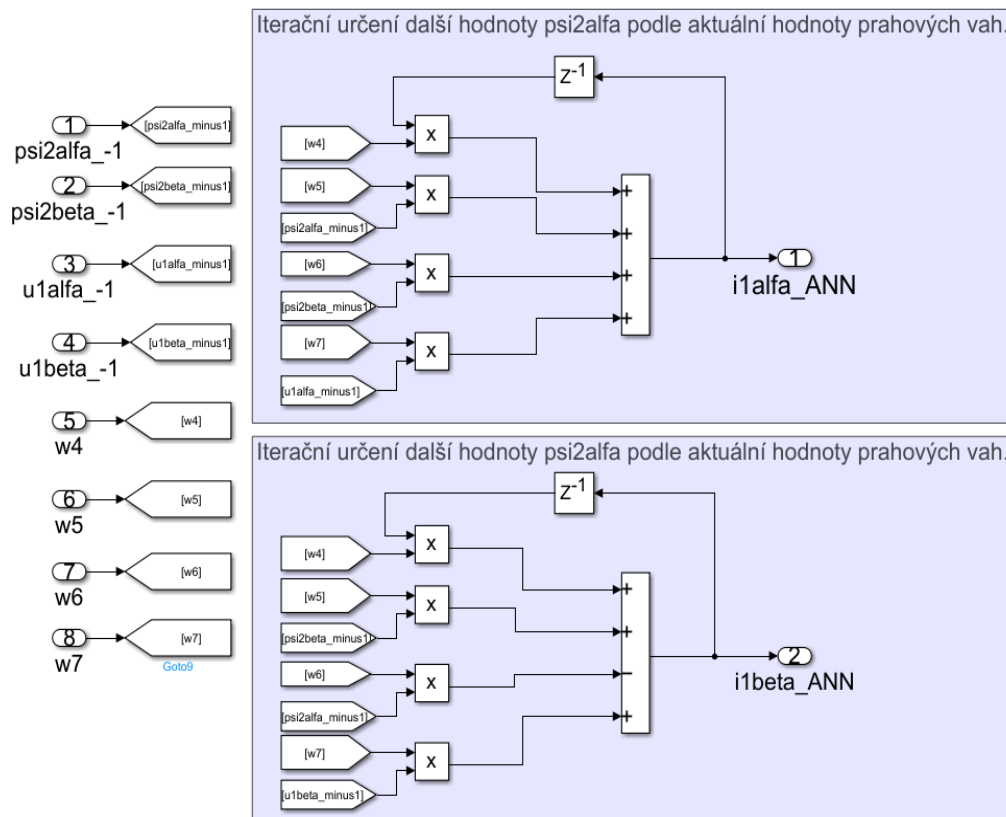
Stejně jako v předešlé kapitole, je i zde aplikován model MRAS spolu s adaptivní umělou neuronovou sítí a jedná se o poslední část, která je řešena diskretně a její detail je na Obr. 6-17. Teoretický základ je blíže popsán v kapitole 5.2. Stejně jako v části pro odhad rotorového odporu je zde model umělé neuronové sítě a blok pro aplikaci algoritmu zpětného šíření chyby a výpočtu statorového odporu. Není zde blok napětového U-I modelu, neboť jako referenční veličina zde slouží složky statorového proudu v souřadnicovém systému $\alpha\beta$, které jsou získávány transformací statorového proudu pomocí Clarkové transformace.



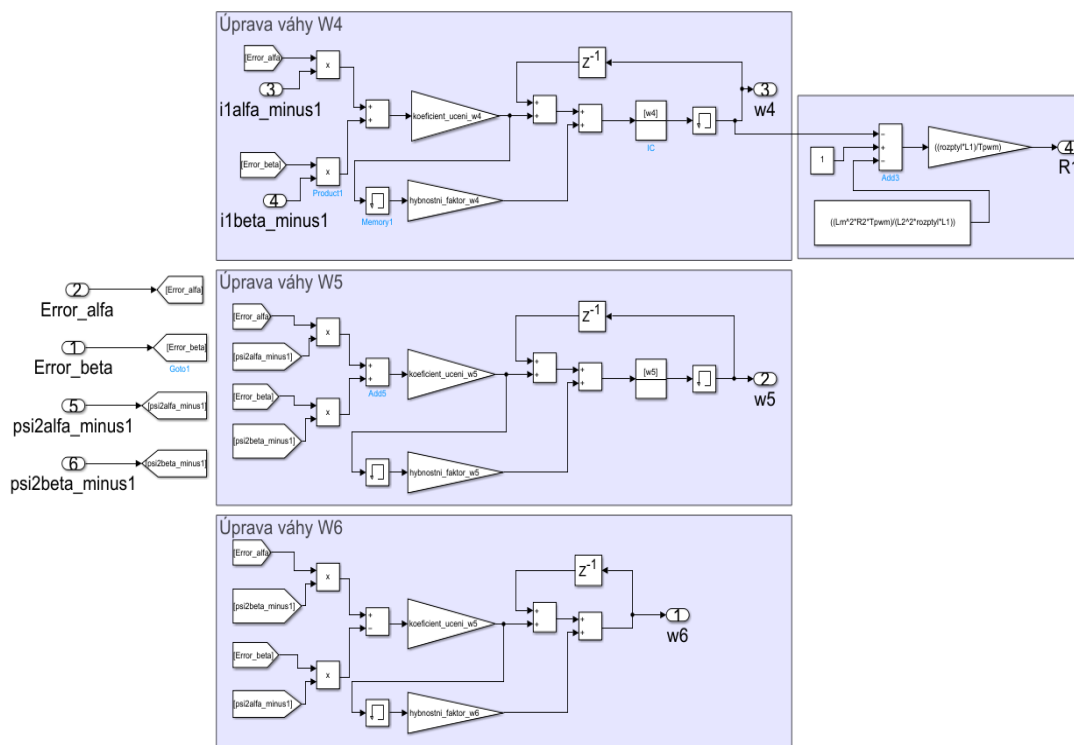
Obr. 6-17 Část spínaného bloku pro odhad R_1 pomocí MRAS a ANN

Umělá neuronová síť sloužící jako adaptivní část pro MRAS je čistě aplikací rovnice (5-7). Jednotlivé složky statorového odporu v příslušném souřadnicovém systému jsou na výstupu bloku porovnávány s žádanými hodnotami. Rozdíl těchto hodnot pak vstupuje do bloku pro úpravu vah a výpočtu odhadovaného statorového odporu. Pro dobrou kvalitu odhadu je opět nutné najít správnou kombinaci koeficientů pro jednotlivé váhy W_4 , W_5 a W_6 . Koeficienty učení se u všech vah mohli pohybovat v rozsahu 0,001– 0,00000001 a záleželo především na jejich vzájemné kombinaci. U tohoto odhadu však bylo náročnější docílit alespoň uspokojivé kvality

ve srovnání s odhadem složek magnetického toku rotoru. Podrobné průběhy i s hodnotami pro jednotlivé váhy jsou pak v kapitole 7. Detaily obou bloků v této části jsou na Obr. 6-18 a Obr. 6-19.



Obr. 6-18 Model umělé neuronové sítě pro odhad složek $i_{1\alpha}$ a $i_{1\beta}$.



Obr. 6-19 Úprava vah W_4 , W_5 a W_6 podle algoritmu zpětného šíření a odhad R_1 .

KAPITOLA 7: SIMULOVANÉ PRŮBĚHY MODELU

V této kapitole jsou prezentovány naměřené průběhy pro různé varianty žádaných otáček, zátěžného momentu a změny statorového a rotorového odporu. Obecně bylo možné pro základní ladění modelu použít zmíněný řešič „ode4“ (Runge-Kutta), pro přesnější průběhy je však nutné použít alespoň „ode8“ (Dormand-Prince) nebo snížit krok řešiče o jeden řád, tedy na 0,5 μ s. Při snížení kroku, jak již bylo zmíněno v kapitole Kapitola 6., je problém s délkou simulace, proto je řešič „ode8“ rozumná varianta, která výrazně zlepšuje výpočet a zachová rozumnou kvalitu simulace. Výběr řešiče a kroku vnáší do celého modelu veliký vliv a v optimálním případě by bylo pracovat stále s menším krokem a případně najednou i s přesnějším řešičem. Toho bohužel nebylo možné dosáhnout, neboť použitý přenosný počítač nemá dostatečný výpočetní výkon. V následujících kapitolách jsou tedy vybrané průběhy některých veličin s uvedeným vybraným řešičem a s konkrétním nastavením koeficientů pro váhy neuronové sítě.

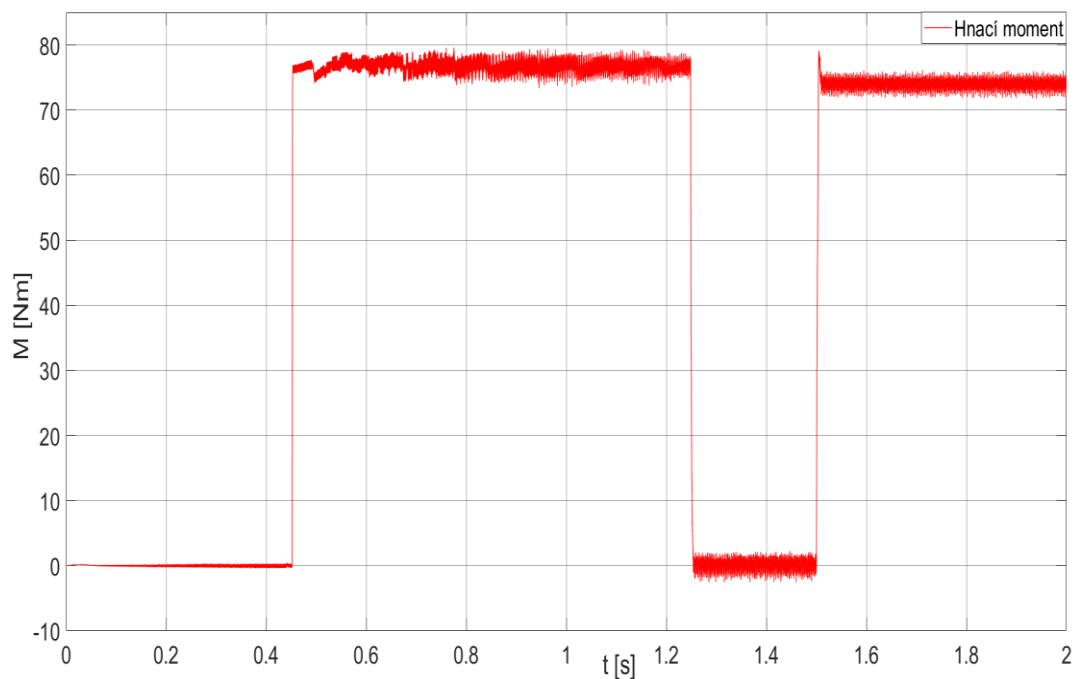
Jelikož řízení motoru je realizováno pomocí vektorového řízení, kdy je asynchronní motor řízen jako motor stejnosměrný s cizím buzením, tak je před rozběhem motor nejdříve nabuzen. To je realizováno tím způsobem, že otáčky jsou žádané až potom, co velikost magnetického toku dosáhne jmenovité hodnoty, která je přibližně 0.9 Wb.

Celkový model pohonu pro jednotlivé simulace se skládá z modelu AM a střídače z kapitoly 6.1, z modelu regulátoru s odvazbením z kapitoly 6.2 a z proudového I-n modelu spolu s transformacemi, což bylo diskutováno v kapitole 6.3. Samotná regulační struktura, která řídí spínání měniče, a tedy i samotného pohonu, je tvořena zmíněným regulátorem a proudovým I-n modelem. Právě proudový model slouží k výpočtu referenční hodnoty magnetického rotorového toku pro tokotvorný regulátor a ze složek tohoto vektoru zároveň počítá transformační úhel pro Parkovu transformaci. Výstupem transformace jsou tokotvorná a momentotvorná složka statorového proudu, jenž jsou posílány do regulátoru. V ideálním případě by pak tato základní struktura vektorového řízení měla být doplněna o odhad rotorového a statorového odporu a případně i elektrické úhlové rychlosti. Odhadovaný rotorový odpor by měl být posílán právě do proudového I-n modelu a dynamicky tento odpor upravovat podle toho, jak se změní skutečná hodnota rotorového odporu v modelu AM. To se v základním modelu pro simulace neděje a proudový model počítá s konstantním rotorovým odporem o jmenovité hodnotě. Na stejném principu by měl fungovat i odhad statorového odporu, avšak ten upravuje jeho hodnotu v napěťovém U-I modelu, sloužícího jako referenční model MRAS pro odhad rotorového odporu. Odhad otáček by pak znamenal, že metoda je v optimálním případě schopna jejich bezsenzorového odhadu.

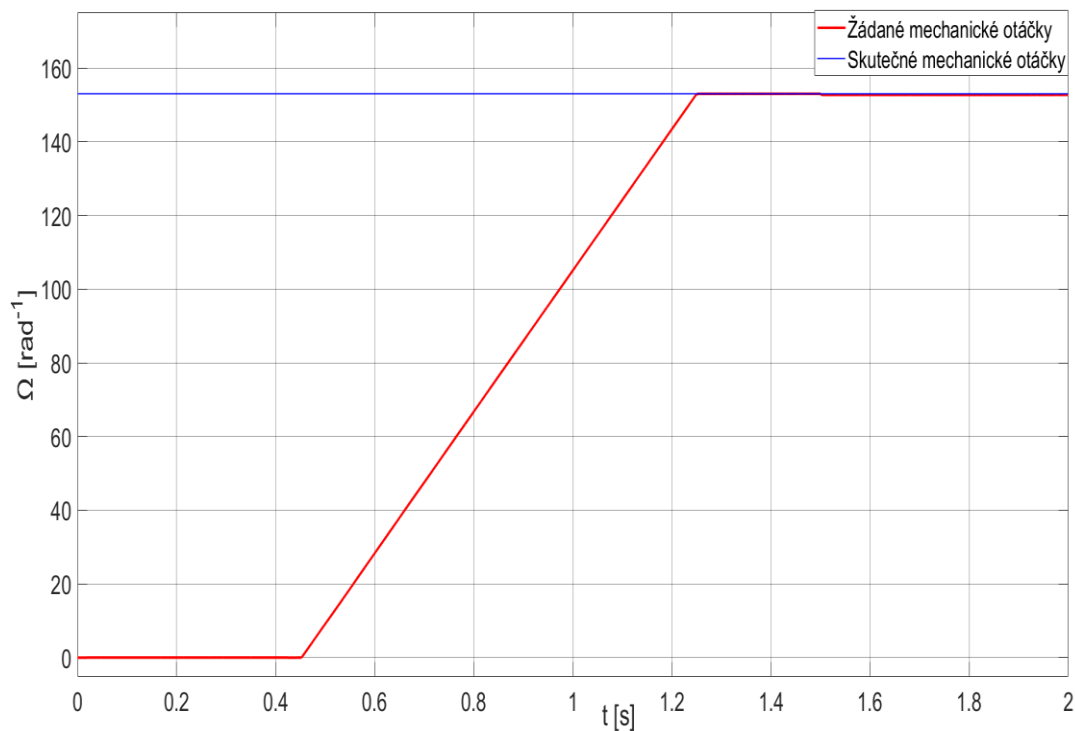
7.1 Základní průběhy

Pro úplnost jsou zde uvedeny průběhy hnacího momentu, otáček, proudu rotoru a magnetického toku rotoru při rozběhu motoru na žádané otáčky a při následném zatížení. Jak bylo uvedeno na konci předchozí kapitoly, tak nejdříve dochází k nabuzení na jmenovitou hodnotu a až potom k rozběhu. Žádané jmenovité otáčky

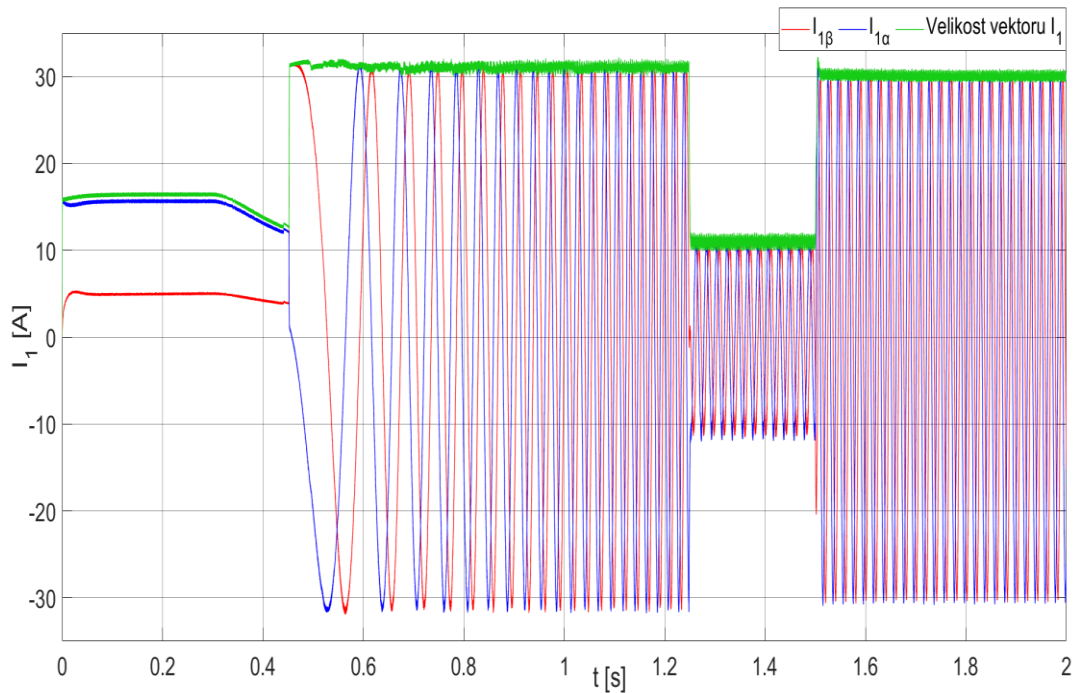
jsou 153 rad^{-1} a v čase $t = 1,5 \text{ s}$ dojde k zatížení jmenovitým momentem 74 Nm . Průběhy zmíněných veličin jsou na Obr. 7-1 – Obr. 7-4.



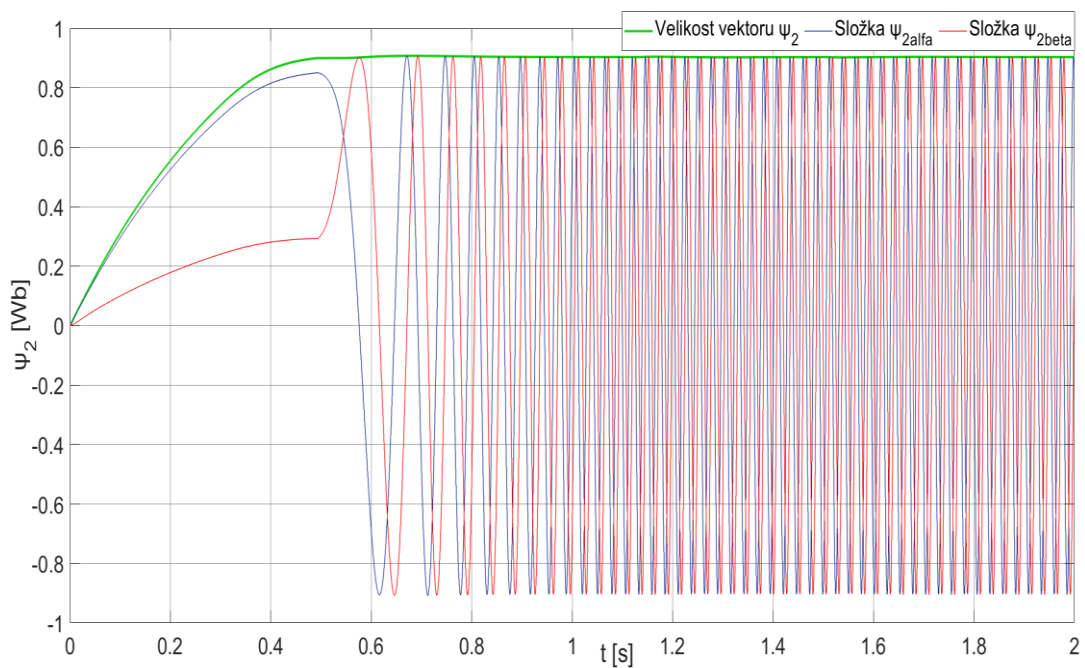
Obr. 7-1 Průběh hnacího momentu při rozběhu a následném zatížení v čase $t=1,5 \text{ s}$.



Obr. 7-2 Průběh skutečných a žádaných otáček při rozběhu a následném zatížení v čase $t=1,5 \text{ s}$.



Obr. 7-3 Průběh složek a modulu vektoru statorového proudu v souřadnicovém sstému $\alpha\beta$.



Obr. 7-4 Průběh složek a modulu vektoru magnetického toku rotoru v souřadnicovém sstému $\alpha\beta$.

Především z průběhu otáček je patrné, že regulátory jsou nastaveny tak, aby nedocházelo k překmitu a zároveň při jmenovitém zatížení dojde pouze k malému poklesu otáček. Na průběhu hnacího momentu a statorového proudu je patrný rozdíl kdy je motor zatížen, ať už rozběhem nebo zátěží, a kdy běží naprázdno. Zároveň je patrné, že i při rozběhu dosahuje maximální hodnota proudu zhruba 31 A. Z posledního průběhu na Obr. 7-4 je vidět, že motor pracuje s konstantním jmenovitou hodnotou buzení.

7.2 Průběhy odhadu rotorového odporu s potlačenou vahou W_3

Jedním z hlavních cílů je právě odhad rotorového odporu z důvodu korekce jeho hodnoty pro účely zlepšení přesnosti vektorového řízení. Přesnost odhadu je ovšem ovlivněn několika faktory. Nepřesnost spojená s výběrem řešiče a jeho kroku již byla zmíněna, avšak velký vliv má kompenzace DC offsetu u referenčního napětového U-I modelu. Především je důležité nastavení PI regulátorů pro kompenzaci, ale i zde je znatelný vliv vybraného řešiče a kroku. Nepřesnost odhadnutých složek magnetického rotorového toku se pak přenáší do odhadu umělé neuronové sítě.

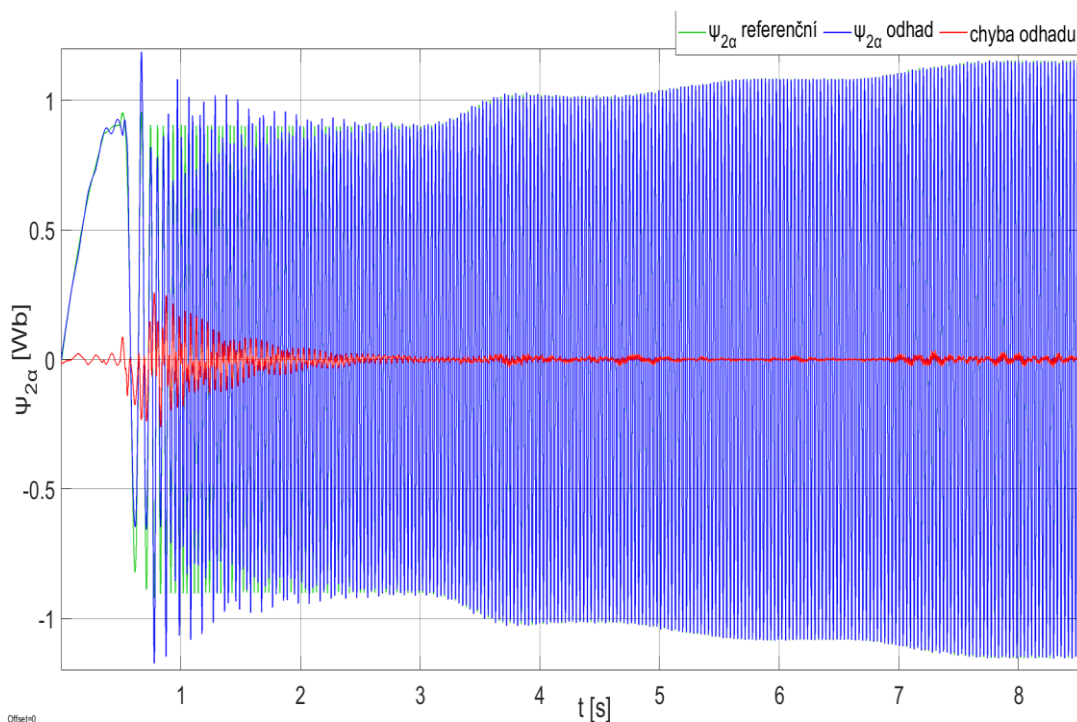
Hlavní vliv je dán ale především provozními podmínkami, tedy požadovanou rychlostí, zátěžným momentem, a především změnou rotorového, případně satorového odporu. Z tohoto důvodu bylo nasimulováno několik scénářů pro demonstraci možné vlivu jednotlivých zmíněných parametrů na kvalitu odhadu. Nastavení koeficientů učení a momentových faktorů je v Tab. 7-1. Z té je vidět, že jak koeficient učení, tak momentový faktor pro váhu W_3 jsou poměrně malé a úprava této váhy je tak značně potlačena. Hlavní podíl na odhadu tak mají zbylé dvě váhy.

Tab. 7-1 Nastavení koeficientů učení a momentových faktorů pro odhad R_2 s potlačenou vahou W_3 .

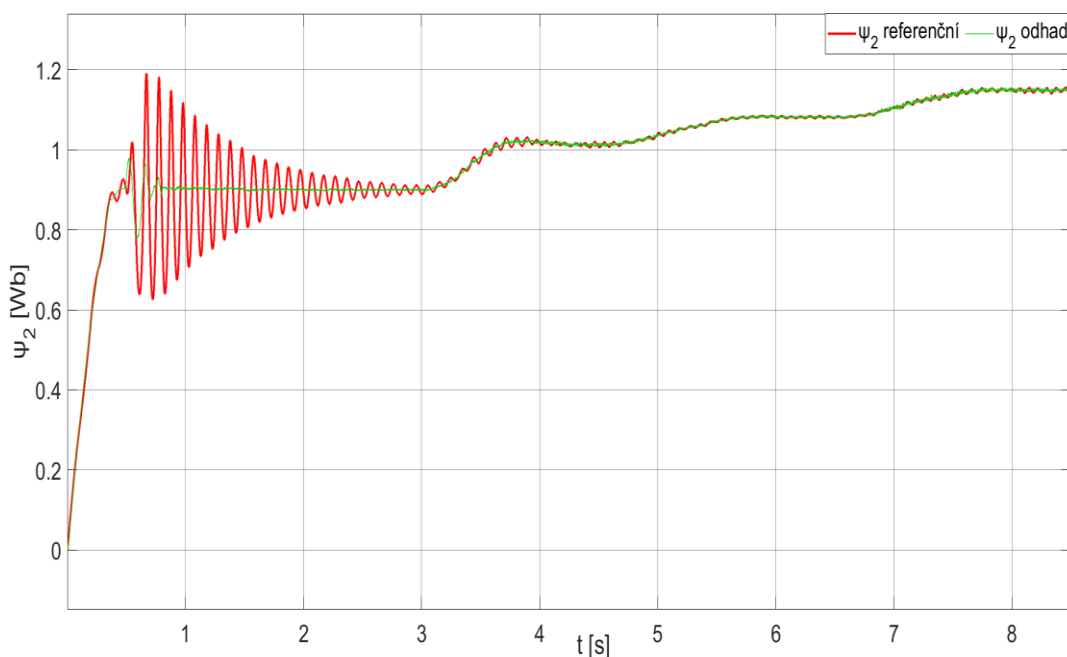
	Koeficient učení η	Momentový faktor μ
W_1	0,0001	0,005
W_2	0,0005	0,005
W_3	0,000000000001	0,00000000001

Průběh simulace je takový, že je motor rozeběhnut na žádané otáčky $100 \text{ rad}\cdot\text{s}^{-1}$, přibližně 955 min^{-1} a v čase $t = 1,5$ sekundy je motor zatížen momentem 65 Nm . S tímto zatížením běží motor až do $t = 3$ sekundy, aby měla neuronová síť čas se ustálit a poté je postupně zvyšována hodnota rotorového odporu v modelu asynchronního motoru o 30, 60 a 100 % oproti jmenovité hodnotě. Odpor je tak navýšen až o dvojnásobek původní hodnoty. Zvolené otáčky jsou nižší než jmenovité, neboť při vyšších otáčkách není motor schopen udržet požadovanou rychlost i při dvojnásobné hodnotě odporu. Díky tomu je možné sledovat kvalitu odhadu.

Důležitým faktorem pro změnu vah, je chyba odhadu složek rotorového magnetického toku oproti referenčnímu modelu. Průběh složky $\psi_{2\alpha}$ pro zmíněný průběh simulace je na Obr. 7-5, na Obr. 7-6 je pak velikost vektoru $\overrightarrow{\psi_{2\alpha\beta}}$. Průběh složky $\psi_{2\beta}$ je kvalitou odhadu stejně dobrý jako složka $\psi_{2\alpha}$, a proto zde není prezentován.



Obr. 7-5 Průběh $\psi_{2\alpha}$ pro referenční $U-I$ a adaptivní ANN model a chyba odhadu daná jejich rozdílem.

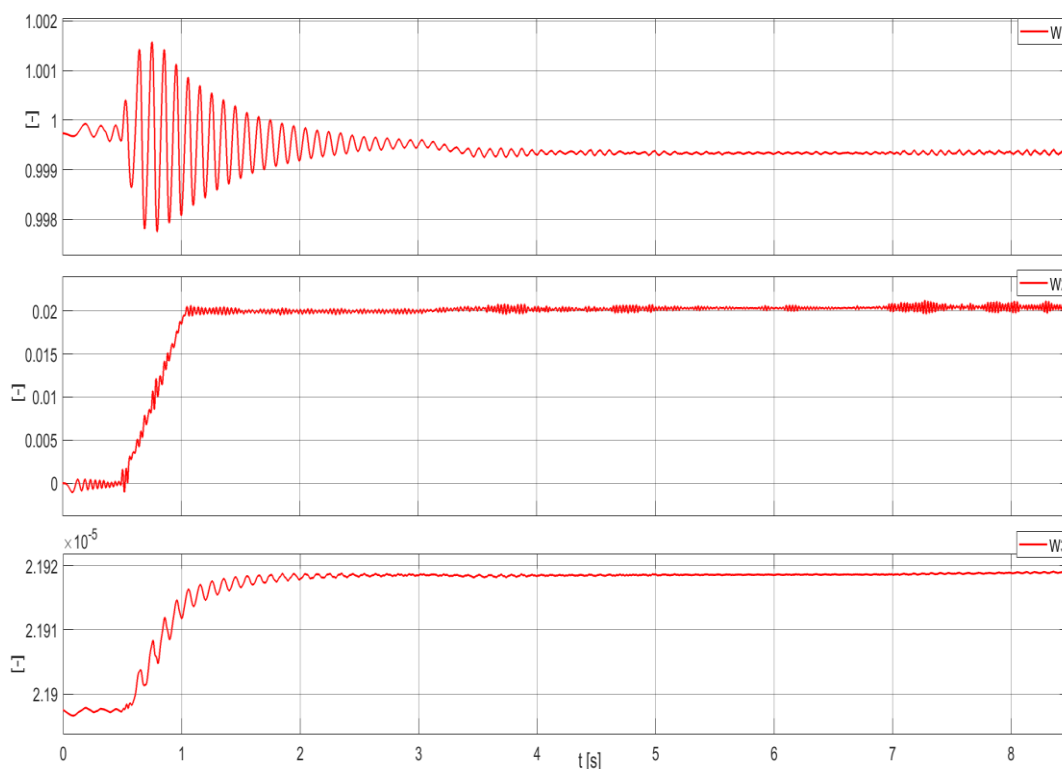


Obr. 7-6 Průběh velikosti ψ_2 pro referenční $U-I$ a adaptivní ANN model.

Na prvním obrázku je referenční hodnota vidět pouze v prvních zhruba dvou sekundách simulace. Tomu odpovídá převážně rozběh motoru, kdy se síť postupně ustaluje až se stabilizuje na žádané hodnotě a prakticky kopíruje referenční hodnotu. To, jestli je motor zatížený či nikoliv, nemá vliv na rychlost ustálení.

Pro přehlednost je lepší sledovat červený průběh, znázorňující chybu mezi požadovanou a skutečnou hodnotou. Při rozběhu je chyba větší a dosahuje maximálně 0.26 Wb, avšak důležitý je hlavně průběh při zatížení. Od času $t = 1,5$

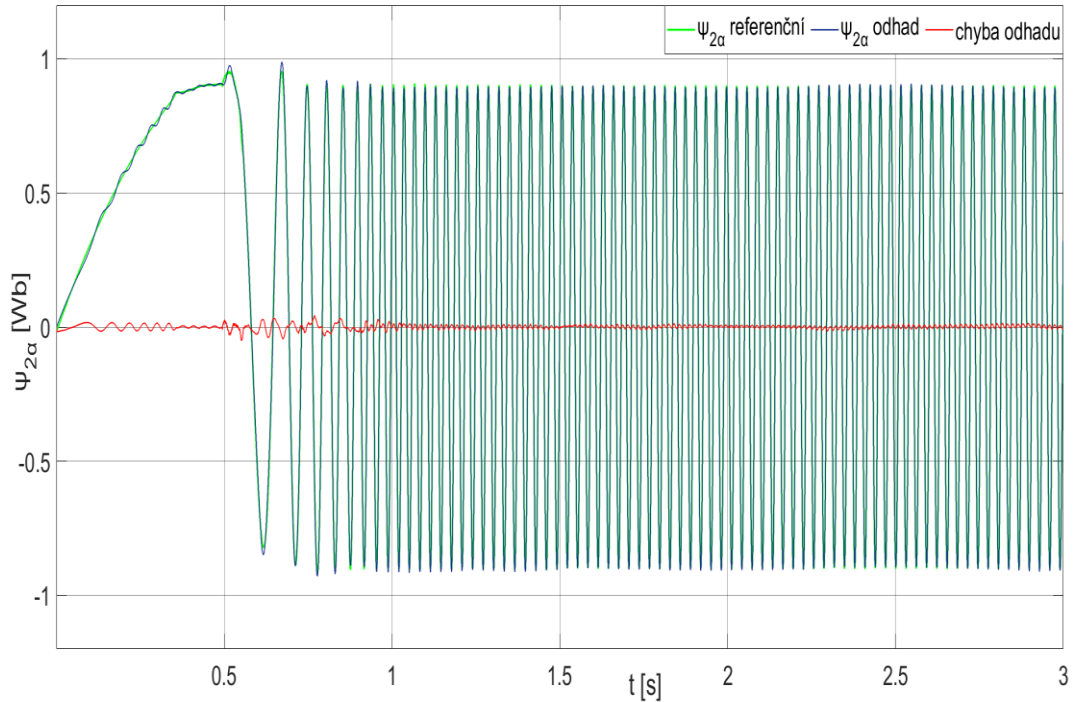
kdy je již možné považovat síť za ustálenou, se chyba pohybuje okolo 0,003 Wb a maximálně pak 0,03 Wb. To je při jmenovité hodnotě 0,9 Wb chyba, která nemá na řízení takového motoru vliv. Velikost magnetického toku navíc roste z důvodu zvětšujícího se odporu. To má pouze malý vliv na odhad, neboť je vidět, že chyba se ani při dvojnásobku hodnoty odporu výrazněji nezvětšuje. Na druhém obrázku je pak z průběhu velikosti vektoru magnetického toku vidět, jak po zatížení a ustálení již odhad neuronové sítě pouze s malým zvlněním kopíruje žádaný průběh. Úprava vah k dosažení prezentovaného odhadu je na Obr. 7-7.



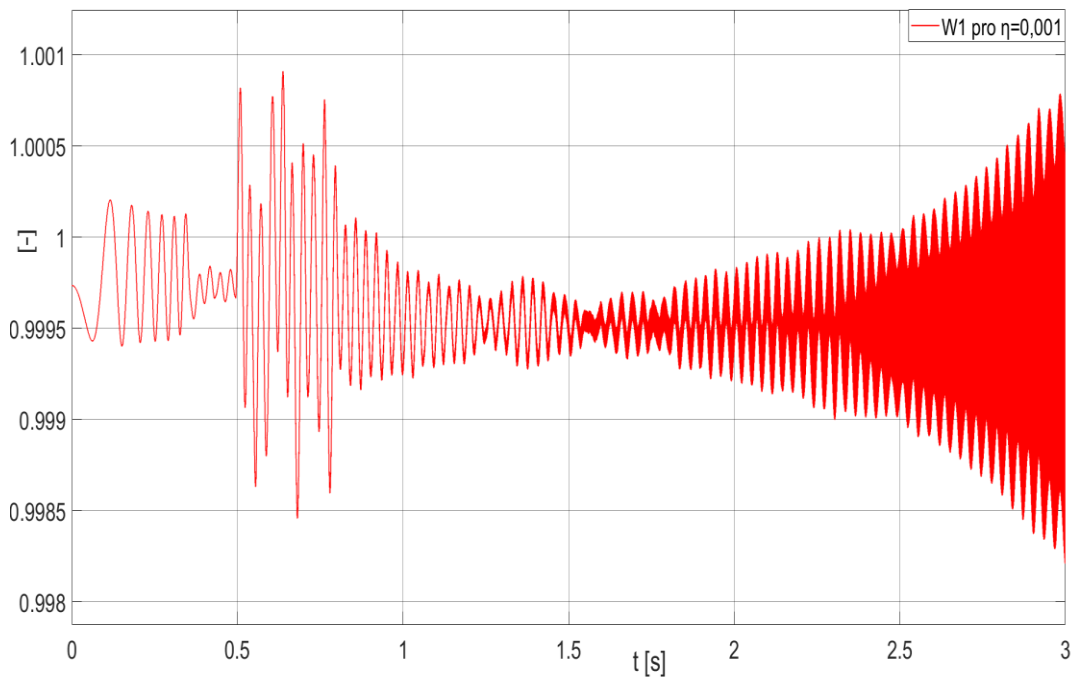
Obr. 7-7 Průběh úpravy vah W_1 , W_2 a W_3 .

Je patrné, že nejvíce se změní hodnota váhy W_2 , což je dáno především nulovou počáteční hodnotou. Zároveň je patrné, že váhy W_2 a W_3 se ustálí poměrně rychle v porovnání s průběhem na Obr. 7-5, kdy odhad celé neuronové sítě ještě vykazuje poměrně velkou chybu oproti průběhu referenčnímu. Ustálení na žádanou hodnotou je tak dáno hlavně úpravou zbylé váhy W_1 , která se ustaluje výrazně déle.

Pro lepší odhad už při rozběhu motoru je možné zvýšit hodnotu koeficientu učení η_1 . To má za následek zlepšení přesnosti odhadu, avšak začne docházet k již zmiňovanému stálému nárůstu hodnoty váhy W_1 a tedy i odhadu rotorového odporu. Toto chování je patrné z průběhu složky $\psi_{2\alpha}$ na Obr. 7-8 a váhy W_1 na Obr. 7-9 při zvýšení koeficientu učení z Tab. 7-1 na 0,001. Ostatní hodnoty z této tabulky zůstaly nezměněny. Průběh simulace byl zkrácen na 3 sekundy, aby bylo demonstrováno chování sítě především při rozběhu. Ke stejnému narůstání váhy, ale i zlepšení přesnosti, dochází i v opačném případě, kdy je naopak potlačena váha W_1 a koeficient učení η_3 váhy W_3 je příliš veliký.



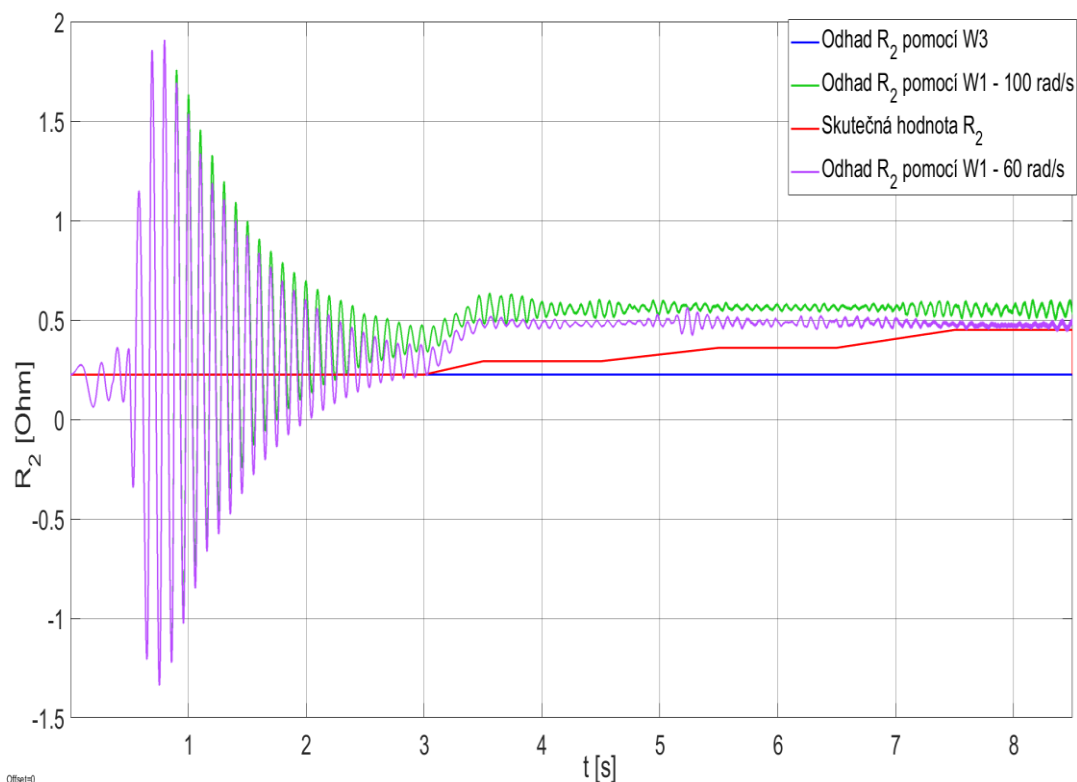
Obr. 7-8 Průběh referenční a odhadnuté složky $\psi_{2\alpha}$ pro koeficient učení $\eta_1 = 0,001$.



Obr. 7-9 Průběh W_1 s koeficientem učení $\eta_1 = 0,001$.

Na Obr. 7-8 referenční a odhadnutý průběh téměř splývají a chyba odhadu je výrazně menší než na Obr. 7-5. Při pohledu na průběh změny váhy W_1 na Obr. 7-9 je však potřeba si uvědomit, že je především nutné sledovat průběhy vah W_1 případně W_3 , neboť z nich je odhadována hodnota R_2 . Neplatí tak přímá úměra, že s přesnějším odhadem složek rotorového magnetického toku je vždy i lepší odhad rotorového odporu. To je patrné i z průběhu odhadu R_2 na Obr. 7-11, který je počítán právě z váhy W_1 na Obr. 7-9 při vyšším koeficientu učení. Na Obr. 7-10 je ovšem

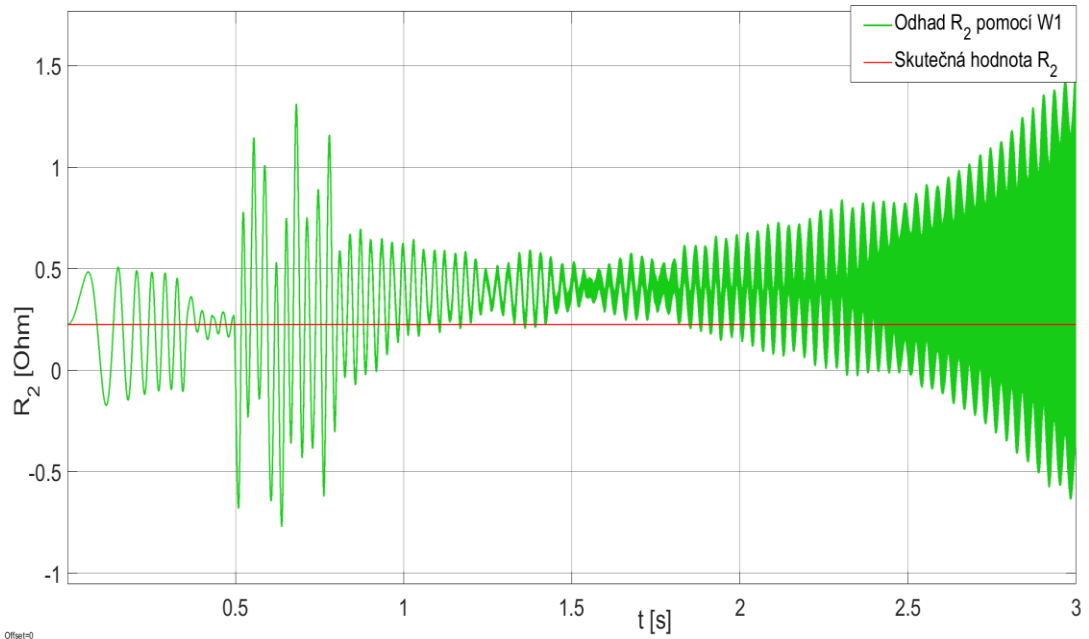
nejdříve k vidění odhad R_2 , vypočítaný z průběhů vah W_1 a W_3 na Obr. 7-7. Tedy pro původní odhad s nižší hodnotou koeficientu učení η_1 .



Obr. 7-10 Průběh odhadu R_2 pomocí W_1 a W_3 při postupném navýšení odporu o 30, 60 a 100 %.

Červenou barvou je označena skutečná hodnota rotorového odporu, která se v modelu asynchronního motoru postupně zvyšuje z původní štítkové hodnoty o 30 %, 60 % a na konci až o 100 %. Modrou barvou je označen odhad tohoto odporu pomocí váhy W_3 , který na změnu odporu vůbec nereaguje. To je dáno celkovým potlačením W_3 pomocí nízkých hodnot η_3 a μ_3 . Díky tomu je dosaženo toho, že odhad pomocí W_1 se pohybuje „poměrně“ blízko skutečné hodnotě. Pokud v čase $t = 3$ s se již považuje odhad za ustálený předtím, než se začne zvyšovat R_2 , tak odhadnutá hodnota je v důsledku zvlnění přibližně 0,4 Ohm. To je ovšem téměř dvojnásobná hodnota oproti skutečné, avšak je to nejlepší odhad, kterého bylo pro různá nastavení dosaženo. Po následném navýšení rotorového odporu o 30 %, je sice vidět, že i odhadnutá hodnota se zvýší, avšak ustálí se kolem hodnoty 0,56 Ohm a na další zvýšení odporu už nereaguje.

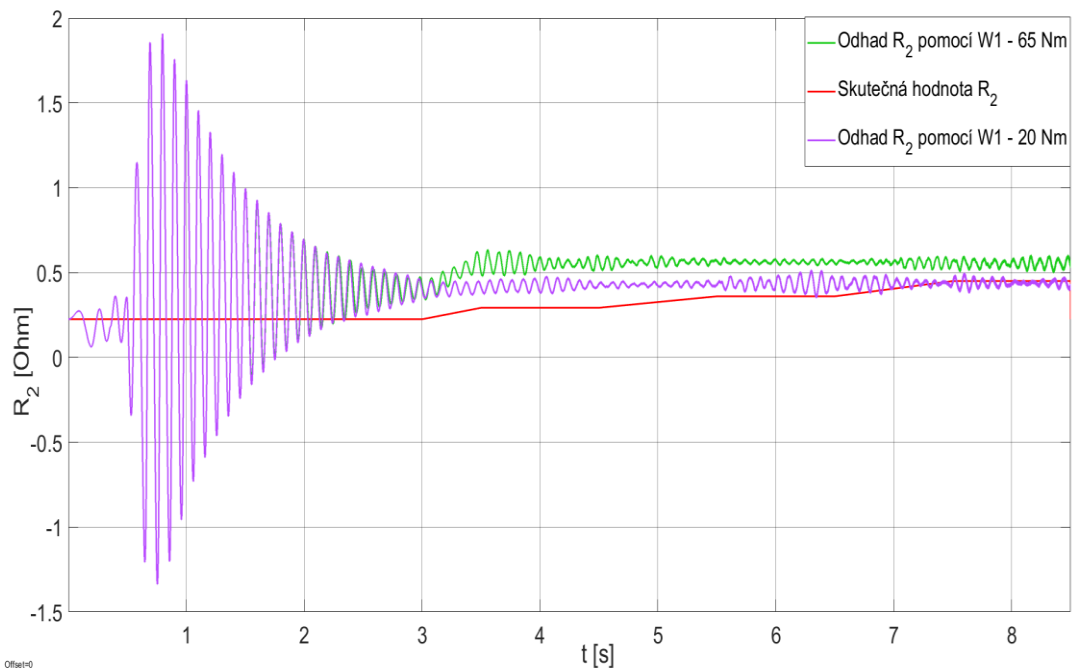
Na Obr. 7-10 je možné vidět ještě fialový průběh. Ten se od zeleného průběhu liší pouze žádanou rychlostí, která nebyla 100, nýbrž $60 \text{ rad}\cdot\text{s}^{-1}$. Jak je patrné, tak jiná žádaná hodnota otáček vedla i k jiné odhadnuté hodnotě. S nižšími otáčkami klesá odhadnutá hodnota, a naopak s rostoucími otáčkami by byla tato hodnota vyšší. To je samozřejmě nežádoucí, neboť v ideálním případě by měl být odhad stejný v celém rozsahu otáček.



Obr. 7-11 Průběh odhadu R_2 pomocí W_1 pro vyšší koeficient učení $\eta_1 = 0,001$ a při postupném navýšení odporu o 30, 60 a 100 %

Na tomto průběhu je vidět již zmiňovaný odhad rotorového odporu při vyšším koeficientu učení $\eta_1 = 0,001$, který demonstruje postupné „zabloudění“ váhy W_1 .

Další faktor, který výrazně ovlivňuje odhad R_2 , je velikost zátěžného momentu. S nižším zátěžným momentem již síť prakticky nereaguje. To je patrné i z průběhů na Obr. 7-12, kde je kromě původního odhadu pro 65 Nm ještě odhad při zatížení 20 Nm, který je označený fialovou barvou.



Obr. 7-12 Průběhy odhadu R_2 pomocí W_1 při zatížení 65 Nm a 20 Nm a postupném navýšení odporu o 30, 60 a 100 %.

Z provedených simulací je patrné, že prezentovaná umělá neuronová síť je schopna s dostatečnou přesností odhadovat magnetický tok rotoru, ale odhad

rotorového odporu je oproti jmenovité hodnotě dvojnásobný. To i přesto, že jedná o největší dosaženou přesnost.

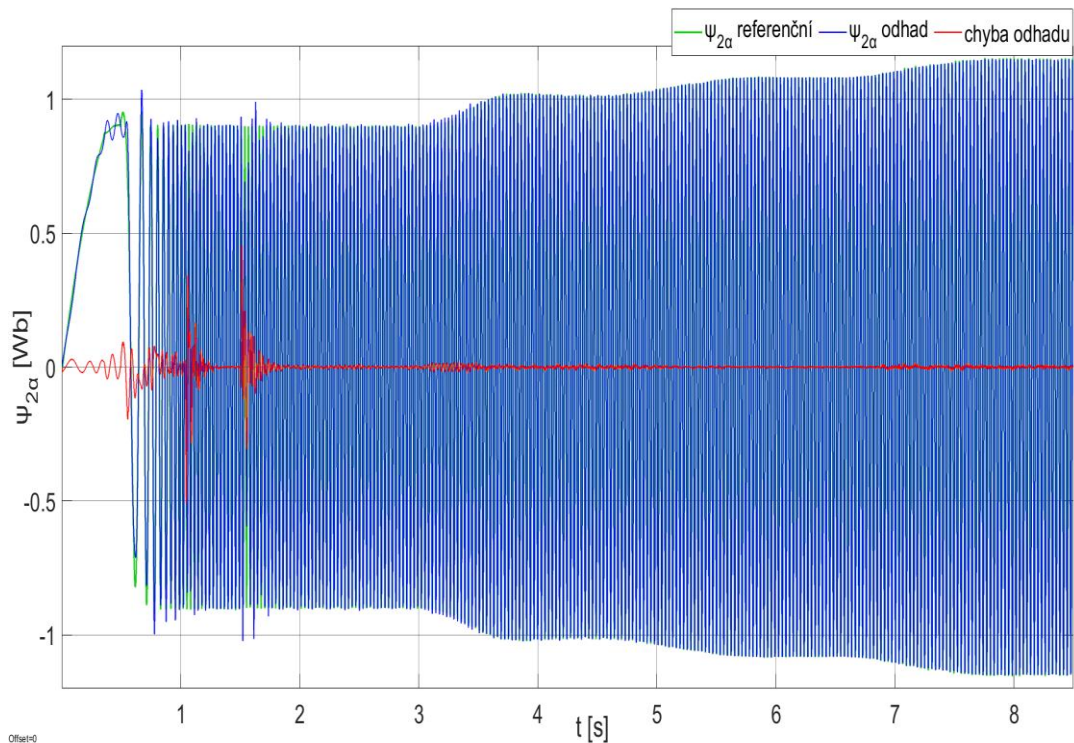
Se změnou rotorového odporu pak bylo uvedeno více faktorů, které odhad významně ovlivňují a síť především reaguje na jakoukoliv změnu odporu podobně a nezáleží tolik na velikosti této změny. Jelikož odpor určený pomocí neuronové sítě je často i víc jak dvakrát větší, než je reálná hodnota, tak je zjevné, že tento odhad nelze použít pro korekci. V další kapitole je na pár průbězích znázorněn vliv váhy W_3 na odhad, pokud není potlačen.

7.3 Průběhy odhadu rotorového odporu bez potlačení váhy W_3

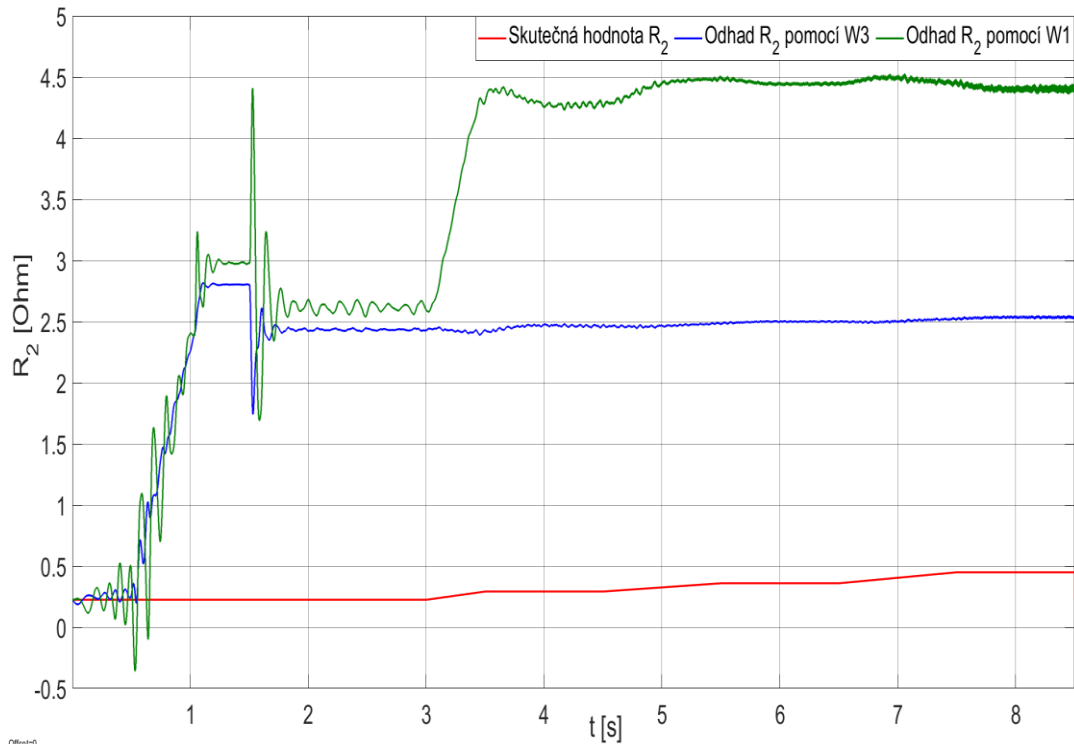
V této kapitole je na dvou obrázcích demonstrován odhad složky $\psi_{2\alpha}$ vektoru rotorového magnetického toku a odhad rotorového odporu pro případ, kdy váha W_3 není tolik potlačena. Byla snaha, aby při tomto nastavení byly jednotlivé váhy co nejvíce vyrovnané a aby dávaly co nejpřesnější odhad. Nastavení koeficientů učení a momentových faktorů je v Tab. 7-2 a na Obr. 7-13 a Obr. 7-14 jsou pak naměřené průběhy zmiňovaných veličin.

Tab. 7-2 Nastavení koeficientů učení a momentových faktorů pro odhad R_2 .

	Koeficient učení η	Momentový faktor μ
W_1	0,0001	0,005
W_2	0,0001	0,002
W_3	0,0000001	0,000001



Obr. 7-13 Průběh $\psi_{2\alpha}$ pro referenční U-I a adaptivní ANN model a chyba odhadu daná jejich rozdílem bez potlačení váhy W_3 .



Obr. 7-14 Průběhy odhadu R_2 pomocí W_1 a W_3 při postupném navýšení odporu o 30, 60 a 100 % bez potlačení váhy W_3 .

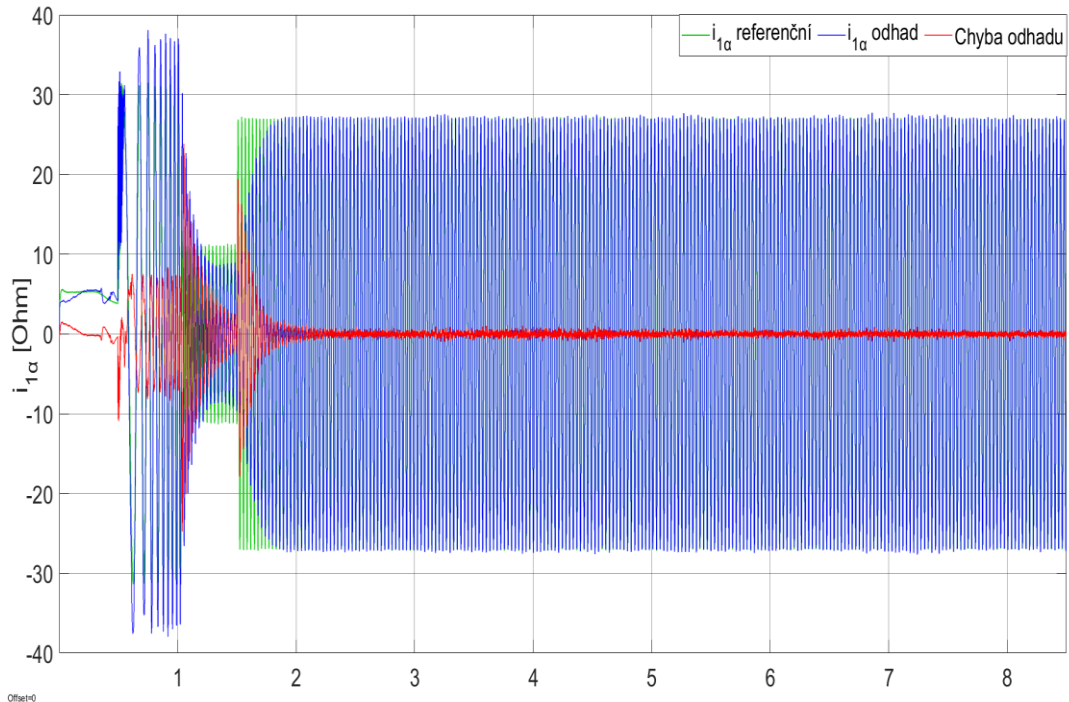
Z průběhu referenčního a odhadnutého magnetického toku je vidět, že přesnost při rozběhu není oproti předchozí kapitole výrazně přesnější, avšak ustálení je rychlejší. Pro průběh rotorového odporu je ale patrné, že odhadnutá hodnota jak pomocí W_1 , tak W_3 je naprosto nepřesná a pohybuje se okolo 2,5 Ohmu i při jmenovité hodnotě rotorového odporu v modelu motoru. Při jeho nárůstu odhad pomocí W_3 téměř vůbec nereaguje. Druhá váha sice reaguje, ale špatně a chyba odhadu se akorát zvětšuje. Tato varianta se tak ukazuje jako méně přesná, než při než při potlačení váhy W_3 a nelze ji použít.

7.4 Odhad statorového odporu

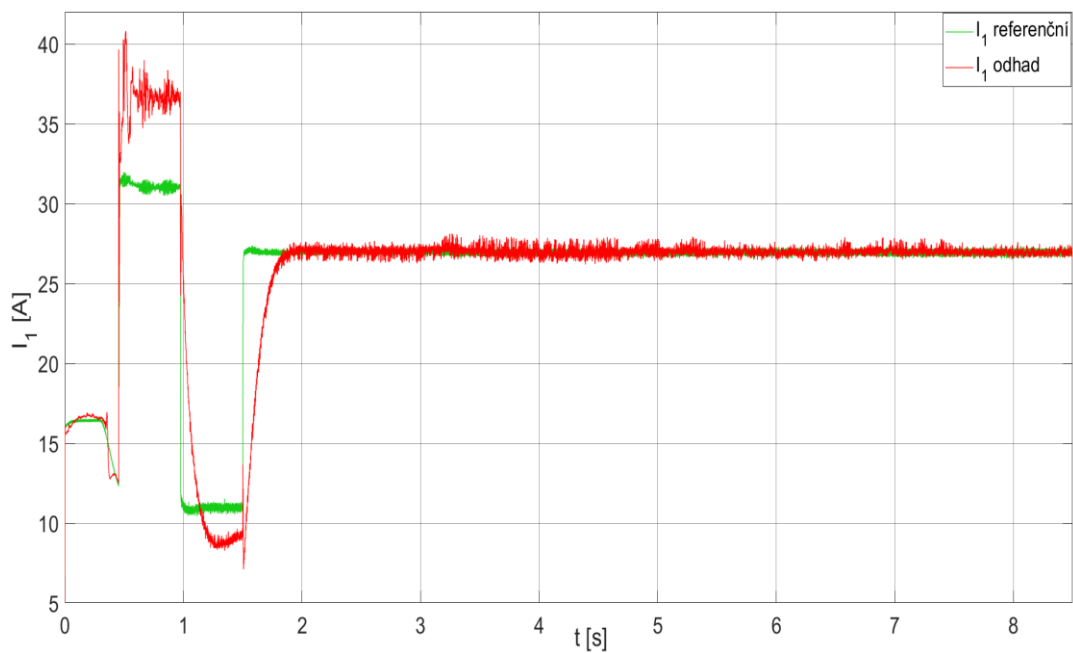
Odhad statorového odporu je druhým cílem této práce, neboť jeho změna může ovlivnit referenční napěťový U-I model pro odhad rotorového odporu. Byl použit stejný průběh simulace jako u druhého zmíněného odporu. Tedy motor je rozeběhnut na $100 \text{ rad}\cdot\text{s}^{-1}$, v čase $t = 1,5 \text{ s}$ dojde k zatížení momentem 65 Nm a následně dojde k navýšení statorového odporu postupně o 30, 60 a 100 % v čase $t = 3; 4,5; 6,5 \text{ sekund}$. Stejně jako u odhadu rotorového odporu, i zde je důležitá chyba odhadu, která vstupuje do bloku pro úpravu vah. Zde jako referenční sloužily složky statorového proudu. Konkrétně průběh $i_{1\alpha}$ je i spolu s odhadem a chybou na Obr. 7-15, průběh velikosti vektoru \vec{i}_1 pak na Obr. 7-16. Nastavení koeficientů vah a momentových faktorů je v následující Tab. 7-3.

Tab. 7-3 Nastavení koeficientů učení a momentových faktorů pro odhad R_1 .

	Koeficient učení η	Momentový faktor μ
W_4	0,0000001	0,000001
W_5	0,00007	0,0005
W_6	0,0007	0,005

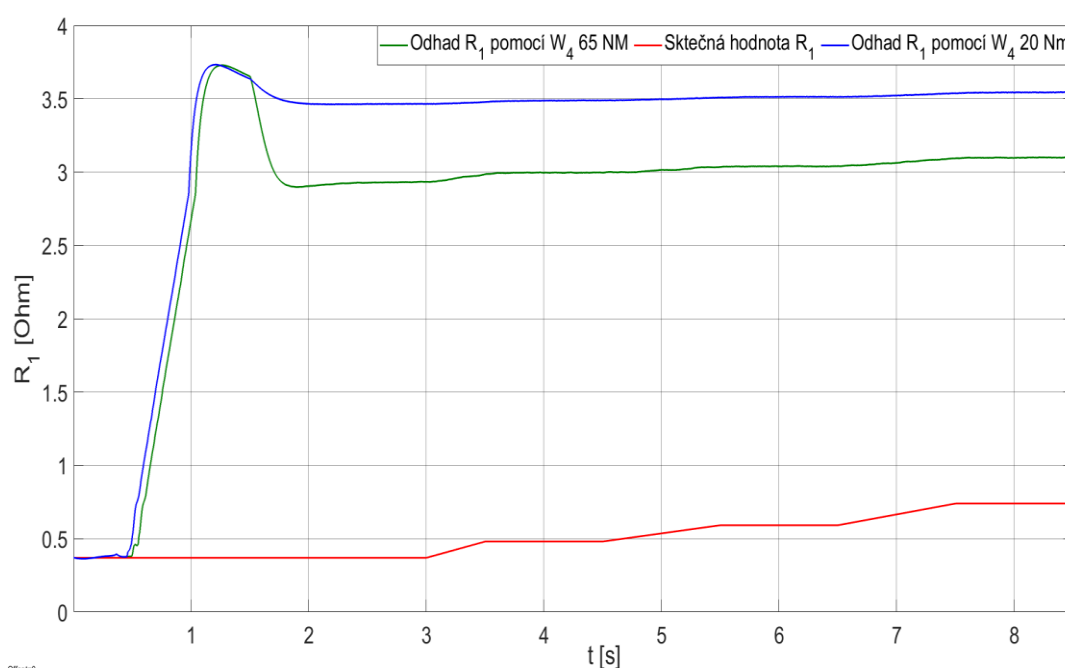


Obr. 7-15 Průběhy odhadu R_2 pomocí W_1 při zatížení 65 Nm a 20 Nm a postupném navýšení odporu o 30, 60 a 100 %.



Obr. 7-16 Průběhy odhadu R_2 pomocí W_1 při zatížení 65 Nm a 20 Nm a postupném navýšení odporu o 30, 60 a 100 %.

Je patrné, že satorový proud se v průběhu simulace s nárustem satorového odporu prakticky nezvyšuje, ale výrazně se mění se zatížením. To, jak je vidět, ovlivňuje i odhad neuronové sítě. Rozběh trvá relativně krátce a síť se nestihne ustálit, načež dojde k dosažení žádané rychlosti, poklesu proudu a snížení chyby odhadu. Následně je však motor zatížen, což má za následek výrazné navýšení proudu a opět nárůst chyby odhadu. Poté ovšem motor pracuje s trvalým zatížením, síť se tak poměrně rychle ustálí a s poměrně malou chybou kopíruje žádanou hodnotu. Chyba odhadu se v ustáleném stavu pohybuje maximálně kolem 1 A, ale většinou je tato hodnota nižší. Právě tato chyba je důležitá pro úpravu vah, aby v další iteraci došlo, pokud možno, k jejímu snížení. Hodnotu satorového odporu pak lze určit pomocí váhy W_4 podle rovnice (5-36). Nasimulovaný průběh odhadnutého a skutečného satorového odporu při jeho navýšení v modelu asynchronního motoru o 30, 60 a 100 % je na Obr. 7-17.

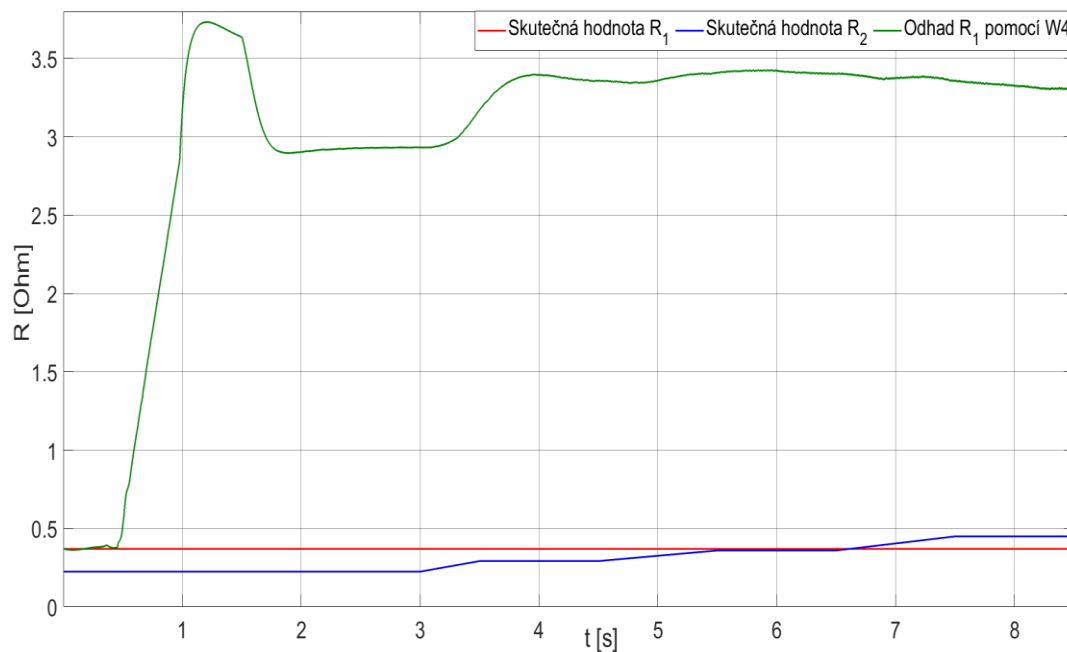


Obr. 7-17 Průběhy odhadu R_1 pomocí W_4 při zatížení 65 Nm a 20 Nm a postupném navýšení odporu o 30, 60 a 100 %.

Z obrázku je patrné, že odhadovaná hodnota satorového odporu je výrazně vyšší než skutečná velikost odporu v modelu motoru. V ustáleném stavu při jmenovité hodnotě satorového odporu se pohybuje odhad neuronové sítě kolem 2,9 Ohm, což je 7,8krát více. S nárustem odporu v modelu motoru je vidět, že i síť lehce reaguje a odhadnutá hodnota se zvyšuje, avšak stále se jedná o velice nepřesnou hodnotu, kterou nelze použít ke korekci. Nedá se ani uvažovat o možnosti odhadu pomocí poměrného navýšení. Kdyby totiž při určitém procentuálním nárůstu satorového odporu v motoru narostl procentuálně i odhad oproti hodnotě odhadu při jmenovité hodnotě, tak by se dalo dále uvažovat nad korekcí pomocí takovéto procentuální změny. To je však v tomto případě evidentně nemožné, neboť reakce odhadu na navýšení odporu v modelu motoru je minimální.

Co se týče vlivu ostatních veličin na změnu odhadu, tak ten je podobný jako je tomu u odhadu rotorového odporu. S rostoucí rychlostí tak odhadovaná velikost

roste, a naopak s klesající rychlostí se zmenšuje. To je dobře patrné na Obr. 7-23, kde je právě průběh odhadu satorového odporu v širokém rozsahu otáček. Pro různé zátěžné momenty je vliv patrný z Obr. 7-17, kdy s rostoucí zátěží i klesá odhadnutá hodnota. Velký vliv má i změna rotorového odporu v modelu motoru, jelikož při jeho navýšení se mění hodnota satorového proudu a tím i odhad satorového odporu. Byla proto provedena simulace se stejným průběhem jako pro výše prezentovaných obrázcích, ale satorový odpor v modelu motoru se neměnil a měl jmenovitou hodnotu, zatím co satorový odpor rostl postupně o 30, 60 a 100 %. Na Obr. 7-18 je vidět, jak se s navýšením rotorového odporu změní poměrně výrazně i hodnota odhadu. Při kompenzaci satorového odporu by tak měla být nejdříve zajištěna kompenzace rotorového odporu, aby byl tento vliv omezen.



Obr. 7-18 Průběhy odhadu R_1 pomocí W_4 při postupném navýšení rotorového odporu o 30, 60 a 100 % a se jmenovitou hodnotou satorového odporu.

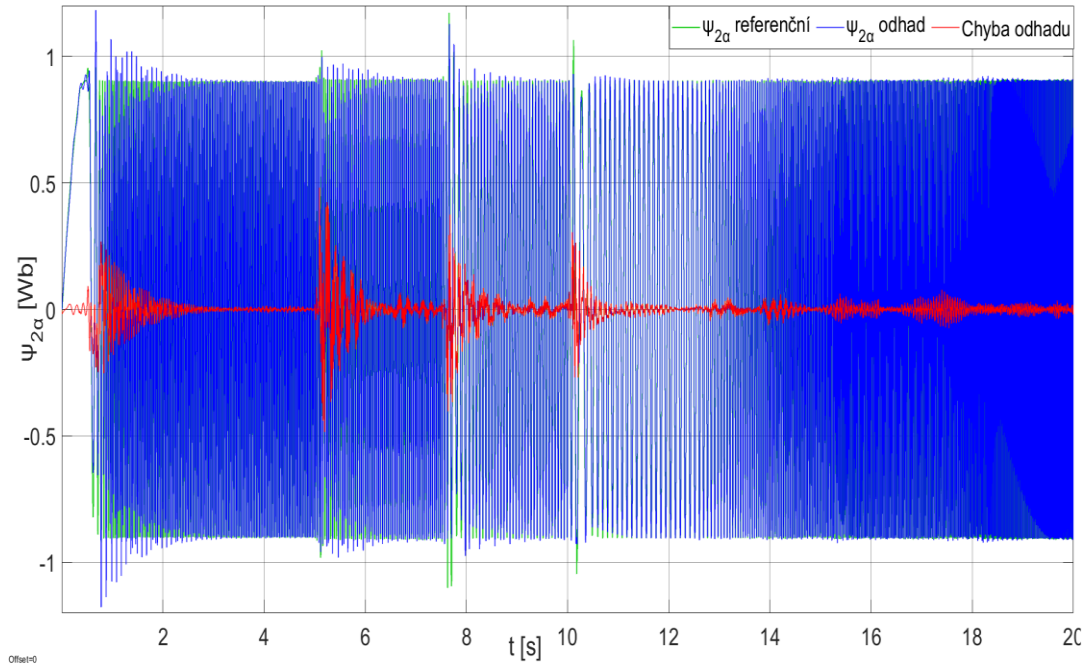
7.5 Odhad elektrické úhlové rychlosti rotoru

Odhad elektrické úhlové rychlosti byl zařazen, neboť její výpočet vychází přímo z již implementované umělé neuronové sítě pro odhad rotorového odporu. Pro připomenutí lze elektrickou úhlovou rychlost určit pomocí váhy W_2 jako $\omega = W_2 / T_s$ a mechanická úhlová rychlost tak je $\Omega = W_2 / (p_p T_s)$. Nastavení koeficientů učení pro odhad rychlosti je v následující tabulce.

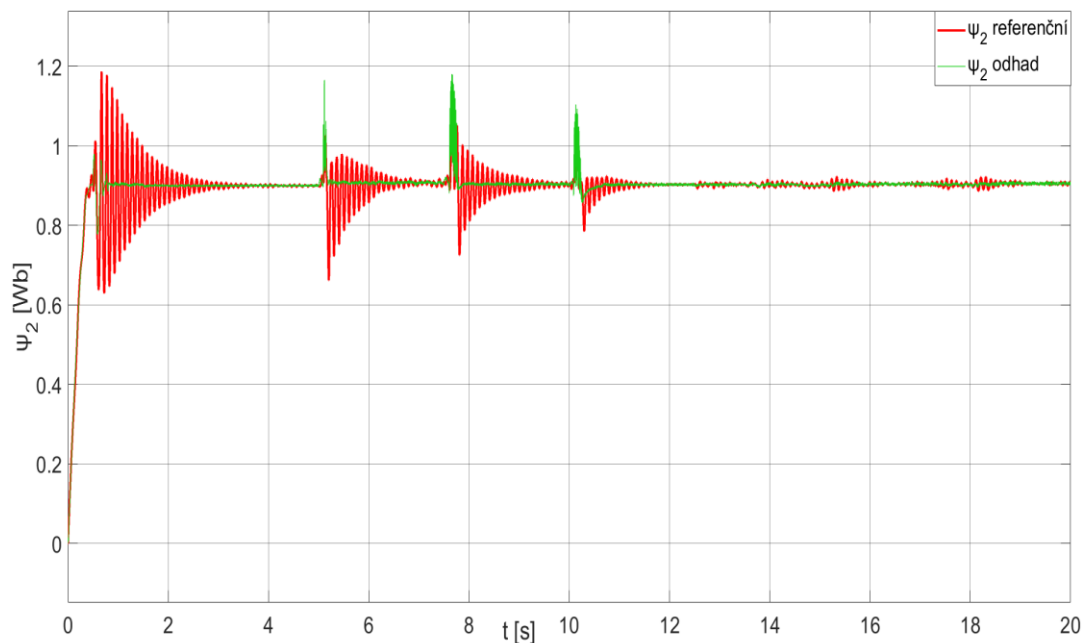
Tab. 7-4 Nastavení koeficientů učení a momentových faktorů pro odhad elektrické úhlové rychlosti.

	Koeficient učení η	Momentový faktor μ
W_1	0,0001	0,005
W_2	0,0002	0,001
W_3	0,0000000001	0,000000001

Při simulaci byl motor rozeběhnut na $100 \text{ rad}\cdot\text{s}^{-1}$ a v čase $t = 1,5 \text{ s}$ je zatížen momentem 65 Nm . Od $t = 5 \text{ s}$ je pak každých $2,5$ sekundy změněna žádaná rychlost, a to postupně na $80, 50, 30, 100, 120$ a $153 \text{ rad}\cdot\text{s}^{-1}$. Hodnoty statorového a rotorového odporu jsou jmenovité. Odhad a ustalování neuronové sítě je při nižších rychlostech ovlivněn nepřesností referenčního U-I modelu. To je patrné i z následujících dvou obrázků, na kterých je nejdříve průběh složky magnetického rotorového toku a na druhém pak velikost vektoru tohoto toku.

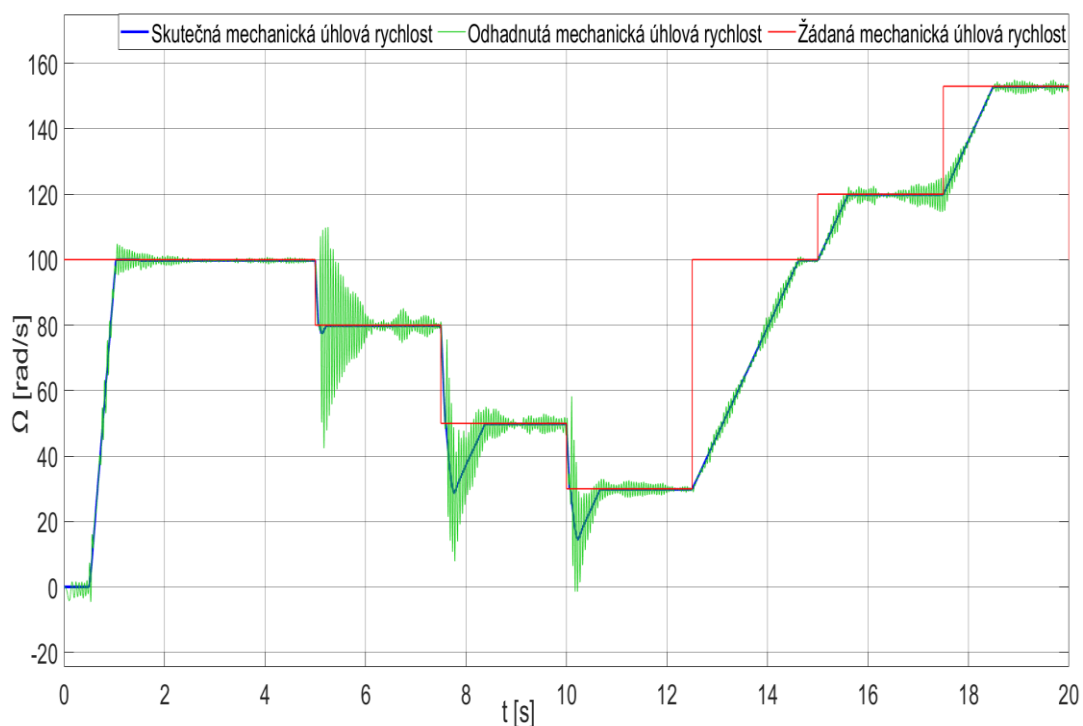


Obr. 7-19 Průběh $\psi_{2\alpha}$ pro referenční U-I a adaptivní ANN model a chyba odhadu daná jejich rozdílem při odhadu rychlosti.



Obr. 7-20 Průběh velikosti vektoru ψ_2 pro referenční U-I a adaptivní ANN model a chyba odhadu daná jejich rozdílem při odhadu rychlosti.

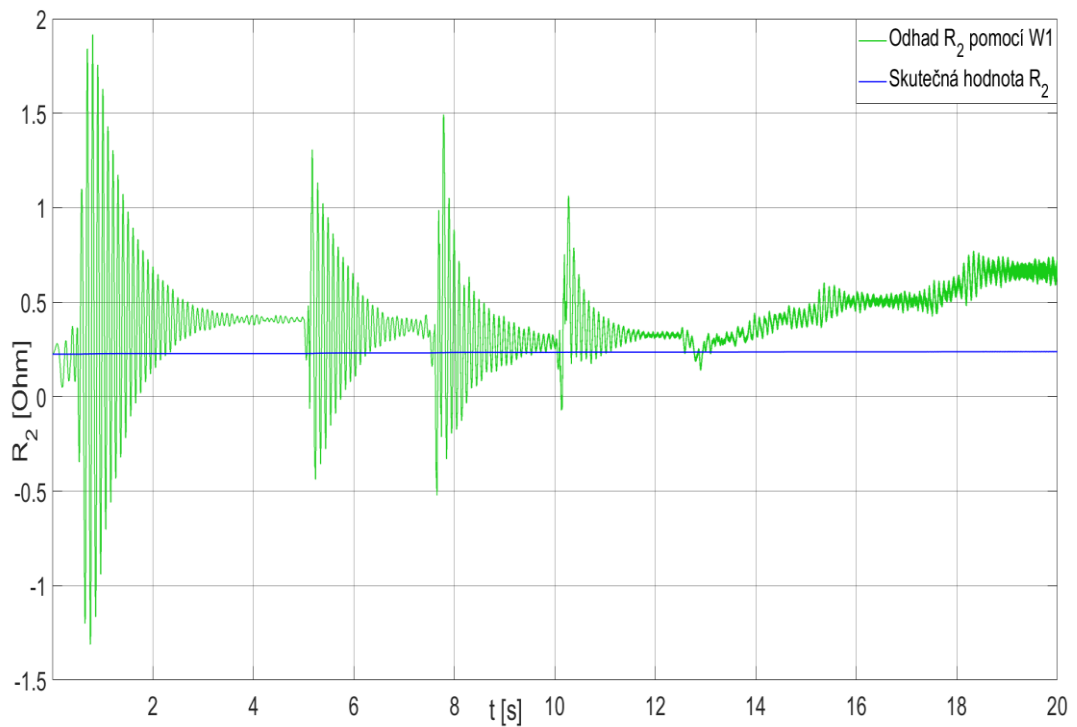
Jelikož koeficienty jsou nastaveny velice podobně jako v kapitole 7.2, tak z průběhů složky rotorového magnetického toku na Obr. 7-19 je vidět, že se síť opět první zhruba 3 sekundy ustaluje, ale potom již vykazuje malou chybu odhadu. Při změnách žádané rychlosti na 80, 50 a 30 $\text{rad}\cdot\text{s}^{-1}$ v čase $t = 5; 7,5; 10$ sekund je patrné, že síť vypadne z ustáleného stavu, chyba odhadu se skokově zvětší a následně opět dochází k ustálení. Tato nepřesnost při změně rychlosti však není úplně zapříčiněna neuronovou sítí, ale referenčním U-I modelem. Na Obr. 7-20 s průběhy velikosti vektoru rotorového toku, jsou vidět překmity referenčního modelu při poklesech na zmíněné otáčky, na což síť reaguje. Při nárůstu rychlosti na 100, 120 a 153 $\text{rad}\cdot\text{s}^{-1}$ již tyto překmity nevznikají. Vliv změn v průběhu rotorového toku je patrný i na samotném odhadu mechanické úhlové rychlosti na následujícím obrázku.



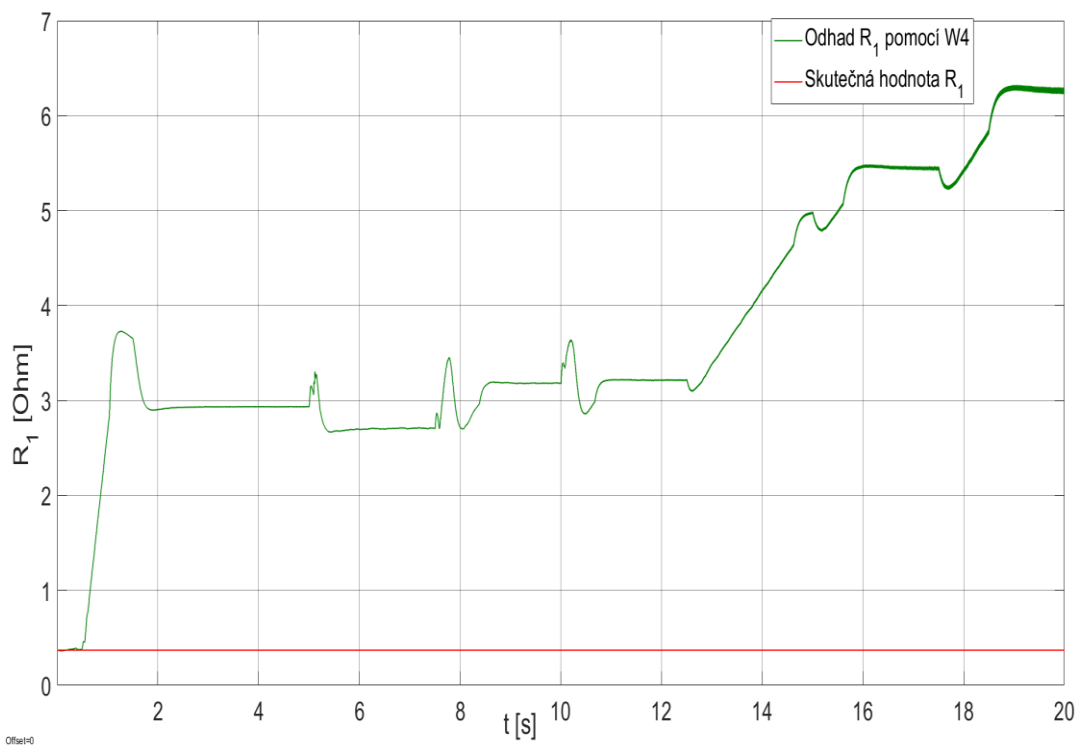
Obr. 7-21 Průběh žádané, skutečné a odhadnuté mechanické úhlové rychlosti při postupné změně otáček na 100, 80, 50, 30, 100, 120 a 153 $\text{rad}\cdot\text{s}^{-1}$.

Je patrné, že odhad rychlosti vykazuje lepší výsledky, než tomu bylo u odporů. I přesto jsou však patrné veliká zvlnění především v již zmíněných časech, kdy dochází k poklesu rychlosti. Především u zpomalení na 50 a 30 $\text{rad}\cdot\text{s}^{-1}$ není regulátor schopen s danou zátěží schopen dostatečně rychle reagovat a dochází k poklesu pod žádané otáčky. To způsobuje větší zvlnění i na straně odhadu. Při ustálených otáčkách je odhad viditelně nejlepší kolem hodnoty 100 $\text{rad}\cdot\text{s}^{-1}$ a poměrně přesný je i při zrychlování. Kvůli četným zvlněním odhadnuté rychlosti bohužel není možné použít tyto otáčky pro účely řízení, neboť takové změny regulátory nejsou schopny vykompenzovat. Vliv otáček na odhad statorového a rotorového odporu byl sice již probrán v předchozích kapitolách, avšak v této simulaci byly žádané otáčky ve skoro celém rozsahu, a proto pro demonstraci byli sejmuty tyto odhady i zde. Jejich průběhy jsou na Obr. 7-22 a Obr. 7-23 a je z nich patrné, že i při jmenovité hodnotě odporů se jednotlivé odhady liší až dvojnásobně oproti odhadu při jiných otáčkách.

Oproti skutečné hodnotě je pak odhad 2 – 4krát větší v případě rotorového odporu a 10 – 20krát větší u statorového odporu.



Obr. 7-22 Průběhy odhadu R_2 pomocí W_1 při postupné změně žádané rychlosti na 100, 80, 50, 30, 100, 120 a 153 $\text{rad}\cdot\text{s}^{-1}$.



Obr. 7-23 Průběhy odhadu R_1 pomocí W_4 při postupné změně žádané rychlosti na 100, 80, 50, 30, 100, 120 a 153 $\text{rad}\cdot\text{s}^{-1}$.

ZÁVĚR

V úvodu této diplomové práce bylo představeno shrnutí základních informací o fungování umělých neuronových sítí. Problémem je obrovské množství jejich modelů a rozdílných přístupů používaných pro různé aplikace v mnoha odvětvích. Proto se tato práce zaměřuje na části použitelné především v elektrotechnice, případně přímo související s použitím pro odhad nelineárních systémů. Naneštěstí tuzemská literatura se v této oblasti neukázala jako příliš obsáhlá a v zahraniční literatuře je možné se setkat s nesjednoceným názvoslovím. To se jeví jako zapříčiněné právě širokým rozsahem jejich použití v průběhu desítek let od jejich vzniku. V některých případech tak bylo potřeba zvolit určité pojmenování, i když nebylo jasné, zda je správné, neboť se literatura v daných případech rozcházela.

V následující druhé kapitole byla shrnuta teorie asynchronního motoru, řízení měniče pomocí prostorového vektoru a vektorově orientovaná regulace spolu s potřebnými matematickými rovnicemi. Zde navázala kapitola projednávající již možnost použití umělých neuronových sítí v kombinaci s MRAS pro odhad odporů motoru a případně i mechanické úhlové rychlosti. Použití jednoduché struktury jednovrstvých sítí s lineární aktivační funkcí jako adaptivních modelů se jeví jako efektivní a relativně snadná možnost. K jejich implementaci navíc nebylo nutné učit se s toolboxem „Deep Learning“ pro Simulink obsahujícím bloky pro práci s neuronovými sítěmi, ale sítě byly aplikovány čistě na bázi jejich matematického popisu.

Při simulacích celého modelu bylo nejdříve nutné správně nastavit koeficienty učení a momentové faktory pro jednotlivé váhy. Podle literatury měly pro odhad rotorového odporu stačit upravované váhy W a W_3 a pro určení statorového odporu váha W_4 . Úprava koeficientů a momentových faktorů se ukázala být poměrně náročným problémem, neboť pro některé hodnoty ani změna v řádu desítek neznamenala výraznější vliv na kvalitu odhadu, a naopak při dosažení dobré kvality odhadu i menší změna mohla mít výrazný vliv. Zároveň nebylo předem jasné, jakou hodnotu by koeficienty učení či momentové faktory měly mít, což značně prodlužovalo ladění sítě. To bylo silně ovlivněno i zvoleným řešičem, případně krokem simulace, neboť přesnější řešič dával lepší výsledky, ale za cenu pomalejšího výpočtu. Byl vyzkoušen i přístup, kdy se koeficienty učení adaptivně měnily v závislosti na chybě odhadu. To však nepřineslo zlepšení a jelikož bylo nutné zavést další koeficient, který bylo nutné ladit, tak se tím akorát ztížilo nastavení celé sítě, a proto byla tato metoda zavrhnuta.

Nejdříve tak byl zvolen přístup kdy koeficienty byly upravovány rovnoměrně tak, aby se váhy podílely na odhadu stejným dílem. Tento přístup ovšem nevykazoval dobré výsledky a jako adaptivní byla zavedena i váha W_2 do modelu pro odhad rotorového odporu a váhy W_5 a W_6 do modelu pro odhad statorového odporu. Tímto krokem bylo dosaženo zlepšení kvality odhadu veličin mezi referenčníma a adaptivním modelem, avšak i přes intenzivní snahu najít optimální nastavení koeficientů učení a momentových faktorů nebylo ani u jedné z nich dosaženo uspokojivého odhadu rotorového a statorového odporu. Pro rotorový odpor bylo nejlepšího výsledku dosaženo po potlačení váhy W_3 , kdy byl odpor počítán z váhy W_1 , ale i tak byla tato hodnota nejméně dvakrát vyšší, než byla

skutečná hodnota. Takovýto odhad, nelze použít ke korekci v modelu řízení. U statorového odporu sice zavedení dalších dvou adaptivních členů výrazně snížilo chybu mezi referenčním a adaptivním modelem, tedy odhad statorového proudu, ale samotný odpor počítaný z váhy W_4 byl nejméně zhruba 10krát vyšší, než byla skutečná hodnota. To je velice špatný výsledek, který nelze nijak použít pro korekci.

Odhad obou sítí se navíc výrazně lišil podle aktuálních provozních podmínek. Na simulacích byl demonstrován vliv žádaných otáček a zátěžného momentu. Tyto vlivy by se v reálné aplikaci navíc kombinovaly. U statorového odporu se ještě výrazně projevil vliv změny rotorového odporu v modelu motoru a odhad tohoto odporu by tak neměl smysl bez korekce rotorového odporu. To ovšem byly pouze další nedokonalosti obou sítí, neboť ty nebyly schopny s dostatečnou přesností odhadnout požadované hodnoty odporů ani v ustáleném stavu při jmenovitých podmínkách. Tím pádem nebylo možné ani použít odhad obou odporů k jejich korekci v jednotlivých modelech.

Překvapivé se ale ukázalo, že sítě byly schopny velice dobře odhadovat žádanou referenční hodnotu a chyba mezi referenčním a adaptivním modelem tak byla malá. Jelikož tato chyba pak sloužila k úpravě vah a z toho plynoucího výpočtu jednotlivých odporů, tak se očekávalo, že i chyba odhadu odporů bude malá. To se však ukázalo jako mylný předpoklad.

Nejvíce přesný se nakonec ukázal odhad mechanických otáček rotoru. Při poklesech rychlosti sice vykazoval poměrně velké zvlnění, ale při vyšších otáčkách a v ustálených stavech byla kvalita odhadu dobrá. I přesto nebylo tento odhad možné použít pro řízení celého modelu, neboť regulátor není schopen takové zvlnění otáček kompenzovat.

Pro úpravu všech prahových vah byly postupně vyzkoušeny i jiné aktivační funkce než prezentovaná lineární funkce. Otestovány byly ReLU, Sigmoid, Hyperbolický tangent aktivační funkce a lineární funkce s omezením na ± 1 . Ani jedna ovšem nevykazovala lepší výsledky než čistě lineární aktivační funkce bez omezení.

Potenciál využití neuronových sítí pro odhad parametrů asynchronní motory tímto určitě není vyčerpán a nelze je paušálně zatracovat, avšak prezentovaný přístup se jeví jako nefunkční, přestože převzaté algoritmy byly publikovány v prestižních časopisech v oboru. Jednou z možností by bylo použít složitější strukturu sítě s více skrytými vrstvami a aktivačními funkcemi. Taková to síť by se pak mohla učit ze sad naměřených dat pro různé provozní podmínky. Zde však vznikají otázky, jak moc složitá taková síť musí být, jaké aktivační funkce by byly nejlepší, případně jaké množství a jakých sad dat by bylo potřeba. Zde se nabízí nespočet možností a tento přístup je může být předmětem dalšího zkoumání, neboť umělé neuronové sítě nabízejí nepřeborné množství více či méně složitých modelů a struktur.

POUŽITÉ ZDROJE

- [1] BOSE, B. K. *Modern power electronics and AC drives*. Upper Saddle River, NJ : Prentice Hall PTR, 2001. 978-0-13-016743-9.
- [2] VAS, Peter. *Sensorless vector and direct torque control*. Oxford : Oxford University Press, 1998. Oxford science publications. 42. 9780198564652.
- [3] MCCULLOCH, Warren S. a Walter PITTS. A logical calculus of the ideas immanent in nervous activity [online]. *The Bulletin of Mathematical Biophysics*. 1943, **5**(4), 115-133. Dostupné z: 10.1007/BF02478259.
- [4] SMALHEISER, N. R. Walter Pitts [online]. *Perspectives in biology and medicine*. 2000, **43**(2), 217-226. Dostupné z: 10.1353/pbm.2000.0009.
- [5] WU, Yu-chen a Jun-wen FENG. Development and Application of Artificial Neural Network [online]. *Wireless Personal Communications*. 2018, **102**(2), 1645-1656. Dostupné z: 10.1007/s11277-017-5224-x.
- [6] SEUNG, H. S. Half a century of Hebb [online]. *Nature neuroscience*. 2000, **3 Suppl**, 1166. Dostupné z: 10.1038/81430.
- [7] HAYKIN, Simon S. *Neural networks and learning machines*. 3rd ed. New York : Prentice Hall, 2009. 978-0-13-147139-9.
- [8] ARBIB, Michael A. *Brains, Machines, and Mathematics*. New York, NY : Springer US, 1987. 978-1-4612-9153-4.
- [9] SHEPHERD, GORDON M. Introduction to Synaptic Circuits. In: GORDON M. SHEPHERD, ed. *The Synaptic Organization of the Brain* : Oxford University Press, 2004, s. 1-38.
- [10] MIROSLAV, O. *Nervové buňky a jejich svět: jak přesvědčivě, srozumitelně a působivě prezentovat* : Grada, 2015. Poradce pro praxi. 9788024790572.
- [11] ABIODUN, Oludare Isaac, Aman JANTAN, Abiodun Esther OMOLARA, Kemi Victoria DADA, Nachaat Abdelatif MOHAMED a Humaira ARSHAD. State-of-the-art in artificial neural network applications: A survey [online]. *Heliyon*. 2018, **4**(11), e00938. Dostupné z: 10.1016/j.heliyon.2018.e00938.
- [12] LEK, S. a Y. S. PARK. Artificial Neural Networks. *Encyclopedia of Ecology* : Elsevier, 2008, s. 237-245.
- [13] JASON BROWNLEE. *How to Choose an Activation Function for Deep Learning* [online] [cit. 9. ledna 2022]. Dostupné z: <https://machinelearningmastery.com/choose-an-activation-function-for-deep-learning/>.
- [14] SHARMA, Sagar, Simone SHARMA a Anidhya ATHAIYA. Activation functions in neural networks. *towards data science*. 2017, **6**(12), 310-316.
- [15] KARLIK, Bekir a A. Vehbi OLGAC. Performance analysis of various activation functions in generalized MLP architectures of neural networks. *International Journal of Artificial Intelligence and Expert Systems*. 2011, **1**(4), 111-122.
- [16] KRIS BOLTON. *A QUICK INTRODUCTION TO ARTIFICIAL NEURAL NETWORKS (PART 2)* [online] [cit. 10. ledna 2022]. Dostupné z: <https://krisbolton.com/a-quick-introduction-to-artificial-neural-networks-part-2>.

- [17] HU, Zheng, Jiaojiao ZHANG a Yun GE. Handling Vanishing Gradient Problem Using Artificial Derivative [online]. *IEEE Access*. 2021, **9**, 22371-22377. Dostupné z: 10.1109/ACCESS.2021.3054915.
- [18] JAIN, A. K., Jianchang MAO a K. M. MOHIUDDIN. Artificial neural networks: a tutorial [online]. *Computer*. 1996, **29**(3), 31-44. Dostupné z: 10.1109/2.485891.
- [19] DA SILVA, Ivan Nunes, Danilo HERNANE SPATTI, Rogerio ANDRADE FLAUZINO, Luisa Helena Bartocci LIBONI a Silas Franco DOS REIS ALVES. *Artificial Neural Networks*. Cham : Springer International Publishing, 2017. 978-3-319-43161-1.
- [20] CAMUÑAS-MESA, Luis A., Bernabé LINARES-BARRANCO a Teresa SERRANO-GOTARREDONA. Neuromorphic Spiking Neural Networks and Their Memristor-CMOS Hardware Implementations [online]. *Materials (Basel, Switzerland)*. 2019, **12**(17). Dostupné z: 10.3390/ma12172745.
- [21] MANDIC, Danilo P. a Jonathon A. CHAMBERS. *Recurrent neural networks for prediction. Learning algorithms, architectures, and stability / Danilo P. Mandic, Jonathon A. Chambers*. Chichester : John Wiley, 2001. Wiley series on adaptive and learning systems for signal processing, communications, and control. 978-0-471-49517-8.
- [22] ABRAHAM, Ajith. Artificial neural networks. *Handbook of measuring system design*. 2005.
- [23] MISHRA, M. VIDUSHIR.S., SMT. MANISHA AGARWAL a NEHA PURI. COMPREHENSIVE AND COMPARATIVE ANALYSIS OF NEURAL NETWORK [online]. *INTERNATIONAL JOURNAL OF COMPUTER APPLICATION*. 2018, **2**(8). Dostupné z: 10.26808/rs.ca.i8v2.15.
- [24] GALLO, Crescenzo. Artificial Neural Networks Tutorial. In: D.B.A.Mehdi Khosrow-Pour, ed. *Encyclopedia of Information Science and Technology, Third Edition* : IGI Global, 2015, s. 6369-6378.
- [25] HASSOUN, Mohamad H. *Fundamentals of artificial neural networks*. Cambridge, Mass., London : MIT Press, 1995. A Bradford book. 9780262082396.
- [26] KLINE, Ronald. Science and Engineering Theory in the Invention and Development of the Induction Motor, 1880-1900 [online]. *Technology and Culture*. 1987, **28**(2), 283. Dostupné z: 10.2307/3105568.
- [27] ALGER, P. L. a R. E. ARNOLD. The history of induction motors in America [online]. *Proceedings of the IEEE*. 1976, **64**(9), 1380-1383. Dostupné z: 10.1109/PROC.1976.10329.
- [28] ZECHMAIR, Derrick a Kurt STEIDL. Why the Induction Motor Could be the Better Choice for Your Electric Vehicle Program [online]. *World Electric Vehicle Journal*. 2012, **5**(2), 546-549. Dostupné z: 10.3390/wevj5020546.
- [29] BHATT, Pooja, Hemant MEHAR a Manish SAHAJWANI. Electrical Motors for Electric Vehicle – A Comparative Study [online]. *SSRN Electronic Journal*. 2019. Dostupné z: 10.2139/ssrn.3364887.
- [30] VOŽENÍLEK, Petr, Vladimír NOVOTNÝ a Pavel MINDL. *Elektromechanické měniče*. 2. vyd. V Praze : ČVUT, 2011. 978-80-01-04875-7.

- [31] PAREKH, Rakesh. AC induction motor fundamentals [online]. *Microchip Technology Inc.* 2003, (DS00887A), 1-24. Dostupné z: <http://ww1.microchip.com/downloads/en/AppNotes/00887A.pdf>.
- [32] HUGHES, Austin a Bill DRURY. *Electric motors and drives. Fundamentals, types, and applications.* Fourth edition / Austin Hughes and Bill Drury. Amsterdam : Newnes, 2013. 9780080983325.
- [33] SAIF ALDEEN SAAD ALKADHIM. *Three-phase Induction Motor: Types and Structure*, 2020.
- [34] FADY MANSOUR. Induction Motors: Construction, Principle of Operation, Power and Torque Calculations, Characteristics and Speed Control. 2020. Dostupné z: https://www.researchgate.net/publication/343546075_Induction_Motors_Construction_Principle_of_Operation_Power_and_Torque_Calculations_Characteristics_and_Speed_Control.
- [35] PAVELKA, Jiří a Pavel KOBRLE. *Elektrické pohony a jejich řízení.* 3. přepracované vydání. 978-80-01-06007-0.
- [36] MICROSEMI. Park, Inverse Park; Clarke, Inverse Clarke Transformations MSS. Software Implementation. User guide.(nd). microsemi. com.
- [37] JAVŮREK, Jiří. *Regulace moderních elektrických pohonů.* 1. vyd. Praha : Grada, 2003. 80-247-0507-9.
- [38] QUANG, Nguyen Phung a Jörg-Andreas DITTRICH. *Vector control of three-phase AC machines. System development in the practice.* Berlin : Springer, 2008. Power systems. 978-3-540-79028-0.
- [39] ZEMAN, Karel, Zdeněk PEROUTKA a Martin JANDA. *Automatická regulace pohonů s asynchronními motory.* 1. - dotisk. Plzeň : Západočeská univerzita, 2007. 9788070433508.
- [40] LIPČÁK ONDŘEJ. *Vektorové řízení asynchronního motoru pomocí DSP.* Master. Praha: České vysoké učení technické v Praze, 2018.
- [41] VUKOSAVIC, Slobodan N. *Grid-Side Converters Control and Design. Interfacing Between the AC Grid and Renewable Power Sources.* 1st ed. 2018. Cham : Springer International Publishing; Imprint: Springer, 2018. Power Electronics and Power Systems. 978-3-030-10346-0.
- [42] YU, Zhenyu. Space-vector PWM with TMS320C24x/F24x using hardware and software determined switching patterns [online]. *Texas Instruments Application Report SPRA524.* 1999. Dostupné z: <https://www.semanticscholar.org/paper/Space-Vector-PWM-With-TMS320C24x%2FF24x-Using-and-Yu/5330f49bc12022dc1c123efa7e5b817287befc77>.
- [43] SIMON STROBL. *Space Vector Modulation (SVM)* [online]. 27. září 2021 [cit. 28. února 2022]. Dostupné z: <https://imperix.com/doc/implementation/space-vector-modulation>.
- [44] KUMAR, Rakesh, Sukanta DAS, Prasad SYAM a Ajit Kumar CHATTOPADHYAY. Review on model reference adaptive system for sensorless vector control of

- induction motor drives [online]. *IET Electric Power Applications*. 2015, **9**(7), 496-511. Dostupné z: 10.1049/iet-epa.2014.0220.
- [45] GADOUE, Shady M., Damian GIAOURIS a J. W. FINCH. A neural network based stator current MRAS observer for speed sensorless induction motor drives. *2008 IEEE International Symposium on Industrial Electronics : IEEE*, 2008, s. 650-655.
- [46] ORLOWSKA-KOWALSKA, T. a M. DYBKOWSKI. Stator-Current-Based MRAS Estimator for a Wide Range Speed-Sensorless Induction-Motor Drive [online]. *IEEE Transactions on Industrial Electronics*. 2010, **57**(4), 1296-1308. Dostupné z: 10.1109/TIE.2009.2031134.
- [47] KARANAYIL, Baburaj, Muhammed Fazlur RAHMAN a Colin GRANTHAM. Online Stator and Rotor Resistance Estimation Scheme Using Artificial Neural Networks for Vector Controlled Speed Sensorless Induction Motor Drive [online]. *IEEE Transactions on Industrial Electronics*. 2007, **54**(1), 167-176. Dostupné z: 10.1109/TIE.2006.888778.
- [48] PHAM VAN, Tuan, Dung VO TIEN, Zbigniew LEONOWICZ, Michal JASINSKI, Tomasz SIKORSKI a Prasun CHAKRABARTI. Online Rotor and Stator Resistance Estimation Based on Artificial Neural Network Applied in Sensorless Induction Motor Drive [online]. *Energies*. 2020, **13**(18), 4946. Dostupné z: 10.3390/en13184946.
- [49] GUTIERREZ-VILLALOBOS, Jose M., Juvenal RODRIGUEZ-RESENDIZ, Edgar A. RIVAS-ARAIZA a Moisés A. MARTÍNEZ-HERNÁNDEZ. Sensorless FOC Performance Improved with On-Line Speed and Rotor Resistance Estimator Based on an Artificial Neural Network for an Induction Motor Drive [online]. *Sensors (Basel, Switzerland)*. 2015, **15**(7), 15311-15325. Dostupné z: 10.3390/s150715311.
- [50] VUKADINOVIC, Dinko a Mateo BAŠIČIĆ. *Artificial Neural Network Applications in Control of Induction Machines*, 2011. 978-1-61209-976-7.
- [51] TEXAS INSTRUMENTS. *Digital Motor Control* [online]. *Software Library*. 2003. Dostupné z: <https://www.ti.com/lit/ug/spru485a/spru485a.pdf?ts=1649952993213>. Dostupné z: https://www.researchgate.net/publication/343546075_Induction_Motors_Construction_Principle_of_Operation_Power_and_Torque_Calculations_Characteristics_and_Speed_Control