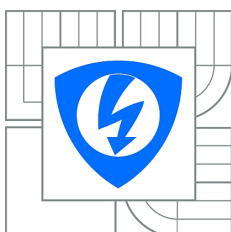


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV AUTOMATIZACE A MĚŘICÍ TECHNIKY

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF CONTROL AND INSTRUMENTATION

ADAPTIVNÍ REGULÁTORY PRO SYSTÉMY S DOPRAVNÍM ZPOŽDĚNÍM A JEJICH POROVNÁNÍ S KLASICKÝMI PEVNĚ NASTAVENÝMI REGULÁTORY.

ADAPTIVE CONTROLLERS FOR SYSTEMS WITH TIME DELAY AND ITS COMPARISON
WITH CLASSICAL CONTROLLERS.

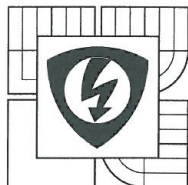
DIPLOMOVÁ PRÁCE
DIPLOMA THESIS

AUTOR PRÁCE
AUTHOR

Bc. DANIEL KRYKORKA

VEDOUCÍ PRÁCE
SUPERVISOR

Prof. Ing. PETR PIVOŇKA, CSc.



VYSOKÉ UČENÍ
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

Ústav automatizace a měřicí techniky

Diplomová práce

magisterský navazující studijní obor
Kybernetika, automatizace a měření

Student: Bc. Daniel Krykorka

Ročník: 2

ID: 121028

Akademický rok: 2014/15

NÁZEV TÉMATU:

**Adaptivní regulátory pro systémy s dopravním zpožděním a jejich
porovnání
s klasickými pevně nastavenými parametry regulátorů.**

POKYNY PRO VYPRACOVÁNÍ:

1. Zpracujte rešerši týkající se adaptivních regulátorů.
2. Seznamte se s metodikou návrhu adaptivních regulátorů.
3. Seznamte se s řízením systémů s dopravním zpožděním.
4. Zaměřte se na možnosti implementace adaptivního regulátoru z prostředí programu MATLAB do programovatelného automatu.
5. Zabývejte se vlastnostmi funkčního bloku dopravního zpoždění v kombinaci spojitá soustava - diskretní regulátor.
6. Ověřte možnost použití adaptivních regulátorů u soustav s dopravním zpožděním.
7. Výsledky porovnejte s klasickými regulátory s pevně nastavenými parametry.

DOPORUČENÁ LITERATURA:

PIVOŇKA, P.: Optimalizace regulátorů, VUT Brno, skriptum, 2004.

BOBÁL, V. a kol.: Praktické aspekty samočinně se nastavujících regulátorů. VUT IUM, 1999.

Termín zadání: 9. 2. 2015

Termín odevzdání: 18.5.2015

Vedoucí práce: prof. Ing. Petr Pivoňka, CSc.

Konzultanti diplomové práce:

doc. Ing. Václav Jirsík, CSc.
předseda oborové rady

UPOZORNĚNÍ:

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



Abstrakt

Práce se zabývá metodikou návrhu adaptivního regulátoru. V teoretické části práce jsou rozebrány jednotlivé části adaptivního regulátoru. Těmi jsou online identifikace pomocí metody nejmenších čtverců a PSD regulátor, který si nastavuje parametry dle identifikované soustavy (pomocí metody Z-N). Závěr teoretické části je zaměřen na regulaci soustav s dopravním zpožděním a to zejména při využití Smithova prediktoru.

V praktické části jsou pomocí simulací ověřeny všechny algoritmy uvedené v teoretické části. Simulace probíhají jak na simulačních, tak reálných modelech.

Summary

Master thesis is focused on the philosophy of design adaptive controller. In the theoretic part are described parts of the adaptive controller, which belongs parts as online identification by recursive least-squares method and PSD controller, which can set its parameters according to identified system (use Z-N method). The part of control system with transport delay is situated at the conclusion of the theoretic part, there are focused on Smith predictor.

Practical part is focused on verification of all algorithms, which was performed on models and real systems.

Klíčová slova

Adaptivní regulátor, metoda nejmenších čtverců, online identifikace, PSD regulátor, dopravní zpoždění, Smithův prediktor

Keywords

Adaptive controller, Recursive Least-Squares method, online identification, PSD controller, Transport Delay, Smith predictor

KRYKORKA, D. *Adaptivní regulátory pro systémy s dopravním zpožděním a jejich porovnání s klasickými pevně nastavenými regulátory*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2015. ?? s. Vedoucí prof. Ing. Petr Pivoňka, CSc.

„Prohlašuji, že svou diplomovou práci na téma Adaptivní regulátory pro systémy s dopravním zpožděním a jejich porovnání s klasickými pevně nastavenými regulátory jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené diplomové práce dále prohlašuji, že v souvislosti s vytvořením této diplomové práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.“

Bc. Daniel Krykorka

Děkuji vedoucímu diplomové práce prof. Ing. Petru Pivoňkovi, CSc. za účinnou metodickou, pedagogickou a odbornou pomoc a další cenné rady při zpracování mé diplomové práce.

Bc. Daniel Krykorka

Obsah

1 Úvod	3
2 Řídicí systémy	4
2.1 Vliv A/D a D/A převodníků	4
2.2 Vliv vzorkování	4
2.3 Varianty PSD regulátorů	5
2.3.1 Diskrétní ekvivalent PID regulátoru	6
2.3.2 PSD regulátor s filtrací derivační složky	7
2.3.3 β PSD regulátor	8
2.3.4 Feed Forward regulátor	9
2.3.5 Takahashiho regulátor	9
2.4 Regulační obvody s modelem regulované soustavy	9
2.5 Návrh regulátoru	10
2.5.1 Pokus-omyl	10
2.5.2 Modifikované Ziegler-Nicholsovo kritérium	11
2.6 Adaptivní řídicí systémy	11
2.6.1 Samočinně se nastavující regulátory (STC)	12
2.6.2 Výpočet kritických parametrů	13
3 Identifikace dynamického systému	16
3.1 Testovací signál	16
3.1.1 Skoková změna	16
3.1.2 Pseudonáhodná binární posloupnost	17
3.1.3 Součet harmonických průběhů	17
3.2 Modely pro identifikaci procesů	18
3.2.1 ARX	18
3.2.2 ARMAX	18
3.3 Volba řádu modelu	19
3.4 Identifikační metody	20
3.4.1 Metoda nejmenších čtverců	21
3.4.2 Průběžná metoda nejmenších čtverců	21
3.4.3 Konstantní exponenciální zapomínání	22
3.4.4 Proměnné exponenciální zapomínání	22
4 Dopravní zpoždění	24
4.1 Identifikace	24
4.2 Simulace	24
4.3 Smithův prediktor	25
5 Návrh a realizace řídicího systému	27
5.1 TomPack	27
5.1.1 Struktura TomPacku	27
5.1.2 Postup realizace	28
5.2 MATLAB	30
5.3 Fyzická soustava	31

OBSAH

5.3.1	Akční člen	31
5.3.2	Měřicí člen	32
5.3.3	Realizace	33
5.3.4	Model soustavy	35
6	Ověření teorie	36
6.1	Simulace	36
6.2	Fyzická soustava	39
7	Závěr	40

1. Úvod

Díky velkému výpočetnímu výkonu dnes není problém realizovat výpočetně náročné algoritmy. Mohli bychom tedy očekávat velké rozšíření pokročilých způsobů řízení v praxi. Opak je však pravdou. Navíc většina regulátorů ani není správně nastavena, a tak plní svoji funkci jenom omezeně. [9]

V praxi stěžují takové ladění regulátoru nejenom pracovní podmínky a časová tíseň, ale i proměnné dynamiky řízených systémů v čase, nebo nepřesnosti měření doplněné šumem. Dále pak může potrápít systém s neminimální fází, nebo všudypřítomné dopravním zpožděním (dále DZ). Postupně budou rozebrány všechny problémy, které mohou nastat, a bude uveden i možný způsob řešení. Důraz je kladen zejména na systémy s dopravním zpožděním, které jsou pomocí klasických PID regulátorů velmi těžko regulovatelné.

Tato práce se zabývá primárně možnostmi zlepšení tohoto stavu. Je rozdělena do 4 logických celků. V prvním je popsána struktura pevně nastavitelných a adaptivních regulátorů s různými vlivy, které na ně působí. Dále je zde uveden způsob jejich nastavení. Ve druhém pak možnosti a způsoby identifikace dynamického systému, což je klíčová část při návrhu regulátoru. Další část se pak zabývá DZ, jeho vlivem na regulaci a možnostmi regulace systému s dominantním DZ zejména pomocí Smithova prediktoru. V poslední části se pak zabývá možnostmi implementace různých algoritmů pro variaci řídicích systémů.

2. Řídicí systémy

V praxi se nejčastěji můžeme setkat s pevně nastavenými regulátory typu PID více či méně vhodně nastavenými konstantami. Je to způsobeno časovou tísňí nálezů technologie, neodborném zásahu obsluhy, nezkušeností technika. Nebo se může jednat o systém, který pozměnil svoje vlastnosti svojí dynamiku v důsledku drobné závady. Statistika uvádí, že více jak 3/4 PID regulátorů v provozu jsou špatně nastaveny. [9]

V této kapitole se budeme věnovat běžným problémům, které nastávají v reálných systémech a budou ukázány základní postupy při návrhu pevně nastavitelných regulátorů a některé jejich variace. V další části rozebereme a přiblížíme skupinu adaptivních regulátorů.

2.1. Vliv A/D a D/A převodníků

Na kvalitu regulace má samozřejmě vliv i kvalita převodníků, čidel pro zpětnou vazbu i akčních členů. Záleží na jejich přesnosti a rozlišení. Převodník má dva základní parametry. Rozsah měření a rozlišení. Při převodu z analogového na číslicový signál vždy dostaneme konečný počet prvků, kterých může veličina nabývat. Tato hodnota se z pravidla udává v bitech. Konečný počet prvků dostaneme jednoduše z následující rovnice:

$$N = 2^b \quad (2.1)$$

Hodnoty úrovní signálu tedy projdou zaokrouhlením (kvantizací) na svoji kvantovou úroveň. Rozdíl mezi reálnou a kvantovanou se nazývá kvantizační chyba. Její maximální velikost odpovídá polovině kvantizační úrovně.

Pokud budeme používat běžný průmyslový převodník, např. pro měření proudové smyčky 4 – 20 mA při rozlišení 13-bit, dostaneme se na přesnost 2.44 μ A. Z toho důvodu můžeme v praktických příkladech počítat s maximální přesností na 4 platná místa, a je to stejný důvod, proč při zpětné rekonstrukci nemůžeme nikdy dostat stejný signál, jako před vzorkováním.

Ve většině případů je tato chyba zanedbatelná, ale pokud se pohybujeme v hodnotách velmi blízkých kvantizačním úrovním, může se výrazně projevit kvantizační šum způsobený právě kvantizační chybou.

2.2. Vliv vzorkování

Ať už se jedná o pevně nastavitelný regulátor, nebo o adaptivní, vždy budeme muset brát v úvahu vliv vzorkování. Velikost vzorkovací periody výrazně ovlivňuje dynamiku systému a má tak vliv i na jeho stabilitu. V reálných systémech musíme navíc počítat jak s procesním, tak i s kvantizačním šumem. Navíc musíme počítat s dopravním zpožděním, které samotné vzorkování způsobuje, a které se přičítá k fázi regulovaného systému. To můžeme přibližně aproximovat časovou konstantou dopravního zpoždění rovnou $\tau = \frac{T_{vz}}{2}$.

Pokud používáme periferie pro získávání dat, je třeba započítat do celkového zpoždění i odezvu T_D těchto periférií. U některých technologických sítí můžeme počítat s cyklem přes $100ms$ a odezvou v řádech jednotek μs až po stovky μs . Je třeba dodat, že ne všechny technologické sítě umožňují synchronizaci, nebo nedodávají časovou značku. A tak nelze zaručit, v jaký okamžik přesně vzorkování proběhlo. Dále musíme započítat dobu výpočtu T_V , tedy dobu jednoho cyklu procesoru, která nemusí být vždy zanedbatelná. Celková doba zpoždění tedy odpovídá $\tau_c \approx (\frac{T_{vz}}{2} + T_V + T_D)$. [12]

Pokud však výpočet řídicího algoritmu neprobíhá na PLC, ale na ovládacím PC, je nutno k τ_c připočítat desetiny až jednotky sekund s ohledem použitý komunikační protokol, převodníky, a množství komunikovaných dat.

Samotné vzorkování se z hlediska dynamiky jeví jako dolní propust. Pokud např. nastavíme moc krátkou periodu vzorkování, pak bude PI regulátor na většině soustav stále dobře fungovat, ale pokud se budeme snažit přidávat D složku, zjistíme, že se vlastní dynamika soustavy nemění, ale výstup z regulátoru začne být výrazně zašuměný. Při dalším zesilování vlivu D složky se pak může regulátor chovat doslova náhodně, a soustava tak může přejít do nestabilní oblasti.

Vzhledem k tomu, že nám perioda vzorkování ovlivňuje i pro proces identifikace systémů, je obtížné najít optimální hodnotu. To platí obzvláště u adaptivních systémů, kde je proces identifikace svázána s regulačním algoritmem. Existují ovšem jistá doporučení:

$$T_{vz} = \left(\frac{1}{6} - \frac{1}{15} \right) T_{95} \quad (2.2a)$$

$$T_{vz} = \left(\frac{1}{8} - \frac{1}{16} \right) f \quad (2.2b)$$

$$T_{vz} = \frac{1}{10} T_G \quad (2.2c)$$

T_{95} ... 95% doby náběhu.

f ... Vlastní frekvence uzavřené smyčky.

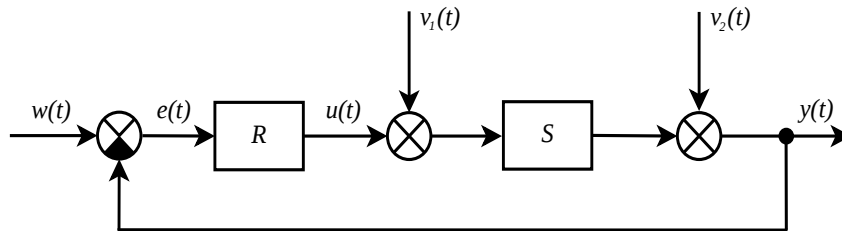
T_G ... Součet všech časových konstant systému.

Pro vyregulování poruchy se u PSD regulátorů volí až o řád kratší perioda, než je uvedeno např. v (2.2c). Naopak s delší periodou vzorkování rychleji konvergují parametry modelu k reálným hodnotám. Proto je volba T_{vz} vždy kompromisem.

2.3. Varianty PSD regulátorů

V této části se budeme zabývat diskretním ekvivalentem PID regulátoru a jeho různými variacemi. Budeme vycházet za základního regulačního schématu, který je na obr. 2.1. Regulátor má zajistit, aby sledovaná veličina $y(t)$ (teplota, hladina, průtok, kvalita, ...) sledovala co nejpřesněji žádanou veličinu $w(t)$. Krom toho požadujeme, aby se kvalitně vypořádal s poruchovými signály $v(t)$, a byl dostatečně robustní vzhledem ke stabilitě a kvalitě řízení. [15]

2.3. VARIANTY PSD REGULÁTORŮ



Obrázek 2.1: Regulační schéma.

PID regulátor se skládá jak je znázorněno v rovnici 2.3 ze tří složek: proporční, integrační a derivační.

$$u(t) = r_0 e(t) + r_D \frac{de(t)}{dt} + r_I \int_0^t e(\tau) d\tau + u(0) \quad (2.3)$$

Proporční složka je základní složkou regulátoru, je velmi rychlá, ale sama nedokáže docílit ustálené nulové regulační odchylky $e(t)$. Tomu pomáhá integrační složka, která ale celý děj zpomaluje. Derivační složka přispívá do určité meze ke zvýšení stability systému a zároveň celý děj zrychluje.

V praxi se většina technologických procesů reguluje pomocí PI regulátorů. Samotný P regulátor se používá hojně v kaskádní regulaci, nebo kde dáváme přednost rychlé reakci před ustálenou regulační odchylkou.

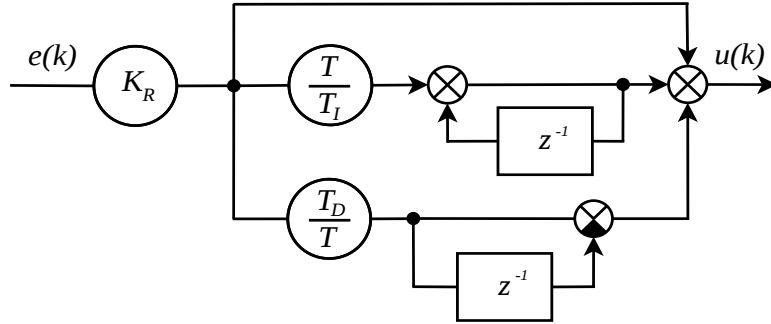
2.3.1. Diskrétní ekvivalent PID regulátoru

Vzhledem k tomu, že dnes celý provozní soubor dokáže řídit jeden průmyslový automat, setkáváme se pouze s diskrétní formou PID regulátoru. Tedy integrační složku zaměníme za sumační a dostaneme regulátor PSD. Díky tomu, že velký soubor regulací dokáže řídit jeden automat, dokážeme celý systém nejenom postavit za zlomek ceny, ale jednotlivé regulátory libovolně modifikovat jak uvidíme dále. Např. tam kde to je nutné opatřit případně číslicovými filtry, což nejenom kompenzuje problémy spojené s diskretizací a kvantováním, ale dodává díky velkému výkonu dnešních procesorů i velkou přidanou hodnotu.

$$F_R(p) = \frac{X(p)}{E(p)} = K_R \left(1 + T_D p + \frac{1}{T_I p} \right) \quad (2.4)$$

$$F_R(z^{-1}) = K_R \left(1 + \frac{T_D}{T} (1 - z^{-1}) + \frac{T}{T_I} \frac{1}{1 - z^{-1}} \right) \quad (2.5)$$

Pokud provedeme Laplaceovu transformaci rovnice 2.3, pak dostaneme jako jeden z možných tvarů rovnici 2.4, která odpovídá přesně struktuře PID regulátoru. Z této rovnice už pomocí drobných úprav dostaneme diskrétní tvar v \mathcal{Z} -transformaci, který zobrazuje rovnice 2.5 a blokové schéma na obr. 2.2.



Obrázek 2.2: Regulační schéma.[16]

Z hlediska rovnic se může jevit, že diskrétní ekvivalent PID regulátoru se bude chovat stejně, ale vzhledem k tomu, že jsme přidali vzorkovač, tak jsme do systému přidali i další dynamiku (popsanou podrobněji výše v textu), která nám způsobuje větší překmit.

2.3.2. PSD regulátor s filtrací derivační složky

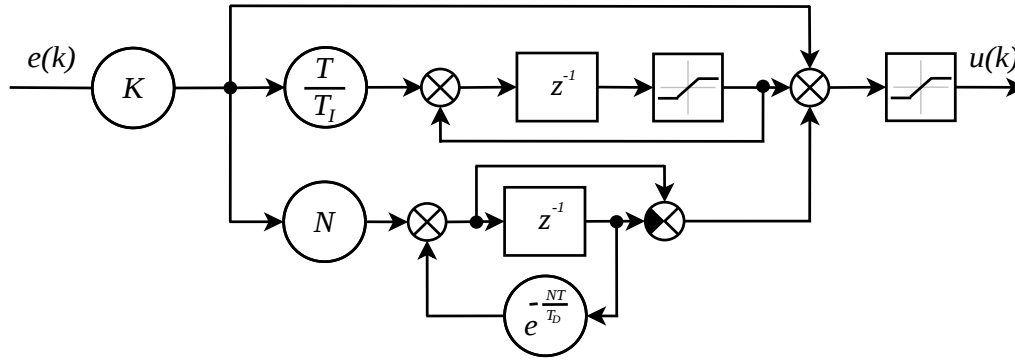
V případech, kdy máme krátkou periodu vzorkování, a do systému nám proniká rušení, je vhodné zavést filtr derivační složky. Rozkmitaný akční zásah způsobený nefiltrovanou derivační složkou regulátoru na rychlém procesu může rozkmitat výstup procesu. Dolní propust u derivační složky zamezí silnému pronikání šumu do systému, zlepší tak celkovou kvalitu regulace a může výrazně prodloužit životnost akčního členu. Přenosovou funkci tohoto regulátoru zobrazuje rovnice 2.6.

$$F_R(z) = K \left(1 + \frac{Tz^{-1}}{T_I(1 - z^{-1})} + N \frac{1 - z^{-1}}{1 - e^{-\frac{T_N}{T_D}} z^{-1}} \right) \quad (2.6)$$

Další inovací, která zde byla zavedena, je jednoduché omezení wind-up jevu. Tento jev nastává pokud akční člen nedokáže dodat dostatečně velký akční zásah, aby snížil regulační odchylku. To způsobí narůstání biasu integrátoru, které způsobí velké zpoždění při další reakci. Tím zpozdí reakci akčního zásahu na další změnu. Vlastní zásah spočívá v omezení biasu pouze na meze akčního členu. Výsledný stavový diagram PSD regulátoru s filtrací derivační složky je zobrazen na obr. 2.3.

Parametr N určuje časovou konstantu filtru $\frac{T_D}{N}$. Jeho typická hodnota leží v intervalu $N = \langle 3; 20 \rangle$ [12], kde v nízkých hodnotách silně potlačují derivační složku a PSD se více blíží PS regulátoru. [15]

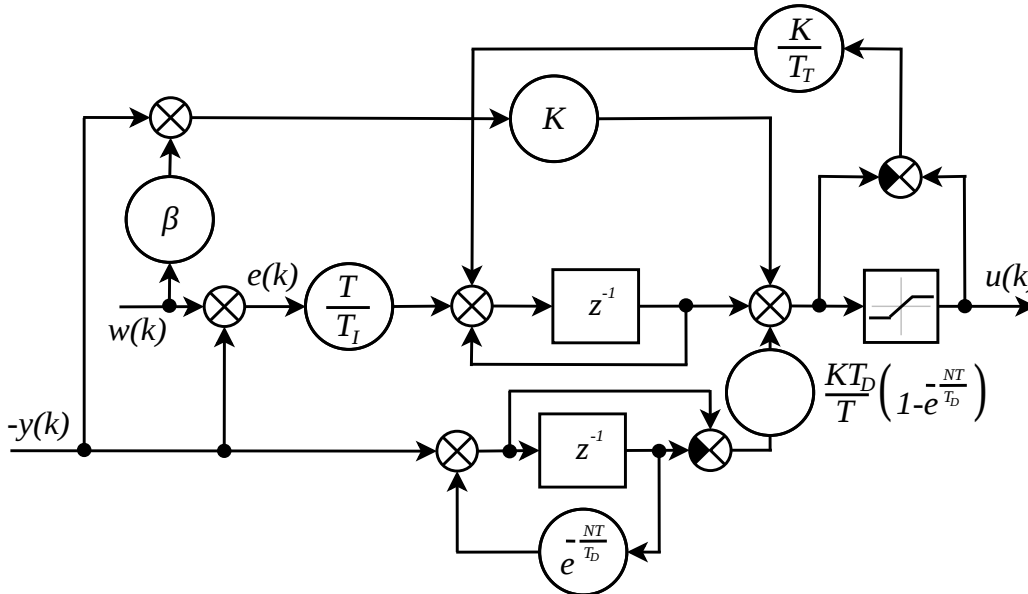
2.3. VARIANTY PSD REGULÁTORŮ



Obrázek 2.3: PSD regulátor s filtrací derivační složky. [12]

2.3.3. β PSD regulátor

Regulátorem se dvěma stupni volnosti můžeme nezávisle na sobě ladit dynamiku odezvy na vstupní signál a dynamiku odezvy na působení poruchy.



Obrázek 2.4: Stavový diagram diskrétního S-PD a PS-D regulátoru. [12]

Odchylka pro výpočet proporcionální části je:

$$e_p = \beta w - y \quad (2.7)$$

Parametr β zde nabývá nejčastěji pouze hodnot 0, nebo 1, ale může nabývat jakýchkoliv hodnot v tomto rozmezí. Pokud bude $\beta = 0$, bude se zde jednat o S-PD regulátor. Pokud bude $\beta = 1$, dostaneme SP-D regulátor.

Pro výpočet sumační části musí zůstat regulační odchylka.

$$e = w - y \quad (2.8)$$

Pro derivační část je možné volit stejný postup, jako u proporční části, ale trvale odstraníme vliv žádané veličiny a zamezíme tak prudkým akčním zásahům. Opačně tomu

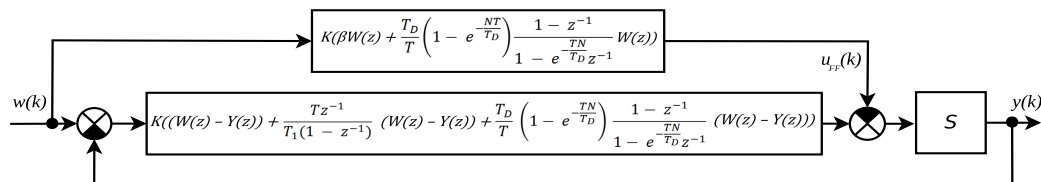
může být, pokud použijeme regulátor uvnitř kaskády, kde je žádaná veličina dostatečně hladká, a nehrozí rázy. [15]

$$e_d = -y \quad (2.9)$$

Na obr. 2.4 je zobrazen diagram β PSD regulátoru s pokročilejším potlačením wind-up jevu. Jedná se o zavedení další vazby od rozdílového členu, jehož účelem je rychlejší odintegrování biasu. Pokud ovšem zle nastavíme konstantu zesílení $\frac{K}{T_i}$ můžeme zavést do systému nestabilitu. Rozdíl se zde bere od akčních zásahů regulátoru a hodnot, kterých může reálně nabývat akční člen.

2.3.4. Feed Forward regulátor

Tento regulátor je variantou S-PD regulátoru. V regulační smyčce nalezneme PSD regulátor a v součtovém členu je od žádané veličiny odečtena P a D složka. Schéma je znázorněno na obr. 5.1.



Obrázek 2.5: Feed Forward. [12]

Tento regulátor je vhodný pokud potřebujeme překonat počáteční tření, nebo chceme omezit překmit.

2.3.5. Takahashiho regulátor

Toto je další forma regulátoru vhodná pro eliminaci rychlých změn akční veličiny v důsledku skokové změny žádané veličiny. Pro P a D složku zaměníme $e(k)$ za $y(k)$. Nevýhodou tohoto regulátoru je pomalá odezva na poruchový signál.

$$u(k) = K_P[y(k-1) - y(k)] + K_I[w(k) - y(k)] + K_D[2y(k-1) - y(k-2) - y(k)] + u(k-1) \quad (2.10)$$

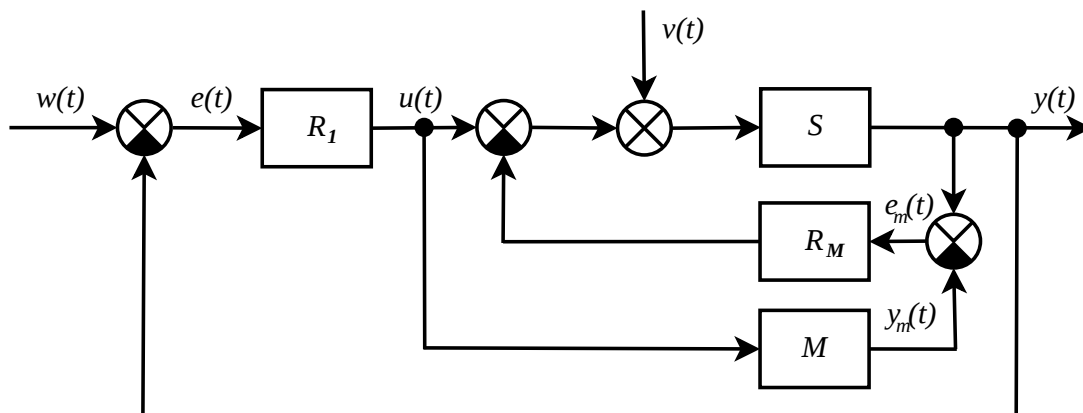
Konstanty určíme následovně:

$$K_P = 0,6K_{krit} - \frac{K_I}{2}; K_I = \frac{1,2K_{krit}T}{T_{krit}}; K_D = \frac{3K_{krit}T_{krit}}{40T} \quad (2.11)$$

2.4. Regulační obvody s modelem regulované soustavy

Tyto systémy se používají převážně v adaptivních obvodech, ale i v jednoduchém regulačním obvodu mohou zlepšit kvalitu regulace především u hůře regulovatelných systémů, kde kvalita řízení dosažená PID regulátorem není vyhovující.

2.5. NÁVRH REGULÁTORU



Obrázek 2.6: Schéma regulačního obvodu s modelem regulované soustavy .[16]

Blokové schéma je znázorněno na obr. 2.6. Akční veličina působí jak na regulovanou soustavu S , tak na model M . Rozdíl výstupů tvoří pomocnou akční veličinu $e_M = y(t) - y_M(t)$, kterou zpracovává regulátor R_M . Přenos takového řízení pak vypadá následovně:

$$F_w(p) = \frac{R_1(p)S(p) + R_1(p)S(p)R_M(p)M(p)}{1 + R_1(p)S(p) + S(p)R_M(p) + S(p)R_1(p)M(p)R_M(p)} \quad (2.12)$$

Hlavní výhodou této nové vazby je značná necitlivost kvality regulace na změny parametrů regulované soustavy. Tuto vlastnost nám zabezpečuje regulátor R_M , který vyrovnává rozdíly mezi soustavou a jejím modelem.[16].

2.5. Návrh regulátoru

Nastavování či seřizování regulátorů je jedna z nejdůležitějších oblastí průmyslové regulace. Správné odladění má vliv nejenom na kvalitu produktu (nepřekročení kritické teploty atd.), ale má pozitivní ekonomické důsledky. Oproti tomu praxe ukazuje, že většina regulátorů vyžaduje neustálý dohled operátorů. Mnohé z nich dokonce pracují trvale, nebo přerušovaně v manuálním režimu. [9]

2.5.1. Pokus-omyl

Nejběžnější způsob návrhu regulátorů v praxi spočívá v odborném odhadu a odladění najíždějího technika způsobem „pokus-omyl“. Ne vždy má technik dostatek času na správné odladění, a tak stačí, když technický soubor „funguje“ a drží se v předem stanovených mezích.

Matematicko-fyzikální analýza procesu není většinou možná z důvodu velké složitosti procesu, a experimentální identifikace (metodou černé skříňky) bývá příliš nákladná a v průmyslových podmínkách většinou dokonce neproveditelná. Naštěstí k tomu, abychom mohli navrhnout vyhovující řízení procesu, není nutné znát univerzální model procesu. Dlouholeté empirické zkušenosti potvrzují např., že pro návrh rozumného PI(D) regulátoru nám obvykle stačí znát dvě nebo tři charakteristická čísla procesu (dobu průtahu, dobu náběhu a statické zesílení). Podobné závěry lze učinit též na základě populární

Zieglerovy-Nicholsovy metody, která pro návrh PI(D) regulátorů využívá pouze dvě tzv. kritické hodnoty (kritické zesílení a kritickou frekvenci). [15]

2.5.2. Modifikované Ziegler-Nicholsovo kritérium

Původní metoda byla zveřejněna J.G. Zieglerem a N.B. Nicholsonem před více jak 70 lety, a je hojně používána až dodnes. Metoda vychází z uzavřeného regulačního obvodu a určení kritických parametrů soustavy. Praktický postup při nastavování parametrů pomocí metody Z-N probíhá následovně:

1. Vyřadíme integrační a derivační složku regulátoru ($T_D = 0$, $T_I \rightarrow \infty$).
2. Zvyšujeme zesílení proporcionální složky, dokud nedosáhneme ustálených kmitů na výstupu soustavy.
3. Zesílení K , při kterém jsme dostali netlumené kmity, označíme jako K_{KRIT} a periodu těchto kmitů jako T_{KRIT} .
4. Parametry regulátoru nastavíme podle tab. 2.1

Podmínkou použití této metody je, aby přenosová funkce procesu byla třetího a vyššího řádu, nebo aby měla dopravní zpoždění. U nižších řádů soustav nemůžeme určit kritické parametry, protože je nelze rozkmitat.

Tabulka 2.1: Nastavení parametrů podle Z-N bez omezení kmitavého průběhu

Typ regulátoru	K	T_I	T_D
PID	$0,6K_{KRIT}$	$0,5T_{KRIT}$	$0,125T_{KRIT}$
PI	$0,45K_{KRIT}$	$0,83T_{KRIT}$	-
P	$0,5K_{KRIT}$	-	-

Toto nastavení odpovídá odezvě na jednotkový skok žádané veličiny se třemi až čtyřmi viditelnými překmity. K omezení kmitavého průběhu můžeme použít experimentálně zjištěnou úpravu výpočtu konstant, pomocí které dostaneme přechodový děj s jedním až dvěma překmity.

Tabulka 2.2: Nastavení parametrů podle Z-N s omezením kmitavého průběhu[12]

Typ regulátoru	K	T_I	T_D
PID	$0,3K_{KRIT}$	T_{KRIT}	$0,125T_{KRIT}$
PI	$0,2K_{KRIT}$	T_{KRIT}	-
P	$0,25K_{KRIT}$	-	-

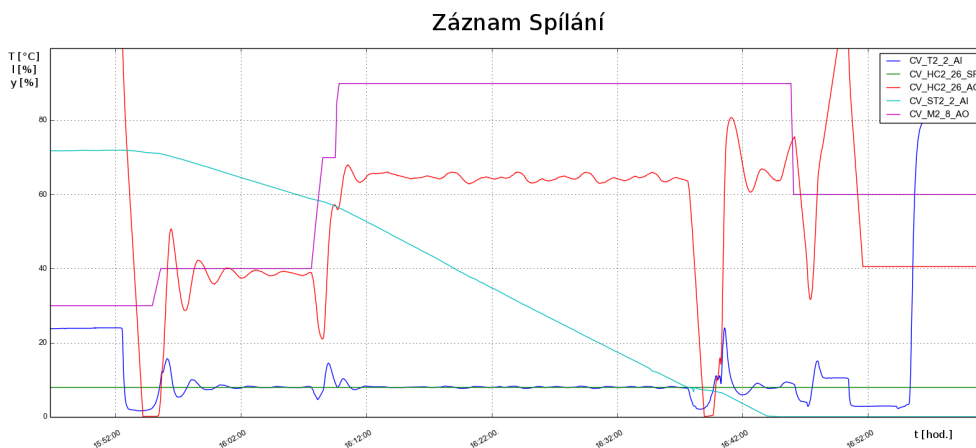
2.6. Adaptivní řídicí systémy

Převážná většina procesů řízených v průmyslu má proměnné dynamiky vstupních veličin. Změna parametrů procesu je způsobena změnami vlastností surovin, paliva, zařízení

2.6. ADAPTIVNÍ ŘÍDICÍ SYSTÉMY

(opotřebení) a změnami v provozních režimech, které budou později popsány na konkrétním příkladu. Klasické regulátory s pevně nastavenými parametry často nevyhovují řízení takovýchto procesů, neboť při změnách parametrů procesu není jejich průběh optimální, a dochází tak ke ztrátám energie, materiálu, snižování životnosti zařízení atd.

Uvedený příklad, jak málo stačí pro změnu dynamiky, znázorňuje obr. 2.7 (chlazení mladiny piva). Spílání je proces, kdy z vířivé kádě čerpáme mladinu do spilek. Ve vířivé kádě se po chmelovaru odstraní drobné nečistoty a zbytky chmele. Když se nečistoty usadí, má mladina stále přes 80 °C. Abychom nezničili kvasinky, musíme zchladit mladinu na společnou teplotu, která se pohybuje okolo 8 °C. Však kritické hranice jsou 4 – 12 °C, které nesmíme překročit.



Obrázek 2.7: Záznam chlazení mladiny na společnou teplotu 8 °C

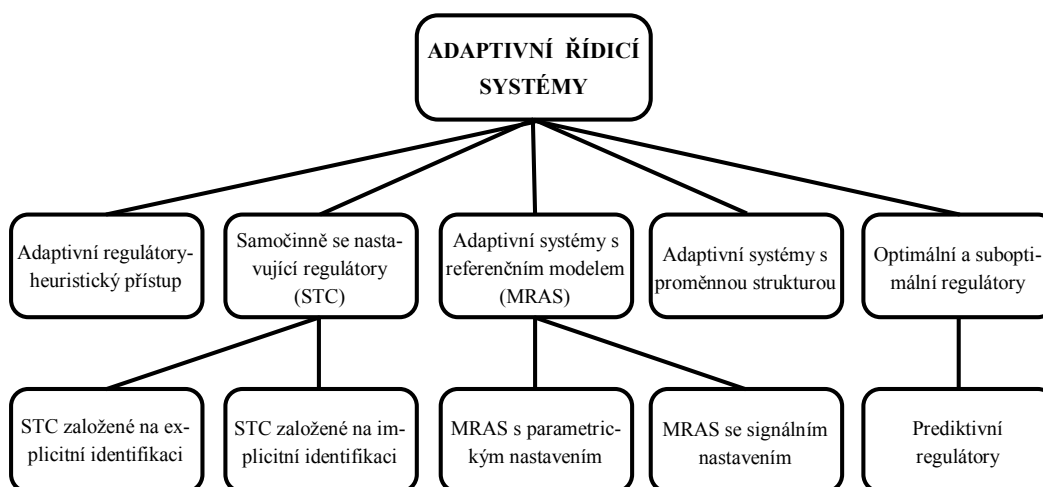
Na obr. 2.7 je bledě modrou barvou znázorněna hladina ve vířivé kádě, zelenou naše žádaná veličina, modrou měřená veličina (teplota mladiny), červenou akční zásah (úroveň otevření ventilu) a fialovou procenta výkonu společného čerpadla. Je zde patrné, jak při nižším průtoku je soustava silně kmitavá, a když je výkon čerpadla na 90%, tak je překmit výrazně menší. Další malé výchylky ukazují, kdy obsluha zvedla průtok vzduchu na provzdušnění mladiny. Následující velký překmit byl dle hladiny ve vířivé kádě způsoben evidentně manuálním zásahem.

Jedním z možných řešení tohoto problému je umožnit regulátorům, aby se mohli těmto změnám přizpůsobit. Jinými slovy, aby se dokázali adaptovat na změny. Základní dělení těchto regulátorů je na obr. 2.8.

2.6.1. Samočinně se nastavující regulátory (STC)

Tento přístup je založen na průběžném odhadování vlastností soustavy a poruch. Pomocí těchto dat následně provádí syntézu regulátoru. Adaptivní číslicový regulátor pracuje s pevně nastavenou periodou vzorkování. Dynamika akčního členu je zahrnuta do dynamiky soustavy stejně jako vliv vzorkování i případné filtry měřené veličiny. Co se týká parametrů soustavy, tak budeme uvažovat, že jejich vývoj v čase je pomalejší než proces adaptace. Filozofie regulátoru se dá popsat v následujících 3 krocích:

1. Vektor parametrů θ se pro daný krok považuje za známý a rovný bodovému odhadu, který je v daném okamžiku k k dispozici $\theta(k) = \hat{\theta}(k - 1)$.



Obrázek 2.8: Klasifikace adaptivních řídicích systémů.[11]

2. Za tohoto předpokladu se navrhne strategie řízení pro zvolené kritérium kvality řízení a vypočítá se podle něj potřebný akční zásah.
3. Po získání nové měřené hodnoty se provede další krok identifikace a provede se aktualizace odhadu $\hat{\theta}$. Celý algoritmus se opakuje.

Vnucenou separací identifikace a řízení se rozpadá vnitřní struktura regulátoru na identifikační a řídicí část, které spojuje pouze přenos bodových odhadů $\hat{\theta}(k)$. Správné fungování takového regulátoru zajistí pouze rychle konvergující a spolehlivá identifikace.

2.6.2. Výpočet kritických parametrů

Výpočet kritické periody kmitů závisí na umístění pólů na jednotkové kružnici v komplexní z rovině.

$$z = e^{j\omega T_0} = \cos \omega T_0 + j \sin \omega T_0 \quad (2.13)$$

Kritickou periodu kmitů pak můžeme vypočítat ze vztahů

$$\alpha = \cos T_0 \omega_K \quad (2.14a)$$

$$\omega_K = \frac{1}{T_0} \arccos \alpha \quad (2.14b)$$

$$T_K(T_0) = \frac{2\pi}{\omega_K} \quad (2.14c)$$

Výpočet kritického zesílení pro model prvního řádu

Reálný proces aproximujeme modelem prvního řádu. Dostaneme charakteristický polynom ve tvaru:

$$D(z) = z + a_1 + K_P b_1 \quad (2.15)$$

2.6. ADAPTIVNÍ ŘÍDICÍ SYSTÉMY

Potom bude polynomiální rovnice ve tvaru:

$$z + a_1 + K_P(T_0)b_1 = z + 1 \quad (2.16)$$

Pro kritické zesílení bude platit vztah:

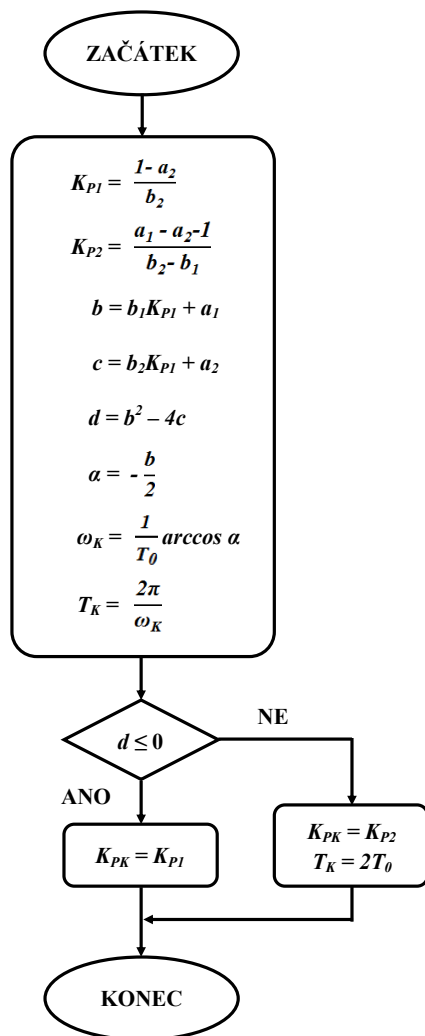
$$K_P(T_0) = \frac{1 - a_1}{b_1} \quad (2.17)$$

Výpočet kritického zesílení pro model druhého řádu

Pokud budeme mít soustavu s přenosem:

$$F(z) = \frac{a_1 z + a_2}{z^2 + b_1 z + b_2} \quad (2.18)$$

bude výpočet probíhat podle obr. 2.9



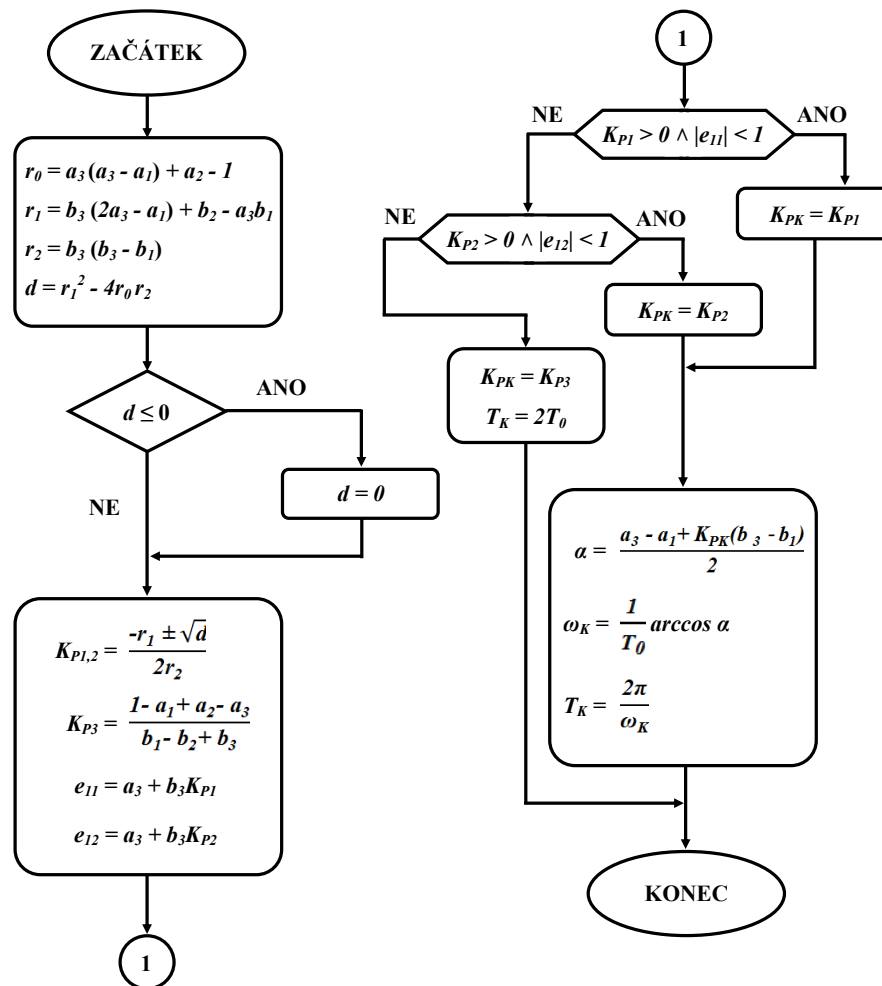
Obrázek 2.9

Výpočet kritického zesílení pro model třetího řádu

Zde budeme uvažovat přenos:

$$F(z) = \frac{a_1 z^2 + a_2 z + a_3}{z^3 + b_1 z^2 + b_2 z + b_3} \quad (2.19)$$

pak se hodnoty vypočítají dle následujícího schématu.



Obrázek 2.10

3. Identifikace dynamického systému

Technologické procesy, které chceme řídit, jsou většinou velmi složité, nelineární a proměnné v čase. Možnost, že získáme jejich přesný matematický model ve všech pracovních režimech v konečném čase, je nereálná. Jak bylo řečeno v sekci 2.5, pro návrh klasického PID regulátoru si ve většině případů vystačíme s dvěma parametry pro libovolnou soustavu, tak zde potřebujeme znát kompletní model s co nejmenší odchylkou od reálné soustavy. Pro optimální řízení procesu je tato část stejně důležitá jako vlastní syntéza regulátoru.

Pro přesnou identifikaci je třeba vhodně zvolit testovací signál a jeho amplitudu, odstranit trendy (jako např. stejnosměrnou složku), zvolit správný model odpovídající soustavě, a v neposlední řadě je třeba zvolit vhodný algoritmus výpočtu. Alternativní metodou pro získání dat k identifikaci může být dočasné řízení klasickým PID regulátorem, než se identifikované konstanty ustálí. Krajním řešením může být i dočasné řízení na ruku. Základní možnosti a postupy budou podrobněji probrány v následující kapitole.

3.1. Testovací signál

Než začneme uvažovat nad tím, jaký testovací signál použijeme, je třeba zvážit, jestli je vůbec takový experiment bezpečný. Některé soustavy jsou velmi citlivé, a dají se velmi lehko dostat do nestabilního stavu, nebo nesnesou ani relativně malé kmity. U systémů s dlouhými časovými konstantami je třeba počítat se značnou časovou náročností.

Volba testovacího signálu a jeho nastavení výrazně ovlivňují kvalitu identifikace. Nejčastěji používané typy jsou:

- skoková změna
- pseudonáhodná binární posloupnost
- součet harmonických průběhů

Testovací signál nikdy nejde jako žádaná veličina, ale jeví se jako výstup z regulátoru. Někdy se dává paralelně s regulátorem relé. To je aktivní pouze při identifikační fázi a generuje nám zdrojový signál. Další možnost je takový člen umístit do zpětné vazby. Nyní si trochu rozebereme jednotlivé signály.

3.1.1. Skoková změna

Je to dvoustavový signál, kdy do času t má nulovou úroveň, a v čase t přejde na úroveň u_0 . Funkce má jediný parametr, a to právě u_0 . To se volí s ohledem na:

- odstup signálu od šumu
- linearitu systému
- možnosti vybuzení

Touto metodou můžeme určit:

- statické zesílení
- dopravní zpoždění
- dobu průtahu / náběhu
- překmit (rezonanci)

3.1.2. Pseudonáhodná binární posloupnost

Jedná se opět o dvoustavový signál, který však vytváří určitou posloupnost. Velikost úrovně se volí obdobně jako u odezvy na jednotkový skok. Signál se generuje pomocí posuvného registru a sčítaček modulo 2. Jsou povoleny všechny stavy kromě samých nul. Maticový zápis takového generátoru vypadá následovně:

$$\mathbf{x}(k+1) = \begin{pmatrix} a_1 & a_2 & a_3 & \cdots & a_{n-1} & a_n \\ 1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 1 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & 0 \end{pmatrix} \mathbf{x}(k) \quad (3.1a)$$

$$u(k) = (0 \ 0 \ \cdots \ 1) \mathbf{x}(k) \quad (3.1b)$$

Jedná se zde o konečný stavový automat. Délka jedné periody se odvíjí od velikosti použitého registru. Pro správnou identifikaci statického zesílení je třeba, aby délka největšího impulsu T_{MAX} byla větší, než doba náběhu t_n . A na základě toho určíme i délku posloupnosti.

$$T_{MAX} = n \cdot T_{vz} < t_n \quad (3.2)$$

$$M = 2^n - 1 \quad (3.3)$$

Kde M vyjadřuje PRBS maximální délky a n počet posuvných registrů. Další podmínkou je, aby jsme postihli co nejvíce frekvencí. Pokud je délka experimentu L zadaná, musí platit:

$$M = 2^n - 1 < L \quad (3.4)$$

V případě nízkého odstupu signálu od šumu lze zlepšení identifikace dosáhnout prodloužením experimentu. [6]

3.1.3. Součet harmonických průběhů

Vstupní signál je určen následovně:

$$u(t) = \sum_{j=1}^m \sin(\omega_j t + \varphi_j) \quad (3.5)$$

Kde $0 \leq \omega \leq \pi$. Zde 0 odpovídá stejnosměrné složce, a π signálu, který v každém kroku mění své znaménko. Počet parametrů se odvíjí od počtu harmonických průběhů. Zde platí jednoduché pravidlo. Čím bohatší spektrum máme, tím jednodušší je identifikace. [6]

3.2. Modely pro identifikaci procesů

Cílem je nalézt model, který bude vhodný pro průběžnou identifikaci v reálném procesu. Zde si pro zjednodušení zavedeme operátor posunutí q , kde pro posunutí o jeden krok vzad platí:

$$q^{-1}u(k) = u(k-1) \quad (3.6)$$

Pak lze obecný model procesu zapsat jako:

$$A(q^{-1})y(k) = \frac{B(q^{-1})}{F(q^{-1})}u(k) + \frac{C(q^{-1})}{D(q^{-1})}e(k) \quad (3.7)$$

Pokud zahrneme dopravní zpoždění, bude model vypadat následovně:

$$A(q^{-1})y(k) = q^{-n_k} \frac{B(q^{-1})}{F(q^{-1})}u(k) + \frac{C(q^{-1})}{D(q^{-1})}e(k) \quad (3.8)$$

Tento model je však příliš obecný, proto některé polynomy položíme rovno 1. Nakonec i podle principu Occamovi břitvy je nejjednodušší možný model ten správný.

3.2.1. ARX

Budeme mít diferenční rovnici ve tvaru:

$$A(q^{-1})y(k) = B(q^{-1})u(k) + e(k) \quad (3.9)$$

Kde $e(k)$ představuje bílý šum. AR zde představuje Auto Regresivní část $A(q^{-1})y(k)$ a X je část s externím vstupem $B(q^{-1})u(k)$. Za těchto podmínek a pomocí následující lineární regrese lze dosáhnout:

$$\theta = (a_1 \ a_2 \ \dots \ a_n \ b_1 \ b_2 \ \dots \ b_n)^T \quad (3.10)$$

$$\varphi(k) = (-y(k-1) \ \dots \ -y(k-a_n) \ u(k-1) \ \dots \ u(k-b_n))^T \quad (3.11)$$

$$\hat{y}(k|\theta) = \varphi^T \theta + e(k) \quad (3.12)$$

Na obr. 3.1 je blokové schéma tohoto modelu.

V praxi tento model nenajdeme, ale přesto se používá, protože vede na lineární regresi. [4]

3.2.2. ARMAX

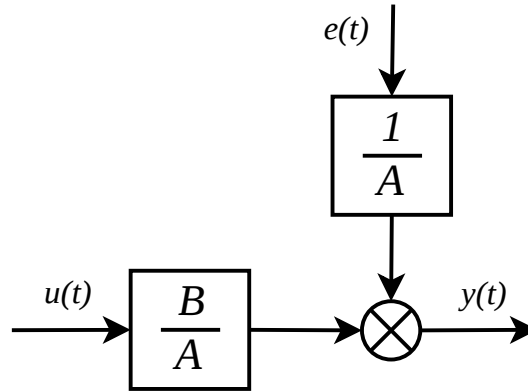
Jedná se o model ve tvaru:

$$A(q^{-1})y(k) = B(q^{-1})u(k) + C(q^{-1})e(k) \quad (3.13)$$

Kde $C(q^{-1})e(k)$ přidává část MA, tedy vážený průměr (moving average). Zde pak nahradíme $e(k)$ chybou predikce $\varepsilon(k|\theta) = y(k) - \hat{y}(k|\theta)$ a dostáváme:

$$\hat{y}(k|\theta) = \frac{B(q^{-1})}{C(q^{-1})}u(k) + \left[1 - \frac{A(q^{-1})}{C(q^{-1})}\right]y(k) \quad (3.14)$$

3. IDENTIFIKACE DYNAMICKÉHO SYSTÉMU



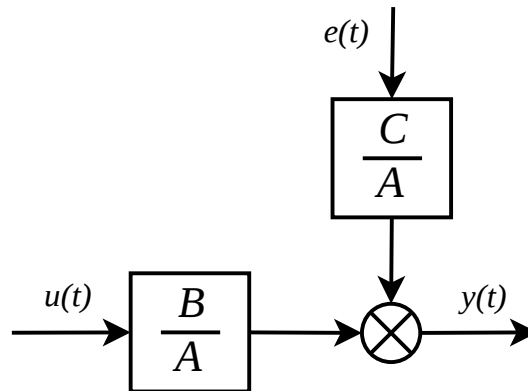
Obrázek 3.1: Model ARX

Predikci získáme vyfiltrováním vstupu a výstupu filtrem, jehož jmenovatelem je $C(q^{-1})$. Vektor pozorování pak vypadá následovně:

$$\varphi(k, \theta) = (-y(k-1) \quad \dots \quad -y(k-a_n) \quad u(k-1) \quad \dots \quad u(k-b_n) \quad \varepsilon(k-1|\theta) \quad \dots \quad \varepsilon(k-b_n|\theta))^T \quad (3.15)$$

Vektor odhadnutých parametrů má následující tvar:

$$\theta = (a_1 \quad \dots \quad a_n \quad b_1 \quad \dots \quad b_n \quad c_1 \quad \dots \quad c_n)^T \quad (3.16)$$



Obrázek 3.2: Blokové schéma modelu ARMAX

3.3. Volba řádu modelu

Volba správného modelu je klíčová pro správnou identifikaci modelu. Pro správné zvolení je třeba znát soustavu, kterou se snažíme řídit, a vědět jak se chová. Dále musíme vždy zvážit jaké nástroje máme k dispozici. V ideálním případě, kdy máme k dispozici nekonečně mnoho hodnot, existuje model M^* , který přesně popisuje chování systému.

3.4. IDENTIFIKAČNÍ METODY

Další zvyšování řádu modelu systému nevede ke zlepšování kritéria. Jako kritérium lze použít:

$$R_{ee}(0) = Ee(k)e(k) = \sum_{k=1}^N e^2(k) \quad (3.17)$$

V praxi se obvykle volí model odpovídající $R_{ee_x}(0)$ pro který platí:

$$0,8R_{ee_x}(0) \leq R_{ee_{x+1}}(0) \quad (3.18)$$

kde $R_{ee_{x+1}}(0)$ odpovídá modelu s o jedničku větším počtem parametrů. [5]

Model druhého řádu můžeme popsat rovnicí:

$$F_C(z) = \frac{Y(z)}{U(z)} = \mathcal{Z} \left\{ \mathcal{L}^{-1} \left\{ \frac{1 - e^{-Tp}}{p} \frac{K}{(T_1p + 1)(T_2p + 1)} \right\} \right\} = \frac{b_1z^{-1} + b_2z^{-2}}{1 + a_1z^{-1} + a_2z^{-2}} \quad (3.19)$$

Model druhého řádu s dopravním zpožděním $\tau < T$:

$$F_C(z) = \frac{Y(z)}{U(z)} = \mathcal{Z} \left\{ \mathcal{L}^{-1} \left\{ \frac{1 - e^{-Tp}}{p} \frac{Ke^{-\tau p}}{(T_1p + 1)(T_2p + 1)} \right\} \right\} = \frac{b_1z^{-1} + b_2z^{-2} + b_3z^{-3}}{1 + a_1z^{-1} + a_2z^{-2}} \quad (3.20)$$

Pro model třetího řádu lze použít následující rovnici:

$$F_C(z) = \frac{Y(z)}{U(z)} = \mathcal{Z} \left\{ \mathcal{L}^{-1} \left\{ \frac{1 - e^{-Tp}}{p} \frac{K}{(T_1p + 1)(T_2p + 1)(T_3p + 1)} \right\} \right\} = \frac{b_1z^{-1} + b_2z^{-2} + b_3z^{-3}}{1 + a_1z^{-1} + a_2z^{-2} + a_3z^{-3}} \quad (3.21)$$

Jak přesně určit velikost DZ bude rozebráno v sekci 4.1

3.4. Identifikační metody

Pro identifikaci soustav existuje celá řada metod:

- Neparаметrické metody - Analýza přechodové charakteristiky, Frekvenční analýza...
- Parametrické metody - Identifikace modely MA, ARX...
 - Offline (Jednorázové) - MNČ
 - Online - rekurzivní verze offline metod + vylepšení

Neparаметrické metody jsou dobré, pokud se jedná o soustavu nižšího řádu a je nám umožněno otestovat soustavu např. na jednotkový skok, a velmi rychle nalézt hledané

konstanty. To nám však zpravidla nikdo v praxi nedovolí. Buď to neumožňuje technologie, nebo je pokus jednoduše moc drahý. Navíc technologové se velmi zdráhají, aby se na jejich technologických souborech prováděli jakýkoliv experimenty.

Offline metody se nám hodí v případě, že máme historická data, a systém na nich byl dostatečně vybudovaný, to bývá případ nájezdů technologie, ale zejména poruch a neočekávaných stavů.

V adaptivních řídicích systémech využíváme rekurzivních parametrických metod, které umožní nejenom přizpůsobovat řízení změnám v regulované soustavě, ale zároveň můžeme pomocí analýzy změn konstant systému identifikovat závady na zařízení.

3.4.1. Metoda nejmenších čtverců

Jedná se o nejpoužívanější metodu identifikace. Snažíme se pomocí lineární regrese minimalizovat kvadrát odchylky. Ve vektorovém zápisu lze tuto regrese zapsat následovně:

$$y(t) = \phi^T(k)\theta \quad (3.22)$$

Kde $y(t)$ představuje měřenou veličinu, ϕ sloupcový vektor známých veličin a θ sloupcový vektor hledaných parametrů. Chybu měření lze pak formulovat následovně:

$$\varepsilon(k) = y(k) - \phi^T(k)\theta \quad (3.23)$$

Metoda nejmenších čtverců vychází z minimalizace následující ztrátové funkce:

$$J(\theta) = \frac{1}{2} \sum_{k=1}^N [y(k) - \phi^T(k)\theta]^2 = \frac{1}{2} (Y - \Phi\theta)^T (Y - \Phi\theta) \quad (3.24)$$

Když položíme derivaci ztrátové funkce podle vektoru θ , dostaneme výslednou rovnici pro výpočet neznámých parametrů.

$$\theta = (\theta^T\theta)^{-1}\theta^TY \quad (3.25)$$

Matice $(\theta^T\theta)^{-1}$ nesmí být singulární, ale většinou však bude pozitivně definitní. Pro další výpočty si označíme $P(k) = (\theta^T\theta)^{-1}$, kde $P(k)$ je kovarianční matice.

3.4.2. Průběžná metoda nejmenších čtverců

Pokud chceme identifikovat systém za chodu, musíme zavést rekurzivní výpočet. Cílem rekurzivního výpočtu je do stávajícího rozměru matice ϕ začlenit další měření.

$$K(k+1) = \frac{P(k)\varphi(k)}{1 + \varphi^T(k+1)P(k)\varphi(k+1)} \quad (3.26a)$$

$$\hat{\theta}(k+1) = \hat{\theta}(k) + K(k+1)\hat{e}(k+1|k) \quad (3.26b)$$

$$P(k+1) = P(k) - K(k+1)\varphi^T(k+1)P(k) \quad (3.26c)$$

$$\hat{e}(k+1|k) = y(k+1) - \varphi^T(k+1)\hat{\theta}(k) \quad (3.26d)$$

3.4. IDENTIFIKAČNÍ METODY

První rovnice vyjadřuje míru shody mezi procesem a jeho modelem. Matice $K(k)$ určuje jakou rychlostí bude probíhat vývoj nových parametrů. Druhá rovnice určuje odhad parametrů pro následující krok. Veličina $\hat{e}(k+1|k)$ představuje apriorní chybu predikce závisující na rozdílu mezi skutečným a odhadovaným výstupem. Kovarianční matice $P(k)$ je v podstatě akumulátor informací o minulých hodnotách vstupů a výstupů. Prvky na její hlavní diagonále vyjadřují rozptyl mezi skutečnými a odhadovanými parametry soustavy.

Počáteční podmínky volíme následovně:

$$P(0) = 10^9 I \quad (3.27)$$

$$\hat{\theta}(0) = (1, 0, 0, \dots, 0)^T \quad (3.28)$$

Jestliže zvolíme např. model druhého řádu následovně:

$$\varphi^T = [u(k-1), u(k-2), -y(k-1), -y(k-2)] \quad (3.29)$$

Pak dostaneme odhad parametrů ve stavu:

$$\theta = (\hat{b}_1, \hat{b}_2, \hat{a}_1, \hat{a}_2)^T \quad (3.30)$$

3.4.3. Konstantní exponenciální zapomínání

Tato metoda je vhodná pro určení plynule se měnících parametrů. Budeme předpokládat, že nová data vystihují proces lépe, než stará. Po té je dobré zavést postupné zapomínání veškeré staré informace.

$$\hat{e}(k+1|k) = y(k+1) - \varphi^T(k+1)\hat{\theta}(k) \quad (3.31a)$$

$$K(k+1) = \frac{P(k)\varphi(k)}{\lambda_e + \varphi^T(k+1)P(k)\varphi(k+1)} \quad (3.31b)$$

$$\hat{\theta}(k+1) = \hat{\theta}(k) + K(k+1)\hat{e}(k+1|k) \quad (3.31c)$$

$$P(k+1) = \frac{1}{\lambda_e} [P(k) - K(k+1)\varphi^T(k+1)P(k)] \quad (3.31d)$$

Kde λ_e je faktor exponenciálního zapomínání, který může nabývat hodnot $0 < \lambda_e \leq 1$. Pro $\lambda_e = 1$ se algoritmus zjednoduší na metodu nejmenších čtverců, a všechna získaná data pak mají stejnou váhu. Menší hodnoty koeficientu λ_e vedou k rychlejší adaptaci, ale na úkor přesnosti a nechtěného zvýšení citlivosti algoritmu na působení šumu. Obvykle se koeficient zapomínání volí v rozsahu $0.995 \leq \lambda_e \leq 0.999$.

3.4.4. Proměnné exponenciální zapomínání

Pokud se řízený systém nachází delší dobu v ustáleném stavu, může se stát, že vlastní čísla kovarianční matice $P^{-1}(k)$ budou vést k nule a zesílení $P(k)\varphi(k)$ nabývat nekonečných hodnot. Tomuto jevu se také říká estimator windup. Nejjednodušší způsob zamezení tomuto jevu je zavést proměnné exponenciální zapomínání, kde při začátku identifikace

3. IDENTIFIKACE DYNAMICKÉHO SYSTÉMU

necháme relativně malé λ_e , a postupně ho začneme zvyšovat. Předchozí rovnice výpočtu se nezmění, pouze přidáme následující:

$$\lambda_e(k) = \lambda_0 \lambda_e(k-1) + (1 - \lambda_0) \quad (3.32)$$

Počáteční hodnoty zde volíme následovně $\lambda_0 = 0.99$ a $\lambda_e(0) = 0.95$ [3]

4. Dopravní zpoždění

Jak již bylo řečeno, zavedením diskrétních řídicích systémů zavedeme vždy i dopravní zpoždění. Pro systémy s časovými konstantami značně většími jak DZ, můžeme toto DZ úplně zanedbat. To však nejde dělat v systémech, kde je DZ dominantní. To jsou systémy kde proudí různá média, nebo dokonce kde se k dávkování používají dopravníky. Obecně vždy trvá nějakou dobu, než se náš produkt dostane od akčního členu k měřicímu čidlu. Tím pádem se nám zpozdí vazba mezi akčním zásahem a projevenou změnou na tento zásah. Na přechodové charakteristice se to projeví posunem odezvy od akčního zásahu. To přináší často velké problémy při regulaci. Čím je DZ větší, tím se stává soustava více nestabilní a hůře regulovatelná.

Existuje několik možností, jak tento problém řešit. U klasických regulátorů je nutné držet velmi nízké zesílení proporcionální složky, aby zůstala soustava stabilní. U adaptivních regulátorů je třeba zvolit vhodný model soustavy se zahrnutým DZ. Pokud do modelu nezahrneme DZ, a budeme se snažit aproximovat soustavu modelem vyššího řádu, dostaneme sice model který se bude blížit naší soustavě, ale její dynamika bude odlišná od fyzické soustavy. Další možností je použít rozvětvený obvod typu Smithova prediktoru, který je podrobněji popsán v sekci 4.3.

4.1. Identifikace

Existuje zde opět několik možných způsobů. Nejjednodušší způsob je tuto hodnotu přímo odečíst z přechodové charakteristiky. Pokud to není možné, máme k dispozici následující postup. Pomocí rekurzivní MNC určíme číselný polynom.

$$\hat{B}(q^{-1}) = \hat{b}_1 q^{-1} + \hat{b}_2 q^{-2} + \hat{b}_3 q^{-3} + \dots \quad (4.1)$$

Pokud vyjde $|\hat{b}_1| < 0.15|\hat{b}_2|$, zvýšíme hodnotu dopravního zpoždění o 1 a $|\hat{b}_1| = 0$. Po úpravě modelu provedeme znovu identifikaci.[4]

4.2. Simulace

V prostředí MATLAB lze implementovat dopravní zpoždění několika typů:

- *InputDelay* - Dopravní zpoždění na vstupu systému
- *OutputDelay* - Dopravní zpoždění na výstupu systému
- *ioDelay* - Matici pro definování nezávislých vazeb DZ pro jednotlivý pár I/O
- *InternalDelay* - U SS systému umožňuje sledovat celkové DZ po kombinování více systémů.

Systém s dopravním zpožděním pak lze modelovat několika způsoby:

- Klasicky pomocí laplacovy transformace.
 $F = \exp(-2*s*p)*3/(5*p+1)$
- Pro SISO systém pomocí funkce vytvářející model *tf*, *ss*, *drss*, atd..
 $F = tf(3,[5,1], 'OutputDelay',2);$
popř. $F.OutputDelay = 2$
- Pro MIMO systém pomocí matice.
 $G.ioDelay = [2; 3 -2]$
- Pomocí funkčního bloku *Transport Delay*, který funguje jako shift registr.
- Pomocí bloku Z^{-1} , pokud simulujeme dopravní zpoždění v řádech několika vzorků.

V modelech diskrétního času jsou tyto parametry vztaženy na celočíselný násobek vzorkovací periody. Pro modelování systému s kratším dopravním zpožděním, než vzorkovací frekvence, lze použít funkce *thiran*.

U bloku *Transport Delay* lze manuálně nastavit velikost posuvného registru. U této možnosti je třeba dbát, aby velikost byla větší než počet vzorků potřebných na zachycení dopravního zpoždění při nastaveném kroku simulace. Pokud je krok simulace větší než nastavené dopravní zpoždění, může docházet ke zkreslení výsledku, protože simulační software musí výsledek aproximovat. Pokud naopak chceme přesnou hodnotu na konci simulačního děje, musíme simulaci prodloužit. Zde totiž software dopočítává hodnoty v čase $\langle t_k - DZ; t_k \rangle$, kde t_k je čas konce simulace, pomocí extrapolace. [10]

Aproximaci pomocí padeho rozvoje je sice možná, ale nedoporučuje se. Vždy totiž použijeme rozvoj nižšího řádu, a dostaneme tak do modelu dynamiku, která v reálné soustavě není.

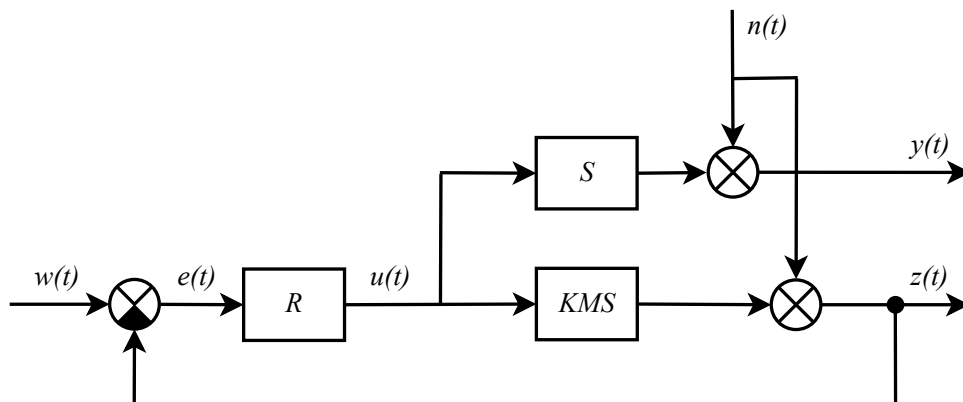
4.3. Smithův prediktor

Je vhodný zejména v případě, kdy řízený systém obsahuje dominantní zpoždění, nebo je fázově neminimální. Kvalitu řízení takového systému lze podstatně zlepšit, pokud doplníme PID regulátor kladnou zpětnou vazbou, která obsahuje model řízeného systému. Smithův prediktor je speciální případ rozvětveného obvodu s modelem regulované soustavy, který kompenzuje přítomnost dopravního zpoždění v regulované soustavě. Jeho blokové schéma je znázorněno na obr. 4.2

Hlavní myšlenka spočívá ve vypuštění dopravního zpoždění z regulační smyčky. Vytvoříme zpětnou vazbu s kompenzovaným modelem soustavy, ve kterém vypustíme dopravní zpoždění, a akční zásah regulátoru využijeme pro regulaci původní soustavy. Toto schéma je zobrazeno na obr. 4.1. Tento princip slouží pouze jako ověření teorie. Není totiž možné měřit všechny poruchy a mít dokonalý model.

Budeme uvažovat, že v soustavě je článek s dopravním zpožděním Δ . Model obsahuje stejný článek, ale máme k dispozici nezpožděný výstup z modelu. Důsledkem toho je, že nám DZ nebude zasahovat do stability systému jak je patrné z následné rovnice přenosu:

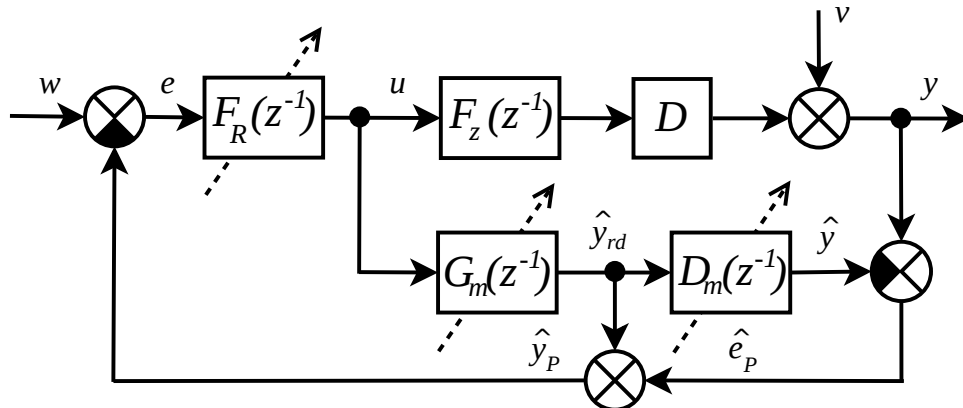
4.3. SMITHŮV PREDIKTOR



Obrázek 4.1: Regulační obvod s kompenzovaným modelem soustavy[15]

$$F_w = \frac{R(p)S(p)e^{-\Delta p}}{1 + R(p)S(p)} \quad (4.2)$$

Budeme tedy navrhovat PID regulátor na upravený model bez dopravního zpoždění, který už je dobře regulovatelný. Pokud ho navrheme optimálně na tento model, pak jsme ho navrhli optimálně na naši soustavu.



Obrázek 4.2: Smithův prediktor

Kde

F_R ... PID regulátor.

F_z ... Soustava bez DZ.

D ... Dopravní zpoždění.

G_M ... Kompenzovaný model.

D_M ... Zjištěné DZ.

5. Návrh a realizace řídicího systému

Jako první myšlenka se nabízí vše kompletně integrovat do PLC. To však není u všech běžně používaných automatů realizovatelné. Buď nemají potřebný výpočetní výkon, nebo narazíme na problém s pamětí. Můžeme narazit na případ, kde má jedno PLC nižší řady řídit zároveň několik technologických celků. V neposlední řadě nemusí programovací jazyk pro PLC umožňovat samotný návrh komplexnějšího algoritmu.

Pro různé experimenty můžeme použít simulační software, který propojíme s reálnou soustavou. Takovým příkladem může být např. SIMULINK, ve kterém máme vlastní výpočet, a PLC nebo samotnou periférii využívanou pouze pro interakci s fyzickým světem. Tato varianta je však vhodná pouze pro vývoj a testování nových algoritmů. Pro praxi je moc nákladná, výkonově moc náročná a málo robustní.

Existuje však střední cesta, kdy pro vyčítání dat použijeme mezivrstvu. Tedy nejlépe univerzální OPC server, který má za účel jednoduše umožnit komunikaci s PLC po různých protokolech. Můžeme pak různé výpočty provádět na řídicím počítači, což má svoje úskalí, ale na trhu se již začínají objevovat PLC, na kterých běží plnohodnotný Windows 7. Tím odpadá i riziko, že nám někdo přeruší komunikační kabel, obsluha vypne PC atd. Nejsme tedy limitováni možnostmi simulačního softwaru, ani vázání na jistý druh PLC. To nám umožní implementovat navržený řídicí algoritmus v našem oblíbeném jazyce, a dál ho používat pro libovolnou technologii.

5.1. TomPack

Systém TomPack je souborem programů pro vizualizaci technologických procesů typu SCADA (Supervisory Control And Data Acquisition). Je určen jak pro malé systémy s jedním počítačem, tak pro rozsáhlé systémy mnoha počítačů propojených sítí včetně Internetu. Umožňuje redundanci jednotlivých částí, tj. při výpadku jednoho PC lze technologii řídit pomocí záložního počítače. Architektura TomPacku je typu klient/server, jednotlivé části mohou běžet na různých počítačích, které jsou spojeny sítí. TomPack má podporu pro řadu řídicích systémů - Siemens, ABB, Rockwell, Saia, Teco atd. Navíc má široké možnosti propojení se systémy třetích stran - komunikace DDE, OPC, ODBC, ADO, COM atd.[13]

5.1.1. Struktura TomPacku

Samotný Systém TomPack je složen ze čtyř programů – TPConfig, TPServer, TPView a UniServer. Jedná se o několikvrstvou architekturu klient-server. Tím je umožněno mnoho různých konfigurací jak pro malé, tak i pro velké projekty. Tyto programy mají následující funkci:

UniServer – Shromažďuje hodnoty proměnných v PLC a případně do nich zapisuje, slouží jako převodník mezi protokolem řídicího systému a protokolem DDE, OPC apod.

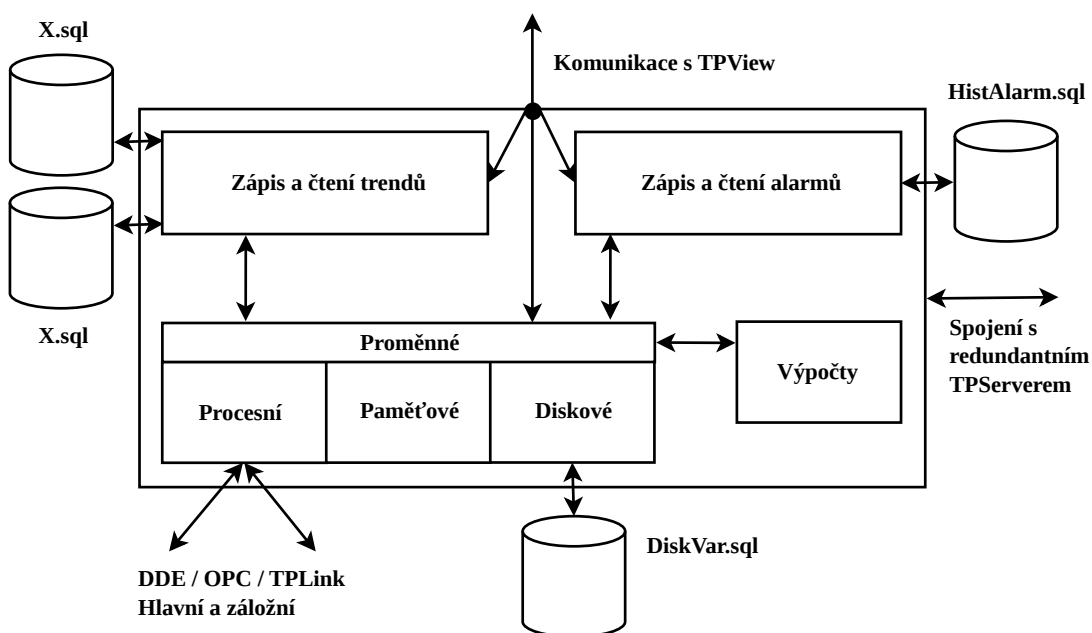
5.1. TOMPACK

TPServer – Hlavní výkonná část TomPacku. Stará se o vyčítání potřebných hodnot z UniServeru, kontroluje alarmy, zapisuje do trendových a alarmových souborů, vykonává skripty.

TPView - Vlastní vizualizační část TomPacku. Až tento program zobrazuje veškeré prvky na obrazovce a slouží k ovládání technologie.

TPConfig – Slouží pro editaci projektu, který poté posílá programům TPServer a TPView.

Další užitečnou vlastností tohoto systému je knihovna **TPCom**, která umožňuje komunikovat s TPserverem. Jedná se zde o dynamický objekt, který lze propojit s libovným programovacím jazykem, který podporuje načtení COM objektů. Připojení pomocí takového objektu se pro TPServer jeví jako další klient.[13]



Obrázek 5.1: Struktura TomPack serveru.[13]

Tato architektura nám umožňuje libovolně diverzifikovat náš řídicí systém.

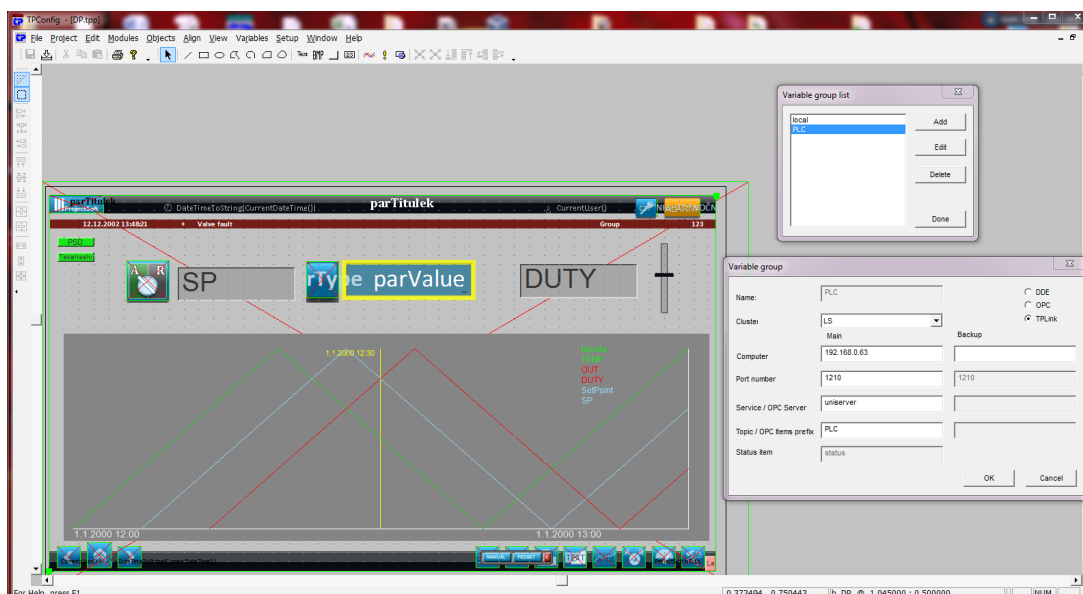
5.1.2. Postup realizace

Můžeme zde využít dvou postupů. První případ je kdy v PLC máme pouze přepisy proměnných do DB oblasti. Druhým případem pak může být, kdy máme celý řídicí systém v PLC a externě provádíme adaptaci proměnných. V obou případech musíme spustit Uniserver, na kterém nakonfigurujeme protokol příslušný danému PLC.

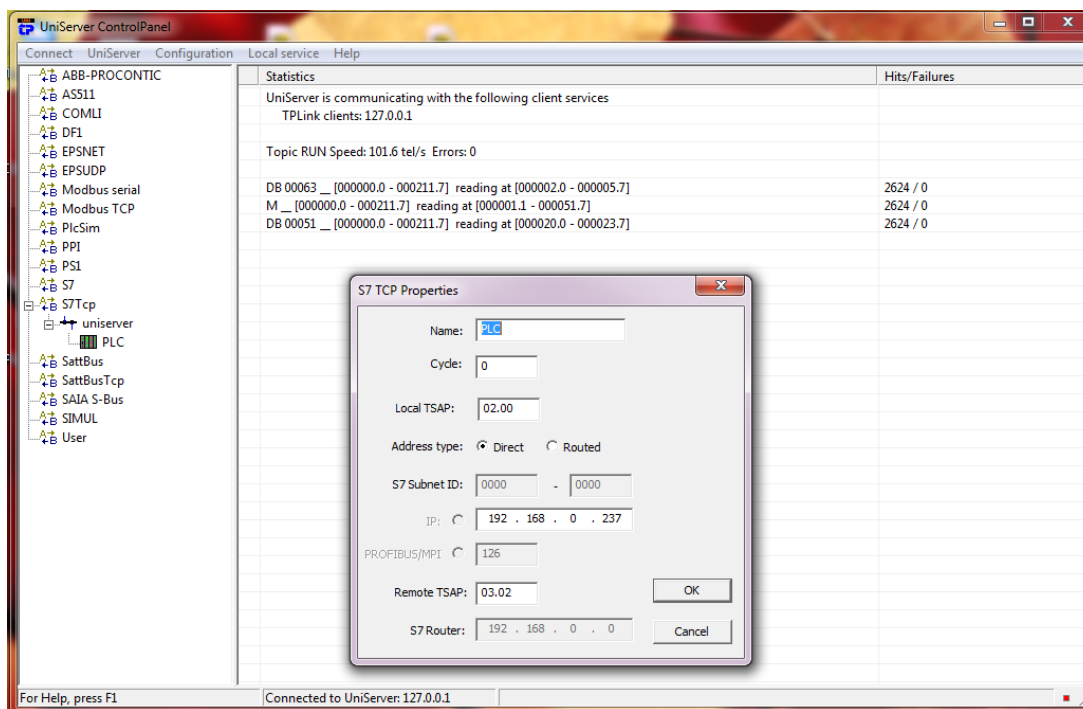
V TPConfigu vytvoříme nový projekt, ve kterém namapujeme proměnné na adresy podle projektu v PLC a nastavíme adresu clusteru (adresa stroje, kde běží TPServer s Uniserverem). Tento projekt odešleme do našeho clusteru. V tuto chvíli by měl TPServer odeslat požadavek na Uniserver a ten potom dále na PLC. Pokud vše proběhlo správně, neměli bychom na Uniserveru vidět žádné červené značky. Že nám vše funguje správně

5. NÁVRH A REALIZACE ŘÍDICÍHO SYSTÉMU

můžeme ověřit pomocí I/O modulu zasláného do TPView.



Obrázek 5.2: Obrazovka TPCConfig.

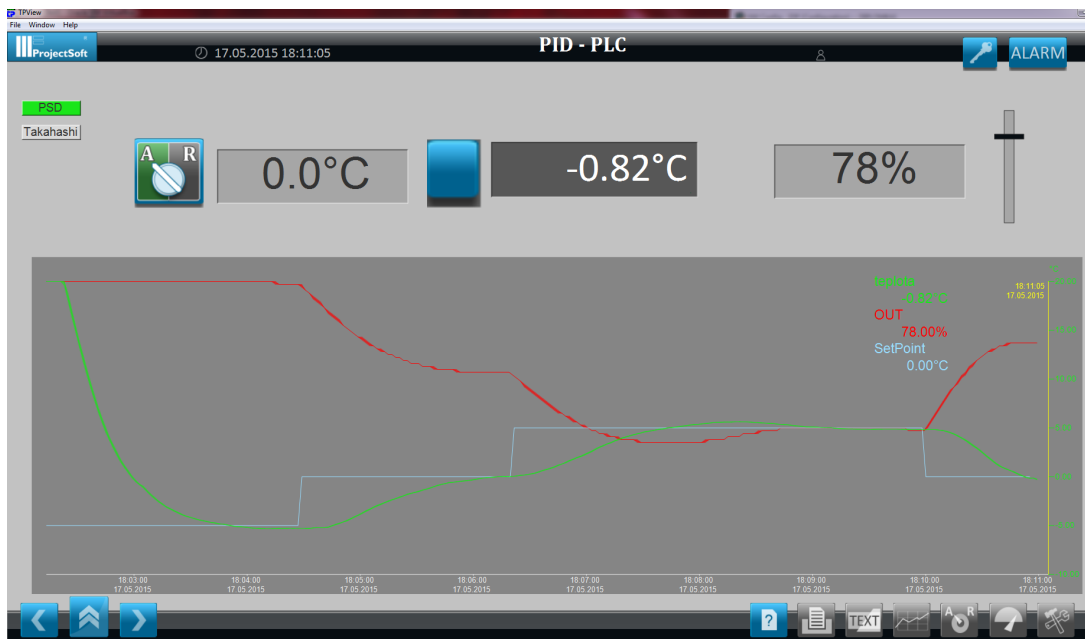


Obrázek 5.3: Ukázka nastavení Uniserveru.

V tuto chvíli můžeme vyčítat a zapisovat do našich proměnných podle jejich jména pomocí zmíněného objektu TPCom.

Na obr.5.4 je vidět obrazovka TPView, neboli náš real-time klient pomocí kterého můžeme jak zobrazit stav našeho systému, ale v případě nouze i pomocí něho zasáhnout.

5.2. MATLAB



Obrázek 5.4: Ukázka funkčního spojení s PLC.

5.2. MATLAB

Matlab je výpočetní a skriptovací systém určený pro rychlé prototypování, simulaci a analýzu různých systémů. K tomu všemu nám umožňuje jednoduše načíst registrovaný COM objekt pomocí funkce `actxserver()`.

Komunikovanou proměnnou můžeme importovat pomocí následující funkce.

```
1 function [ e ] = Variable(value, server, topic )
2 global init
3
4 e = actxserver('TPCom211.TPValue');
5 e.Address(topic, server, '', value);
6 if ~init
7     e.LogFile('TPLog.log')
8     init = 1;
9 end
10 end
```

Zdrojový kód 5.1: Funkce Variable

S takto definovanou proměnnou se pak pracuje velice intuitivně viz. následující příklad. Ovšem než se vytvoří komunikační kanál k TPServeru je zapotřebí nějaký čas. Do doby než se naváže spojení je hodnota proměnné `NaN` a funkce `.Valid()` vrací hodnotu `False`.

```
1 R = Variable('R', '127.0.0.1', 'Topic1');
2 disp(R.value) % read
3 R.value = 5.2; % write
```

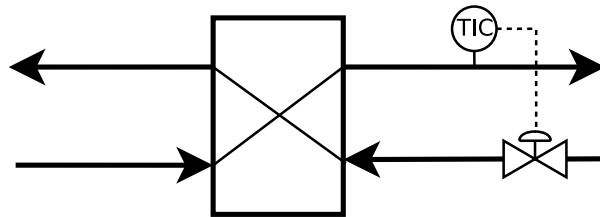
Zdrojový kód 5.2: Ukázka práce s komunikovanou proměnnou

Pokud se spojení nenaváže do několika vteřin, tak se provede zápis do logu o této události.

Do prostředí Simulink se dá tato proměnná načíst jako jakákoliv jiná proměnná z MATLABu. V této práci bylo využito globálních proměnných v m-funkcích.

5.3. Fyzická soustava

Velmi častá regulační úloha v praxi je regulace teploty za tepelným výměníkem. Principiální schéma je znázorněno na obr. 5.5. Ze schématu je vidět pozice měřícího a akčního členu. Vzhledem k tomu, že akční od změny akčního zásahu musí médium urazit nějakou cestu, vyměnit teplo s produktem, a produkt musí následně urazit cestu potrubím k měřicímu členu, tak tu máme nezanedbatelné dopravní zpoždění.



Obrázek 5.5: Regulační schéma s tepelným výměníkem

Tato úloha vypadá zdánlivě jednoduše pokud máme konstantní podmínky a ideálně se snažíme pouze doregulovat regulační odchylku. Problematické začíná být, pokud v průběhu regulace se nám skokově mění rychlost proudění produktu, nebo vlastnosti technologického média. Příkladem takových změn může být kolísání teploty chladicího média, nebo kolísání tlaku páry. To může nastat pokud kotel nemá dostatečný výkon, nebo dostatečně rychle nestihne vykompenzovat zvětšený odběr.

V následujících sekcích bude rozebrán návrh a postupná realizace funkčního jednoduchého modelu tepelného výměníku. Základem tohoto modelu je peltierův článek. Peltierův jev je reverzní jev k jevu Seebeckovu. Elektrony přecházejí z materiálu s větší výstupní prací do materiálu s menší výstupní prací a v místě styku je jejich pohybu kladen menší odpor. Elektrony zde mají přebytek kinetické energie a tu uvolňují v podobě tepla. [7]

Článek má teplou a studenou stranu. Která bude teplá a která studená určí směr proudu. Aby se článek nepřehřál je nutnou teplou stranu neustále chladit. Výhodou toho článku je jeho velikost a to, že se skládá pouze z pevných částí.

5.3.1. Akční člen

Jako akční člen se v praxi pro tento typ úlohy používají různé škrtící členy. Nejčastěji se používají regulační ventily, které omezují průtok technologického média. V našem modelu je akčním členem výkonový tranzistor, který reguluje pomocí pulsně šířkové modulace (PWM) výkon článku TEC1-12708. Tento článek byl zvolen po jeho nízkou cenu v poměru k jeho výkonu.

5.3. FYZICKÁ SOUSTAVA

Tabulka 5.1: Parametry článku TEC1-12708

ΔT_{MAX}	66 °C
I_{MAX}	8.5 A
V_{MAX}	15.4 V
Q_{MAX}	71 W

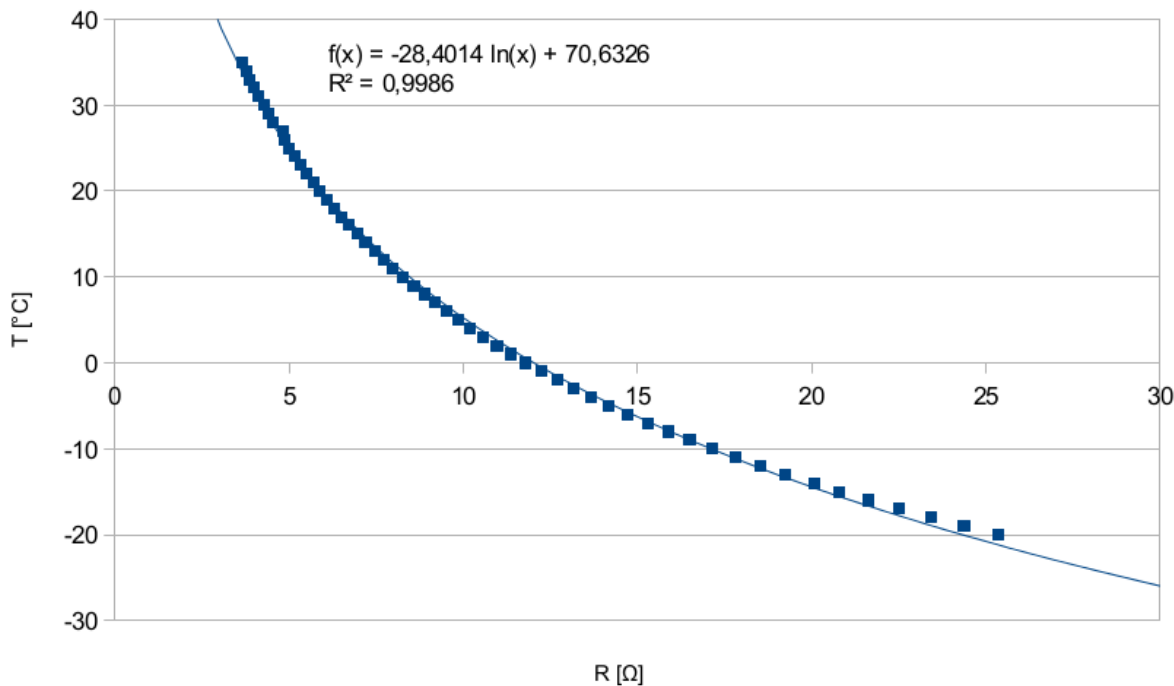
5.3.2. Měřicí člen

Pro měření teploty byl zvolen termistor NTC 5D-11. Toto označení říká, že při teplotě $T = 25\text{ °C}$ má odpor $R = 5\ \Omega$. NTC termistory mají záporný teplotní součinitel odporu. Teplotní rozsahy termistorů se pohybují od -50 °C do 150 °C . Závislost odporu NTC termistoru na teplotě je dána přibližným vztahem:

$$R = Ae^{\frac{B}{T}} \quad (5.1)$$

Kde $B[K]$ je teplotní konstanta závislá na materiálu a $A[\Omega]$ je konstanta závislá na tvaru a materiálu. Výhodou tohoto čidla je jsou velmi malé rozměry. Nevýhodou v porovnání s Pt senzory je však menší stabilita a nelineární závislost. [7]

Teplotní charakteristika snímače NTC 5D-11



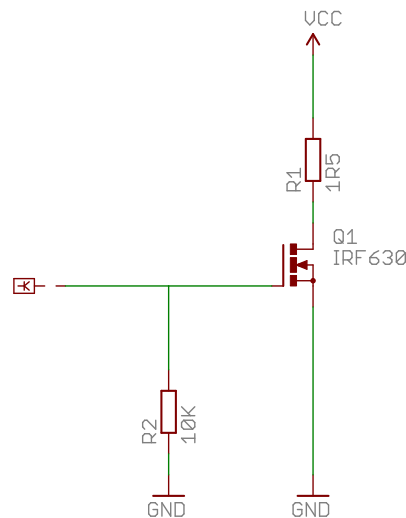
Obrázek 5.6: Převodní charakteristika termistoru 5D-11.

Funkce získaná z aproximace naměřených zobrazená na obr. 5.6 byla implementována do PLC pro přímý výčet teploty z DB. RTD karta má rozlišení 15 bit a byla nastavena na rozsah $150\ \Omega$. To znamená, že měříme s přesností $4.58\ \text{mA}$ což je pro toto měření dostatečně přesné.

5.3.3. Realizace

Celý systém je postaven na technologii SIEMENS S7-300, konkrétně IM 151-3 PN osazeným PM modulem, DI a RTD kartou. Měřená data se z PLC vyčítají pomocí protokolu S7TCP přes Uniserver do TPServeru, udkud má MATLAB možnost přistoupit na tyto data.

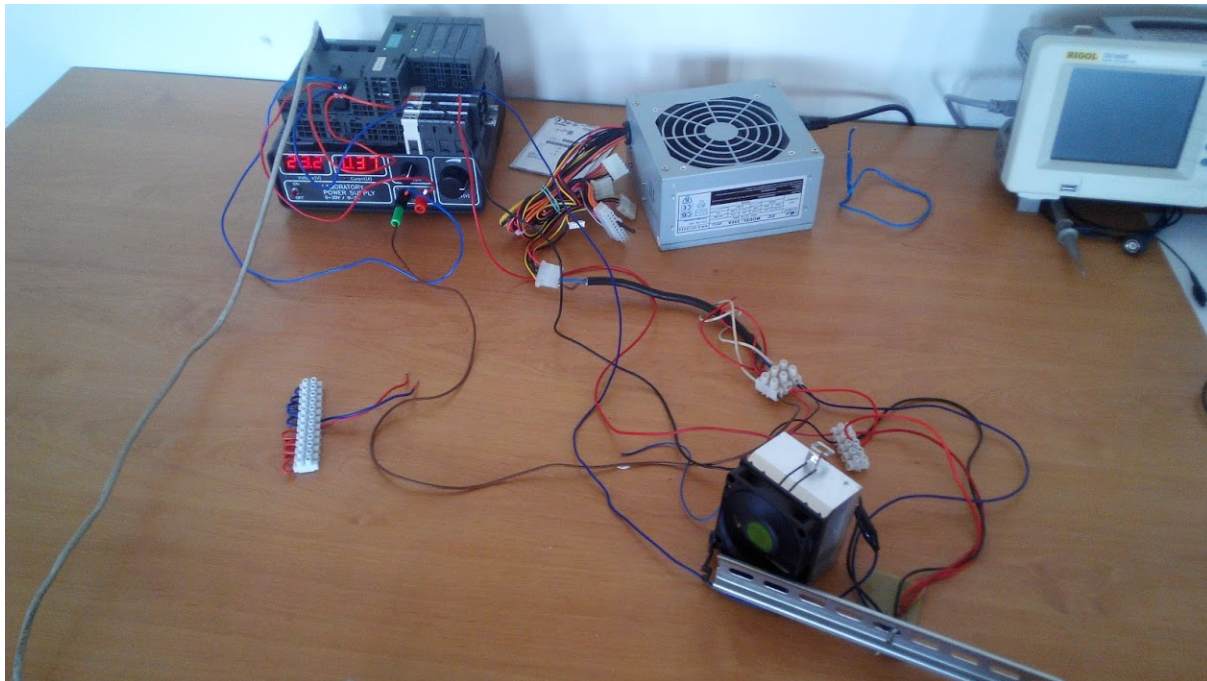
Peltierův článek je zapojen přes N-MOSFET, který v tomto zapojení umožňuje pomocí PWM signálu regulovat výkon článku. Zapojení je znázorněno na obr. 5.7. R1 Představuje náš peltierův článek a R2 pull-down rezistor, který drží tranzistor rozpojený pokud nepřichází log.1



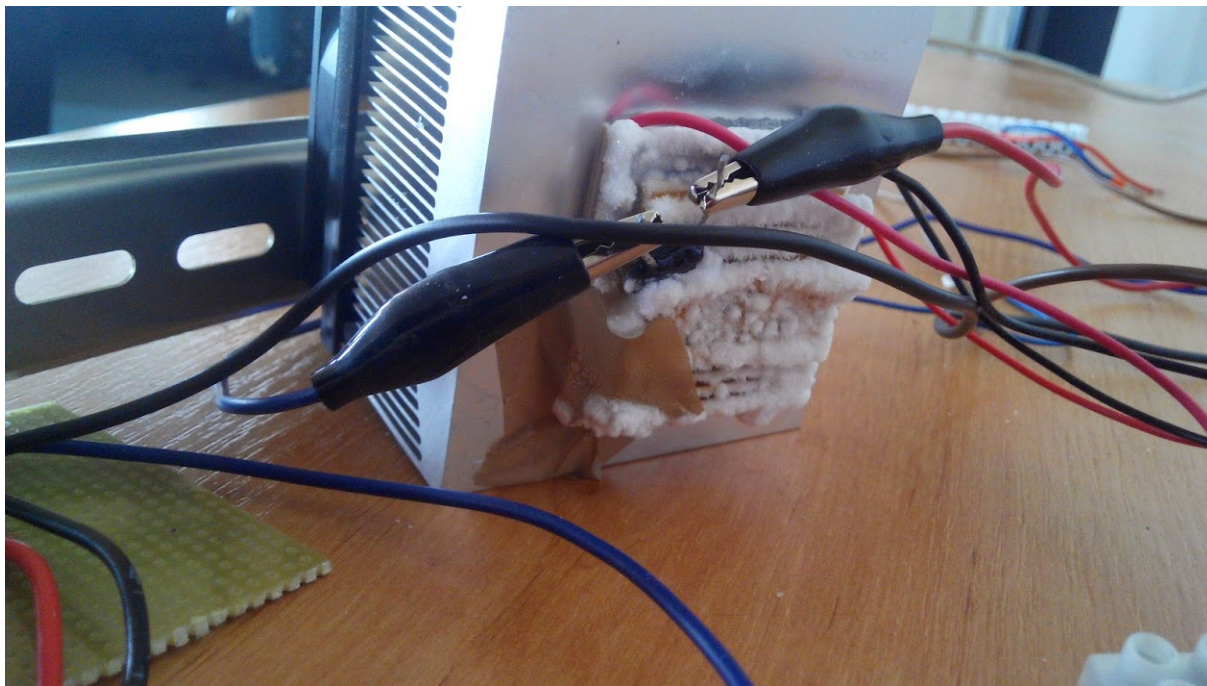
Obrázek 5.7: Zapojení zkčního členu.

Na následujících fotografiích je vidět praktická realizace soustavy. Na první fotografii je vidět celkové rozložení přístrojů. Zde je využito dvou napájecích zdrojů. Laboratorní zdroj pro napájení PLC a měřících karet. ATX zdroj zde napájí pouze peltierův článek. Ten je možno vidět podrobněji na obr. 5.9. Zde schovaný na druhé straně aktivního chladiče. DIN lišta v popředí se ukázala být výborným pasivním chladičem pro MOSFET řídicí výkon akčního členu.

5.3. FYZICKÁ SOUSTAVA



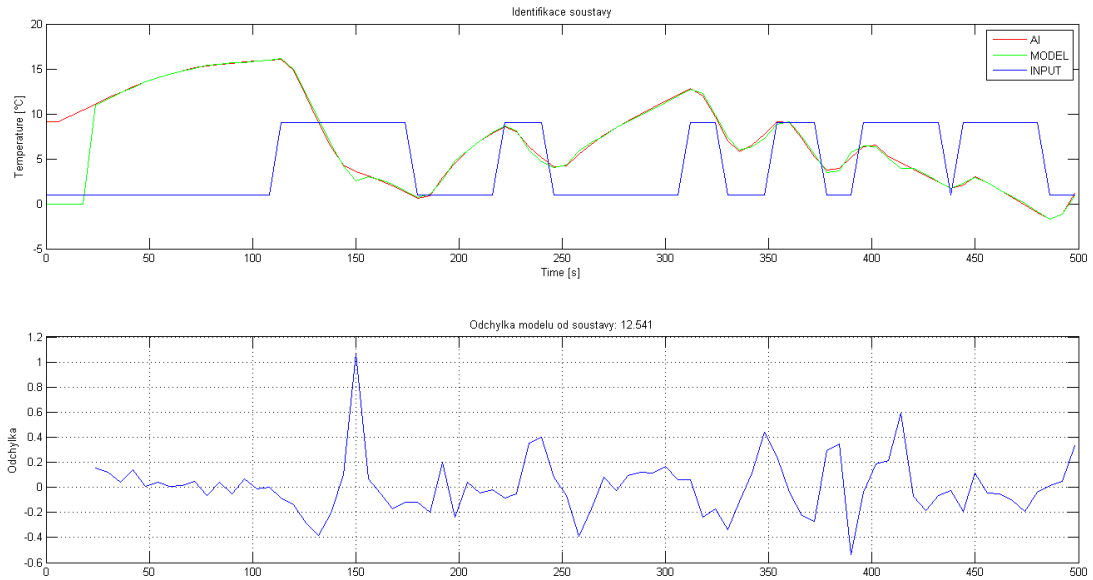
Obrázek 5.8: Fotografie fyzické soustavy.



Obrázek 5.9: Detail uchycení akčního a měřícího členu.

5.3.4. Model soustavy

Model soustavy vznikl identifikací identifikací metodou nejmenších čtverců, kde pro testování byl použit signál PRBS. Následnými pokusy byla nejnižší vhodná vzorkovací frekvence zvolena $T_{vz} = 6$ s.



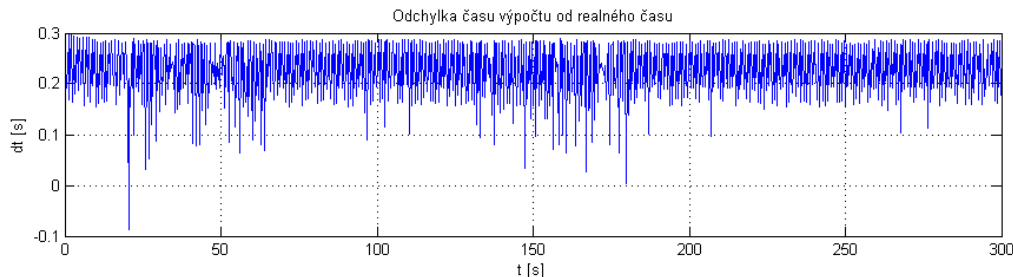
Obrázek 5.10: Porovnání modelu s fyzickou soustavou.

Jako model byl zvolen systém druhého řádu s pěti nulami. Vektor Φ vypadá následovně $\Phi(k) = [u(k); u(k-1); u(k-2); u(k-3); u(k-4); -y(k-1); -y(k-2)]$. Vektor parametrů soustavy pak vyšel následovně $\Theta = [0.0010; -0.0148; -0.0056; 0.0161; 0.0031; -1.8238; 0.8239]$.

Tento model se ukázal být velmi přesný při porovnání s reálnou soustavou, však komplikovaný pro implementaci do následujících algoritmů.

6. Ověření teorie

Pro simulování reálného času v Simulinku byl využit blok OPC configuration, který má i možnost záznamu odchylky času při výpočtu od reálného. Tento záznam je zobrazen na obr. 6.1, a představuje další dopravní zpoždění pro náš systém.



Obrázek 6.1: Odchylka času výpočtu od reálného času.

6.1. Simulace

V této části se nacházejí odezvy regulátorů na různé systémy. Jedná se vždy o systém druhého řádu, kterému měníme dopravní zpoždění a periodu vzorkování. Tyto hodnoty jsou v grafu znázorněny hodnotami d a Tvz .

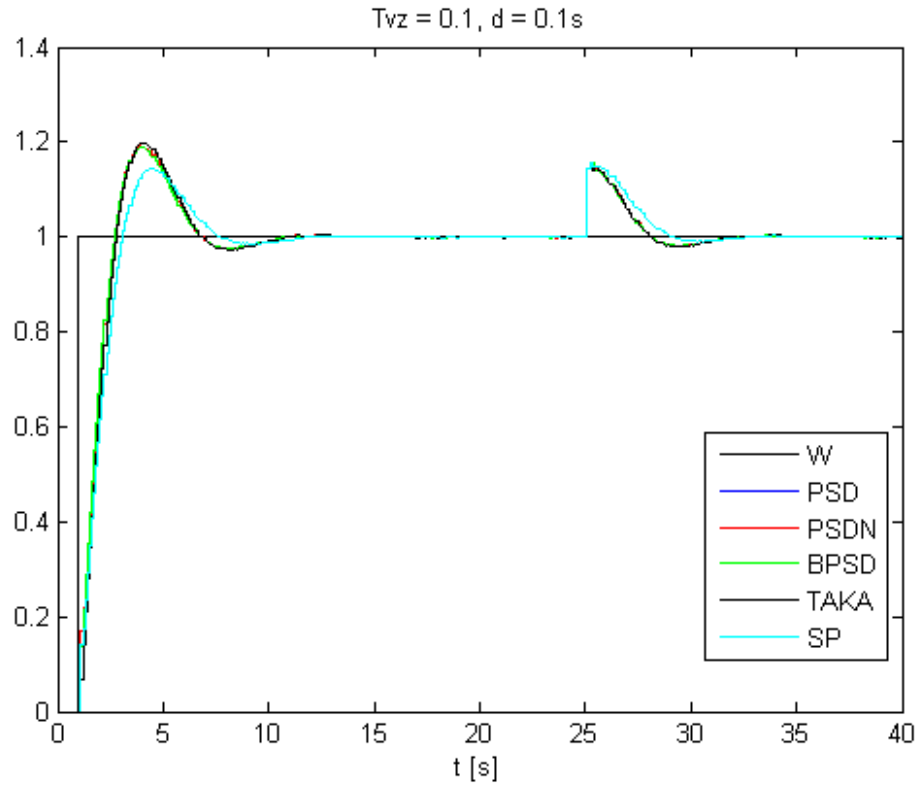
Parametry regulátorů byli vypočteny podle Z-N s omezením kmitavého průběhu. Žádaná hodnota je v grafech znázorněna písmenem W. V kroku 25 s se přidá porucha o velikost 0, 2.

Na prvním grafu je vidět soustava s minimálním DZ. Žádný regulátor nemá problém vyregulovat odchylku, nebo reakci na poruchu.

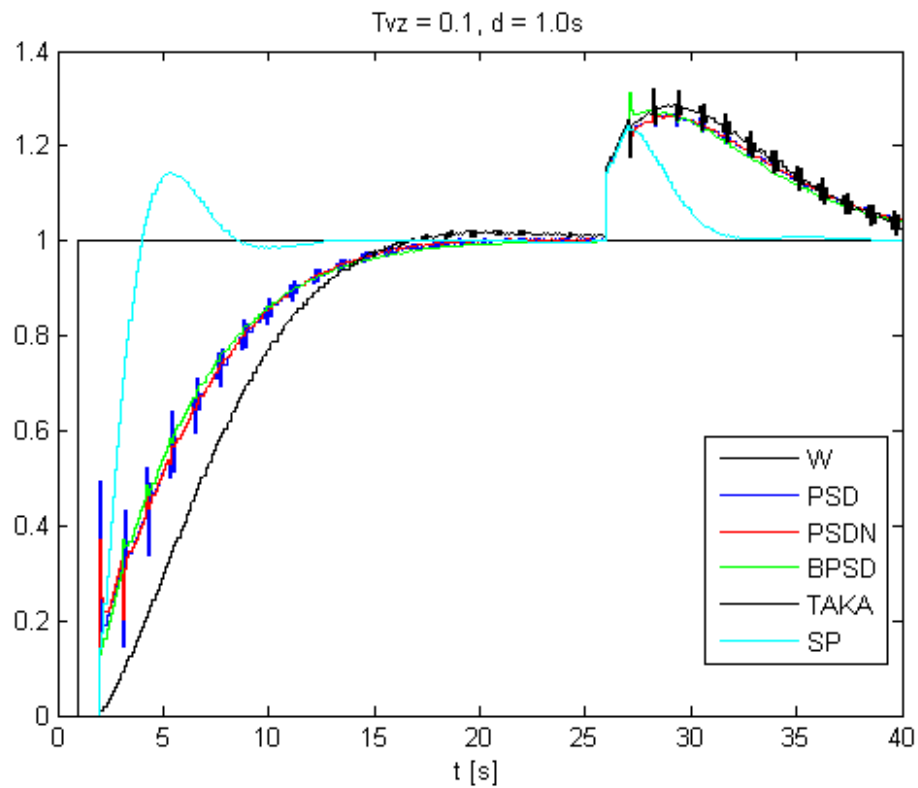
Na dalším grafu můžeme pozorovat, jak většina algoritmů se dostává do problémů způsobenými kvantizačním šumem, DZ a rychlou periodou vzorkování. Zde opět nejlepší výsledek dává Smithův prediktor. Takahashiho regulátor se také rozkmitá po přidání poruchy. Nejodolnějším regulátorem proti rozkmitání bez modelu soustavy je zde BPSD regulátor. Tyto kmity se dají ošetřit redukováním vlivu derivační složky, nebo prodloužením periody vzorkování.

Na obr. 6.6 je vidět stejná soustava jako na obr. 6.3, ale s větší periodou vzorkování. Žádný z regulátorů již nekmitá, ale na druhou stranu se zhoršila schopnost vyregulovat poruchu.

Na posledním grafu je vidět reakci na systém kde je DZ značně dominantní a malá perioda vzorkování. Dle očekávání nemá smithův prediktor s touto soustavou problém. Ovšem Takahashiho a PSD regulátor bez filtrace derivační složky přešli do nestabilní oblasti.

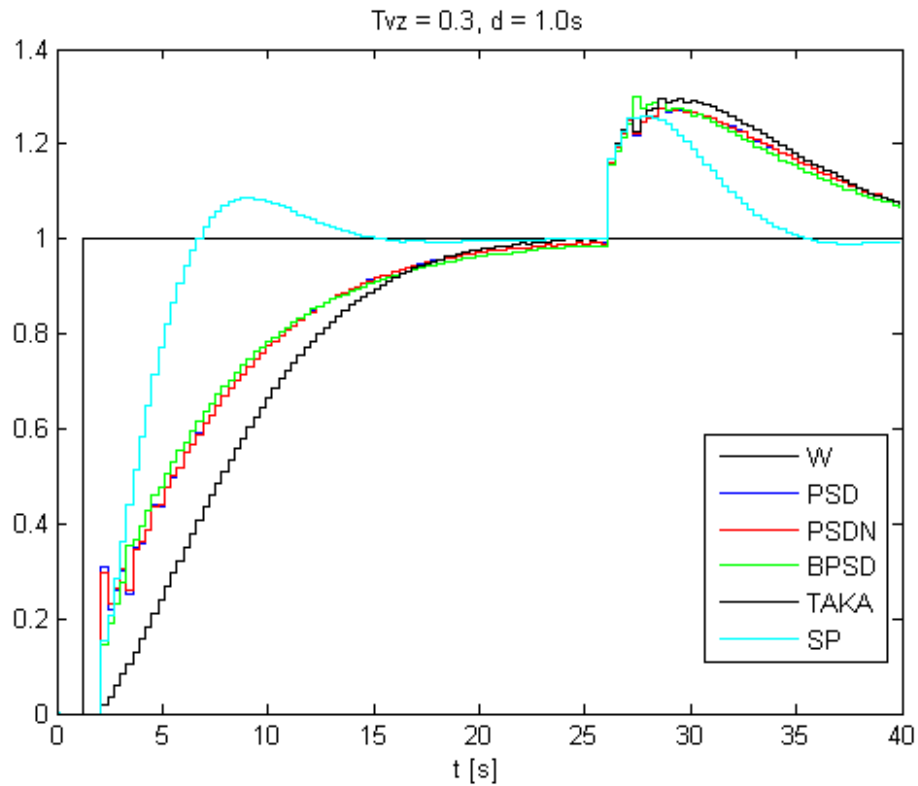


Obrázek 6.2: Soustava druhého řádu se zanedbatelným DZ.

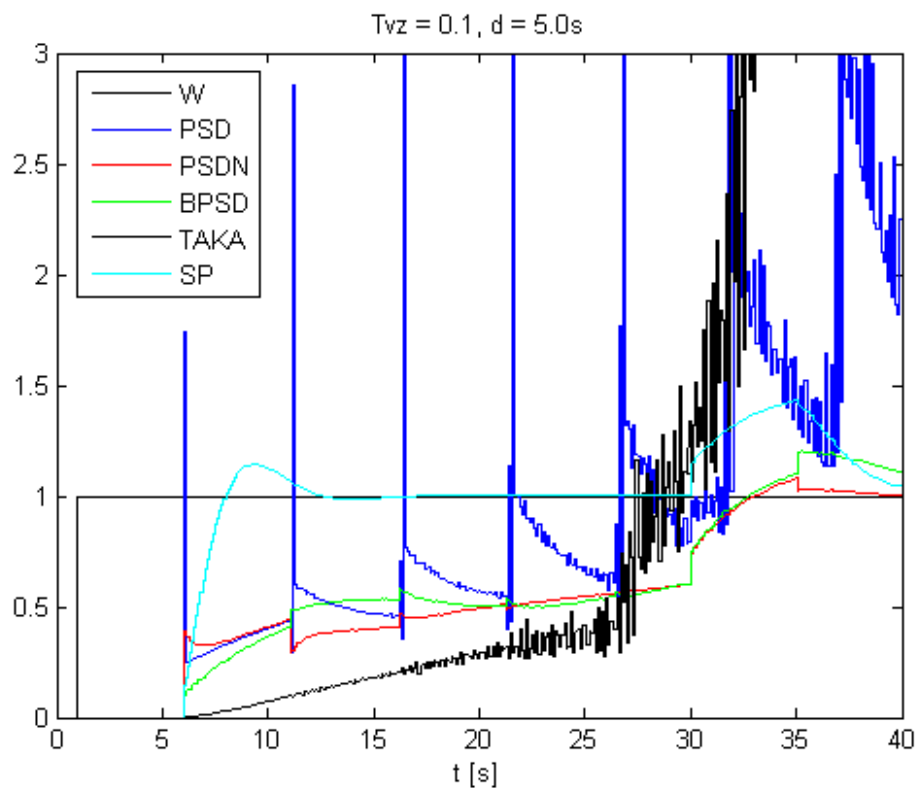


Obrázek 6.3: Soustava druhého řádu s dominantním DZ.

6.1. SIMULACE



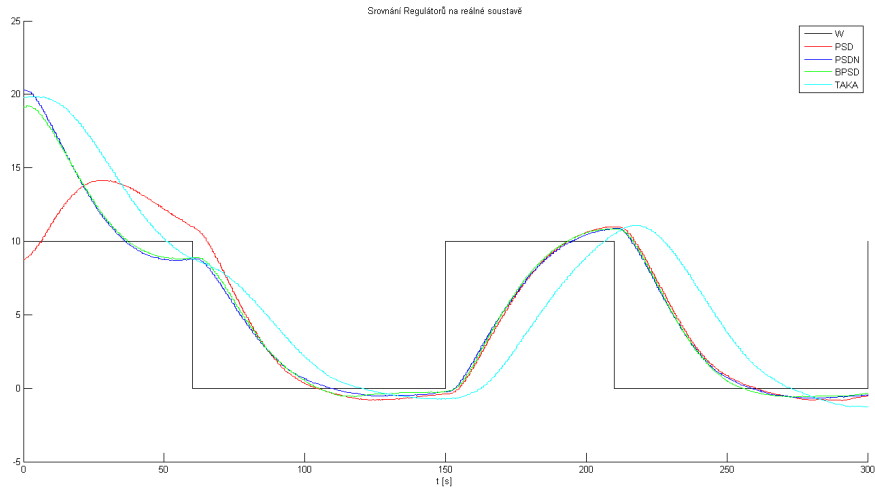
Obrázek 6.4: Soustava druhého řádu s dominantním DZ a prodlouženou T_{vz} .



Obrázek 6.5: Soustava druhého řádu s výrazně dominantním DZ.

6.2. Fyzická soustava

Tato část má sloužit k ověření poznatků ze simulace na reálném modelu. Žádný z regulátorů zde nevykazuje deviace pozorované v simulacích. To bude zapříčiněno velkou setrvačností naší soustavy.



Obrázek 6.6: Srovnání regulátorů na reálné soustavě.

7. Závěr

V této práci byli uvedeny návrhy a postupy k zefektivnění návrhu regulátorů, které by měli usnadnit uzavírání regulační smyčky, usnadnit samotné nastavování parametrů a v neposlední řadě přiblížit se optimálnímu regulačnímu průběhu u těžce regulovatelných soustav.

V první kapitole se nejdříve rozebírají různé vlivy, které ovlivňují všechny diskrétní formy regulátorů. Jsou zde uvedeny nejčastější formy PID regulátorů, jejich přínos do kvality řízení, a nejčastější způsob jejich nastavování. Dále jsou zde uvedeny Adaptivní řídicí systémy, jejich rozdělení, a způsob adaptace těchto regulátorů.

Druhá kapitola se zabývá identifikací dynamických systémů. Je zde rozebrán celý postup procesu identifikace, a to od použití správného signálu přes vhodnou volbu modelu procesu až po výběr identifikační metody. To je klíčová část pro navržení optimálního adaptivního regulátoru. Znalosti získané těmito metodami mohou být velmi cenné i u pevně nastavitelných regulátorů. Zejména ve chvíli, kdy pracujeme s citlivou soustavou a není možné klasickými metodami takovýto regulátor nastavit. Nebo když hrozí, že soustava během experimentu přejde do nestabilní oblasti.

Ve třetí kapitole se rozebírá dopravní zpoždění se všemi jeho dopady a důsledky na regulační děj. Je zde uvedeno, jak lze DZ stanovit, nebo postupně identifikovat pomocí úpravy čitatele polynomu. V této práci jsou také popsány možnosti, jak takovýto děj simulovat. Ne vždy totiž dosáhneme stejných výsledků.

Ve čtvrté kapitole je uvedeno několik možných realizací řídicího systému. Podrobněji je pak popsáno univerzální řešení s OPC serverem. Dále je zde uveden návrh a realizace malého modelu výměníku pro testování různých postupů při návrhu řídicího systému.

V poslední kapitole je jsou uvedeny výsledky ze simulací a praktického měření. Nejlépe si vedl dle očekávání smithův prediktor, a nejhůře PSD regulátor bez filtrace derivační složky.

Tato práce je doplněna i o praktické poznatky z průmyslové výroby.

Tato práce je velmi časově náročná, ovšem z důvodu plánování moc malých časových rezerv je práce tímto hodně poznamenaná.

Literatura

- [1] ALFA LAVAL; SPIRAX SACRO: *Steam and Plate Heat Exchangers. A technical guide to the effective use of steam in plate heat exchangers*. ALFA LAVAL and SPIRAX SACRO, 2006.
- [2] BARTOŇ, M.: *Univerzální mikroprocesorový regulátor*. Diplomová práce, VUT Brno, 2012.
- [3] BLAHA, P.: *Rekurzivní metoda nejmenších čtverců*. Prezentace, VUT Brno, 2010.
- [4] BLAHA, P.: *Modely lineárních systémů*. Prezentace, VUT Brno, 2012.
- [5] BLAHA, P.: *Metoda nejmenších čtverců*. Prezentace, VUT Brno, 2013.
- [6] BLAHA, P.: *Vstupní signály pro identifikaci systémů*. Prezentace, VUT Brno, 2014.
- [7] ĎAĎO, S.: *Senzory a převodníky*. Skriptum, ČVUT v Praze, 2006.
- [8] GAJDŮŠKOVÁ, M.: *Praktické aspekty implementace jednoduchých číslicových regulátorů*. Diplomová práce, UTB ve Zlíně, 2012.
- [9] LAY, G.; SCHIRRMEISTER, E.: *Sackgasse Hochautomatisierung? Praxis des Abbaus von Overengineering in der Produktion. Zpráva z průzkumů inovací ve výrobě č.22*. Karlsruhe, Fraunhofer ISI, 2001.
- [10] MATHWORKS: MATLAB Documentation. 12 2014.
URL www.mathworks.com/help
- [11] PIVOŇKA, P.: *Optimalizace regulátorů*. Skriptum, VUT Brno, 2005.
- [12] PIVOŇKA, P.: *Číslicová řídicí technika*. Skriptum, VUT Brno, 2012.
- [13] PROJECTSOFT: *TomPack*. Uživatelská příručka, 2006.
- [14] RUŠÁR, L.: *Návrh a ověření řídicích algoritmů s omezením pro laboratorní model tepelné soustavy*. Diplomová práce, UTB ve Zlíně, 2014.
- [15] SCHLEGEL, M.: *Průmyslové PID regulátory*. Tutoriál, Rexcontrols, 2003.
- [16] VAVŘÍN, P.; BLAHA, P.: *Řízení a regulace I*. Skriptum, VUT Brno, 2014.

Seznam zkratek

ADO - ActiveX Data Objects

COM - Component Object Model

DDE - Dynamic Data Exchange

DZ - Dopravní zpoždění

MNČ - Metoda nejmenších čtverců

MRAS - Adaptivní systémy s referenčním modelem

ODBC - Open Database Connectivity

OPC - OLE for Process Control

PLC - Programable logick controler

PRBS - Pseudo Random Binary Signal

PWM - Pulse-width modulation

SCADA - Supervisory Control And Data Acquisition

T_{vz} - Vzorkovací perioda