

UNIVERZITA PARDUBICE

Fakulta elektrotechniky a informatiky

DIPLOMOVÁ PRÁCE

2025

Bc. David Hybeš

UNIVERZITA PARDUBICE

Fakulta elektrotechniky a informatiky

**PREDIKTIVNÍ REGULÁTOR S MĚŘENOU NEBO
PREDIKOVANOU PORUCHOU**

Bc. David Hybeš

Diplomová práce

2025

Podklad pro zadání DIPLOMOVÉ práce studenta

Jméno a příjmení: **Bc. David Hybeš**
Osobní číslo: **I22182**

Téma práce: **Prediktivní regulátor s měřenou nebo predikovanou poruchou**
Téma práce anglicky: **Predictive controller with measured or predicted disturbance**
Jazyk práce: **Čeština**

Související osoby: **Ing. Daniel Honc, Ph.D. (Vedoucí)**
Katedra automatizace a matematiky

Zásady pro vypracování:

Cíl práce: Cílem práce je navrhnout a otestovat prediktivní regulátor, který pro výpočet volné odezvy soustavy použije informaci o měřené nebo predikované poruchové veličině.

Obsah teoretické části: Student provede rešerši problematiky řízení ve spojitosti s možností pracovat s informací o měřené nebo predikované poruchové veličině.

Obsah praktické části: Student provede návrh algoritmu prediktivního regulátoru, který umožní pracovat s měřenou nebo predikovanou poruchovou veličinou. Řešení ověří simulačně a také na laboratorní úloze. Porovná výsledek řízení prediktivním regulátorem (bez informace o poruše, s informací o měřené poruše a s informací o predikované poruše) se standardním PID regulátorem.

Seznam doporučené literatury:

CAMACHO, E. F.; BORDONS, C. Model Predictive Control (Second Edition). London : Springer-Verlag London Limited, 2007. 405 s.
HABER, R.; BARS, R. a SCHMITZ, U. Predictive Control in Process Engineering. Weinheim : Wiley-VCH Verlag, 2011. 600 s.
ROSSITER, J. A. Model-based Predictive Control – A Practical Approach. Boca Raton (Florida) : CRC Press, 2004. 318 s.

Podpis studenta:

Datum:

Podpis vedoucího práce:

Datum:

Prohlášení

Prohlašuji:

Tuto práci jsem vypracoval samostatně. Veškeré literární prameny a informace, které jsem v práci využil, jsou uvedeny v seznamu použité literatury.

Byl jsem seznámen s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorský zákon, zejména se skutečností, že Univerzita Pardubice má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Pardubice oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

Beru na vědomí, že v souladu s § 47b zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů, a směrnicí Univerzity Pardubice č. 9/2012, bude práce zveřejněna v Univerzitní knihovně a prostřednictvím Digitální knihovny Univerzity Pardubice.

V Pardubicích dne

Bc. David Hybeš

Poděkování

Rád bych poděkoval svému vedoucímu práce Ing. Danielu Honcovi Ph.D. za pomoc při tvorbě zadání, rady při experimentech do praktické části a vlídný přístup při řešení problémů. Dále chci poděkovat Univerzitě Pardubice, která mě poskytla prostředky pro dokončení práce především v praktické části. Nakonec chci poděkovat své rodině a přátelům za velmi cenné rady a podpoře v celém procesu vzniku diplomové práce.

V Pardubicích dne

Bc. David Hybeš

ANOTACE

Tato diplomová práce se věnuje problematice návrhu a laboratorní aplikace prediktivního regulátoru s měřenou nebo predikovanou poruchou. Byl vytvořen prediktivní regulátor ve vývojovém prostředí MATLAB. Regulátor byl otestován na řadě experimentů v simulačním prostředí a na reálné soustavě GUNT RT 050. Nakonec byl porovnán s regulačními pochody PID regulátoru.

KLÍČOVÁ SLOVA

prediktivní regulátor, prediktivní řízení, informace o poruše, MATLAB, soustava GUNT RT 050, PID regulátor

TITLE

PREDICTIVE CONTROLLER WITH MEASURED OR PREDICTED FAULT

ANNOTATION

This master thesis deals with the issue of design of predictive controller with measured or predicted disturbance. Predictive controller was designed in the development platform MATLAB. Controller was tested on series of experiments in simulated environment and on real system GUNT RT 050. Finally, it was compared to the control responses of PID controller.

KEYWORDS

predictive controller, predictive control, information about disturbance, MATLAB, system GUNT RT 050, PID controller

OBSAH

	Seznam zkratk a značek.....	10
	Seznam symbolů a veličin.....	11
	Seznam ilustrací.....	13
	Seznam tabulek.....	15
	Úvod.....	16
1	Automatické řízení.....	17
2	Úvod do prediktivního řízení.....	19
2.1	Prediktivní regulátory v průmyslu.....	20
2.2	Strategie prediktivního řízení.....	21
3	Prediktivní regulátor.....	23
3.1	Predikční model.....	23
3.2	Model procesu.....	23
3.2.1	Impulzní odezva.....	24
3.2.2	Přechodová odezva.....	25
3.2.3	Přenosová funkce.....	26
3.2.4	Stavový popis.....	27
3.2.5	Další modely.....	28
3.3	Účelová funkce.....	28
3.3.1	Účelová funkce pro jednorozměrnou soustavu.....	28
3.3.2	Účelová funkce pro vícerozměrnou soustavu.....	29
3.4	Regulační zákon.....	31
3.4.1	Analytické řešení.....	32
3.4.2	Numerické řešení.....	32
3.4.3	Výpočet optimálních akčních zásahů.....	32
4	Prediktor regulátoru ve stavovém popisu.....	34

4.1	Základní varianta prediktoru	34
4.1.1	Model pro vícerozměrnou soustavu.....	34
4.1.2	Model pro vícerozměrnou soustavu s rozšířeným stavem	34
4.1.3	Odvození pro predikci stavu a výstupu	35
4.1.4	Odvození pro predikci stavu a výstupu v maticové formě.....	36
4.2	Varianta prediktoru s poruchovou veličinou	36
4.2.1	Model pro vícerozměrnou soustavu s poruchovou veličinou.....	36
4.2.2	Model pro vícerozměrnou soustavu s poruchovou veličinou s rozšířeným stavem	37
4.2.3	Odvození pro predikci stavu a výstupu s poruchovou veličinou.....	37
4.2.4	Odvození pro predikci stavu a výstupu v maticové formě s poruchovou veličinou.....	38
4.3	Omezení veličin	39
4.3.1	Omezení v případě akčních veličin.....	39
4.3.2	Omezení v případě regulovaných veličin	40
4.3.3	Omezení v případě stavových veličin	41
5	Realizace prediktivního regulátoru	43
5.1	Vývojové prostředí MATLAB	43
5.2	Algoritmus prediktivního regulátoru	46
6	Laboratorní aplikace prediktivního regulátoru	55
6.1	Soustava GUNT RT 050	55
6.2	Identifikace soustavy.....	57
6.3	PID regulátor	60
6.3.1	Experimenty s PID regulátorem	61
6.4	Prediktivní regulátor	63
6.4.1	Experimenty s prediktivním regulátorem bez omezení, bez znalosti poruchy	64
6.4.2	Experimenty s prediktivním regulátorem bez omezení, se znalostí poruchy	66
6.4.3	Experimenty s prediktivním regulátorem s omezeními, bez znalosti poruchy	68

6.4.4 Experimenty s prediktivním regulátorem s omezeními, se znalostí poruchy	70
Závěr	73
Použitá literatura	74

SEZNAM ZKRATEK A ZNAČEK

APC	Advanced Process Control
DMC	Dynamic Matrix Control
MPC	Model Predictive Control
RMPCT	Robust Model Predictive Control Technology
OPC	Optimum Predictive Control
PC	Osobní počítač
PID	Proporcionálně Integračně Derivační regulátor

SEZNAM SYMBOLŮ A VELIČIN

<i>A</i>	stavová matice
<i>B</i>	matice vstupů
<i>C</i>	matice výstupů
<i>D</i>	přenosová matice
<i>d</i>	poruchová veličina
<i>e</i>	regulační odchylka
<i>e</i> (<i>k</i> – 1)	hodnota regulační odchylky v čase <i>k</i> – 1,
<i>e</i> (<i>k</i> – 2)	hodnota regulační odchylky v čase <i>k</i> – 2,
<i>f</i>	volná odezva
<i>G</i>	přenosová funkce
<i>g_j</i>	vzorkovaný výstup pro <i>j</i> -tý vzorkovací krok
<i>H</i> (<i>z</i> ⁻¹)	přenosová funkce systému v diskrétní oblasti <i>z</i>
<i>h_j</i>	impulzní odezva pro <i>j</i> -tý vzorkovací krok
<i>i</i>	rozmezí pro omezení veličin
<i>J</i>	účelová funkce
<i>j</i>	vzorkovací krok
<i>k</i>	krok v čase
<i>K</i>	zesílení
<i>N</i>	horizont zahrnutých vzorků
<i>N</i> ₂	horizont sledování žádané hodnoty
<i>N</i> _u	horizont řízení
<i>Q</i>	matice penalizace změn akčních veličin
<i>q</i>	penalizace akční veličiny
<i>q_{ij}</i>	váhový faktor pro penalizaci akčních zásahů pro vícerozměrnou soustavu
<i>q_j</i>	váhový faktor pro penalizaci akčních zásahů
<i>q</i> ₀	diskrétní koeficient závislé na parametrech PID regulátoru
<i>q</i> ₁	diskrétní koeficient závislé na parametrech PID regulátoru
<i>q</i> ₂	diskrétní koeficient závislé na parametrech PID regulátoru
<i>R</i>	matice penalizace regulovaných veličin
<i>r</i>	penalizace regulační odchylky
<i>r_{ij}</i>	váhový faktor pro penalizaci regulační odchylky pro vícerozměrnou soustavu
<i>r_j</i>	váhový faktor pro penalizaci regulační odchylky

r_0	zesílení
t	čas
T	vzorkovací perioda,
T_I	integrační časová konstanta
T_D	derivační časová konstanta
T_1	časová konstanta
T_2	časová konstanta
U	matice změn akčních veličin
u	akční veličina
\mathbf{u}_{\max}	maximum omezení akčních veličin
\mathbf{u}_{\min}	minimum omezení akčních veličin
$u(t+k t)$	vypočítávané akční zásahy v čase t
$u(t+k-j t)$	vypočítávané akční zásahy v čase $t+k-j$
$\Delta u(k+j-1)$	změna přírůstků vypočítávaných akčních zásahů v čase $k+j-1$
du	změna akční veličiny
W	matice žádaných hodnot
w	žádaná veličina
$w(k+j)$	žádaná hodnota w v čase $k+j$
$w(t+k)$	trajektorie žádané veličiny v prediktivním řízení
$w_i(k+j)$	žádaná hodnota w v čase $k+j$ pro vícerozměrnou soustavu
$\mathbf{x}(k)$	aktuální stav soustavy
\mathbf{x}_{\max}	maximum omezení stavových veličin
\mathbf{x}_{\min}	minimum omezení stavových veličin
Y	matice výstupních veličin
y	regulovaná veličina
$y(k+j)$	predikovaná výstupní hodnota y v čase $k+j$
\mathbf{y}_{\max}	maximum omezení regulovaných veličin
\mathbf{y}_{\min}	minimum omezení regulovaných veličin
$y(t+k t)$	predikované výstupy v čase t
$y_i(k+j)$	predikovaná výstupní hodnota y v čase $k+j$
y_0	hodnota počátečních podmínek

SEZNAM ILUSTRACÍ

Obrázek 1.1 – Schéma obvodu zpětnovazebního řízení (Cvejn, nedatováno)	17
Obrázek 2.1 – Strategie prediktivního řízení (Camacho, 2007)	21
Obrázek 2.2 – Základní sktruktura prediktivního řízení (Brázdil, 2011)	22
Obrázek 3.1 – Průběh impulzní odezvy (Camacho, 2007).....	25
Obrázek 3.2 – Průběh přechodové odezvy (Camacho, 2007).....	26
Obrázek 3.3 – Schéma regulačního zákona (Honc, nedatováno)	33
Obrázek 5.1 – Příklad vizualizace dat v MATLABu	45
Obrázek 5.2 – Prostředí MATLABu s otevřeným skriptem.....	45
Obrázek 5.3 – Vývojový diagram hlavní části regulátoru.....	46
Obrázek 5.4 – Vývojový diagram pro plnění matic G a F_p	48
Obrázek 5.5 – Vývojový diagram plnění matice G_d	49
Obrázek 5.6 – Vývojový diagram cyklu regulace a vykreslení grafů, první část	50
Obrázek 5.7 – Vývojový digram cyklu regulace a vykreslení grafů, druhá část	51
Obrázek 5.8 – Vývojový diagram omezení akčních veličin.....	53
Obrázek 6.1 – Zapojení soustavy GUNT RT 050 (RT 050 Training system speed control, HSI; nedatováno)	55
Obrázek 6.2 – Soustava GUNT RT 050 – detail.....	56
Obrázek 6.3 – Soustava GUNT RT 050.....	57
Obrázek 6.4 – Průběhy identifikace modelu soustavy GUNT RT 050	58
Obrázek 6.5 – Průběhy identifikace modelu poruchy	59
Obrázek 6.6 – Blokové schéma přenosů soustavy a poruchy.....	60
Obrázek 6.7 – Průběhy regulace s PID regulátorem v simulaci	62
Obrázek 6.8 – Průběhy regulace s PID regulátorem na soustavě GUNT RT 050	63
Obrázek 6.9 – Průběhy regulace prediktivního regulátoru bez omezení, bez znalosti poruchy v simulaci.....	65
Obrázek 6.10 – Průběhy regulace prediktivního regulátoru bez omezení, bez znalosti poruchy na soustavě GUNT RT 050.....	66

Obrázek 6.11 – Průběhy regulace prediktivního regulátoru bez omezení, se znalostí poruchy v simulaci.....	67
Obrázek 6.12 - Průběhy regulace prediktivního regulátoru bez omezení, se znalostí poruchy na soustavě GUNT RT 050.....	68
Obrázek 6.13 – Průběhy regulace prediktivního regulátoru s omezeními, bez znalosti poruchy v simulaci.....	69
Obrázek 6.14 – Průběhy regulace prediktivního regulátoru s omezeními, bez znalosti poruchy na soustavě GUNT RT 050.....	70
Obrázek 6.15 - Průběhy regulace prediktivního regulátoru s omezeními, se znalostí poruchy v simulaci.....	71
Obrázek 6.16 - Průběhy regulace prediktivního regulátoru s omezeními, se znalostí poruchy na soustavě GUNT RT 050.....	72

SEZNAM TABULEK

Tabulka 6.1 – Hodnoty pracovního bodu	57
Tabulka 6.2 – Parametry PID regulátoru.....	61
Tabulka 6.3 – Zvolené póly pozorovatele stavu	64
Tabulka 6.4 – Parametry prediktivního regulátoru bez omezení	64
Tabulka 6.5 – Hodnoty omezení veličin.....	69

ÚVOD

Cílem práce je rešerše, návrh a aplikace prediktivního regulátoru s měřenou nebo predikovanou poruchou. Odlišnost této práce oproti jiným je zaměření na odregulování poruchy ve variantách, kdy regulátor má informaci o poruše nebo nikoli. Prediktivní regulátor bude srovnáván se standardním PID regulátorem.

V teoretické části bude podán obecný popis prediktivních regulátorů a prediktivního řízení. V práci budou zmíněné alternativy prediktivních regulátorů v moderním průmyslu. Budou uvedeny matematické vztahy, na kterých se zakládá prediktivní řízení včetně implementace omezení akční veličiny, změny akční veličiny a regulované veličiny.

V praktické části budou nejdříve uvedeny matematické vztahy pro stavový prediktivní regulátor s uvažováním omezení veličin. Poté se bude práce zabývat experimenty v simulačním prostředí a na reálné soustavě GUNT 050. Budou použity různé varianty prediktivního regulátoru a výsledky budou porovnány s regulačními pochody PID regulátoru.

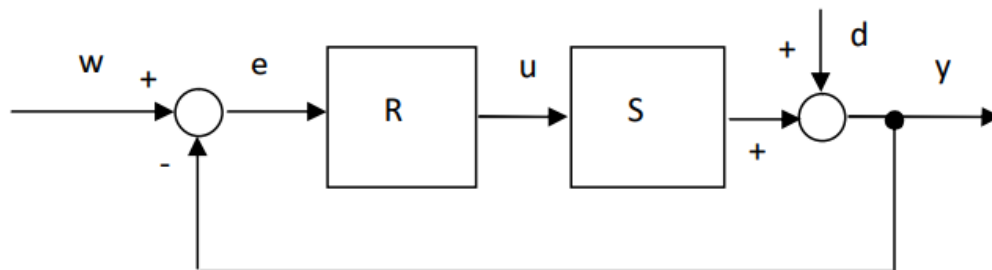
1 AUTOMATICKÉ ŘÍZENÍ

Pojem řízení se v automatizaci týká řízení regulovaných soustav různých typů. Řeší kontrolu a udržování různých parametrů, které chceme kontrolovat. Jedná se o parametry konstantní nebo proměnlivé v čase. Rozlišujeme ho na otevřené řízení a uzavřené řízení (Cvejn, nedatováno).

Jiné označení pro otevřené řízení je ovládání. Není zde zavedena zpětná vazba o regulované veličině y . Lze ji použít pouze pokud jsou konstantní vnější vlivy, což je velmi nepravděpodobné v reálném světě (Cvejn, nedatováno).

Dopředné řízení funguje na základě stanoveného matematického modelu kvůli chybějící informaci o skutečné hodnotě. Akční veličina u reaguje pouze na žádanou veličinu w nebo poruchovou veličinu d (Garzinová, 2013).

Uzavřené řízení neboli zpětnovazební obsahuje zpětnou vazbu o regulované veličině y . To znamená, že je známa informace o reálném výstupu. Díky tomu není potřeba znát přesné hodnoty parametrů systémů a případných poruchových veličin d (Cvejn, nedatováno).



Obrázek 1.1 – Schéma obvodu zpětnovazebního řízení (Cvejn, nedatováno)

Zpětnovazební řízení se může nazvat jako regulace. Na rozdíl od dopředného řízení pracuje s regulační odchylkou e , která je vyjádřena vztahem

$$e = w - y, \quad (1.1)$$

kde e – regulační odchylka,

w – žádaná veličina,

y – regulovaná veličina.

Typ regulátoru, který pracuje jinak, než jenom s vnějšími veličinami se nazývá stavový regulátor. Pracuje se stavovými veličinami, které mohou být měřené nebo odhadované. Dokáže být přesnější, ale je může být náročnější na kvalitu matematického popisu (Honc, nedatováno).

2 ÚVOD DO PREDIKTIVNÍHO ŘÍZENÍ

Prediktivní řízení neboli MPC, což je zkratka pro anglické označení Model Predictive Control vzniklo na konci 70. let 20. století. Nejedná se o konkrétní metodu regulace, ale spíše o skupinu regulačních metod, které na daném matematickém modelu procesu získá, respektive vypočte optimální hodnoty akčních zásahů podle kritéria penalizující budoucí regulační odchylky a změny akčních veličin (Honc, nedatováno; Camacho, 2007).

Díky použitým metodám jsou MPC regulátory prakticky stejné struktury. Principy, které se objevují u všech regulátorů jsou:

- použití matematického modelu na predikci výstupu ze systému v určené časové oblasti neboli v časovém horizontu,
- vypočtení sekvence akčních zásahů, které se zakládají na minimalizaci účelové funkce v dané časové oblasti, což se také nazývá jako horizont řízení,
- tzv. „ustupující horizont“, tato strategie funguje na základě použití pouze jednoho akčního zásahu z vypočtené sekvence. Toto platí pro jednorozměrné systémy, pro vícerozměrné systémy se nepoužije jeden akční zásah, ale první vektor akčních zásahů. Dalším krokem se celý výpočet opakuje a horizont se vždy posune o krok do budoucnosti (Camacho, 2007).

Poslední bod ze seznamu používaných principů se také nazývá „posuvný horizont“. U celého modelu se predikují regulované veličiny. Taková predikce vychází z dynamického matematického modelu řízeného systému. Když je na problematiku prediktivního řízení nahlédnuto z tohoto směru, tak by se jednalo o dopředné řízení. Není totiž přímá zpětná vazba, která by dávala informaci o reálném stavu regulované veličiny y .

Kvůli poruchám a nepřesnosti modelu soustavy se využívá princip „posuvného horizontu“, který byl vysvětlen výše. Díky tomuto principu a použití odhadu stavu se použije informace o regulované veličině, takže vzniká do jisté míry zpětná vazba (Honc, nedatováno).

Prediktivní regulátory vznikly z potřeb průmyslu a lze je využívat ve velkém množství odvětví zejména u vícerozměrných systémů a v případech kde jsou známé budoucí průběhy žádaných hodnot nebo poruch.

Mezi některé výhody MPC patří:

- lze používat pro širokou škálu procesů, které mohou mít velmi jednoduchou dynamiku až k těm, které ji mají složitou, zároveň lze řídit systémy, co jsou nestabilní,
- aplikace metod je intuitivní. Je možné relativně jednoduché seřízení,

Také má svoje nevýhody. Některé z nich jsou:

- pokud by měl uživatel navrhnout regulátor, vyžaduje to hlubší znalosti teorie řízení hlavní nevýhoda se týká znalosti samotného modelu, pokud není dostatečně kvalitní, tak mohou nastat velké rozdíly mezi chováním modelu a reálnou soustavou a tím ke zhoršení regulace.

I přes výše uvedené problémy je prediktivní řízení dobrá volba pro průmyslové řízení, ale není vhodné pro všechny aplikace a ten kdo řídicí systém navrhuje by měl zvážit všechna pro i proti a vybrat „tu správnou“ metodu (Camacho, 2007).

2.1 PREDIKTIVNÍ REGULÁTORY V PRŮMYSLU

Začátek prediktivního řízení v průmyslu se datuje k vytvoření prací z roku 1976 a 1980 (Brian Froisy, 2006).

Prediktivní řízení a prediktivní regulátory jsou využívány hlavně v průmyslovém prostředí. Využívá se v různých odvětvích (Camacho, 2007).

Prediktivní řízení bylo brzo, po jeho vytvoření, využíváno v chemickém průmyslu. Bylo výhodné díky jeho jednoduššímu algoritmu a také využívání impulzní nebo skokové vstupní odezvy. Byl využíván hlavně vstupně-výstupní popis, který byl vnímán jako intuitivnější a vyžadoval méně předchozích informací pro správnou identifikaci. Často se využívaly v systémech s více akčními veličinami a s omezeními. I přes značný úspěch, tak v této době řízení postrádalo stabilitu, robustnost a nebylo skoro možné upravovat řízení pro konečný horizont až na pár výjimek (Camacho, 2007).

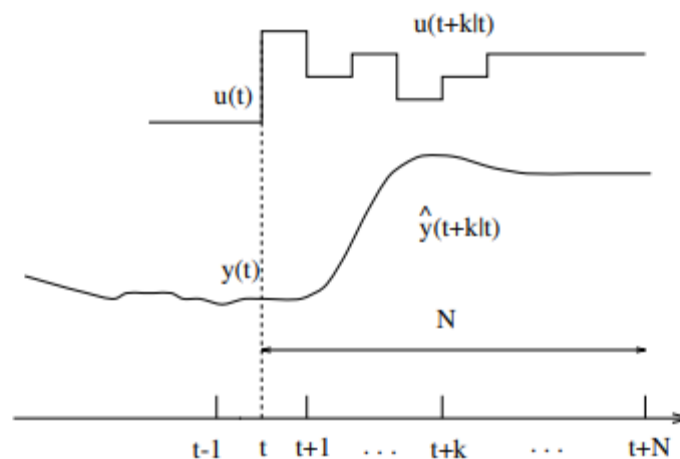
Jednotlivé společnosti poskytující tyto regulátory je vyvíjejí uvnitř firmy a neposkytují svoje metody externím subjektům. Firmy, které poskytují řešení pro prediktivní řízení jsou např.: AspenTech, Honeywell, Adersa nebo Treiber Controls. Bez přímé komunikace s firmami není možné najít pořizovací cenu nebo detaily jejich produktů v odvětví prediktivního řízení, ale lze dohledat obecné informace o tom, jak jednotlivé firmy tento problém řeší nebo co nabízejí za produkty (Camacho, 2007).

Firma AspenTech se zaměřuje na chemický průmysl a využívala tzv. dynamic matrix control (DMC). Dnes nabízí produkt DMC3, který ale využívá modely tzv. advanced process control (APC) s použitím umělé inteligence. Tento produkt se zaměřuje na zlepšení efektivity, snížení energetické náročnosti a zvýšení výroby v chemických továrnách (Aspen DMC3 | Sustain Optimal Performance | Aspentech, nedatováno).

Další firma, která nabízí řešení prediktivního řízení je Honeywell, která se zaměřuje na automatické řízení a měřicí zařízení pro zlepšení procesů v průmyslu. Nabízí například řešení pro zlepšení výroby kvality produktů s vyšší efektivitou strojů a využití nižší množství nákladů. Využívá model AlphaMPC, který mají ověřeny. Dříve byl využíván tzv. Robust Model Predictive Control Technology (RMPCT) (Machine Direction Control, nedatováno).

Treiber Controls je firma, kterou v roce 1998 koupila firma AspenTech, aby posílila svoji strukturu. Ale předtím se tato firma zabývala segmentem prediktivního řízení a bylo zde využíváno tzv. Optimum Predictive Control (OPC). Zabývala se také metodami APC (AspenTech acquires Treiber Controls – Control Engineering, 1998).

2.2 STRATEGIE PREDIKTIVNÍHO ŘÍZENÍ



Obrázek 2.1 – Strategie prediktivního řízení (Camacho, 2007)

Prediktivní regulátory se řídí jednotnou strategií, která je vyobrazena na obrázku 2.1.

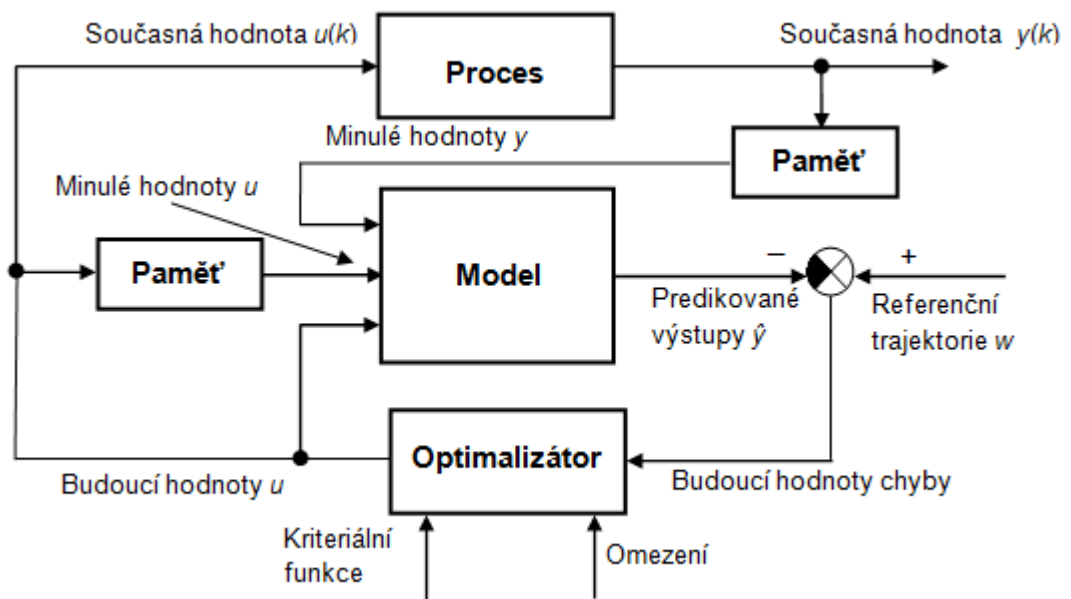
Všechny budoucí výstupy daného modelu, pro již určený horizont N , jsou předpovídány na základě hodnot, které model zná až do okamžiku t . Predikované výstupy $y(t+k | t)$, kde k nabývá hodnot od 1 do N , závisí na hodnotách minulých vstupů a výstupů a na akčních zásazích $u(t+k | t)$, kde k nabývá hodnot od 0 do $N-1$, které jsou vypočteny (Camacho, 2007).

Budoucí akční zásahy jsou vypočítávány a optimalizovány podle vybrané účelové funkce, která se snaží držet proces, co nejlépe referenční trajektorii žádané veličiny $w(t+k)$. Kritérium nebo kritériální funkce bývá kvadratická a snaží se docílit, co nejmenší regulační odchylky. Regulační odchylka u prediktivního řízení je rozdíl mezi predikovaným výstupem a

predikovanou referenční trajektorií žádané veličiny. Kritérium bere v potaz také budoucí akční zásahy (Camacho, 2007).

Při každém kroku k jsou vypočtené všechny hodnoty akčních zásahů, ale použije se pouze vypočtený akční zásah pro následující krok $u(t+1 | t+1)$. Ostatní akční zásahy jsou znovu přepočítávány, jelikož v následujícím kroku $y(t+1)$ je známa aktuální hodnota výstupu. Princip přepočítávání akčních zásahů se nazývá jako použití ustupujícího horizontu (Camacho, 2007).

Na obrázku 2.2 je vyobrazena základní sktruktura prediktivního řízení.



Obrázek 2.2 – Základní sktruktura prediktivního řízení (Brázdil, 2011)

Jedna z velkých předností prediktivního řízení spočívá v možnosti omezení neboli penalizace výstupních veličin přímo při návrhu regulátoru. Samotná kvalita a robustnost řízení je u prediktivních regulátorů vyšší než u PID regulátorů (Brázdil, 2011).

Model v základní struktuře předpovídá budoucí hodnoty predikovaného výstupu y na základě minulých hodnot výstupu y , akčních zásahů x a budoucích akčních zásahů u . Model je diskrétní (Camacho, 2007).

Funkce optimalizátoru je nastavovat budoucí akční zásahy podle daného kritéria a také zvolených penalizací (Machalský, 2011).

3 PREDIKTIVNÍ REGULÁTOR

U prediktivních regulátorů jsou obsaženy základní prvky, které jsou stejné u všech variant. Mají různé varianty a implementace. Jsou využívány různé algoritmy pro každý z prvků.

Tyto prvky jsou:

- Predikční model,
- účelová funkce,
- regulační zákon (Camacho, 2007).

3.1 PREDIKČNÍ MODEL

Základní myšlenka prediktivního regulátoru je založena na predikčním modelu. Predikční model je matematický model, který je použit pro predikce vstupu systému ve zvoleném počtu kroků. Počet kroků předpovědi je určován horizonty řízení N_u a sledování žádané hodnoty N_2 . Predikční modely jsou rozděleny na několik typů. Například stavový nebo vstupně-výstupní model.

Pro výpočet predikce lze použít různé modely procesu. Na vstupech se nachází vstupní veličiny. Při existenci poruch jsou ovlivňovány nejen vstupy, ale i zbytek systému. Chování systému je charakterizováno dynamikou poruchy. Poruchy mohou být měřené, kde je známa znalost o jejich budoucím průběhu. Další typ poruchy je takový, kde není znám její budoucí průběh, ale znám počáteční bod poruchy. Poslední varianta nastává v případě, kdy regulátor nedisponuje znalostí o poruše a není ji schopen odregulovat. Stejně jako u soustavy, tak poruchy disponují svojí dynamikou (Camacho, 2007).

3.2 MODEL PROCESU

Model procesu je matematický model, podle kterého se předpovídá chování systému, když je na vstupu konkrétní signál. V prediktivním řízení jsou používány téměř jakékoli typy modelu procesu. Nejčastěji jsou používány lineární modely. Lineární modely jsou využívány kvůli možnosti výpočtu budoucí trajektorie akčních zásahů. Tyto modely jsou potřeba pro výpočet predikcí budoucí trajektorie regulované veličiny (Brázdil, 2011).

3.2.1 Impulzní odezva

První z modelů procesu je impulzní odezva, která se označuje jako FIR. Je také známa jako konvoluční model. Vztah výstupu ke vstupu je

$$y(t) = \sum_{j=1}^N h_j u(t-j) = H(z^{-1})u(t), \quad (3.1)$$

kde $y(t)$ – výstup modelu v čase t ,
 $u(t)$ – vstup modelu v čase t ,
 $u(t-j)$ – vstup modelu v čase $t-j$,
 h_j – impulzní odezva pro j -tý vzorkovací krok,
 N – horizont zahrnutých vzorků,
 $H(z^{-1})$ – přenosová funkce systému v diskrétní oblasti z ,
 j – vzorkovací krok (Camacho, 2007).

Proměnná impulzní odezvy h_j jsou zde brány jako jednotkové impulzy. Při dostatečně velkém počtu parametru N vyjde predikce. Musí, ale dojít k velkému počtu opakování. Např.: 50 až 60.

Rovnice pro prediktor bude následující

$$y(t+k|t) = \sum_{j=1}^N h_j u(t+k-j|t) = H(z^{-1})u(t+k|t), \quad (3.2)$$

kde $y(t+k|t)$ – predikované výstupy v čase $t+k$,
 $u(t+k|t)$ – vypočítávané akční zásahy v čase $t+k$,
 $u(t+k-j|t)$ – vypočítávané akční zásahy v čase $t+k-j$,
 h_j – impulzní odezva pro j -tý vzorkovací krok,
 N – horizont zahrnutých vzorků,
 $H(z^{-1})$ – přenosová funkce systému v diskrétní oblasti z ,
 j – vzorkovací krok (Camacho, 2007).

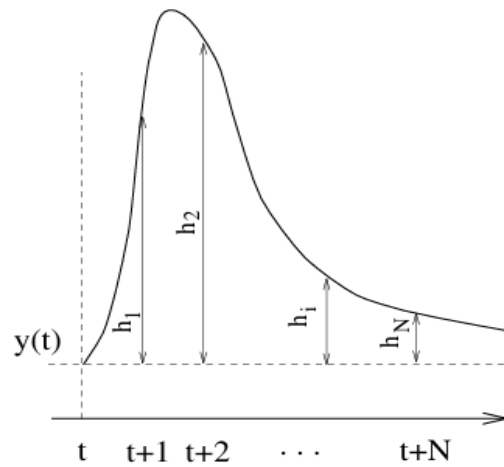
Nevýhody:

- Je potřeba velký počet opakování, aby bylo dostatečné množství dat pro predikce.

Výhody:

- Tento model je dobře využíván v průmyslu kvůli tomu, že je možné vidět vliv každé změněné proměnné na výstupu,
- není také potřeba velká znalost předchozích dat o procesu. Identifikační proces je ulehčen (Camacho, 2007; Rossiter 2003).

Na obrázku 3.1 je vidět příklad impulzní odezvy.



Obrázek 3.1 – Průběh impulzní odezvy (Camacho, 2007)

3.2.2 Přejchodová odezva

Druhý nejpoužívanější model procesu je přechodová odezva. Vstupní signál je skok. Vztah výstupu ke vstupu je

$$y(t) = y_0 + \sum_{j=1}^N g_j \Delta u(t-j) = y_0 G(z^{-1})(1-z^{-1})u(t), \quad (3.3)$$

kde y_0 – hodnota počátečních podmínek,

$y(t)$ – výstup modelu v čase t ,

$u(t)$ – vstup modelu v čase t ,

$\Delta u(t-j)$ – změna přírůstků vypočítávaných akčních zásahů v čase $t-j$,

g_j – vzorkovaný výstup pro j -tý vzorkovací krok,

N – horizont zahrnutých vzorků,

$G(z^{-1})$ – přenos vstupního skoku,

j – vzorkovací krok (Camacho, 2007).

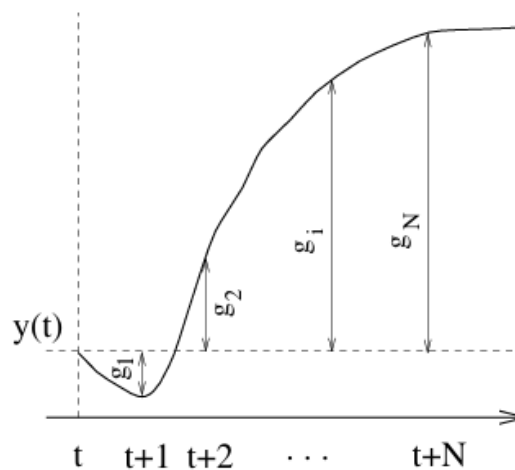
Při hodnotě proměnné $y_0 = 0$ bude rovnice pro prediktor vypadat následujícím způsobem

$$y(t+k|t) = \sum_{j=1}^N g_j \Delta u(t+k-j|t), \quad (3.4)$$

kde $y(t+k | t)$ – predikované výstupy v čase $t+k$,
 $\Delta u(t+k-j | t)$ – změna přírůstků vypočítávaných akčních zásahů v čase $t+k-j$,
 g_j – vzorkovaný výstup pro j -tý vzorkovací krok,
 N – horizont zahrnutých vzorků,
 j – vzorkovací krok (Camacho, 2007).

Výhody a nevýhody přechodové odezvy jsou velmi podobné jako u impulzní odezvy. Tedy potřebuje velký počet dat, aby bylo možné zpracovat spolehlivé predikce. Ale díky jednoduchosti modelu je dobře vidět změny proměnných na výstupu a také není potřeba znát, tolik předchozích dat o procesu (Camacho, 2007; Rossiter 2003).

Na obrázku 3.2 je vidět příklad přechodové odezvy.



Obrázek 3.2 – Průběh přechodové odezvy (Camacho, 2007)

3.2.3 Přenosová funkce

Tento model je používán v širokém množství typů prediktivních regulátorů. Koncept přenosové funkce je vyjádřen jako $G = B / A$. B a A jsou polynomy, které popisují soustavu.

Rovnice výstupu z tohoto modelu bude vypadat jako

$$A(z^{-1})y(k) = B(z^{-1})u(k), \quad (3.4)$$

kde $y(k)$ – výstup modelu v čase k ,

$u(k)$ – vstup modelu v čase k ,

$A(z^{-1})$ – polynom soustavy, který znázorňující dynamiku systému, tzv. denominační polynom,

$B(z^{-1})$ – polynom soustavy znázorňující vliv vstupu na výstup tzv. numerační polynom (Camacho, 2007).

Rovnice pro prediktor vypadá následovně

$$y(t+k|t) = \frac{B(z^{-1})}{A(z^{-1})} u(t+k|t), \quad (3.5)$$

kde $y(t+k|t)$ – predikované výstupy v čase $t+k$,

$u(t+k|t)$ – vypočítávané akční zásahy v čase $t+k$,

$A(z^{-1})$ – polynom soustavy, který znázorňující dynamiku systému, tzv. denominační polynom,

$B(z^{-1})$ – polynom soustavy znázorňující vliv vstupu na výstup tzv. numerační polynom.

Toto vyjádření lze aplikovat i pro nestabilní procesy. U tohoto modelu není potřeba velká znalost minulých dat. Ale oproti minulým modelům je potřeba mít aspoň omezenou znalost pro určení polynomů A a B (Camacho, 2007).

3.2.4 Stavový popis

Stavový popis je vyjádřen soustavou rovnic

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k), \\ y(k+1) &= Cx(k), \end{aligned} \quad (3.6)$$

kde $x(k)$ – stavový vektor v čase k ,

$x(k+1)$ – stavový vektor v čase $k+1$,

$u(k)$ – vstupní vektor v čase k ,

$y(k+1)$ – výstupní vektor v čase $k+1$,

A – stavová matice,

B – matice vstupů,

C – matice výstupů (Honc, nedatováno).

V první rovnici jsou obsaženy prvky popisující vnitřní stav. Neboli popisuje chování systému na základě vnitřních stavů a vstupů. Ve stavové matice A je obsažena dynamika systému. Také je nazývána jako matice přechodu. Je určováno, jak se promítá stav $x(k)$ do stavu $x(k+1)$. Matice vstupů B se nazývá matice řízení. Zde je popisován vliv vstupního vektoru $u(k)$ na změnu stavu.

Ve druhé rovnici jsou prvky popisující výstup systému. V matici výstupu C jsou určovány části vektoru $x(k)$, které jsou pozorovány (Hubáček, 2008).

Největší výhoda tohoto modelu spočívá v dobré práci pro systémy s více proměnnými. Řídicí zákon je zpětná vazba lineární kombinace stavových vektorů. Jednotlivé stavové vektory nemají fyzikální význam.

Nevýhoda spočívá v možném sledování jednotlivých stavů. Jelikož postrádají fyzikální význam, tak není příliš možné jejich měření. Tento problém se následně řeší vytvořením pozorovatele, který sleduje stavy systému. Stavy nejsou přesně měřeny, ale je možné je odhadnout (Camacho, 2007).

3.2.5 Další modely

Všechny modely zmíněné v minulých podkapitolách jsou lineárního typu. Ale je možné využívat i nelineární modely. Problém těchto modelů je v těžké optimalizaci, a proto se v průmyslu využívají lineární modely.

3.3 ÚČELOVÁ FUNKCE

Nejoptimálnější řízení a nastavení prediktivního regulátoru se zajišťuje pomocí minimalizace účelové funkce. Při minimalizaci účelové funkce je zajišťováno zmenšování budoucích akčních zásahů a to včetně současného akčního zásahu a budoucí regulační odchylky (Honc, nedatováno).

3.3.1 Účelová funkce pro jednorozměrnou soustavu

Účelová funkce pro jednorozměrnou soustavu je definovaná vztahem

$$J = \sum_{j=1}^{N_2} r_j (y(k+j) - w(k+j))^2 + \sum_{j=1}^{N_u} q_j \Delta u(k+j-1)^2, \quad (3.7)$$

kde N_2 – horizont sledování žádané hodnoty,

N_u – horizont řízení,

r_j – váhový faktor pro penalizaci regulační odchylky,

q_j – váhový faktor pro penalizaci akčních zásahů,

$y(k+j)$ – predikovaná výstupní hodnota y v čase $k+j$,

$w(k+j)$ – žádaná hodnota w v čase $k+j$,

$\Delta u(k+j-1)$ – změna přírůstků vypočítávaných akčních zásahů v čase $k+j-1$.

V maticovém zápisu je účelová funkce napsána jako

$$J = (\mathbf{Y} - \mathbf{W})^T \mathbf{R}(\mathbf{Y} - \mathbf{W}) + \mathbf{U}^T \mathbf{Q} \mathbf{U}, \quad (3.8)$$

$$\text{kde } \mathbf{Y} = \begin{bmatrix} y(k+1) \\ y(k+2) \\ y(k+3) \\ \vdots \\ y(k+N_2) \end{bmatrix}, \quad \mathbf{R} = \begin{bmatrix} r_1 & 0 & 0 & \cdots & 0 \\ 0 & r_2 & 0 & \cdots & 0 \\ 0 & 0 & r_3 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & r_{N_2} \end{bmatrix},$$

$$\mathbf{W} = \begin{bmatrix} w(k+1) \\ w(k+2) \\ w(k+3) \\ \vdots \\ w(k+N_2) \end{bmatrix}, \quad \mathbf{Q} = \begin{bmatrix} q_1 & 0 & 0 & \cdots & 0 \\ 0 & q_2 & 0 & \cdots & 0 \\ 0 & 0 & q_3 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & q_{N_u} \end{bmatrix},$$

$$\mathbf{U} = \begin{bmatrix} \Delta u(k) \\ \Delta u(k+1) \\ \Delta u(k+2) \\ \vdots \\ \Delta u(k+N_u-1) \end{bmatrix}.$$

V rovnici 3.8 jsou obsažené proměnné r_j a q_j , které jsou důležité pro nastavení řízení reálného systému. Pokud by nebyla aplikovaná penalizace akčních zásahů, tak by bylo možné dosáhnout ideálního sledování žádané hodnoty, ale řízení by bylo náchylné na nepřesnosti modelu. Pokud by byly použity absolutní akční zásahy, tak by došlo k trvalé regulační odchylce. Regulátor lze ovlivňovat pomocí váhy penalizačních členů pro budoucí regulační odchylku a budoucí změny akčních zásahů. Lze tak nastavit rychlost sledování žádané hodnoty nebo robustnost regulátoru. Pokud se nastaví větší penalizace akčních zásahů, tak bude regulátor mít nižší kvalitu řízení, ale bude robustnější a bude fungovat pro více nepřesné nebo nelineární modely. Při menší penalizaci bude regulátor náchylnější na nepřesnosti modelu, ale bude přesněji sledovat žádanou hodnotu (Honc, nedatováno).

Další faktor na řízení jsou horizonty N_u a N_2 . Pokud se nastaví delší horizont N_u , tak výsledek je stabilizace soustavy. Při snížení se zmenší výpočetní náročnost pro regulaci (Honc, nedatováno).

3.3.2 Účelová funkce pro vícerozměrnou soustavu

Účelová funkce pro vícerozměrnou soustavu je definovaná vztahem 3.9.

$$J = \sum_{i=1}^{n_y} \sum_{j=1}^{N_2} r_{ij} (\hat{y}_i(k+j) - w_i(k+j))^2 + \sum_{i=1}^{n_u} \sum_{j=1}^{N_u} q_{ij} \Delta u_i(k+j-1)^2, \quad (3.9)$$

kde N_2 – horizont sledování žádané hodnoty,
 N_u – horizont řízení,
 r_{ij} – váhový faktor pro penalizaci regulační odchylky pro vícerozměrnou soustavu,
 q_{ij} – váhový faktor pro penalizaci akčních zásahů,
 $y_i(k+j)$ – predikovaná výstupní hodnota y v čase $k+j$,
 $w_i(k+j)$ – žádaná hodnota w v čase $k+j$,
 $\Delta u_i(k+j-1)^2$ – změna přírůstků vypočítávaných akčních zásahů v čase $(k+j-1)^2$.

V maticovém zápisu je účelová funkce napsána jako

$$\mathbf{J} = (\mathbf{Y} - \mathbf{W})^T \mathbf{R} (\mathbf{Y} - \mathbf{W}) + \mathbf{U}^T \mathbf{Q} \mathbf{U}, \quad (3.10)$$

kde

$$\mathbf{Y} = \begin{bmatrix} y(k+1) \\ y(k+2) \\ y(k+3) \\ \vdots \\ y(k+N_2) \end{bmatrix}, \quad \hat{\mathbf{y}}(k+1) = \begin{bmatrix} y_1(k+1) \\ y_2(k+1) \\ y_3(k+1) \\ \vdots \\ y_{ny}(k+1) \end{bmatrix}$$

$$\mathbf{W} = \begin{bmatrix} w(k+1) \\ w(k+2) \\ w(k+3) \\ \vdots \\ w(k+N_2) \end{bmatrix}, \quad \mathbf{w}(k+1) = \begin{bmatrix} w_1(k+1) \\ w_2(k+1) \\ w_3(k+1) \\ \vdots \\ w_{ny}(k+1) \end{bmatrix}$$

$$\mathbf{U} = \begin{bmatrix} \Delta u(k) \\ \Delta u(k+1) \\ \Delta u(k+2) \\ \vdots \\ \Delta u(k+N_u-1) \end{bmatrix}, \quad \Delta u(k) = \begin{bmatrix} \Delta u_1(k) \\ \Delta u_2(k) \\ \Delta u_3(k) \\ \vdots \\ \Delta u_{nu}(k) \end{bmatrix}$$

$$\mathbf{R} = \begin{bmatrix} r_{11} & 0 & 0 & \cdots & 0 \\ 0 & r_{21} & 0 & \cdots & 0 \\ 0 & 0 & r_{31} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & r_{N_2} \end{bmatrix}, \quad \mathbf{r}_1 = \begin{bmatrix} r_{11} & 0 & 0 & \cdots & 0 \\ 0 & r_{21} & 0 & \cdots & 0 \\ 0 & 0 & r_{31} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & r_{ny} \end{bmatrix}$$

$$\mathbf{Q} = \begin{bmatrix} q_{11} & 0 & 0 & \cdots & 0 \\ 0 & q_{21} & 0 & \cdots & 0 \\ 0 & 0 & q_{31} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & q_{N_u} \end{bmatrix}, \quad \mathbf{q}_1 = \begin{bmatrix} q_{11} & 0 & 0 & \cdots & 0 \\ 0 & q_{21} & 0 & \cdots & 0 \\ 0 & 0 & q_{31} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & q_{nu} \end{bmatrix}$$

(Honc, nedatováno).

3.4 REGULAČNÍ ZÁKON

Cílem regulátoru je zajistit, co nejpřesněji regulování soustavy. Pro dosažení tohoto cíle se minimalizuje účelová funkce pomocí kritéria. Jsou vypočítávány budoucí akční zásahy U , a to platí i pro aktuální akční zásah. V jednotlivých intervalech vzorkování vypočtené akční zásahy minimalizují kritérium J (Camacho, 2007; Honc, nedatováno).

Následný příklad pro minimalizaci kritéria je ukázán na vícerozměrné soustavě. Nejprve se do kritéria dosadí prediktor, který je určen vztahem

$$Y = GU + f, \quad (3.11)$$

kde GU – vnucená odezva,

f – volná odezva.

Dosazení vypadá následovně

$$\begin{aligned} J &= (GU + f - W)^T R(GU + f - W) + U^T QU = \\ &= (U^T G^T + f^T - W^T)R(GU + f - W) + U^T QU = \\ &= U^T G^T RGU + U^T G^T Rf - U^T G^T RW + \\ &= f^T RGU + f^T Rf - f^T RW - W^T RGU - W^T Rf + W^T RW + U^T QU = \\ &= U^T \underbrace{(G^T RG + Q)}_H U + U^T \underbrace{G^T R(f - W)}_g + \\ &= \underbrace{(f - W)^T RGU}_{g^T} + \underbrace{(f - W)^T R(f - W)}_k, \end{aligned} \quad (3.12)$$

pro vyjádření kritéria do kvadratické formy je předpokládáno, že matice R je symetrická a lze tvrdit, že $R^T = R$. Díky tomuto tvrzení je možné použít předpoklad pro g a g^T pro stejnou operaci. Lze to využít kvůli tomu, že členy kritéria jsou vektory. Kritérium v kvadratické formě bude vypadat

$$J = U^T HU + 2g^T U + k. \quad (3.13)$$

Vyřešit kritérium lze pomocí dvou způsobů (Honc, nedatováno).

3.4.1 Analytické řešení

Toto řešení lze použít pouze ve variantě, když neexistuje omezení. Kritérium bude derivované podle U

$$\frac{dJ}{dU} = 2HU + 2g, \quad (3.14)$$

další krok je položení derivace k nulovému vektoru, který bude mít stejný rozměr jako je rozměr akčních zásahů N_u

$$2HU + 2g = \mathbf{0}, \quad (3.15)$$

při následném vyřešení se rovnice vynásobí H^{-1} zleva a vypadá následovně

$$U = -H^{-1}g = \underbrace{(G^T R G + Q)^{-1} G^T R}_{L} (W - f) = L(W - f), \quad (3.16)$$

existuje podmínka pro platnost minima z tohoto řešení. Matice H musí být semidefinitní (Honc, nedatováno).

3.4.2 Numerické řešení

Tento způsob řešení se využívá, pokud existují omezení a nelze tedy využít analytické řešení. Omezovat lze, jak akční veličiny, tak regulované veličiny. Řešení je zajišťováno metodou kvadratického programování. Výstup tohoto řešení je vektor optimálních budoucích změn a také aktuální změny akčních veličin U (Honc, nedatováno).

3.4.3 Výpočet optimálních akčních zásahů

Pro následující vztahy na výpočet optimálních akčních zásahů se počítá s variantou bez existujících omezení.

V případě jednorozměrné soustavy se optimální změna akčního zásahu vyjádří jako

$$\Delta u(k) = K(W - f), \quad (3.17)$$

kde K – první řádek matice L ,

W – matice žádaných hodnot,

f – volná odezva.

Optimální akční zásah se vypočítá pomocí vztahu

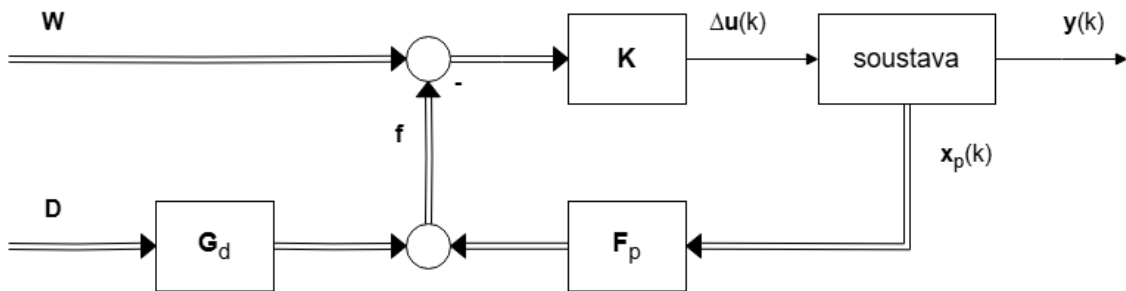
$$u(k) = \Delta u(k) + u(k - 1). \quad (3.18)$$

V případě vícerozměrné soustavy se optimální změna akčního zásahu počítá stejně jako v případě jednorozměrné soustavy. Rozdíl je, v jaké podobě se bere \mathbf{K} jako nu prvních řádků matice \mathbf{L} . Vztah je stejný.

Optimální akční zásahy jsou popsány vztahem

$$\mathbf{u}(k) = \Delta \mathbf{u}(k) + \mathbf{u}(k - 1). \quad (3.19)$$

Na obrázku 3.3 je vidět vyjádření regulačního zákona, kde je zakreslen jako schéma.



Obrázek 3.3 – Schéma regulačního zákona (Honc, nedatováno)

V případě nulových změn akčních změn bude vektor změn budoucích akčních veličin \mathbf{U} také nulový. Při vzniku odchylky mezi žádanou hodnotou \mathbf{W} a volnou odezvou f nastane akční zásah, který vykompenzuje vzniklou odchylku. Pokud bude malá penalizace změny akčních zásahů, tak bude větší zesílení \mathbf{K} . Výsledkem bude sice rychlejší regulační pochod, ale také bude více citlivý na kvalitu modelu a šum. Při větší penalizaci bude regulační pochod pomalejší, ale zase bude méně náchylný na rušení a také bude robustnější. V praktickém použití se regulátor nastavuje podle různých situací. Proto je vždycky potřeba myslet na kýžený výsledek, kterého je třeba dosáhnout (Honc, nedatováno).

4 PREDIKTOR REGULÁTORU VE STAVOVÉM POPISU

Aby došlo k minimalizaci účelové funkce, tak je potřeba určit vztahy mezi budoucími vstupy U , budoucími výstupy Y a také aktuálním stavem $\mathbf{x}(k)$ soustavy. Právě tyto vztahy je určován prediktorem. Prediktor ve stavovém popisu existuje v základní variantě nebo ve variantě s poruchovou veličinou (Honc, nedatováno).

4.1 ZÁKLADNÍ VARIANTA PREDIKTORU

Stavový popis lze zapsat pro vícerozměrné soustavy do bez a s rozšířeným stavem.

4.1.1 Model pro vícerozměrnou soustavu

$$\mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k) + \mathbf{B}\mathbf{u}(k) \quad (4.1)$$

$$\mathbf{y}(k) = \mathbf{C}\mathbf{x}(k),$$

kde $\mathbf{x}(k)$ – aktuální stav soustavy,

$\mathbf{y}(k)$ – výstup systému,

$\mathbf{x}(k+1)$ – stav soustavy v čase $k+1$,

$\mathbf{u}(k)$ – akční zásah (Honc, nedatováno).

první rovnice se označuje jako stavová a druhá rovnice jako výstupní. Matice \mathbf{A} je označována jako matice systému s rozměrem $(n \times n)$. Matice \mathbf{B} je označována jako matice řízení s rozměrem $(n \times r)$. Matice \mathbf{C} je označována jako matice přechodů s rozměrem $(m \times n)$ (Honc, nedatováno).

4.1.2 Model pro vícerozměrnou soustavu s rozšířeným stavem

$$\underbrace{\begin{bmatrix} \mathbf{x}(k+1) \\ \mathbf{u}(k) \end{bmatrix}}_{\mathbf{x}_p(k+1)} = \underbrace{\begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{0}_{nu \times nx} & \mathbf{I}_{nu \times nu} \end{bmatrix}}_M \underbrace{\begin{bmatrix} \mathbf{x}(k) \\ \mathbf{u}(k-1) \end{bmatrix}}_{\mathbf{x}_p(k)} + \underbrace{\begin{bmatrix} \mathbf{B} \\ \mathbf{I}_{nu \times nu} \end{bmatrix}}_N \Delta \mathbf{u}(k) \quad (4.2)$$

$$\mathbf{y}(k) = \underbrace{\begin{bmatrix} \mathbf{C} & \mathbf{0}_{ny \times nu} \\ \mathbf{0} \end{bmatrix}}_0 \underbrace{\begin{bmatrix} \mathbf{x}(k) \\ \mathbf{u}(k-1) \end{bmatrix}}_{\mathbf{x}_p(k)},$$

ve variantě s rozšířeným stavem se stane vstupem změna akčního zásahu $\Delta \mathbf{u}(k)$. Při softwarové realizaci se před vstup soustavy $\mathbf{u}(k)$ dodává sumátor. Do sumátoru vstupuje $\Delta \mathbf{u}(k)$. Zde je potřebná obsažení informace o minulém akčním zásahu $\mathbf{u}(k-1)$. Stává se tedy stavovou veličinou. Další stavovou veličinou se stane rozšířený stavový vektor $\mathbf{x}_p(k)$ (Honc, nedatováno).

4.1.3 Odvození pro predikci stavu a výstupu

$$\mathbf{x}_p(k+1) = \mathbf{M}\mathbf{x}_p(k) + \mathbf{N}\Delta\mathbf{u}(k) \quad (4.3)$$

$$\mathbf{y}(k+1) = \mathbf{0}\mathbf{x}_p(k+1) = \mathbf{0}\mathbf{M}\mathbf{x}_p(k) + \mathbf{0}\mathbf{N}\Delta\mathbf{u}(k),$$

další kroky odvození jsou ukazovány v časových krocích. Každá další rovnice se posouvá o jeden časový krok.

$$\mathbf{x}_p(k+2) = \mathbf{M}\mathbf{x}_p(k+1) + \mathbf{N}\Delta\mathbf{u}(k+1) = \quad (4.4)$$

$$\mathbf{M}^2\mathbf{x}_p(k) + \mathbf{M}\mathbf{N}\Delta\mathbf{u}(k) + \mathbf{N}\Delta\mathbf{u}(k+1)$$

$$\mathbf{y}(k+2) = \mathbf{0}\mathbf{x}_p(k+2) = \mathbf{0}\mathbf{M}^2\mathbf{x}_p(k) + \mathbf{0}\mathbf{M}\mathbf{N}\Delta\mathbf{u}(k) + \mathbf{0}\mathbf{N}\Delta\mathbf{u}(k+1)$$

$$\mathbf{x}_p(k+3) = \mathbf{M}\mathbf{x}_p(k+2) + \mathbf{N}\Delta\mathbf{u}(k+2) = \mathbf{M}^3\mathbf{x}_p(k) + \quad (4.5)$$

$$\mathbf{M}^2\mathbf{N}\Delta\mathbf{u}(k) + \mathbf{M}\mathbf{N}\Delta\mathbf{u}(k+1) + \mathbf{N}\Delta\mathbf{u}(k+2)$$

$$\mathbf{y}(k+3) = \mathbf{0}\mathbf{x}_p(k+3) = \mathbf{0}\mathbf{M}^3\mathbf{x}_p(k) + \mathbf{0}\mathbf{M}^2\mathbf{N}\Delta\mathbf{u}(k) +$$

$$\mathbf{0}\mathbf{M}\mathbf{N}\Delta\mathbf{u}(k+1) + \mathbf{0}\mathbf{N}\Delta\mathbf{u}(k+2)$$

⋮

$$\mathbf{x}_p(k+N_2) = \mathbf{M}\mathbf{x}_p(k+N_2) + \mathbf{N}\Delta\mathbf{u}(k+N_2) = \quad (4.6)$$

$$\mathbf{M}^{N_2}\mathbf{x}_p(k) + \mathbf{M}^{N_2-1}\mathbf{N}\Delta\mathbf{u}(k) + \mathbf{M}^{N_2-2}\mathbf{N}\Delta\mathbf{u}(k+1) +$$

$$\mathbf{M}^{N_2-3}\mathbf{N}\Delta\mathbf{u}(k+2) + \dots + \mathbf{M}^{N_2-N_u}\mathbf{N}\Delta\mathbf{u}(k+N_u-1)\mathbf{y}(k+N_2) =$$

$$\mathbf{0}\mathbf{x}_p(k+N_2) = \mathbf{0}\mathbf{M}^{N_2}\mathbf{x}_p(k) + \mathbf{0}\mathbf{M}^{N_2-1}\mathbf{N}\Delta\mathbf{u}(k) +$$

$$\mathbf{0}\mathbf{M}^{N_2-2}\mathbf{N}\Delta\mathbf{u}(k+1) + \mathbf{0}\mathbf{M}^{N_2-3}\mathbf{N}\Delta\mathbf{u}(k+2) + \dots +$$

$$\mathbf{0}\mathbf{M}^{N_2-N_u}\mathbf{N}\Delta\mathbf{u}(k+N_u-1)$$

(Honc, nedatováno).

4.1.4 Odvození pro predikci stavu a výstupu v maticové formě

$$\underbrace{\begin{bmatrix} \mathbf{y}(k+1) \\ \mathbf{y}(k+2) \\ \mathbf{y}(k+3) \\ \vdots \\ \mathbf{y}(k+N_2) \end{bmatrix}}_{\mathbf{Y}} = \underbrace{\begin{bmatrix} \mathbf{0}N & \mathbf{0}_{ny \times nu} & \mathbf{0}_{ny \times nu} & \cdots & \mathbf{0}_{ny \times nu} \\ \mathbf{0}M\mathbf{N} & \mathbf{0}N & \mathbf{0}_{ny \times nu} & \cdots & \mathbf{0}_{ny \times nu} \\ \mathbf{0}M^2\mathbf{N} & \mathbf{0}M\mathbf{N} & \mathbf{0}N & \cdots & \mathbf{0}_{ny \times nu} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{0}M^{N_2-1}\mathbf{N} & \mathbf{0}M^{N_2-2}\mathbf{N} & \mathbf{0}M^{N_2-3}\mathbf{N} & \cdots & \mathbf{0}M^{N_2-N_u}\mathbf{N} \end{bmatrix}}_{\mathbf{G}} \underbrace{\begin{bmatrix} \Delta \mathbf{u}(k) \\ \Delta \mathbf{u}(k+1) \\ \Delta \mathbf{u}(k+2) \\ \vdots \\ \Delta \mathbf{u}(k+N_u-1) \end{bmatrix}}_{\mathbf{U}} + \underbrace{\begin{bmatrix} \mathbf{0}M \\ \mathbf{0}M^2 \\ \mathbf{0}M^3 \\ \vdots \\ \mathbf{0}M^{N_2} \end{bmatrix}}_{\mathbf{F}_p} \mathbf{x}_p(k) = \underbrace{\mathbf{G}\mathbf{U}}_{\text{vnutená odezva}} + \underbrace{\mathbf{F}_p \mathbf{x}_p(k)}_{\text{volná odezva } f} \quad (4.7)$$

Výsledná odezva soustavy se skládá ze dvou složek. Jedná se o vnutenou odezvu, kterou nelze ovlivnit. Ta je dána dynamikou soustavy. Druhá složka je volná odezva, kterou lze ovlivnit (Honc, nedatováno).

4.2 VARIANTA PREDIKTORU S PORUCHOVOU VELIČINOU

Stejně jako u základní varianty prediktoru zde existují dva zápisy, s rozšířeným stavem a bez rozšířeného stavu.

4.2.1 Model pro vícerozměrnou soustavu s poruchovou veličinou

$$\mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k) + \mathbf{B}\mathbf{u}(k) + \mathbf{B}_d \mathbf{d}(k) \quad (4.8)$$

$$\mathbf{y}(k) = \mathbf{C}\mathbf{x}(k),$$

kde $\mathbf{x}(k)$ – aktuální stav soustavy,

$\mathbf{y}(k)$ – výstup systému,

$\mathbf{x}(k+1)$ – stav soustavy v čase $k+1$,

$\mathbf{u}(k)$ – akční zásah,

$\mathbf{d}(k)$ – porucha.

Je přidána matice \mathbf{B}_d , která je vynásobena s poruchovou veličinou $\mathbf{d}(k)$ (Honc, nedatováno).

4.2.2 Model pro vícerozměrnou soustavu s poruchovou veličinou s rozšířeným stavem

$$\begin{aligned} \underbrace{\begin{bmatrix} \mathbf{x}(k+1) \\ \mathbf{u}(k) \end{bmatrix}}_{\mathbf{x}_p(k+1)} &= \underbrace{\begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{0}_{nu \times nx} & \mathbf{I}_{nu \times nu} \end{bmatrix}}_{\mathbf{M}} \underbrace{\begin{bmatrix} \mathbf{x}(k) \\ \mathbf{u}(k-1) \end{bmatrix}}_{\mathbf{x}_p(k)} + \underbrace{\begin{bmatrix} \mathbf{B} \\ \mathbf{I}_{nu \times nu} \end{bmatrix}}_{\mathbf{N}} \Delta \mathbf{u}(k) + \\ &\underbrace{\begin{bmatrix} \mathbf{B}_d \\ \mathbf{0}_{nu \times nd} \end{bmatrix}}_{\mathbf{N}_d} \mathbf{d}(k) \\ \mathbf{y}(k) &= \underbrace{\begin{bmatrix} \mathbf{C} & \mathbf{0}_{ny \times nu} \\ \mathbf{0} \end{bmatrix}}_{\mathbf{0}} \underbrace{\begin{bmatrix} \mathbf{x}(k) \\ \mathbf{u}(k-1) \end{bmatrix}}_{\mathbf{x}_p(k)} \end{aligned} \quad (4.9)$$

(Honc, nedatováno).

4.2.3 Odvození pro predikci stavu a výstupu s poruchovou veličinou

$$\mathbf{x}_p(k+1) = \mathbf{M}\mathbf{x}_p(k) + \mathbf{N}\Delta\mathbf{u}(k) + \mathbf{N}_d\mathbf{d}(k) \quad (4.10)$$

$$\mathbf{y}(k+1) = \mathbf{0}\mathbf{x}_p(k+1) = \mathbf{0}\mathbf{M}\mathbf{x}_p(k) + \mathbf{0}\mathbf{N}\Delta\mathbf{u}(k) + \mathbf{0}\mathbf{N}_d\mathbf{d}(k),$$

další kroky odvození jsou ukazovány v časových krocích. Každá další rovnice se posouvá o jeden časový krok.

$$\mathbf{x}_p(k+2) = \mathbf{M}\mathbf{x}_p(k+1) + \mathbf{N}\Delta\mathbf{u}(k+1) + \mathbf{N}_d\mathbf{d}(k+1) = \quad (4.11)$$

$$\mathbf{M}^2\mathbf{x}_p(k) + \mathbf{M}\mathbf{N}\Delta\mathbf{u}(k) + \mathbf{N}\Delta\mathbf{u}(k+1) + \mathbf{M}\mathbf{N}_d\mathbf{d}(k) + \mathbf{N}_d\mathbf{d}(k+1)$$

$$\mathbf{y}(k+2) = \mathbf{0}\mathbf{x}_p(k+2) = \mathbf{0}\mathbf{M}^2\mathbf{x}_p(k) + \mathbf{0}\mathbf{M}\mathbf{N}\Delta\mathbf{u}(k) + \mathbf{0}\mathbf{N}\Delta\mathbf{u}(k+1) +$$

$$\mathbf{0}\mathbf{M}\mathbf{N}_d\mathbf{d}(k) + \mathbf{0}\mathbf{N}_d\mathbf{d}(k+1)$$

$$\mathbf{x}_p(k+3) = \mathbf{M}\mathbf{x}_p(k+2) + \mathbf{N}\Delta\mathbf{u}(k+2) + \mathbf{N}_d\mathbf{d}(k+2) = \quad (4.12)$$

$$\mathbf{M}^3\mathbf{x}_p(k) + \mathbf{M}^2\mathbf{N}\Delta\mathbf{u}(k) + \mathbf{M}\mathbf{N}\Delta\mathbf{u}(k+1) + \mathbf{N}\Delta\mathbf{u}(k+2) +$$

$$\mathbf{M}^2\mathbf{N}_d\mathbf{d}(k) + \mathbf{M}\mathbf{N}_d\mathbf{d}(k+1) + \mathbf{N}_d\mathbf{d}(k+2)$$

$$\mathbf{y}(k+3) = \mathbf{0}\mathbf{x}_p(k+3) = \mathbf{0}\mathbf{M}^3\mathbf{x}_p(k) + \mathbf{0}\mathbf{M}^2\mathbf{N}\Delta\mathbf{u}(k) +$$

$$\mathbf{0}\mathbf{M}\mathbf{N}\Delta\mathbf{u}(k+1) + \mathbf{0}\mathbf{N}\Delta\mathbf{u}(k+2) + \mathbf{0}\mathbf{M}^2\mathbf{N}_d\mathbf{d}(k) +$$

$$\begin{aligned}
& \mathbf{0MN}\Delta\mathbf{u}(k+1) + \mathbf{0N} \times \Delta\mathbf{u}(k+2) + \mathbf{0M}^2\mathbf{N}_d\mathbf{d}(k) + \mathbf{0MN}_d\mathbf{d}(k+1) + \\
& \mathbf{0N}_d\mathbf{d}(k+2) \\
& \vdots \\
& \mathbf{x}_p(k+N_2) = \mathbf{M}\mathbf{x}_p(k+N_2) + \mathbf{N}\Delta\mathbf{u}(k+N_2) + \mathbf{N}_d\mathbf{d}(k+N_2) = \tag{4.13} \\
& \mathbf{M}^{N_2}\mathbf{x}_p(k) + \mathbf{M}^{N_2-1}\mathbf{N}\Delta\mathbf{u}(k) + \mathbf{M}^{N_2-2}\mathbf{N}\Delta\mathbf{u}(k+1) + \\
& \mathbf{M}^{N_2-3}\mathbf{N}\Delta\mathbf{u}(k+2) + \dots + \mathbf{M}^{N_2-N_u}\mathbf{N}\Delta\mathbf{u}(k+N_u-1) + \\
& \mathbf{M}^{N_2-1}\mathbf{N}_d\mathbf{d}(k) + \mathbf{M}^{N_2-2}\mathbf{N}_d\mathbf{d}(k+1) + \mathbf{M}^{N_2-3}\mathbf{N}_d\mathbf{d}(k+2) + \dots + \\
& \mathbf{M}^{N_2-N_u}\mathbf{N}_d\mathbf{d}(k+N_2-1) \\
& \mathbf{y}(k+N_2) = \mathbf{0}\mathbf{x}_p(k+N_2) = \mathbf{0M}^{N_2}\mathbf{x}_p(k) + \mathbf{0M}^{N_2-1}\mathbf{N}\Delta\mathbf{u}(k) + \\
& \mathbf{0M}^{N_2-2}\mathbf{N}\Delta\mathbf{u}(k+1) + \mathbf{0M}^{N_2-3}\mathbf{N}\Delta\mathbf{u}(k+2) + \\
& \mathbf{0M}^{N_2-N_u}\mathbf{N}\Delta\mathbf{u}(k+N_u-1) + \mathbf{0M}^{N_2-1}\mathbf{N}_d\mathbf{d}(k) + \\
& \mathbf{0M}^{N_2-2}\mathbf{N}_d\mathbf{d}(k+1) + \mathbf{0M}^{N_2-3}\mathbf{N}_d\mathbf{d}(k+2) + \dots + \mathbf{0N}_d\mathbf{d}(k+N_2-1)
\end{aligned}$$

(Honc, nedatováno).

4.2.4 Odvození pro predikci stavu a výstupu v maticové formě s poruchovou veličinou

$$\begin{aligned}
& \underbrace{\begin{bmatrix} \mathbf{y}(k+1) \\ \mathbf{y}(k+2) \\ \mathbf{y}(k+3) \\ \vdots \\ \mathbf{y}(k+N_2) \end{bmatrix}}_{\mathbf{Y}} = \tag{4.14} \\
& \underbrace{\begin{bmatrix} \mathbf{0N} & \mathbf{0}_{ny\nu} & \mathbf{0}_{ny \times nu} & \dots & \mathbf{0}_{ny \times nu} \\ \mathbf{0MN} & \mathbf{0N} & \mathbf{0}_{ny \times nu} & \dots & \mathbf{0}_{ny \times nu} \\ \mathbf{0M}^2\mathbf{N} & \mathbf{0MN} & \mathbf{0N} & \dots & \mathbf{0}_{ny \times nu} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{0M}^{N_2-1}\mathbf{N} & \mathbf{0M}^{N_2-2}\mathbf{N} & \mathbf{0M}^{N_2-3}\mathbf{N} & \dots & \mathbf{0M}^{N_2-N_u}\mathbf{N} \end{bmatrix}}_{\mathbf{G}} \underbrace{\begin{bmatrix} \Delta\mathbf{u}(k) \\ \Delta\mathbf{u}(k+1) \\ \Delta\mathbf{u}(k+2) \\ \vdots \\ \Delta\mathbf{u}(k+N_u-1) \end{bmatrix}}_{\mathbf{U}} +
\end{aligned}$$

$$\begin{aligned}
& \underbrace{\begin{bmatrix} \mathbf{0}N_d & \mathbf{0}_{ny \times nd} & \mathbf{0}_{ny \times nd} & \cdots & \mathbf{0}_{ny \times nd} \\ \mathbf{0}M N_d & \mathbf{0}N_d & \mathbf{0}_{ny \times nd} & \cdots & \mathbf{0}_{ny \times nd} \\ \mathbf{0}M^2 N_d & \mathbf{0}M N_d & \mathbf{0}N_d & \cdots & \mathbf{0}_{ny \times nd} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{0}M^{N_2-1} N_d & \mathbf{0}M^{N_2-2} N_d & \mathbf{0}M^{N_2-3} N_d & \cdots & \mathbf{0}N_d \end{bmatrix}}_{G_d} \underbrace{\begin{bmatrix} \mathbf{d}(k) \\ \mathbf{d}(k+1) \\ \mathbf{d}(k+2) \\ \vdots \\ \mathbf{d}(k+N_2-1) \end{bmatrix}}_d + \\
& \underbrace{\begin{bmatrix} \mathbf{0}M \\ \mathbf{0}M^2 \\ \mathbf{0}M^3 \\ \vdots \\ \mathbf{0}M^{N_2} \end{bmatrix}}_{F_p} x_p(k) = \underbrace{GU}_{\text{vnucená odezva}} + \underbrace{F_p x_p(k)}_{\text{volná odezva } f},
\end{aligned}$$

z matice je vidět, že budoucí a minulé poruchové veličiny jsou obsaženy ve volné odezvě (Honc, nedatováno).

4.3 OMEZENÍ VELIČIN

Pokud budeme regulovat soustavu pouze v rámci simulace, tak není potřeba řešit omezování různých veličin. Mezi takové veličiny patří akční, stavové a regulované. Simulace není omezena reálnými parametry soustavy.

V případě regulace otáček motoru není žádoucí, aby měla sakční veličina neomezený rozsah. Vstupní napětí není neomezené, a proto se v reálném použití akční veličina omezuje do určitého rozsahu. Tyto omezení se liší v závislosti na fyzikálních a reálných podmínkách. Omezení se řeší pomocí rovnic a maticových vyjádření.

4.3.1 Omezení v případě akčních veličin

Při existenci omezení je dána pro akční veličiny následující podmínka, která musí být splněna

$$\mathbf{u}_{min} \leq \mathbf{u}(i) \leq \mathbf{u}_{max}, \quad i \in \{k, k + N_u - 1\}, \quad (4.15)$$

\mathbf{u}_{min} a \mathbf{u}_{max} jsou jednotlivé nastavené omezení pro danou situaci. Index i je v rozměrech, který je uveden v závorkách vedle vztahu. Jinak řečeno \mathbf{u} nabývá hodnoty od času k do času $k + N_u - 1$, kde N_u označuje horizont řízení.

Obecná podoba vztahu při konečném čase je

$$\mathbf{u}_{min} \leq \mathbf{u}(k + N_u - 1) = \mathbf{u}(k - 1) + \Delta\mathbf{u}(k) + \Delta\mathbf{u}(k + 1) + \dots + \quad (4.16)$$

$$\Delta\mathbf{u}(k + N_u - 1) \leq \mathbf{u}_{max}.$$

V maticové formě vypadají minimální omezení následovně

$$\underbrace{\begin{bmatrix} -\mathbf{I}_{nu \times nu} & \cdots & \mathbf{0}_{nu \times nu} \\ \vdots & \ddots & \vdots \\ -\mathbf{I}_{nu \times nu} & \cdots & -\mathbf{I}_{nu \times nu} \end{bmatrix}}_{A_{umin}} \mathbf{U} \leq \underbrace{\begin{bmatrix} -\mathbf{I}_{nu \times nu} \\ \vdots \\ -\mathbf{I}_{nu \times nu} \end{bmatrix}}_{b_{umin}} (\mathbf{u}_{min} - \mathbf{u}(k - 1)), \quad (4.17)$$

a omezení maxima jsou definované jako

$$\underbrace{\begin{bmatrix} \mathbf{I}_{nu \times nu} & \cdots & \mathbf{0}_{nu \times nu} \\ \vdots & \ddots & \vdots \\ \mathbf{I}_{nu \times nu} & \cdots & \mathbf{I}_{nu \times nu} \end{bmatrix}}_{A_{umax}} \mathbf{U} \leq \underbrace{\begin{bmatrix} \mathbf{I}_{nu \times nu} \\ \vdots \\ \mathbf{I}_{nu \times nu} \end{bmatrix}}_{b_{umax}} (\mathbf{u}_{max} - \mathbf{u}(k - 1)). \quad (4.18)$$

Obecně jsou limitace ve vztahu nerovnosti $\mathbf{AU} \leq \mathbf{b}$. Jednotlivé matice jsou tedy označeny podle takového vztahu (Honc, nedatováno).

Stejný princip platí pro změny akčních veličin. Na ně se vztahuje upravené vyjádření podle nerovnice 4.15

$$\Delta\mathbf{u}_{min} \leq \Delta\mathbf{u}(i) \leq \Delta\mathbf{u}_{max}, \quad i \in \{k, k + N_u - 1\}, \quad (4.19)$$

v maticovém zápisu dojde také k úpravě matice 4.17 pro minimální hranici

$$\underbrace{\begin{bmatrix} -\mathbf{I}_{nu \times nu} & \cdots & \mathbf{0}_{nu \times nu} \\ \vdots & \ddots & \vdots \\ \mathbf{0}_{nu \times nu} & \cdots & -\mathbf{I}_{nu \times nu} \end{bmatrix}}_{A_{\Deltaumin}} \mathbf{U} \leq \underbrace{\begin{bmatrix} -\mathbf{I}_{nu \times nu} \\ \vdots \\ -\mathbf{I}_{nu \times nu} \end{bmatrix}}_{b_{\Deltaumin}} \Delta\mathbf{u}_{min}, \quad (4.20)$$

A také matice 4.18 pro maximální hranici

$$\underbrace{\begin{bmatrix} \mathbf{I}_{nu \times nu} & \cdots & \mathbf{0}_{nu \times nu} \\ \vdots & \ddots & \vdots \\ \mathbf{0}_{nu \times nu} & \cdots & \mathbf{I}_{nu \times nu} \end{bmatrix}}_{A_{\Deltaumax}} \mathbf{U} \leq \underbrace{\begin{bmatrix} \mathbf{I}_{nu \times nu} \\ \vdots \\ \mathbf{I}_{nu \times nu} \end{bmatrix}}_{b_{\Deltaumax}} \Delta\mathbf{u}_{max}, \quad (4.21)$$

(Honc, nedatováno).

4.3.2 Omezení v případě regulovaných veličin

Nerovnost pro regulované veličiny se definují podmínkou. Ta musí být splněna

$$\mathbf{y}_{min} \leq \mathbf{y}(i) \leq \mathbf{y}_{max}, \quad i \in \{k + 1, k + N_2\}, \quad (4.22)$$

Hodnota \mathbf{y} je v rozmezí minimálního omezení \mathbf{y}_{min} a maximálního omezení \mathbf{y}_{max} . Index i dosahuje hodnot od času $k + 1$ do času $k + N_2$, kde N_2 je horizont sledování žádané hodnoty.

Z hlediska vnučené a volné odezvy se nerovnost vyjádří jako

$$\mathbf{Y}_{min} \leq \mathbf{Y} = \mathbf{G}\mathbf{U} + \mathbf{F}_p \mathbf{x}_p(k) \leq \mathbf{Y}_{max}. \quad (4.23)$$

V maticovém zápisu vypadá minimální hranice jako

$$\underbrace{-\mathbf{G}}_{\mathbf{A}_{ymin}} \mathbf{U} \leq \underbrace{\begin{bmatrix} -\mathbf{I}_{ny \times ny} \\ \vdots \\ -\mathbf{I}_{ny \times ny} \end{bmatrix} \mathbf{y}_{min} + \mathbf{F}_p \mathbf{x}_p(k)}_{\mathbf{b}_{ymin}}, \quad (4.24)$$

a maximální hranice je zobrazena jako

$$\underbrace{\mathbf{G}}_{\mathbf{A}_{ymax}} \mathbf{U} \leq \underbrace{\begin{bmatrix} \mathbf{I}_{ny \times ny} \\ \vdots \\ \mathbf{I}_{ny \times ny} \end{bmatrix} \mathbf{y}_{max} - \mathbf{F}_p \mathbf{x}_p(k)}_{\mathbf{b}_{ymax}}, \quad (4.25)$$

(Honc, nedatováno).

4.3.3 Omezení v případě stavových veličin

Poslední typ veličiny, které můžeme limitovat jsou stavové veličiny. Tyto veličiny jdou limitovat pouze v případě stavového popisu, jelikož se ve vstupně-výstupním modelu nevyskytují. Znovu pro ně musí platit podmínka, která je dána jako

$$\mathbf{x}_{min} \leq \mathbf{x}_p(i) \leq \mathbf{x}_{max}, \quad i \in \{k+1, k+N_2\}, \quad (4.26)$$

kde platí stejná situace jako u omezení regulovaných veličin z kapitoly 4.3.2. Index i dosahuje stejných hodnot jako v nerovnici 4.22.

Obecně vyjádření vztahu pro konečnou dobu vypadá jako

$$\mathbf{x}_{min} \leq \mathbf{x}_p(k+N_2) = \mathbf{M}^{N_2} \mathbf{x}_p(k) + \mathbf{M}^{N_2-1} \mathbf{N} \Delta \mathbf{u}(k) \mathbf{M}^{N_2-2} \quad (4.27)$$

$$\mathbf{N} \Delta \mathbf{u}(k+1) + \dots + \mathbf{N} \Delta \mathbf{u}(k+N_2-1) \leq \mathbf{x}_{max}$$

Výsledné maticové vyjádření pro minimální hranici je

$$\underbrace{\begin{bmatrix} -\mathbf{N} & \mathbf{0} & \dots & \mathbf{0} \\ -\mathbf{M}\mathbf{N} & -\mathbf{N} & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ -\mathbf{M}^{N_2-1} & -\mathbf{M}^{N_2-2} \mathbf{N} & \dots & -\mathbf{M}^{N_2-N_u} \mathbf{N} \end{bmatrix}}_{\mathbf{A}_{xmin}} \mathbf{U} \leq \quad (4.28)$$

$$\underbrace{\begin{bmatrix} -\mathbf{I}_{nx \times nx} \\ -\mathbf{I}_{nx \times nx} \\ \vdots \\ -\mathbf{I}_{nx \times nx} \end{bmatrix} \mathbf{x}_{min} + \begin{bmatrix} \mathbf{M} \\ \mathbf{M}^2 \\ \vdots \\ \mathbf{M}^{N_2} \end{bmatrix} \mathbf{x}_p(k)}_{\mathbf{b}_{xmin}},$$

Pro maximální rozmezí bude maticový zápis vypadat jako

$$\underbrace{\begin{bmatrix} \mathbf{N} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{MN} & \mathbf{N} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{M}^{N_2-1} & \mathbf{M}^{N_2-2}\mathbf{N} & \cdots & \mathbf{NM}^{N_2-N_u}\mathbf{N} \end{bmatrix}}_{\mathbf{A}_{xmax}} \mathbf{U} \leq \quad (4.29)$$

$$\underbrace{\begin{bmatrix} \mathbf{I}_{nx \times nx} \\ \mathbf{I}_{nx \times nx} \\ \vdots \\ \mathbf{I}_{nx \times nx} \end{bmatrix}}_{\mathbf{b}_{xmax}} \mathbf{x}_{min} + \underbrace{\begin{bmatrix} \mathbf{M} \\ \mathbf{M}^2 \\ \vdots \\ \mathbf{M}^{N_2} \end{bmatrix}}_{\mathbf{b}_{xmax}} \mathbf{x}_p(k),$$

V kapitole 4.3.1 je uveden výsledná nerovnost podle které jsou dány omezení. Matice \mathbf{A} a vektor \mathbf{b} tedy obsahují komplexně všechny zvolené omezení. Jsou naplněny všemi prvky jednotlivých typů limitů. Jak vypadají je uvedeno následovně

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_{umax} \\ \mathbf{A}_{umin} \\ \mathbf{A}_{\Delta umax} \\ \mathbf{A}_{\Delta umin} \\ \mathbf{A}_{xmax} \\ \mathbf{A}_{xmin} \\ \mathbf{A}_{ymax} \\ \mathbf{A}_{ymin} \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} \mathbf{b}_{umax} \\ \mathbf{b}_{umin} \\ \mathbf{b}_{\Delta umax} \\ \mathbf{b}_{\Delta umin} \\ \mathbf{b}_{xmax} \\ \mathbf{b}_{xmin} \\ \mathbf{b}_{ymax} \\ \mathbf{b}_{ymin} \end{bmatrix},$$

(Honc, nedatováno).

5 REALIZACE PREDIKTIVNÍHO REGULÁTORU

Prediktivní regulátor vychází z rovnic a vztahů v kapitole 4. Z této kapitole víme, že se jedná o stavovou verzi. Je v něm zakomponovaná porucha veličina. Zároveň je v regulaci počítáno s omezeními regulované a akční veličiny. Samotnou realizaci jsem řešil softwarově ve vývojovém prostředí MATLAB.

5.1 VÝVOJOVÉ PROŠŘEDÍ MATLAB

MATLAB je vývojové prostředí, kde je možno pracovat s různými programovacími jazyky. Například C, C++ nebo Python. Jedná se o matematické a programovací prostředí pro zpracování různých signálů, dat a informací pro použití ve vědeckých a inženýrských aplikacích. Typy jednotlivých aplikací jsou nabídnuty ve velké rozmanitosti.

Příklady aplikací jsou

- analýza dat,
- obrazové a signálové zpracování,
- řídicí systémy,
- bezdrátová komunikace,
- robotika,
- umělá inteligence a další.

V MATLABu jsou obsaženy různé programovací jazyky, specializované knihovny a interaktivní aplikace. Také je možno generování tzv. embedded kódu. Tento typ kódu je používán pro mikroprocesory v jazyce C nebo C++ (What Is MATLAB? – MATLAB & Simulink, nedatováno).

MATLAB není dostupný zdarma, ale uživatel musí vlastnit některou z licencí. Ty se liší cenově podle použití. Některé z licencí se liší v podpoře od firmy MathWorks, která vývojové prostředí vlastní. Ovšem lze využít 30-denní verzi zdarma na zkoušku. Každý půl rok jsou vydávány aktualizace. S každým novým rokem je vydávána verze s novým číselným označením. Firma MathWorks nabízí akademické licence pro školy, které jsou využívány studenty. Studentské licence patří mezi nejlevnější ze všech. Vývojové prostředí lze instalovat na Windows, distribuce Linuxu i MacOS. Při psaní této práce je nejnovější verze MATLABu R2024b. V roce 2025 bude vycházet pravděpodobně verze R2025a.

Nadstavba k MATLABu je simulační prostředí Simulink. Ten lze použít samostatně, ale také je možno naprogramovat nebo načíst data do skriptu v MATLABu a poté je nainportovat do simulačního schématu.

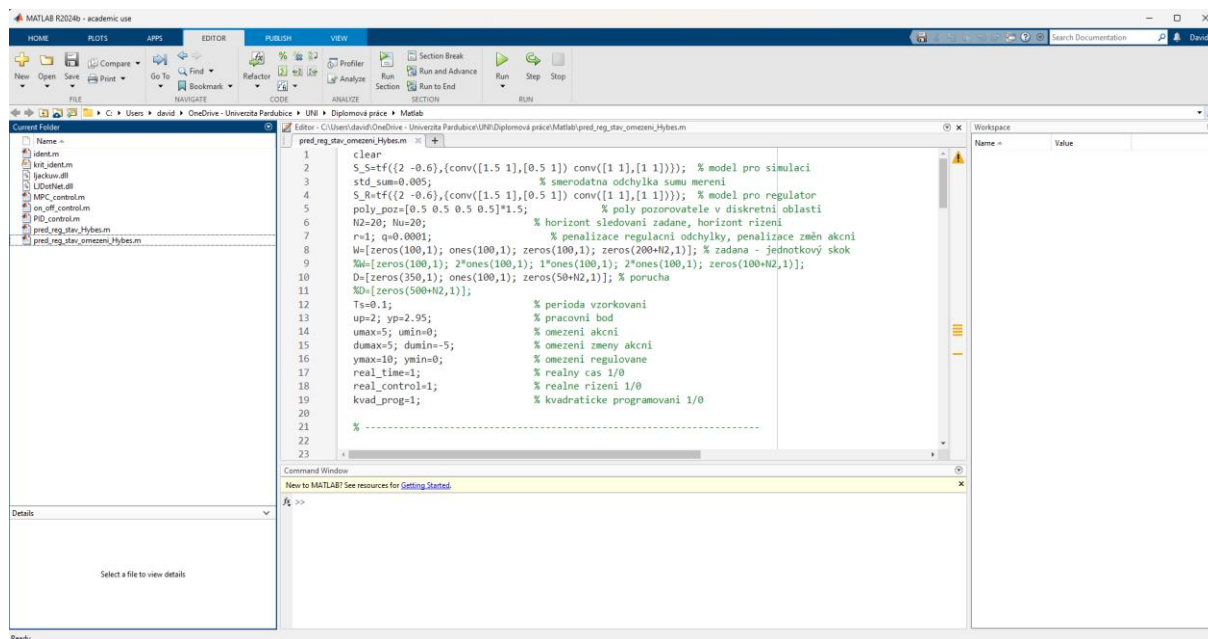
Toto vývojové prostředí je používáno v široké škále oborů jako třeba automatizace, letecké inženýrství, energetika, komunikace nebo pro lékařské zařízení a další. Obě zmíněné části jsou používány pro možnost programování a modelování, ale také pro výzkum a vývoj. Příklady vývoje a výzkumů v projektech jsou

- autonomní vozidla,
- elektrické dopravní prostředky,
- průmyslové zařízení,
- standardizované komunikační vybavení
- obnovitelné zdroje energie,
- chytré lékařské zařízení a další (What Is MATLAB? – MATLAB & Simulink, nedatováno).

Instalace je velmi jednoduchá. Stáhne se instalační soubor, který provede uživatele celým procesem. Nejdříve je potřeba přihlášení k účtu MathWorks a zkontrolování licence. Poté nastává výběr aplikací, které chce uživatel nainstalovat. Poslední krok je vyčkání na konec instalace.

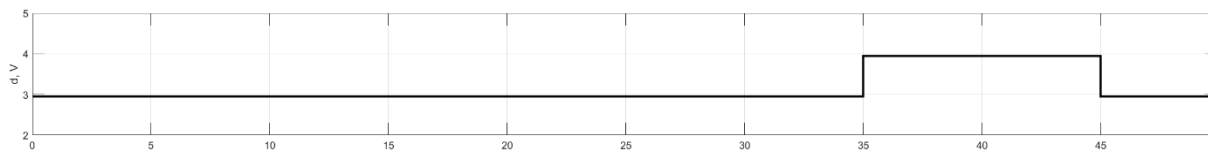
Při zapnutí prostředí uvidíme úvodní obrazovku MATLABu. Je možné zde zadávat příkazy do funkčního řádku nebo si zakládat tzv. skripty a vytvářet oddělené programy, které se následně ukládají ve formátu s koncovkou .m. Prostředí se skriptem lze vidět na obrázku 5.1.

V levé části prostředí je vidět vybraná složka, kde jsou uloženy například skripty nebo dodatečné soubory odkud otevřený skript bere data. Pro správnou funkci čerpání dat musí být zvolena složka, kde jsou uloženy jako cílová. Uprostřed je vidět samotný skript, ve spodní části je funkční řádek, kde se mohou psát separátní příkazy nebo volat jednotlivé skripty. A v pravé části jsou po spuštění skriptu načtené jednotlivé proměnné a jejich hodnota. Ve vrchní části je panel nástrojů.



Obrázek 5.2 – Prostředí MATLABu s otevřeným skriptem

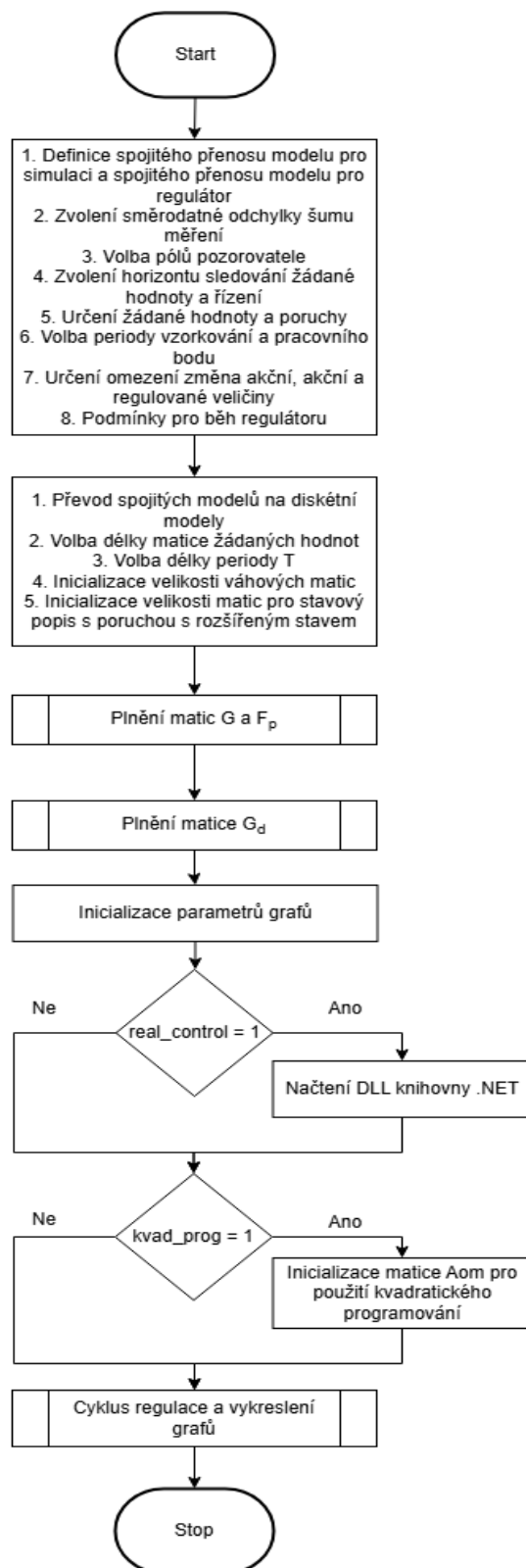
I bez použití Simulinku může MATLAB zobrazovat průběhy různých měřených veličin či dat. Taková vizualizace může vypadat jako na obrázku 5.2.



Obrázek 5.1 – Příklad vizualizace dat v MATLABu

Styl programování, který je ukázán na obrázku 5.1 je sice procedurální, ale MATLAB dokáže pracovat i s objektově zaměřeným programováním. Tam by se použila definice tříd „classdef“.

5.2 ALGORITMUS PREDIKTIVNÍHO REGULÁTORU



Obrázek 5.3 – Vývojový diagram hlavní části regulátoru

Celý program je popsán pomocí vývojových diagramů. Vývojové diagramy jsou rozděleny, tak aby co nejlépe popsaly operace regulátoru. Také zajišťují přehlednost funkcí.

První vývojový diagram, který je vidět na obrázku 5.3, ukazuje celkovou funkčnost regulátoru. Na vytvoření programu jsem použil procedurální programování. Nejdříve se určí proměnné a nastavení regulátoru, podle čeho má pracovat. Poté přichází práce s modely a inicializace velikosti jednotlivých matic.

Pro chod regulátoru je potřeba jednotlivé matice správně naplnit daty. Tento postup popisuje vývojový diagram na obrázku 5.4. Zde se plní matice G a F_p . Matice G je součástí vnučené odezvy a matice F_p je součástí volné odezvy. Plnění jsem řešil přes cyklus *for*, který se každým krokem zvyšuje o jedničku až do hranice horizontu žádané hodnoty N_2 , který se volí na začátku programu. Nejdříve se pracuje s pomocnou maticí G_v , která plní postupně sloupce do matice G . Matice G je postupně plněna díky druhému cyklu *for*, který má stejné omezení jako první cyklus. Nakonec se určí omezení počtu sloupců matice G , na horizont řízení N_u , aby program fungoval bez chyb. Pokud by toto omezení nenastalo, tak se program bude snažit matici číst přes hranici a vytvoří se chyba a program se nedokončí.

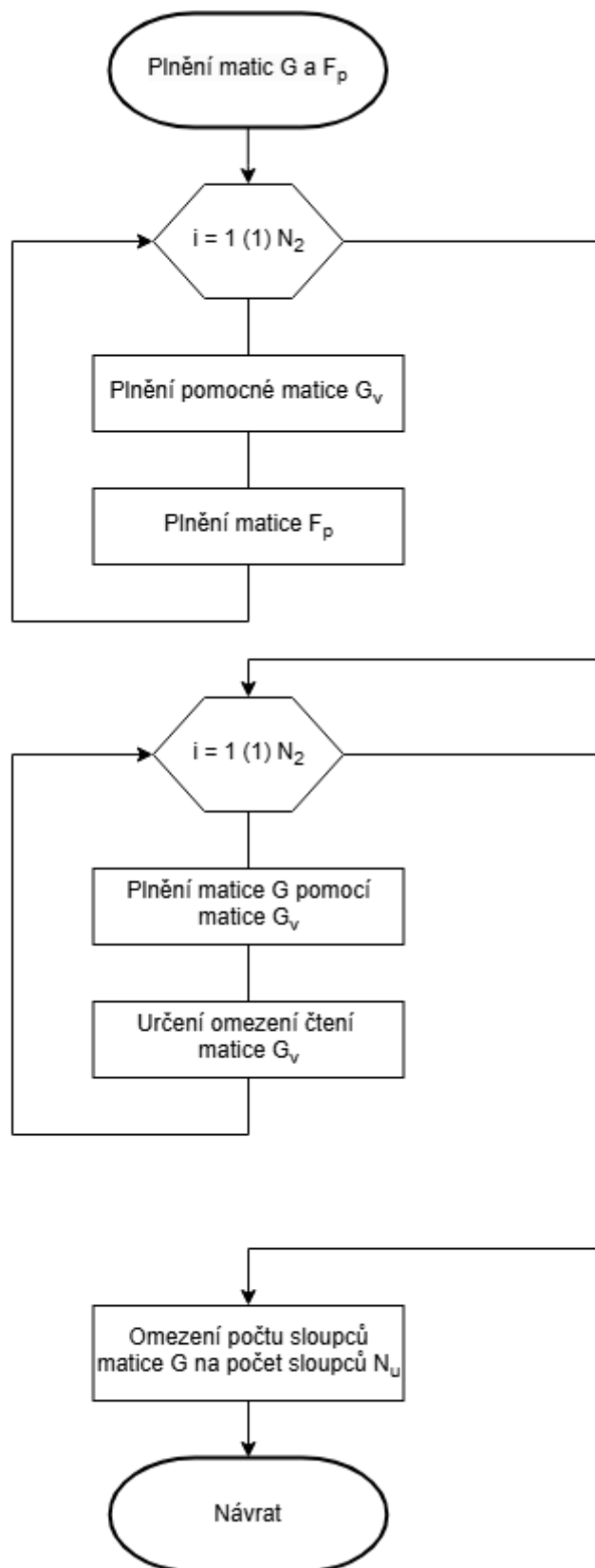
Jelikož regulátor počítá i s poruchou, tak stejný proces naplnění dochází i pro matici G_d . Tento proces je vyobrazen na obrázku 5.5, kde je vidět příslušný vývojový diagram. Plní se zde pouze jedna matice, takže vývojový diagram neobsahuje více cyklů, ale pouze jeden. Funguje zde stejný princip plnění a omezení jako na obrázku 5.4. Pomocná matice je označena jako G_{vd} .

Jelikož jsou grafy průběhu na konci vykreslovány, tak dalším krokem programu je inicializace parametrů jednotlivých grafů.

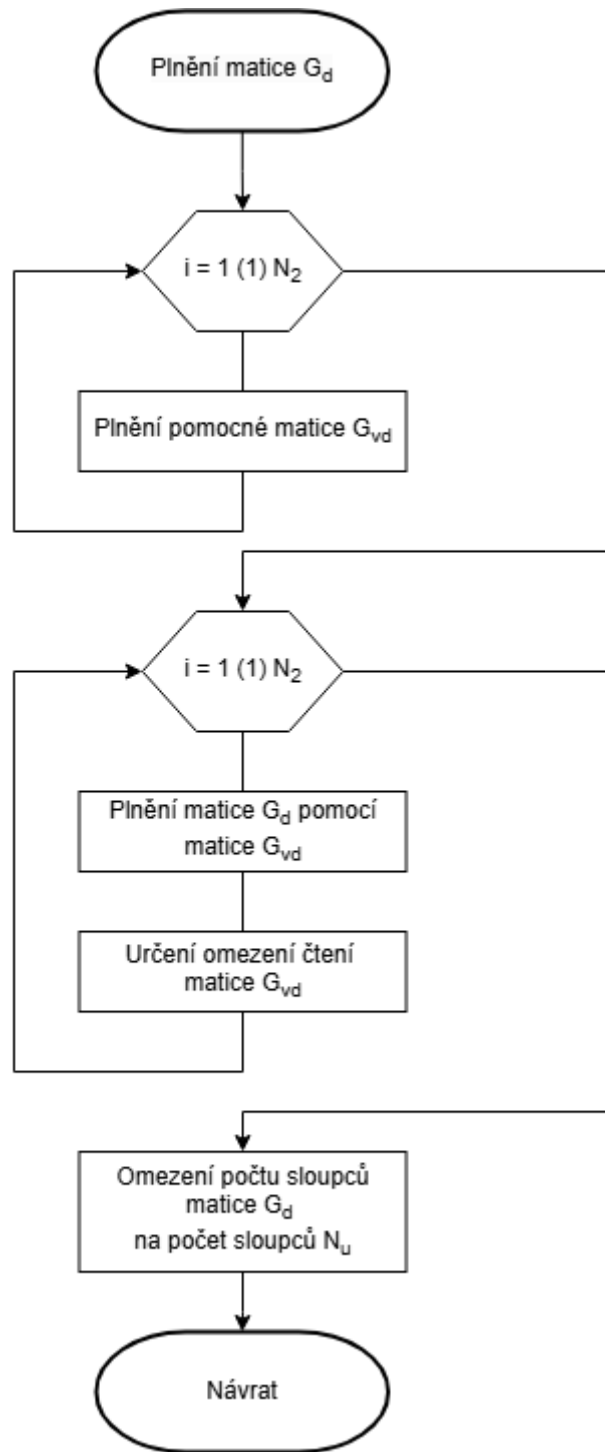
Pro potřeby simulace není tak důležité regulovat soustavu v reálném času, ale při použití reálné soustavy se tento řízení v reálném času využívá. K tomu slouží podmínka *if*, která načte potřebnou knihovnu z MATLABu.

Pro následnou funkci omezení, na kterou se využívá kvadratické programování se inicializuje velikost matice A_{om} . Kroky v jednotlivých podmínkách se vykonají, pokud na začátku programu tyto funkce nastavíme do jedničky. Pokud se tak nestane, tak program pracuje jako stavový regulátor bez omezení a bez sledování v reálném času.

Největší blok pro zpracování výpočtů se nachází v bloku „Cyklus regulace a vykreslení grafů“.

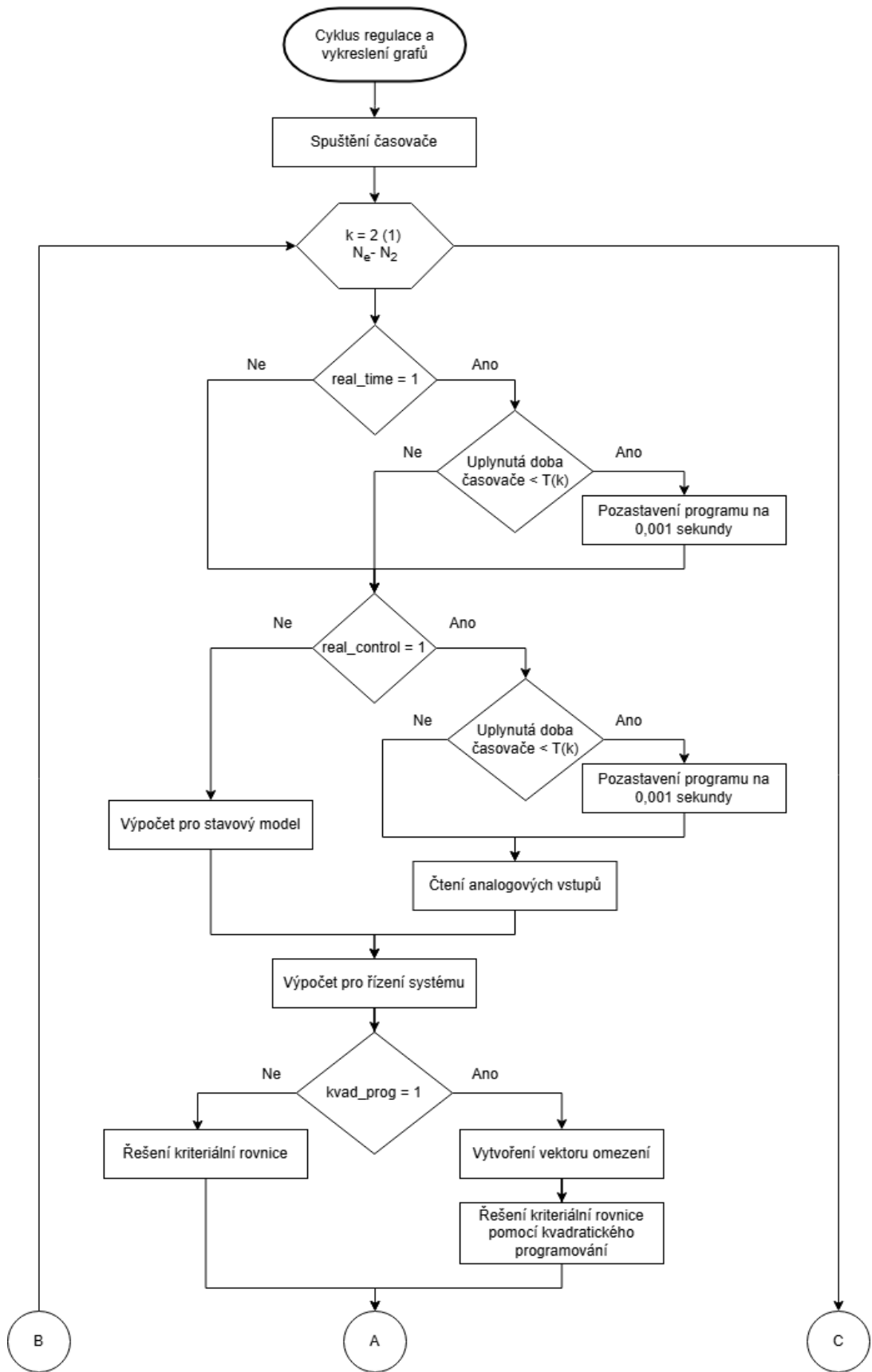


Obrázek 5.4 – Vývojový diagram pro plnění matic G a F_p

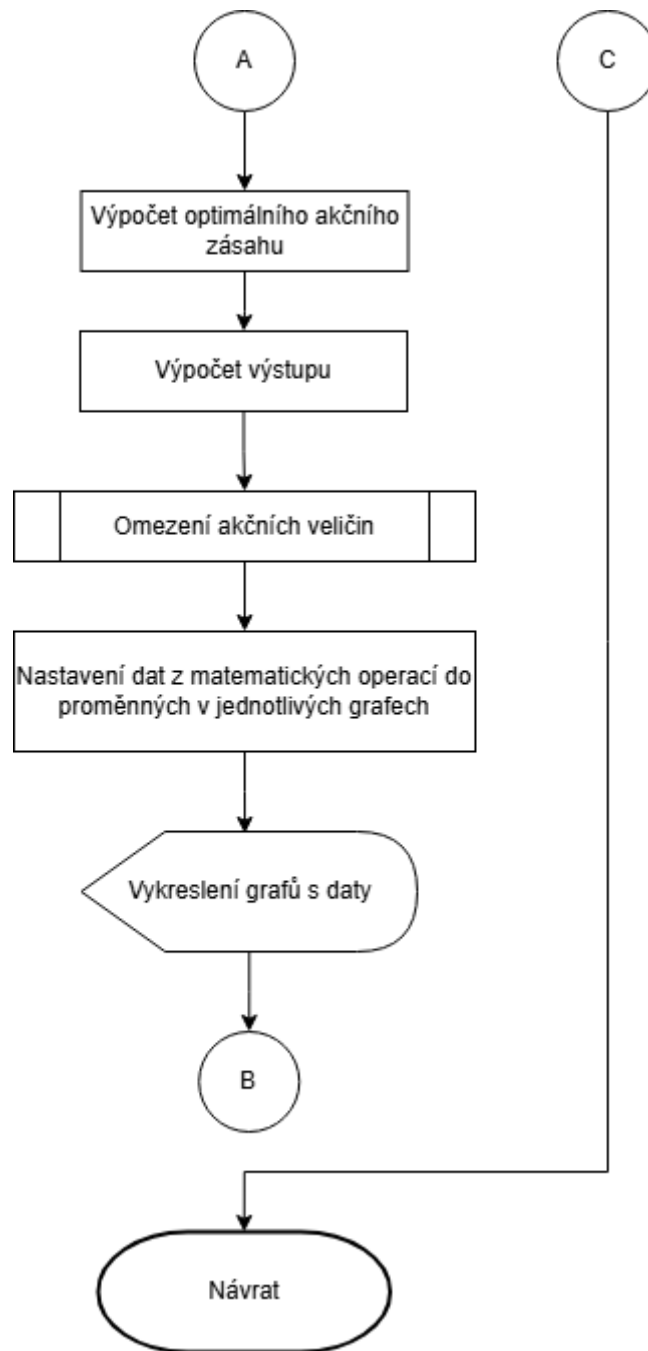


Obrázek 5.5 – Vývojový diagram plnění matice

G_d



Obrázek 5.6 – Vývojový diagram cyklu regulace a vykreslení grafů, první část



Obrázek 5.7 – Vývojový digram cyklu regulace a vykreslení grafů, druhá část

Cyklus regulace a vykreslení grafů je ukázán na vývojovém diagramu z obrázků 5.6 a 5.7. Tento vývojový diagram je rozdělen na dva samostatné kvůli jeho složitosti.

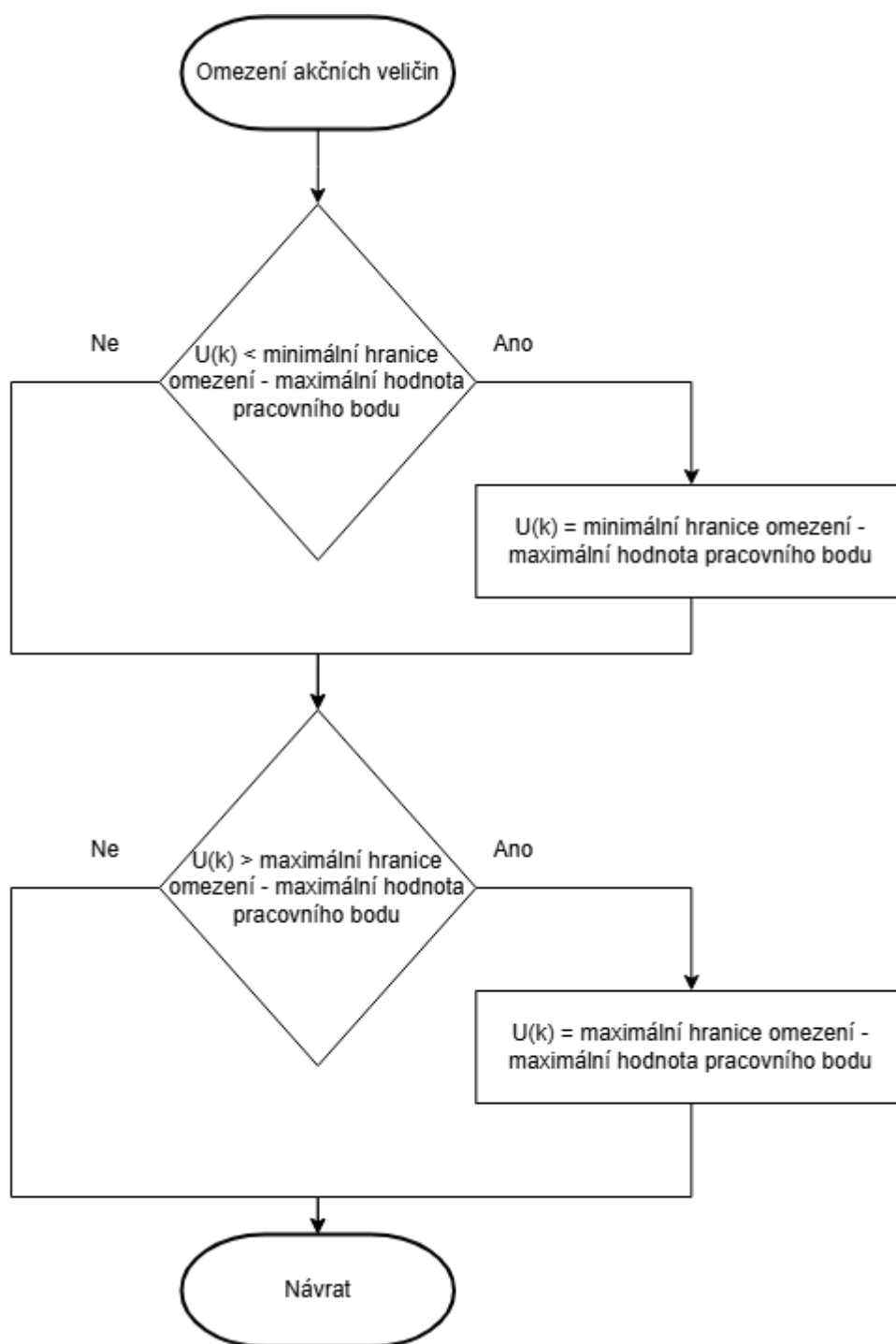
Před samotným cyklem *for*, který zajišťuje všechny procesy se zapne časovač příkazem *tic*. Poté je definovaný cyklus *for*, který začíná na časovém kroku 2 až do stavu $N_e - N_2$. Hodnota N_e je délka matice žádaných hodnot W . Pokud by nepočítal konec cyklu s pomocí hodnoty N_e , tak by se dostal program do chyby.

V programu existuje možnost spuštění řízení v reálném čase. Při spuštění není tedy regulace zrychlená, ale běží podle reálného času, což je potřebné pro řízení reálné soustavy, a ne jenom simulace. V programu je tato funkce řešena podmínkou *if*. Rozhoduje, zda je proměnná *real_time* v hodnotě 1 nebo 0. Pokud je v hodnotě 1, tak se přechází na další podmínku *if*, zda je uplynulý čas menší než nastavená perioda v proměnné $T(k)$. Uplynulý čas se určí podle příkazu *toc*, který odměří čas od času příkazu *tic*, kdy se spustil časovač. Pokud je podmínka platná, tak se program pozastaví na 0,001 sekundy. Tato akce umožňuje synchronizaci s reálným časem. Pokud není podmínka uplynulého času splněna, tak program pokračuje dále beze změn. Pokud je proměnná *real_time* v hodnotě 0, tak se také nic nestane.

Další podmínka *if* řeší proměnnou *real_control*, která zajišťuje možnost ovládání reálné soustavy, pokud je v hodnotě 1 nebo pouze spuštění simulace v hodnotě 0. Při ovládání reálné soustavy je nejdříve uvedena podmínka pro synchronizaci času stejně jako u kladné větve podmínky pro reálný čas. Poté se začnou číst analogové vstupy, které jsou připojené z reálné soustavy. V jednotlivých příkazech analogových vstupů jsou nastavené bity pro naše požadované funkce podle návodu ke konkrétní soustavě. V případě tohoto programu se jedná o soustavu, kde se regulují otáčky motoru. Pokud je proměnná *real_control* v hodnotě 0 a pouští se pouze simulace, tak se vypočtou hodnoty podle stavového popisu, který je popsán v kapitole 4.2.1. Následují rovnice pro výpočet regulace soustavy.

Další podmínka nám zajišťuje možnost omezení různých veličin pomocí tzv. kvadratického programování. Jedná se o optimalizační metodu, která řeší kvadratickou rovnici. Respektive hledá její minima nebo případně maxima. V MATLABu je zprostředkována funkcí *quadprog*. Proměnné, které patří do argumentů funkce jsme již nastavili na začátku programu. Pokud je proměnná *kvad_prog* v hodnotě 1, tak se spustí funkce kvadratického programování, kde se nejdříve vytvoří vektor omezení pro funkci a pak se hledá pomocí *quadprog* minimum kritériální funkce. Pokud je proměnná *kvad_prog* v hodnotě 0, tak regulace funguje bez omezení a kritériální funkce se vyřeší analyticky.

Pokračování cyklu regulace a vykreslení grafů je na obrázku 5.7, kde cyklus *for* začíná od bodu A. Zde je vypočítán optimální akční zásah a poté výstup. Jako v případě celého cyklu jsou tyto výpočty omezeny na konečnou hodnotu, kde jeden z rozhodujících prvků je hodnota N_2 .



Obrázek 5.8 – Vývojový diagram omezení akčních veličin

Poslední menší část vývojového diagramu, co je vidět na obrázku 5.7 jsou podmínky pro omezení akčních veličin. Tyto podmínky jsou vyobrazeny na obrázku 5.8. Skládá se z dvou podmínek *if*, které jsou jednoduššího rázu.

Obě podmínky fungují na stejném principu. První je pro omezení minimální hranice a druhá je pro omezení maximální hranice. Pracuje se zde s vrchní hodnotou pracovního bodu,

který je označen proměnnou up . Minimální a maximální hranice se nastavuje na začátku programu.

V první podmínce platí, že pokud je hodnota matice U v časovém bodě k menší, než je hodnota určená rozdílem minimální hranice omezení a proměnnou up , tak se nastaví právě na hodnotu rozdílu. Když je hodnota matice větší, tak se neděje žádná akce.

Ve druhé podmínce platí, že pokud je hodnota matice U v časovém bodě k větší, než je hodnota určená rozdílem maximální hranice omezení a proměnnou up , tak se nastaví právě na hodnotu rozdílu. Když je hodnota matice větší, tak se neděje žádná akce.

Z obrázku 5.7 vidíme, že cyklus pokračuje nastavením dat do proměnných pro grafy. Data jsou získány z různých výpočtů, a nakonec jsou vykresleny příslušné grafy. Jedná se o regulaci výstupní veličiny soustavy, hodnoty akční veličiny a zatížení na výpočetní zařízení.

Nakonec se přes bod B vrací do první části vývojového diagramu z obrázku 5.6, kde se uzavírá cesta cyklu *for* a přes bod C se vrací do vývojového diagramu z obrázku 5.3, kde se ukončuje celý program.

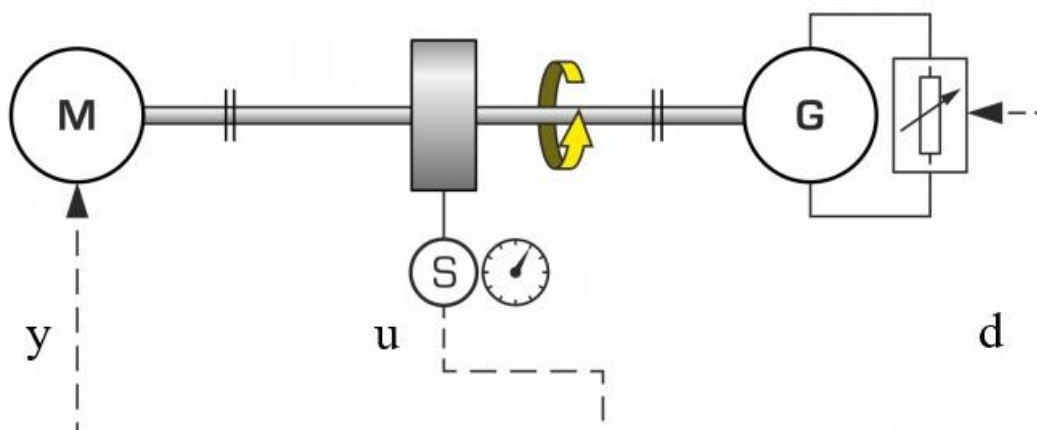
6 LABORATORNÍ APLIKACE PREDIKTIVNÍHO REGULÁTORU

Po návrhu regulátoru následuje ověření regulátoru. Pro ověření funkčnosti jsem zvolil metody simulace a experimentální ověření na reálné soustavě. Měření jsem provedl na laboratorní úloze pomocí soustavy RT 050 od firmy GUNT. Jedná se o soustavu, na které je regulována rychlost pomocí různých typů regulátorů.

Nejdříve jsem udělal simulační ověření ve vývojovém prostředí MATLAB. Experimentální ověření na laboratorní úloze jsem uskutečnil navrženým prediktivním regulátorem a porovnal jsem se standardním PID regulátorem. Měření na prediktivním regulátoru jsem provedl nejdříve bez informace o poruše. Poté jsem změnil variantu s informací o poruše, a nakonec s informací o predikované poruše.

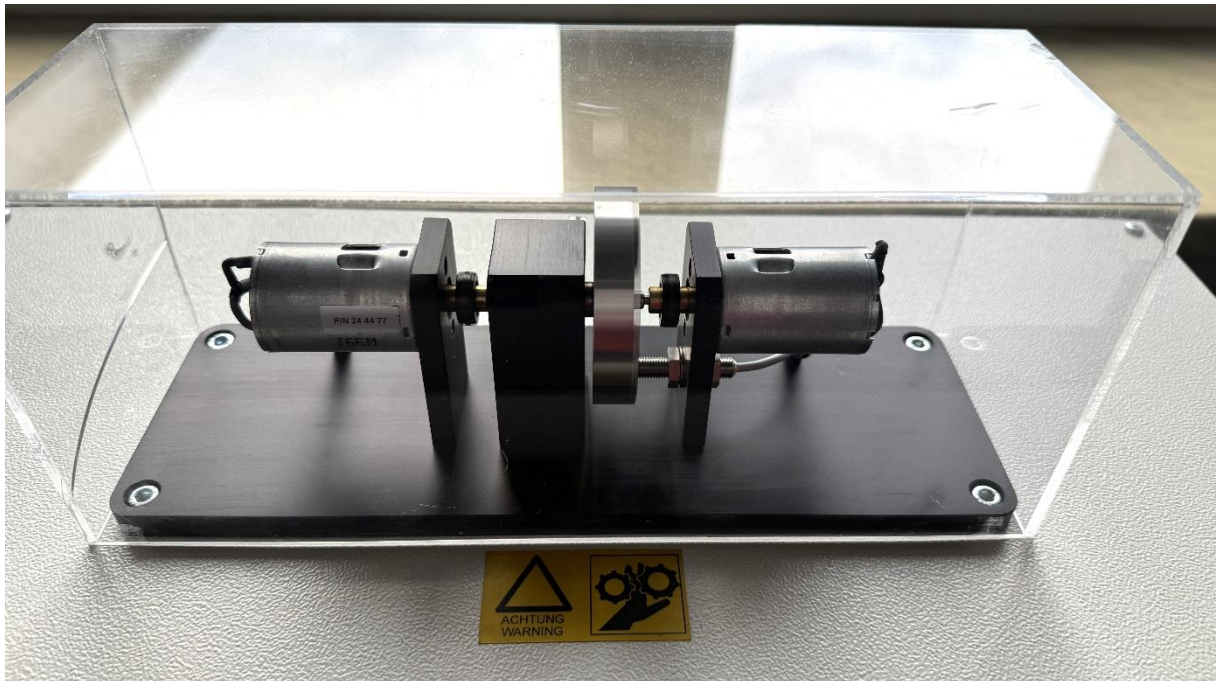
6.1 SOUSTAVA GUNT RT 050

Soustava RT 050 od firmy GUNT je navržena jako výuková soustava pro řízení rychlosti. Na obrázku 6.1 je vidět celkové zapojení soustavy.



Obrázek 6.1 – Zapojení soustavy GUNT RT 050 (RT 050 Training system speed control, HSI; nedatováno)

Řízení probíhá na soustavě hřídele se setrvačnickem. Rychlost odpovídá regulované veličině a je měřena indukčním senzorem. Výstupní signál ze senzoru jde do softwarového regulátoru, který následně působí akční veličinou na motor připojený k hřídeli.

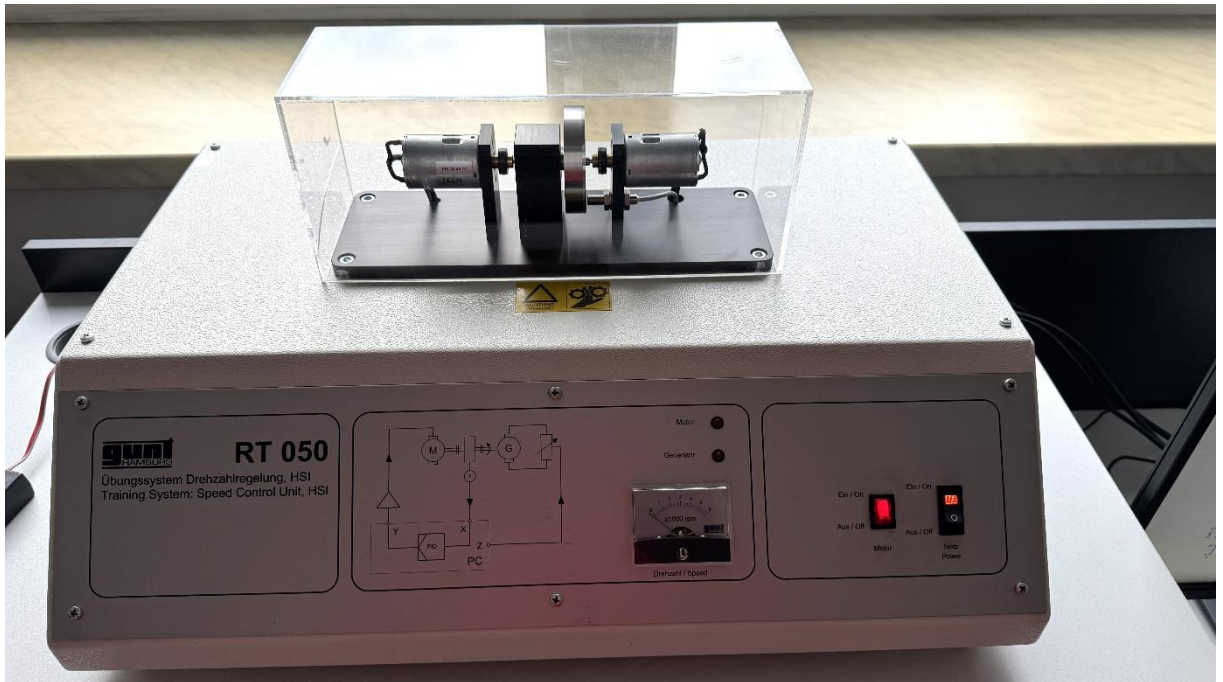


Obrázek 6.2 – Soustava GUNT RT 050 – detail

Na druhé straně hřídele působí v opačném směru rotace motor, ke kterému jsou připojeny proměnné odpory jako zátěž, to zajišťuje možnost volby různé intenzity poruchy působící na regulovanou veličinu. Na setrvačnicku je umístěn rychlostní snímač, který umožňuje čtení rychlosti v reálném čase. Okolo řízeného systému je umístěný ochranný box, aby nedošlo k poškození a měřicí experimenty byly sledovány bezpečně.

V řadě produktů do které patří i tato soustava je obsažena hardwarová/softwarová integrace, tzv. HSI. Soustava a externí PC, kde je nahrán software regulátoru, jsou propojeny skrze USB.

Díky vlastnostem soustavy jsou vidět akce v reálném čase a jednoduše lze identifikovat změny působení poruchy nebo nastavení regulátoru (RT 050 Training system speed control, HSI; nedatováno).



Obrázek 6.3 – Soustava GUNT RT 050

Na obrázku 6.3 je vidět celá soustava RT 050 v zapnutém stavu před spuštěním regulátoru z externího PC.

6.2 IDENTIFIKACE SOUSTAVY

Pro samotné měření bylo potřeba nejprve provést identifikaci modelu soustavy pomocí experimentální metody. Výsledkem identifikace je matematický model soustavy v podobě přenosu. Pro identifikaci byl zvolen pracovní bod, jehož hodnoty jsou uvedeny v tabulce 6.1.

Tabulka 6.1 – Hodnoty pracovního bodu

Pracovní bod	Hodnota	Veličina
Akční veličina u_0	2,5	V
Výstupní veličina y_0	4	V

Pro určení správného modelu je potřeba připojit soustavu k PC, ve kterém se bude spouštět skript MATLABu. Nejprve se zvolí průběh vstupní veličiny – průběh napětí po dobu 35 sekund. Tyto průběhy mají tvar jako jeden skok o 0,5 V a druhý o 1 V. Nakonec se napětí nastaví zpátky do pracovního bodu 2,5 V. Každá skoková změna probíhá po dobu 10 sekund.

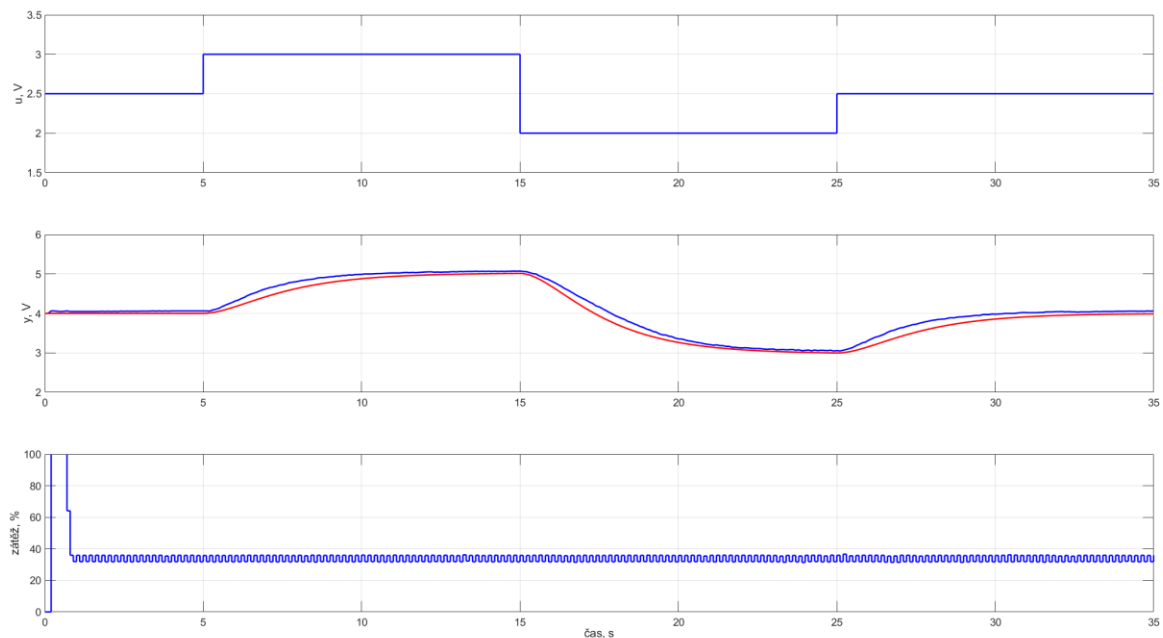
Je změřena odezva soustavy – napětí odpovídající výstupu soustavy. Nakonec software vypočte pomocí optimalizačního kritéria parametry modelu. Výsledný matematický model je uveden podle obecné rovnice 6.1.

$$F(s) = \frac{k}{(T_1s + 1) + (T_2s + 1)} \quad (6.1)$$

kde k – parametr zesílení,
 T_1 a T_2 – časové konstanty.

V konkrétním případě identifikace soustavy GUNT 050 vyšel matematický model podle rovnice 6.2.

$$F_s(s) = \frac{2.05}{(0,98s + 1) + (1,89s + 1)} \quad (6.2)$$



Obrázek 6.4 – Průběhy identifikace modelu soustavy GUNT RT 050

Na obrázku 6.4 je vidět jsou vidět tři průběhy, které znázorňují různé veličiny při identifikaci modelu. Na prvním průběhu je znázorněná vstupní veličina. Na průběhu vstupní veličiny jsou vidět skokové změny napětí.

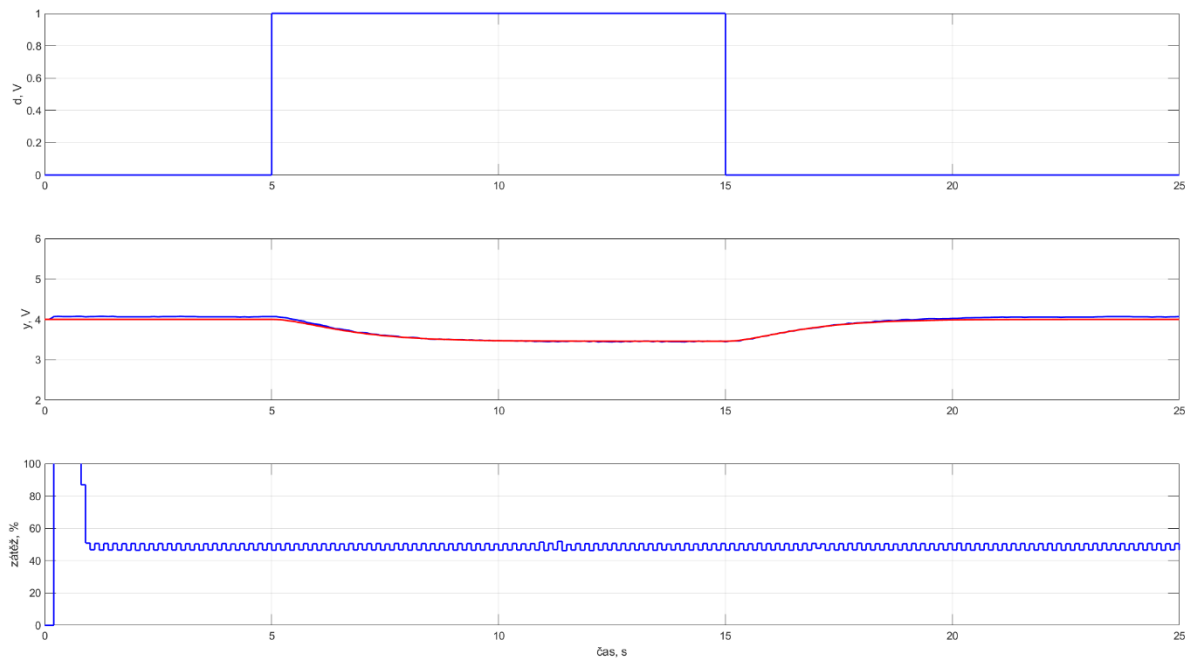
Na druhém průběhu jsou vidět dvě křivky. Znázorňují výstupní veličinu y . Modrá křivka označuje výstup ze soustavy, který je získán z analogového vstupu. Červená křivka je vypočtený průběh modelu soustavy. Na ose y je napětí uvedeno ve voltech.

Na posledním grafu je ukázán průběh, který zobrazuje, jak velká je výpočetní náročnost algoritmu měření vyjádřená v procentech doby strávené v programu ku periodě vzorkování. Na ose x je čas, který je uveden v sekundách.

Stejný proces identifikace modelu je proveden pro poruchu. Rozdíl je v nastavení analogových vstupů. Na jednotlivých vstupech lze nastavit síla poruchy, která je vyjádřena v procentech. Poruchy lze nastavit na úrovně 0 %, 33 %, 66 % a 100 %. V případě následné identifikace a měření je nastavena porucha na nejvyšší úroveň. Porucha má na průběh výstupu soustavy nejsilnější vliv.

Po procesu identifikace poruchy vyšel matematický model, který je vidět na rovnici 6.3.

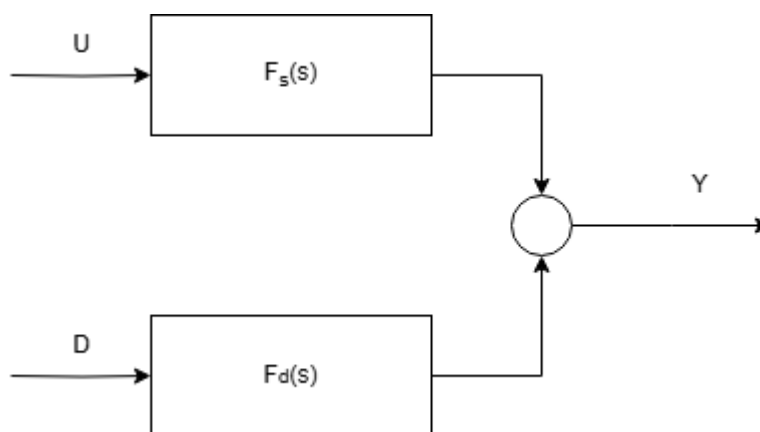
$$F_d(s) = \frac{-0,54}{(0,93s + 1) + (0,93s + 1)} \quad (6.3)$$



Obrázek 6.5 – Průběhy identifikace modelu poruchy

Na prvním grafu je vidět průběh poruchové veličiny d . Zde je vidět skok v napětí na který reaguje soustava. Reakce výstupní veličiny y je vidět na druhém grafu. Jsou zde znovu dvě křivky. Tyto křivky popisují stejné souvislosti jako v průbězích z obrázku 6.4. Rozdíl mezi křivkami je méně znatelný, jelikož porucha není tak silná. Na posledním grafu je uvedena výpočetní náročnost na PC.

Blokové schéma přenosů modelů poruchy a soustavy jsou vidět na obrázku 6.6.



Obrázek 6.6 – Blokové schéma přenosů soustavy a poruchy

V rovnici 6.4 je definice stavového popisu s rozšířeným stavem. V rovnici stavového popisu jsou obsaženy pět matic A , B , B_d , C , D .

$$\mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k) + \mathbf{B}\mathbf{u}(k) + \mathbf{B}_d\mathbf{d}(k) \quad (6.4)$$

$$\mathbf{y}(k) = \mathbf{C}\mathbf{x}(k) + \mathbf{D}\mathbf{u}(k),$$

(Honc, nedatováno).

Hodnota matic je vyobrazena v rovnici 6.5.

$$\mathbf{A} = \begin{bmatrix} 0,8540 & -0,0500 & 0 & 0 \\ 0,0926 & 0,9974 & 0 & 0 \\ 0 & 0 & 0,8015 & -0,1038 \\ 0 & 0 & 0,0898 & 0,9946 \end{bmatrix}, \quad (6.5)$$

$$\mathbf{B} = \begin{bmatrix} 0,0926 & 0 \\ 0,0047 & 0 \\ 0 & 0,0898 \\ 0 & 0,0047 \end{bmatrix},$$

$$\mathbf{C} = [0 \quad 1,1068 \quad 0 \quad -0,6243],$$

$$\mathbf{D} = [0 \quad 0].$$

6.3 PID REGULÁTOR

PID regulátor je použit k porovnání regulačních pochodů s prediktivní regulátorem. Byly provedeny simulace a laboratorní experiment.

Je použité standardní zapojení spojitého PID regulátoru. Ten je definován vztahem z rovnice 6.6.

$$u(t) = r_0 \left[e(t) + \frac{1}{T_I} \int_0^t e(\tau) d\tau + T_D \frac{de(t)}{dt} \right] \quad (6.6)$$

kde r_0 – zesílení,
 T_I – integrační časová konstanta,
 T_D – derivační časová konstanta,
 e – regulační odchylka.

Číslicová varianta PID regulátoru je definována vztahem z rovnice 6.7.

$$u(k) = u(k-1) + q_0 e(k) + q_1 e(k-1) + q_2 e(k-2) \quad (6.7)$$

kde u – akční veličina,
 e – regulační odchylka,
 T – vzorkovací perioda,
 $q_0 = r_0 \left(1 + \frac{T}{T_I} + \frac{T_D}{T} \right)$,
 $q_1 = -r_0 \left(1 + 2 \frac{T_D}{T} \right)$,
 $q_2 = r_0 \frac{T_D}{T}$,
(Šeda, 2007).

Chování PID regulátoru je definováno podle nastavených parametrů jednotlivých složek. Existují různé metody, jak nastavit tyto parametry. Parametry PID regulátory byly stanoveny metodou pokus-omyl. PID regulátor je použit jako porovnání k prediktivnímu regulátoru. Jednotlivé parametry jsou ukázány v tabulce 6.2.

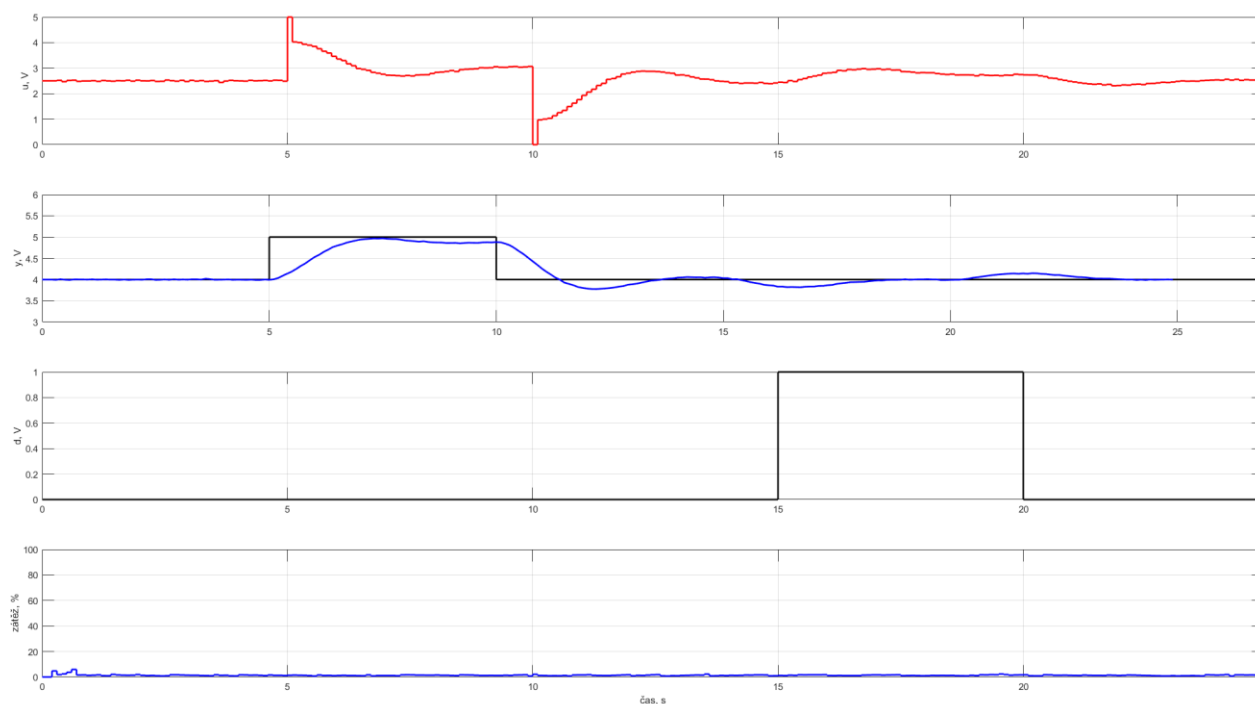
Tabulka 6.2 – Parametry PID regulátoru

Parametry	Hodnoty
Zesílení r_0	2
Integrační časová konstanta T_I	3
Derivační časová konstanta T_D	0,05

6.3.1 Experimenty s PID regulátorem

První experiment na PID regulátoru je simulace. V těchto grafech je vidět, jak se regulace chová v ideálním prostředí s poruchovou veličinou.

Na obrázku 6.7 jsou zobrazeny průběhy simulace na PID regulátoru.



Obrázek 6.7 – Průběhy regulace s PID regulátorem v simulaci

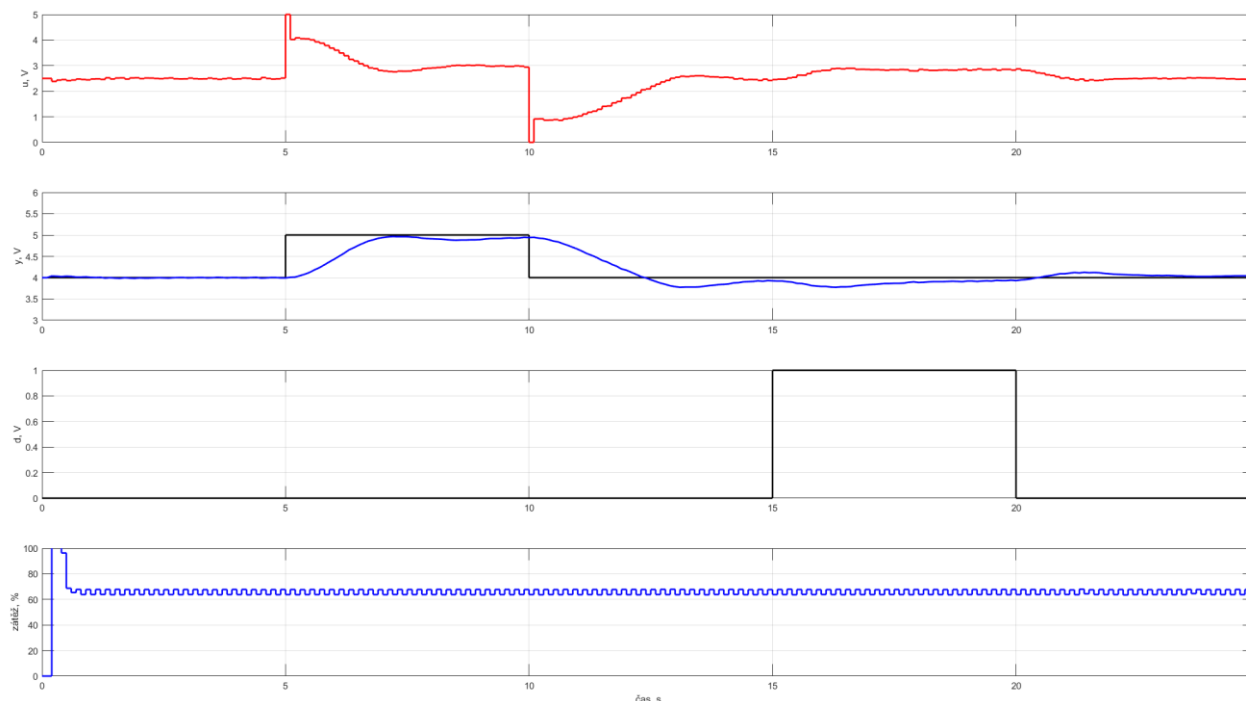
První průběh zobrazuje akční veličinu, kde jsou vidět prudké změny v místech, kde se mění žádaná hodnota což je způsobeno zejména použitím derivační složky regulátoru. Na poruchu reaguje regulátor plynuleji. Díky integračnímu charakteru nezanechává regulátor trvalou regulační odchylku.

Druhý průběh obsahuje dvě křivky, kde černá křivka ukazuje žádanou hodnotu. A modrá křivka je regulovaná veličina.

Ve třetím průběhu je vidět poruchová veličina, jejíž změny nastaly v druhé polovině simulačního experimentu.

Na posledním průběhu je vidět výpočetní náročnost algoritmu simulace na použitém PC.

Na obrázku 6.8 jsou zobrazeny průběhy regulace s připojenou soustavou GUNT RT 050.



Obrázek 6.8 – Průběhy regulace s PID regulátorem na soustavě GUNT RT 050

Průběhy žádané hodnoty a poruchy jsou identické, jak v simulaci. Stejně tak i parametry PID regulátoru. Regulační pochody jsou od simulace mírně odlišné, protože model nepopisuje reálnou soustavu dokonale. Výpočetní náročnost v případě reálného řízení je vyšší než u simulace. Je to způsobeno voláním funkcí pro měření regulované veličiny a aplikaci akčního zásahu.

6.4 PREDIKTIVNÍ REGULÁTOR

Pro ověření prediktivního regulátoru byly zvoleny 3 typy nastavení. Na všech typech byly provedeny experimenty v simulaci a následně na laboratorním zařízení. První typ prediktivního regulátoru byl bez omezení veličin ve variantách bez znalosti a se znalostí poruchy. Druhý typ prediktivního regulátoru byl s omezením veličin ve variantách bez znalosti a se znalostí poruchy. Pro všechny typy regulátoru byla zvolena jednotná perioda vzorkování $T_s = 0,05$ s.

Pro správnou funkci prediktivního regulátoru je potřeba mít k dispozici pozorovatel stavu, který má nastavené póly pozorovatele. Póly pozorovatele byla zvoleny tak, aby byl pozorovatel dostatečně rychlý vzhledem k dynamice soustavy a současně filtroval šum měření.

Tabulka 6.3 – Zvolené póly pozorovatele stavu

Póly	Hodnoty
P ₁	0,85
P ₂	0,85
P ₃	0,85
P ₄	0,85

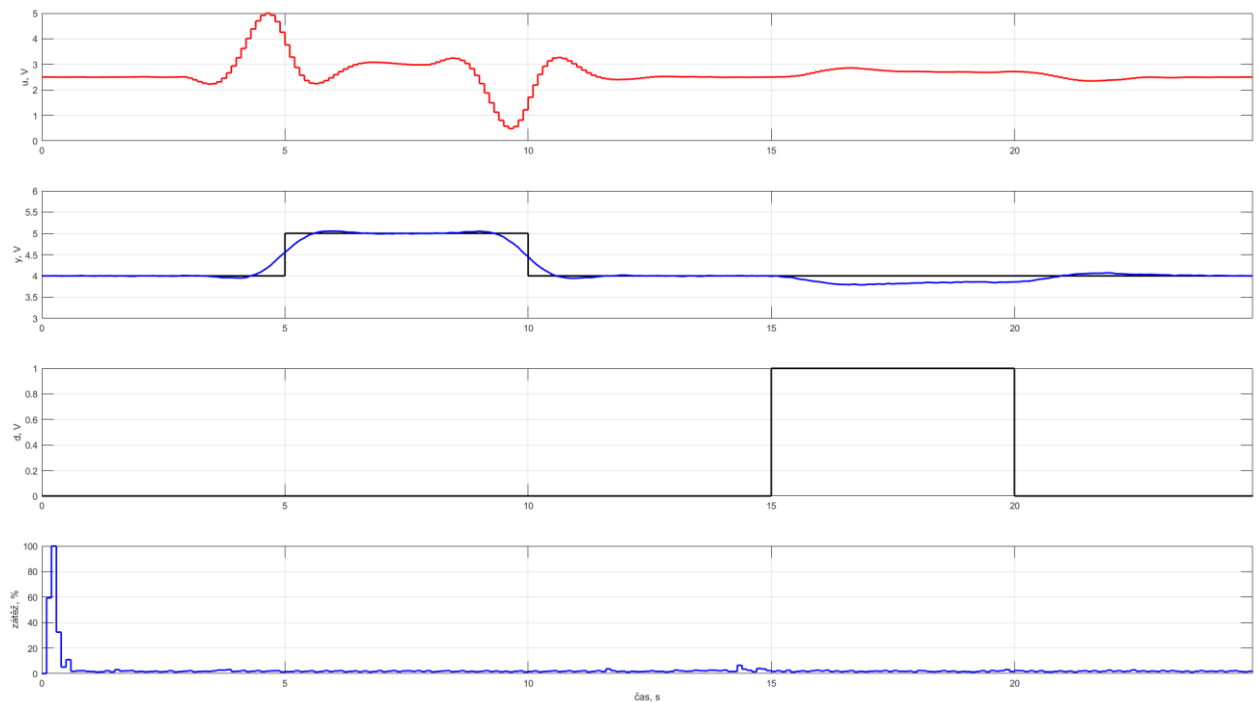
6.4.1 Experimenty s prediktivním regulátorem bez omezení, bez znalosti poruchy

V tabulce 6.4 jsou zobrazeny zvolené parametry pro prediktivní regulátor. Jsou zde uvedeny horizonty N_2 a N_u společně s jednotlivými penalizacemi veličin.

Tabulka 6.4 – Parametry prediktivního regulátoru bez omezení

Parametry	Hodnoty
Penalizace regulační odchylky r	1
Penalizace změn akční veličiny q	0,1
Horizont sledování žádané hodnoty N_2	20
Horizont řízení N_u	20

Tyto parametry platí pro simulaci i při řízení soustavy GUNT RT 050. Horizonty byly zvoleny dostatečně dlouhé vzhledem k dynamice soustavy. Penalizace veličin byly zvoleny tak abychom dostaly regulační pochod dostatečné kvality, a přitom nedocházelo k příliš velkým změnám akční veličiny při skokových změnách žádané hodnoty. Jednotlivé průběhy simulace pro prediktivní regulátor bez omezení, bez znalosti poruchy jsou vidět na obrázku 6.9.

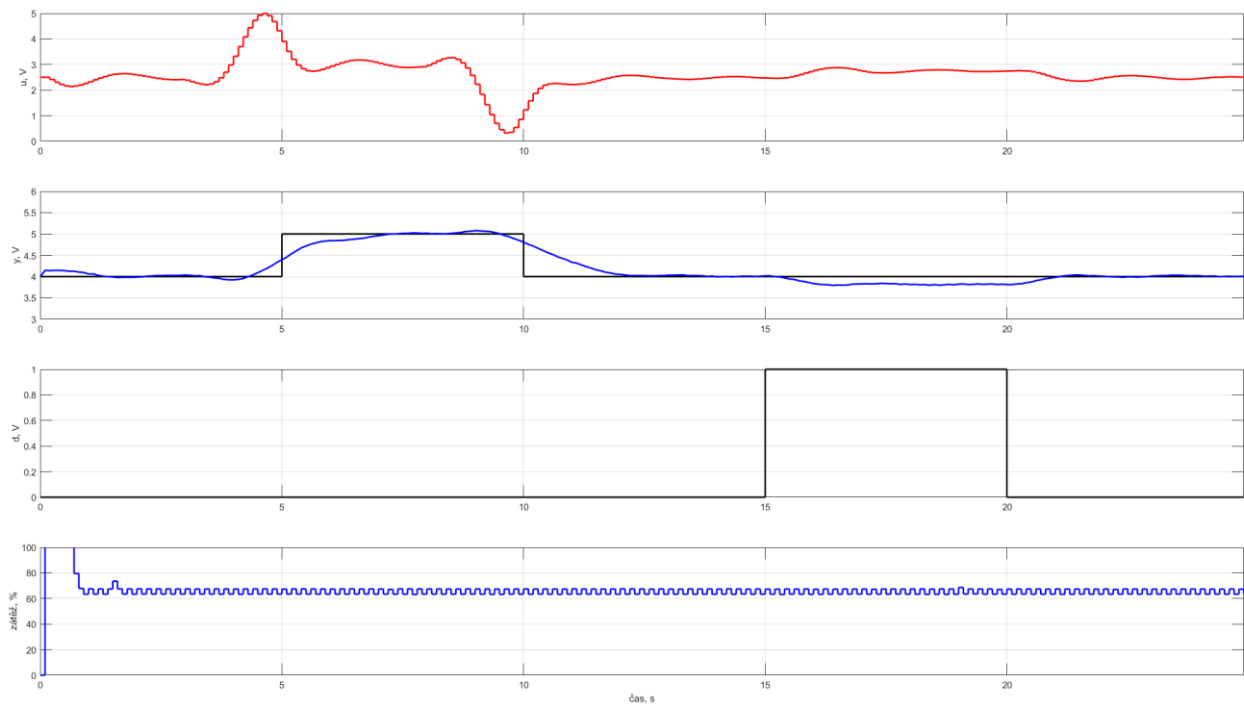


Obrázek 6.9 – Průběhy regulace prediktivního regulátoru bez omezení, bez znalosti poruchy v simulaci

V prvním průběhu je vidět akční veličina, která při změně žádané hodnoty vzroste s ještě před změnou žádané hodnoty – regulátor má k dispozici její budoucí průběh.

V druhém průběhu je regulovaná veličina, která prochází přes skokovou změnu žádané hodnoty. Regulátor nedokáže odregulovat poruchu bez trvalé regulační odchylky – tento typ regulátoru nemá integrační charakter. Vytížení systému je srovnatelné se simulací řízení s PID regulátorem.

Na obrázku 6.10 jsou zobrazeny průběhy pro prediktivní regulátor bez omezení, bez znalosti poruchy pro reálné řízení na soustavě GUNT RT 050.



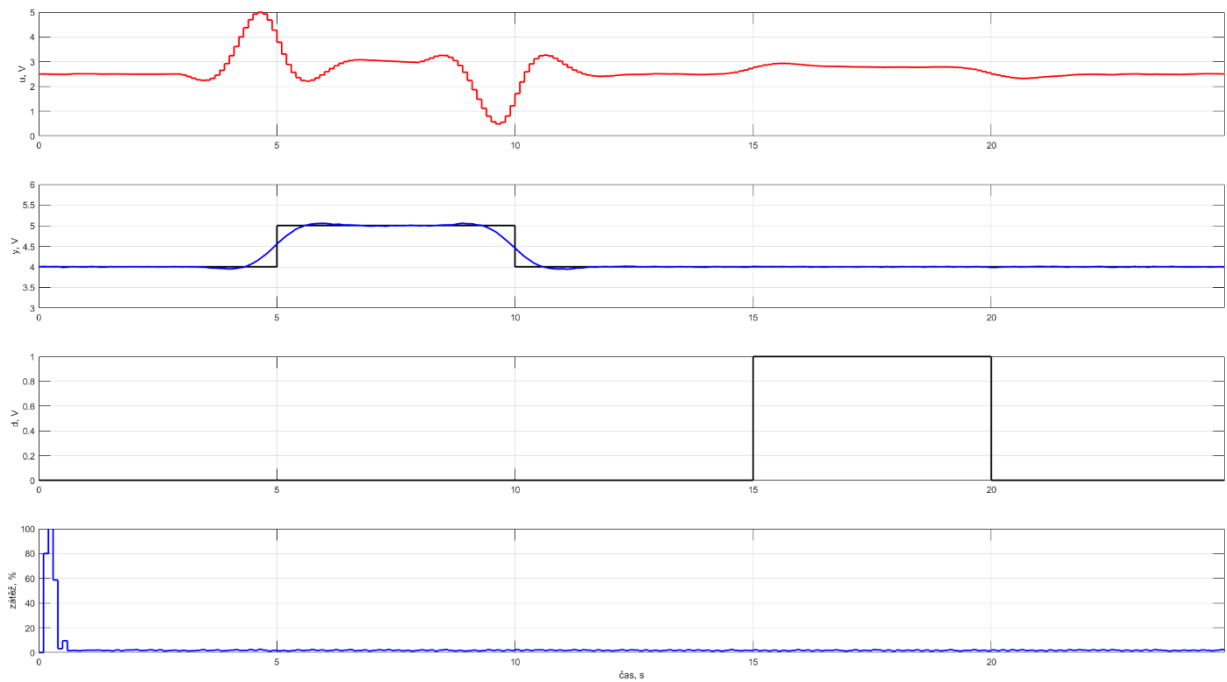
Obrázek 6.10 – Průběhy regulace prediktivního regulátoru bez omezení, bez znalosti poruchy na soustavě GUNT RT 050

V grafu jsou opět zobrazeny průběhy akční veličiny, žádané hodnoty, regulační odchylky a regulované veličiny, poruchy a vytížení systému.

Průběhy jsou velmi podobné simulacím, což značí o dobré shodě mezi modelem a reálným systémem. Opět je vytížení PC vyšší u reálného řízení než u simulace, ale je srovnatelné s PID regulátorem.

6.4.2 Experimenty s prediktivním regulátorem bez omezení, se znalostí poruchy

Parametry prediktivního regulátoru pro tuto variantu jsou stejné jako v tabulce 6.4. V této variantě má regulátor znalost o poruše a odreguluje ji. Průběhy ze simulace jsou vidět na obrázku 6.11.

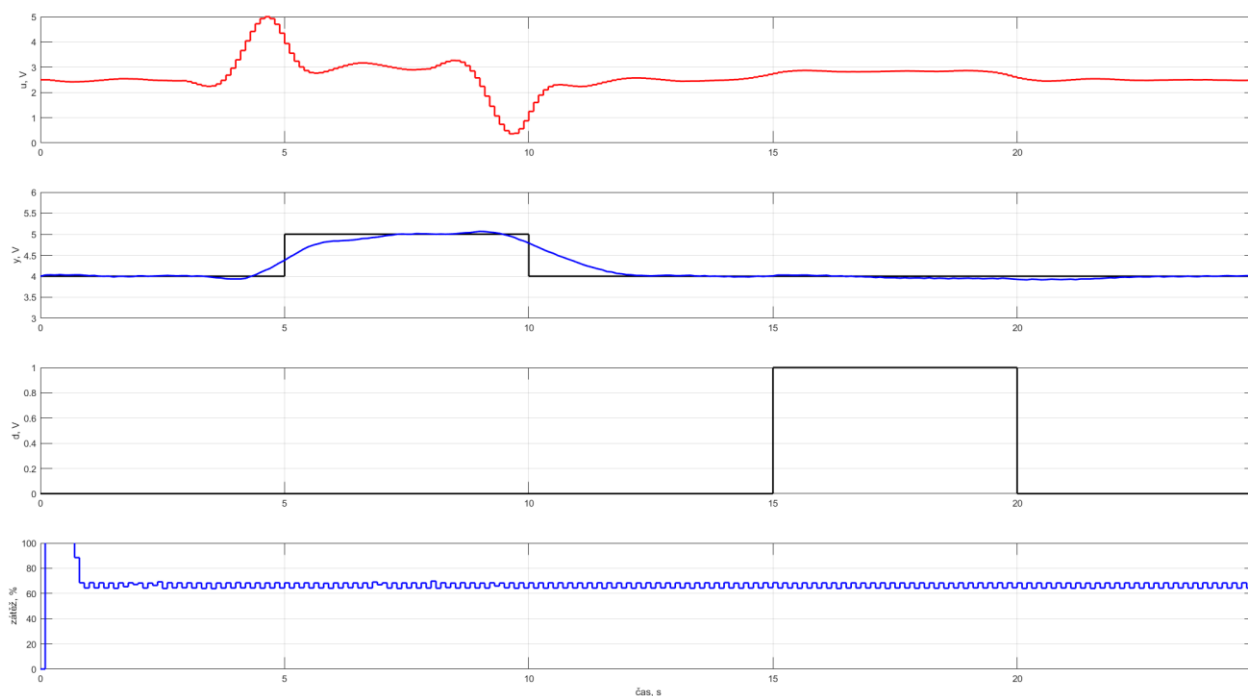


Obrázek 6.11 – Průběhy regulace prediktivního regulátoru bez omezení, se znalostí poruchy v simulaci

Jedná o variantu regulátoru bez omezení, ale se znalostí budoucího průběhu poruchy. Ke změně akční veličiny dojde ještě dřív, než dojde k poruše a průběh akčního zásahu je plynulý.

Rozdíl oproti variantě bez znalosti žádané hodnoty nastává v druhém průběhu, kde je ukázána žádaná hodnota a regulovaná veličina. Na obrázku 6.9 je vidět, že se regulovaná veličina nedostane na žádanou hodnotu v době, kdy nastane porucha. Vzniká trvalá regulační odchylka. Ale ve variantě se znalostí poruchy je vidět, že je odregulována a regulovaná veličina se ustálí na žádané hodnotě.

Poslední průběh zobrazuje vytížení PC, které je opět podobné jak v předchozích případech simulace.



Obrázek 6.12 - Průběhy regulace prediktivního regulátoru bez omezení, se znalostí poruchy na soustavě GUNT RT 050

Regulátor je opět připojen na soustavu GUNT RT 050 a jsou změřeny reálné pochody. V porovnání s průběhem na obrázku 6.10 má akční veličina podobný průběh. Regulovaná veličina se mírně liší od simulace, což je způsobeno nesouladem modelu (nelinearitou reálné soustavy). Regulátor prakticky nezanechává regulační odchylku a také dojde k odregulování poruchy. Vytížení PC je mírně vyšší než u případu, kdy neuvažujeme znalost poruchy.

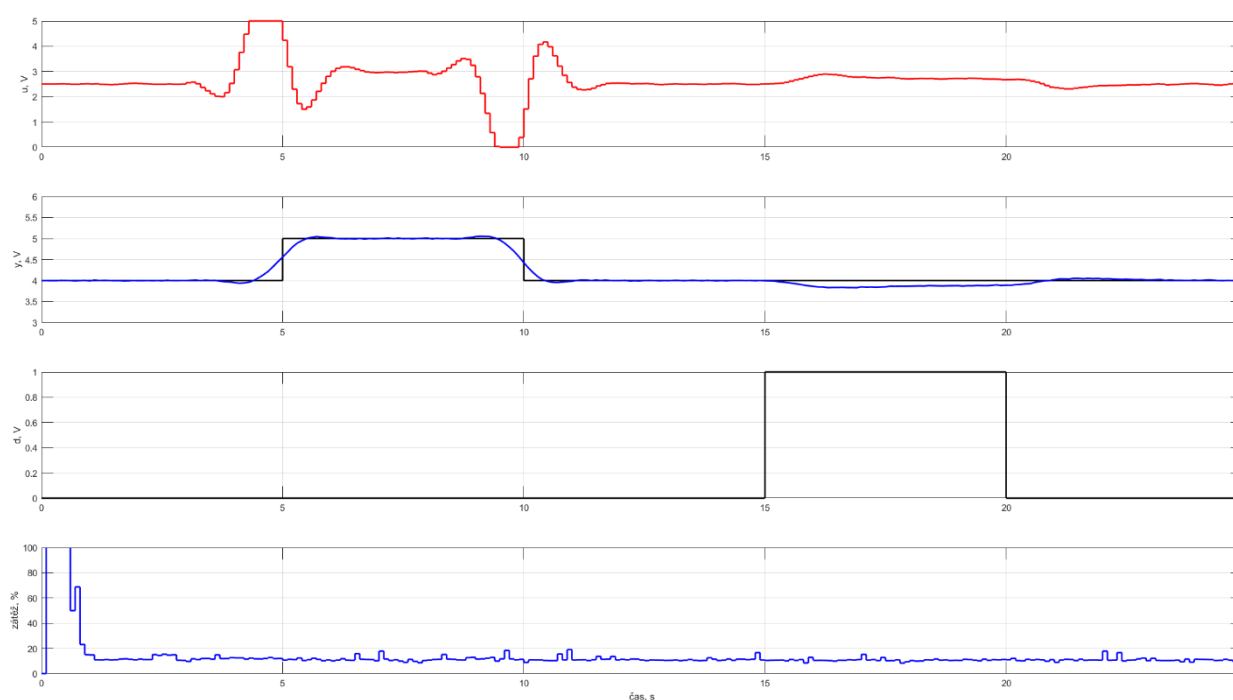
6.4.3 Experimenty s prediktivním regulátorem s omezeními, bez znalosti poruchy

Následující experimenty jsou provedeny s variantou prediktivního regulátoru, kde je využito kvadratické programování. Matematické vztahy, na kterých je založeno kvadratické programování jsou popsány v kapitole 4.3. Jsou aplikovány omezení akční veličiny, změny akční veličiny a regulované veličiny. Parametry regulátoru jsou stejné jako v předchozích případech a jsou uvedeny v tabulce 6.4.

Tabulka 6.5 – Hodnoty omezení veličin

Omezení	Minimální	Maximální
Omezení akční veličiny u	0	5
Omezení změn akční veličiny du	-5	5
Omezení regulované veličiny y	0	10

V tabulce 6.5 jsou zapsány hodnoty omezení akční veličiny, změn akčních veličin a regulované veličiny.



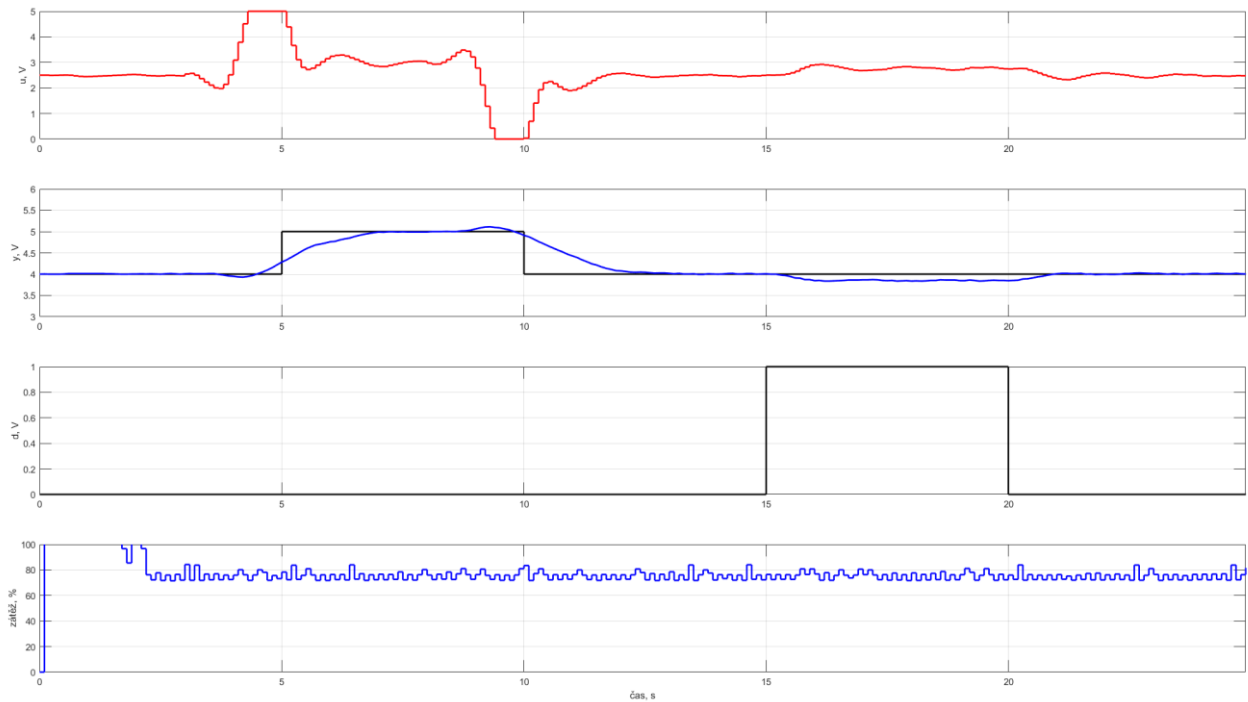
Obrázek 6.13 – Průběhy regulace prediktivního regulátoru s omezeními, bez znalosti poruchy v simulaci

Na prvním průběhu z obrázku 6.13 je vidět vliv omezení akční veličiny. Toto omezení zajišťuje kratší dobu působení akční veličiny s rychlejším nárůstem.

Druhý průběh je téměř identický jako na obrázku 6.9. Při aktivní poruše se vytvoří trvalá regulační odchylka. Jelikož prediktivní regulátor nemá integrační charakter ani informaci o budoucím průběhu poruchy, tak ji neodreguluje.

Vytížení PC je srovnatelné s ostatními variantami regulátorů v simulaci.

Na obrázku 6.14 jsou vidět průběhy regulace prediktivního regulátoru s připojenou soustavou GUNT RT 050.



Obrázek 6.14 – Průběhy regulace prediktivního regulátoru s omezeními, bez znalosti poruchy na soustavě GUNT RT 050

Rozdíl v prvním průběhu z obrázku 6.13 je při ustálení regulované veličiny na žádané hodnotě. Dochází k vyššímu využití akční veličiny při ustálení.

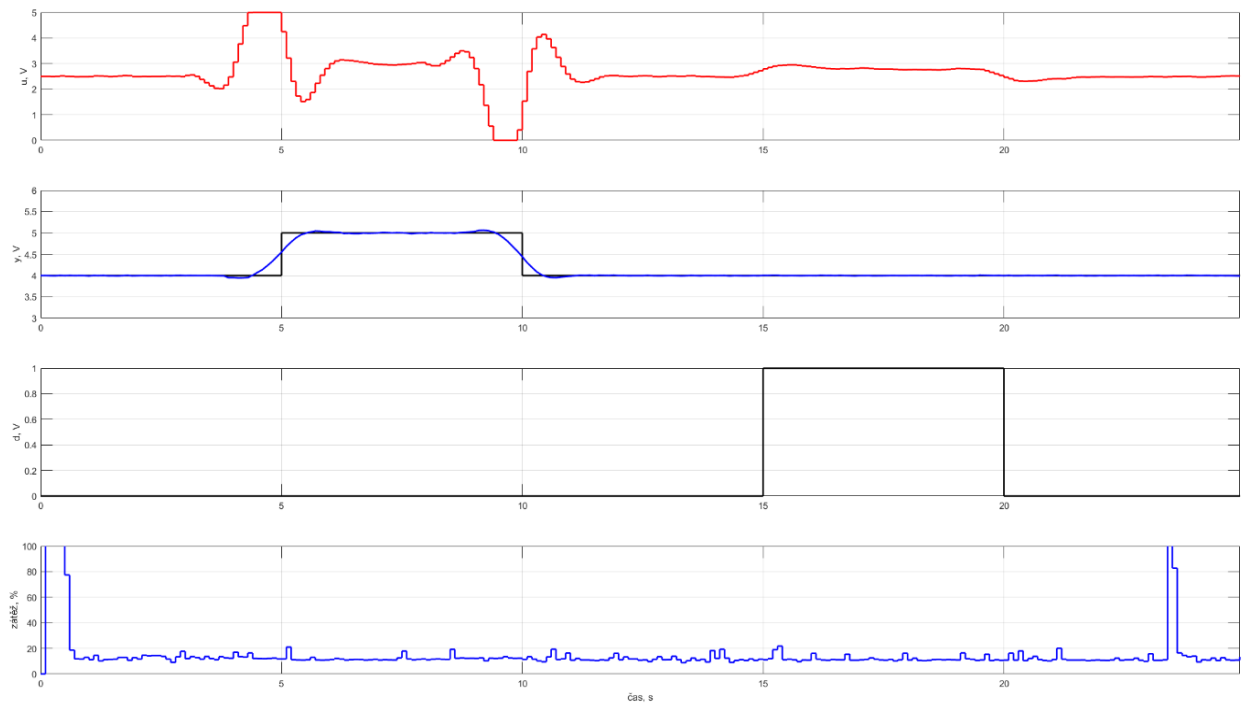
Regulované veličině v druhém průběhu trvá déle, než dosáhne žádané hodnoty. Oproti simulaci stejné varianty regulátoru dochází k větším překmitům.

Porucha zůstává neodregulována jako u variant prediktivního regulátoru bez znalosti poruchy. Vytížení PC je více proměnlivé než PID regulátoru.

6.4.4 Experimenty s prediktivním regulátorem s omezeními, se znalostí poruchy

Parametry a omezení regulátoru jsou stejné jako ve variantě s omezeními, bez znalosti poruchy. Jsou uvedeny v tabulce 6.4.

Na obrázku 6.15 jsou zobrazeny průběhy regulačních pochodů pro variantu se znalostí poruchy.



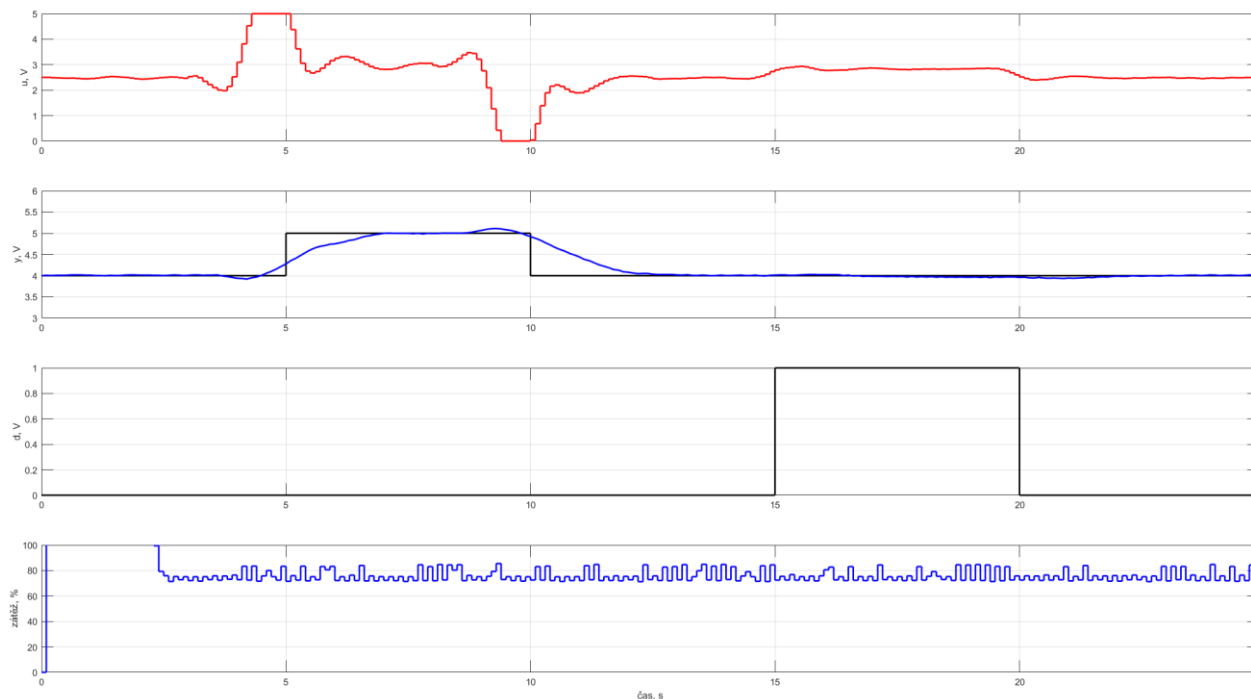
Obrázek 6.15 - Průběhy regulace prediktivního regulátoru s omezeními, se znalostí poruchy v simulaci

V porovnání s variantou prediktivního regulátoru s omezeními, bez znalosti poruchy je průběhy akční veličiny srovnatelný.

Rozdíl je v regulované veličině při aktivní poruše. V tomto případě má regulátor informaci o budoucím průběhu poruchy, takže při poruše nevzniká trvalá regulační odchylka.

Vytížení na PC je vyšší než při použití ostatních variant regulátorů v simulaci. Podle pozorování při měření se jedná o problém výkonnosti PC.

V posledním měření je prediktivní regulátor s omezeními, se znalostí poruchy připojený na soustavu GUNT RT 050. Průběhy regulace jsou vidět na obrázku 6.16.



Obrázek 6.16 - Průběhy regulace prediktivního regulátoru s omezeními, se znalostí poruchy na soustavě GUNT RT 050

V porovnání s variantou prediktivního regulátoru bez znalosti poruchy na soustavě GUNT RT 050 nedochází ke změně průběhu akční veličiny. Regulátor má informaci o budoucím průběhu poruchy, takže akční veličina roste dříve, než začne porucha působit.

Regulovaná veličina v druhém průběhu má téměř stejný průběh jako ve variantě prediktivního regulátoru bez znalosti poruchy.

Vytížení na PC je srovnatelné s variantou prediktivního regulátoru s omezeními připojeným na soustavu GUNT RT 050. Vytížení je vyšší ve srovnání s PID regulátorem připojeným k soustavě.

ZÁVĚR

Diplomová práce byla zaměřena na návrh a laboratorní aplikaci prediktivního regulátoru s uvažováním modelu poruchy. Nejprve byly odvozeny potřebné výpočetní vztahy a následně byla ověřena činnost prediktivního regulátoru na sérii experimentů. Tyto experimenty byly provedeny nejprve v simulačním prostředí a posléze na soustavě GUNT 050. Řízeny byly otáčky stejnosměrného motoru pomocí vstupního napětí.

V teoretické části je uveden úvod do teorie prediktivních regulátorů a užití v průmyslu v dnešní době. Dále jsou popsány matematické vztahy, na kterých prediktivní regulátor obecně funguje.

V praktické části jsou nejdříve popsány matematické vztahy, které jsou vztaženy na stavový prediktivní regulátor, který byl také realizován ve vývojovém prostředí MATLAB. Dále jsou zde uvedeny vývojové diagramy regulátoru. V poslední části byly provedeny experimenty, jak v simulovaném prostředí, tak v režimu s připojenou soustavou GUNT RT 050. Byly změřeny různé varianty prediktivního regulátoru a porovnány se standardním PID regulátorem. V experimentech je provedena skoková změna žádané hodnoty a v druhé části je zavedena skoková změna poruchy. Na těchto změnách je ukázán rozdíl mezi jednotlivými typy a variantami regulátorů.

Z porovnání měřených experimentů na prediktivním regulátoru a PID regulátoru je vidět, že prediktivní regulátor dokáže řídit danou soustavu lépe. Prediktivní regulátor dokázal lépe reagovat na změnu žádané hodnoty. V práci byly porovnány různé varianty prediktivního regulátoru. Varianty bez omezení veličin a bez znalosti poruchy a se znalostí poruchy. Druhý typ byla varianta s omezením veličin opět bez znalosti poruchy a se znalostí poruchy.

Když nebyla známá porucha, nedošlo k jejímu kompletnímu odregulování. V případě znalosti poruchy je kvalita regulačního pochodu mnohem vyšší. Pokud by byla porucha pouze měřena, došlo by k reakci regulátoru až v okamžiku jejího vzniku. V uvedených případech je uvažována dopředná znalost poruchy ve smyslu její predikce.

U všech variant prediktivního regulátoru byla dosažena lepší kvalita regulace než u PID regulátoru. Výsledek prediktivního regulátoru je hodně závislý na kvalitě modelu a na nastavení regulátoru. Určitou nevýhodou použité verze prediktivního regulátoru je fakt, že nemá integrační charakter. I tak nebyla trvalá regulační odchylka prakticky viditelná a regulátor by byl i v této verzi prakticky použitelný.

POUŽITÁ LITERATURA

- Aspen DMC3 | Sustain Optimal Performance | AspenTech. [online]. *Aspen Technology Inc.* [cit. 2025-03-25]. Dostupné z: <https://www.aspentech.com/en/products/msc/aspen-dmc3>
- AspenTech acquires Treiber Controls – Control Engineering. 1998. [online]. *WTWH Media, LLC.* [cit. 2025-03-25]. Dostupné z: <https://www.controleng.com/aspentech-acquires-treiber-controls/>
- BRÁZDIL, M. 2011. Výpočet obecného řídicího zákona prediktivního řízení pro systému druhého řádu bez omezení. *Posterus*. Ročník 4, číslo 7. [online]. [cit. 2025-03-25]. ISSN 1338-0087. Dostupné z: <https://www.posterus.sk/?p=11147>
- BRIAN FROISY, J. 2006. Model predictive control – Building a bridge between theory and practice. *Computers & Chemical Engineering*. Volume 30, Issues 10-12, p. 1426-1435. [online]. [cit. 2025-03-25]. ISSN 0098-1354. Dostupné z: <https://www.sciencedirect.com/science/article/abs/pii/S0098135406001542>
- CAMACHO, E.F.; BORDONS, C. 2007. *Model predictive control*. 2. vydání. Londýn: Springer. [cit. 2025-03-25]. ISBN 978-1-85233-694-3
- CVEJN, J. *Automatizace 1*. Pardubice: Univerzita Pardubice, Fakulta elektrotechniky a informatiky. [cit. 2025-03-25]
- GARZINOVÁ, R; JANČÍKOVÁ, Z; ZIMNÝ, O. 2013. Základní pojmy v teorie regulace. In: *Základy automatizace technologických procesů v teorii*. [online]. Ostrava: Vysoká škola báňská – Technická univerzita Ostrava. [cit. 2025-03-25]. ISBN 978-80-248-3044-5. Dostupné z: https://projekty.fs.vsb.cz/463/edubase/VY_01_035/Z%C3%A1klady%20automatizace%20technologick%C3%BDch%20proces%C5%AF%20v%20teorii/02%20Text%20pro%20e-learning/Z%C3%A1klady%20automatizace%20technologick%C3%BDch%20proces%C5%AF%20v%20teorii%20%2005.pdf
- HUBÁČEK, J. 2008. *Prediktivní řízení s omezením vstupních a výstupních veličin*. [online]. Diplomová práce. Zlín: Univerzita Tomáše Bati ve Zlíně, Fakulta aplikované informatiky, Ústav řízení procesů. [cit. 2025-03-25]. Dostupné z: https://digilib.k.utb.cz/bitstream/handle/10563/6645/hub%C3%A1%C4%8Dek_2008_dp.pdf?sequence=1
- HONC, D. Stavový popis dynamického systému, regulace, pozorovatel. *Automatizace*. Pardubice: Univerzita Pardubice, Fakulta elektrotechniky a informatiky. [cit. 2025-03-25]
- HONC, D. Prediktivní řízení. Pardubice: Univerzita Pardubice, Fakulta elektrotechniky a informatiky. [cit. 2025-03-25]
- MACHALSKÝ, O. 2011. *Prediktivní řízení s měřenou poruchovou veličinou*. [online]. Diplomová práce. Zlín: Univerzita Tomáše Bati ve Zlíně, Fakulta aplikované informatiky, Ústav řízení procesů. [cit. 2025-03-25]. Dostupné z: https://digilib.k.utb.cz/bitstream/handle/10563/14782/machalsk%C3%BD_2011_dp.pdf?sequence=1&isAllowed=y
- Machine Direction Control. [online]. *Honeywell International Inc.* [cit. 2025-03-25]. Dostupné z: <https://process.honeywell.com/us/en/products/control-and-supervisory-systems/quality-control-systems-qcs/qcs-se/machine-direction-controls>
- ROSSITER, J, A. 2003. *Model-Based Predictive Control: A Practical Approach*. Boca Raton: CRC Press. [cit. 2025-03-25]. ISBN 0-8493-1291-4

- RT 050 Training systém speed control, HSI. [online]. *G.U.N.T. Gerätebau GmbH*.
[cit. 2025-03-25]. Dostupné z: <https://www.gunt.de/en/products/training-system-speed-control-hsi/080.05000/rt050/glct-1:pa-148:pr-1046>
- ŠEDA, M. 2007. *Diskrétní systémy řízení*. [online]. Brno: Vysoké učení technické v Brně, Fakulta strojního inženýrství. [cit. 2025-03-25]. Dostupné z:
https://matlab.fei.tuke.sk/raui_new/subory/literatura/seda_diskretni_systemy_rizeni.pdf
- What Is MATLAB? – MATLAB & Simulink. [online]. *The MathWorks, Inc.*
[cit. 2025-03-25]. Dostupné z: <https://www.mathworks.com/discovery/what-is-matlab.html>