

**SLOVENSKÁ TECHNICKÁ UNIVERZITA
V BRATISLAVE
FAKULTA CHEMICKEJ A POTRAVINÁRSKEJ
TECHNOLÓGIE**

Tvorba knižnice modelov technologických procesov

DIPLOMOVÁ PRÁCA

FCHPT-5414-58709

2013

Bc. Katarína Bugárová

**SLOVENSKÁ TECHNICKÁ UNIVERZITA
V BRATISLAVE
FAKULTA CHEMICKEJ A POTRAVINÁRSKEJ
TECHNOLÓGIE**

Tvorba knižnice modelov technologických procesov

DIPLOMOVÁ PRÁCA

FCHPT-5414-58709

Študijný program: Automatizácia a informatizácia v chémii a potravinárstve

Číslo a názov študijného odboru: 5.2.14 Automatizácia

Školiace pracovisko: Oddelenie informatizácie a riadenia procesov

Vedúci záverečnej práce/školiteľ: Ing. Ľuboš Čirka, PhD

2013

Bc. Katarína Bugárová



ZADANIE DIPLOMOVEJ PRÁCE

Študentka: **Bc. Katarína Bugárová**
ID študenta: 58709
Študijný program: automatizácia a informatizácia v chémii a potravinárstve
Študijný odbor: 5.2.14 automatizácia
Vedúci práce: Ing. Ľuboš Čírka, PhD.

Názov práce: **Tvorba knižnice modelov technologických procesov**

Špecifikácia zadania:

Cieľom práce je upraviť a doplniť knižnicu modelov technologických procesov v MWS (HTTP Matlab Web Server). Knižnica je základom virtuálneho laboratória na monitorovanie a riadenie procesov chemickej, biochemickej a potravinárskej technológie.

Úlohy:

1. Rešerš dostupných riešení virtuálnych laboratórií.
2. Naštudovať jazyk MATLAB.
4. Vykonať analýzu, špecifikovať požiadavky a navrhnuť štruktúru nových modulov.
5. Programovo realizovať moduly v MATLABe a Simulinku.
6. Implementovať vytvorené moduly do LMS Moodle.
7. Overiť funkčnosť a vypracovať postup inštalácie.
8. Vypracovať dokumentáciu.

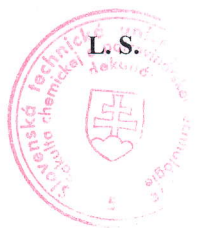
Rozsah práce: 50

Riešenie zadania práce od: 18. 02. 2013

Dátum odovzdania práce: 25. 05. 2013

Bc. Katarína Bugárová
študentka

prof. Ing. Miroslav Fikar, DrSc.
vedúci pracoviska



prof. Ing. Miroslav Fikar, DrSc.
garant študijného programu

PodĎakovanie

Touto cestou by som sa chcela poďakovať svojmu školiteľovi a vedúcemu záverečnej práce pánovi Ing. Ľubošovi Čirkovi, PhD za odborné rady, užitočné pripomienky a odbornú pomoc počas vypracovávania diplomovej práce.

Čestné prehlásenie

Čestne prehlasujem, že som predloženú záverečnú diplomovú prácu vypracovala samostatne podľa pokynov vedúceho práce, pričom som ťažila informácie z uvedenej literatúry v zozname literatúry.

Abstrakt

Diplomová práca pojednáva o tvorbe knižnice technologických modelov. Práca rozširuje projekt virtuálneho laboratória o nové modely, vlastnosti a dopĺňa ho vybranými typmi riadenia. Teoretická časť obsahuje matematické modely procesov, na základe ktorých sa vytvárajú simulácie v aplikácii Web Server v spolupráci s programom MATLAB. Simulácie sú implementované do kurzu Virtuálne a vzdialené laboratórium, kde sa dáta pre simuláciu vkladajú do HTML formulárov. Tieto formuláre spracúva Web Server a simulácie sa vykonávajú v programe MATLAB. Odpoveďou na odoslané dáta po vykonaní simulácie je HTML súbor vygenerovaný aplikáciou Web Server. Táto stránka obsahuje grafický priebeh simulácie vykreslený pomocou knižnice Flot a samotné vstupné dáta simulácie. Praktická časť opisuje postup vytvorenia simulácie vo virtuálnom laboratóriu. Sumárne obsahuje knižnica modely prietokového chemického reaktora, etážovej kolóny a dvoch zásobníkov, ktoré je možné riadiť vybranými typmi regulátorov.

Kľúčové slová: model, formulár, Web Server, MATLAB, PID, 1DoF, 2DoF

Abstract

The diploma thesis deals with the development of a library of technological process models. The work extends the virtual lab project of new models, properties and supplements it by several types of controllers. Simulations of models are created in Web Server in cooperation with MATLAB. These are based on mathematical models described in the theoretical part. The simulations are implemented in the course Virtual and Remote Laboratory, where the simulation data have to be inserted in a HTML form. This form is processed by the Web Server and the simulations are done in MATLAB. The answer to the sent data after the simulation is a HTML file generated by Web Server. This web page contents the simulation diagram created by Flot library and also the input simulation data. The practical part describes the procedure of simulation design in the virtual lab. To summarize the library contents models of continuous stirred tank reactor, etage distillation column and two tanks, these could be controlled by several types of controllers.

Key words: model, form, Web Server, MATLAB, PID, 1DoF, 2DoF

Obsah

Abstrakt	6
Zoznam obrázkov	10
Zoznam tabuliek	12
Zoznam použitých symbolov a skratiek	13
Zoznam príloh	14
Úvod	15
1 Cieľ práce	16
2 Teoretické princípy.....	17
2.1 Riešenia virtuálnych laboratórií pomocou MWS	17
2.2 Teoretické základy modelovania vybraných procesov.....	21
2.2.1 Model sériovo zapojených zásobníkov kvapaliny	21
2.2.2 Model plášťového prietokového chemického reaktora.....	27
2.2.3 Model etážovej rektifikačnej kolóny.....	30
2.3 Teoretické základy vybraných metód riadenia	35
2.3.1 PID regulátor	35
2.3.2 Polynomické vstupnovýstupné regulátory	37
2.4 Prehľad použitých technológií.....	39
2.4.1 Formulár simulačných údajov	40
2.4.2 Matlab Web Server a Web Server.....	43
2.5 MATLAB	45
2.5.1 Simulink	46
3 Praktická časť	46
3.1 Formulár vstupných dát etážovej rektifikačnej kolóny	46
3.2 M-file a spracovanie simulačných údajov	51
3.2.1 Formát vstupných a výstupných dát z m-file WS	51
3.2.2 Telo m-file a výpočtové funkcie	52
3.2.3 Simulácia v Simulinku pomocou m-file	54

3.3	Flot a grafy.....	55
3.4	Overenie správnosti simulačných dát	58
3.5	Príklady simulácie riadenia.....	62
3.6	Implementácia knižnice v LMS Moodle	64
	Záver.....	65
	Zoznam použitej literatúry.....	66

Zoznam obrázkov

Obr. 1: Grafické zobrazenie zaťaženia sedlovej strechy vrstvou snehovej pokrývky.....	18
Obr. 2: Výpočet prenosových parametrov štruktúrovanej metalickej kabeláže.....	19
Obr. 3: Riadenie výmenníka tepla pomocou DMC	19
Obr. 4: Riadenie výmenníkov tepla pomocou PID regulátora	20
Obr. 5: Ilustrácia sériového zapojenia zásobníkov za sebou s interakciou	22
Obr. 6: Ilustrácia sériového zapojenia zásobníkov za sebou bez interakcie.....	25
Obr. 7: Ilustrácia prietokového chemického reaktora	27
Obr. 8: Ilustrácia etážovej rektifikačnej kolóny	30
Obr. 9: Rovnovážna krivka x-y etážovej rektifikačnej kolóny so 7 etážami.....	33
Obr. 10: Porovnanie lineárneho (identifikovaného) a nelineárneho systému	35
Obr. 11: URO s PID regulátorom a nelineárnym systémom.....	37
Obr. 12: Schéma URO s PA regulátorom 1DoF	38
Obr. 13: Schéma URO s PA regulátorom 1DoF s integračnou konštantou	39
Obr. 14: Schéma URO s PA regulátorom 2DoF	39
Obr. 15: Príklad dynamicky generovaného grafu pomocou Flot	42
Obr. 16: Schematické znázornenie činnosti WS	44
Obr. 17: Spustenie programu WS.....	45
Obr. 18: Formulár vstupných údajov simulácie etážovej rektifikačnej kolóny.....	48
Obr. 19: Upozornenie užívateľa na nesprávnosť zadaných údajov	50
Obr. 20: Schéma riadenia rektifikačnej kolóny pomocou PID regulátora	54
Obr. 21: Blok skokovej zmeny v Simulinku	55
Obr. 22: Simulácia na overenie ustálených stavov rektifikačnej kolóny	59
Obr. 23: Simulácia skokovej zmeny v rektifikačnej kolóne.....	60
Obr. 24: Simulácia riadenia na ustálený stav v rektifikačnej kolóne	60
Obr. 25: Ilustrácia riadenia na ustálený stav v rektifikačnej kolóne bez akčného zásahu61	
Obr. 26: Riadenie zloženia destilátu na žiadanú veličinu PID regulátorom.....	61
Obr. 27: Riadenie výšky hladiny v 2. zásobníku pomocou 1DoF regulátora.....	63
Obr. 28: Riadenie výšky hladiny v 2. zásobníku pomocou 1DoF regulátora s integračnou zložkou.....	63

Obr. 29: Riadenie výšky hladiny v 2. zásobníku pomocou 2DoF regulátora s integračnou zložkou.....	64
Obr. 30: Virtuálne laboratórium Vlab v LMS Moodle.....	64

Zoznam tabuliek

Tabuľka 1: Veličiny a parametre popisujúce dva zásobníky kvapaliny	22
Tabuľka 2: Veličiny a parametre popisujúce prietokový chemický reaktor	28
Tabuľka 3: Veličiny a parametre popisujúce etážovú rektifikačnú kolónu.....	32

Zoznam použitých symbolov a skratiek

MWS	Matlab Web Server od firmy Mathworks
WS	Web Server od Dirk J. Kroon
HTTP	<i>HyperText Transfer Protocol</i> - protokol definujúci prenos hypertextových dokumentov, teda webových stránok
Server	v kontexte klient-server architektúry je server počítač, kde je uložený program spravujúci požiadavky ostatných počítačov
CGI	<i>Common Gateway Interface</i> - protokol na prepojenie externých aplikácií s webovým serverom
W3C	<i>World Wide Web Consortium</i> – organizácia na správu štandardov programovacích jazykov webových stránok
URL	<i>Uniform Resource Locator</i> - pre užívateľa je to adresa webovej stránky alebo inak reťazec identifikujúci umiestnenie dokumentu v internetovej štruktúre, resp. lokalizáciu dokumentu na serveri
UTF	<i>Unicode Transformation Format</i> - bezstratové kódovanie s variabilnou dĺžkou určené pre Unicode znaky
deg	označenie stupňa polynómu
PA	Pole assignment, metóda umiestnenia pólov
URO	uzavretý regulačný obvod
LMS	Learning Management System
Moodle	Modular Object-Oriented Dynamic Learning Environment

Zoznam príloh

Príloha č. 1.: CD obsahujúce program WS spolu s vytvorenými modelmi simulácii a riadenia a elektronickú verziu práce

Úvod

Pokrok v oblasti informačných technológií má výrazný vplyv aj na vzdelávanie. Významným nástrojom elektronického vzdelávania je internet. Výmena informácií prostredníctvom internetu umožňuje nielen diaľkové štúdium, ale aj ovládanie aplikácií či zariadení. Vzdelávacie on-line kurzy môžu byť rôzneho charakteru. Môžu reprezentovať zdroj výukových materiálov, video prednášok alebo obsahovať cvičenia, testy či dokonca rôzne virtuálne simulačné prostredia.

Jedným z rozšírených nástrojov on-line vzdelávania je Moodle. Moodle je voľne dostupný LMS systém, ktorý ponúka veľa možností vytvárania a správy on-line kurzov.

Súčasťou on-line kurzu môže byť virtuálne laboratórium zamerané na simulácie ekonomického, biologického, fyzikálneho či chemického charakteru. Virtuálne laboratórium je prostredie, kde sú simulované podmienky veľmi podobné tým reálnym. Simulácie reálnych javov a procesov v týchto laboratóriách majú však veľkú výhodu čo sa týka bezpečnosti a financií. Najmä ak ide o javy a procesy, pri ktorých sa vyžaduje veľká opatrnosť alebo by prevedenie simulácie či skúmaného javu, vyžadovalo značné finančné prostriedky.

Virtuálne laboratóriá sú často využívané na prevedenie simulácií aj v oblasti chemickej technológie a automatizácie. Študenti môžu prostredníctvom virtuálnych laboratórií simulovať riadenie reaktorov, destilačných kolón alebo prevádzkať simulácie modelov chemickotechnologických procesov. Je to nástroj, ako vytvoriť predstavu reálneho procesu a zároveň poskytnúť možnosť vykonať simulácie, teda pochopiť čo sa stane ak ...

Existuje veľa nástrojov a aplikácií na vytvorenie simulačného prostredia či prevedenie simulácií. Jazyky ako JAVA, PHP, JavaScript či rôzne programy ako Flash, Dynsim, Unisim, alebo MATLAB so Simulinkom či už samostatne alebo v kombinácii, môžu byť použité na vytvorenie veľmi jednoduchého a účelného simulačného prostredia, ale aj veľmi zložitého a realistického.

1 Cieľ práce

Cieľom diplomovej práce je rozšírenie už existujúceho virtuálneho laboratória o knižnicu modelov technologických procesov aj s vybranými typmi riadenia. Virtuálne laboratórium je sprístupnené v kurze Virtuálne a vzdialené laboratórium na LMS portáli Moodle Fakulty chemickej a potravinárskej technológie Slovenskej Technickej Univerzity v Bratislave. [1]

Práca sa zaoberá rozšírením ponuky metód riadenia o polynomicke regulátory typu 1DoF, 1DoF s integračnou zložkou a 2DoF. Ako ďalšie procesy pre rozšírenie ponuky virtuálneho laboratória boli vybrané plášťový chemický prietokový reaktor a etážová rektifikačná kolóna. Virtuálne laboratórium je implementované vo Web Serveri pomocou jazyka MATLAB.

2 Teoretické princípy

Vytvorenie knižnice modelov procesov chemickej technológie si vyžaduje znalosť matematických modelov vybraných procesov a taktiež simulačné prostredie virtuálneho laboratória, teda program, v ktorom budú modely implementované.

Diplomová práca sa prednostne nezameriava na teoretické odvodenie modelov procesov. Z tohto dôvodu sú v práci uvedené len základné vzťahy, aby bolo zrejmé o aké modely sa jedná. Pri každom modeli procesu sú taktiež uvedené zjednodušujúce predpoklady. Matematické vzťahy modelov procesov uvedené v tejto kapitole sú čerpané z [2] [3] [4].

Knižnica technologických modelov v tejto práci zahŕňa aj simuláciu riadenia. Návrh regulátorov pre simuláciu riadenia nie je súčasťou práce. Základné teoretické pozadie metód riadenia je uvedené v kapitole 2.3. [5].

Samotná simulácia procesov prebieha v prostredí programu MATLAB a jeho súčasti Simulink. Proces implementácie matematických modelov vysvetľuje kapitola 3.

Simulačné prostredie je však potrebné ešte prepojiť s internetom, ktorý slúži ako komunikačný kanál na odosielanie údajov simulácie. Prostriedkom komunikácie medzi internetom a programom MATLAB sa venuje kapitola 2.4.2.

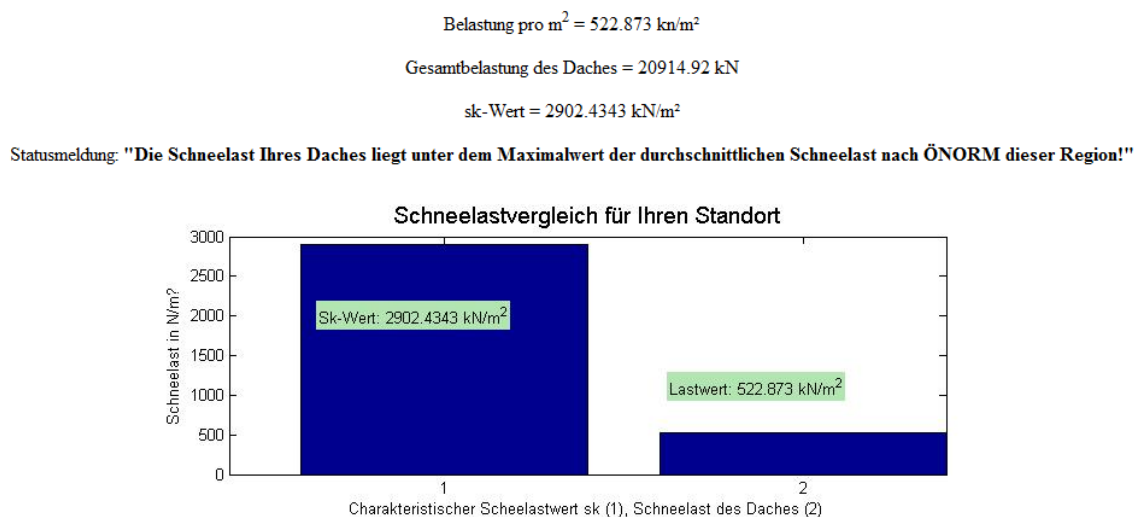
2.1 Riešenia virtuálnych laboratórií pomocou MWS

Podpora vyučovania prostredníctvom virtuálnych laboratórií sa teší veľkej obľube v celom svete. Projekty virtuálnych laboratórií zväčša vychádzajú z iniciatívy vzdelávacích inštitúcií s technickým zameraním a sú realizované ako vedecké či študentské práce.

Sústrediac sa na virtuálne laboratóriá využívajúce MWS nachádzame rozmanité množstvo riešení simulácií.

Všestranné využitie MWS nám ponúkajú projekty realizované na Univerzite aplikovanej vedy Carinthia v Rakúsku. Môžete si napríklad vypočítať zaťaženie sedlovej strechy vrstvou snehovej pokrývky podľa rakúskej normy. Stačí zadať

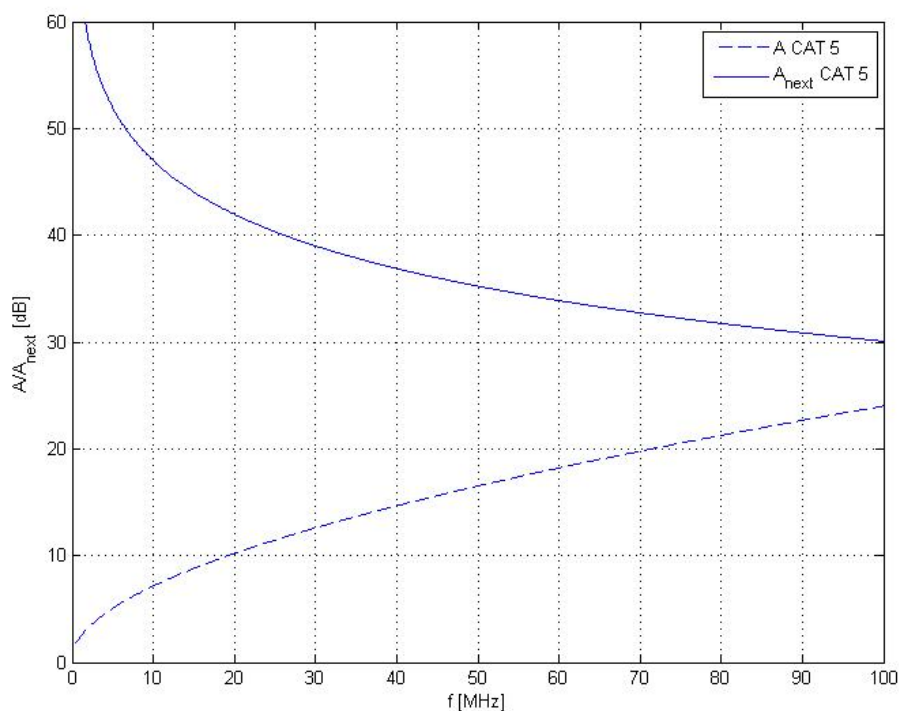
parametre ako sklon strechy, výška snehovej pokrývky, plochu strechy, dôležitý je typ snehovej pokrývky a nadmorská výška a získate výpočet a grafické zobrazenie zaťaženia strechy špecifikovanou snehovou pokrývkou. Výsledok simulácie je ilustrovaný na Obr. 1. [6]



Obr. 1: Grafické zobrazenie zaťaženia sedlovej strechy vrstvou snehovej pokrývky

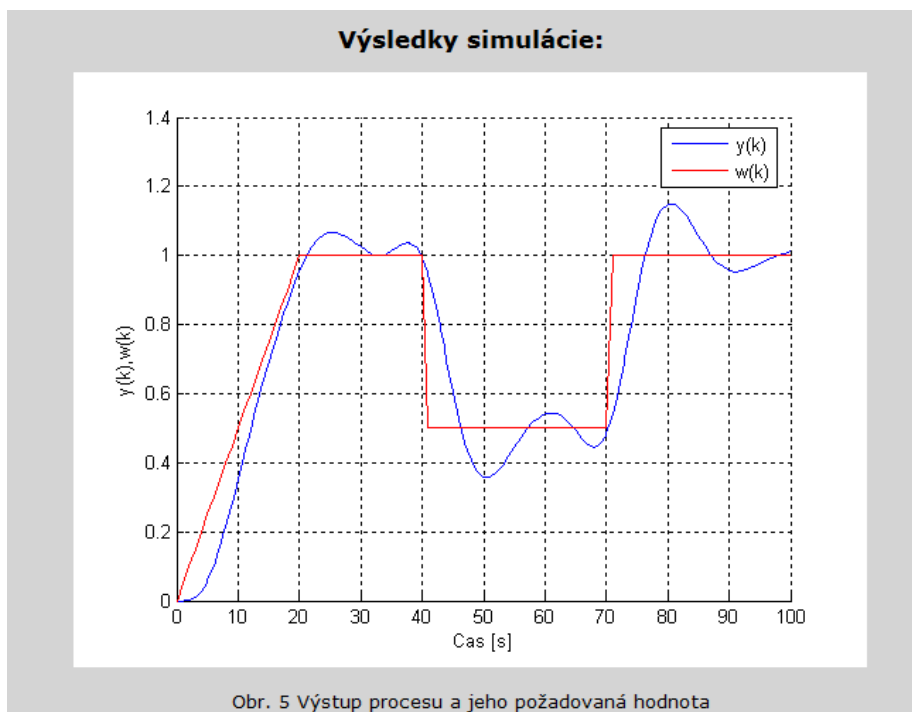
České Vysoké Učení technické v Praze vypracovalo taktiež množstvo pozoruhodných simulácií ako je napríklad výpočet prenosových parametrov štruktúrovanej metallickej kabeláže. Čiastočné výsledky simulácie sú zobrazené na Obr. 2. [7]

Virtuálne laboratóriá využívajúce MWS nájdeme nielen v zahraničí, ale i na Slovensku. Technická univerzita v Košiciach vytvorila viaceré modely procesov, ktoré je možné simulovať priamo cez ich webovú stránku. Vypracovali model simulácie a riadenia viacerých zložitých systémov ako je napríklad model riadenia helikoptéry, udržanie guľôčky na ploche alebo magnetická levitácia. Uvažujú taktiež o DMC riadení výmenníka tepla, čo je prediktívne riadenie pre MIMO systém. Po simulácii sa zobrazia výstupy v tvare priebehov výstupnej veličiny a akčného zásahu (Obr. 3). [8]



Vypočtený útlum a přeslech NEXT přenosové cesty s lineární osou kmitočtu

Obr. 2: Výpočet prenosových parametrov štruktúrovanej metallickej kabeláže

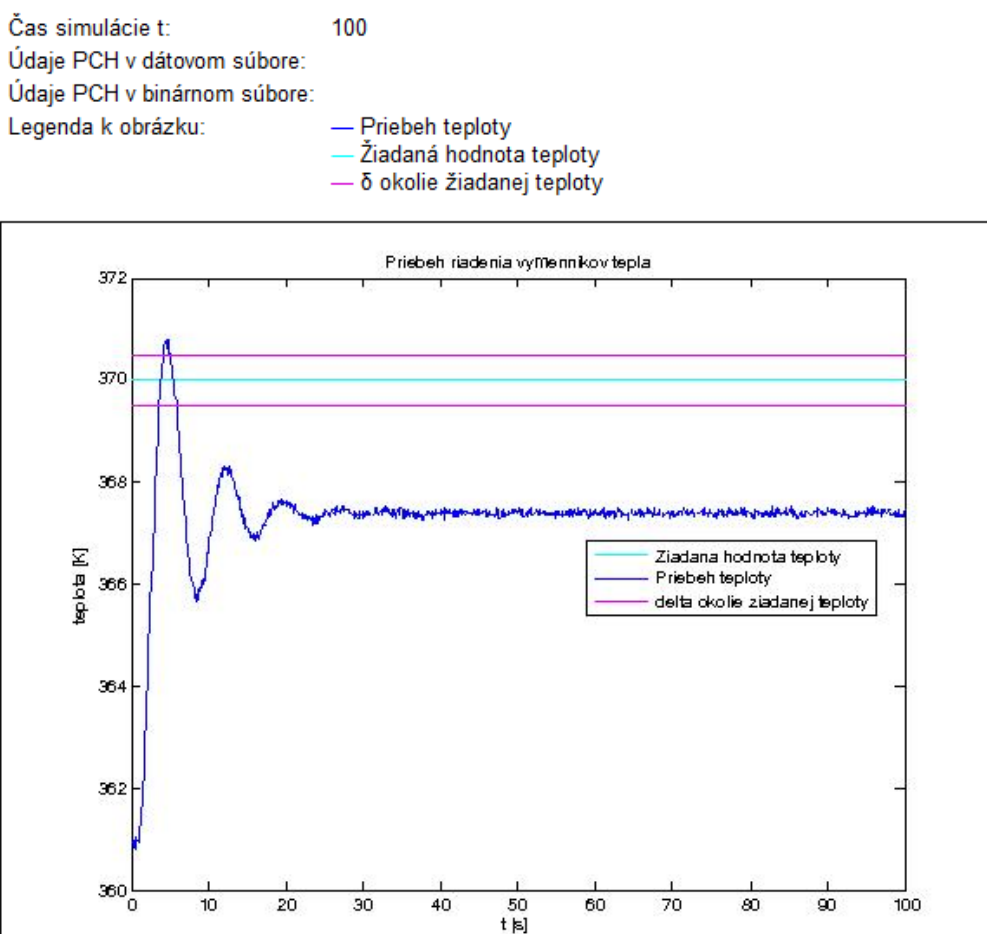


Obr. 5 Výstup procesu a jeho požadovaná hodnota

Obr. 3: Riadenie výmenníka tepla pomocou DMC

Vzdelávací kurz Laboratórne cvičenia z riadenia procesov na Moodle FCHPT ponúka ucelený úvod do kurzu Riadenie procesov, kde si môžu študenti krok po kroku overiť vedomosti z algebry prenosov, Laplaceovej transformácie alebo vypočítať korene polynómu tým zistiť póly a nuly systému a následne si popríklad vykresliť impulznú alebo prechodovú charakteristiku. Kurz obsahuje simulácie dvoch zásobníkov zapojených sériovo a simuláciu plášťových výmenníkov tepla. Pre oba procesy je vytvorená aj simulácia riadenia pomocou PID regulátora. (Obr. 4). [9]

Virtuálne laboratórium na Moodle bolo povodne implementované v Matlab Web Serveri od firmy Mathworks. Od roku 2013 je však virtuálne laboratórium implementované vo Web Serveri od Dirka J. Kroona.



Obr. 4: Riadenie výmenníkov tepla pomocou PID regulátora

2.2 Teoretické základy modelovania vybraných procesov

Teoretické predpoklady modelovania sú neodmysliteľnou súčasťou reprezentácie reálneho objektu ako modelu. Model je synteticky vytvorený reprezentant jedného alebo viacerých objektov zahrňujúcich informácie o ňom, ktoré vznikajú projekciou z originálu. [10]

Pri vytvorení matematického modelu zariadenia technologického procesu sa uvažujú zjednodušujúce predpoklady, ktoré uľahčujú sformulovanie problému a zovšeobecňujú model procesu.

Kapitola obsahuje stručné odvodenie matematických modelov vybraných procesov a taktiež zjednodušujúce predpoklady, podľa ktorých boli modely odvodené.

2.2.1 Model sériovo zapojených zásobníkov kvapaliny

[2]

Presný matematický model poskytuje veľmi dobré predpoklady na dobrú zhodu modelu a reálneho procesu. Pre odvodenie matematického modelu sériovo zapojených zásobníkov kvapaliny či už s interakciou, alebo bez je potrebné určiť zjednodušujúce predpoklady za akých bol model odvodený:

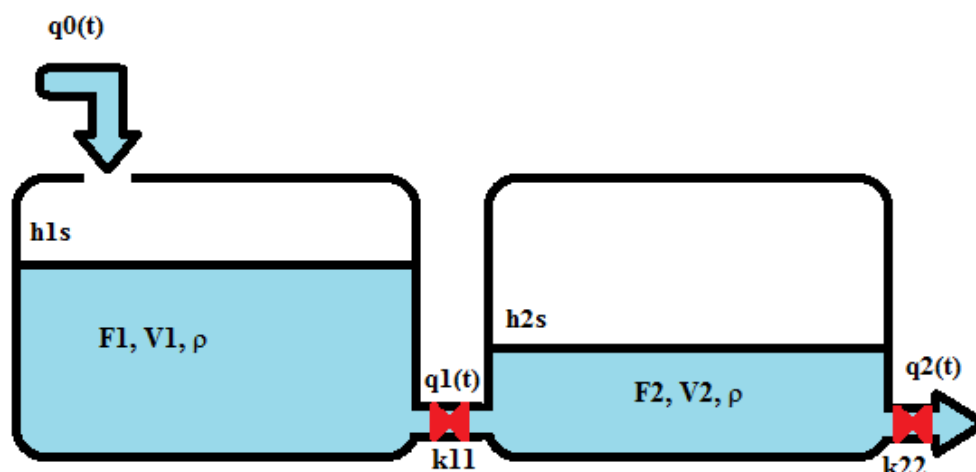
- ide o nepretržitý prietokový systém
- hustota kvapaliny v zásobníku je konštantná
- teplota kvapaliny je konštantná
- kvapalina v sústave má homogénne zloženie
- tlak nad kvapalinou vo všetkých zásobníkoch je rovnaký a konštantný
- steny plášťa zásobníkov sú kolmé na ich podstavy
- nedochádza k interakcii s okolím
- uvažujeme len materiálový tok pre účely materiálovej bilancie jednotlivých členov

Aby bolo zrejmé aké veličiny v modeli vystupujú, čo vyjadrujú a aké majú jednotky, sú tieto informácie zhrnuté v tabuľke 1.

Tabuľka 1: Veličiny a parametre popisujúce dva zásobníky kvapaliny

$h_1[m]$	výška hladiny v 1. zásobníku kvapaliny
$h_2[m]$	výška hladiny v 2. zásobníku kvapaliny
$q_0 [m^3 s^{-1}]$	prietok kvapaliny na vstupe do 1. zásobníka kvapaliny
$q_1 [m^3 s^{-1}]$	prietok kvapaliny na výstupe z 1. zásobníka kvapaliny
$q_2 [m^3 s^{-1}]$	prietok kvapaliny na výstupe z 2. zásobníka kvapaliny
$F_1[m^2]$	plocha prierezu 1. zásobníka kvapaliny
$F_2[m^2]$	plocha prierezu 2. zásobníka kvapaliny
$V_1 [m^3]$	objem 1. zásobníku kvapaliny
$V_2 [m^3]$	objem 2. zásobníku kvapaliny
$\rho [kg m^3]$	hustota kvapaliny v zásobníku kvapaliny
$k_{11} [m^{2,5} s^{-1}]$	konštanta 1. ventilu
$k_{22} [m^{2,5} s^{-1}]$	konštanta 2. ventilu

Zásobníky kvapaliny s interakciou



Obr. 5: Ilustrácia sériového zapojenia zásobníkov za sebou s interakciou

Uvažujeme o vstupnom prúde, kde vteká kvapalina prietokom q_0 do prvého zásobníka. Kvapalina z prvého zásobníka preteká do druhého cez ventil, ktorého parametre sú zohľadnené v konštante k_{11} . Oba zásobníky sú v rovnakej nadmorskej výške a nedochádza k spätnému toku kvapaliny z druhého zásobníka. Na výtok

z druhého zásobníka je taktiež umiestnený ventil, ktorého vlastnosti zahŕňa konštanta k_{22} . Predpokladá sa, že kvapalina vyteká z druhého zásobníka kontinuálne konštantnou rýchlosťou. Ilustrácia procesu sériovo zapojených zásobníkov s interakciou je na Obr. 5.

Ďalej sú uvedené matematické vzťahy charakteristické pre uvedený proces.

Materiálová bilancia:

$$q_0(t)\rho = q_1(t)\rho + \rho \frac{dV_1(t)}{dt} \quad (1)$$

$$q_1(t)\rho = q_2(t)\rho + \rho \frac{dV_2(t)}{dt} \quad (2)$$

Pre q_1, q_2, V_1, V_2 platí

$$V_1 = F_1 h_1(t) \quad (3)$$

$$V_2 = F_2 h_2(t) \quad (4)$$

$$q_1(t) = k_{11}\sqrt{h_1(t) - h_2(t)} \quad (5)$$

$$q_2(t) = k_{11}\sqrt{h_2(t)} \quad (6)$$

Po dosadení q_1, q_2, V_1, V_2 je zrejmé, že náš model obsahuje nelinearity v tvare odmocniny.

Získavame dynamický matematický model:

$$q_0(t) = k_{11}\sqrt{h_1(t) - h_2(t)} + F_1 \frac{h_1(t)}{dt} \quad (7)$$

$$k_{11}\sqrt{h_1(t) - h_2(t)} = k_{22}\sqrt{h_2(t)} + F_2 \frac{h_2(t)}{dt} \quad (8)$$

Väčšina procesov vykazuje nelinearity. Matematický model procesu sa linearizuje, v prípade výskytu nelinearit, aby bolo možné na základe prenosu navrhnúť regulátor, a tak zabezpečiť stabilitu a riaditeľnosť systému.

Odvedenie nelineárneho dynamického matematického modelu:

Začiatkové podmienky:

$$h_1(0) = h_1^s$$

$$h_2(0) = h_2^s$$

Stavové veličiny $h_1(t), h_2(t)$

Vstupné veličiny $q_0(t)$

Výstupné veličiny $h_1(t), h_2(t)$

Matematický model zásobníkov v ustálenom stave

V ustálenom stave sa kvapalina v systéme neakumuluje, akumulčný člen je $\frac{d}{dt} = 0$.

$$q_0^s = k_{11}\sqrt{h_1^s - h_2^s} \quad (9)$$

$$k_{11}\sqrt{h_1^s - h_2^s} = k_{22}\sqrt{h_2^s} \quad (10)$$

Všeobecné vzťahy pre výpočet výšok hladín v ustálenom stave

$$h_1^s = \left(\frac{q_0^s}{k_{11}}\right)^2 + h_2^s \quad (11)$$

$$h_2^s = \left(\frac{q_0^s}{k_{11}}\right)^2 \quad (12)$$

Definovanie odchýlkových veličín pre účely linearizácie

$$x_1(t) = h_1(t) - h_1^s \quad (13)$$

$$x_2(t) = h_2(t) - h_2^s \quad (14)$$

$$u_1(t) = q_0(t) - q_0^s \quad (15)$$

Pomocné konštanty získané po dosadení linearizovaných a odchýlkových členov

$$k_1 = \frac{k_{11}}{2\sqrt{h_1^s - h_2^s}} \quad (16)$$

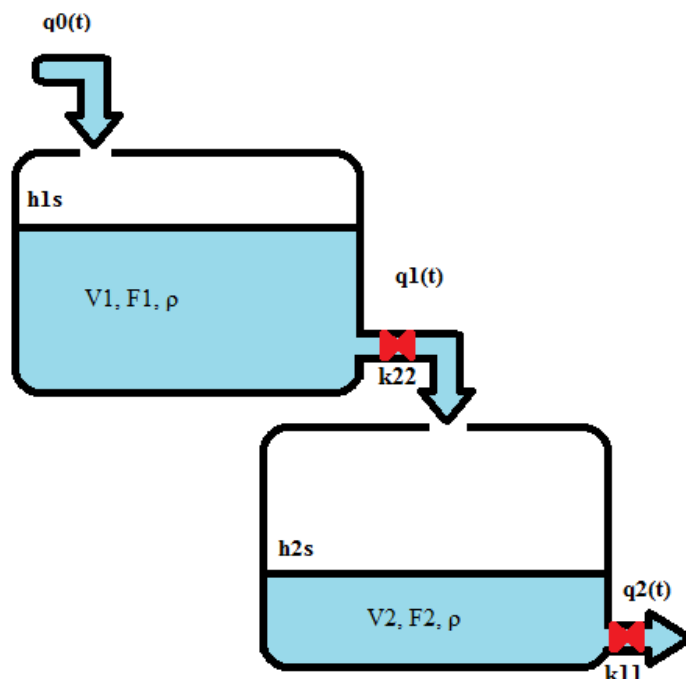
$$k_2 = \frac{k_{22}}{2\sqrt{h_2^s}} \quad (17)$$

Linearizovaný odchýlkový dynamický matematický model

$$u_1(t) = k_1 x_1(t) - k_1 x_2(t) + F_1 \frac{x_1(t)}{dt} \quad x_1(t) = 0 \quad (18)$$

$$k_1 x_1(t) - k_1 x_2(t) = k_2 x_2(t) + F_2 \frac{x_2(t)}{dt} \quad x_2(t) = 0 \quad (19)$$

2.2.1.1 Zásobníky kvapaliny bez interakcie



Obr. 6: Ilustrácia sériového zapojenia zásobníkov za sebou bez interakcie

Skutočnosti uvedené pre model zásobníkov kvapaliny v predchádzajúcej kapitole sú totožné aj pre zásobníky bez interakcie, avšak s rozdielom, že kvapalina z prvého zásobníka preteká do druhého bez pôsobenia hydrostatického tlaku na ventil s konštantou k_{11} . Na Obr. 6 sa nachádza ilustrácia sériového zapojenia dvoch zásobníkov kvapaliny bez interakcie.

Materiálovou bilanciou vstupného, výstupného prúdu a uvažovaním akumulácie kvapaliny v zásobníkoch získame matematický model:

Materiálová bilancia:

$$q_0(t)\rho = q_1(t)\rho + \rho \frac{dV_1(t)}{dt} \quad (20)$$

$$q_1(t)\rho = q_2(t)\rho + \rho \frac{dV_2(t)}{dt} \quad (21)$$

Pre q_1, q_2, V_1, V_2 platí

$$V_1 = F_1 h_1(t) \quad (22)$$

$$V_2 = F_2 h_2(t) \quad (23)$$

Po dosadení q_1 , q_2 , V_1 , V_2 je zrejmé, že aj model sériovo zapojených zásobníkov bez interakcie obsahuje nelinearity v tvare odmocniny.

Získavame dynamický matematický model:

$$q_1(t) = k_{11}\sqrt{h_1(t)} \quad (24)$$

$$q_2(t) = k_{11}\sqrt{h_2(t)} \quad (25)$$

Odvozenie nelineárneho dynamického matematického modelu:

$$q_0(t) = k_{11}\sqrt{h_1(t)} + F_1 \frac{h_1(t)}{dt} \quad (26)$$

$$k_{11}\sqrt{h_1(t)} = k_{22}\sqrt{h_2(t)} + F_2 \frac{h_2(t)}{dt} \quad (27)$$

Začiatkové podmienky:

$$h_1(0) = h_1^s$$

$$h_2(0) = h_2^s$$

Stavové veličiny $h_1(t)$, $h_2(t)$

Vstupné veličiny $q_0(t)$

Výstupné veličiny $h_1(t)$, $h_2(t)$

Matematický model zásobníkov v ustálenom stave

V ustálenom stave sa kvapalina v systéme neakumuluje, akumulačný člen je $\frac{d}{dt} = 0$.

$$q_0^s = k_{11}\sqrt{h_1^s} \quad (28)$$

$$k_{11}\sqrt{h_1^s} = k_{22}\sqrt{h_2^s} \quad (29)$$

Všeobecné vzťahy pre výpočet výšok hladín v ustálenom stave

$$h_1^s = \left(\frac{q_0^s}{k_{11}}\right)^2 \quad (30)$$

$$h_2^s = \left(\frac{q_0^s}{k_{11}}\right)^2 \quad (31)$$

Definovanie odchýlkových veličín pre účely linearizácie

$$x_1(t) = h_1(t) - h_1^s \quad (32)$$

$$x_2(t) = h_2(t) - h_2^s \quad (35)$$

$$u_1(t) = q_0(t) - q_0^s \quad (36)$$

Pomocné konštanty získané po dosadení linearizovaných a odchýlkových členov

$$k_1 = \frac{k_{11}}{2\sqrt{h_1^s}} \quad (37)$$

$$k_2 = \frac{k_{22}}{2\sqrt{h_2^s}} \quad (38)$$

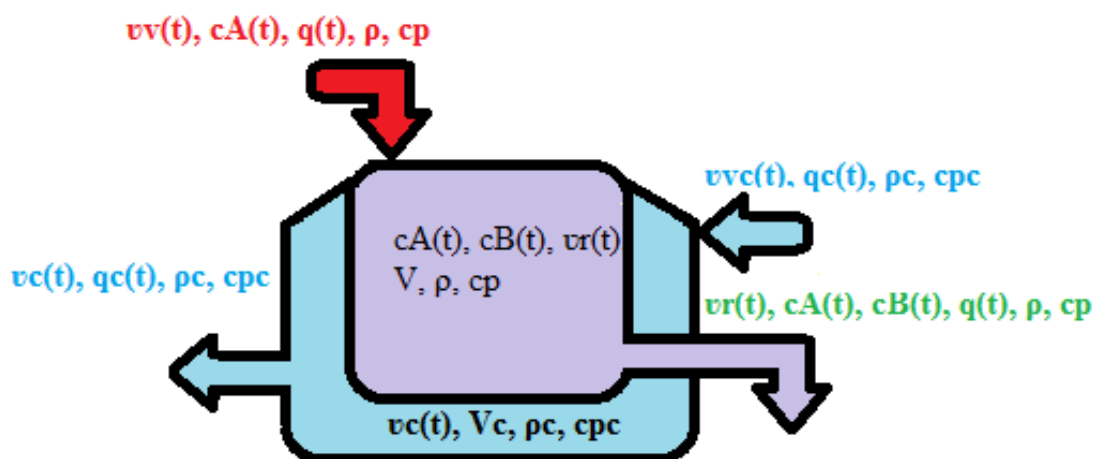
Linearizovaný odchýlkový dynamický matematický model v tvare dvoch lineárnych diferenciálnych rovníc so začiatočnými podmienkami

$$u_1(t) = k_1 x_1(t) + F_1 \frac{x_1(t)}{dt} \quad x_1(t) = 0 \quad (39)$$

$$k_1 x_1(t) = k_2 x_2(t) + F_2 \frac{x_2(t)}{dt} \quad x_2(t) = 0 \quad (40)$$

2.2.2 Model plášťového prietokového chemického reaktora

[3]



Obr. 7: Ilustrácia prietokového chemického reaktora

Práca sa taktiež zaoberá chemickým reaktorom (Obr. 7) s dokonalým miešaním reakčnej zmesi s exotermickou reakciou 1. poriadku, $A \xrightarrow{k} B$.

Vstupom do reaktora je prietok reakčnej zmesi q , ktorý obsahuje iba zložku A. Vstupná zmes má teplotu v a koncentrácia zložky A je c_{vA} . Počas reakcie sa zložka A premení na zložku B o koncentrácii c_B . Reakcia je chladená chladiacou zmesou o teplote v_c , ktorá preteká plášťom reaktora. Výstupom je reakčná zmes, ktorá prevažne obsahuje iba zložku B. Uvažuje sa, že výstupná teplota je zhodná s teplotou v reaktore. Prehľad názvov veličín vystupujúcich vo matematickom modeli sa nachádza v tabuľke 2. Pri odvádzaní matematického modelu uvažujeme materiálovú bilanciu látky A, entalpickú bilanciu reakčnej zmesi a entalpickú bilanciu chladiaceho média.

Zjednodušujúce predpoklady:

- reagujú 2 zložky: reaktant A a produkt B v reakcii 1. poriadku $A \xrightarrow{k} B$
- reakcia je exotermická, reakčná entalpia je záporná. $\Delta_r H < 0$
- tepelná kapacita stien reaktora je zanedbateľná
- straty tepla do okolia sú zanedbateľné
- tlak v systéme je konštantný, t.j. zmena entalpie sa rovná zmene tepla
- technologické parametre sú konštantné: V , V_c , ρ , ρ_c , c_{cp} , c_{cpc} , A , α .

Tabuľka 2: Veličiny a parametre popisujúce prietokový chemický reaktor

$v_v [K]$	vstupná teplota reakčnej zmesi
$v [K]$	teplota reakčnej zmesi
$v_{cv} [K]$	vstupná teplota chladiaceho média v plášti reaktora
$v_c [K]$	teplota chladiaceho média v plášti reaktora
$c_{vA} [mol\ m^{-3}]$	vstupná koncentrácia látky A
$c_A [mol\ m^{-3}]$	koncentrácia látky A
$c_{vB} [mol\ m^{-3}]$	vstupná koncentrácia látky B
$c_B [mol\ m^{-3}]$	koncentrácia látky B
$q [m^3\ s^{-1}]$	prietok reakčnej zmesi
$q_c [m^3\ s^{-1}]$	prietok chladiaceho média v plášti reaktora
$V [m^3]$	objem reakčnej zmesi
$V_c [m^3]$	objem chladiaceho média v plášti reaktora
$\rho [kg\ m^{-3}]$	hustota reakčnej zmesi

$\rho_c [kg\ m^{-3}]$	hustota chladiaceho média v plášti reaktora
$c_{cp} [J\ kg^{-1}\ K^{-1}]$	špecifická tepelná kapacita reakčnej zmesi
$c_{cpc} [J\ kg^{-1}\ K^{-1}]$	špecifická tepelná kapacita chladiaceho média
$A [m^2]$	plocha prestupu tepla
$\alpha [W\ m^{-2}\ K^{-1}]$	koeficient prestupu tepla

Dynamický matematický model

Materiálová bilancia reagujúcej zložky A:

$$\frac{dc_A(t)}{dt} = \frac{q}{V} c_{vA}(t) - k_{\infty} e^{\frac{-g}{v(t)}} c_A(t) - \frac{q}{V} c_A(t) \quad c_A(0) \quad (41)$$

Materiálová bilancia reagujúcej zložky B

$$\frac{dc_B(t)}{dt} = \frac{q}{V} c_{vB}(t) - k_{\infty} e^{\frac{-g}{v(t)}} c_B(t) - \frac{q}{V} c_B(t) \text{ kde } c_{vB}(t) = 0 \text{ mol.dm}^{-3} \quad c_B(0) \quad (42)$$

Entalpická bilancia reakčnej zmesi:

$$\frac{dv(t)}{dt} = \frac{q}{V} v_v(t) + \frac{k_{\infty} e^{\frac{-g}{v(t)}} c_A(t) V (-\Delta_r H)}{\rho c_p} - \frac{q}{V} v(t) - \frac{\alpha A}{\rho V c_p} (v(t) - v_c(t)) \quad v(0) \quad (43)$$

Entalpická bilancia chladiaceho média:

$$\frac{dv_c(t)}{dt} = \frac{q_c}{V_c} v_{cv}(t) - \frac{q_c}{V_c} v_c(t) + \frac{\alpha A}{\rho_c V_c c_{pc}} (v(t) - v_c(t)) \quad v_c(0) \quad (44)$$

Stavové veličiny: $c_A(t), v(t), v_c(t)$

Vstupné veličiny: $c_{vA}(t), v_v(t), v_{cv}(t)$

Výstupné veličiny: $c_A(t), c_B(t), v(t), v_c(t)$

Matematický model v ustálenom stave

Materiálová bilancia reagujúcej zložky A:

$$0 = \frac{q}{V} c_{vA}^s - k_{\infty} e^{\frac{-g}{v^s}} c_A^s - \frac{q}{V} c_A^s \quad c_A(0) = c_A^s \quad (45)$$

Materiálová bilancia reagujúcej zložky B

$$0 = \frac{q}{V} c_{vB}^s - k_{\infty} e^{\frac{-g}{v^s}} c_B^s - \frac{q}{V} c_B^s \text{ kde } c_{vB}(t) = 0 \text{ mol.dm}^{-3} \quad c_B(0) = c_B^s \quad (46)$$

Entalpická bilancia reakčnej zmesi:

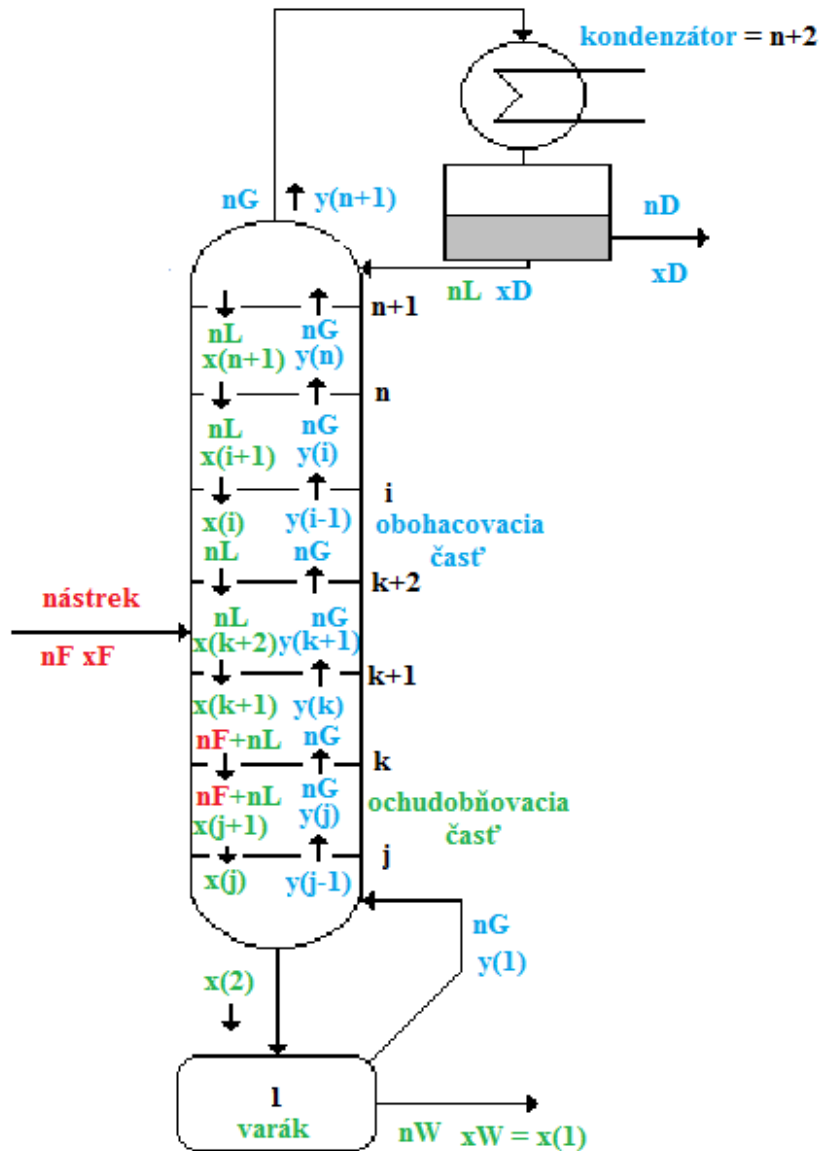
$$0 = \frac{q}{V} v_v^s + \frac{k_{\infty} e^{\frac{-g}{v^s}} c_A^s V (-\Delta_r H)}{\rho c p} - \frac{q}{V} v^s - \frac{\alpha A}{\rho V c p} (v^s - v_c^s) \quad v(0) = v^s \quad (47)$$

Entalpická bilancia chladiaceho média:

$$0 = \frac{q_c}{V_c} v_{cv}^s - \frac{q_c}{V_c} v_c^s + \frac{\alpha A}{\rho_c V_c c p c} (v^s - v_c^s) \quad v_c(0) = v_c^s \quad (48)$$

2.2.3 Model etážovej rektifikačnej kolóny

[4]



Obr. 8: Ilustrácia etážovej rektifikačnej kolóny

Etážová rektifikačná kolóna je systém s diskkrétne rozloženými parametrami, teda jej veličiny sa menia spojitou v čase, ale nespojitou v priestore. Rektifikácia je proces separácie vzájomne rozpustných látok založený na rozdielnych parciálnych tlakoch zložiek tvoriacich zmes. Rektifikačná kolóna je komplexný proces, kde je potrebné uvažovať viacero vzájomne súvisiacich parametrov. Kolóny sa vo všeobecnosti vyznačujú veľkými časovými konštantami a dopravným oneskorením. Model rektifikačnej kolóny uvažuje len bilanciu látkového množstva sústredujúcu sa na výsledné zloženie destilátu. Nástrekom je kvapalina o teplote bodu varu, nástreková etáž je voliteľná, rovnako ako aj zádrž na etážach, vo varáku alebo zatopenie kondenzátora. Účinnosť varáka je 100%.

Pre účely virtuálneho laboratória sa uvažuje zjednodušený model podľa uvedených zjednodušujúcich podmienok:

- nástrekom je binárna zmes, ktorá má ideálne vlastnosti, t.j. obe kvapaliny sa dokonale miešajú pozdĺž celej kolóny vrátane varáku a kondenzátora
- k prestupu látky dochádza iba na etážach
- uvažujeme teoretickú etáž, teda para odchádzajúca z etáže je v rovnováhe s kvapalinou na etáži
- skutočné zloženie kvapalnej fázy je rovné rovnovážnemu zloženiu kvapalnej fázy
- skutočné zloženie parnej fázy nie je rovné rovnovážnemu zloženiu parnej fázy
- zádrž kvapaliny na etážach, vo varáku a v kondenzátore je konštantná
- teplota kvapalnej fázy ako aj nástreku (kvapalina) je bod varu v celej kolóne a je konštantná
- tok látkového množstva parnej fázy je konštantný
- tok látkového množstva kvapalnej fázy je konštantný v ochudobňovacej časti ako aj v obohacovacej časti
- tlak a relatívna prchavosť sú konštantné
- mólové výparné teploty oboch oddelených zložiek sú rovnaké
- zjavné teploty ako aj straty do okolia sú zanedbateľné, kolóna je dokonale izolovaná

Tabuľka 3: Veličiny a parametre popisujúce etážovú rektifikačnú kolónu

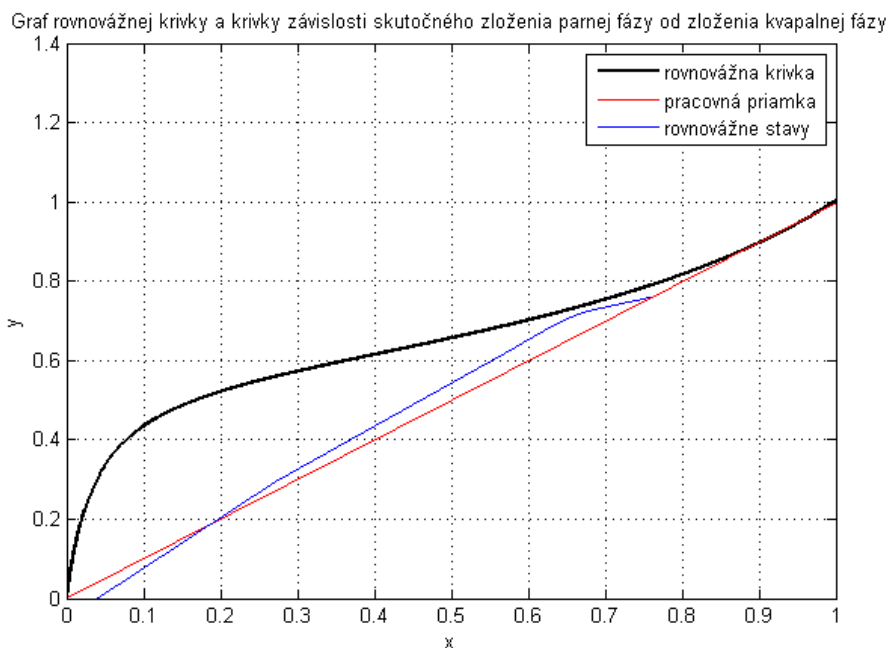
x	mólový zlomok prchavejšej zložky v kvapalnej fáze
y	mólový zlomok prchavejšej zložky v parnej fáze
y^*	rovnovážny mól. zlomok prchavejšej zložky v parnej fáze
a, b, c, d, e	Konštanty rovnovážnej krivky
$\dot{n}_G [mol\ s^{-1}]$	tok látkového množstva pary
$\dot{n}_L [mol\ s^{-1}]$	tok látkového množstva spätného toku
$\dot{n}_F [mol\ s^{-1}]$	tok látkového množstva nástreku
$\dot{n}_D [mol\ s^{-1}]$	tok látkového množstva destilátu
$\dot{n}_W [mol\ s^{-1}]$	tok látkového množstva zvyšku vytekajúceho z varáku
η	účinnosť etáže
x_F	mól. zlomok nástreku
x_D	mól. zlomok destilátu
x_W	mól. zlomok zvyšku
i	index obohacovacej časti kolóny
j	index obohacovacej časti kolóny
$k+1$	index nástrekovej časti kolóny
n	počet etáží
l	index varáka
$n+l$	index hlavy kolóny
H_{kond}	zádrž kvapaliny v kondenzátore
$H_{etáž}$	zádrž kvapaliny na etážach
$H_{varák}$	zádrž kvapaliny vo varáku

Uvažujúc vymenované zjednodušujúce predpoklady je možné odvodiť dynamický matematický model etážovej rektifikačnej kolóny pre n -etáží s premennými uvedenými v tabuľke 3. Mieru separovateľnosti nástrekovej zmesi vyjadruje:

Rovnica rovnovážnej krivky

$$y^* = \frac{a + bx + cx^2}{1 + dx + ex^2} \quad (49)$$

Rovnovážna krivka vyjadruje závislosť ako sa mení kvapalná fáza v závislosti od parnej. Z priebehu na Obr. 9 je zrejmé, že prednastavený príklad simulácie nie je azeotropická zmes. Zmes, ktorej x-y graf nevykazuje azeotropický bod je dobre separovateľná, t.j. je možné získať kvalitné zloženie destilátu.



Obr. 9: Rovnovážna krivka x-y etážovej rektifikačnej kolóny so 7 etážami

Výpočet rovnovážneho zloženia parnej fázy je pre každú etáž rovnaký, preto ho neuvádzame. Vo výpočtoch sa však uvažujú premenné s príslušným indexom pre konkrétne etáže.

Rovnovážne zloženie parnej fázy y^* prchavejšej zložky je potrebné na výpočet skutočného zloženia parnej fázy y . Z rovnice Murphreeho účinnosti pre n -tú etáž získavame zloženie parnej fázy (rovnaké pre obohacovaciu aj ochudobňovaciu časť kolóny) pri známej účinnosti etáží.

Rovnica Murphreeho účinnosti

$$\eta_i = \frac{y_i - y_{i+1}}{y_i^* - y_{i+1}} \rightarrow y_i = \eta_i y_i^* + (1 - \eta_i) y_{i+1} \quad (50)$$

Neodmysliteľnou súčasťou kolóny je varák, ktorý zohrieva kvapalinu stekajúcu po etážach. Varák sa bilancuje ako etáž. V tejto práci sa uvažuje ako prvá etáž (Obr. 8). Kondenzátor je takisto uvažovaný ako etáž. V tomto prípade ako $n+2$ etáž.

Bilancia látkového množstva pre kondenzátor => výpočet zloženia destilátu

$$\frac{dx_d}{dt} = \frac{\dot{n}_G}{H_{kond}} y_{n+1}(t) - \frac{(\dot{n}_D + \dot{n}_L)}{H_{kond}} x_d(t) \quad (51)$$

Zloženie kvapalnej fázy v obohacovacej časti

$$\frac{dx_i}{dt} = \frac{\dot{n}_G}{H_{etáž}} (y_{i-1}(t) - y_i(t)) + \frac{\dot{n}_L}{H_{etáž}} (x_{i+1}(t) - x_i(t)) \quad (52)$$

Zloženie parnej fázy v obohacovacej časti

$$y_i = \eta_i y_i^* + (1 - \eta_i) y_{i-1} \quad (53)$$

Bilancia nástrekovej etáže

$$\frac{dx_{k+1}}{dt} = \frac{\dot{n}_F}{H_{etáž}} x_F(t) + \frac{\dot{n}_L}{H_{etáž}} x_{k+2}(t) - \frac{(\dot{n}_F + \dot{n}_L)}{H_{etáž}} x_{k+1}(t) \quad (54)$$

$$+ \frac{\dot{n}_G}{H_{etáž}} (y_k(t) - y_{k+1}(t))$$

Zloženie parnej fázy nástrekovej etáže

$$y_{k+1} = \eta_k y_{k+1}^* + (1 - \eta_k) y_k \quad (55)$$

Zloženie kvapalnej fázy v ochudobňovacej časti

$$\frac{dx_j}{dt} = \frac{\dot{n}_G}{H_{etáž}} (y_{j-1}(t) - y_j(t)) + \frac{(\dot{n}_F + \dot{n}_L)}{H_{etáž}} (x_{j+1}(t) - x_j(t)) \quad (56)$$

Zloženie parnej fázy v ochudobňovacej časti

$$y_j = \eta_j y_j^* + (1 - \eta_j) y_{j-1} \quad (57)$$

Zloženie kvapalnej fázy zvyšku vo varáku

$$\frac{dx_1}{dt} = \frac{(\dot{n}_F + \dot{n}_L)}{H_{varák}} x_2(t) - \frac{\dot{n}_G}{H_{varák}} y_1(t) - \frac{\dot{n}_W}{H_{varák}} x_1(t) \quad (58)$$

Zloženie parnej fázy zvyšku vo varáku

$$y_1 = y^* \quad (59)$$

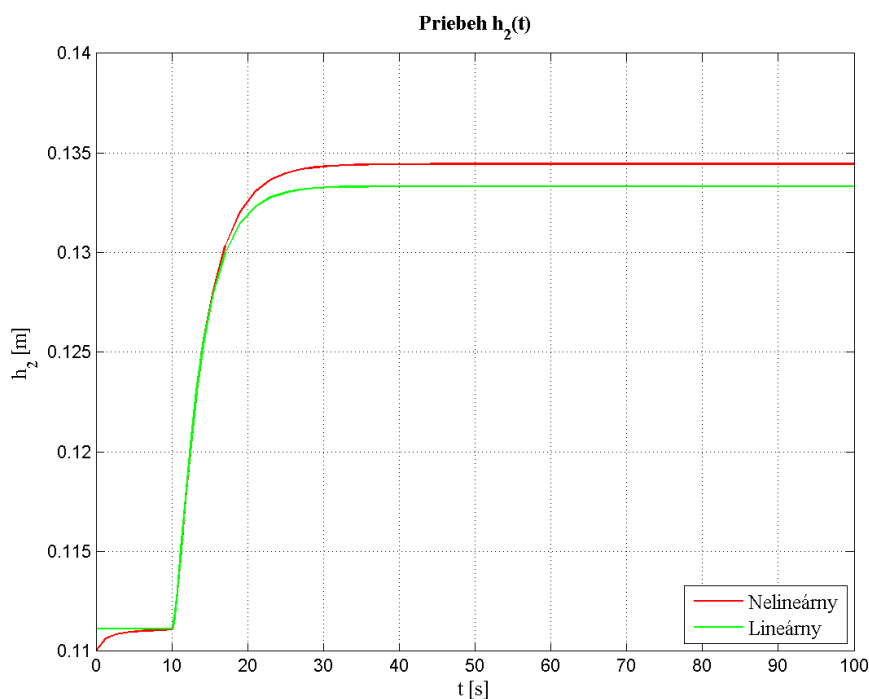
2.3 Teoretické základy vybraných metód riadenia

Riadiť proces je možné viacerými spôsobmi. Riadenie sa zväčša navrhuje pomocou identifikácie procesu ako prenosovej funkcie medzi vstupnou riadiacou veličinou u a sledovaným výstupom y .

Prenos medzi vstupnou a výstupnou veličinou:

$$G = \frac{u(s)}{y(s)} = \frac{b(s)}{a(s)} \quad (60)$$

Identifikácia prenosu prebieha v tzv. pracovnom bode. Takto získavame lineárny model procesu v okolí pracovného bodu. V praxi je však veľmi málo procesov lineárnych, preto je potrebné vedieť riadiť nelineárne procesy. Rozdiel medzi identifikovaným lineárnym a nelineárnym v okolí pracovného bodu je na Obr. 10.



Obr. 10: Porovnanie lineárneho (identifikovaného) a nelineárneho systému

2.3.1 PID regulátor

Proporcionálno integračno derivačný (PID) je typ spätnoväzbového regulátora, ktorý je často používaný v priemyselnej automatizácii. Samotný PID regulátor

nezabezpečuje stabilitu systému alebo optimálne riadenie. Je možné ho však efektívne používať na riadenie na žiadanú hodnotu alebo na odstránenie regulačnej odchýlky v okolí pracovného bodu. PID regulátor je stabilný v okolí pracovného bodu, na základe ktorého bol navrhnutý.

Odstránenie alebo minimalizácia regulačnej odchýlky, ktorá je rozdielom meranej a žiadanej veličiny sa dosahuje kombináciou 3 konštantných parametrov t.j. proporcionálnej (P), integračnej (I) a derivačnej (D) zložky regulátora. Charakteristiky zložiek PID regulátora:

- P – zosilnenie slúži na odstránenie regulačnej odchýlky v danom momente, zosilňuje meranú veličinu
- I – integrátor akumuluje predošlé odchýlky, zabezpečuje rýchlejšie dosiahnutie žiadanej veličiny a zároveň dokáže trvale odstrániť regulačnú odchýlku
- D - derivátor predikuje budúcu hodnotu odchýlky, teda odstraňuje nežiaduce prekmity meranej veličiny okolo žiadanej

Vážená suma zložiek PID regulátora reprezentuje akčný zásah prostredníctvom ventilu alebo iného inštrumentu na dosiahnutie požadovanej hodnoty meranej veličiny.

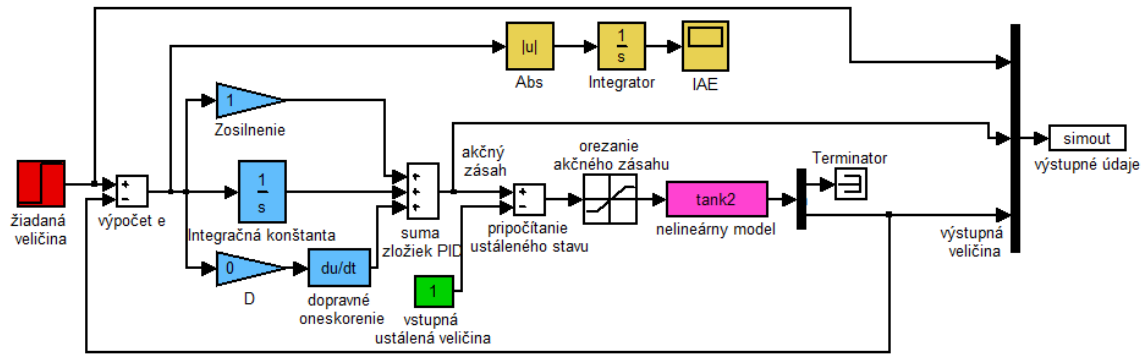
$$u(t) = Pe(t) + I \int_0^t e(t)dt + D \frac{de(t)}{dt}, \quad \text{kde} \quad P = Z_R, \quad I = \frac{Z_R}{T_I}, \quad (61)$$

$$D = Z_R T_D$$

$$u(t) = Z_R e(t) + \frac{Z_R}{T_I} \int_0^t e(t)dt + Z_R T_D \frac{de(t)}{dt} \quad (62)$$

Pričom nie je nutné používať všetky 3 zložky regulátora. Najčastejšou kombináciou parametrov regulátora je PI, keďže derivačná zložka regulátora je veľmi citlivá na šum merania.

Na Obr. 11 je znázornené schematické zapojenie PID regulátora v uzavretom regulačnom obvode v programe Simulink. Regulátor je reprezentovaný zložkami P, I a D. Vstupom do regulátora je odchýlková veličina, ktorú získame ako rozdiel žiadanej veličiny a výstupnej veličiny. K odchýlkovej veličine sa pripočítava ustálený stav vstupnej veličiny. Výstupom z regulátora je akčný zásah, ktorý vchádza do nelineárneho modelu procesu. [5]



Obr. 11: URO s PID regulátorom a nelineárnym systémom

2.3.2 Polynomicke vstupnovýstupné regulátory

[5], [11]

Polynomicke návrh regulátorov pomocou metódy PA (Pole Assignment) vychádza z Diofantickej rovnice

$$a(s)p(s) + b(s)q(s) = c(s) \quad (63)$$

Prenos spätnoväzbového regulátora:

$$Q = \frac{q(s)}{p(s)} \quad (64)$$

kde polynómy $a(s)$, $b(s)$ sú polynómy prenosu riadeného systému (60) s reálnymi koeficientmi. Polynómy regulátora $p(s)$, $q(s)$ sú neznáme. Diofantická rovnica je riešiteľná, ak najväčší spoločný deliteľ $a(s)$ a $b(s)$ je polynóm $c(s)$. Z toho vyplýva, že ak $a(s)$ a $b(s)$ sú nesúdeliteľné, diofantická rovnica je riešiteľná pre ľubovoľný polynóm $c(s)$. Pre stupne polynómov $p(s)$, $q(s)$ a $c(s)$ platí

$$\deg(p(s)) < \deg(b(s)) \quad (65)$$

$$\deg(q(s)) < \deg(a(s)) \quad (66)$$

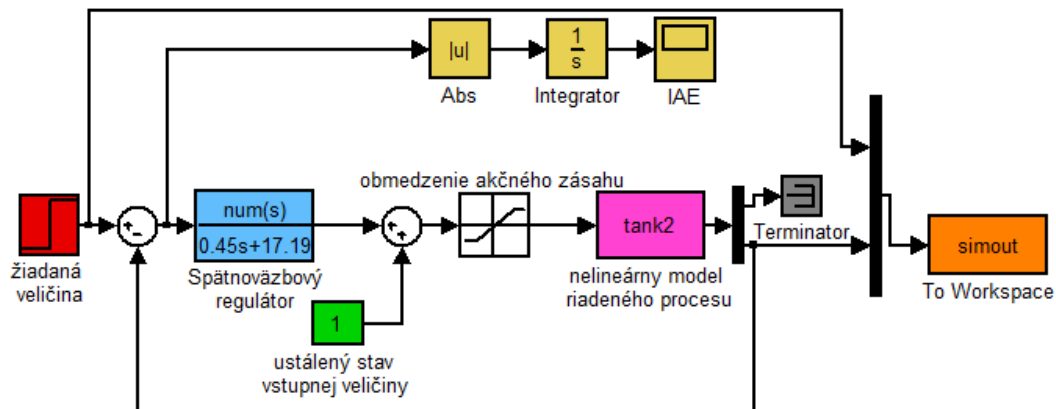
$$\deg(a(s)) + \deg(b(s)) \geq \deg(c(s)) \quad (67)$$

Pokiaľ platí nerovnosť (67), diofantická rovnica má riešenie minimálneho stupňa, a teda platí

$$\deg(p(s)) = \deg(q(s)) \quad (68)$$

Na riešenie diofantických rovníc existuje funkcia, ktorá je súčasťou Polynomickeho toolboxu $[p,q] = axbyc(a,b,c,stupen)$. Pokiaľ parameter *stupen* nie je uvedený, hľadá sa riešenie najnižšieho stupňa. [12] Polynóm $c(s)$ je žiadaný stabilný charakteristický polynóm uzavretého regulačného obvodu. Polynómy $p(s)$, $q(s)$ PA regulátora s jedným stupňom voľnosti sú riešením diofantickej rovnice (63).

Príslušná schéma riadenia nelineárneho systému s PA regulátorom s jedným stupňom voľnosti je na Obr. 12.



Obr. 12: Schéma URO s PA regulátorom 1DoF

Polynómy $p(s)$, $q(s)$ PA regulátora s jedným stupňom voľnosti s integračnou zložkou sú riešením diofantickej rovnice:

$$a(s)sp(s) + b(s)q(s) = c(s) \quad (69)$$

Schéma riadenia nelineárneho systému s PA regulátorom s jedným stupňom voľnosti s integračnou zložkou je na Obr. 13.

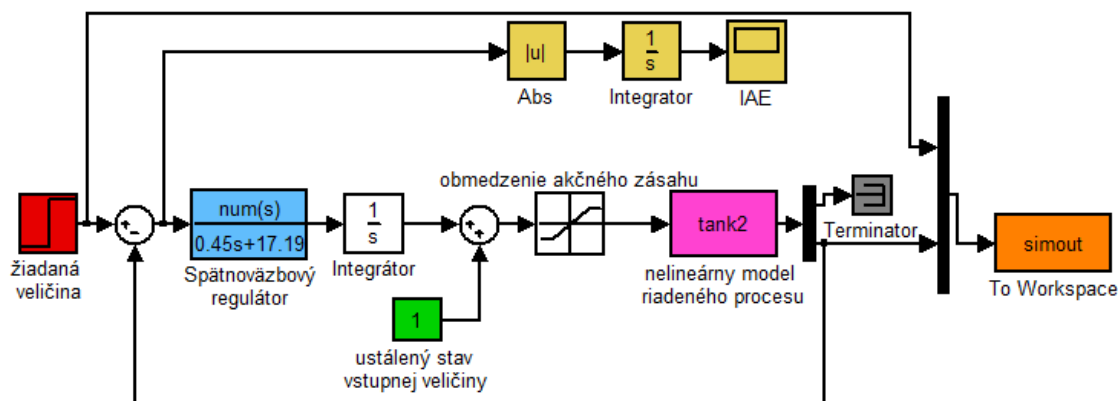
Riadenie s dvomi stupňami voľnosti sa skladá zo spätnoväzbového (64) a dopredného regulátora s prenosom

$$R = \frac{r(s)}{p(s)} \quad (70)$$

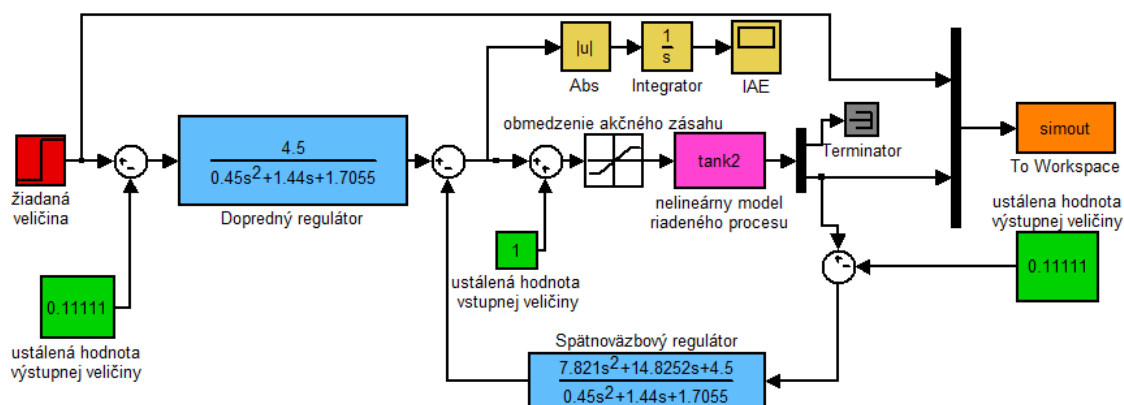
ktorého polynóm $r(s)$ získame riešením diofantickej rovnice

$$st(s) + b(s)r(s) = d(s) \quad (71)$$

Schéma riadenia nelineárneho systému s PA regulátorom s dvomi stupňami voľnosti je na Obr. 14.



Obr. 13: Schéma URO s PA regulátorom 1DoF s integračnou konštantou



Obr. 14: Schéma URO s PA regulátorom 2DoF

2.4 Prehľad použitých technológií

Virtuálne laboratórium v zmysle tejto práce je rozhranie, kde môže užívateľ modelovať rôzne procesy chemickej technológie prostredníctvom internetu. Užívateľ zadáva do formulára na webovej stránke simulačné údaje modelovaného procesu, ktoré sa spracujú na pozadí a užívateľ priamo dostáva výstupné dáta, resp. grafickú reprezentáciu dát. Táto časť práce vysvetľuje komunikáciu, a teda spracovanie dát medzi virtuálnym laboratóriom a simulačným programom.

Podkapitola 2.4.1 obsahuje úvod do značkovacieho jazyka HTML a taktiež popisuje formulárové prvky dokumentu, kde sa užívateľ vloží simulačné údaje vo webovom prehliadači.

Kapitola 2.4.2 vysvetľuje spôsob odosielania a prijímania simulačných dát.

Posledná časť prehľadu je venovaná charakteristike simulačného programu MATLAB a využitiu funkcií v ňom.

2.4.1 Formulár simulačných údajov

Webová stránka je dokument (napr. súbor typu .html) umiestnený na webovom serveri. Užívateľ k nej pristupuje prostredníctvom webového prehliadača, kde zadáním url adresy, požiadá o zaslanie tohto dokumentu (tým, že zadá v prehliadači jeho adresu). Formulár určený na vkladanie údajov užívateľom je reprezentovaný dynamickou webovou stránkou.

2.4.1.1 HTML značky

Webové prehliadače interpretujú obsah HTML dokumentov prostredníctvom HTML značiek, tzv. tagov. Hypertextový značkový jazyk slúži na definíciu štruktúry webovej stránky. Značky jazyka HTML sa používajú na rozdelenie webovej stránky na tzv. bloky. Jazyk HTML poskytuje možnosť rozčleniť obsah webovej stránky pomocou nadpisov, odsekov, hypertextových odkazov, tabuliek a podobne. Do dokumentov HTML je možné vložiť aj skripty jazyka JavaScript, ktoré poskytujú veľa možností pri spracovaní dát z formulára. Správcom špecifikácie jazyka HTML je organizácia W3C. Aktuálna verzia HTML je 5. Pre účely práce však bola použitá verzia HTML 4.01 Transitional.

Základná štruktúra HTML dokumentu sa skladá z niekoľkých kľúčových častí zahrňujúcich elementy a ich atribúty, znakové dátové typy, znakové referencie a iné odkazy. Väčšina používaných HTML značiek sú párové. Párové značky opisujú ako bude zobrazený objekt , ktorý sa nachádza medzi otváracou (napr. <a>) a zatváracou () značkou.

Nepárové značky neopisujú zobrazenie objektu, ale v mieste volania nejaký objekt na stránku zavedú. Vo formulári vstupných údajov boli použité párové aj nepárové značky.

Formulár vytvorený jazykom HTML dokáže čítať WS, ktorý údaje predá programu MATLAB a ten spustí simuláciu alebo výpočet. Po prebehnutí simulácie sa spätný dokument generuje v m-file (súbor obsahujúci zápis príkazov programu MATLAB s príponou .m) alebo sa do vytvorenej šablóny zapíše údaje z m-file prostredníctvom WS.

2.4.1.2 JavaScript

[15]

Na ošetrenie správneho formátu zadáných údajov na strane užívateľa, teda ešte pred odoslaním formulára do WS, bol použitý jazyk JavaScript. JavaScript je skriptovací jazyk, ktorého činnosť zabezpečuje prehliadač webových stránok návštevníka stránky a nie server, na ktorom sú stránky uložené. [13]

JavaScript dodáva stránke dynamiku, typickým využitím je spustenie skriptu pri aktivovaní udalosti napr. vloženie textu do textového poľa alebo kliknutie na tlačidlo umiestnené vo webovom dokumente. Táto vlastnosť umožňuje vykonať kontrolu vložených simulačných údajov do formulára pred odoslaním do WS.

Kódy napísané v jazyku JavaScript sú vložené medzi otváraciu `<script>` a zatváraciu značku `</script>`. Môžu byť priamo súčasťou HTML dokumentu, resp. sa môžu volať z externého súboru s príponou `.js`. Je potrebné však spresniť atribúty `<script language="JavaScript" type="text/JavaScript"></script>`, kvôli štandardom W3C. V prípade uloženia funkcií v externom súbore sa tento súbor volá príkazom `<script language="JavaScript" src="nazov_saboru.js" type="text/JavaScript">`.

Funkcie programované pomocou jazyka JavaScript nie sú aktívne, pokiaľ užívateľ postupuje správne pri zadávaní dát do formulára simulačných údajov. Ak však zadá znak, písmeno, medzeru alebo inú hodnotu ako číslo, pri pokuse odoslať formulár sa spustí JavaScript (Obr. 19). Užívateľ bude upozornený výstražným oknom, v ktorom sú uvedené nesprávne zadane hodnoty. Týmto spôsobom sa predchádza tomu, aby sa odosieli údaje nesprávneho formátu do simulačného programu.

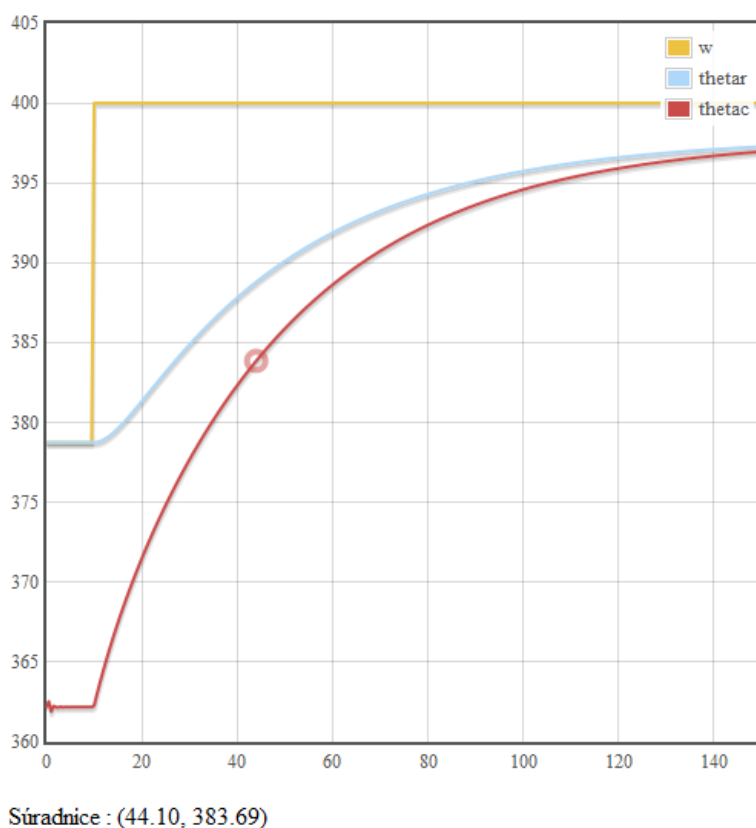
2.4.1.3 jQuery a grafy

jQuery je knižnica funkcií jazyka JavaScript, ktorá obsahuje vopred vytvorené funkcie pre zjednodušenie práce programátorov. Funkcie z knižnice je možné nalinkovať priamo vo webovom dokumente ako odkaz v hlavičke

```
<script src="//ajax.googleapis.com/ajax/libs/jquery/1.9.1/jquery.min.js"></script>
```

Odkaz sa môže viazať aj na stiahnutý súbor *jquery.min.js*. Knižnicou jQuery sa práca zaoberá z dôvodu, že grafy získané pomocou simulačného programu MATLAB je možné vykresľovať na webovú stránku iba ako obrázok.

Často je však potrebné zistiť presnú polohu dát pri odčítavaní z grafu, preto pre účely práce bola použitá knižnica funkcii JavaScript tzv. Flot. Knižnica Flot obsahuje súbor funkcii, ktoré na základe databázy zo simulačného programu MATLAB dokážu vykresliť na strane užívateľa graf. Takýto graf je možné približovať a oddiaľovať a taktiež sa zobrazujú súradnice kurzora pri prechode grafom (Obr. 15). [14]



Obr. 15: Příklad dynamicky generovaného grafu pomocou Flot

2.4.2 Matlab Web Server a Web Server

Predanie odoslaných dát z formulára simulačnému programu MATLAB zabezpečuje server WS. Ku komunikácii medzi serverom a webovým prehliadačom dochádza vďaka tzv. IP (Internet Protocol). HTTP je základným komunikačným protokolom internetu. Slúži na výmenu alebo transfer hypertextových dokumentov, teda HTML dokumentov. Tento typ dokumentu dokáže čítať WS, a preto sa prostredníctvom neho odosielať simulačné údaje programu MATLAB.

V prípade odosielania HTML dokumentu webovému prehliadaču je potrebné zabezpečiť správne zobrazovanie diakritiky. Pre tieto účely slúžia podporné informácie tzv. metadáta. Webový prehliadač ako HTTP klient nadväzuje komunikáciu s WS prostredníctvom TCP (Transmission Control Protocol). WS následne predá údaje programu MATLAB, kde je možné vykonať výpočty alebo simulácie. [15]

Do septembra roku 2006 bolo možné pre tento typ prepojenia použiť toolbox Matlab Web Server od firmy Mathworks. [16]

Matlab Web Server v priamom slova zmysle nebol serverom, pracoval ako CGI aplikácia v spolupráci s web serverom APACHE, ktorý podporuje CGI.

Matlab Web Server bol štandardne voliteľnou súčasťou inštalácie programu MATLAB verzie 5.3 až verzie 7.2 distribuovaného firmou MathWorks. Novšie verzie ako MATLAB 7.2 však už nepodporujú Matlab Web Server. Miesto toho firma MathWorks poskytla užívateľom možnosť vytvorenia vlastnej aplikácie pre komunikáciu s webom. Počnúc teda verziou programu MATLAB 7.3 je podporovaných niekoľko spôsobov vytvorenia spojenia medzi programom MATLAB a webom:

- MATLAB® Builder™ NE slúži na tvorbu doplnku COM alebo .NET na strane servera
- MATLAB® Builder™ JA slúži na tvorbu JAVA servletu alebo Web servisu na strane servera
- COM automation, CGI alebo iné

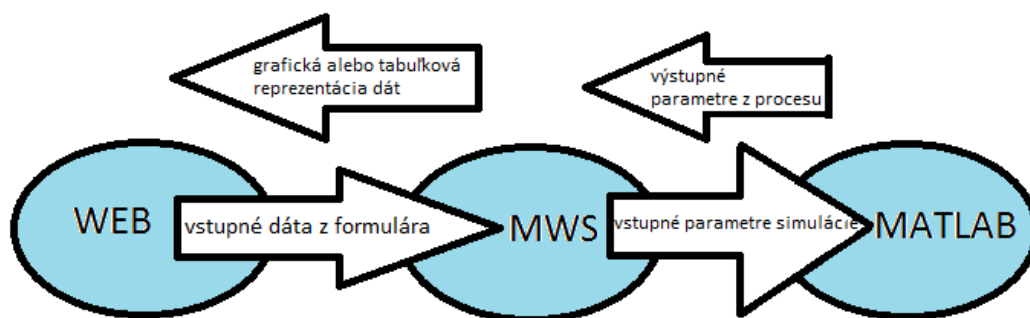
Dirk Jan Kroon vedecký pracovník z University Twente v Holandsku vyvinul v novembri roku 2010 Web Server, ktorý dokáže odosielať dáta z HTML formulára do programu MATLAB a spracovať m-file prostredníctvom, ktorých sa generuje stránka s výsledkami simulácii z programu MATLAB. [17]

Web Server je voľne dostupným HTTP serverom, ktorý je plne kompatibilný s verziou programu MATLAB 7.10 (R2010a) a novšími. Je schopný prijať dáta zasielané z HTML formulára internetových prehliadačov. Následne ich predať ako parametre obslužnému skriptu/programu (m-file) a výstup týchto skriptov odoslať vo forme HTML stránky aj s podporou zobrazovania obrázkov späť užívateľovi.

Na Obr. 16 je zjednodušená ilustrácia komunikácie medzi webovým prehliadačom, WS a programom MATLAB.

WS je veľmi jednoducho inštalovateľný. Stačí si nakopírovať do ľubovoľného priečinka celý súbor WS, obsahujúci m-file, ktoré spracovávajú simulácie.

WS obsahuje spustiteľný m-file *webserver.m*, ktorý nadväzuje spojenie s webovým prehliadačom prostredníctvom programu MATLAB. Po zapnutí programu MATLAB je potrebné sa nastaviť do adresára programu Matlab Web Server a zadať do príkazového riadku *webserver(4000)*. Prednastavené hodnota portu je 4000, je však voliteľná.



Obr. 16: Schematické znázornenie činnosti WS

Následne sa zobrazí okno typu *figure* (Obr. 17), ktorého zatvorením by sa prerušilo spojenie web serveru a webového prehliadača. Počas spojenia je možné odosielať vyplnené dáta formulára do programu WS prostredníctvom webového prehliadača, kde sa spracujú v príslušnom m-file, na ktorý odkazuje formulár. Po spracovaní údajov sa zobrazia dáta vo webovom prehliadači, podľa definovaných parametrov v odkazovanom m-file.

Odporúčané webové prehliadače pre spoluprácu s WS sú Mozilla Firefox a Internet Explorer. Úpravy v programe MATLAB, resp. v m-file je odporúčané robiť, pokiaľ je WS vypnutý, resp. je vhodné ho reštartovať.



Obr. 17: Spustenie programu WS

2.5 MATLAB

MATLAB® je integrované prostredie pre vedeckotechnické výpočty, modelovanie, návrhy algoritmov, simulácie, analýzu a prezentáciu údajov, meranie a spracovanie signálov, návrhy riadiacich a komunikačných systémov. MATLAB je nástroj tak pre pohodlnú interaktívnu prácu, ako aj pre vývoj širokého spektra aplikácií. [18]

Názov MATLAB vznikol skrátením slov MATrix LABoratory (voľne preložené „laboratórium s maticami“). Kľúčovou dátovou štruktúrou pri výpočtoch v MATLABe sú matice a polia. Program MATLAB má vlastný programovací jazyk. Je to interaktívny jazyk, ktorého základný element je pole, ktoré nemusí užívateľ definovať.

Táto skutočnosť umožňuje riešiť technické výpočtové problémy a najmä tie, ktoré sú formulované vo vektorovom a maticovom tvare. [19]

Nadstavbou Matlabu je Simulink – program pre simuláciu a modelovanie dynamických systémov, ktorý využíva algoritmy programu MATLAB pre numerické riešenia predovšetkým nelineárnych diferenciálnych rovníc.

2.5.1 Simulink

Simulink je softvérový balík určený na modelovanie, simuláciu a analýzu dynamických systémov. Podporuje lineárne a nelineárne systémy, modely v spojitom čase, diskretnom alebo v kombinácii. [20]

Poskytuje interaktívne grafické prostredie a prispôsobiteľnú zostavu blokov, ktoré si užívateľ vyberá z tzv. blokovej knižnice. Možnosť skladania vlastného systému z blokov umožňuje simulovať, implementovať a testovať širokú škálu časovo premenlivých systémov, vrátane problematiky komunikácie, riadenia, spracovania signálu, videa, či obrázkov. Príklad schémy v Simulinku je na Obr. 11.

3 Praktická časť

Predchádzajúce kapitoly stručne definovali teoretické pozadie matematických modelov, ktoré boli implementované do jazyka programu MATLAB.

Praktická časť popisuje proces implementácie etážovej rektifikačnej kolóny do virtuálneho laboratória, počnúc vytvorením vstupného formulára dát cez vytvorenie funkcie v programe MATLAB, až po vykreslenie priebehu simulácie na webovej stránke na strane užívateľa. Kapitola obsahuje príklady simulácie vytvorenej knižnice modelov.

3.1 Formulár vstupných dát etážovej rektifikačnej kolóny

V kapitole 2.4.1.1 a 2.4.1.2 boli stručne popísané vybrané značky jazyka HTML a niektoré JavaScriptové funkcie. Táto kapitola je venovaná podrobnejšiemu vysvetleniu vlastností formulára v HTML dokumente s prvkami JavaScriptu. Na Obr. 18 je znázornený formulár, ktorý sa zobrazí užívateľovi, keď si v menu virtuálneho

laboratória vyberie simuláciu rektifikačnej kolóny. V štruktúre HTML dokumentu sa na začiatku vždy deklaruje o aký typ dokumentu ide:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
```

Základná štruktúra HTML dokumentu je

```
<html>
```

```
<head>
```

```
<meta http-equiv="content-type" content="text/html; charset=UTF-8">
```

```
<title></title>
```

```
</head>
```

```
<body>
```

```
</body>
```

```
</html>
```

V tzv. hlavičke v párových značkách `<head></head>` sa definuje kódovanie dokumentu, ktoré informuje prehliadač o tom, aká znaková sada je použitá vo formulári a ako ju má prečítať.

```
<meta http-equiv="content-type" content="text/html; charset=UTF-8">
```

Štandardnou znakovou sadou je UTF-8. V hlavičke HTML dokumentu sa môžeme odkazovať na rôzne typy externých súborov. Pre účely tejto práce sa formulár v hlavičke odkazuje knižnicu JQuery (vid' 2.4.1.3). V tzv. tele HTML dokumentu sa definuje hlavná funkcia HTML stránky. Samotný formulár je zahrnutý do párových značiek

```
<form action="html_form_action.asp" method="post" onsubmit="return  
kontrola(this);">  
</form>
```

kde *action* je odkaz na súbor, kde sa odošlú dáta z formulára na spracovanie. V tomto prípade sa formulár vždy odkazuje na m-file (viac v kapitole 2.4.2). Atribút *method* je metóda akou sa budú dáta z formulára odosielať. Formulárové dáta je možné odosielať ako URL premenné pomocou *method="get"* alebo ako HTTP pomocou *method="post"*. [21]

Simulácia ustálených stavov rektifikačnej kolóny

Tok látkového množstva nástreku n_F^S [kmol.h ⁻¹] :	<input type="text" value="22.5"/>
Tok látkového množstva spätného toku n_L^S [kmol.h ⁻¹] :	<input type="text" value="22.26"/>
Tok látkového množstva pary n_G^S [kmol.h ⁻¹] :	<input type="text" value="30"/>
Mólový zlomok v kvapalnej fáze x_F^S :	<input type="text" value="0.23"/>
Počet etáží :	<input type="text" value="7"/>
Poradie nástrekovej etáže :	<input type="text" value="4"/>
Účinnosť etáží v [%] :	<input type="text" value="0.6"/>
Zadrž kvapaliny na etážach v [%] :	<input type="text" value="0.1"/>
Účinnosť varáka v [%] :	<input type="text" value="1"/>
Zatopenie kondenzátora v [%] :	<input type="text" value="0.2"/>
Koeficienty rovnice rovnovážnej krivky:	
$y_{n+1}^*(t) = \frac{a + bx_{n+1}(t) + cx_{n+1}(t)^2}{1 + dx_{n+1}(t) + ex_{n+1}(t)^2}$	
a :	<input type="text" value="0.0021567456"/>
b :	<input type="text" value="2.3066176"/>
c :	<input type="text" value="-0.88823513"/>
d :	<input type="text" value="0.82112327"/>
e :	<input type="text" value="-0.39916116"/>
Čas simulácie t :	<input type="text" value="50"/> rozsah: (0,100)
Skoková zmena spätného toku :	<input type="text" value="22.26"/>
Čas skoku :	<input type="text" value="20"/>
Skoková zmena toku parnej fázy :	<input type="text" value="30"/>
Čas skoku :	<input type="text" value="20"/>
<input type="button" value="Odoslať"/> <input type="button" value="Príklad"/>	

Obr. 18: Formulár vstupných údajov simulácie etážovej rektifikačnej kolóny

JavaScriptová funkcia *kontrola()* sa aktivuje iba pri kliknutí na tlačidlo *Odoslať* (Obr. 18), t.j. keď užívateľ chce odoslať dáta.

```
function kontrola(formular)

{missinginfo = "";

    if ((formular.nfs.value == "") || isNaN(formular.nfs.value))

        missinginfo += '\n - Tok látkového množstva nástreku';

    if (missinginfo != "")

        {missinginfo = 'Položky formulára neboli vyplnené, alebo nie je správny ich
formát!\n';

            alert(missinginfo);

            return false;}

    else

        return true;}

</script>
```

Funkcia *formular.nfs.value == ''* v rámci JavaScriptovej funkcie *kontrola()* overuje, či je vyplnené políčko s názvom *nfs*. Názov premennej *nfs* sa vzťahuje na políčko, kde sa daná premenná zadáva. JavaScriptová funkcia *isNaN()* slúži na overenie, či zadané číslo je naozaj výlučne číslo a nie písmeno alebo iný znak.

Názov premennej políčka sa získava ako atribút *name* nepárovej značky *input*. Značka *input* práca využíva ako textové pole (*type="text"*) o veľkosti *size="5"* pixelov a taktiež ako tlačidlo *Odoslať* (*type=" submit "*). Atribút *value* obsahuje hodnotu, ktorá je prednastavená pre políčko hneď pri načítaní stránky. Na Obr. 18 je vidno tlačidlo *Príklad*, ktoré slúži na znovu vloženie prednastavených hodnôt políčok, v prípade ak užívateľ prepísal niektoré políčko a želá si vyskúšať prednastavený príklad simulácie.

Príklad textového poľa toku látkového množstva nástreku:

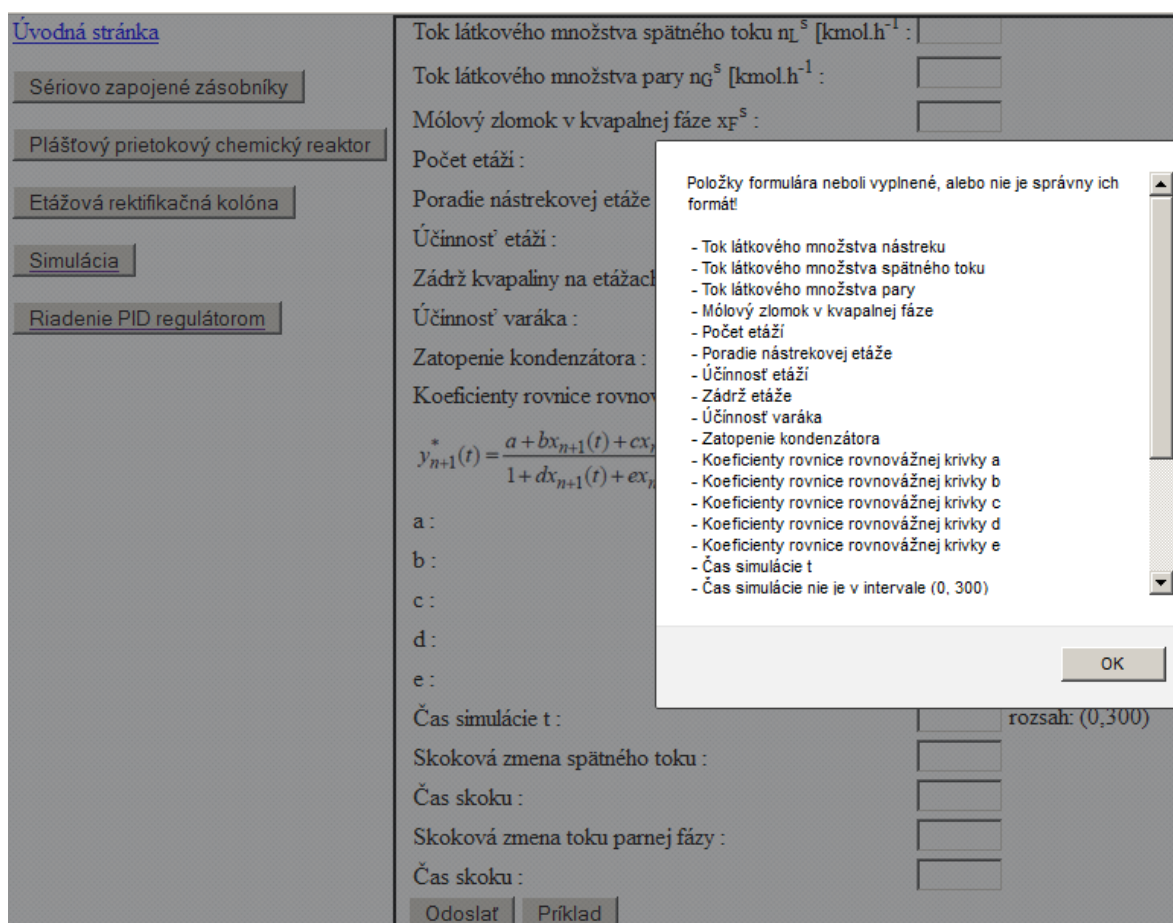
```
<input type="text" class="text" name="nfs" size="5" value="22"
onfocus="this.value=""/>
```

Príklad tlačidla Odoslať:

```
<input type="submit" value="Odoslať" />
```

Funkcia *alert(missinginfo)* zobrazí pop-up okno, ktoré upozorní užívateľa na nesprávne zadané údaje ako náhle sa ich bude snažiť odoslať kliknutím na tlačidlo *Odoslať* (Obr. 19).

Správnosť zadaných údajov sa teda overuje nielen pomocou funkcie *kontrola()* ale aj pomocou funkcií programovaných v jazyku MATLAB. Tejto problematike sa bude venovať nasledujúca kapitola.



Obr. 19: Upozornenie užívateľa na nesprávnosť zadaných údajov

3.2 M-file a spracovanie simulačných údajov

MATLAB obsahuje veľa možností na vytváranie funkcií, ktoré uľahčujú prácu s ním. Napríklad ak sa vykonáva viackrát tá istá operácia je vhodné vytvoriť si funkciu, kde sa zadajú vstupné parametre, telo funkcie obsahuje príslušné výpočtové príkazy a následne získate v želanom formáte.

Všeobecným príkladom funkcie je

```
function [variables_out]=function_name(variables_in)
```

```
N=5;
```

```
for i=1:N
```

```
variables_in= variables_in^2;
```

```
end
```

```
variables_out=variables_in;
```

```
end
```

3.2.1 Formát vstupných a výstupných dát z m-file WS

Ďalším príkladom funkcie v programe MATLAB je funkcia, pomocou ktorej sa načítavajú údaje predané WS z webového formulára.

```
function html=load_data(headers,config)
```

```
html='<html><head>';
```

```
html=[html '</head><body>' ];
```

```
in=headers.Content;
```

```
html=[html '</body></html>'];
```

Do premennej *html* sa ukladá obsah HTML dokumentu, ktorý sa má po prebehnutí simulácie a výpočtov zobrazit' užívateľovi.

Značky `<body></body>` majú rovnakú funkciu ako v HTML dokumente. V hlavičke `<head>'</head>` je možné sa odkazovať na JavaScript alebo jQuery súbory.

3.2.2 Telo m-file a výpočtové funkcie

V rámci tejto práce sa pomocou premennej *html* zobrazuje text, hodnoty, obrázky a grafy.

Reťazec, teda text alebo číslo je potrebné pred vloženíím do „zátvoriek“ HTML (*html=[html 'TEXT']*) previesť na text. Prevod reťazca na text je možný niekoľkými funkciami programu MATLAB. Napríklad na prevod čísla na text je vhodná funkcia *num2str*. Pre správne zobrazovanie textu s diakritikou je potrebné nahradiť znaky s diakritikou v pôvodnom texte za HTML entity.

Funkcia *make_unicode* na nahradenie znakov s diakritikou obsahuje funkciu *regexprep(text 1, text 2, text 3)* programu MATLAB, ktorá v *texte 1* nahradí *text 2* *textom 3*.

Príklad nahradenia znaku veľké Š za HTML entitu *&Scaron*::

```
function htmlString = make_unicode(htmlString)

    htmlString = regexprep(htmlString,'Á','&Aring;');

    ...

    htmlString = regexprep(htmlString,'Š','&Scaron;');

    ...

end
```

Ako vstup do funkcie *make_unicode* je napríklad reťazec *tok látkového množstva nástreku* v tvare *výstup=make_unicode(' tok látkového množstva nástreku ')* a výstup získame do premennej *výstup*. Premenná *výstup* sa na stránke zobrazí ak sa zadá v tvare *html=[html výstup]*.

Ďalšími výstupmi na stránke môžu byť všetky HTML značky v tvare `html=[html '']`. Značka `img` je nepárová značka používaná na zavedenie obrázku na stránku. Zobrazovaniu grafov ako obrázkov a možnosti zobrazovania grafov pomocou HTML a Flot je venovaná kapitola 3.3.

Vstupná premenná `headers` obsahuje všetky dáta odoslané z webového formulára ako reťazce. Obsah premennej `headers` sa odovzdá pomocou funkcie `Content` do premennej `in`, ku ktorej je možné objektovo pristupovať.

V predchádzajúcej kapitole je ako príklad dátového poľa, ktoré užívateľ vyplňa uvedená premenná `toku látkového množstva nástreku` s hodnotou 22.5. Teda hodnotu `toku látkového množstva nástreku` sa z formulára získava ako `nfs=str2num(in.nfs)`, kde funkcia `str2num` je preddefinovaná funkcia programu MATLAB.

V prípade, ak by overenie zadaných údajov pomocou JavaScriptu zlyhalo, obsahuje m-file overenie, či vôbec boli nejaké dáta odoslané. Na overenie obsahu premenných obsahuje MATLAB takisto preddefinovanú funkciu `isempty(premenna)`. Ak je obsah premennej prázdny, MATLAB vráti výsledok 1, ak premenná obsahuje nejaké dáta vráti hodnotu 0.

Nižšie je uvedený príklad na overenie, či bol `tok látkového množstva nástreku` vyplnený. V prípade, že je hodnota prázdna webový prehliadač zobrazí dokument, kde bude vypísané *Musíte vyplniť všetky položky vo formulári* a zároveň sa v m-file nastaví hodnota pomocnej premennej `errorcode` na hodnotu 1, čo znamená, že výpočtové funkcie v tele m-file neprebehnú, a tým sa predídde chybovým hláseniam. Na začiatku m-file je samozrejme hodnota `errorcode` nastavená na 0, teda nie je detekovaná žiadna chyba a výpočet môže prebehnúť.

```
errorcode = 0;
```

```
if (isempty(in.nfs) )
```

```
html=[html 'Musíte vyplniť všetky položky vo formulári.' '<br>'];
```

```
errorcode = 1;
```

```
elseif ((str2num(in.tsim) < 0) | (str2num(in.tsim) > 300))
```

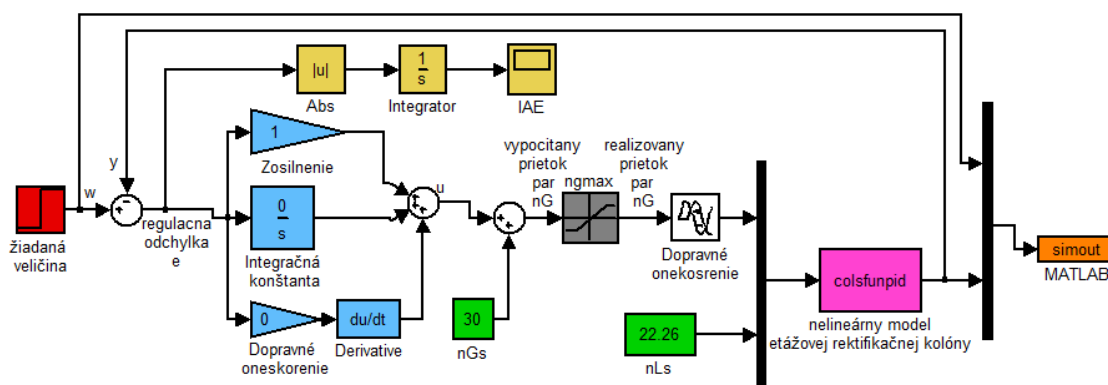
```
html=[html 'Čas nie je v intervale (0, 300)' '<br>'];
```

```
errorcode = 1;
```

```
end
```

3.2.3 Simulácia v Simulinku pomocou m-file

Parametre schémy zostavenej v programe Simulink môžeme zadať priamo do blokov, načítať ich z tzv. Workspace (premenná zadaná v príkazovom riadku programu MATLAB) alebo ich nastaviť z m-file pomocou funkcie programu MATLAB *set_param*.



Obr. 20: Schéma riadenia rektifikačnej kolóny pomocou PID regulátora

Nižšie je uvedený súbor ilustračných príkazov, ktoré nastaví príslušné parametre bloku žiadaná veličina v schéme na Obr. 19. Názov bloku je *scheme_name='PIDcol'*. Funkcia *open_system(scheme_name)* je preddefinovaná v programe MATLAB a slúži na otvorenie schémy za účelom zápisu parametrov.

```
scheme_name='PIDcol';
```

```
open_system(scheme_name);
```

Funkcia *sprintf('%s/required',scheme_name)* vytvorí reťazec *scheme_name/required*, čo je fyzická cesta k bloku.

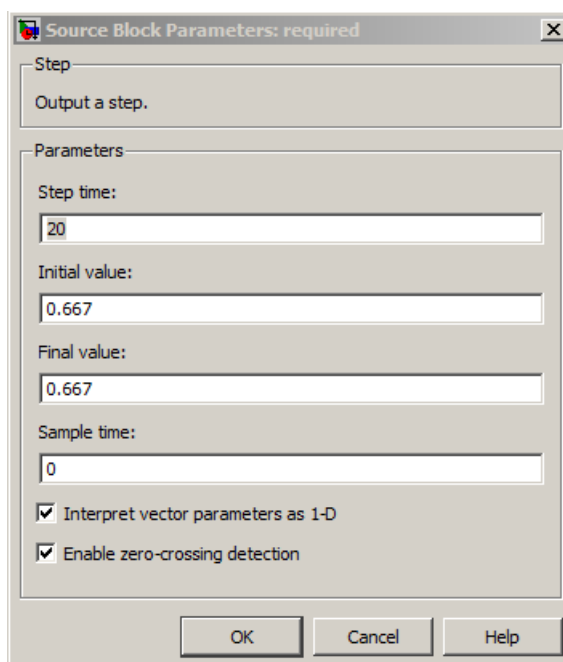
```
wx= sprintf('%s/required',scheme_name);
```

Funkcia *set_param* nastaví tri rôzne parametre bloku a to 'Time', 'Before', 'After', teda čas skokovej zmeny, hodnotu pred skokovou zmenou a hodnotu po skokovej zmene. Tieto tri parametre prislúchajúce bloku na Obr. 21.

```
set_param(wx,'Time',sprintf('%s',num2str(w_time)));
```

```
set_param(wx,'Before',sprintf('%s',num2str(w_init)));
```

```
set_param(wx,'After',sprintf('%s',num2str(w_final)));
```



Obr. 21: Blok skokovej zmeny v Simulinku

3.3 Flot a grafy

Ako sme uviedli v kapitole 3.2.1, HTML značky umožňujú zobrazovanie obrázkov, resp. grafov na webovej stránke. Presné odčítanie veličín je veľmi dôležité pri simuláciách nielen technologických procesov. Uvažujme výstupy z programu MATLAB ako dáta alebo grafy (uložené vo formáte png). Graf uložený ako obrázok nie je možné približovať alebo odčítavať z neho hodnotu. Z tohto dôvodu práca využíva možnosti knižnice JavaScriptových funkcií Flot.

V prípade použitia akejkoľvek JavaScriptovej funkcie je potrebné odkazovať sa na knižnicu jQuery. Knižnica jQuery tak ako aj v HTML dokumente, tak aj v m-file sa musí nachádzať v hlavičke v rámci `html='<html><head>';html=[html '</head><body>'];`

Vykreslenie grafu pomocou FLOT však vyžaduje ďalšie jQuery knižnice. [22]

Do hlavičky m-file teda zahrnieme nasledovné:

```
html='<html><head>';
```

```
html=[html '<script language="JavaScript" type="text/JavaScript"
src="../../js/jquery.min.js"></script>'];
```

```
html=[html '<script language="JavaScript" type="text/JavaScript"
src="../../js/excanvas.min.js"></script>'];
```

```
html=[html '<script language="JavaScript" type="text/JavaScript"
src="../../js/jquery.Flott.js"></script>'];
```

```
html=[html '<script language="JavaScript" type="text/JavaScript"
src="../../js/jquery.Flott.navigate.min.js"></script>'];
```

```
html=[html '</head><body>' ];
```

Do tela m-file medzi `html=[html '</head><body>'];` a `html=[html '</body></html>'];` sa vložia funkcie, ktoré vykresľujú FLOT graf zo simulačných dát.

Funkcia *Flotfig* slúži na dynamické generovanie názvov premenných v závislosti od rozmeru výstupu zo simulácie. Názvy premenných **varnam** sa následne vykreslia na stránke v podobe legendy. Premenná **comp** obsahuje dáta na vykreslenie v tvare $[t_1, y_1, t_2, y_2, \dots]$, kde t je čas a y závislá premenná.

Premenné *y_{max}*, *y_{min}* a *x_{max}* slúžia na definíciu okrajov grafu, aby bola zabezpečená dobrá čitateľnosť grafu. Pomocou premennej **placeholder** sa uvádza miesto, kde sa zobrazí graf. Možnosť približovania grafu určuje vlastnosť *zoom*: `{interactive: true}`. Približovať a oddaľovať graf je možné tzv. scrollovaním.

Súradnice kurzora pohybujúceho sa po grafe umožňuje vlastnosť *pan*: *{interactive: true}*. Obe tieto vlastnosti sú aktívne ak majú hodnotu *true*, v prípade ak majú hodnotu *false*, sú deaktivované.

[comp varnam ymax ymin xmax]=Flotfig(t,y)

```
html=[html '<script type="text/JavaScript" language="JavaScript">'];
```

```
html=[html 'function test(){$(function(){'};
```

```
html=[html '$("#placeholder").bind("plothover", function (event, pos, item)');
```

```
html=[html '{var str = "(" + pos.x.toFixed(2) + ", " + pos.y.toFixed(2) +  
"}";$("#simxy").text(str);}'}];
```

```
html=[html 'var placeholder = $("#placeholder");'];
```

```
html=[html 'var options = {series:{lines: {show: true}},xaxis: {zoomRange: [0, 'xmax];
```

```
html=[html '],panRange: [0, 'xmax];
```

```
html=[html '],yaxis:{zoomRange: ['ymin];
```

```
html=[html ','ymax];
```

```
html=[html '],panRange: ['ymin];
```

```
html=[html ','ymax];
```

```
html=[html ''],zoom: {interactive: true}, grid:{hoverable: true},pan: {interactive:  
true}}'];
```

```
html=[html comp];
```

```
html=[html ' $.plot(placeholder, ['varnam];
```

```
html=[html ' ],options)'];
```

```
html=[html ' }');'];
```

```
html=[html ' } '];
```

```
html=[html '</script>'];
```

```
html=[html '<script type="text/JavaScript"> test();</script>'];
```

```
html=[html '<div id="placeholder" style="width:500px;height:500px"></div>'];
```

```
suradnice=make_unicode('Súradnice : ');
```

```
html=[html '<p> ' suradnice '<span id="simxy"></span> </p>'];
```

Zobrazovanie súradníc vykonáva funkcia

```
html=[html '{var str = "(" + pos.x.toFixed(2) + ", " + pos.y.toFixed(2) +  
"}";$("#simxy").text(str);}'];
```

Rovnako ako je potrebné definovať polohu grafu, je tiež potrebné určiť, kde sa budú zobrazovať súradnice kurzora.

3.4 Overenie správnosti simulačných dát

Správnosť simulácie pri navrhnutom modeli je možné overiť výpočtom ustálených stavov výstupných veličín a následnou simuláciou. V prípade, že vstupom do simulácie sú ustálené hodnoty vstupných veličín a počiatočné hodnoty pre výstup sú taktiež ustálené hodnoty, mali by sme získať po simulácii konštantný priebeh Obr. 22.

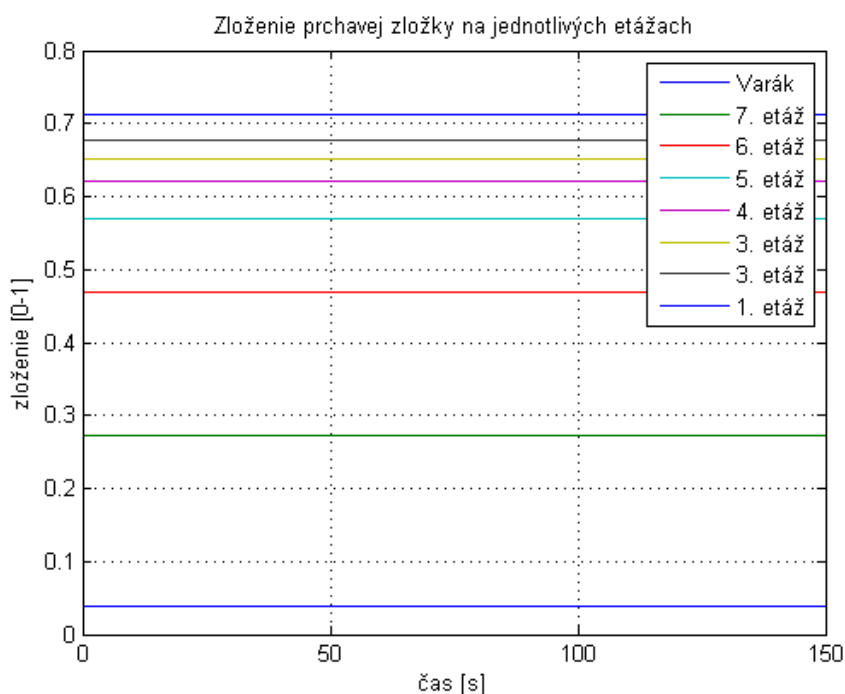
Zároveň, pokiaľ vykonáme skokovú zmenu vstupnej veličiny, v tomto prípade sa jedná o skokovú zmenu toku látkového množstva pár kolónu z hodnoty 22.26 kmol/hod na 25kmol/hod v čase 20s. Získavame priebeh simulácie na Obr. 23.

Týmto spôsobom boli overené simulácie všetkých modelov uvedených v tejto práci. Rovnakým spôsobom je možné overiť, či je zapojenie regulátora do uzavretého regulačného obvodu prevedené správne.

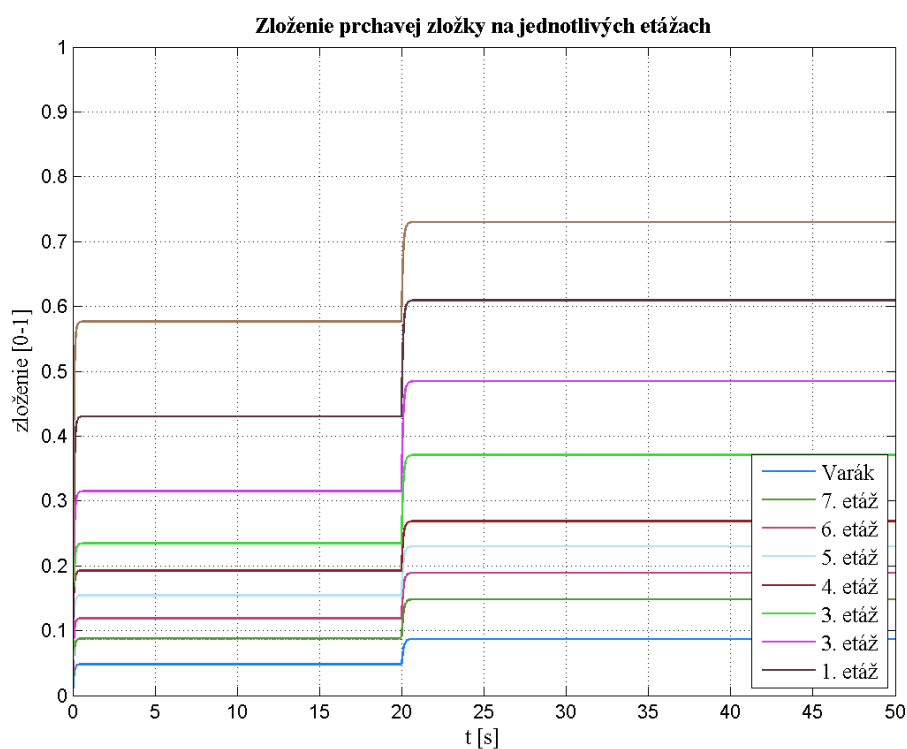
Uvedieme príklad overenia správneho zapojenia PID regulátora do obvodu s nelineárnym modelom rektifikačnej kolóny. Žiadaná veličina je teda ustálená hodnota zloženia destilátu 0.667. Na Obr. 24 je vidno, že počiatočné hodnoty kolóny sú nulové, avšak ustálená hodnota zloženia destilátu sa dosiahla veľmi rýchlo.

Priebeh je konštantný a regulátor nevykoná žiadny akčný zásah, pretože vstup do modelu je ustálená hodnota spätný tok látkového množstva, t.j. 30kmol/hod. Priebeh vstupnej veličiny je znázornený na Obr. 25.

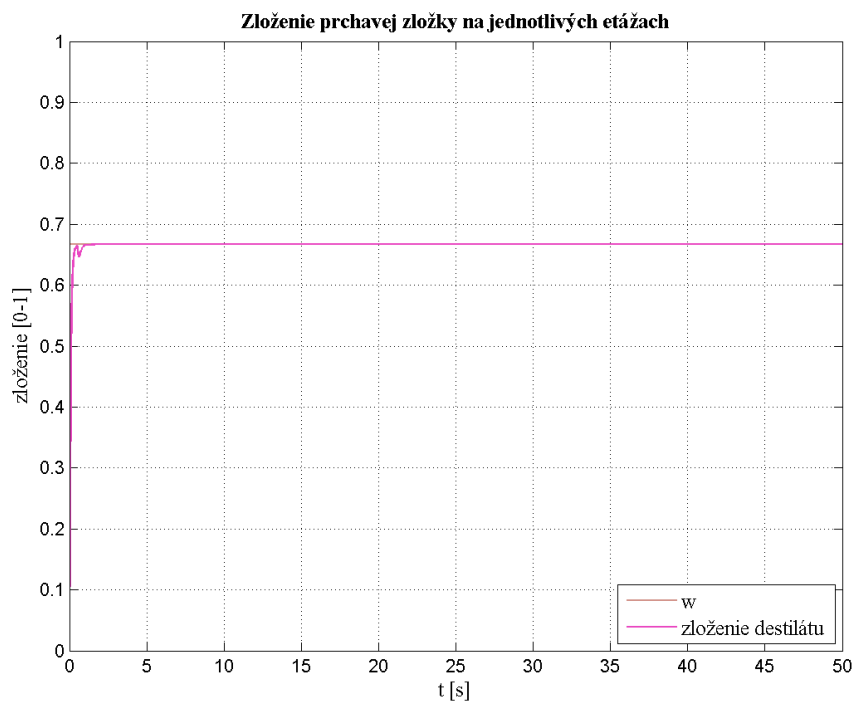
Na demonštráciu, že regulátor naozaj vykoná akčný zásah, ak sa žiadaná veličina zmení, uvádzame ten istý systém so žiadanou veličinou zloženia destilátu 0.8. Regulátor sme navolili PI s P zložkou v tvare $Zr = -10$ a I zložkou v tvare $Zr/Ti = 3$. Ako je zrejmé z Obr. 26, pri skokovej zmene v čase 20 s sa regulátor naozaj snaží dosiahnuť žiadanú veličinu. I zložka zabezpečuje, že trvalá regulačná odchýlka je minimálna.



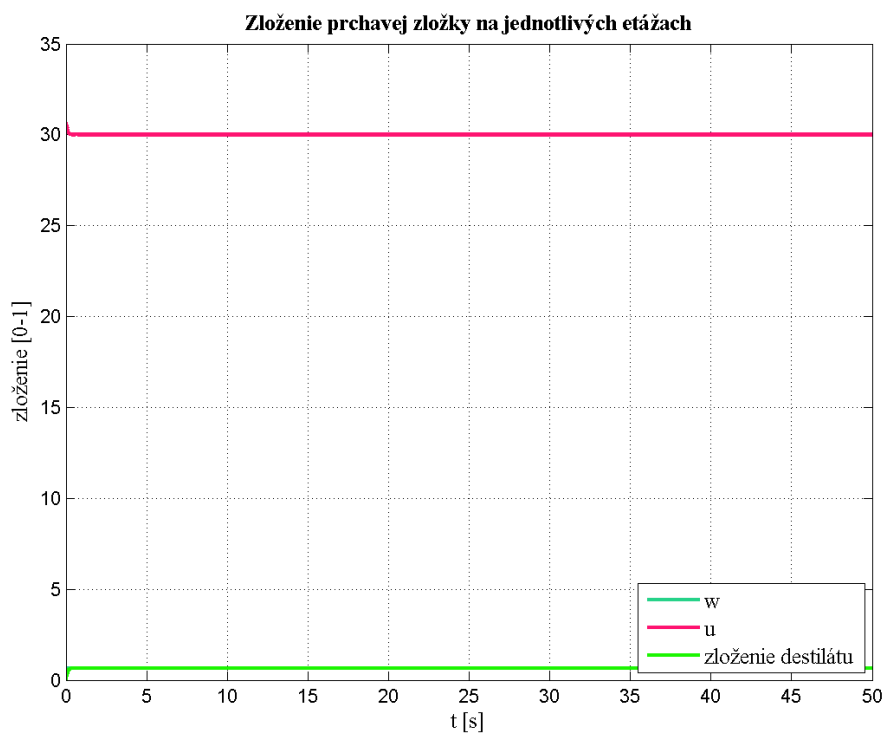
Obr. 22: Simulácia na overenie ustálených stavov rektifikačnej kolóny



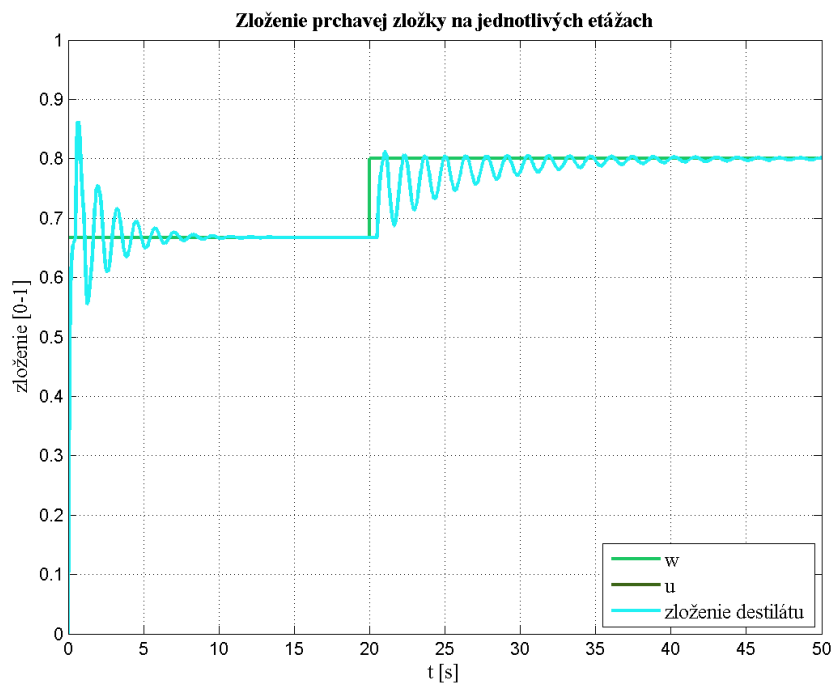
Obr. 23: Simulácia skokovej zmeny v rektifikačnej kolóne



Obr. 24: Simulácia riadenia na ustálený stav v rektifikačnej kolóne



Obr. 25: Ilustrácia riadenia na ustálený stav v rektifikačnej kolóne bez akčného zásahu



Obr. 26: Riadenie zloženia destilátu na žiadanú veličinu PID regulátorom

3.5 Príklady simulácie riadenia

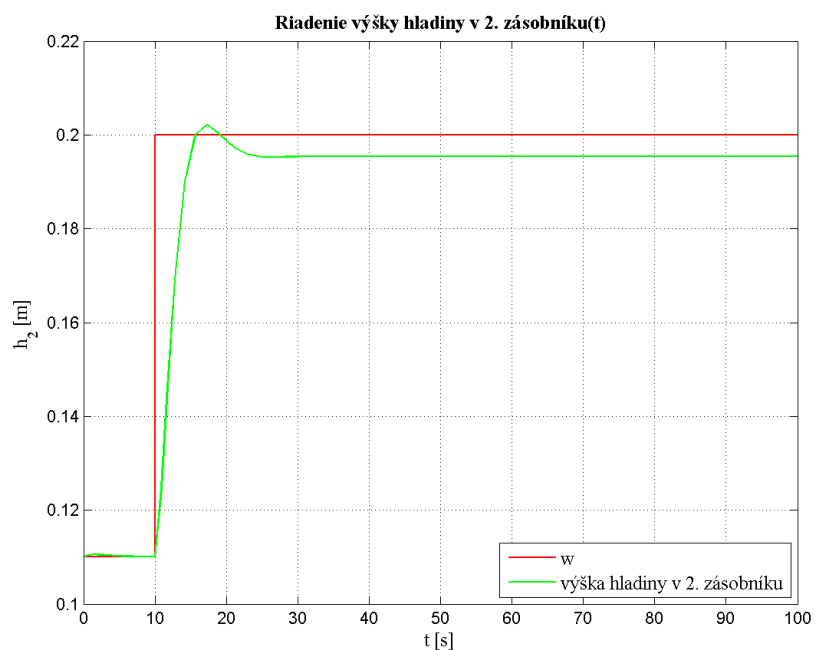
Ďalej uvedieme príklady simulácie riadenia PA regulátorov na sériovo zapojených zásobníkoch.

V prípade 2 zásobníkov kvapaliny zapojených sériovo sme riadili výšku hladiny v 2. zásobníku pomocou vstupného prietoku do 1. zásobníka. Na Obr. 27 je uvedený priebeh riadenia 1DoF regulátorom s polynómom URO v tvare $(s+0.61)^3$.

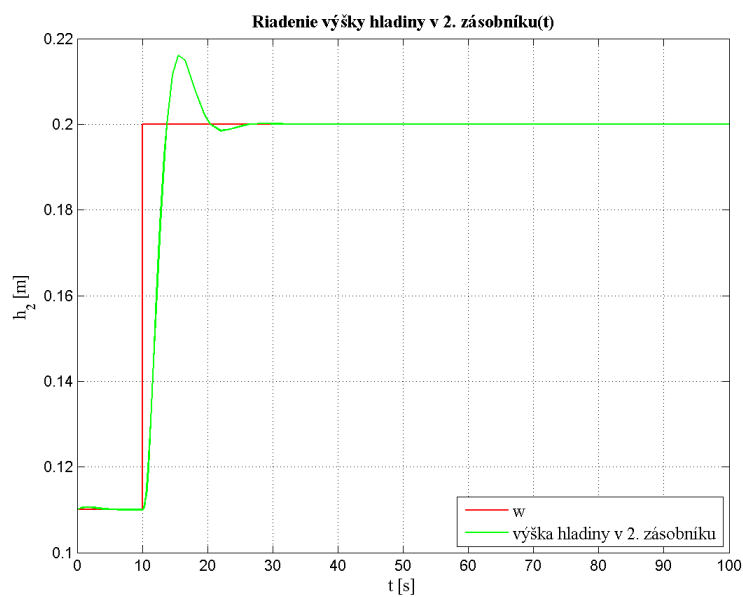
Ako je zrejmé z priebehu riadenia na Obr. 27, regulátor zanecháva trvalú regulačnú odchýlku. V istých prípadoch je postačujúce riadenie procesu v tzv. delta okolí. Delta okolie je definované ako povolená odchýlka žiadanej veličiny od skutočnej dosiahnutej.

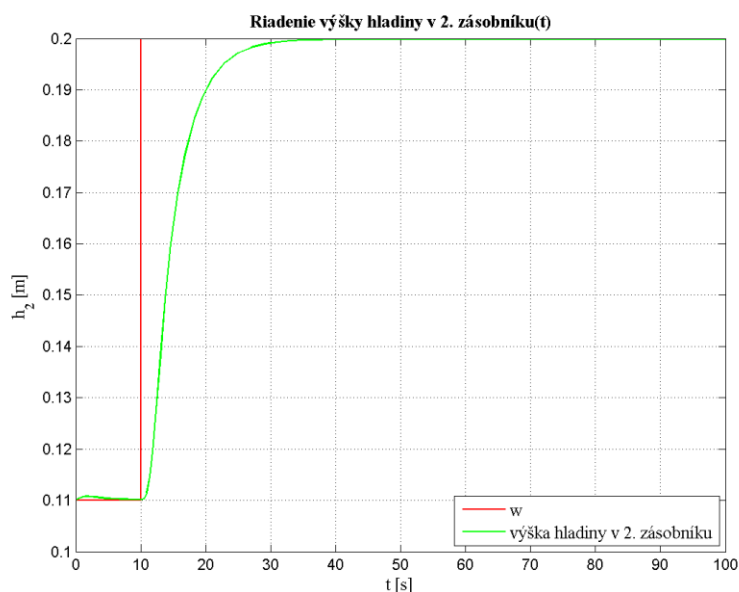
Pokiaľ si riadenie procesu vyžaduje odstránenie trvalej regulačnej odchýlky, je možné použiť 1DoF riadenie s integračnou zložkou. Pre uvedený príklad bol polynómom URO zvolený v tvare $(s+1)^5$. Priebeh riadenia je ilustrovaný na Obr. 28. Riadenie vykazuje prekmit, avšak výstup rýchlo dosahuje žiadajú veličinu, čo je pre ukážku postačujúce.

Riadenie pomocou 2DoF regulátora je znázornené na Obr. 29. Regulátor obsahuje integračnú zložku.



Obr. 27: Riadenie výšky hladiny v 2. zásobníku pomocou 1DoF regulátora

Obr. 28: Riadenie výšky hladiny v 2. zásobníku pomocou 1DoF regulátora
s integračnou zložkou



Obr. 29: Riadenie výšky hladiny v 2. zásobníku pomocou 2DoF regulátora s integračnou zložkou






3.6 Implementácia knižnice v LMS Moodle

Vytvorená knižnica technologických modelov je implementovaná do e-learningového systému Moodle v kurze Virtuálne a vzdialené laboratórium (Vlab) (Obr. 30). [1]



Samotná aplikácia Web Server je vložená na serveri Ústavu informatizácie, automatizácie a matematiky. Kurz Vlab obsahuje odkazy na jednotlivé simulácie, prostredníctvom, ktorých sa Web Server spojí s programom MATLAB.

Virtuálne laboratórium WS

Sériovo zapojené zásobníky

-  Simulácia
-  Riadenie PID regulátorom
-  Riadenie 1DoF regulátorom
-  Riadenie 1DoF s integračnou zložkou
-  Riadenie 2DoF regulátorom

Etážová rektifikačná kolóna

-  Simulácia
-  Riadenie PID regulátorom

Obr. 30: Virtuálne laboratórium Vlab v LMS Moodle

Záver

Prínosom práce je doplnenie projektu virtuálneho laboratória na Moodle o knižnicu modelov technologických procesov aj s riadením. Knižnica dopĺňa riadenie zásobníkov kvapaliny o riadenie polynomickými regulátormi. Obohacuje virtuálne laboratórium o procesy plášťového prietokového chemického reaktora a etážovej rektifikačnej kolóny. Oba procesy je možné taktiež riadiť vybranými regulátormi opísanými v práci.

V práci bol stručne opísaný postup na vytvorenie simulácie modelu procesu vo virtuálnom laboratóriu. Knižnica virtuálneho laboratória je vytvorená na základe matematických modelov uvedených v práci. Simulácie sú vytvorené v prostredí aplikácie Web Server s pomocou programu MATLAB.

Projekt virtuálneho laboratória je možné v budúcnosti rozšíriť o ďalšie modely procesov alebo typy riadenia.

Zoznam použitej literatúry

- [1]. Moodle. *Vlab*. [Online] [Dátum: 23. 5 2013.] <http://www.kirp.chtf.stuba.sk/moodle/course/view.php?id=312>.
- [2]. **Bakošová, M.** Moodle. *Modelovanie - 2 zásobníky kvapaliny zapojené za sebou s interakciou a bez interakcie*. [Online] <http://www.kirp.chtf.stuba.sk/moodle/mod/resource/view.php?id=19262>.
- [3]. —. Moodle. *Modelovanie - Prietokový chemický reaktor s jednou reakciou 1. poriadku*. [Online] [Dátum: 21. 5 2013.] www.kirp.chtf.stuba.sk/moodle/mod/resource/view.php?id=19352.
- [4]. —. Moodle. *Modelovanie v procesnom priemysle - Etážová rektifikačná kolóna*. [Online] [Dátum: 21. 5 2013.] <http://www.kirp.chtf.stuba.sk/moodle/mod/resource/view.php?id=18567>.
- [5]. **Mikleš, J. – Fikar, M.** *Modelovanie, identifikácia a riadenie procesov 2*. Bratislava : STU Press, 2004.
- [6]. Carinthia University of Applied Science. *CTI Matlab WebServer Page* . [Online] [Dátum: 13. 5 2013.] <http://sim01.cti.ac.at/>.
- [7]. CVUT Praha. *MATLAB Web Server*. [Online] [Dátum: 14. 5 2013.] <http://matlab.feld.cvut.cz/view.php?navezclanku=vypocet-prenosovych-parametru-strukturovane-metalicke-kabelaze&cislocclanku=2012080001>.
- [8]. FEI TUKE. *CyberEduCentre-Výmenníky tepla*. [Online] http://matlab.fei.tuke.sk/applications/CyberVirtLabMWS/Vymennik_tepla/DMC_simulacie.html.
- [9]. Moodle. *Laboratórne cvičenia z Riadenia procesov*. [Online] [Dátum: 21. 5 2013.] <http://www.kirp.chtf.stuba.sk/moodle/course/view.php?id=231>.
- [10]. Moodle. *Modelovanie*. [Online] [Dátum: 24. 5 2013.] www.kirp.chtf.stuba.sk/moodle/mod/resource/view.php?id=19225.

- [11]. **Fikar, M.** Moodle. *Algebraic Control Theory*. [Online] [Dátum: 22. 5 2013.] <http://www.kirp.chnik.stuba.sk/moodle/mod/resource/view.php?id=18361>.
- [12]. Polyx. [Online] [Dátum: 24. 5 2013.] <http://www.polyx.com/>.
- [13]. **Škultéty, Rastislav.** *JavaScript - Programujeme internetové aplikácie*. Brno : Computer Press, 2004 . ISBN: 8025101444.
- [14]. W3CSchools. [Online] [Dátum: 17. 5 2013.] http://www.w3schools.com/jquery/jquery_intro.asp.
- [15]. W3Schools. *TCP/IP Protocols*. [Online] [Dátum: 16. 5 2013.] http://www.w3schools.com/tcpip/tcpip_protocols.asp.
- [16]. Mathworks. *MWS*. [Online] [Dátum: 21. 5 2013.] http://www.mathworks.com/products/new_products/webserver_discontinued.html;jsessionid=4cccc557959ff98f7068acf95738?s_cid=r2006b_webserver.
- [17]. **Kroon, Dirk Jan.** MathWorks. *Matlab Web Server*. [Online] 12. 5 2012. <http://www.mathworks.com/matlabcentral/fileexchange/29027>.
- [18]. MathWorks. *MATLAB*. [Online] 12. 5 2012. <http://www.mathworks.com/products/matlab/>.
- [19]. **Ing. Ľuboš Čirka, PhD.** Špeciálne jazykové prostriedky počítačov - SJPP. *Kapitola 1: Úvod do MATLABu*. [Online] [Dátum: 18. 5 2013.] <http://www.kirp.chnik.stuba.sk/~cirka/vyuka/matlab/matlab.php?kniha=1#kap1>.
- [20]. **Mathworks.** Simulink® Model-based and system-based design. [Online] [Dátum: 18. 5 2013.] http://www.kirp.chnik.stuba.sk/~cirka/vyuka/matlab/files/sl_using.pdf.
- [21]. W3CSchool. [Online] [Dátum: 19. 5 2013.] http://www.w3schools.com/tags/att_form_method.asp.
- [22]. **Laursen, IOLA and Ole.** FLOT. [Online] [Dátum: 17. 5 2013.] <http://www.flotcharts.org/>.