

Robustní prediktivní řízení laboratorní soustavy

Robust Predictive Control of Laboratory Plant

Bc. Václav Růžek

Diplomová práce
2014



Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

ZADÁNÍ DIPLOMOVÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Bc. Václav Růžek**
Osobní číslo: **A12414**
Studijní program: **N3902 Inženýrská informatika**
Studijní obor: **Automatické řízení a informatika**
Forma studia: **prezenční**

Téma práce: **Robustní prediktivní řízení laboratorní soustavy**

Zásady pro vypracování:

1. Vypracujte literární rešerši zabývající se robustností prediktivního řízení.
2. Navrhněte a popište algoritmy, které jsou vhodné pro robustní prediktivní řízení v reálném čase.
3. Naprogramujte vybrané algoritmy robustního prediktivního řízení v prostředí MATLAB/Simulink a proveďte jejich simulační ověření na různých typech soustav. Vyhodnoťte vhodnost algoritmů pro reálné nasazení.
4. Vytvořte simulační model vybrané laboratorní soustavy, na kterou bude aplikováno robustní prediktivní řízení.
5. Aplikujte vhodné algoritmy na reálnou soustavu a vyhodnoťte výsledky především z hlediska kvality regulace a výpočetní náročnosti.

Rozsah diplomové práce:

Rozsah příloh:

Forma zpracování diplomové práce: **tištěná/elektronická**

Seznam odborné literatury:

1. BEMPORAD, Alberto a Manfred MORARI. Robust Model Predictive Control: A Survey. Dostupné z: <http://www8.cs.umu.se/research/ifor/dl/survey-robust-mpc.pdf>
2. BOBÁL, Vladimír. Adaptivní a prediktivní řízení. 1. vyd. Zlín: Univerzita Tomáše Bati ve Zlíně, 2008, 134 s. ISBN 978-80-7318-662-3.
3. MATUŠŮ, Radek a Petr DOSTÁL. Stavová a algebraická teorie řízení [online]. 2010 [cit. 2014-01-24]. ISBN 978-80-7318-991-4. Dostupné z: <http://hdl.handle.net/10563/18580>
4. CAMACHO, E a Carlos BORDONS. Model predictive control. 2nd ed. London: Springer, c2007, xxii, 405 s. ISBN 1-85233-694-3.
5. HUANG, Sunan, Tong Heng LEE a Kok Kiong TAN. Applied predictive control. London: Springer, 2002, xvii, 264 s. ISBN 1852333383.
6. ROSSITER, J. Model-based predictive control: a practical approach. Boca Raton: CRC Press, c2003, 318 s. ISBN 0849312914.
7. SCHLEGEL, Miloš a Jaroslav SOBOTA. Prediktivní regulátor pro průmyslovou praxi. AUTOMA. 2007, č. 2, s. 12-16. Dostupné z: <http://www.odbornecasopisy.cz/download/au020712.pdf>

Vedoucí diplomové práce:

Ing. Petr Chalupa, Ph.D.

Ústav řízení procesů

Datum zadání diplomové práce:

7. března 2014

Termín odevzdání diplomové práce:

11. června 2014

Ve Zlíně dne 7. března 2014



prof. Ing. Vladimír Vašek, CSc.
děkan



prof. Ing. Vladimír Vašek, CSc.
ředitel ústavu

ABSTRAKT

Práce se zabývá robustním prediktivním řízením. Popisuje několik verzí metody Generalized Predictive Control a metodu Offset-free Model Predictive Control. Jednotlivé prediktory jsou naprogramovány v prostředí Matlab/Simulink a ověřeny jednak na simulačních modelech, tak i na reálné laboratorní soustavě Amira DR300.

Klíčová slova: Prediktivní řízení, robustnost, Generalized Predictive Control, Offset-free Model Predictive Control, omezení veličin, Matlab/Simulink, kvadratické programování.

ABSTRACT

The thesis deals with the robust Model Predictive Control. It describes several techniques of Generalized Predictive Control and Offset-free Model Predictive Control. Each predictor is programmed in Matlab/Simulink. They are tested on simulation models and also on the laboratory plant Amira DR300.

Keywords: Model Predictive Control, robustness, Generalized Predictive Control, Offset-free Model Predictive Control, constraints, Matlab/Simulink, quadratic programming.

Tímto bych chtěl poděkovat vedoucímu diplomové práce Ing. Petru Chalupovi, Ph.D. za cenné rady a odborné vedení při tvorbě práce. Dále rodině a přítelkyni za podporu během studia.

Prohlašuji, že

- beru na vědomí, že odevzdáním diplomové/bakalářské práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že diplomová/bakalářská práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk diplomové/bakalářské práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně a jeden výtisk bude uložen u vedoucího práce;
- byl/a jsem seznámen/a s tím, že na moji diplomovou/bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užít své dílo – diplomovou/bakalářskou práci nebo poskytnout licenci k jejímu využití jen s předchozím písemným souhlasem Univerzity Tomáše Bati ve Zlíně, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše);
- beru na vědomí, že pokud bylo k vypracování diplomové/bakalářské práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky diplomové/bakalářské práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem diplomové/bakalářské práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

Prohlašuji,

- že jsem na diplomové práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků budu uveden jako spoluautor.
- že odevzdaná verze diplomové práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně 11. 6. 2014

.....
podpis diplomanta

OBSAH

ÚVOD.....	9
I TEORETICKÁ ČÁST.....	10
1 HISTORIE PREDIKTIVNÍHO ŘÍZENÍ.....	11
2 ROBUSTNOSTNÍ ŘÍZENÍ.....	12
2.1 PARAMETRICKÁ NEURČITOST	13
2.2 NEPARAMETRICKÁ NEURČITOST.....	14
2.2.1 Multiplikativní model.....	14
2.2.2 Aditivní model	14
2.3 ROBUSTNÍ STABILITA	15
2.4 ROBUSTNÍ KVALITA REGULACE	16
3 PREDIKTIVNÍ ŘÍZENÍ.....	18
3.1 PRINCIP	18
3.2 MODEL PREDIKTIVNÍHO ŘÍZENÍ	20
3.2.1 Impulzní funkce	20
3.2.2 Přechodová funkce	20
3.2.3 Stavový popis	20
3.2.4 Přenosové funkce	21
3.2.5 Funkce s poruchou	21
3.3 ÚČELOVÁ FUNKCE.....	21
3.4 OMEZENÍ ŘÍDÍCÍCH VELIČIN.....	24
3.4.1 Tvrdá omezení.....	25
3.4.2 Měkká omezení	28
4 GENERALIZED PREDICTIVE CONTROL	29
4.1 ODVOZENÍ PREDIKTORU	29
4.2 OPTIMÁLNÍ ŘÍZENÍ.....	31
4.3 VÝPOČET PREDIKTORU PRO MODEL DRUHÉHO ŘÁDU	32
4.3.1 Prediktor bez omezujících podmínek.....	32
4.3.2 Prediktor s omezujícími podmínkami	34
5 OFFSET-FREE LINEAR MODEL PREDICTIVE CONTROL	36
5.1 ODVOZENÍ PREDIKTORU	36
5.2 VÝPOČET PORUCHY	38
6 IDENTIFIKACE MODELU	40
6.1 METODA NEJMENŠÍCH ČTVERCŮ.....	40
6.2 REKURZIVNÍ METODA NEJMENŠÍCH ČTVERCŮ.....	41
6.3 REKURZIVNÍ METODA NEJMENŠÍCH ČTVERCŮ S EXPONENCIÁLNÍM ZAPOMÍNÁNÍM	42
6.4 REKURZIVNÍ METODA NEJMENŠÍCH ČTVERCŮ S ADAPTIVNÍM SMĚROVÝM ZAPOMÍNÁNÍM	43
II PRAKTICKÁ ČÁST	44
7 SERVOMECHANISMUS AMIRA DR300.....	45

7.1	MATEMATICKÝ POPIS MODELU.....	45
7.2	STATICKÁ CHARAKTERISTIKA	47
7.3	IDENTIFIKACE SYSTÉMU	48
8	SIMULACE PREDIKTIVNÍHO ŘÍZENÍ.....	52
8.1	GPC INVERZÍ MATIC.....	52
8.1.1	Nastavení parametrů.....	52
8.1.2	Robustnost řízení.....	55
8.1.3	Omezení veličin	59
8.2	GPC KVADRATICKÝM PROGRAMOVÁNÍM.....	60
8.2.1	Omezení veličin	61
8.3	ADAPTIVNÍ GPC	62
8.3.1	Kvalita regulace	62
8.4	OFFSET-FREE LINEAR MPC	66
8.4.1	Nastavení parametrů.....	66
8.4.2	Robustnost.....	68
9	MĚŘENÍ NA REÁLNÉM ZAŘÍZENÍ.....	70
9.1	GPC	70
9.1.1	Bez omezení veličin	70
9.1.2	S omezením veličin	71
9.1.3	Adaptivní GPC	72
9.2	OFFSET-FREE MPC.....	74
9.2.1	Robustní kvalita regulace.....	75
	ZÁVĚR	77
	ZÁVĚR V ANGLIČTINĚ.....	78
	SEZNAM POUŽITÉ LITERATURY	79
	SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK.....	82
	SEZNAM OBRÁZKŮ	83
	SEZNAM TABULEK.....	85
	SEZNAM PŘÍLOH.....	86

ÚVOD

Automatizace může být charakterizována jako řízení procesů a strojů pomocí řídicího systému. Název pochází z řeckého slova Autómatos, což lze volně přeložit jako sám od sebe jednající. První zmínky o automatizaci můžeme vystopovat až do starověku, například v 3. století př. n. l. byly těžké chrámové dveře v Alexandrii automaticky otvírány pomocí mechanismu založeném na principu vodní páry. [1]

První rozvoj automatizace nastal pak především s příchodem průmyslové revoluce v 18. století. Všeobecně známý je odstředivý regulátor parního stroje, který zkonstruoval James Watt roku 1782. V průběhu 19. století můžeme začít hovořit o automatizaci jako o vědním oboru, který se postupně více matematizuje, kde hlavními osobnostmi byli E. J. Routh, A. Hurwitz a A. M. Ljapunov, který položil základy teorie stability. Ve 30. letech 20. století H. Nyquist a H. W. Bode prezentovali práce zabývající se frekvenční analýzou systému. Ve 40. letech J. G. Ziegler a N. B. Nichols vypracovali práci zabývající se efektivním nastavením regulátorů. [2] [3]

Masivní rozvoj automatizace nastává v období po 2. světové válce společně s rozvojem elektrotechniky. Automatizovány jsou v té době rozsáhlé průmyslové celky. S nástupem digitálních počítačů v 70. letech 20. století byla vypracována teorie diskrétního řízení. V 80. letech se objevují první počítače schopné řízení v reálném čase. V posledních desetiletích došlo k obrovskému nárůstu výkonu výpočetní techniky a zároveň se tato technika stala cenově dostupnější. To umožnilo její široké nasazení do procesu řízení a regulace a umožnilo běžné použití moderních metod řízení, které byly do té doby příliš výpočetně náročné. S těmito metodami se otevřely zcela nové možnosti řešení regulačních úloh. [3]

Prediktivní řízení založené na modelu (Model Predictive Control) je metoda, která zaznamenala prudký rozvoj v posledních letech a v současnosti je široce rozšířená. Pod MPC patří mnoho různých metod. Všechny tyto metody obsahují model řízeného systému, s jehož pomocí predikují budoucí řízení výstupního systému. Dále je výpočet budoucích řídicích hodnot minimalizován dle zvolené účelové funkce. [4]

I. TEORETICKÁ ČÁST

1 HISTORIE PREDIKTIVNÍHO ŘÍZENÍ

Model Predictive Control má dlouhou historii. Jeho počátky lze vystopovat do 60. let minulého století. V 70. letech společnost Shell Oil vyvíjela technologii prediktivního řízení, kterou použila v praxi již v roce 1973. [5].

Později v 70. letech vyšlo několik prací zabývajících se nasazením MPC v průmyslu. V roce 1978 byl představen *Model Predictive Heuristic Control*, který je dnes známý spíše pod názvem *Model Algorithmic Control* (MAC). Dvojice Cutler a Ramaker prezentovali roku 1980 řízení *Dynamic Matrix Control* (DMC). Společné pro tyto prediktory je použití dynamického modelu procesu, buď ve tvaru impulzní funkce, nebo později ve tvaru přechodové charakteristiky. Pomocí těchto modelů jsou predikovány budoucí změny akčního zásahu v závislosti na minimalizaci chyby. Postup se opakuje každou periodu. U těchto prvotních MPC nebyla zaručena stabilita. [6]

Novější generace MPC jako například *Quadratic Dynamic Matrix Control* (QDMC) prezentovaná v roce 1986, začala využívat kvadratické programování pro řešení optimalizačního problému, jež je dán cenovou funkcí a omezujícími podmínkami, zadanými jako lineární nerovnosti. [6]

Další vývojovou linií, jež vznikla nezávisle, byly prediktory pro SISO systémy, kde byla využita přenosová funkce jako model a diofantické rovnice pro výpočet predikovaných akčních zásahů. Zástupcem takového prediktoru je *Generalized Minimum Variance Control* (GVM). Tento prediktor dokázal pracovat i s neminimálně fázovými systémy a omezujícími podmínkami vstupních a výstupních veličin. V roce 1987 byl na principu GVM vyvinuta metoda *Generalized Predictive Control* (GPC), která je jednou z nejpoužívanějších metod prediktivního řízení. [6] [7]

V roce 1999 Bemporad a Morari prezentovali práci zabývající se robustností v prediktivním řízení. Vývoj moderních technik prediktivního řízení se zaměřil na robustnost, stabilitu a výkon. [5] [8]

2 ROBUSTNOSTNÍ ŘÍZENÍ

Při návrhu regulátoru většinou pracujeme s určitou formou matematického modelu řízené soustavy. V praxi se téměř nikdy náš matematický model přesně neshoduje s chováním reálného systému. Toto může být způsobeno mnoha okolnostmi. Často je to způsobeno nelinearitou reálného systému, kterou zanedbáme lineární aproximací modelu ve zvoleném pracovním bodě. Dále může být tento rozdíl způsobem zjednodušením reálně velmi složitého systému, tak abychom byli schopni s modelem jednoduše pracovat. Taktéž nám nemusí být přesně známy veškeré parametry reálného řízeného systému. Pokud budeme uvažovat tuto odchylku matematického modelu od původního reálného, budeme pracovat s tzv. *neurčitým modelem*. [9]

Neurčitý model popisuje celou množinu modelů v okolí modelu nominálního. Rozsah tohoto okolí popisujeme pomocí *parametrické* a *neparametrické* neurčitosti, popř. kombinací obou tzv. *smíšenou* neurčitostí. V případě parametrické neurčitosti, definujeme interval, ve kterém se mohou jednotlivé parametry nacházet. V tomto případě bychom tedy měli znát řád a strukturu zkoumaného systému. U neparametrické neurčitosti nemusí být předem známa struktura systému, jelikož interval určíme pomocí rozptylu frekvenčních charakteristik. [9] [10]

Robustnost řízení je vlastnost regulátoru, jež umožňuje řídit s určitými požadavky na přesnost nominálního modelu, ale i množinu modelů v jeho okolí, zadaných neurčitým modelem. Z výše uvedeného je zřejmé, že určitou míru robustnosti má většina funkčních regulátorů, jelikož téměř nikdy nepracujeme s naprosto přesným modelem řízené soustavy. [10]

Pokud $G_0(s)$ je přenos nominálního modelu systému, pak platí

$$G_0(s) \in \Pi \quad (2.1)$$

kde Π je množina všech modelů popsanych neurčitým modelem. Také platí

$$G(s) \in \Pi \quad (2.2)$$

kde $G(s)$ je libovolný model z této množiny.

2.1 Parametrická neurčitost

Parametrickou neurčitost použijeme u systémů, u nichž známe strukturu a řád, jen přesně neznáme jeden či více parametrů. Nejčastější zápis neurčitých parametrů je vektorem reálných neurčitých parametrů q . Pokud máme n neurčitých parametrů, zapíšeme q jako n -rozměrný vektor:

$$q = (q_1 \quad q_2 \quad \dots \quad q_n) \quad q \in R \quad (2.3)$$

Vektor q představuje možné hodnoty určitého parametru z přenosové funkce. Zápis přenosové funkce i s vektorem neurčitosti můžeme provést ve tvaru [9]

$$G(s, q) = \frac{b(s, q)}{a(s, q)} = \frac{\sum_{i=0}^m \beta_i(q) s^i}{\sum_{i=0}^n \alpha_i(q) s^i} \quad m < n \quad (2.4)$$

Nejčastěji se zajímáme o charakteristický polynom uzavřeného regulačního obvodu, jež určuje dynamické vlastnosti a stabilitu zkoumaného systému. Nejběžnějším typem zápisu je tzv. *intervalový zápis*, který můžeme popsat jako

$$p(s, q) = \sum_{i=0}^n [q_i^-, q_i^+] s^i \quad (2.5)$$

kde:

q_i^- dolní mez parametru,

q_i^+ horní mez parametru.

Pokud budeme chtít prohlásit takto zadaný polynom za robustně stabilní, musíme projít všechny kombinace polynomů a určit stabilitu všech polynomů. S rostoucím počtem neurčitých parametrů roste i počet možných kombinací. Pro zjednodušení se využívá *Charitonův teorém*, který říká, že intervalový polynom s invariantním stupněm je stabilní tehdy, jestliže jsou stabilní čtyři *Charitonovovy polynomy*. Postačí tedy zkontrolovat stabilitu čtyř polynomů, bez ohledu na množství neurčitých parametrů. Jednotlivé Charitonovovy polynomy se střídáním dolních – a horních + hraničních mezí, dle následujícího pořadí. [9]

1. --++
 2. ++--
 3. +--+
 4. -++-
- (2.6)

2.2 Neparametrická neurčitost

V literatuře se často používá i pojem *nestrukturovaná neurčitost* [9]. Při použití neparametrické neurčitosti nám nemusí být přesně známá struktura zadaného systému. Neurčitost definujeme dle několika různých neurčitých modelů.

2.2.1 Multiplikativní model

Multiplikativní model můžeme vyjádřit ve tvaru

$$G(s) = [1 + W_M(s)\Delta_M(s)]G_0(s) \quad (2.7)$$

kde:

$G(s)$ je přenos skutečného systému,

$G_0(s)$ je přenos nominální soustavy,

$W_M(s)$ váhová stabilní funkce, určuje dynamiku neurčitosti

$\Delta_M(s)$ libovolná stabilní funkce, která splňuje podmínku

$$\|\Delta(s)\|_{\infty} \leq 1 \Rightarrow |\Delta(j\omega)| \leq 1 \quad \forall \omega. \quad (2.8)$$

Pro tento model musí platit vztah [9]

$$\left| \frac{G(j\omega)}{G_0(j\omega)} - 1 \right| \leq |W_M(j\omega)| \quad \forall \omega \quad (2.9)$$

2.2.2 Aditivní model

Aditivní model můžeme vyjádřit ve tvaru

$$G(s) = G_0(s) + W_A(s)\Delta_A(s). \quad (2.10)$$

Pro aditivní model platí

$$\|G(j\omega) - G_0(j\omega)\| \leq \|W_A(j\omega)\| \quad \forall \omega \quad (2.11)$$

Základním předpokladem je, aby počet nestabilních pólů pro všechny soustavy $G(s)$ byl stejný. [9]

2.3 Robustní stabilita

Jedním ze základních požadavků při návrhu regulátoru je, abychom dosáhli stability zpětnovazebního regulačního obvodu. Stejně je tomu i u robustní stability, kde navíc požadujeme, aby regulační obvod byl stabilní i pro všechny modely z množiny určené neurčitým modelem.

Robustní stabilitu systému určíme podle *věty o malém zesílení*. Pokud je přenos otevřené smyčky $L(s)$ stabilní, tak přenos uzavřené smyčky je stabilní za předpokladu

$$\|L(j\omega)\| < 1 \quad \forall \omega \quad (2.12)$$

kde $\|L\|$ splňuje normu submultiplikativnosti $\|AB\| \leq \|A\| \cdot \|B\|$

Pro multiplikativní model platí, že je robustně stabilní za předpokladu

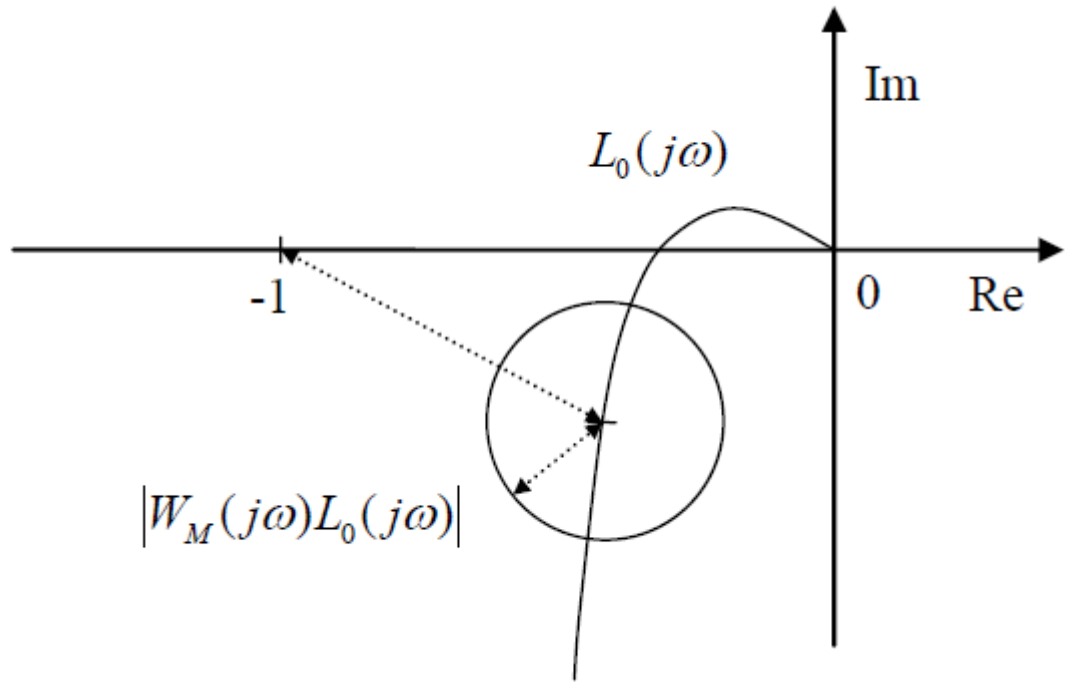
$$\|W_M(j\omega)L_0(j\omega)\| < \|L_0(j\omega) - (-1)\| \quad \forall \omega \quad (2.13)$$

kde:

W_M je stabilní váhová matice multiplikativního systému,

L_0 je uzavřená smyčka nominálního systému.

Z praktického hlediska tedy neurčitost systému nesmí změnit frekvenční charakteristiku otevřeného regulačního obvodu tak, aby se změnil počet oběhu kolem bodu $(-1 \ 0i)$ v komplexní rovině, graficky znázorněné na (Obr. 1) [9]



Obr. 1. Grafické znázornění robustní stability pro multiplikativní neurčitost [9]

2.4 Robustní kvalita regulace

Pro vyhodnocení kvality regulace se běžně používají kritéria založené na sumaci kvadrátů regulační odchylky [11]

$$S_y = \frac{1}{k_2 - k_1 + 1} \sum_{k=k_1}^{k_2} [w(k) - y(k)]^2 \quad (2.14)$$

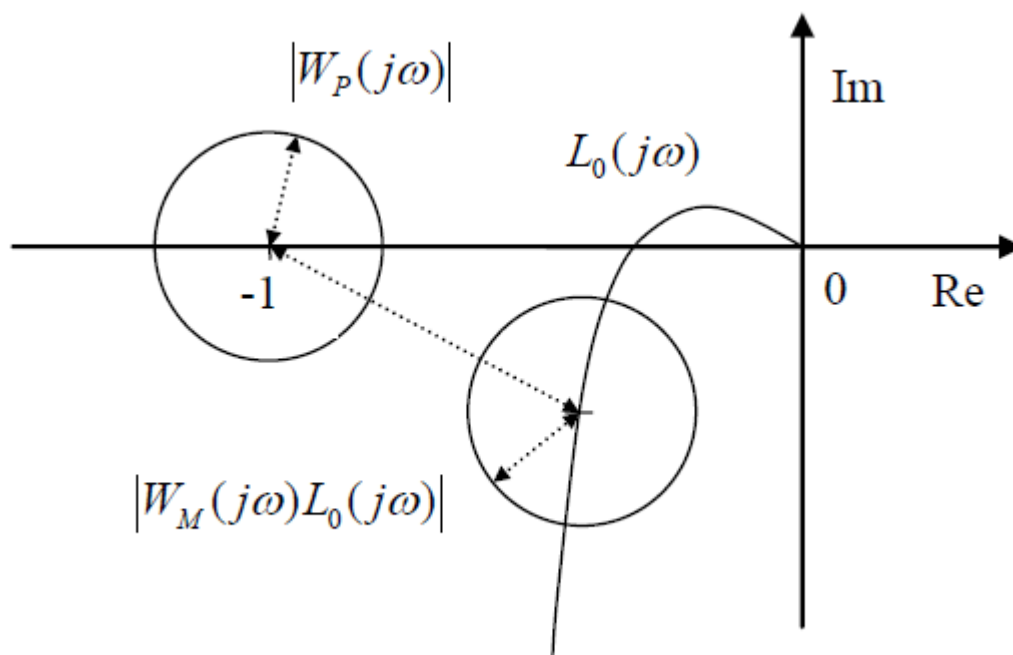
kde k_1 a k_2 určují interval kvality regulace. Nebo také kvadrátu rozdílů akčních zásahů

$$S_u = \frac{1}{k_2 - k_1 + 1} \sum_{k=k_1}^{k_2} [u(k) - u(k-1)]^2 \quad (2.15)$$

Pod pojmem kvalita regulace můžeme také vyhodnocovat velikost přechmitu regulačního děje, velikost trvalé regulační odchylky apod. Z pohledu robustní kvality regulace budeme uvažovat, že regulátor je robustní, pokud je hodnota regulační odchylky nepřesáhne požadovanou hodnotu pro všechny modely z množiny Π . Robustní kvalitu regulace můžeme vyjádřit nerovnicí, pokud upravíme vztah (2.13) [9]

$$|W_P(j\omega)| + |W_M(j\omega)L_0(j\omega)| < |L_0(j\omega) - (-1)| \quad \forall \omega \quad (2.16)$$

Grafické vyjádření nerovnice (2.16) je na (Obr. 2)



Obr. 2. Grafické znázornění robustní kvality regulace pro multiplikativní neurčitost [9]

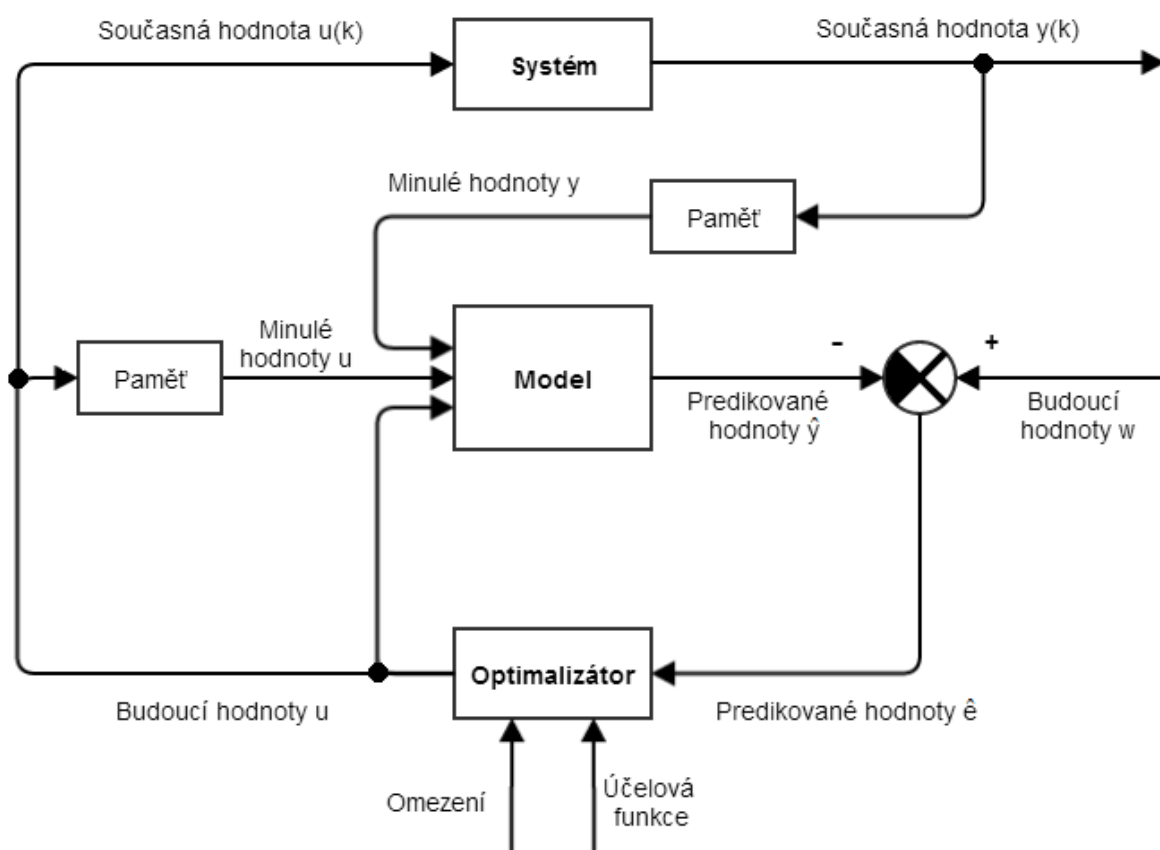
3 PREDIKTIVNÍ ŘÍZENÍ

MPC je v dnešní době široce využívaná metoda řízení především v oblasti průmyslu, ale i na univerzitách. MPC mají oproti klasickým PID regulátorům ve většině případů vyšší kvalitu řízení. Můžeme je použít na neminimálně fázové, nestabilní procesy, také dokáží pracovat s mnohazměrovými procesy a procesy s velkým dopravním zpožděním. Můžeme do nich zakomponovat omezení vstupních a výstupních veličin.

3.1 Princip

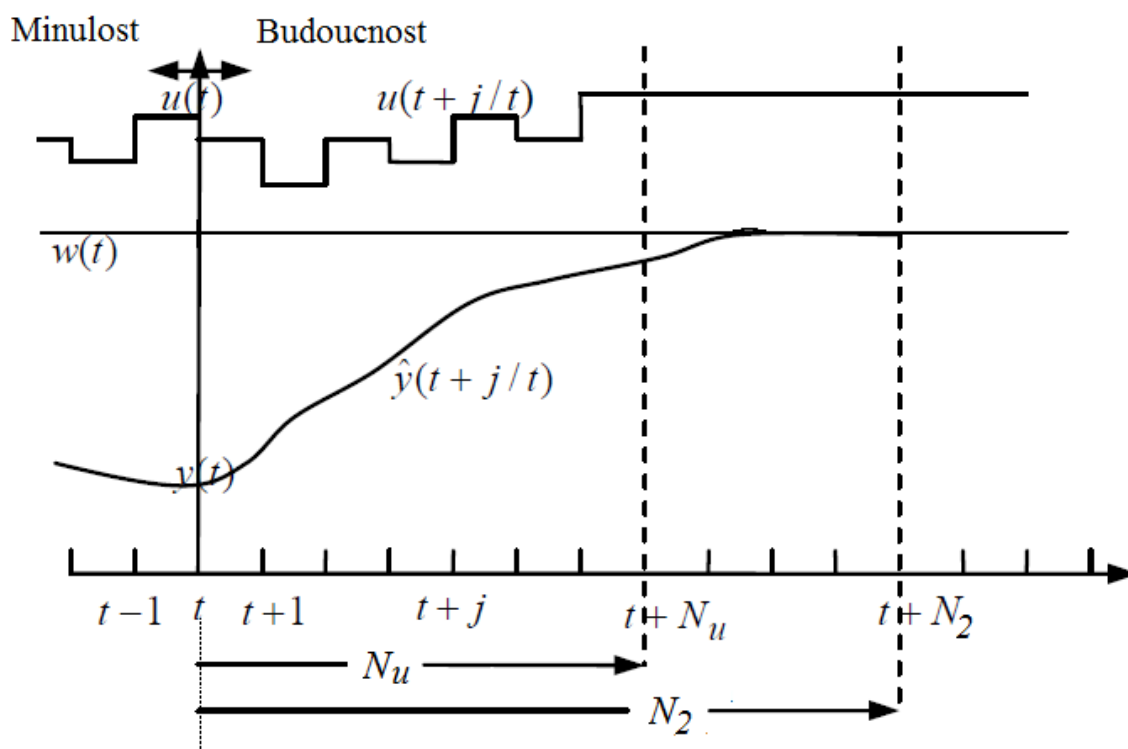
MPC obsahuje čtyři základní myšlenky: [4]

1. Použití explicitního modelu pro predikci budoucích výstupů.
2. Výpočet optimálních akčních zásahů podle minimalizace účelové funkce.
3. Využijeme pouze první akční zásah, celý výpočet opakujeme v další periodě.
4. Nutnost znalosti budoucích hodnot žádané veličiny.



Obr. 3. Bloková struktura prediktivního řízení [4]

Základní struktura MPC je zobrazena na obrázku (Obr. 3). Model má informace o minulých výstupních hodnotách a současné výstupní hodnotě $y(k)$ a minulých hodnotách akčního zásahu $u(k)$ a budoucích predikovaných akčních veličinách. Model generuje N budoucích predikovaných výstupů $[\hat{y}(k+1) \dots \hat{y}(k+N)]$, vzhledem k informacím dostupným v čase k . Tento vektor je porovnán s vektorem budoucích žádaných hodnot, čímž získáme predikované hodnoty odchylky $[\hat{e}(k+1) \dots \hat{e}(k+N)]$. Řešením optimalizační úlohy, kde bereme v potaz účelovou funkci a případná omezení na jednotlivé veličiny je vektor N hodnot budoucích akčních zásahů $[u(k) \dots u(k+N-1)]$. Na systém pošleme první akční zásah a celý výpočet se zopakuje v další periodě.



Obr. 4. Princip MPC [5]

Princip MPC je popsán graficky (Obr. 4), kde:

N_1 je minimální horizont,

N_2 je maximální horizont,

N_u je řídicí horizont.

3.2 Model prediktivního řízení

Model je základním prvkem prediktoru, jenž dokáže predikovat budoucí výstupy. Musí dobře popisovat dynamické chování procesu, může obsahovat i chování poruchy. Dle konkrétní metody MPC můžeme použít model na základě různých popisů. [4]

3.2.1 Impulzní funkce

Prakticky se používá jen několik prvních členů impulzní odezvy. Pomocí impulzní odezvy můžeme modelovat pouze stabilní systém. Výstupní funkce pro impulzní odezvu má tvar

$$y(k) = \sum_{i=1}^N h_i u(k-i) \quad (3.1)$$

3.2.2 Přechodová funkce

Jako základ modelu se používá přechodová funkce, konkrétně několik jejích prvních členů. Model pomocí přechodové funkce lze použít stejně jako impulzní odezvu pouze pro stabilní systémy. Výstupní funkce má tvar

$$y(k) = \sum_{i=1}^N g_i \Delta u(k-i) \quad (3.2)$$

3.2.3 Stavový popis

Model popsaný stavovým popisem má maticový zápis

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) \\ y(k) &= Cx(k) \end{aligned} \quad (3.3)$$

kde:

A je matice systému,

B je matice vstupu,

C je váhová matice stavu.

3.2.4 Přenosové funkce

Přenosová funkce nám určuje vzájemný vztah vstupní a výstupní veličiny. Model na základě přenosové funkce je možné použít i na nestabilní systémy. Model má tvar

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

3.2.5 Funkce s poruchou

Pro vyjádření poruchy se běžně model ve tvaru ARMAX

$$A(z^{-1})y(k) = B(z^{-1})u(k) + C(z^{-1})e_s(k) \quad (3.5)$$

kde $e_s(k)$ je bílý šum. [4]

Případně můžeme využít i jiné modely, v závislosti na konkrétním případě procesu. Pro účely dále zvolené metody GPC (Generalized Predictive Control) budeme využívat pro model přenosovou funkci popř. jeho modifikaci s poruchou.

3.3 Účelová funkce

Standardní účelová funkce musí obsahovat kvadratickou regulační odchylku a kvadratickou změnu přírůstku akční veličiny. Pro účely GPC využijeme účelovou funkci ve tvaru [12]:

$$J = \sum_{i=N_1}^{N_2} [\hat{y}(k+i) - w(k+i)]^2 + \sum_{i=N_1}^{N_u} [\lambda(i)\Delta u(k+i-1)]^2 \quad (3.6)$$

kde:

$\hat{y}(k+i)$ predikované výstupy;

$w(k+i)$ budoucí hodnota žádané veličiny;

$\Delta u(k+i-1)$ budoucí změny akčního zásahu;

$\lambda(i)$ je váhový parametr,

N_1 je minimální horizont,

N_2 je maximální horizont,

N_u je řídicí horizont [12].

Hodnota N_1 by se měla rovnat $T_D + 1$, kde T_D je hodnota dopravního zpoždění. Hodnota N_2 by se měla volit s ohledem na to, abychom dokázali pokrýt významnou část přechodové charakteristiky, obvykle se volí hodnota okolo T_{90} , což je čas, kdy se děj dostane na 90% ustálené hodnoty. Hodnota N_u může snížit výpočetní složitost prediktoru, pokud to výpočetní výkon umožní, volíme hodnotu N_u co nejblíže k hodnotě N_2 , ideálně ji volíme stejnou. Váhový parametr $\lambda(i)$ nám umožní v účelové funkci nastavit, zda bude mít větší váhu regulační odchylka nebo změna přírůstku akční veličiny. Pokud je $\lambda(i) < 1$ bude docíleno lepší sledování trajektorie žádané veličiny, avšak za cenu vyššího rozkmitání akční veličiny. Naopak pokud $\lambda(i) > 1$, pak prediktor se zaměří na snížení změn akčního zásahu, avšak zvýší se hodnota kvadratické odchylky. Hodnotu je nutné nastavit dle konkrétního procesu, nemusí mít konstantní hodnotu, ale můžeme ji volit i ve formě exponenciálních vah. Dále uvažujeme, že hodnota budoucí žádané trajektorie $w(k + i)$ je známá. [4]

Predikovaný výstup \hat{y} můžeme určit jakou součet nucené y_n a volné y_0 odezvy modelu

$$\hat{y} = y_n + y_0 \quad (3.7)$$

kde:

\hat{y} je vektor predikovaných výstupů $\hat{y}^T = [\hat{y}(k + 1) \quad \hat{y}(k + 2) \quad \dots \quad \hat{y}(k + N_2)]$,

y_n je vektor nucené odezvy systému $y_n^T = [y_n(k + 1) \quad y_n(k + 2) \quad \dots \quad y_n(k + N_2)]$,

y_0 je vektor volné odezvy systému $y_0^T = [y_0(k + 1) \quad y_0(k + 2) \quad \dots \quad y_0(k + N_2)]$,

Volná odezva y_0 se vypočítá tak, že se uvažuje pouze s minulými hodnotami akčního zásahu a minulými hodnotami výstupu. Budoucí změna akčního zásahu se uvažuje nulová. Naproti tomu nucená odezva y_n zohledňuje budoucí změnu akčního zásahu a lze ji vypočítat jako

$$y_n = H\Delta u \quad (3.8)$$

kde:

Δu je vektor změn akčního zásahu $\Delta u^T = [\Delta u(k) \quad \Delta u(k + 1) \quad \dots \quad \Delta u(k + N_2 - 1)]$,

H je Jacobiho matice.

Jacobiho matice je parciální derivace modelu dle jednotlivých proměnných. Pro lineární systém můžeme popsat matici ve tvaru:

$$H = \begin{bmatrix} h_1 & 0 & 0 & \cdots & 0 \\ h_2 & h_1 & 0 & \cdots & 0 \\ h_3 & h_2 & h_1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & 0 \\ h_{N_2} & h_{N_2-1} & h_{N_2-2} & \cdots & h_{N_2-N_u+1} \end{bmatrix} \quad (3.9)$$

Jednotlivé koeficienty h_i jsou odezvy na jednotkový skok.

Účelovou funkci (3.6) můžeme vyjádřit ve tvaru:

$$J = (\hat{y} - w)^T (\hat{y} - w) + \lambda \Delta u^T \Delta u \quad (3.10)$$

Do rovnice dosadíme vyjádření predikovaného \hat{y} z rovnic (3.7) a (3.8) a získáme vyjádření ve tvaru:

$$J = (H\Delta u + y_0 - w)^T (H\Delta u + y_0 - w) + \lambda \Delta u^T \Delta u \quad (3.11)$$

Rovnici upravíme do tvaru vhodného pro kvadratické programování: [4]

$$J = \frac{1}{2} \Delta u^T H_H \Delta u + b^T \Delta u + f_0 \quad (3.12)$$

kde:

$$H_H = 2(H^T H + \lambda I),$$

$$b^T = 2(y_0 - w)^T (y_0 - w),$$

$$f = (y_0 - w)^T (y_0 - w).$$

V rovnici (3.12) můžeme zanedbat parametr f_0 , jež není závislý na budoucím akčním zásahu a neovlivní tedy optimalizační řešení. Upravená rovnice má tvar [4] [12]:

$$J = \frac{1}{2} \Delta u^T H_H \Delta u + b^T \Delta u \quad (3.13)$$

3.4 Omezení řídicích veličin

V běžném provozu potřebujeme omezovat akční zásah, výstupní veličinu apod. Důvod je buď čistě fyzikální (maximální průtok ventilu, napětí na motoru, otáčky motoru) nebo může být omezení technologického procesu (maximální teplota). Nejjednodušším běžně používaným řešením je nepočítat s těmito omezeními do optimalizační úlohy a poté nasazení saturace jednotlivých veličin. Výsledek může být funkční, avšak ne vždy. Navíc nám nezajistí optimalizaci dle zadaných kritérií. Omezení je možné aplikovat pouze na akční zásah popř. výstup systému. Problém, který zde může nastat, je tzv. *wind-up efekt*. K tomu může dojít, pokud budeme veličiny omezovat pomocí saturace. Regulátor nemá informaci o omezení, a tak stále zvyšuje akční zásah s cílem dosáhnout žádané hodnoty, kterou nemůže dosáhnout v souvislosti s daným omezením. Následně při snížení žádané hodnoty není schopen dostatečně rychle snížit vysoký akční zásah a dochází tak k nežádoucímu regulačnímu průběhu.

Optimálním řešením je zakomponování omezujících podmínek do optimalizační úlohy. Oproti předešlému řešení bude výsledek vždy optimální dle zvolených kritérií. Velikou výhodou je i možnost omezit výstupní veličiny modelu popř. i jeho vnitřní stavy. Pokud řešíme úlohu s omezením, vede to ke kvadratické objektivní optimalizační úloze s lineárními omezeními. Tento problém se řeší kvadratickým programováním. Úlohy kvadratického programování se řeší různými metodami, neexistuje jediné řešení pro všechny možné případy. Jednou z možností řešení je zakomponování omezujících podmínek přímo do optimalizační úlohy. Další možností řešení je rozdělení problému na množinu jednodušších optimalizačních úloh [4] [13]

Omezující podmínky můžeme rozdělit do dvou základních skupin:

- tvrdá omezení – pevně daná omezení, jež není možné překročit
- měkká omezení – omezení s určitou tolerancí pro překročení samotného omezení.

3.4.1 Tvrdá omezení

Omezení vyjádříme maticovou nerovností

$$Au \leq b \quad (3.14)$$

Nejběžnější omezení jsou:

- akčního zásahu $u_{\min} \leq u(k) \leq u_{\max}$
- výstupní veličiny $y_{\min} \leq y(k) \leq y_{\max}$
- změny akčního zásahu $\Delta u_{\min} \leq \Delta u(k) \leq \Delta u_{\max}$
- překmitu $y(k+i) \leq w(k)$

Omezení akčního zásahu:

$$\begin{aligned}
 &u(k) \leq u_{\max} \\
 &u(k-1) + \Delta u(k) \leq u_{\max} \\
 &\Delta u(k) \leq u_{\max} - u(k-1) \\
 &\begin{bmatrix} 1 & 0 & 0 & \cdots \\ 1 & 1 & 0 & \cdots \\ 1 & 1 & 1 & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} u \leq \begin{bmatrix} u_{\max} - u(k-1) \\ u_{\max} - u(k-1) \\ u_{\max} - u(k-1) \\ \vdots \end{bmatrix} \\
 &Tu \leq u_{\max} - u_{k-1} \\
 &u(k) \geq u_{\min} \\
 &-u(k-1) - \Delta u(k) \leq -u_{\min} \\
 &-\Delta u(k) \geq -u_{\min} + u(k-1) \\
 &\begin{bmatrix} -1 & 0 & 0 & \cdots \\ -1 & -1 & 0 & \cdots \\ -1 & -1 & -1 & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} u \leq \begin{bmatrix} -u_{\min} + u(k-1) \\ -u_{\min} + u(k-1) \\ -u_{\min} + u(k-1) \\ \vdots \end{bmatrix} \\
 &-Tu \leq -u_{\min} + u_{k-1}
 \end{aligned} \quad (3.15)$$

Omezení výstupní veličiny:

$$\begin{aligned}
 y(k) &\leq y_{\max} \\
 Hu + y_0 &\leq y_{\max} \\
 Hu &\leq y_{\max} - y_0 \\
 \begin{bmatrix} h_1 & 0 & 0 & \cdots & 0 \\ h_2 & h_1 & 0 & \cdots & 0 \\ h_3 & h_2 & h_1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ h_{N_2} & h_{N_2-1} & h_{N_2-2} & \cdots & h_{N_2-N_u+1} \end{bmatrix} u &\leq \begin{bmatrix} y_{\max} - y_0(k+1) \\ y_{\max} - y_0(k+2) \\ y_{\max} - y_0(k+3) \\ \vdots \\ y_{\max} - y_0(k+N_2) \end{bmatrix} \\
 y(k) &\geq y_{\min} \\
 -Hu - y_0 &\leq -y_{\min} \\
 -Hu &\leq -y_{\min} + y_0 \\
 -\begin{bmatrix} h_1 & 0 & 0 & \cdots & 0 \\ h_2 & h_1 & 0 & \cdots & 0 \\ h_3 & h_2 & h_1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ h_{N_2} & h_{N_2-1} & h_{N_2-2} & \cdots & h_{N_2-N_u+1} \end{bmatrix} u &\leq \begin{bmatrix} -y_{\min} + y_0(k+1) \\ -y_{\min} + y_0(k+2) \\ -y_{\min} + y_0(k+3) \\ \vdots \\ -y_{\min} + y_0(k+N_2) \end{bmatrix}
 \end{aligned} \tag{3.16}$$

Omezení změny akčního zásahu:

$$\begin{aligned}
 \Delta u(k) &\leq \Delta u_{\max} \\
 \begin{bmatrix} 1 & 0 & 0 & \cdots \\ 0 & 1 & 0 & \cdots \\ 0 & 0 & 1 & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} u &\leq \begin{bmatrix} \Delta u_{\max} \\ \Delta u_{\max} \\ \Delta u_{\max} \\ \vdots \end{bmatrix} \\
 Iu &\leq \Delta u_{\max} \\
 u(k) &\geq u_{\min} \\
 -\Delta u(k) &\geq -u_{\min} + u(k-1) \\
 \begin{bmatrix} -1 & 0 & 0 & \cdots \\ 0 & -1 & 0 & \cdots \\ 0 & 0 & -1 & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} u &\leq \begin{bmatrix} -\Delta u_{\min} \\ -\Delta u_{\min} \\ -\Delta u_{\min} \\ \vdots \end{bmatrix} \\
 -Iu &\leq -\Delta u_{\min}
 \end{aligned} \tag{3.17}$$

Jednotlivé omezení zapíšeme do maticové nerovnosti [4]

$$\begin{bmatrix} I \\ -I \\ T \\ -T \\ H \\ -H \end{bmatrix} u \leq \begin{bmatrix} \Delta u_{\max} \\ \Delta u_{\min} \\ u_{\max} \\ u_{\min} \\ y_{\max} - y_0 \\ y_{\min} - y_0 \end{bmatrix} \quad (3.18)$$

kde:

I je jednotková matice o rozměrech $\begin{bmatrix} N_2 & N_u \end{bmatrix}$,

T je dolní trojúhelníková matice o rozměrech $\begin{bmatrix} N_2 & N_u \end{bmatrix}$,

H je Jacobiho matice o rozměrech $\begin{bmatrix} N_2 & N_u \end{bmatrix}$.

3.4.2 Měkká omezení

U měkkého omezení je možné překročit omezení o tzv. mez tolerance ε . Často se také měkká omezení zadávají jako součást kritéria (3.6). Překročení meze pak vede k růstu kritéria. Jinak se omezení dosazují do rovnice (3.14).

$$\begin{aligned}
 & y(k) \leq y_{\max} + \varepsilon_{\max} \\
 & Hu + y_0 \leq y_{\max} + \varepsilon_{\max} \\
 & Hu \leq y_{\max} + \varepsilon_{\max} - y_0 \\
 & \begin{bmatrix} h_1 & 0 & 0 & \cdots & 0 \\ h_2 & h_1 & 0 & \cdots & 0 \\ h_3 & h_2 & h_1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ h_{N_2} & h_{N_2-1} & h_{N_2-2} & \cdots & h_{N_2-N_u+1} \end{bmatrix} u \leq \begin{bmatrix} y_{\max} + \varepsilon_{\max} - y_0(k+1) \\ y_{\max} + \varepsilon_{\max} - y_0(k+2) \\ y_{\max} + \varepsilon_{\max} - y_0(k+3) \\ \vdots \\ y_{\max} + \varepsilon_{\max} - y_0(k+N_2) \end{bmatrix} \\
 & y(k) \geq y_{\max} + \varepsilon_{\min} \\
 & -Hu - y_0 \leq -y_{\max} - \varepsilon_{\min} \\
 & -Hu \leq -y_{\max} - \varepsilon_{\min} + y_0 \\
 & \begin{bmatrix} h_1 & 0 & 0 & \cdots & 0 \\ h_2 & h_1 & 0 & \cdots & 0 \\ h_3 & h_2 & h_1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ h_{N_2} & h_{N_2-1} & h_{N_2-2} & \cdots & h_{N_2-N_u+1} \end{bmatrix} u \leq \begin{bmatrix} y_{\max} - \varepsilon_{\min} - y_0(k+1) \\ y_{\max} - \varepsilon_{\min} - y_0(k+2) \\ y_{\max} - \varepsilon_{\min} - y_0(k+3) \\ \vdots \\ y_{\max} - \varepsilon_{\min} - y_0(k+N_2) \end{bmatrix}
 \end{aligned} \tag{3.19}$$

4 GENERALIZED PREDICTIVE CONTROL

Generalized Predictive Control je jeden z nejvíce populárních algoritmů prediktivního řízení. Algoritmus byl popsán D. W. Clarkem v roce 1987. [7] Tato metoda je vhodná především pro SISO systémy. Je zde možnost využít i adaptivní řízení. [8] Tato metoda využívá vstupně-výstupní model procesu.

4.1 Odvození prediktoru

Stochastický SISO systém můžeme popsat modelem ARMAX (Auto-Regressive Moving Average with eXogenous inputs) ve tvaru

$$A(z^{-1})y(k) = z^{-d}B(z^{-1})u(k) + C(z^{-1})e_s(k) \quad (4.1)$$

Pro některé průmyslové aplikace, kde jsou poruchy nestacionární, je vhodnější využít model CARIMA (Controlled Auto-Regressive Integrated Moving-Average) ve tvaru [4]

$$A(z^{-1})y(k) = B(z^{-1})u(k) + \frac{C(z^{-1})}{\Delta}e_s(k) \quad (4.2)$$

kde:

$y(k)$ je výstupní signál,

$u(k - 1)$ je vstupní signál,

$e_s(k)$ obsahuje neměřitelné poruchy,

$$A(z^{-1}) = 1 + a_1z^{-1} + a_2z^{-2} + \dots + a_{Na}z^{-Na},$$

$$B(z^{-1}) = b_1z^{-1} + b_2z^{-2} + \dots + b_{Nb}z^{-Nb},$$

$$C(z^{-1}) = 1 + c_1z^{-1} + c_2z^{-2} + \dots + c_{Nc}z^{-Nc}.$$

Pokud uvažujeme i budoucích kroků, pak rovnici vynásobíme členem z^i a poté vydělíme polynomem A . Výsledkem operací je rovnice [4]

$$y(k + i) = \frac{B(z^{-1})}{A(z^{-1})}u(k + i - 1) + \frac{C(z^{-1})}{\Delta A(z^{-1})}e_s(k + i) \quad (4.3)$$

Rovnici můžeme rozdělit polynomiálním dělením $C/\Delta A$ a následnou separací prvních i členů s kladnými mocninami z^i . Získáme rovnici

$$\frac{C(z^{-1})}{\Delta A(z^{-1})} = E_i(z^{-1}) + q^{-i} \frac{F_i(z^{-1})}{\Delta A(z^{-1})} \quad (4.4)$$

Dosadíme pravou stranu rovnice (4.4) do rovnice (4.3) a získáme rovnici ve tvaru

$$y(k+i) = \frac{B(z^{-1})}{A(z^{-1})} u(k+i-1) + E_i(z^{-1}) e_s(k+i) + \frac{F_i(z^{-1})}{\Delta A(z^{-1})} e_s(k) \quad (4.5)$$

Aktuální neměřitelnou poruchu určíme jako

$$e_s(k) = \frac{\Delta A(z^{-1})}{C(z^{-1})} y(k) - \frac{B(z^{-1})}{C(z^{-1})} \Delta u(k-1) \quad (4.6)$$

Dosazením rovnice (4.6) zpět do rovnice (4.5) získáme

$$\begin{aligned} y(k+i) &= \frac{B(z^{-1})}{A(z^{-1})} u(k+i-1) - \frac{F_i(z^{-1})B(z^{-1})}{\Delta A(z^{-1})C(z^{-1})} u(k-1) + \frac{F_i(z^{-1})}{C(z^{-1})} y(k) \\ &\quad + E_i(z^{-1}) e_s(k+i) = \\ &= \frac{B(z^{-1})}{C(z^{-1})} \left[\frac{B(z^{-1})}{\Delta A(z^{-1})} - z^{-i} \frac{F_i(z^{-1})}{\Delta A(z^{-1})} \right] u(k+i-1) + \frac{F_i(z^{-1})}{C(z^{-1})} y(k) \\ &\quad + E_i(z^{-1}) e_s(k+i) \end{aligned} \quad (4.7)$$

Pokud do rovnice (4.7) dosadíme pravou stranu rovnice (4.4) získáme vztah ve tvaru

$$\begin{aligned} y(k+i) &= \frac{B(z^{-1})}{C(z^{-1})} \left[E_i(z^{-1}) + z^{-i} \frac{F_i(z^{-1})}{\Delta A(z^{-1})} - z^{-i} \frac{F_i(z^{-1})}{\Delta A(z^{-1})} \right] u(k+i-1) \\ &\quad + \frac{F_i(z^{-1})}{C(z^{-1})} y(k) + E_i(z^{-1}) e_s(k+i) \end{aligned} \quad (4.8)$$

Pomocí polynomiálního dělení oddělíme současné a budoucí přírůstky řízení a získáme výslednou rovnici ve tvaru

$$\begin{aligned} y(k+i) &= G_i(z^{-1}) \Delta u(k+i-1) + \frac{\Gamma_i(z^{-1})}{C(z^{-1})} \Delta u(k-1) + \frac{F_i(z^{-1})}{C(z^{-1})} y(k) \\ &\quad + E_i(z^{-1}) e_s(k+i) \end{aligned} \quad (4.9)$$

Pokud chceme dosáhnout optimálního výstupu prediktoru, můžeme zanedbat budoucí hodnoty poruchy. Výsledná rovnice po úpravě má tvar

$$y(k+i) = G_i(z^{-1})\Delta u(k+i-1) + \frac{\Gamma_i(z^{-1})}{C(z^{-1})}\Delta u(k-1) + \frac{F_i(z^{-1})}{C(z^{-1})}y(k) \quad (4.10)$$

Poslední dva prvky rovnice označíme jako $y_0(k+i)$. Po této úpravě rovnice získá tvar [4]

$$y(k+i) = G_i(z^{-1})\Delta u(k+i-1) + y_0(k+i) \quad (4.11)$$

4.2 Optimální řízení

Uvažujeme rovnici účelové funkce ve tvaru (3.11)

$$J = (H\Delta u + y_0 - w)^T(H\Delta u + y_0 - w) + \lambda\Delta u^T\Delta u \quad (4.12)$$

Tuto rovnici roznásobíme

$$J = \Delta u^T(H^TH + \lambda I)\Delta u + \Delta u^TH^T(y_0 - w) + (y_0 - w)^TH\Delta u + (y_0 - w)^T(y_0 - w) \quad (4.13)$$

Minimální hodnotu účelové funkce získáme tak, že parciální derivaci rovnice (4.13) podle Δu položíme rovnu nule. Výsledná derivovaná rovnice má tvar

$$2(H^TH + \lambda I)\Delta u + 2H^T(y_0 - w) = 0 \quad (4.14)$$

Tím přesuneme problém optimalizační úlohy na problém algebraický. Po úpravě získáme

$$\Delta u = -(H^TH + \lambda I)^{-1}H^T(y_0 - w) = -H_H^{-1}g \quad (4.15)$$

kde:

$H_H = (H^TH + \lambda I)$ je Hessova matice,

$g = H^T(y_0 - w)$ je gradient.

Pokud označíme první řádek matice $(H^TH + \lambda I)^{-1}H^T$ jako K , tak aktuální změnu řídicí veličiny vypočítáme ze vztahu

$$\Delta u = K(w - y_0) \quad (4.16)$$

Pokud je tedy rozdíl mezi budoucí trajektorií žádané veličiny a volnou odezvou, je změna akční veličiny vypočtena podle vektoru K . [4] [12]

4.3 Výpočet prediktoru pro model druhého řádu

4.3.1 Prediktor bez omezujících podmínek

Uvažujeme proces popsaný modelem CARIMA

$$A(z^{-1})y(k) = B(z^{-1})u(k-1) + \frac{C(z^{-1})}{\Delta}e_s(k) \quad (4.17)$$

kde:

$$A(z^{-1}) = 1 + a_1 z^{-1} + a_2 z^{-2},$$

$$B(z^{-1}) = b_1 z^{-1} + b_2 z^{-2},$$

$$C(z^{-1}) = 1.$$

Po dosazení polynomů do rovnice obdržíme [4]

$$(1 - z^{-1})(1 + a_1 z^{-1} + a_2 z^{-2})y(k) = (b_1 z^{-1} + b_2 z^{-2})\Delta u(k) + e_s(k) \quad (4.18)$$

Po roznásobení obdržíme:

$$[1 + (a_1 - 1)z^{-1} + (a_1 - a_2)z^{-2} - a_2 z^{-3}]y(k) = (b_1 z^{-1} + b_2 z^{-2})\Delta u(k) + e_s(k) \quad (4.19)$$

Převédeme na časové vyjádření

$$y(k) = (1 - a_1)y(k-1) - 1 + (a_1 - a_2)y(k-2) + a_2 y(k-3) + b_1 \Delta u(k-1) + b_2 \Delta u(k-2) + e_s(k) \quad (4.20)$$

Pokud budeme uvažovat, že neměřitelná porucha $e_s(k) = 0$ a parametry $N_1 = 1$, $N_2 = 3$,

$N_u = 3$. Tři budoucí predikované výstupy budou mít tvar:

$$\begin{aligned} \hat{y}(k+1) &= (1 - a_1)y(k) + (a_1 - a_2)y(k-1) + a_2 y(k-2) + b_1 \Delta u(k) + b_2 \Delta u(k+1) \\ \hat{y}(k+2) &= (1 - a_1)y(k+1) + (a_1 - a_2)y(k) + a_2 y(k-1) + b_1 \Delta u(k+1) + b_2 \Delta u(k) \\ \hat{y}(k+3) &= (1 - a_1)y(k+2) + (a_1 - a_2)y(k+1) + a_2 y(k) + b_1 \Delta u(k+2) + b_2 \Delta u(k+1) \end{aligned} \quad (4.21)$$

Vyšší odhady vyjádříme pomocí nižších, poté separujeme veličiny známé a veličiny predikované a rovnice zapíšeme v maticové podobě: [12]

$$\begin{aligned}
 \begin{bmatrix} \hat{y}(k+1) \\ \hat{y}(k+2) \\ \hat{y}(k+3) \end{bmatrix} &= \begin{bmatrix} b_1 & 0 & 0 \\ b_1(1-a_1)+b_2 & b_1 & 0 \\ (a_1-a_2)b_1+(1-a_1)^2b_1+(1-a_1)b_2 & b_1(1-a_1)+b_2 & b_1 \end{bmatrix} \times \begin{bmatrix} \Delta u(k) \\ \Delta u(k+1) \\ \Delta u(k+2) \end{bmatrix} + \\
 &+ \begin{bmatrix} (1-a_1) & (a_1-a_2) \\ (1-a_1)^2+(a_1-a_2) & (1-a_1)(a_1-a_2)+a_2 \\ (1-a_1)^3+2(1-a_1)(a_1-a_2)+a_2 & (1-a_1)^2(a_1-a_2)+a_2(1-a_1)+(a_1-a_2)^2 \end{bmatrix} \\
 &\times \begin{bmatrix} y(k) \\ y(k-1) \\ y(k-2) \end{bmatrix} = \\
 &= \begin{bmatrix} h_1 & 0 & 0 \\ h_2 & h_1 & 0 \\ h_3 & h_2 & h_1 \end{bmatrix} \times \begin{bmatrix} \Delta u(k) \\ \Delta u(k+1) \\ \Delta u(k+2) \end{bmatrix} + \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{bmatrix} \times \begin{bmatrix} y(k) \\ y(k-1) \\ y(k-2) \\ \Delta u(k-1) \end{bmatrix}
 \end{aligned} \tag{4.22}$$

Matice H má rozměry $[N_2, N_u]$ pokud bychom zvýšili parametr N_2 přidáme další řádek, u N_u další sloupec. Matice má vždy stejné hodnoty ve směru hlavní diagonály. Nové hodnoty se vypočítají rekursivně dle vzorce

$$h_i = (1-a_1)h_{i-1} + (a_1-a_2)h_{i-2} + a_2h_{i-3}, \quad i \geq 4 \tag{4.23}$$

Matice P má rozměr $[N_2, 4]$ pro přenos druhého řádu. Při zvýšení parametru zvýšili parametr N_2 zvýšíme počet řádků. Výpočet dalších parametrů se provádí rekursivně dle vzorce [12]

$$\begin{aligned}
 p_{(i-1)} &= (1-a_1)p_{(i-1-1)} + (a_1-a_2)p_{(i-2-1)} + a_2p_{(i-3-1)}, \\
 p_{(i-2)} &= (1-a_1)p_{(i-1-2)} + (a_1-a_2)p_{(i-2-2)} + a_2p_{(i-3-2)}, \\
 p_{(i-3)} &= (1-a_1)p_{(i-1-3)} + (a_1-a_2)p_{(i-2-3)} + a_2p_{(i-3-3)}, \\
 p_{(i-4)} &= (1-a_1)p_{(i-1-4)} + (a_1-a_2)p_{(i-2-4)} + a_2p_{(i-3-4)}, \quad i \geq 4
 \end{aligned} \tag{4.24}$$

Pro výpočet akčního zásahu je potřeba určit vektor K , jenž je prvním řádkem matice O

$$O = (H^T H + \lambda I)^{-1} H^T \tag{4.25}$$

Vektor K má poté hodnotu

$$H_H = (H^T H + \lambda I)^{-1} \quad (4.26)$$

Volnou odezvu vypočítáme jako koeficienty prvního řádku matice Q

$$Q = OP = \begin{bmatrix} q_{11} & q_{12} & q_{13} & q_{14} \\ q_{21} & q_{22} & q_{23} & q_{24} \\ q_{31} & q_{32} & q_{33} & q_{34} \end{bmatrix} \quad (4.27)$$

Výpočet změny akčního zásahu je dán rovnicí [4]

$$\begin{aligned} \Delta u(k) = & -q_{11}y(k) - q_{12}y(k-1) - q_{13}y(k-2) - q_{14}\Delta(k-1) + o_{11}w(k+1) + \\ & + o_{12}w(k+2) + o_{13}w(k+3) \end{aligned} \quad (4.28)$$

4.3.2 Prediktor s omezujícími podmínkami

Základ prediktoru je optimalizační úloha, do níž jsou zakomponovány omezující podmínky.

Minimální kritérium vychází z rovnice účelové funkce (3.12) a má tvar [12]

$$\min_{\Delta u} 2g^T \Delta u^T + \Delta u^T H_H \Delta u \quad (4.29)$$

kde:

$H_H = (H^T H + \lambda I)^{-1}$ je Hessova matice,

$g = H^T (y_0 - w)$ je gradient.

Omezující podmínky dosadíme do rovnice (3.14) pro tento konkrétní případ budeme uvažovat parametry $N_2 = N_u = 3$ a omezení akčního zásahu a omezení výstupní veličiny.

Výsledná rovnice bude

$$\begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \\ -1 & 0 & 0 \\ -1 & -1 & 0 \\ -1 & -1 & -1 \\ h_1 & 0 & 0 \\ h_2 & h_1 & 0 \\ h_3 & h_2 & h_1 \\ -h_1 & 0 & 0 \\ -h_2 & -h_1 & 0 \\ -h_3 & -h_2 & -h_1 \end{bmatrix} \begin{bmatrix} \Delta u(k) \\ \Delta u(k+1) \\ \Delta u(k+2) \end{bmatrix} \leq \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \begin{bmatrix} [u_{\max} - u(k-1)] \\ [u_{\min} - u(k-1)] \\ [y_{\max} - y_0] \\ [y_{\min} - y_0] \end{bmatrix} \quad (4.30)$$

Tyto jednotlivé kritéria zadané pomocí matic budeme řešit kvadratickým programováním.

Výsledkem je vektor změn akčního zásahu $[\Delta u(k) \ \Delta u(k+1) \ \dots \ \Delta u(k+N_u-1)]$.

5 OFFSET-FREE LINEAR MODEL PREDICTIVE CONTROL

Offset-free je variantou MPC, která dokáže eliminovat poruchu, která je způsobena rozdílem mezi lineárním modelem prediktoru a řízeným systémem. Budoucí porucha je odhadována na základě poruch minulých. [14]

5.1 Odvození prediktoru

Při odvození budeme vycházet s ARX modelu ve tvaru [15]

$$y(k) + \sum_{i=1}^{n_a} a_i y(k-i) = \sum_{i=0}^{n_b} b_i u(k-i) + \sum_{i=0}^{n_d} d_i v(k-i) + e(k) \quad (5.1)$$

kde:

$$d(k) = \sum_{i=0}^{n_d} d_i v(k-i) \text{ je porucha.}$$

Pokud budeme uvažovat model 2. řádu, můžeme model zapsat za předpokladu, že bílý šum $e(k) = 0$ jako:

$$y(k) + a_1 y(k-1) + a_2 y(k-2) = b_1 u(k-1) + b_2 u(k-2) + d(k) \quad (5.2)$$

Budoucích N_2 výstupů systému lze zapsat pomocí rovnice (5.2)

$$\begin{aligned} y(k+1) + a_1 y(k) + a_2 y(k-1) &= b_1 u(k) + b_2 u(k-1) + d(k) \\ \vdots \\ y(k+N_2) + a_1 y(k+N_2-1) + a_2 y(k+N_2-2) &= b_1 u(k+N_2-1) + \\ &+ b_2 u(k+N_2-2) + d(k+N_2-1) \end{aligned} \quad (5.3)$$

Výsledek upravíme do maticové podoby, získáme:

$$A \cdot \vec{y} = B \cdot \vec{u} + \vec{d} \quad (5.4)$$

kde:

$$A \text{ je matice } [N_2 \quad N_2 + 2], [A_r \mid A_p] = \left[\begin{array}{cc|ccccccc} a_2 & a_1 & 1 & 0 & \cdots & 0 & 0 & 0 \\ 0 & a_2 & a_1 & 1 & \cdots & 0 & 0 & 0 \\ 0 & 0 & a_2 & a_1 & \cdots & 0 & 0 & 0 \\ 0 & 0 & 0 & a_2 & \cdots & 1 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & a_1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & a_2 & a_1 & 1 \end{array} \right],$$

$$B \text{ je matice } [N_2 \quad N_2 + 1], B = [B_t \mid B_p] = \begin{bmatrix} b_2 & b_1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & b_2 & b_1 & 0 & \cdots & 0 & 0 \\ 0 & 0 & b_2 & b_1 & \cdots & 0 & 0 \\ 0 & 0 & 0 & b_2 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & b_1 & 0 \\ 0 & 0 & 0 & 0 & 0 & b_2 & b_1 \end{bmatrix},$$

$$\vec{y} = [y(k-1) \quad y(k) \quad \cdots \quad y(k+N_2)]^T,$$

$$\vec{u} = [u(k-1) \quad u(k) \quad \cdots \quad u(k+N_2-1)]^T,$$

$$\vec{d} = [d \quad d \quad \cdots \quad d]^T \text{ [15].}$$

Rovnici (5.4) můžeme zapsat ve tvaru:

$$[A_t \mid A_p] \cdot \begin{bmatrix} \vec{y}_t \\ \vec{y}_p \end{bmatrix} = [B_t \mid B_p] \cdot \begin{bmatrix} \vec{u}_t \\ \vec{u}_p \end{bmatrix} + \vec{d} \quad (5.5)$$

Můžeme minimalizovat vektor \vec{y}_p , pak získáme:

$$\vec{y}_p = A_p^{-1} (B_p \vec{u}_p) + A_p^{-1} (B_t \vec{u}_t + \vec{d} - A_t \vec{y}_t) \quad (5.6)$$

Vycházíme s účelové funkce popsanou vzorcem (3.6), kterou upravíme do tvaru:

$$J = (w - \vec{y}_p)^T (w - \vec{y}_p) + \lambda \Delta \vec{u}^T \Delta u \quad (5.7)$$

Jednoduchou úpravou získáme Δu :

$$\begin{aligned} \begin{bmatrix} \Delta u(k) \\ \Delta u(k+1) \\ \vdots \\ \Delta u(k+N_2) \end{bmatrix} &= \begin{bmatrix} u(k) - u(k-1) \\ u(k+1) - u(k) \\ \vdots \\ u(k+N_2) - u(k+N_2-1) \end{bmatrix} = \begin{bmatrix} -1 & 1 & \cdots & 0 & 0 \\ 0 & -1 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & 1 & 0 \\ 0 & 0 & 0 & -1 & 1 \end{bmatrix} \cdot \begin{bmatrix} \vec{u}_t \\ \vec{u}_p \end{bmatrix} = \\ &= [R_t \mid R_p] \cdot \begin{bmatrix} \vec{u}_t \\ \vec{u}_p \end{bmatrix} \end{aligned} \quad (5.8)$$

Poté $\Delta \vec{u}$ bude mít tvar:

$$\Delta \vec{u} = R_d \vec{u}_t + R_p \vec{u}_p \quad (5.9)$$

Účelovou funkci upravíme do tvaru:

$$J = \left[A_p^{-1} (B_p \vec{u}_p) + A_p^{-1} (B_t \vec{u}_t + \vec{d} - A_t \vec{y}_t) \right]^T \left[A_p^{-1} (B_p \vec{u}_p) + A_p^{-1} (B_t \vec{u}_t + \vec{d} - A_t \vec{y}_t) \right] + \lambda (R_d \vec{u}_t + R_p \vec{u}_p)^T (R_d \vec{u}_t + R_p \vec{u}_p) \quad (5.10)$$

Rovnici (5.10) převedeme do tvaru vhodného pro kvadratické programování:

$$\min_{\vec{u}_p} \frac{1}{2} \vec{u}_p^T \cdot H \cdot \vec{u}_p + f^T \vec{u}_p \quad (5.11)$$

kde:

$$H = \frac{\left[B_p^T (A_p^{-1})^T A_p^{-1} B_p + R_p^T \lambda R_p \right] + \left[B_p^T (A_p^{-1})^T A_p^{-1} B_p + R_p^T \lambda R_p \right]^T}{2},$$

$$f = \left(\frac{A_p}{B_t \vec{u}_t + \vec{d} - A_t \vec{y}_t} - \vec{w} \right)^T A_p^{-1} B_p + \vec{u}_t^T R_t^T \lambda R_p.$$

5.2 Výpočet poruchy

Pro výpočet poruchy upravíme vektory rovnice (5.4)

$$A \cdot \vec{y} = B \cdot \vec{u} + \vec{d} \quad (5.12)$$

kde:

$$\vec{y} = [y(k - N_2 - 1) \quad y(k - N_2) \quad \dots \quad y(k)]^T,$$

$$\vec{u} = [u(k - N_2 - 1) \quad u(k - N_2) \quad \dots \quad u(k - 1)]^T,$$

$$\vec{d} = [d(k - N_2 + 1) \quad d(k - N_2 + 1) \quad \dots \quad d(k)]^T.$$

Matice A a B zůstanou nezměněny. Vektor \vec{d} poté vypočteme ze vztahu

$$\vec{d} = A \cdot \vec{y} - B \cdot \vec{u}. \quad (5.13)$$

Výsledný skalár d použitý pro predikci budoucího řízení můžeme pro SISO systém uvažovat ve tvaru:

$$d = \frac{\sum \vec{d}}{N_2}. \quad (5.14)$$

Výpočet skaláru d můžeme provést i pomocí sofistikovanějších metod jako jsou výpočty s exponenciálním nebo adaptivním zapomínáním apod. [14] [15]

6 IDENTIFIKACE MODELU

6.1 Metoda nejmenších čtverců

Regresní metoda pro vyšetření statických a dynamických vlastností zkoumaného systému. MNČ byla popsána již v roce 1795, kdy ji využil Gauss k výpočtu drah planet. [11] Systém můžeme popsat pomocí vektorů

$$y(k) = \Theta^T(k)\Phi(k) + e_s(k) \quad (6.1)$$

kde:

$\Theta(k)$ je vektor parametrů systému,

$\Phi(k)$ je vektor dat,

$e_s(k)$ je neměřitelná porucha.

Vektor parametrů obsahuje odhady jednotlivých parametrů systému

$$\hat{\Theta}^T(k) = [\hat{a}_1 \quad \hat{a}_2 \quad \dots \quad \hat{a}_n \quad \hat{b}_1 \quad \hat{b}_2 \quad \dots \quad \hat{b}_m] \quad (6.2)$$

Přenosová funkce identifikovaného systému má obecný tvar

$$G(z) = \frac{b_1 z^{-1} + b_2 z^{-2} + \dots + b_m z^{-m}}{1 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_n z^{-n}} \quad (6.3)$$

Cílem MNČ je přiblížit odhad parametrů, co nejblíže ke skutečným parametrům systému. Tedy aby chyba vyjádřená jako rozdíl mezi naměřenými hodnotami $y(k)$ a hodnotami dopočtenými z odhadů parametrů \hat{y} byl co nejmenší. Budeme minimalizovat kritérium kvadratické chyby

$$J = \sum_{k=1}^N e^2(k) = \sum_{k=1}^N \left[y(k) - \sum_{l=k}^n a_l f_l(k) \right]^2 \quad (6.4)$$

kde:

a_l jsou jednotlivé parametry vektoru Θ ,

f_l funkce hodnot závisle proměnných (výstup, akční zásah).

Pokud roznásobíme rovnici (5.4) získáme soustavu N rovnice o n neznámých

$$\begin{aligned} y(1) &= a_1 f_1(1) + a_2 f_2(1) + \dots + a_n f_n(1) + e(1) \\ y(2) &= a_1 f_1(2) + a_2 f_2(2) + \dots + a_n f_n(2) + e(2) \\ &\vdots \\ y(N) &= a_1 f_1(N) + a_2 f_2(N) + \dots + a_n f_n(N) + e(N) \end{aligned} \quad (6.5)$$

Výstup systému teď můžeme zapsat jako

$$y = F\Theta + e \quad (6.6)$$

kde:

$$y^T = [y(1) \ y(2) \ \dots \ y(N)],$$

$$\Theta^T = [a_1 \ a_2 \ \dots \ a_n],$$

$$e^T = [e(1) \ e(2) \ \dots \ e(N)],$$

$$F = \begin{bmatrix} f_1(1) & f_2(1) & \dots & f_n(1) \\ f_1(2) & f_2(2) & \dots & f_n(2) \\ \vdots & \vdots & \ddots & \vdots \\ f_1(N) & f_2(N) & \dots & f_n(N) \end{bmatrix}.$$

Úpravou kvadratického kritéria na vektorový zápis získáme minimalizační kritérium ve tvaru

$$J = e^T e = (y - F\Theta)^T (y - F\Theta). \quad (6.7)$$

Minimum z kritéria získáme, pokud parciální derivaci J podle Θ položíme rovnu nule. Po úpravě získáme maticový zápis pro odhad parametrů [11] [16]

$$\Theta = (F^T F)^{-1} F^T y \quad (6.8)$$

6.2 Rekurzivní metoda nejmenších čtverců

Rekurzivní metoda nejmenších čtverců je modifikací klasické MNČ. Kdy je odhad počítán průběžně v každém kroku a je závislý na odhadu v kroku předešlém. Oproti jednorázové identifikaci je méně náročná na velikost paměti. Minimalizační kritérium má tvar

$$J = \frac{1}{2} [y(k) - \hat{y}(k)]^T [y(k) - \hat{y}(k)]. \quad (6.9)$$

Konečná rovnice pro výpočet odhadu parametrů má tvar

$$\hat{\Theta}(k) = \hat{\Theta}(k-1) + \frac{C(k-1)\Phi(k-1)}{1 + \xi(k)} \hat{e}(k). \quad (6.10)$$

$C(k)$ je kovarianční matice, která se také počítá rekurzivně dle vztahu

$$C(k) = C(k-1) - \frac{C(k-1)\Phi(k)\Phi^T(k)C(k-1)}{1 + \xi(k)}. \quad (6.11)$$

$\xi(k)$ je pomocný skalár, jeho výpočet je dle vztahu

$$\xi(k) = \Phi^T(k)C(k-1)\Phi(k). \quad (6.12)$$

$\hat{e}(k)$ je chyba predikce, jež lze vypočítat jako [11] [16]

$$\hat{e}(k) = [y(k) - \Phi^T(k-1)\hat{\Theta}(k-1)]. \quad (6.13)$$

6.3 Rekurzivní metoda nejmenších čtverců s exponenciálním zapomínáním

U použití klasické RMNČ mají stejný dopad na výsledné odhady všechny vstupní a výstupní data. U neměnných lineárních systémů, získáme velmi přesné výsledky, problém může nastat u nelineárních systémů a systémů, kde se parametry mění v čase. RMNČ s exponenciálním zapomínáním je tedy modifikací klasické metody, kdy novější data mají větší vliv na odhad parametrů než data starší. Minimalizační funkce má tvar:

$$J = \frac{1}{2} [y(k) - \hat{y}(k)]^T W [y(k) - \hat{y}(k)]. \quad (6.14)$$

kde W je váhová matice:

$$W = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & \varphi & \cdots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & 0 & \varphi^{n-1} \end{bmatrix}. \quad (6.15)$$

kde φ je koeficient zapomínání, jež volíme v rozsahu $0 < \varphi \leq 1$. Čím je koeficient menší, tím je zapomínání dat rychlejší. Obvykle se volí hodnota koeficientu okolo 0.95, v závislosti na daném systému. [16]

6.4 Rekurzivní metoda nejmenších čtverců s adaptivním směrovým zapomínáním

RMNČ s adaptivním směrovým zapomínáním je modifikací metody s exponenciálním zapomínáním. Hlavním rozdílem je změna koeficientu zapomínání v závislosti na vstupních datech. Pokud se parametry modelu v čase téměř nemění, pak se hodnota koeficientu zvyšuje a naopak. Odhad parametrů se počítá stejně jako u klasické RMNČ

$$\hat{\Theta}(k) = \hat{\Theta}(k-1) + \frac{C(k-1)\Phi(k-1)}{1 + \xi(k)} \hat{e}(k). \quad (6.16)$$

Rozdílně se počítá kovarianční matice $C(k)$

$$C(k) = C(k-1) - \frac{C(k-1)\Phi(k)\Phi^T(k)C(k-1)}{\varepsilon^{-1}(k) + \xi(k)}. \quad (6.17)$$

$\varepsilon(k)$ je pomocný parametr, jež určíme se vztorce

$$\varepsilon = \varphi(k) - \frac{1 - \varphi(k)}{\xi(k-1)}. \quad (6.18)$$

Hodnotu koeficientu zapomínání určíme dle vztahu)

$$\varphi(k) = \frac{1}{1 + (1 + \rho) \left\{ \ln(1 + \xi) + \left[\eta \frac{\nu(k) + 1}{1 + \xi + \eta} - 1 \right] \frac{\xi}{1 + \xi} \right\}}. \quad (6.19)$$

kde skalární parametry určíme ze vztahů

$$\eta = \frac{[y(k) - \hat{\Theta}^T(k-1)\Phi(k-1)]^2}{\lambda(k)} \quad (6.20)$$

$$\nu(k) = \varphi(k-1)[\nu(k-1) + 1] \quad (6.21)$$

$$\lambda(k) = \varphi(k-1) \left\{ \lambda(k-1) + \frac{[y(k) - \hat{\Theta}^T(k-1)\Phi(k-1)]^2}{1 + \xi} \right\}. \quad (6.22)$$

Doporučená volba prvotních parametrů $\varphi(0)=1$, $\lambda(0)=0.001$, $\nu(0)=10^{-6}$, $\rho=0.99$.

Prvky na hlavní diagonále matice $C(0)$ volíme 10^3 . [11] [16]

II. PRAKTICKÁ ČÁST

7 SERVOMECHANISMUS AMIRA DR300

Jako model pro následné otestování prediktivních algoritmů, jsem zvolil servomechanismus Amira DR300. Model je nelineární jednorozměrný systém. Jedná se o servomechanismus tvořený dvěma stejnosměrnými elektromotory navzájem spojenými pevnou hřídelí. Mechanismus obsahuje tachodynamo určené k měření rychlosti otáčení a IRC senzor určený k měření natočení hřídele. Na straně servomechanismu se o komunikaci stará převodní skříň, která mimo jiné model napájí elektrickou energií. Pro komunikaci s PC slouží multifunkční karta MF 614 od firmy Humusoft. Na straně počítače využívám program Matlab/Simulink a jeho Toolbox pro real-time komunikaci s kartou MF 614. [17] [18]



Obr. 5. Servomechanismus DR300 AMIRA [17]

7.1 Matematický popis modelu

Základní rovnice popisující model mají tvar [17] [19]

$$\begin{aligned} L \frac{di(t)}{dt} &= -Ri(t) - k_e \omega(t) + u(t) \\ J \frac{d\omega(t)}{dt} &= k_m i(t) - b\omega(t) - m_z(t) \end{aligned} \quad (7.1)$$

kde:

$u(t)$ [V] je vstupní napětí na motoru,

$i(t)$ [A] je elektrický proud motoru,

$\omega(t)$ [s^{-1}] jsou otáčky motoru,

$\varphi(t)$ [$ot \cdot min^{-1}$] je úhel natočení hřídele,

$R [\Omega]$. je elektrický odpor motoru,

$L [H]$. je elektromagnetická indukčnost motoru,

$J [Kg \cdot m^2 \cdot s^{-1}]$. je úhel moment setrvačnosti motoru,

$m_z [N \cdot m]$. je mechanický zatěžovací moment motoru,

$k_m [Kg^2 \cdot m^2 \cdot s^{-1}]$. je mechanická konstanta motoru,

$k_e [s \cdot V^{-1}]$. je elektrická konstanta motoru,

$b [Kg \cdot m^2 \cdot s^{-1}]$. je konstanta tření motoru.

$J [Kg \cdot m^2 \cdot s^{-1}]$. je úhel moment setrvačnosti motoru,

$J [Kg \cdot m^2 \cdot s^{-1}]$. je úhel moment setrvačnosti motoru,

Po úpravě získáme soustavu rovnic ve tvaru:

$$\begin{aligned}\frac{di(t)}{dt} &= -\frac{R}{L}i(t) - \frac{k_e}{L}\omega(t) + \frac{1}{L}u(t) \\ \frac{d\omega(t)}{dt} &= -\frac{k_m}{J}i(t) - \frac{b}{J}\omega(t) - \frac{1}{J}m_z(t)\end{aligned}\quad (7.2)$$

Pokud budeme uvažovat, že napětí $u(t)$ je vstupním parametrem systému a rychlost otáčení motoru $\omega(t)$ je výstupním parametrem systému, můžeme zapsat tyto rovnice ve formě stavového popisu:

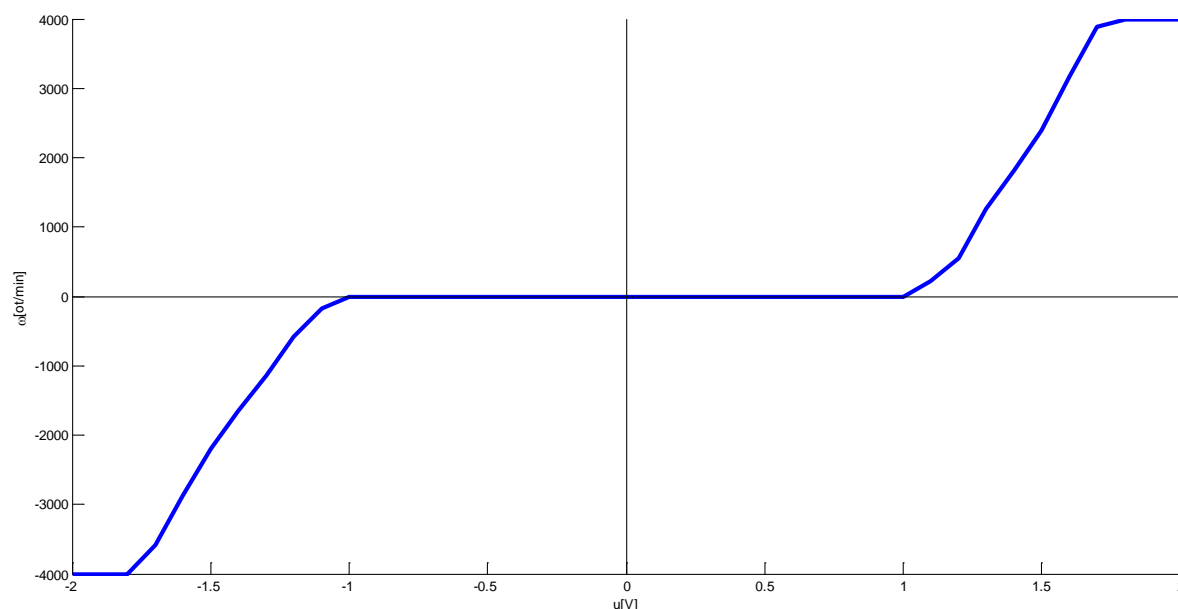
$$\begin{aligned}\begin{bmatrix} \frac{di(t)}{dt} \\ \frac{d\omega(t)}{dt} \end{bmatrix} &= \begin{bmatrix} -\frac{R}{L} & -\frac{k_e}{L} \\ \frac{k_m}{J} & -\frac{b}{J} \end{bmatrix} \cdot \begin{bmatrix} i(t) \\ \omega(t) \end{bmatrix} + \begin{bmatrix} \frac{1}{L} \\ -\frac{1}{J} \end{bmatrix} \cdot \begin{bmatrix} u_m(t) \\ m_z(t) \end{bmatrix} \\ y(t) = \omega(t) &= [0 \quad 1] \cdot \begin{bmatrix} i(t) \\ \omega(t) \end{bmatrix}\end{aligned}\quad (7.3)$$

Ze stavového popisu systému (7.3) můžeme určit výsledný přenos systému [18]:

$$G(s) = \frac{Y(s)}{U(s)} = \frac{\frac{k_m}{LJ}}{s^2 + \left(\frac{R}{L} + \frac{b}{J}\right)s + \frac{Rb + k_e k_m}{LJ}}\quad (7.4)$$

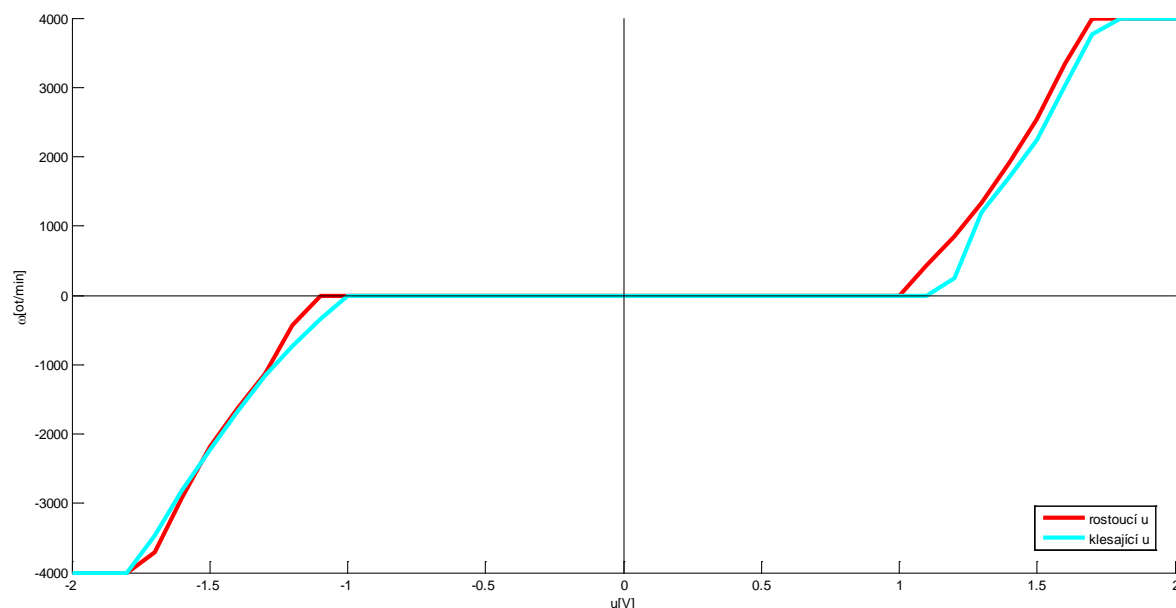
7.2 Statická charakteristika

Pro měření statické charakteristiky jsem zvolil rozsah vstupního napětí od -2V do 2V, s krokem měření 0.1V. Pro tyto hodnoty vstupního napětí budeme zaznamenávat ustálenou hodnotu napětí na tachogenerátoru, jenž převedeme na otáčky motoru.



Obr. 6. Zprůměrovaná statická charakteristika servomechanismu Amira DR300

Ze statické charakteristiky je patrné pásmo necitlivosti pro vstupní napětí od -1.1V do 1.1V. Dále je patrné maximální omezení rychlosti otáček na hodnotě 4000 ot/min a minimální omezení -4000 ot/min. Vyšší rychlost motoru již nejsme schopni tachodynamem zaznamenat. Maximální hodnotu otáček motoru dosáhneme již při vstupním napětí 2V resp. -2V. Maximální a minimální hodnota řídicího napětí, jež můžeme poslat přes komunikační kartu je 10V resp. -10V.



Obr. 7. Statická charakteristika servomechanismu pro rostoucí a klesající vstupní napětí

U statické charakteristiky je vidět rozdíl chování v případě, že napětí zvyšujeme nebo naopak snižujeme, což je patrné na (Obr. 7). Tyto průběhy jsou navíc značně ovlivněny jak okolními vlivy, tak i samotnými vlivy uvnitř servomotoru. V praxi tedy nejsme schopni naměřit vícekrát po sobě totožná data, jelikož dynamické vlastnosti modelu se neustále mění.

7.3 Identifikace systému

Model Amira DR300 budeme nejprve identifikovat pomocí jednorázové metody nejmenších čtverců popsané v kapitole 6.1. Ze statické charakteristiky je zřejmé, že díky pásmu necitlivosti dochází k posunutí statické charakteristiky, tedy i k posunutí linearizovaného modelu. Pokud bychom tento posun neuvažovali, bude samotný výpočet nepřesný. Proto tuto metodu lehce modifikujeme. Matematicky tuto modifikaci můžeme popsat pro model druhého řádu jako:

$$y(k) = -a_1 y(k-1) - a_2 y(k-2) + b_1 u(k-1) + b_2 u(k-2) + c \quad (7.5)$$

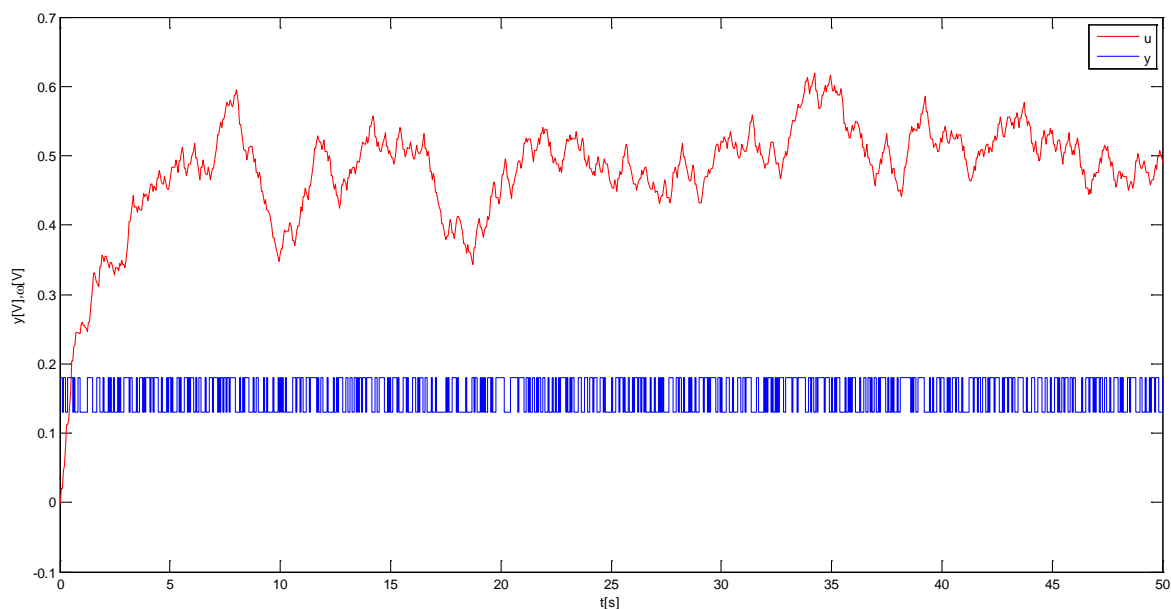
kde parametr c je posunutí pro linearizovaný model.

Dále je nutné brát v úvahu nelinearitu modelu, která je patrná na statické charakteristice. Lineární aproximací modelu ve zvoleném pracovním bodě dojde k určité odchylce od reálného modelu, proto si rozdělíme statickou charakteristiku přibližně na jednotlivé lineární úseky a provedeme identifikaci pro každý tento úsek zvlášť.

Posledním parametrem, který je před identifikací potřeba určit, je perioda vzorkování. Model Amira DR300 může komunikovat s nejnižší periodou $T_0 = 0.01$. Z pohledu kvality řízení je ideální volit periodu co nejnižší, avšak musíme brát v úvahu i výpočetní náročnost, která především pro GPC s omezením může být relativně vysoká. Při řešení podobného problému [12], byla použita perioda vzorkování $T_0 = 0.05$. Při zkoušení prediktivního řízení modelu se tato perioda jeví jako optimální.

V prvním případě se budeme snažit pokrýt co největší rozsah a budeme volit vstupní signál jako náhodný signál v rozsahu od 1.3V do 1.8V. Výsledný přenos bude využit jako simulační model a později i k výpočtu prediktivního řízení.

Ve všech dalších grafech a výpočtech pracujeme s jednotkami Matlabu, které je potřeba vynásobit převodní konstantou, abychom získali skutečnou hodnotu veličinami na servomechanismu. Převodní konstanta pro výstup modelu mezi jednotkou Matlabu a otáčkami motoru je 4000 Rpm/MU. Pro vstup modelu je konstanta mezi jednotkami Matlabu a vstupním napětím 10 V/MU. Následující graf tedy zobrazuje hodnotu vstupu 0.13 až 0.18, což představuje hodnotu vstupního napětí na servomechanismu 1.3V až 1.8V.



Obr. 8. Odezva modelu na náhodný signál

Pro výpočet bylo zvoleno několik modifikací MNČ. Konkrétně jednorázová MNČ pro první a druhý řád, dále rekurzivní MNČ pro druhý řád a nakonec modifikovaná jednorázová MNČ s posunutím linearizovaného modelu pro druhý řád. Jednotlivé metody jsou srovnány podle kvadratické regulační odchylky od reálných naměřených dat v tabulce (Tab. 1).

Tab. 1. Srovnání kvadratických regulačních odchylek pro jednotlivé MNČ

Metoda	S_y
Jednorázová MNČ pro model 1. řádu	$1.8300 \cdot 10^{-3}$
Jednorázová MNČ pro model 2. řádu	$1.8010 \cdot 10^{-3}$
Rekurzivní MNČ pro model 2. řádu	$1.8010 \cdot 10^{-3}$
Modifikovaná jednorázová MNČ pro model 2. řádu	$8.5120 \cdot 10^{-4}$

Z výsledků je patrné, že nejpřesnější vyjádření nám dává právě modifikovaná metoda. Výsledek modifikované MNČ může být zapsán jako:

$$y(k) = 0.9771y(k-1) - 0.0284y(k-2) + 0.0681u(k-1) + 0.22u(k-2) - 0.0316. \quad (7.6)$$

Zápis může být upraven do tvaru:

$$Y(z^{-1}) = \frac{0.0681z^{-1} + 0.22z^{-2}}{1 - 0.9771z^{-1} + 0.0284z^{-2}} U(z^{-1}) - \frac{0.0316}{1 - 0.9771z^{-1} + 0.0284z^{-2}}. \quad (7.7)$$

Tento vztah lze uvažovat i ve tvaru přenosové funkce

$$G_{0.155}(z^{-1}) = \frac{0.0681z^{-1} + 0.22z^{-2}}{1 - 0.9771z^{-1} + 0.0284z^{-2}} \quad (7.8)$$

za předpokladu, že hodnota vstupu je snížena $\tilde{u}(k) = u(k) - 0.1097$, tak aby byla výsledná hodnota přenosu posunuta.

Přenos druhého řádu určený klasickou jednorázovou metodou nejmenších čtverců má hodnotu:

$$G(z^{-1}) = \frac{0.0106z^{-1} + 0.1626z^{-2}}{1 - 0.9710z^{-1} + 0.0239z^{-2}}. \quad (7.9)$$

Jako simulační model bude využit přesnější model (7.8), avšak pro výpočet prediktorů bude použit model (7.9) získaný jednorázovou MNČ a to z důvodu jednoduššího odvozování a výpočtů jednotlivých prediktorů. Dále budou vypočteny i další přenosy pro limitní a záporné oblasti statické charakteristiky.

Přenos pro oblast vstupních hodnot 0.11 až 0.13 má tvar:

$$G_{0.12}(z^{-1}) = \frac{0.0728z^{-1} + 0.2321z^{-2}}{1 - 0.9484z^{-1} + 0.0227z^{-2}} \quad (7.10)$$

když $\tilde{u}(k) = u(k) - 0.1081$.

Přenos pro oblast vstupních hodnot 0.165 až 0.185 má tvar:

$$G_{0.175}(z^{-1}) = \frac{0.0628z^{-1} + 0.2305z^{-2}}{1 - 0.8787z^{-1} + 0.0948z^{-2}} \quad (7.11)$$

když $\tilde{u}(k) = u(k) - 0.0891$.

Přenos pro oblast vstupních hodnot -0.13 až -0.18 má tvar:

$$G_{-0.155}(z^{-1}) = \frac{0.0688z^{-1} + 0.2142z^{-2}}{1 - 1.014z^{-1} + 0.0399z^{-2}} \quad (7.12)$$

když $\tilde{u}(k) = u(k) + 0.1113$.

8 SIMULACE PREDIKTIVNÍHO ŘÍZENÍ

V této kapitole budeme testovat jednotlivé druhy prediktivního řízení, popsané v teoretické části. Zaměříme se především na zkoumání kvality regulace. U jednotlivých simulací budeme také měřit čas, což nám přiblíží a umožní porovnat výpočetní náročnost jednotlivých prediktorů a jejich nastavení. Dále se zaměříme na robustnost těchto regulátorů a to především z pohledu kvality regulace, kdy budeme regulátor považovat za robustní, pokud dokáže dané modely uřídit s nulovou regulační odchylkou. Zanedbáme robustní stabilitu i analytické vyjádření robustnosti, jelikož robustnost u MPC obecně není kompletně dořešená a její analytické vyjádření je stále otevřený problém. [20]

8.1 GPC inverzí matic

8.1.1 Nastavení parametrů

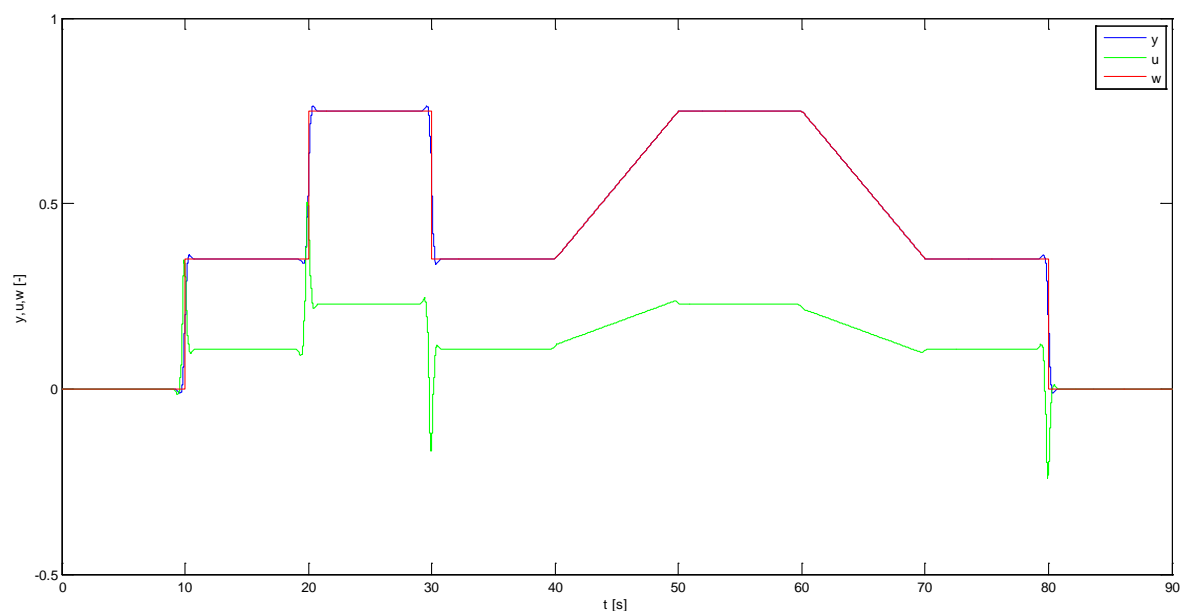
U základní varianty GPC bez omezení máme možnost nastavování minimálního, maximálního, řídicího horizontu a váhového parametru λ . Minimální horizont $N_1 = 1$, řídicí horizont $N_u = N_2$ vždy pokud to dovolí výpočetní výkon. Základní nastavení prediktoru budeme testovat na modelu (7.9).

Tab. 2. Nastavení horizontu predikce pro GPC bez omezení

N_2	N_u	λ	t [s]	S_y	S_u	y_{\max}	y_{\min}
5	5	1	0.4823	$1,6767 \cdot 10^{-4}$	$1,5209 \cdot 10^{-4}$	0,7848	-0,0304
10	10	1	0.4917	$1,8732 \cdot 10^{-4}$	$7,3677 \cdot 10^{-5}$	0,7631	-0,0114
20	20	1	0.5197	$1,8914 \cdot 10^{-4}$	$6,8899 \cdot 10^{-5}$	0,7629	-0,0113
30	30	1	0.5238	$1,8920 \cdot 10^{-4}$	$6,8835 \cdot 10^{-5}$	0,7629	-0,0113
40	40	1	0.5239	$1,8920 \cdot 10^{-4}$	$6,8835 \cdot 10^{-5}$	0,7629	-0,0113

Při hledání optimálního horizontu predikce byl pevně nastaven váhový parametr $\lambda = 1$. S rostoucí hodnotou řídicího horizontu se zvyšuje kvadratická regulační odchylka a naopak klesá kvadratický rozdíl změn akčního zásahu. Taktéž se snižuje hodnota překmitu. Časová

náročnost stoupá jen velmi pozvolna. Od $N_u = N_2 = 20$ se hodnoty kritérií téměř nemění, Jako optimální hodnotu řídicího horizontu jsem zvolil $N_u = N_2 = 20$.



Obr. 9. Regulační průběh GPC bez omezení pro $N_u=N_2=20$

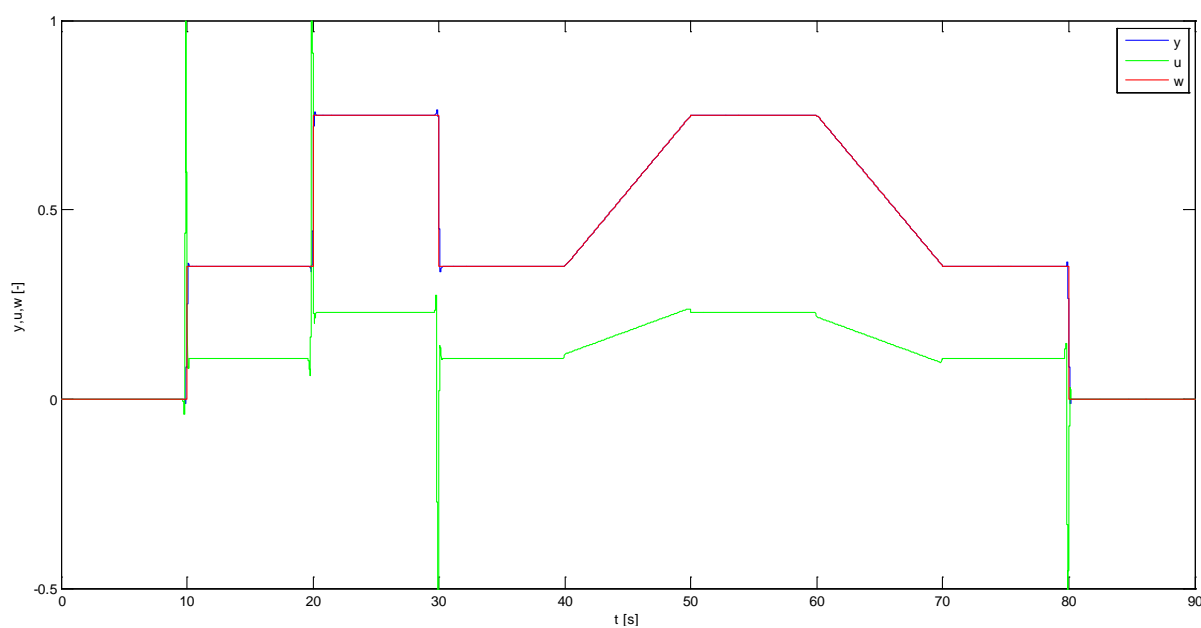
Dále se pokusíme najít optimální hodnotu váhového parametru λ , který udává poměr mezi minimalizací kvadratické regulační odchylky a kvadratické změny akčního zásahu. Hodnota řídicího horizontu bude pevně nastavena na $N_u = N_2 = 20$.

Tab. 3. Nastavení váhového parametru pro GPC bez omezení

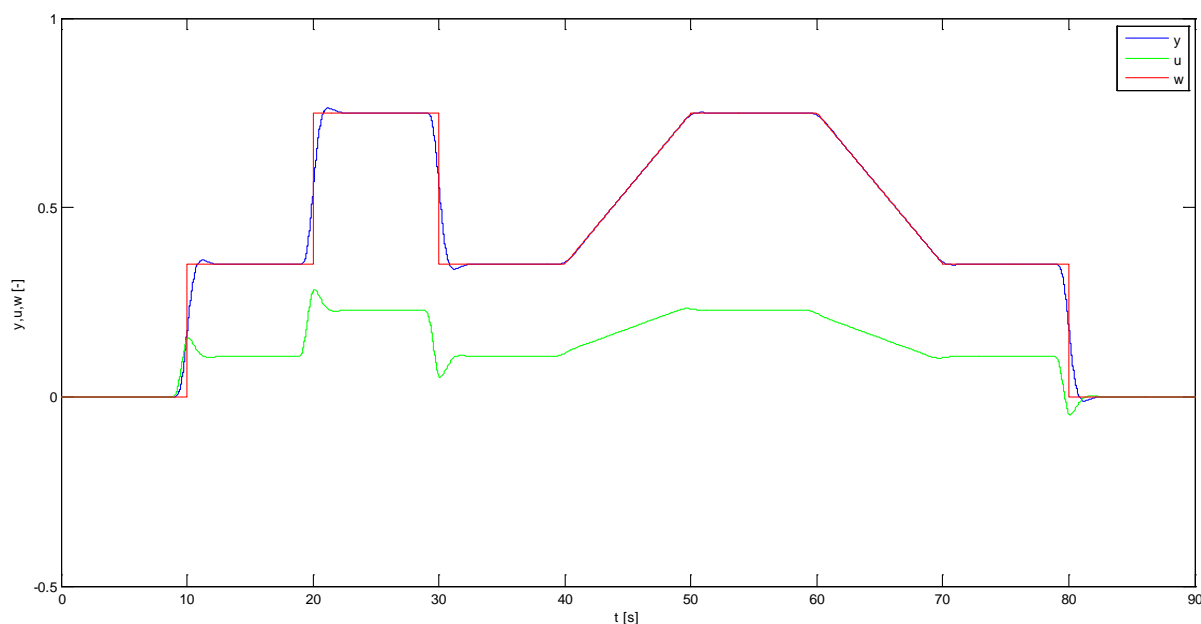
N_2	N_u	λ	t [s]	S_y	S_u	y_{\max}	y_{\min}
20	20	0.01	0.4899	$4,7769 \cdot 10^{-5}$	$2,0768 \cdot 10^{-3}$	0,7625	-0,0109
20	20	0.05	0.5259	$7,7283 \cdot 10^{-5}$	$6,8617 \cdot 10^{-4}$	0,7630	-0,0114
20	20	0.1	0.4804	$9,7507 \cdot 10^{-5}$	$3,9642 \cdot 10^{-4}$	0,7630	-0,0114
20	20	0.5	0.4865	$1,5653 \cdot 10^{-4}$	$1,1554 \cdot 10^{-4}$	0,7631	-0,0114
20	20	1	0.5187	$1,8914 \cdot 10^{-4}$	$6,8899 \cdot 10^{-5}$	0,7629	-0,0113
20	20	10	0,4917	$3,4305 \cdot 10^{-4}$	$1,3815 \cdot 10^{-5}$	0,7625	-0,0109
20	20	50	0.5211	$5,2332 \cdot 10^{-4}$	$4,7417 \cdot 10^{-6}$	0,7614	-0,0100

20	20	100	0.5102	$5,9566 \cdot 10^{-4}$	$3,6062 \cdot 10^{-6}$	0,7627	-0,0112
----	----	-----	--------	------------------------	------------------------	--------	---------

Dle výsledků simulace (Tab. 3) je patrné, že zde není jedna ideální hodnota váhového parametru. Záleží, zda budeme preferovat minimální kvadratickou regulační odchylku nebo kvadratickou změnu akčního zásahu. Minimální hodnotu S_y dosahuje prediktor pro $\lambda = 0.01$. Naopak minimální hodnota S_u nastává při hodnotě $\lambda = 100$.



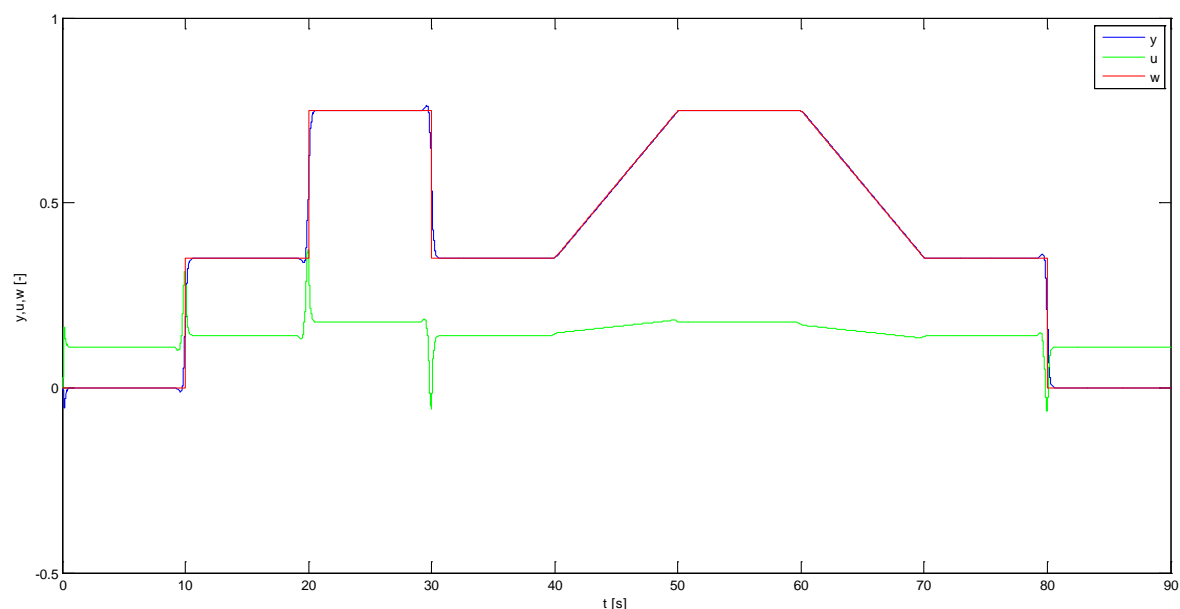
Obr. 10. Regulační průběh GPC bez omezení pro $\lambda=0.01$



Obr. 11. Regulační průběh GPC bez omezení pro $\lambda=100$

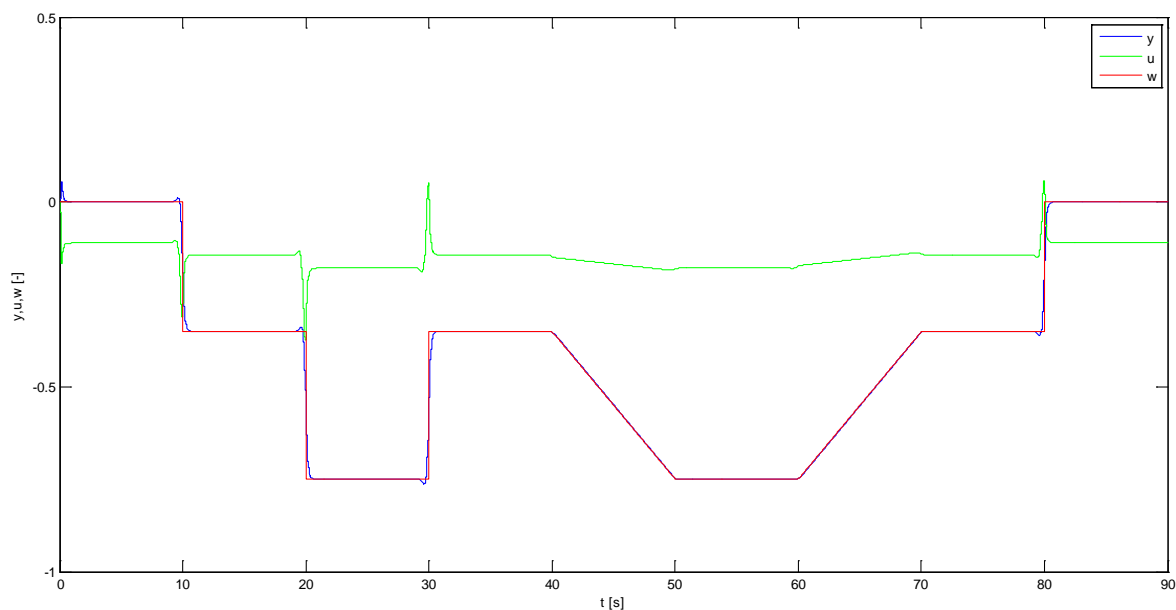
8.1.2 Robustnost řízení

Dále se zaměříme na chování prediktoru v případě, že reálný model neodpovídá přesně našemu odhadu. Jako simulační model použijeme zpřesněný model popsany rovnicí (7.8) a nastavení prediktoru je $N_u = N_2 = 20$ a $\lambda = 1$.



Obr. 12. Regulační průběh GPC bez omezení pro model (7.8)

Regulační průběh je téměř totožný s průběhem (Obr. 9.) a má jen nepatrně horší hodnoty kvality regulace. Model popsany (7.8) je jen určitým zpřesněním modelu pro výpočet prediktoru a oba dávají podobný průběh. Dále budou vyzkoušeny více rozdílné simulační modely.

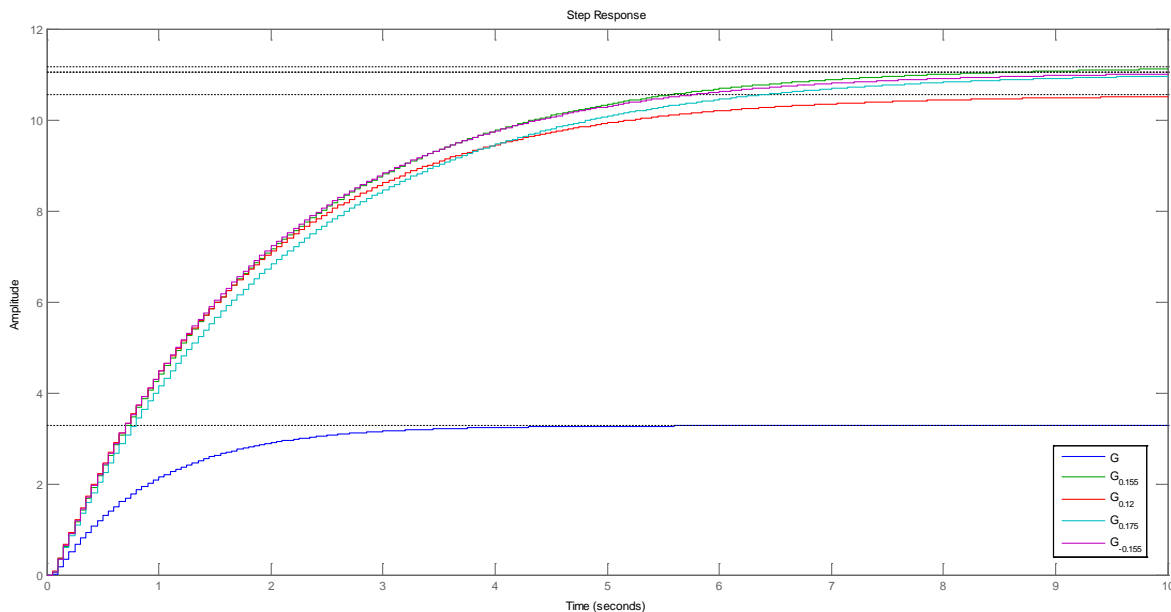


Obr. 13 Regulační průběh GPC bez omezení pro model (6.12)

Tab. 4. Kvalita regulace GPC bez omezení pro různé simulační modely

Model	N_2	N_u	λ	t [s]	S_y	S_u	y_{\max}	y_{\min}
G	20	20	1	0,5187	$1,8914 \cdot 10^{-4}$	$6,8899 \cdot 10^{-5}$	0,7629	-0,0113
$G_{0.155}$	20	20	1	0,5328	$1,9728 \cdot 10^{-4}$	$3,6646 \cdot 10^{-5}$	0,7631	-0,0530
$G_{0.12}$	20	20	1	0,5493	$1,9962 \cdot 10^{-4}$	$3,4438 \cdot 10^{-5}$	0,7629	-0,0532
$G_{0.175}$	20	20	1	0,5006	$1,9750 \cdot 10^{-4}$	$3,6599 \cdot 10^{-5}$	0,7630	-0,0416
$G_{-0.155}$	20	20	1	0,4875	$1,9679 \cdot 10^{-4}$	$3,6651 \cdot 10^{-5}$	0,0542	-0,7631

Z naměřených dat je patrné, že kvalita regulace je podobná pro všechny modely. Pro nejčistší verzi, kdy výpočetní model je zároveň modelem regulovaným, získáme nepatrně lepší kvadratickou regulační odchylku, avšak téměř dvojnásobně větší kvadratickou změnu akčního zásahu. Pro příčinu těchto výsledků je potřeba se podívat na přechodové charakteristiky jednotlivých modelů.



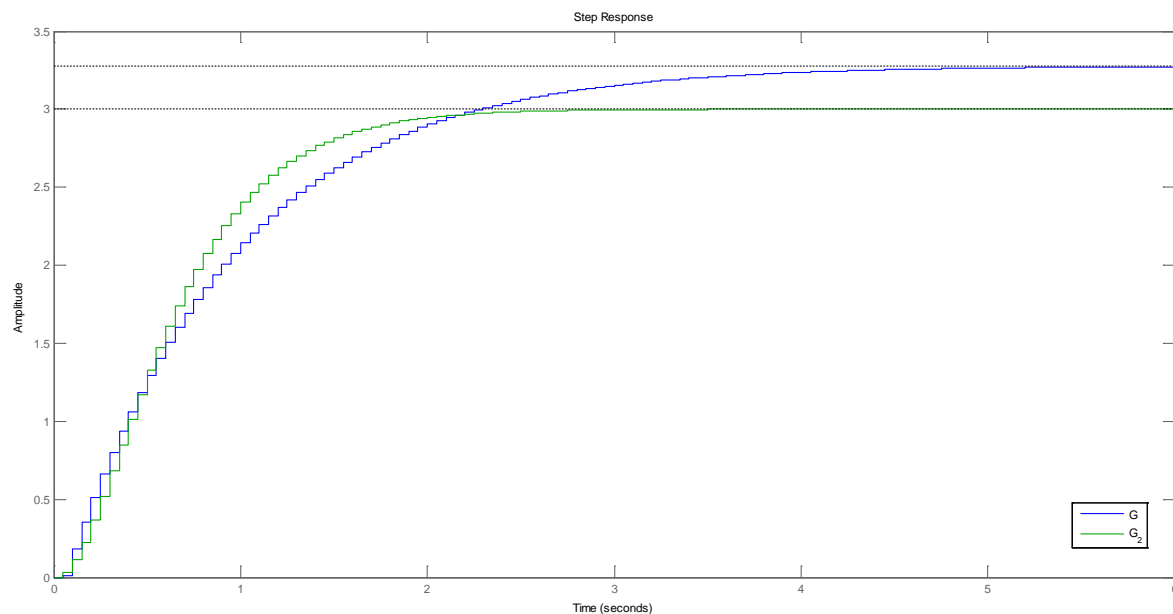
Obr. 14. Přechodová charakteristika pro jednotlivé modely

Z charakteristiky (Obr. 14) je patrné, že modely vypočtené zpřesněnou MNČ mají výrazně větší zesílení. Pro oblast vstupních hodnot měřených MNČ od 0.11 do 0.185 mají tyto přenosy i vlivem posunutí c , téměř totožné zesílení s přenosem G , ze kterého je vypočten prediktor. Pokud je však vstupní hodnota vyšší, je patrný rozdíl v zesílení. Při maximální hodnotě ± 1 ($\pm 10V$ na servomechanismu), což je maximální hodnota vstupního napětí na servomechanismu, je rozdíl v zesíleních výrazný. Současně se nějak výrazně nemění dynamické vlastnosti modelu. Prediktor je tedy schopný pro tyto modely dosáhnout žádané hodnoty za pomoci menšího akčního zásahu, i když model pro jeho výpočet je rozdílný.

Je patrné, že GPC bez omezení si dokáže vcelku dobře poradit s modely s rozdílným zesílením, avšak při zachování podobnosti dynamických vlastností. Dále je potřeba otestovat systém s rozdílným dynamickým chováním od modelu prediktoru. Takovým případem může být model popsáný přenosem:

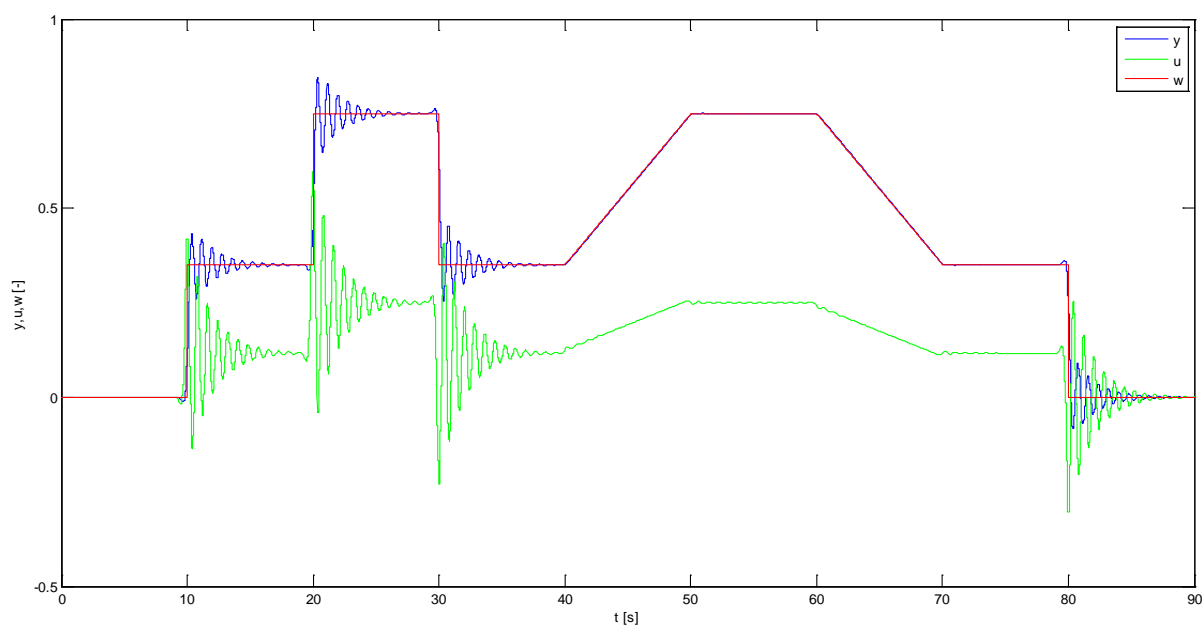
$$G_2(z^{-1}) = \frac{0.0306z^{-1} + 0.0277z^{-2}}{1 - 1.721z^{-1} + 0.7408z^{-2}}. \quad (8.1)$$

Porovnání tvaru přechodové charakteristiky s nominálním modelem prediktoru je v grafu (Obr. 15).



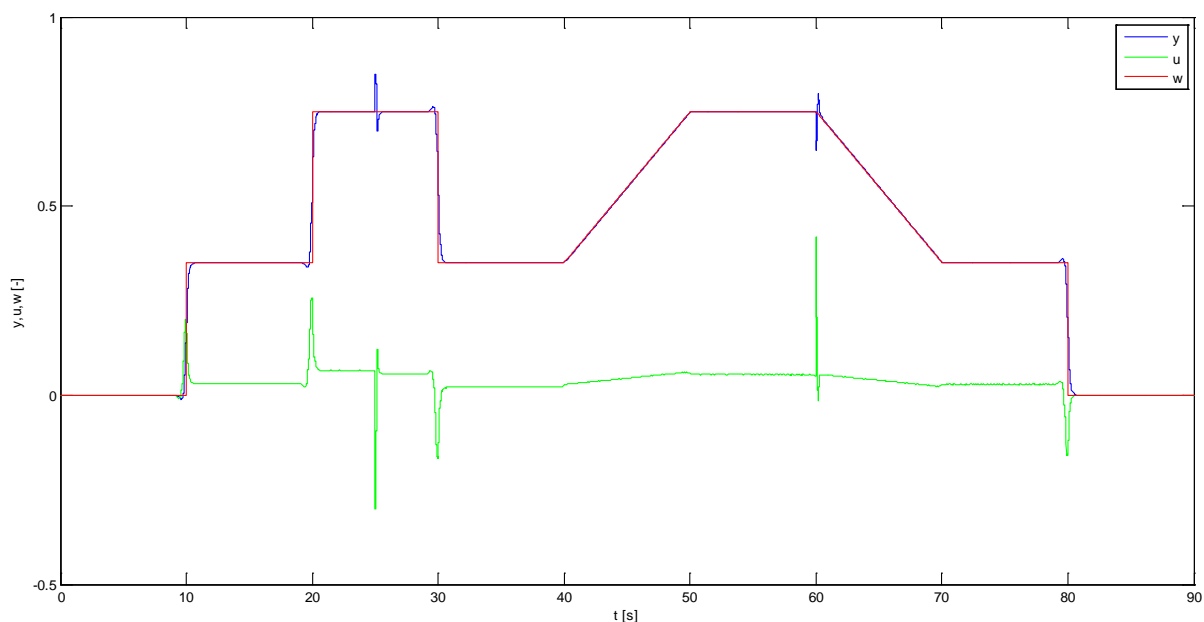
Obr. 15. Přechodová charakteristika různých modelů

Z regulačního průběhu (Obr. 16) je patrné, že při skokové změně dojde k rozkmitání systému. Prediktor dokáže postupně toto rozkmitání utlumit a dosáhnout nulové regulační odchylky. Pokud by byl charakteristický polynom, potažmo dynamika přenosu simulovaného systému více rozdílná oproti nominálnímu přenosu pro výpočet prediktoru, prediktor již pravděpodobně nebude schopen dosáhnout nulové odchylky. Je vcelku zřejmé, že se změnou v charakteristickém polynomu se prediktor vypořádává podstatně hůře, než s pouhou změnou zesílení.



Obr. 16. Regulační průběh GPC bez omezení pro model (8.1)

Dále zkusíme otestovat chování GPC bez omezení na modelu, na který působí různé poruchy, které se mohou vyskytovat i na reálném modelu. Bude zde působit náhodná porucha, ve formě bílého šumu s maximální hodnotou $\pm 1 \cdot 10^{-4}$, v určitém okamžiku dojde navíc ke skokové poruše o hodnotě 0.1. Také bude postupně růst zesílení systému a to o hodnotu 0.1% z celkového zesílení každou periodu. Jako simulační model použijeme přenos popsáný rovnicí (7.8).



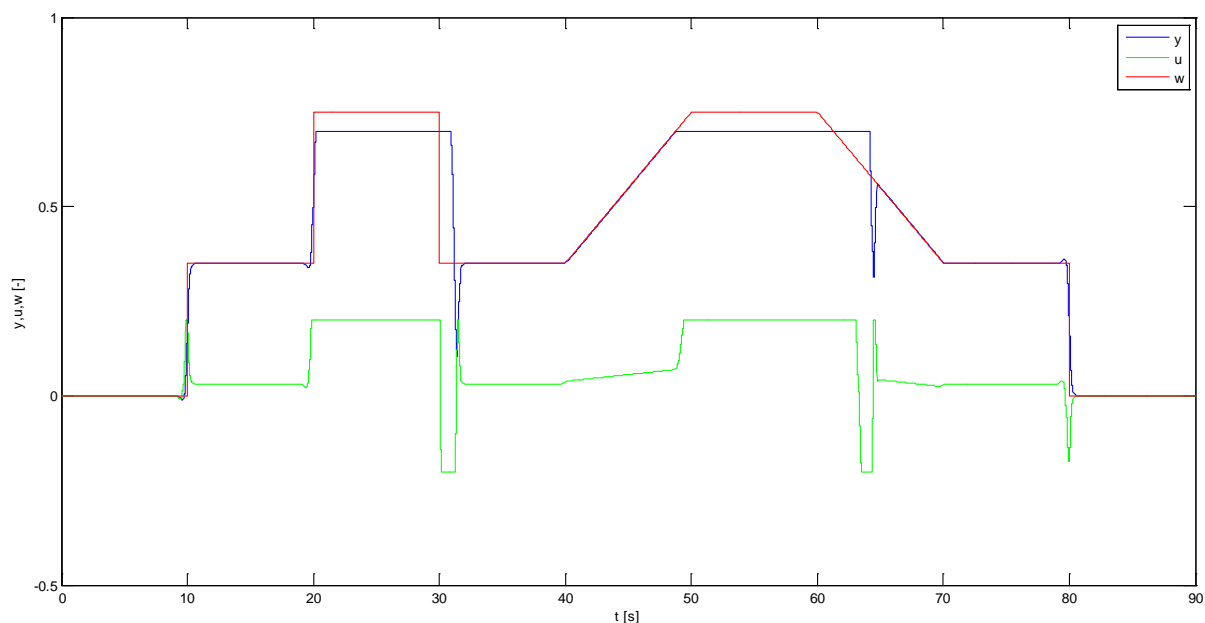
Obr. 17. Regulační průběh GPC bez omezení pro model s vnějšími poruchami

Z průběhu lze vyčíst, že GPC bez omezení se dokáže vypořádat s výše zmíněnými poruchami, pokud ty nepřesahují určitou mez.

8.1.3 Omezení veličin

GPC bez omezení se doposud jeví jako vhodné prediktivní řízení. Jeho hlavní nevýhodou je nemožnost zadat omezení přímo do výpočtu. Omezení musíme dosadit pomocí saturace a to lze pouze na výstupní veličině regulátoru. Výsledné řízení nemusí být optimální dle zvolených kritérií. Navíc může dojít i k tzv. wind-up efektu.

Zvolíme tedy následující omezení, které budeme realizovat pomocí saturace. Výstup z prediktoru omezíme na ± 0.2 ($\pm 2V$ vstupní napětí na servomotoru), výstup z řízeného systému omezíme na ± 0.7 (± 2800 ot/min).



Obr. 18. Regulační průběh GPC bez omezení se saturací veličin

Na regulačním průběhu je patrný zmíněný wind-up efekt, kdy prediktor nemá informaci o omezení a tak stále zvyšuje akční zásah s cílem dosáhnout žádané hodnoty, kterou nemůže dosáhnout právě kvůli omezení. Následně při snížení žádané hodnoty není schopen dostatečně rychle snížit vysoký akční zásah.

GPC bez omezení není tedy nejvhodnější volbou prediktivního řízení, pokud je potřeba omezovat jednotlivé veličiny, což je obecně jednou z hlavních předností MPC.

8.2 GPC kvadratickým programováním

U této varianty GPC nastavujeme totožně maximální, minimální, řídicí horizont a váhový parametr λ . Navíc zde můžeme dosadit omezující podmínky akčního zásahu, výstupu systému atd.

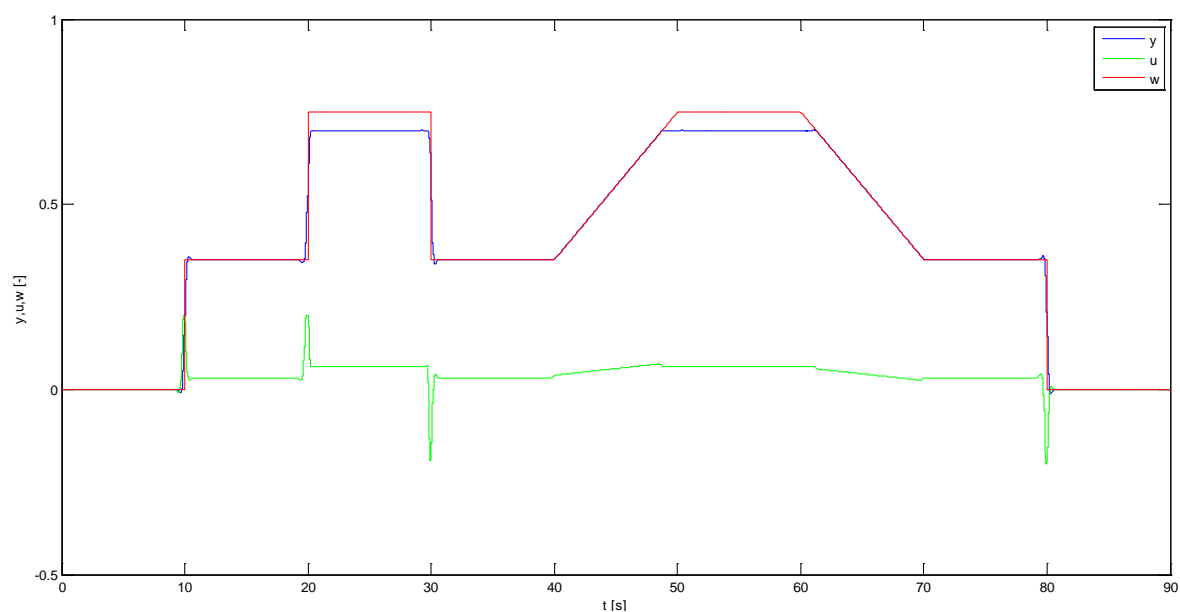
Tab. 5. Srovnání kvality regulace pro jednotlivé verze GPC

Metoda	N_2	N_u	λ	t [s]	S_y	S_u	y_{\max}	y_{\min}
Inv. mat.	10	10	1	0.4917	$1,8732 \cdot 10^{-4}$	$7,3677 \cdot 10^{-5}$	0,7631	-0,0114
Inv. mat.	20	20	1	0.5197	$1,8914 \cdot 10^{-4}$	$6,8899 \cdot 10^{-5}$	0,7629	-0,0113
QP.	10	10	1	8,5552	$1,8732 \cdot 10^{-4}$	$7,3677 \cdot 10^{-5}$	0,7631	-0,0114
QP	20	20	1	8,9471	$1,8914 \cdot 10^{-4}$	$6,8899 \cdot 10^{-5}$	0,7629	-0,0113

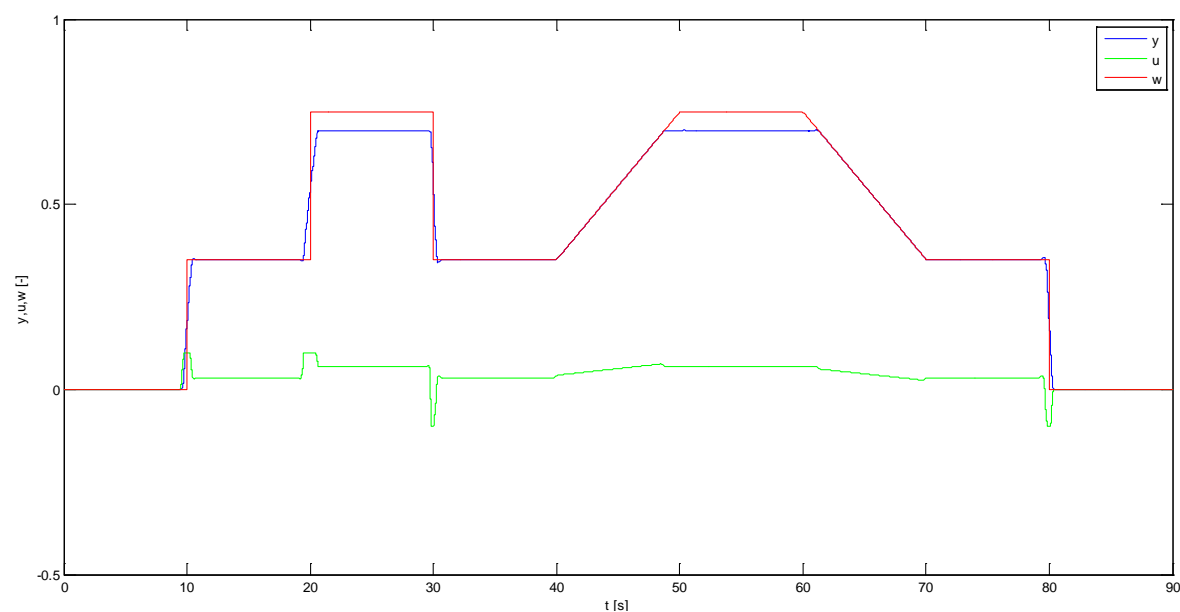
Metoda GPC vypočtená pomocí kvadratického programování dosahuje stejných regulačních průběhů jako GPC vypočtené inverzí matic v případě, že do výpočtu nevstupují omezení. Jediný rozdíl je ve výpočetní náročnosti. Proto se nadále budeme zabývat situací, kdy do výpočtu omezení vstupují.

8.2.1 Omezení veličin

Jako omezení zvolíme příklad použitý v předešlé kapitole. Výstup z prediktoru omezíme na ± 0.2 ($\pm 2V$ vstupní napětí na servomotoru), výstup z řízeného systému omezíme na ± 0.7 ($\pm 2800 \text{ ot/min}$) popř. dále i jiné varianty omezení.



Obr. 19. Regulační průběh GPC s omezením veličin - $0.2 < u < 0.2$ a $-0.7 < y < 0.7$



Obr. 20. Regulační průběh GPC s omezením veličin - $0.1 < u < 0.1$ a $0 < y < 0.7$

Tab. 6. Regulační hodnoty pro GPC s omezením akční veličiny

Průběh	N_2	N_u	λ	t [s]	S_y	S_u	y_{\max}	y_{\min}
(Obr. 19)	20	20	1	14,5597	$8,0583 \cdot 10^{-4}$	$3,8015 \cdot 10^{-5}$	0,7	-0,0112
(Obr. 20)	20	20	1	17,7594	$9,9617 \cdot 10^{-4}$	$1,1263 \cdot 10^{-5}$	0,7	0,0000

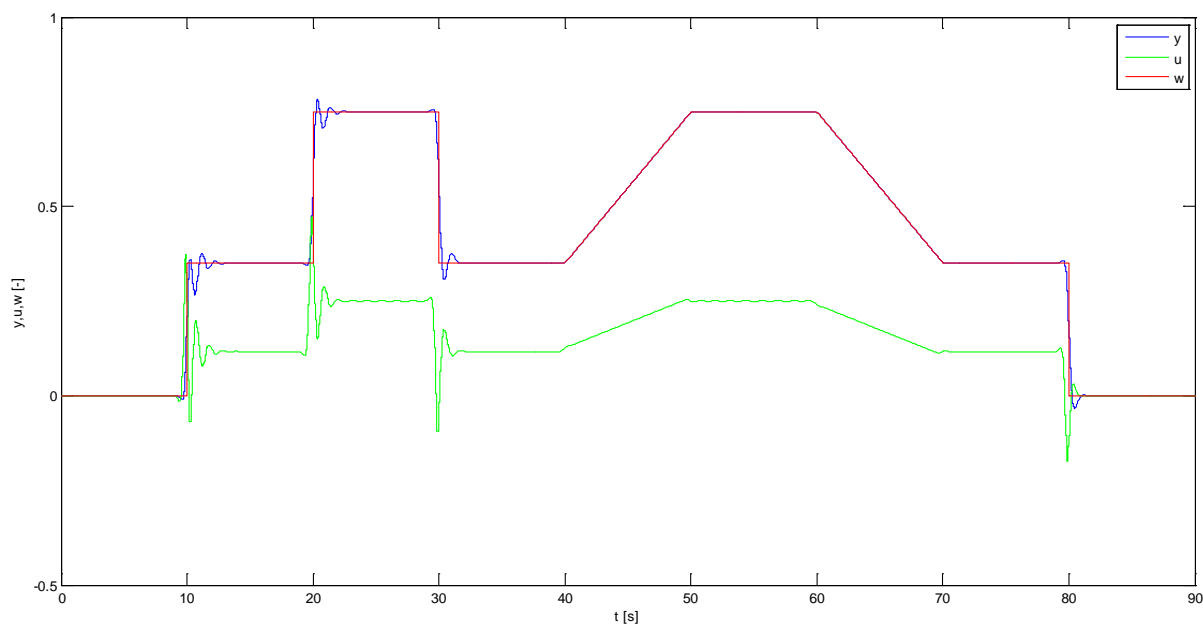
GPC pracuje s omezeními veličin dobře. Z dat je však patrné, že společně s omezeními narůstá i výpočetní výkon celé metody. Za nevýhodu GPC regulátorů můžeme považovat vázanost na model systému, se kterým počítají. Pokud se model pro výpočet GPC a skutečný řízený systém od sebe výrazně liší, GPC regulátor není schopen obvod efektivně regulovat. Tuto nevýhodu jsme schopni eliminovat použitím průběžné identifikace, kdy modifikujeme GPC na samočinně se nastavující regulátor.

8.3 Adaptivní GPC

Adaptivní GPC je samočinně se nastavující prediktivní regulátor. Takto upravený regulátor nelze formálně považovat za robustní, avšak pro naše potřeby, kdy chceme dosáhnout nulové regulační odchylky v určitém čase pro danou skupinu systémů, můžeme tuto modifikaci využít. Adaptivní GPC by měl být schopen efektivně regulovat libovolný systém stejného řádu. Pro identifikaci je využita rekursivní metoda nejmenších čtverců s adaptivním směrovým zapomínáním, která je lepší pro systémy s proměnnými parametry než klasická rekursivní MNČ. [15]

8.3.1 Kvalita regulace

U tohoto regulátoru se tedy zaměříme především na zkoumání odlišných modelů řízených systémů. Jako první otestujeme model (8.1), kde při použití klasické metody GPC dochází k určitému rozkmitání výstupní veličiny při řízení (Obr. 16).



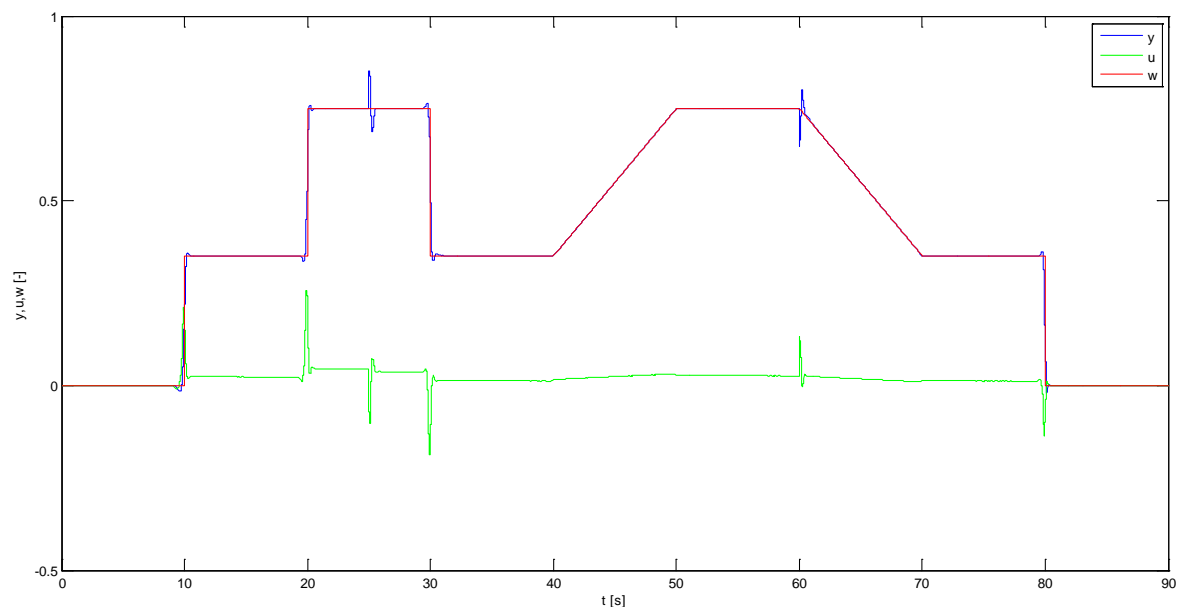
Obr. 21 Regulační průběh adaptivního GPC pro model (8.1)

Tab. 7. Porovnání kvality regulace GPC a adaptivního GPC

Metoda	N_2	N_u	λ	t [s]	S_y	S_u	y_{\max}	y_{\min}
GPC bez omez.	20	20	1	0,5867	$5,3675 \cdot 10^{-4}$	$4,2559 \cdot 10^{-4}$	0,8452	-0,0836
Adaptivní GPC	20	20	1	42,1869	$2,7122 \cdot 10^{-4}$	$9,3779 \cdot 10^{-5}$	0,7728	-0,0281

Oproti původnímu regulačnímu průběhu (Obr. 16) dosahuje adaptivní GPC lepších výsledů kvality regulace. Avšak oproti zjednodušené formě GPC má adaptivní GPC téměř 100 násobnou výpočetní náročnost.

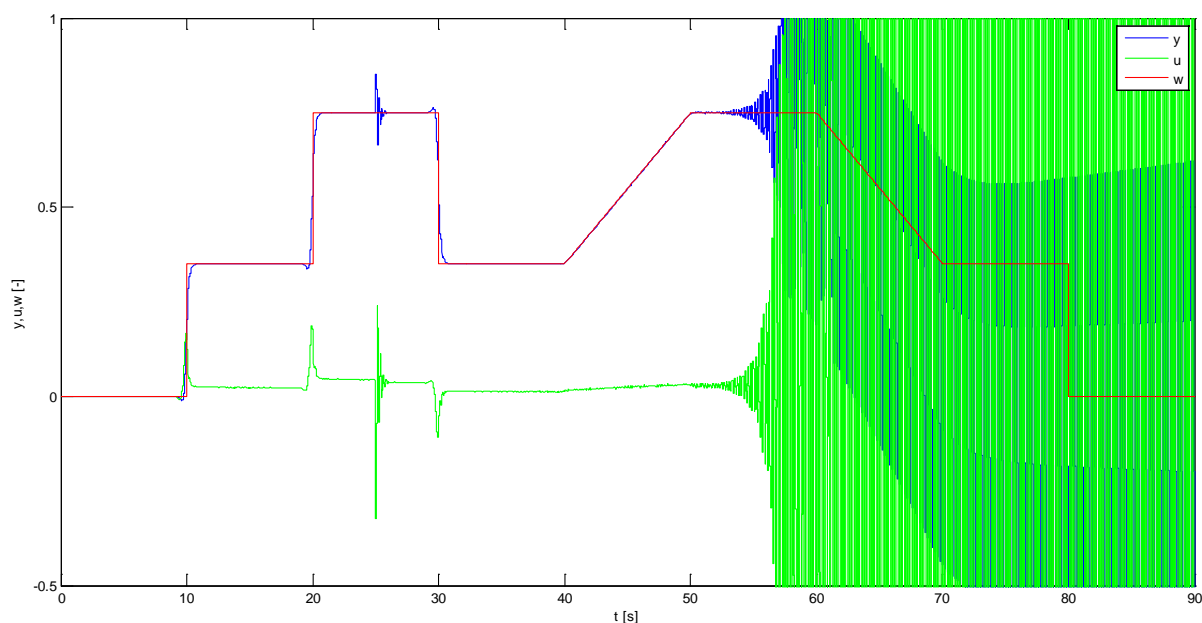
Do řídicího obvodu s adaptivním GPC vložíme různé poruchy, stejné jako při testování GPC. Tedy náhodná porucha ve formě bílého šumu, skoková porucha v určitém čase a růst zesílení modelu.



Obr. 22. Regulační průběh adaptivního GPC pro model (7.8) s poruchami

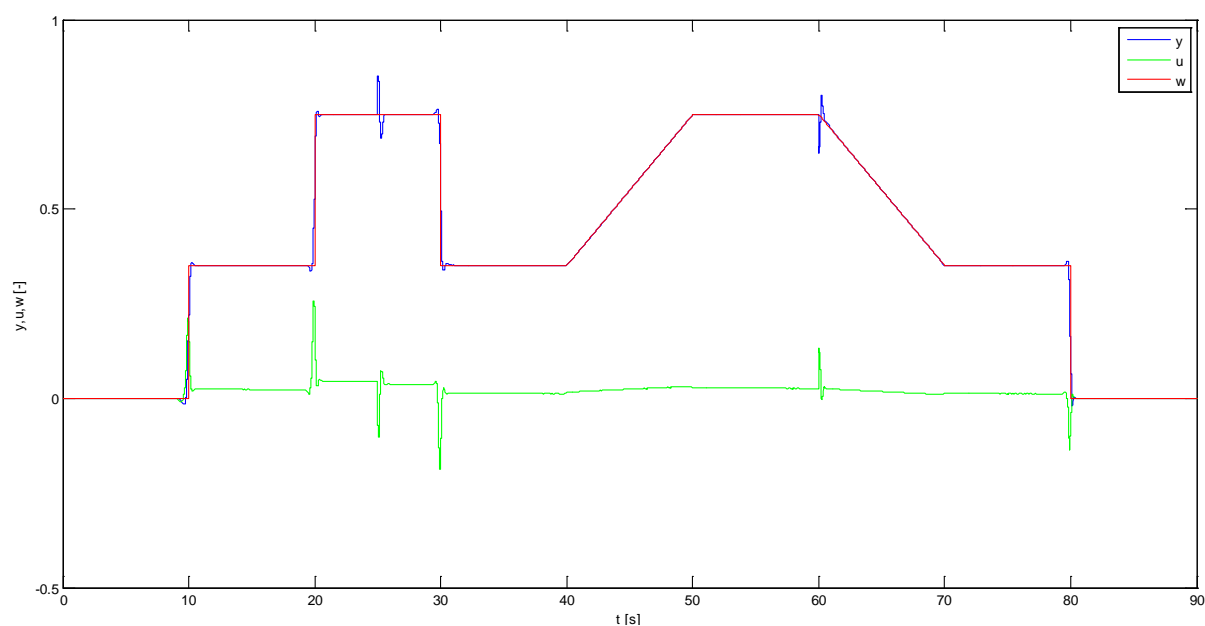
Adaptivní GPC regulátor dobře eliminuje skokové změny a změny v parametrech řízeného systému a to i velmi výrazné, se kterými si již klasické GPC nedokáže poradit. Náhodnou poruchu ve formě bílého šumu eliminuje adaptivní GPC o poznání hůře.

Zaměříme se na rostoucí zesílení řízeného modelu, které v minulém případě rostlo o hodnotu 0.1% z celkového zesílení každou periodu. Zvýšíme růst na 1% na periodu. A porovnáme průběh klasického GPC s adaptivním GPC.



Obr. 23. Regulační průběh klasického GPC pro model (7.8) s rostoucím zesílením systému

U klasického GPC (Obr. 23) dojde okolo času 50 sekund k nárůstu zesílení řízeného systému, při kterém se regulátor trvale rozkmitá. Oproti tomu adaptivní GPC regulátor dokáže tento proměnný systém regulovat bez větších problémů (Obr. 24), obdobně jako reguluje systém beze změny zesílení. V tomto případě tedy u adaptivního GPC nezáleží do jisté míry na velikosti poruchy, pokud má porucha charakter změny zesílení systému. V obou případech reguluje se stejnou kvalitou (Tab. 8).



Obr. 24. Regulační průběh adaptivní GPC pro model (7.8) s rostoucím zesílením systému

Tab. 8. Srovnání GPC a adaptivního GPC pro proměnný systém

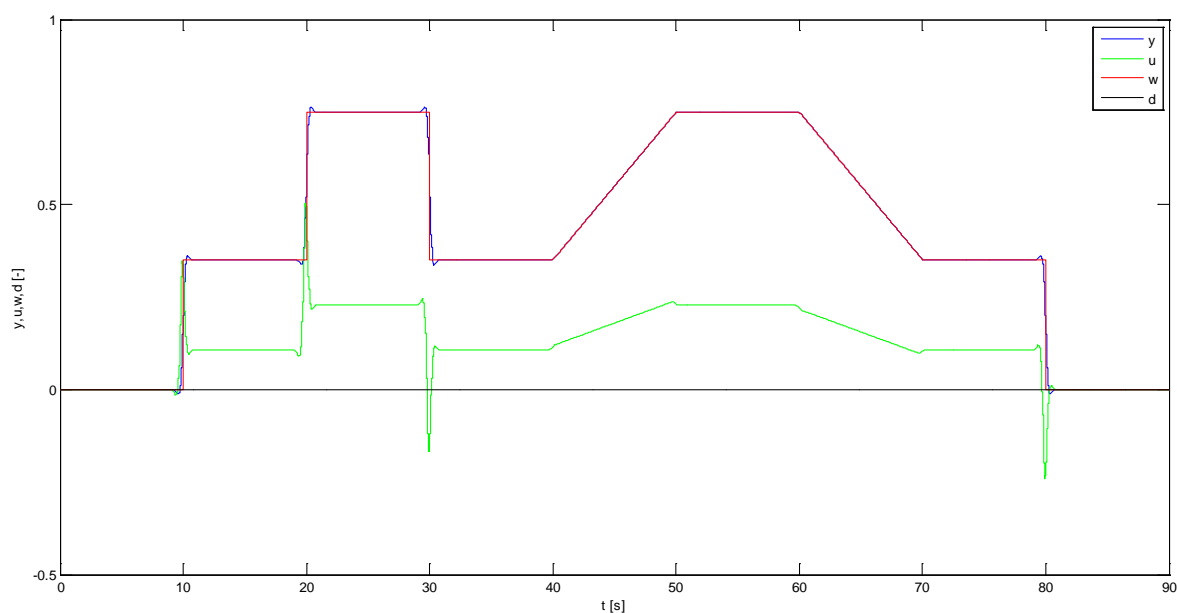
Metoda	N_2	N_u	λ	t [s]	S_y	S_u	y_{\max}	y_{\min}
GPC (Obr. 23)	20	20	1	8,0552	$7,5520 \cdot 10^{-2}$	$7,0269 \cdot 10^{-1}$	1,2349	-0,6219
A. GPC (Obr. 22)	20	20	1	39,9867	$1,9046 \cdot 10^{-4}$	$1,9046 \cdot 10^{-4}$	0,8504	-0,0202
A. GPC (Obr. 24)	20	20	1	40,6342	$1,9046 \cdot 10^{-4}$	$1,9046 \cdot 10^{-4}$	0,8504	-0,0202

8.4 Offset-free linear MPC

Jedná se o variantu lineární MPC, která je schopna potlačit poruchu mezi predikovaným a skutečným výstupem. Tato varianta by měla být schopná zajistit robustní kvalitu regulace pro množinu zvolených modelů.

8.4.1 Nastavení parametrů

U této verze MPC nastavujeme opět minimální horizont $N_1 = 1$, řídicí horizont $N_u = N_2$, pokud to dovolí výpočetní výkon a maximální horizont N_2 . Dále váhový parametru λ popř. omezení jednotlivých veličin. Oproti GPC zde navíc musíme nastavit délku vektoru poruchy. Defaultně budeme volit velikost vektoru stejnou jako velikost N_2 a výslednou poruchu d jako vážený průměr z těchto minulých hodnot poruchy. Jako model pro výpočet i pro regulovaný systém použijme přenos popsany rovnicí (7.9).

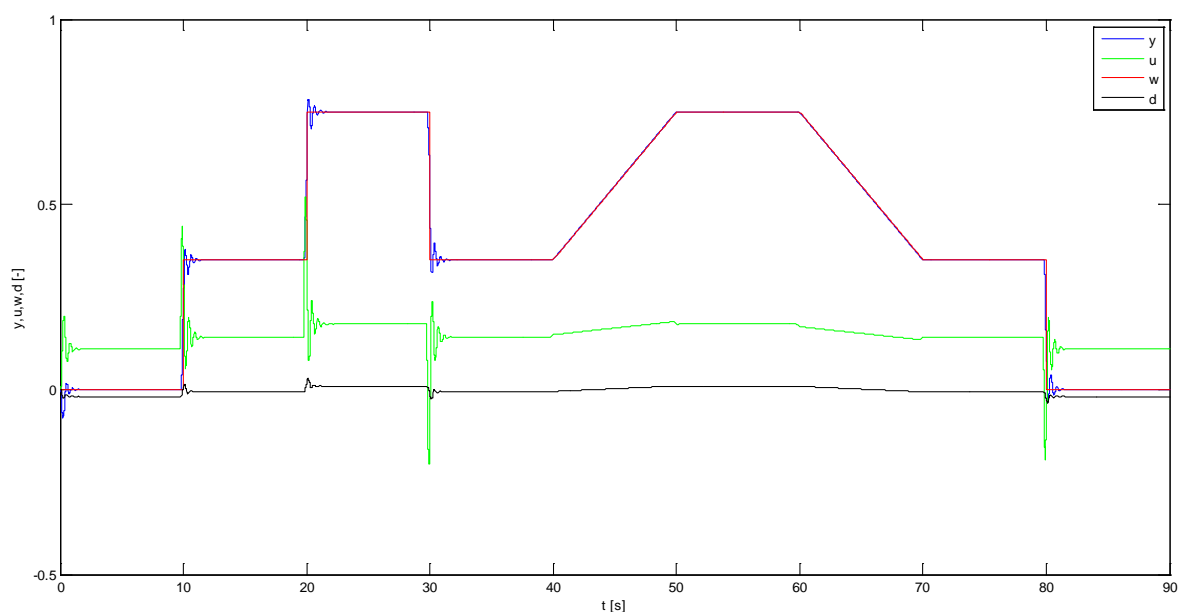


Obr. 25. Regulační průběh Offset-free MPC pro model (7.9)

Tab. 9. Nastavení horizontu predikce a váhového parametru pro Offset-free MPC

N_2	N_u	λ	t [s]	S_y	S_u	y_{\max}	y_{\min}
5	5	1	21,2943	$1,6767 \cdot 10^{-4}$	$1,5209 \cdot 10^{-4}$	0,7848	-0,0304
10	10	1	20,4118	$1,8732 \cdot 10^{-4}$	$7,3677 \cdot 10^{-5}$	0,7631	-0,0114
20	20	1	23,8298	$1,8914 \cdot 10^{-4}$	$6,8899 \cdot 10^{-5}$	0,7629	-0,0113
30	30	1	36,8574	$1,8920 \cdot 10^{-4}$	$6,8835 \cdot 10^{-5}$	0,7629	-0,0113
20	20	0,1	21,7458	$9,7507 \cdot 10^{-5}$	$3,9642 \cdot 10^{-4}$	0,7630	-0,0114
20	20	10	22,1312	$3,4305 \cdot 10^{-4}$	$1,3815 \cdot 10^{-5}$	0,7625	-0,0109

Z výsledků je patrné, že prediktor se pro stejný model chová totožně jako GPC, jelikož zde nevstupuje do hry porucha. Proto v dalším testu vyzkoušíme regulovat lehce rozdílný model popsany přenosem (7.8).

Obr. 26. Regulační průběh Offset-free MPC pro model (7.8) a $N_u=N_2=5$

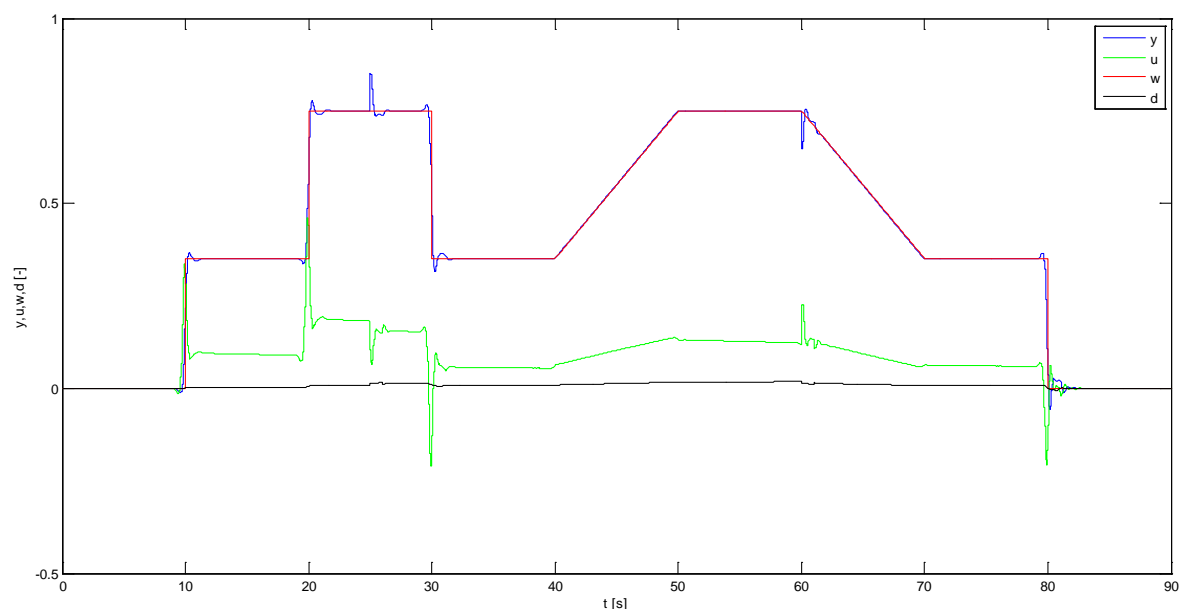
Tab. 10. Porovnání kvality regulace Offset-free MPC pro model (7.8)

N_2	N_u	λ	t [s]	S_y	S_u	y_{\max}	y_{\min}
5	5	1	21,2943	$1,6508 \cdot 10^{-4}$	$1,9150 \cdot 10^{-4}$	0,7829	-0,0756
20	20	1	22,1180	$2,2790 \cdot 10^{-4}$	$3,6646 \cdot 10^{-5}$	0,7631	-0,0530

Pro lehce rozdílný model (7.8) vykazuje Offset-free MPC lepších hodnotu kvadratické regulační odchylky než pro nominální model.

8.4.2 Robustnost

Dále podrobíme Offset-free MPC regulaci obvodu s poruchami. Konkrétně zde bude působit náhodná porucha ve formě bílého šumu s maximální hodnotou $\pm 1 \cdot 10^{-4}$. Skoková porucha o hodnotě 0.1 od času 25 do 60 sekund. Zesílení řízeného modelu bude růst o hodnotu 1% z celkového zesílení každou periodu. Jako simulační model využijeme opět systém popsany rovnicí (7.8).



Obr. 27. Regulační průběh Offset-free MPC pro model (7.8) s poruchou a $N_2=N_u=20$

Tab. 11. Porovnání kvality regulace Offset-free MPC pro model (7.8) s poruchami

N_2	N_u	λ	t [s]	S_y	S_u	y_{\max}	y_{\min}
5	5	1	22,0682	$2,0153 \cdot 10^{-4}$	$2,4586 \cdot 10^{-4}$	0,8505	-0,0468
20	20	1	26,9998	$2,5446 \cdot 10^{-4}$	$7,8662 \cdot 10^{-5}$	0,8508	-0,0573

U volby velikosti vektoru poruchy vykazoval prediktor lepších výsledků, čím byl jeho rozměr kratší. Nejlepší výsledky měl vektor, u kterého jsme brali v potaz pouze poruchu v minulé periodě. Takto krátký vektor však bude pravděpodobně při reálném použití velmi náchylný a nebude jej možné využít. Jako schůdná varianta se jeví upravení vektoru minulých poruch např. pomocí exponenciálního zapomínání.

9 MĚŘENÍ NA REÁLNÉM ZAŘÍZENÍ

Na servomechanismu Amira DR300 si ověříme určité závěry, které jsme získali simulací jednotlivých druhů prediktorů, zda budou platit i pro reálné měření. Na rozdíl od simulace zde budeme pracovat již s jediným modelem, a to tím reálným, čímž otestujeme chování prediktorů na nelineární model, na což jsme se v simulaci více nezaměřovali.

9.1 GPC

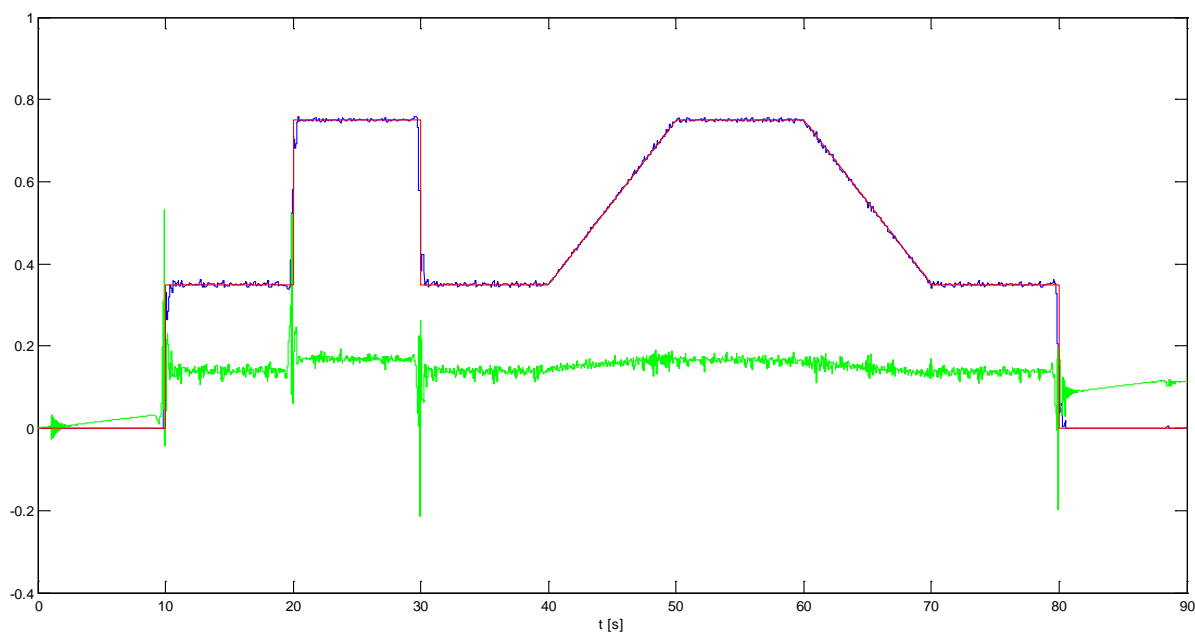
9.1.1 Bez omezení veličin

Pro GPC bez omezení nejprve zjistíme, zda platí ideální parametry, které jsme zvolili jako optimální v simulaci.

Tab. 12. Optimální parametry GPC pro reálné měření

N_2	N_u	λ	S_y	S_u	y_{\max}	y_{\min}
20	20	1	$3.1230 \cdot 10^{-4}$	$2.1579 \cdot 10^{-3}$	0.7734	-0.0068
30	30	1	$3.0852 \cdot 10^{-4}$	$1.4470 \cdot 10^{-3}$	0.7954	-0.0005
40	40	1	$2.5556 \cdot 10^{-4}$	$6.6888 \cdot 10^{-4}$	0.7666	-0.0005
50	50	1	$2.6952 \cdot 10^{-4}$	$1.3779 \cdot 10^{-3}$	0.7632	-0.0005
40	40	0,1	$3.4262 \cdot 10^{-2}$	1.3065	0.9995	-0.6211
40	40	10	$3.7654 \cdot 10^{-4}$	$7.2943 \cdot 10^{-5}$	0.7612	-0.0005

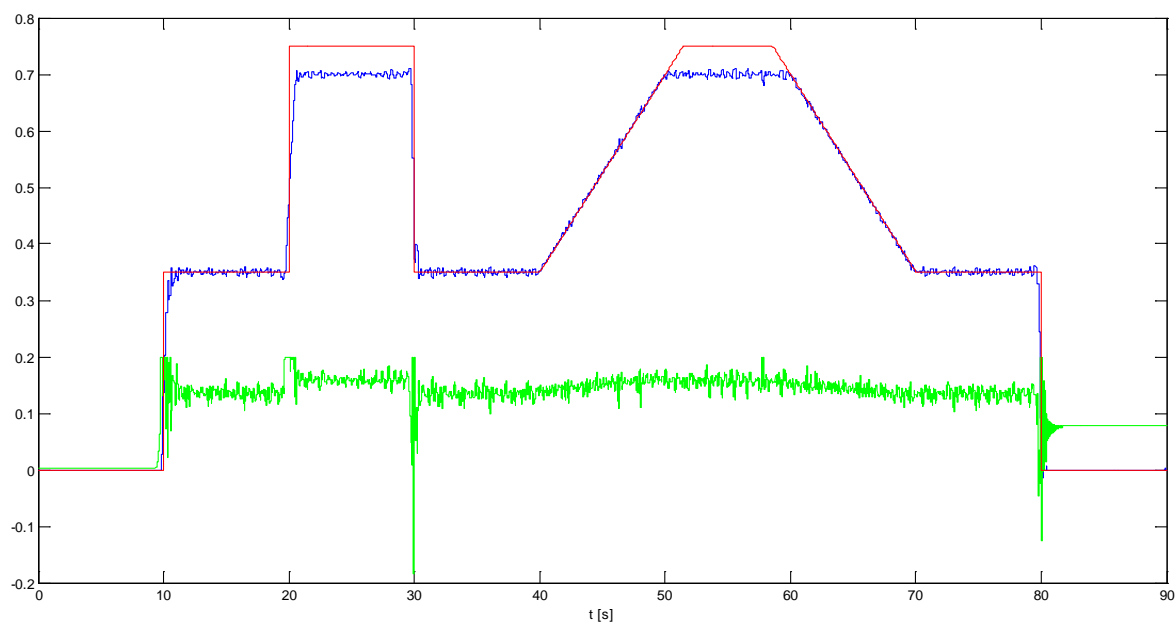
Z naměřených dat lze vyčíst, že při reálném měření je lepší volit vyšší hodnotu řídicího horizontu, což v simulaci nebylo tak patrné. Při volbě váhového parametru λ pod hodnotu 1 dochází k rozkmitání regulačního obvodu. Rozkmitání souvisí se šumem na reálném zařízení, kdy prediktivní regulátor se snaží eliminovat tento šum, čímž dochází k jeho rozkmitání. Hodnota váhového parametru $\lambda \geq 1$ do jisté míry zabraňuje přílišnému rozkmitání.



Obr. 28. Reálný průběh GPC bez omezení pro $N_u=N_2=40$ a $\lambda=1$

9.1.2 S omezením veličin

Hlavním nedostatkem zjednodušené verze GPC je nemožnost zadávat omezení. Proto dále použijeme GPC vypočtené kvadratickým programováním, kde již lze omezení zadat. Jako demonstrační příklad vezmeme omezení $u \in (-0.2, 0.2)$ a $y \in (-0.7, 0.7)$, která byla dříve použita v simulaci.



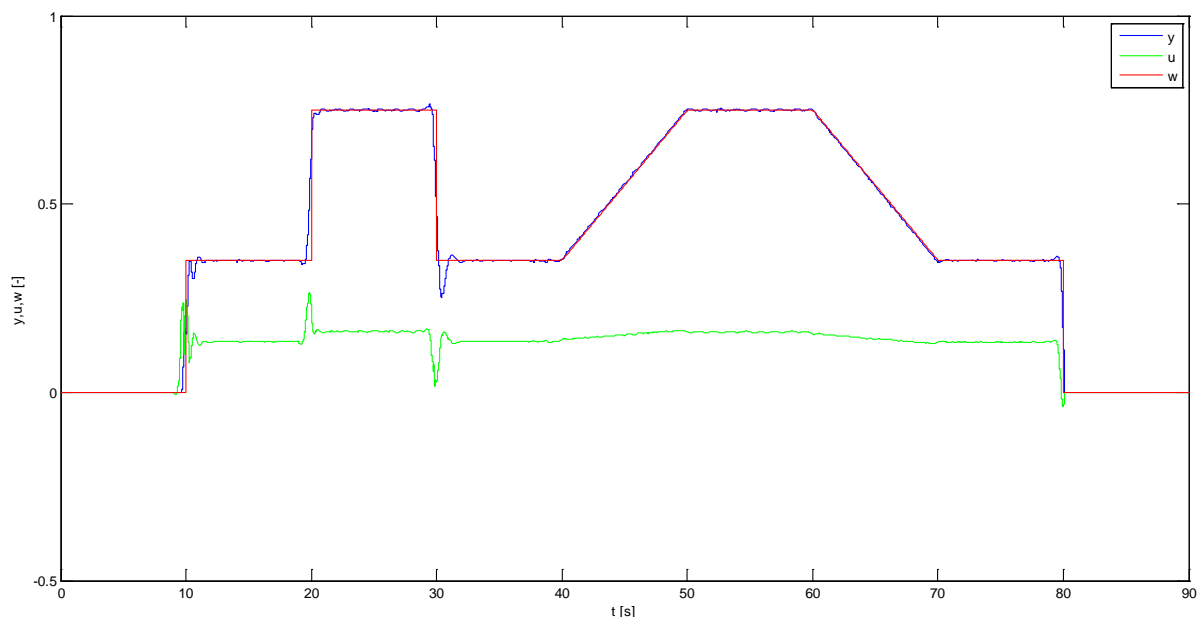
Obr. 29. Reálný průběh GPC QP s omezením veličin pro $N_u=N_2=20$ a $\lambda=1$

9.1.3 Adaptivní GPC

Dále otestujeme adaptivní verzi GPC. Kdy nastavujeme opět stejné parametry jako u klasického GPC, navíc zde můžeme nastavovat i parametry adaptivního směrového zapomínání. Kde jako optimální se jeví hodnoty $\varphi(0)=0.8$, $\lambda(0)=0.001$, $\nu(0)=10^{-6}$, $\rho=0.99$.

Tab. 13. Kvalita regulace pro adaptivní GPC a různé parametry

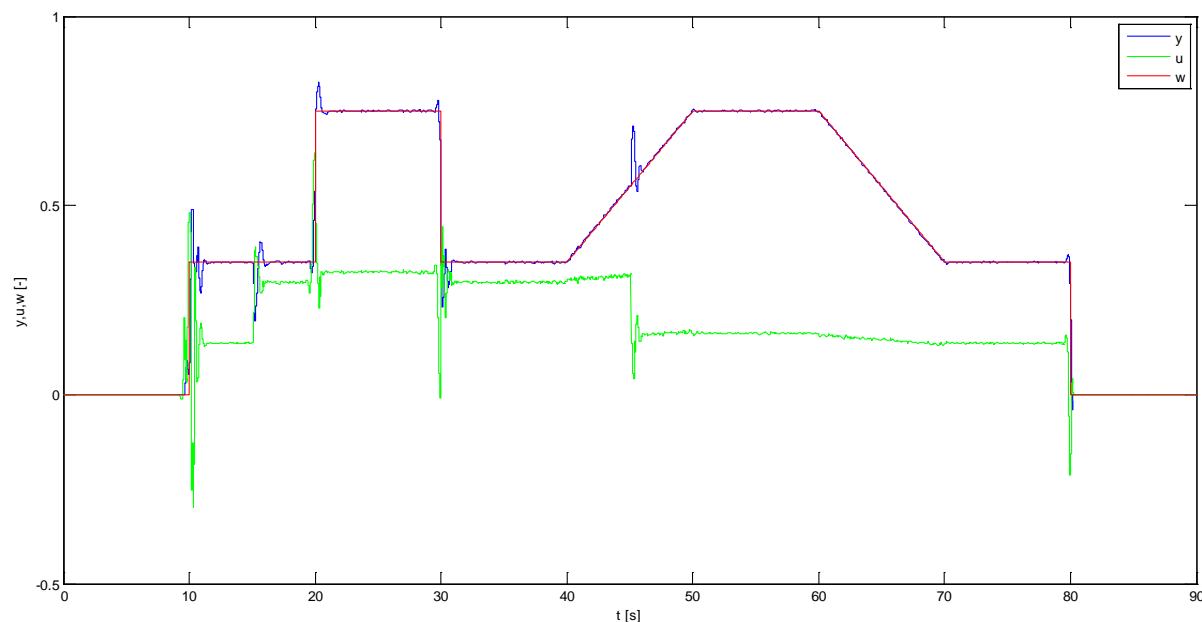
N_2	N_u	λ	S_y	S_u	y_{\max}	y_{\min}
10	10	1	$3,6337 \cdot 10^{-4}$	$3,9255 \cdot 10^{-4}$	0,8721	-0,0015
20	20	1	$3,3568 \cdot 10^{-4}$	$2,8332 \cdot 10^{-4}$	0,8657	-0,0146
20	20	10	$2,8342 \cdot 10^{-4}$	$2,4299 \cdot 10^{-5}$	0,7700	-0,0005
30	30	10	$3,1232 \cdot 10^{-4}$	$2,0616 \cdot 10^{-5}$	0,7681	-0,0020
40	40	10	$3,9070 \cdot 10^{-4}$	$1,0616 \cdot 10^{-5}$	0,7666	-0,0005



Obr. 30. Reálný průběh adaptivního GPC pro $N_2=N_u=20$ a $\lambda=10$

Adaptivní GPC vykazuje nepatrně horší hodnoty kvality regulace než klasické GPC. Je to způsobeno především nelinearitou systému, kdy dochází při větší skokové změně k větším překmitům.

Dále vyzkoušíme adaptivní GPC regulátor pro řízení reálného modelu, na který bude působit skoková porucha. Konkrétně v čase od 15 do 45 sekund bude na protichůdný servomotor poslána hodnota 0.3.



Obr. 31. Reálný průběh adaptivního GPC pro model s poruchou

Tab. 14. Srovnání kvality regulace adaptivního a klasického GPC

Druh	N_2	N_u	λ	S_y	S_u	y_{\max}	y_{\min}
A GPC	20	20	10	$4,1232 \cdot 10^{-4}$	$5,5065 \cdot 10^{-4}$	0,8245	-0,0391
GPC	30	30	1	$4,3852 \cdot 10^{-4}$	$1,0470 \cdot 10^{-3}$	0,7994	-0,0005

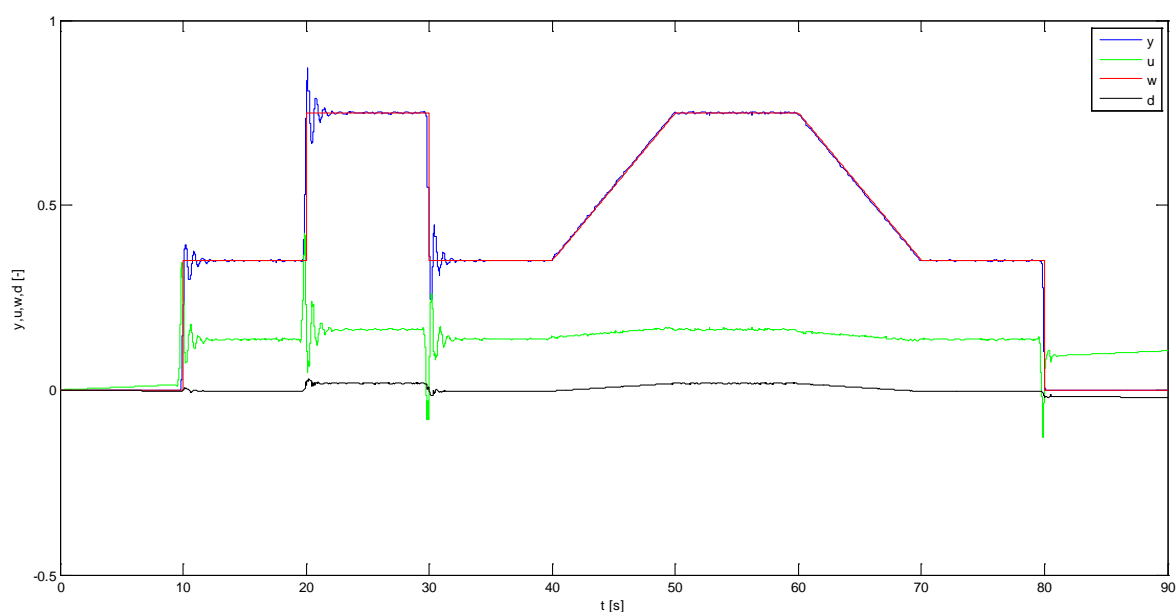
Adaptivní GPC reguluje model s poruchou s nepatrně menší regulační odchylkou než klasické GPC, a také s menší kvadratickou změnou akčního zásahu. Na kvalitu regulace adaptivního GPC má výrazný vliv počáteční nastavení parametrů adaptivního směrového zapomínání.

9.2 Offset-free MPC

U Offset-free MPC budeme nastavovat horizont a váhový parametr. Rozměr vektoru poruchy necháme totožný s rozměrem horizontu řízení, popř. jej dále upravíme.

Tab. 15. Kvalita regulace Offset-free MPC pro různé hodnoty parametrů

N_2	N_u	λ	S_y	S_u	y_{\max}	y_{\min}
5	5	1	$2,9328 \cdot 10^{-4}$	$3,2356 \cdot 10^{-4}$	0,8975	-0,0322
10	10	1	$3,5984 \cdot 10^{-4}$	$1,0990 \cdot 10^{-4}$	0,8823	-0,0005
20	20	1	$4,2540 \cdot 10^{-4}$	$8,2872 \cdot 10^{-5}$	0,8940	-0,0146
30	30	1	$4,5453 \cdot 10^{-4}$	$8,1968 \cdot 10^{-5}$	0,9282	-0,0015
10	10	0,1	$2,2010 \cdot 10^{-4}$	$6,5063 \cdot 10^{-4}$	0,9121	-0,0107
10	10	10	$4,3177 \cdot 10^{-4}$	$2,6719 \cdot 10^{-5}$	0,8384	-0,0005



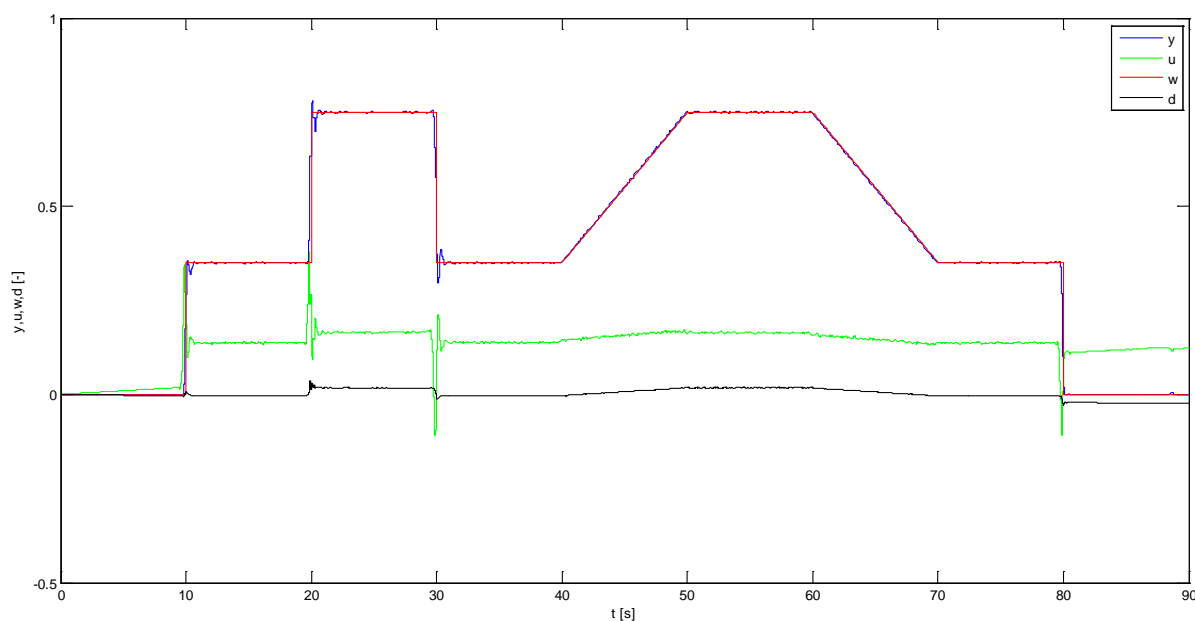
Obr. 32. Reálný průběh regulace Offset-free MPC pro $N_2=N_u=10$ a $\lambda=1$

Kvalita regulace je řádově obdobná jako u GPC. Pokud budeme chtít hodnoty kvality regulace nepatrně vylepšit, můžeme doplnit výpočet poruchy o exponenciální zapomínání. Kde nastavujeme koeficient zapomínání φ .

Tab. 16. Kvalita regulace Offset-free MPC s exponenciální zapomínáním poruchy

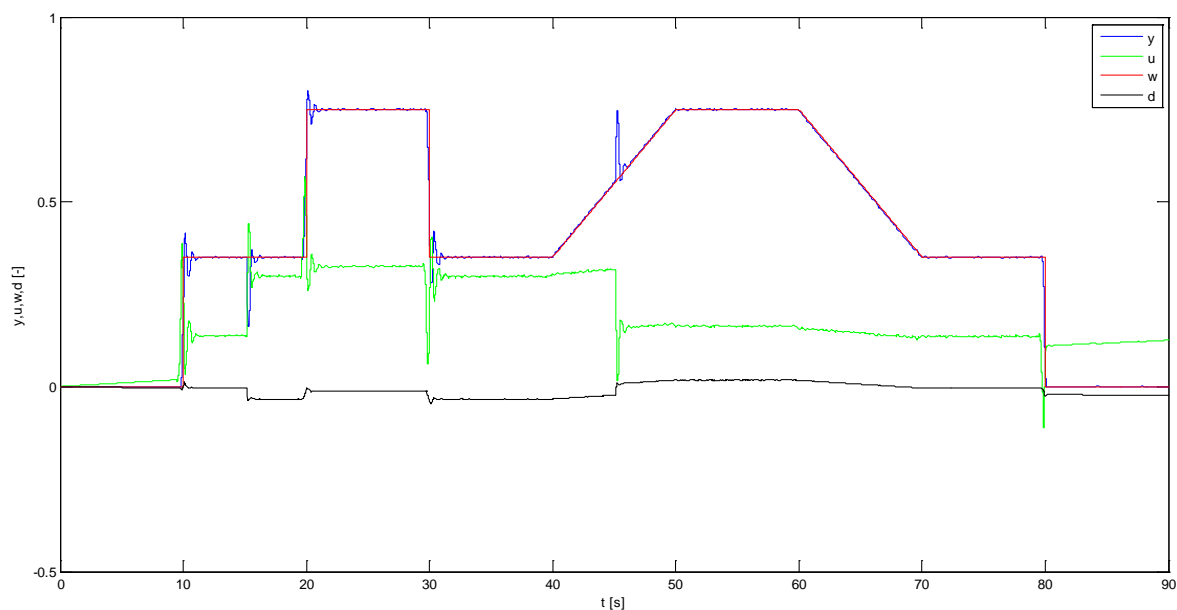
N_2	N_u	λ	φ	S_y	S_u	y_{\max}	y_{\min}
10	10	1	1	$3,6232 \cdot 10^{-4}$	$1,1539 \cdot 10^{-4}$	0,8701	-0,0005
10	10	1	0,9	$3,3448 \cdot 10^{-4}$	$1,0356 \cdot 10^{-4}$	0,8101	-0,0054
10	10	1	0,8	$2,3978 \cdot 10^{-4}$	$7,8362 \cdot 10^{-5}$	0,8110	-0,0005
10	10	1	0,7	$2,5693 \cdot 10^{-4}$	$1,1678 \cdot 10^{-4}$	0,7744	-0,0039
10	10	1	0,6	$2,4203 \cdot 10^{-4}$	$1,2935 \cdot 10^{-4}$	0,7949	-0,0005

Vhodnou volbou koeficientu zapomínání můžeme vcelku výrazně snížit hodnoty kvality regulace. Pro hodnotu $\varphi = 0.8$ má nejnižší kvadratickou regulační odchylku ze všech zkoumaných metod.

Obr. 33. Reálný průběh regulace Offset-free MPC pro $N_2=N_u=10$ a $\lambda=1$ a $\varphi=0.8$

9.2.1 Robustní kvalita regulace

Na závěr podrobíme Free-offset MPC regulaci modelu s poruchou. Stejně jako u GPC použijeme skokovou poruchu, kdy v čase od 15 do 45 sekund bude na protichůdný servomotor poslána hodnota 0.3.



Obr. 34. Reálný průběh Free-offset MPC pro model s poruchou

Tab. 17. Srovnání jednotlivých druhů MPC pro reálné řízení s poruchou

Druh	N_2	N_u	λ	S_y	S_u	y_{\max}	y_{\min}
O-F MPC	10	10	1	$3,7178 \cdot 10^{-4}$	$1,3629 \cdot 10^{-4}$	0,8008	-0,0005
A GPC	20	20	10	$4,1232 \cdot 10^{-4}$	$5,5065 \cdot 10^{-4}$	0,8245	-0,0391
GPC	30	30	1	$4,3852 \cdot 10^{-4}$	$1,0470 \cdot 10^{-3}$	0,7994	-0,0005

U reálného měření neplatí u Offset-free MPC tak jako v simulaci, že menší počet minulých poruch nám dá lepší kvalitu regulace. Při malém počtu minulých poruch dochází k značnému rozkmitání obvodu. Naopak pokud jsme aplikovali na vektor poruch exponenciální zapomínání, bylo pro vhodný parametr zapomínání dosaženo lepších hodnot kvality regulace. Pro regulaci modelu se skokovou poruchou dosahuje právě Offset-free MPC s exponenciálním zapomínáním poruchy nejlepšího výsledku kvality regulace ze zkoumaných metod.

ZÁVĚR

Tato práce se teoreticky zabývá problematikou robustnosti a prediktivního řízení. Vedle obecného popisu robustnosti a prediktivního řízení jsou zde odvozeny i některé konkrétní metody. Zjednodušená metoda GPC vypočtená inverzí matic, klasické GPC vypočtená kvadratickým programováním, samočinně se nastavující GPC a metoda Offset-free MPC. Dále je zde teoreticky odvozena metoda nejmenších čtverců, rekurzivní metoda nejmenších čtverců a rekurzivní MNČ s exponenciálním zapomínáním a adaptivním směrovým zapomínáním.

V praktické části jsou naprogramovány výše zmíněné algoritmy prediktivního řízení v prostředí Matlab/Simulink. Dále je zde vytvořen model laboratorní soustavy Amira DR300. Na tomto simulačním modelu jsou vyzkoušeny jednotlivé metody prediktivního řízení s cílem nalézt optimální nastavení a testovat kvalitu regulace a výpočetní náročnost v různých situacích. Jednotlivé prediktory jsou poté vyzkoušeny na servomechanismu Amira DR300.

Jednoznačně nejnižší výpočetní náročnost má GPC vypočtené inverzí matic, následuje GPC vypočtené kvadratickým programováním, dále Offset-free MPC a nakonec s nejvyšší výpočetní náročností GPC s adaptivním směrovým zapomínáním.

U robustní kvality regulace z pohledu schopnosti uregulovat model, mají adaptivní GPC a Offset-free MPC větší množinu modelů, které jsou schopny efektivně regulovat, v porovnání s GPC. U samotné kvality regulace záleží na regulovaném obvodu, tvaru žádané veličiny, poruchách v obvodu, a také na nastavení horizontu řízení a váhového parametru prediktoru. U Offset-free MPC s exponenciálním zapomínáním poruchy navíc nastavujeme váhový parametr zapomínání. U adaptivního GPC navíc záleží na počátečním nastavení parametrů směrového zapomínání. V průměru dosahovala nejlepších výsledků kvality regulace metoda Offset-free MPC.

ZÁVĚR V ANGLIČTINĚ

This thesis deal with the theory of robustness and predictive control. In addition to the general description of the robustness and Model Predictive Control there are also derived particular methods. Simplified GPC calculated by inversion of matrices, GPC calculated by quadratic programming, self-tuning GPC and Offset-free MPC. There are also theoretically derived least squares method, recursive least squares method and recursive LSM with exponential forgetting and with directional forgetting.

In the practical part of thesis they are programmed algorithms of predictive control in Matlab/Simulink. Here are also created models of laboratory plant Amira DR300. Particular predictors are tested on simulation models with focus on optimal setting, quality of control and computational complexity. Predictors are also tested on the real plant Amira DR300.

The lowest computational complexity has GPC calculated by inversion of matrices, followed by GPC calculated by quadratic programming, Offset-free MPC and finally self-tuning GPC with the highest computational complexity.

We will make focus for robustness of quality control. Self-tuning GPC and Offset-free MPC have larger set of models, which they are able to effectively control, compared with GPC. Its quality of control depends on the system, reference value, disturbance, also on the setting of control horizon and weighting parameter. Offset-free MPC with exponential forgetting has also forgetting parameter and adaptive GPC depends on the initial parameters of directional forgetting. On average has the best result for the quality of control Offset-free MPC.

SEZNAM POUŽITÉ LITERATURY

- [1] LACKO, Branislav. *Automatizace a automatizační technika*. Vyd. 1. Praha: Computer Press, 2000, x, 97 s. ISBN 80-722-6246-7.
- [2] PROKOP, Roman, Radek MATUŠŮ a Zdenka PROKOPOVÁ. *Teorie automatického řízení: lineární spojité dynamické systémy*. Vyd. 1. Zlín: Univerzita Tomáše Bati ve Zlíně, 2006, 102 s. ISBN 8073183692.
- [3] VACÁTKO, Jiří. Milníky a trendy automatizace technologických procesů. In: *Automa* [online]. 2006 [cit. 2014-06-07]. Dostupné z: <http://www.odbornecasopisy.cz/automa-36107.html>
- [4] BOBÁL, Vladimír. *Adaptivní a prediktivní řízení*. 1. vyd. Zlín: Univerzita Tomáše Bati ve Zlíně, 2008, 134 s. ISBN 9788073186623.
- [5] HOLKAR, K. S. a L.M. WAGHMARE. An Overview of Model Predictive Control. In: *International Journal of Control and Automation* [online]. 2010 [cit. 2014-05-15]. Dostupné z: http://www.sersc.org/journals/IJCA/vol3_no4/5.pdf
- [6] NUNES, G. C. *Desing and Analysis of Multivariable Predictive Control Applied to an Oil-Water-Gas Separator: A Polynomial Approach* [online]. Florida, 2001 [cit. 2014-05-22]. Dostupné z: <http://citeseerx.ist.psu.edu/> . University of Florida.
- [7] CLARKE, D. W., C. MOHTADI and P. S. TUFFS. *Generalized Predictive Control - Part I. The Basic Algorithm* [online]. 1987 [cit. 2014-04-19]. Dostupné z: <http://www.sciencedirect.com/science/article/pii/0005109887900872>
- [8] BEMPORAD, Alberto a Manfred MORARI. *Robust model predictive control: A survey* [online]. London: Springer-Verlag, 1999 [cit. 2014-05-27]. ISBN 9781846285387. DOI: 10.1007/BFb0109870. Dostupné z: <http://www8.cs.umu.se/research/ifor/dl/survey-robust-mpc.pdf>
- [9] MATUŠŮ, Radek a Petr DOSTÁL. *Stavová a algebraická teorie řízení* [online]. 2010 [cit. 2014-01-24]. ISBN 978-80-7318-991-4. Dostupné z: <http://hdl.handle.net/10563/18580>

- [10] SKOGESTAD, S. a I. POSTLETHWAITE. *Multivariable Feedback Control: Analysis and Design* [online]. Chichester, UK: John Wiley & Sons, 2005 [cit. 2014-05-27]. Dostupné z: http://www.fluidmal.uma.es/pdfs/Multivariable_Feedback_Control-Analysis_and_Design.pdf
- [11] BOBÁL, Vladimír. *Identifikace systémů*. Vyd. 1. Zlín: Univerzita Tomáše Bati ve Zlíně, 2009, 128 s. ISBN 978-80-7318-888-7.
- [12] BOBÁL, Vladimír, Petr CHALUPA, Marek KUBALČÍK a Petr DOSTÁL. Self-tuning Predictive Control of Nonlinear Servo-motor. In: *Acta Electrotechnica et Informatica*. 2011, 365–372. DOI: 10.2478/v10187-010-0056-x.
- [13] GRIMBLE, M. J. Generalized predictive optimal control: an introduction to the advantages and limitations. In: *International Journal of Systems Science*. Glasgow: University of Strathclyde, 1990, s. 85-98. Dostupné z: http://www.tandfonline.com/doi/abs/10.1080/00207729208949190#.U2qgG_1_tfc
- [14] MAEDER, Urban, Francesco BORRELLI a Manfred MORARI. Linear offset-free Model Predictive Control. *Automatica* [online]. 2009, vol. 45, issue 10, s. 2214-2222 [cit. 2014-06-11]. DOI: 10.1016/j.automatica.2009.06.005. Dostupné z: <http://linkinghub.elsevier.com/retrieve/pii/S0005109809002969>
- [15] ŠTECHA, Jan a Jaroslav PEKAŘ. *Short Course on Model Predictive Control*. Brno, 2010.
- [16] CHALUPA, Petr. PREDIKTIVNÍ ŘÍZENÍ NELINEÁRNÍHO SYSTÉMU. In: *KONFERENCE TECHNICAL COMPUTING PRAGUE 2006* [online]. 2006 [cit. 2014-05-27]. Dostupné z: http://dsp.vscht.cz/konference_matlab/MATLAB06/prispevky/chalupa/chalupa.pdf
- [17] CHALUPA, P. a J. NOVÁK. UNIVERZITA TOMÁŠE BATI. *Servo DR300 AMIRA*. Zlín, 2008.
- [18] ROUBAL, Jiří a AUGUSTA. A BRIEF INTRODUCTION TO CONTROL DESIGN DEMONSTRATED ON LABORATORY MODEL SERVO DR300 – AMIRA. In: *Acta Electrotechnica et Informatica* [online]. Košice, 2005 [cit. 2014-05-27]. Dostupné z: <http://www.aei.tuke.sk/papers/2005/4/Roubal.pdf>
- [19] AMIRA GmbH. *DR300: Laboratory Setup Speed Control with Variable Load*. Duisburg, 2000. Dostupné z: http://www.utm.mx/~jvasquez/motor_lab.pdf

- [20] SCHLEGEL, Miloš a Jaroslav SOBOTA. Prediktivní regulátor pro průmyslovou praxi. *AUTOMA*. 2007, č. 2, s. 12-16. Dostupné z: <http://www.odbornecasopisy.cz/download/au020712.pdf>

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

MPC Model Predictive Control.

GPC Generalized Predictive Control.

QP Quadratic programming.

MNČ Metoda nejmenších čtverců

SEZNAM OBRÁZKŮ

<i>Obr. 1. Grafické znázornění robustní stability pro multiplikativní neurčitost</i>	16
<i>Obr. 2. Grafické znázornění robustní kvality regulace pro multiplikativní neurčitost</i>	17
<i>Obr. 3. Bloková struktura prediktivního řízení</i>	18
<i>Obr. 4. Princip MPC.....</i>	19
<i>Obr. 5. Servomechanismus DR300 AMIRA</i>	45
<i>Obr. 6. Zprůměrovaná statická charakteristika servomechanismu Amira DR300.....</i>	47
<i>Obr. 7. Statická charakteristika servomechanismu pro rostoucí a klesající vstupní napětí</i>	48
<i>Obr. 8. Odezva modelu na náhodný signál.....</i>	49
<i>Obr. 9. Regulační průběh GPC bez omezení pro $N_u=N_2=20$</i>	53
<i>Obr. 10. Regulační průběh GPC bez omezení pro $\lambda=0.01$</i>	54
<i>Obr. 11. Regulační průběh GPC bez omezení pro $\lambda=100$</i>	54
<i>Obr. 12. Regulační průběh GPC bez omezení pro model (7.8)</i>	55
<i>Obr. 13. Regulační průběh GPC bez omezení pro model (6.12)</i>	56
<i>Obr. 14. Přejchodová charakteristika pro jednotlivé modely.....</i>	57
<i>Obr. 15. Přejchodová charakteristika různých modelů</i>	58
<i>Obr. 16. Regulační průběh GPC bez omezení pro model (8.1)</i>	58
<i>Obr. 17. Regulační průběh GPC bez omezení pro model s vnějšími poruchami</i>	59
<i>Obr. 18. Regulační průběh GPC bez omezení se saturací veličin</i>	60
<i>Obr. 19. Regulační průběh GPC s omezením veličin - $0.2 < u < 0.2$ a $-0.7 < y < 0.7$</i>	61
<i>Obr. 20. Regulační průběh GPC s omezením veličin - $0.1 < u < 0.1$ a $0 < y < 0.7$</i>	61
<i>Obr. 21. Regulační průběh adaptivního GPC pro model (8.1)</i>	63
<i>Obr. 22. Regulační průběh adaptivního GPC pro model (7.8) s poruchami</i>	64
<i>Obr. 23. Regulační průběh klasického GPC pro model (7.8) s rostoucím zesílením systému</i>	64
<i>Obr. 24. Regulační průběh adaptivní GPC pro model (7.8) s rostoucím zesílením systému</i>	65
<i>Obr. 25. Regulační průběh Offset-free MPC pro model (7.9)</i>	66
<i>Obr. 26. Regulační průběh Offset-free MPC pro model (7.8) a $N_u=N_2=5$</i>	67
<i>Obr. 27. Regulační průběh Offset-free MPC pro model (7.8) s poruchou a $N_2=N_u=20$</i>	68
<i>Obr. 28. Reálný průběh GPC bez omezení pro $N_u=N_2=40$ a $\lambda=1$</i>	71
<i>Obr. 29. Reálný průběh GPC s omezením veličin pro $N_u=N_2=20$ a $\lambda=1$</i>	71
<i>Obr. 30. Reálný průběh adaptivního GPC pro $N_2=N_u=20$ a $\lambda=10$</i>	72

<i>Obr. 31. Reálný průběh adaptivního GPC pro model s poruchou</i>	<i>73</i>
<i>Obr. 32. Reálný průběh regulace Offset-free MPC pro $N_2=N_u=10$ a $\lambda=1$</i>	<i>74</i>
<i>Obr. 33. Reálný průběh regulace Offset-free MPC pro $N_2=N_u=10$ a $\lambda=1$ a $\varphi=0.8$</i>	<i>75</i>
<i>Obr. 34. Reálný průběh Free-offset MPC pro model s poruchou</i>	<i>76</i>

SEZNAM TABULEK

<i>Tab. 1. Srovnání kvadratických regulačních odchylek pro jednotlivé MNČ</i>	<i>50</i>
<i>Tab. 2. Nastavení horizontu predikce pro GPC bez omezení</i>	<i>52</i>
<i>Tab. 3. Nastavení váhového parametru pro GPC bez omezení</i>	<i>53</i>
<i>Tab. 4. Kvalita regulace GPC bez omezení pro různé simulační modely</i>	<i>56</i>
<i>Tab. 5. Srovnání kvality regulace pro jednotlivé verze GPC</i>	<i>60</i>
<i>Tab. 6. Regulační hodnoty pro GPC s omezením akční veličiny</i>	<i>62</i>
<i>Tab. 7. Porovnání kvality regulace GPC a adaptivního GPC.....</i>	<i>63</i>
<i>Tab. 8. Srovnání GPC a adaptivního GPC pro proměnný systém</i>	<i>65</i>
<i>Tab. 9. Nastavení horizontu predikce a váhového parametru pro Offset-free MPC..</i>	<i>67</i>
<i>Tab. 10. Porovnání kvality regulace Offset-free MPC pro model (7.8)</i>	<i>67</i>
<i>Tab. 11. Porovnání kvality regulace Offset-free MPC pro model (7.8) s poruchami</i>	<i>68</i>
<i>Tab. 12. Optimální parametry GPC pro reálné měření</i>	<i>70</i>
<i>Tab. 13. Kvalita regulace pro adaptivní GPC a různé parametry</i>	<i>72</i>
<i>Tab. 14. Srovnání kvality regulace adaptivního a klasického GPC</i>	<i>73</i>
<i>Tab. 15. Kvalita regulace Offset-free MPC pro různé hodnoty parametrů.....</i>	<i>74</i>
<i>Tab. 16. Kvalita regulace Offset-free MPC s exponenciální zapomínáním poruchy .</i>	<i>75</i>
<i>Tab. 17. Srovnání jednotlivých druhů MPC pro reálné řízení s poruchou</i>	<i>76</i>

SEZNAM PŘÍLOH

- P I Přiložený CD-ROM s jednotlivými algoritmy prediktivního řízení v prostředí Matlab/Simulink.