

Programová podpora laboratorních úloh předmětu „Teorie automatického řízení II“

Software Support for Laboratory Exercises for the
" Theory of Automatic Control II" Course

Tomáš Rapala

Bakalářská práce
2008



Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

Univerzita Tomáše Bati ve Zlíně

Fakulta aplikované informatiky

Ústav automatizace a řídicí techniky

akademický rok: 2007/2008

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Tomáš RAPALA**

Studijní program: **B 3902 Inženýrská informatika**

Studijní obor: **Automatické řízení a informatika**

Téma práce: **Programová podpora laboratorních úloh předmětu
"Teorie automatického řízení II"**

Zásady pro vypracování:

1. Zpracujte literární rešerši problematiky teorie automatického řízení pro lineární diskrétní dynamické systémy se zaměřením na obsah předmětu bakalářského studia "Teorie automatického řízení II" na FAI UTB ve Zlíně.
2. Nastudujte zadání laboratorních úloh v daném kurzu.
3. K těmto úlohám vytvořte programovou podporu v prostředí MATLAB + SIMULINK, která bude umožňovat automatické výpočty či simulace dle jednotlivých zadání.
4. Na vhodně zvolených příkladech, případně na vzorových protokolech, demonstруйте možnosti vytvořeného programu.

Rozsah práce:

Rozsah příloh:

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

Seznam odborné literatury:

1. Balátě, J.: Automatické řízení. BEN -- technická literatura, Praha, 2003.
2. Gerlich, V.: Podpora cvičení předmětu "Teorie automatického řízení II". Bakalářská práce, FAI UTB ve Zlíně, 2006.
3. Navrátil, P.: Teorie automatického řízení II -- diskrétní systémy [online]. [cit. 20. ledna 2008]. Dostupné z URL: .
4. Šulc, B., Vítečková, M.: Teorie a praxe návrhu regulačních obvodů. Vydavatelství ČVUT, Praha, 2004.
5. Švarc, I.: Automatizace Automatické řízení. Akademické nakladatelství CERM, Brno, 2002.
6. The MathWorks -- Online Documentation [online]. [cit. 20. ledna 2008]. Dostupné z URL: .
7. Vašek, V.: Teorie automatického řízení II. Skriptum VUT v Brně, 1990.

Vedoucí bakalářské práce: **Ing. Radek Matušů, Ph.D.**
Ústav automatizace a řídicí techniky

Datum zadání bakalářské práce: **22. února 2008**

Termín odevzdání bakalářské práce: **6. června 2008**

Ve Zlíně dne 22. února 2008

prof. Ing. Vladimír Vašek, CSc.
děkan



prof. Ing. Vladimír Vašek, CSc.
ředitel ústavu

ABSTRAKT

Cílem této práce je zpracovat jednotlivé metody automatického řízení pro lineární diskrétní dynamické systémy se zaměřením na obsah bakalářského předmětu *Teorie automatického řízení II*. Dále vytvoření programové podpory laboratorních úloh výše zmíněného předmětu za pomoci programu MATLAB + SIMULINK a na vhodně zvolených příkladech (vzorových protokolech) demonstrovat možnosti vytvořeného programu.

Klíčová slova: Automatické řízení, Lineární diskrétní dynamické systémy, MATLAB + SIMULINK

ABSTRACT

The goal of this bachelor thesis is to work up individual methods of automatic controls for linear discrete dynamic systems in compliance to the content of the *Theory of automatic control II* course. An additional goal is to create program support for laboratory tasks for this course using program MATLAB + SIMULINK and also illustrate possibilities of the created program on chosen examples.

Keywords: Automatic controls, Linear discrete dynamic systems, MATLAB + SIMULINK

Děkuji vedoucímu své bakalářské práce Ing. Radkovi Matušů Ph. D. za odborné vedení, podnětné připomínky a rady udílené při vypracování práce.

Prohlašuji, že jsem na bakalářské práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků, je-li to uvolněno na základě licenční smlouvy, budu uveden jako spoluautor.

Ve Zlíně

.....
Podpis diplomanta

OBSAH

ÚVOD	8
I TEORETICKÁ ČÁST	9
1 IDENTIFIKACE SOUSTVY A REGULACE DISKRÉTNÍHO SYSTÉMU	10
1.1 IDENTIFIKACE REGULOVANÉ SOUSTAVY.....	10
1.1.1 Strejcova metoda.....	10
1.1.2 Vyhodnocování a aproximace přechodových charakteristik	11
1.2 ČÍSLICOVÝ DVOUPOLOHOVÝ REGULÁTOR	11
1.3 DISKRÉTNÍ PID REGULÁTOR ZIEGLER NICHOLS.....	11
1.4 METODA POŽADOVANÉHO MODELU (INVERZE DYNAMIKY).....	11
1.5 KONEČNÝ POČET KROKŮ REGULACE	12
1.6 URČENÍ KOEFICIENTŮ PSD REGULÁTORU NA ZÁKLADĚ UMÍSTĚNÍ PÓLŮ	12
II PRAKTICKÁ ČÁST	14
2 SOFTWARE	15
2.1 MATLAB	15
2.1.1 SIMULINK	16
2.1.2 GUIDE	17
2.1.2.1 Blank GUI (Default).....	18
3 SILAU	21
3.1 SPUŠTĚNÍ PROGRAMU	21
3.2 HLAVNÍ OKNO	21
3.2.1 Panely (button group) hlavního okna.....	22
3.3 IDENTIFIKACE SOUSTAVY	22
3.4 DVOUPOLOHOVÁ REGULACE	24
3.4.1 Dvoupolohová regulace bez penalizace	24
3.4.2 Dvoupolohová regulace s penalizací.....	25
3.5 ZIEGLER NICHOLS (TAKAHASHIHO MODIFIKACE PSD)	26
3.5.1 Ziegler Nichols odhadem	26
3.5.2 Ziegler Nichols metodou.....	27
3.6 METODA POŽADOVANÉHO MODELU (INVERZE DYNAMIKY).....	28
3.6.1 Metoda požadovaného modelu (inverze dynamiky) PID regulátor	28
3.6.2 Metoda požadovaného modelu (inverze dynamiky) PI regulátor.....	29
3.7 METODA KONEČNÉHO POČTU KROKŮ	30
3.7.1 Metoda konečného počtu kroků bez omezení $u(0)$	30
3.7.2 Metoda konečného počtu kroků s omezením $u(0)$	31
3.8 METODA UMÍSTĚNÍ PÓLŮ.....	32
3.8.1 Metoda umístění pólů pro aperiodický průběh	32
3.8.2 Metoda umístění pólů pro periodický průběh	33

3.9	SIMULAČNÍ SCHÉMATA.....	34
3.10	OCHRANA.....	35
	ZÁVĚR.....	36
	ZÁVĚR V ANGLIČTINĚ.....	37
	SEZNAM POUŽITÉ LITERATURY	38
	SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK	39
	SEZNAM OBRÁZKŮ	40
	SEZNAM TABULEK.....	41
	SEZNAM PŘÍLOH.....	42

ÚVOD

Trendem 21. století je obrovská rychlost vývoje veškerého odvětví, neustálé zdokonalování jednotlivých částí i celků a snaha ulehčit práci sami sobě i lidem obecně. Dříve byly počítače, se kterými uměli zacházet pouze výrobci, či programátoři těchto zařízení. Dnes není velkým problémem komunikace s těmito zařízeními i pro malé děti nebo pro starší generaci. Přestože manuály jsou stále širší, člověk se s daným zařízením seznámí poměrně rychle. Je to dáno jednoduchostí a srozumitelností vytvořeného rozhraní člověk – stroj.

Aniž by si kdokoli uvědomil, je u většiny lidí naprostou součástí jejich každého dne systém automatického řízení, například použití výtahu v budově. Syntéza automatického řízení se tedy nedotýká jen průmyslové automatizace, ale i široké veřejnosti. Proto je nezbytné stále snižovat vzdálenost k produktům samotné automatizace.

Cílem této práce je vytvořit programovou podporu laboratorních úloh předmětu „Teorie automatického řízení II“. Podpora jednotlivých úkolů v laboratorním cvičení předmětu je vytvořena v programu MATLAB + SIMULINK. Obsahem práce je i stručné nastínění teorie k danému tématu.

V rámci této bakalářské práce je také vytvoření vzorových protokolů, které demonstrují možnosti vytvořeného programu. V protokolech jsou vypracovány postupy jednotlivých úloh a jsou zde vypsány i zdrojové kódy výpočtů potřebných parametrů.

Studenti mohou využívat veškeré materiály. Program a vzorové protokoly by jim měly ulehčit práci v laboratořích a pomoci při pochopení jednotlivých látek.

I. TEORETICKÁ ČÁST

1 IDENTIFIKACE SOUSTVY A REGULACE DISKRÉTNÍHO SYSTÉMU

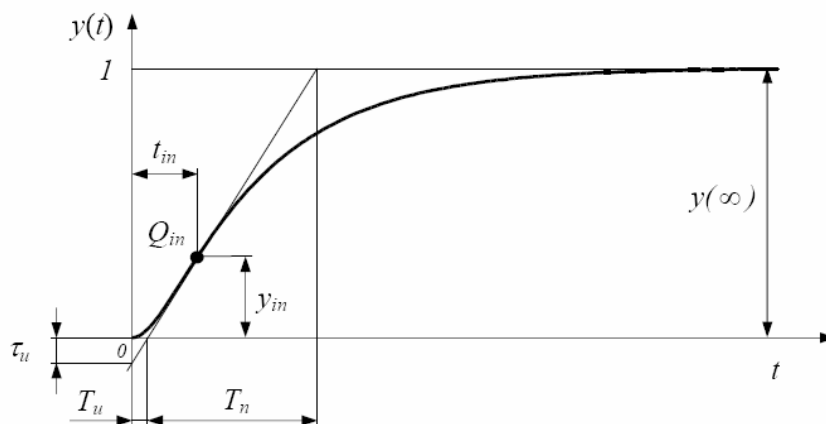
Spojitéj systém odpovídá intuitivní představě dynamického systému. Diskrétní systém je tedy systém s diskrétním časem, může vzniknout tak, že všechny veličiny spojitého systému měříme v diskrétních časových okamžicích.

1.1 Identifikace regulované soustavy

Mnohdy nemusí být znám přenos soustavy, proto je ho potřeba nejprve získat, aby bylo možno provádět syntézu regulovaného obvodu. Tzn., že je třeba aproximovat přechodovou charakteristiku řízeného systému nějakým vhodným modelem (přenosem).

1.1.1 Strejcova metoda

Jednou z nejjednodušších metod aproximace přechodových charakteristik pro statické soustavy navrhl V. Strejc. Je vhodná pro objekty, které můžeme považovat za statické soustavy. Předpokládáme přitom, že kořeny charakteristické rovnice jsou reálné a záporné. Metoda umožňuje aproximovat naměřené data soustavami n -tého řádu se stejnými časovými konstantami, nebo soustavami druhého řádu s různě velkými časovými konstantami. O způsobu aproximace se rozhodne podle úseků, které vytíná na časové ose tečna, sestavená v inflexním bodě aproximované přechodové charakteristiky, resp. podle poměru $\tau_u = T_u / T_n$, přičemž úsek T_u je doba průtahu a úsek T_n je doba náběhu.



Obrázek 1 – Normovaná přechodová charakteristika statické soustavy vyššího řádu

1.1.2 Vyhodnocování a aproximace přechodových charakteristik

- a) Ze statické charakteristiky určit oblast, ve které je chování řízeného systému lineární. V lineární oblasti poté provést měření přechodových charakteristik.
- b) Úprava naměřených dat přechodové charakteristiky, tj. posunutí počáteční hodnoty přechodové charakteristiky do počátku souřadnicového systému, resp. Normování přechodové charakteristiky.
- c) Aproximace (sloučené) přechodové charakteristiky Je možno aproximovat soustavy statické i astatické prvního i vyššího řádu, s dopravním zpožděním i bez dopravního zpoždění. Parametry řízené soustavy je možné určit např. pomocí *Strejcovy metody*. [2]

1.2 Číslicový dvoupolohový regulátor

Jedním z typů regulátoru je např. ON/OFF regulátor, v České republice známější pod názvem dvoupolohový regulátor. Tento typ regulátoru se hodí k regulaci zejména proporciálních soustav, se setrvačností 1. řádu s velkou časovou konstantou, bez dopravního zpoždění, při malých změnách regulované veličiny. [1]

1.3 Diskrétní PID regulátor Ziegler Nichols

Tato metoda patří mezi klasické metody určování hodnot stavitelných parametrů PID regulátorů na základě experimentálního měření. Svůj význam má i dnes, přesto že její kořeny sahají do první poloviny 50. let 20. století. Je velmi často používanou metodou pro porovnání nastavení parametrů regulátoru pomocí jiných metod syntézy.

První metoda Zieglera-Nicholse představuje zřejmě nejrychlejší způsob, jak vhodně nastavit PID regulátor. Stanovuje hodnoty parametrů regulátoru z tzv. kritického zesílení K_{PK} a periody kritických kmitů T_k . [2], [3]

1.4 Metoda požadovaného modelu (inverze dynamiky)

Metoda umožňuje snadné a rychlé seřízení konvenčních typů číslicových a analogových regulátorů pro základní druhy regulovaných soustav s dopravním zpožděním nebo bez

dopravního zpoždění. Přístup umožňuje určit typ vhodného konvenčního regulátoru a seřadit ho tak, aby bylo dosaženo nulové trvalé regulační odchylky a požadovaného relativního překmitu regulované veličiny od 0 do 50% při skokové změně polohy žádané veličiny. [1]

1.5 Konečný počet kroků regulace

Základní myšlenka návrhu Z-přenosu regulátoru r ve zpětné vazbě vychází z definice končeného počtu kroků řízení. Pro obě verze se požaduje, aby regulační odchylka byla nenulová pouze pro končený počet hodnot $e_j, j = 0, 1, 2, \dots, k_{\min} - 1$. To však znamená, že Z-obraz regulační odchylky musí být **konečným polynomem**. Pro silnou verzi musí být **konečným polynomem** i Z-obraz akčních **zásahů**. Je-li počet kroků minimální nazývá se také *časově optimální diskrétní regulací*. Rozlišují se dvě verze ukončení regulačního pochodu – za konečný počet kroků a to verze slabá, tj. že regulační odchylka bude nulová od kroku $k_{\min} \geq k$ pouze v okamžicích vzorkování, na rozdíl od silné verze, kde regulační odchylka bude nulová od kroku $k_{\min} \geq k$ nejen v okamžicích vzorkování, ale i mimo okamžiky vzorkování.[4], [2]

1.6 Určení koeficientů PSD regulátoru na základě umístění pólů

Určení koeficientů PSD regulátoru na základě umístění pólů uzavřeného diskrétního regulačního obvodu se realizuje s cílem dosáhnout požadované dynamické vlastnosti. Poloha pólů zpětnovazebního systému určuje jeho chování. Ze znalosti vlivu polohy pólů spojitého systému na stabilitu a dynamiku systému a z korespondence mezi komplexními proměnnými z a s , pro něž platí

$$z = e^{Ts}$$

kde T je perioda vzorkování, lze pro umístování kořenů charakteristické rovnice (pólů) využít následujících vztahů. [5]

<i>V rovině „s“ u spojitého systému</i>	<i>V rovině „z“ u diskrétního systému</i>	<i>V časové oblasti</i>
v levé polorovině	uvnitř jednotkové kružnice	stabilní
v pravé polorovině	vně jednotkové kružnice	nestabilní
na záporné reálné ose	na reálné ose v intervalu $0 < z < 1$ v intervalu $-1 < z < 0$	stabilní nekmitavý (aperiodický) stabilní kmitavý
na kladné reálné ose	na kladné reálné ose vně jednotkové kružnice	nestabilní nekmitavý (aperiodický)
komplexně sdružené v levé polorovině	komplexně sdružené uvnitř jednotkové kružnice	kmitavý tlumený
na imaginární ose	na jednotkové kružnici	na mezi stability
komplexně sdružené v pravé polorovině	komplexně sdružené a na záporné reálné ose vně jednotkové kružnice	netlumený kmitavý

Tabulka 1 Poloha kořenů charakteristické rovnice

II. PRAKTICKÁ ČÁST

2 SOFTWARE

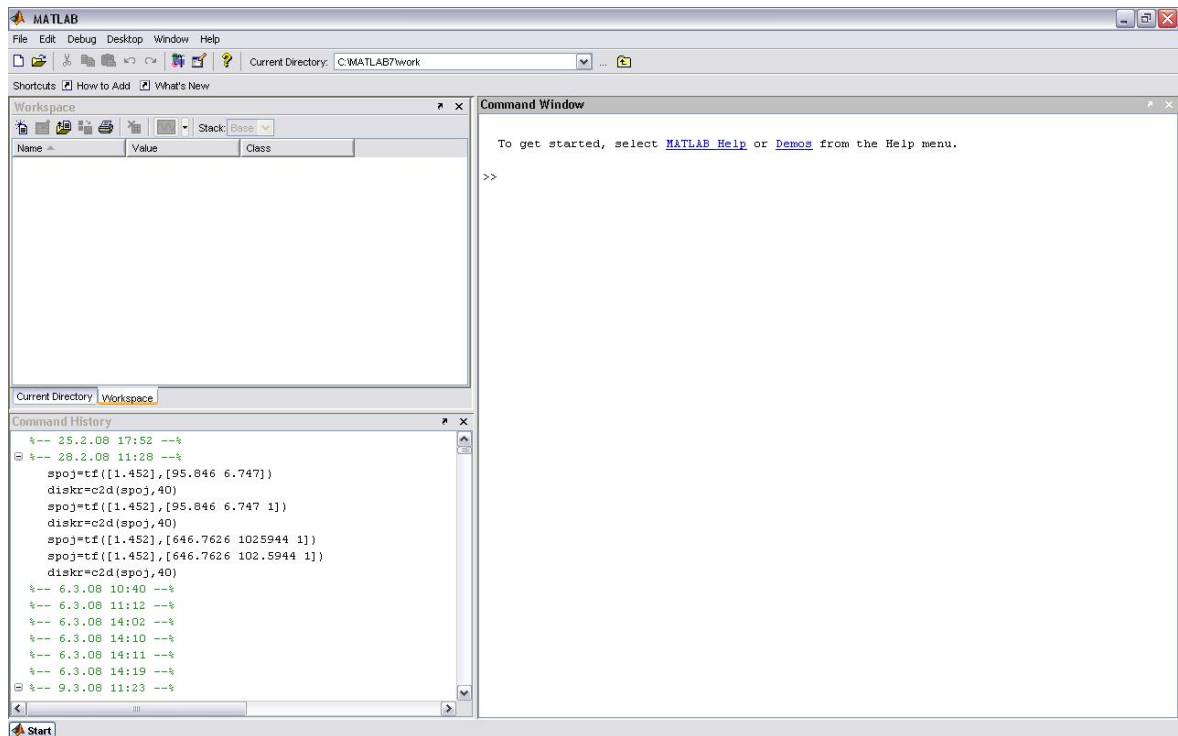
V současné době je široká škála různých programů určených k výpočtu nejrůznějších vědeckotechnických výpočtů, k simulaci či návrhu algoritmů. Pro příklad je uvedeno několik z nich: **Micro-Cap** je přímo určen pro simulaci obvodu. Jeho předností je jednoduchost použití. Vstupy i výstupy jsou grafické. **Mathematica** je software přímo určen pro veškeré výpočty, možnost širokého programování, modelování a analýzy problémů. Program **Maple** slouží pro řešení matematických problémů, ovládá pravidla algebry a matematické analýzy. Maple dokáže vykreslovat dvojrozměrné či trojrozměrné grafy, také v něm lze tvořit pokročilejší grafiku jako animace, pole vektorů, parametrické křivky nebo dynamické systémy. Dalším z programů je **Matlab**. Tento program je více popsán viz 2.1.

2.1 MATLAB

MATLAB je programové prostředí a skriptovací programovací jazyk pro vědeckotechnické numerické výpočty, modelování, návrhy algoritmů, počítačové simulace, analýzu a prezentaci dat, měření a zpracování signálů, návrhy řídicích a komunikačních systémů.

Název MATLAB vznikl zkrácením slov MATrix LABoratory (volně přeloženo „laboratoř s maticemi“), což odpovídá skutečnosti, že klíčovou datovou strukturou při výpočtech v MATLABu jsou matice. Vlastní programovací jazyk vychází z jazyka Fortran.

<http://cs.wikipedia.org/wiki/MATLAB>




Obrázek 2 – Hlavní okno Matlabu

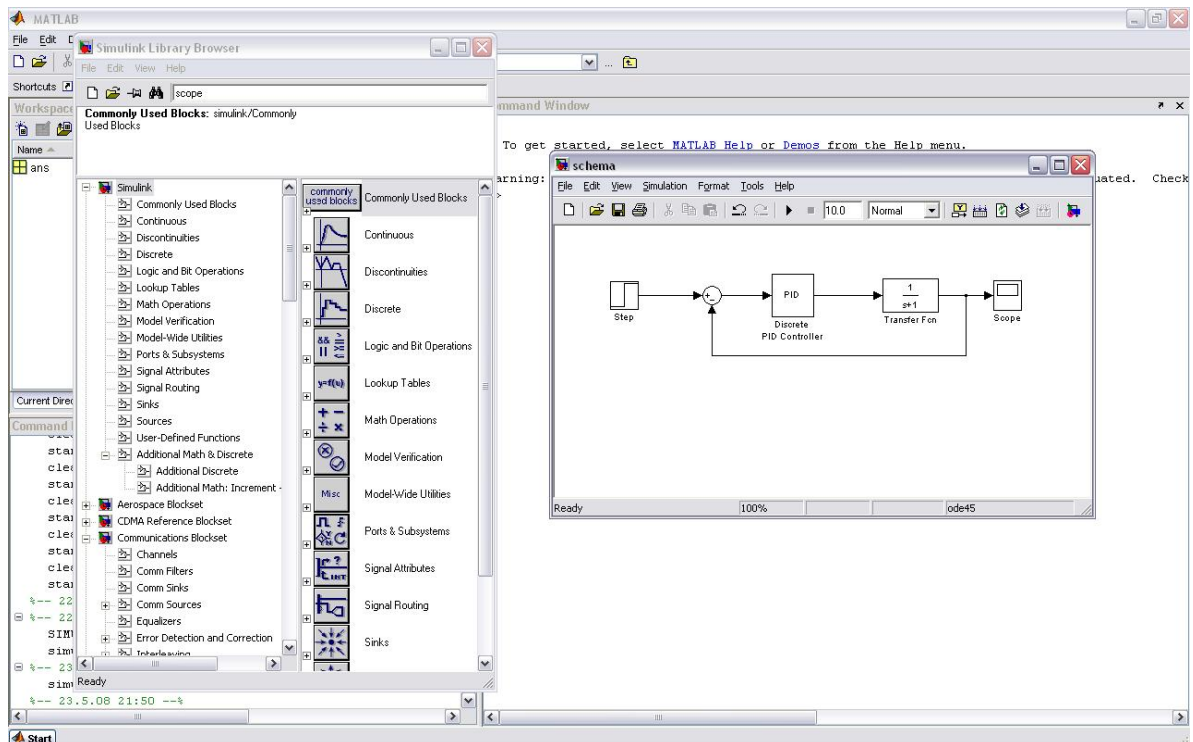
2.1.1 SIMULINK

Simulink je program pro simulaci a modelování dynamických systémů, který využívá algoritmy MATLABu pro numerické řešení nelineárních diferenciálních rovnic. Poskytuje uživateli možnost rychle a snadno vytvářet modely dynamických soustav ve formě blokových schémat a rovnic. [9]

Simulink můžeme spustit

- ikonou SIMULINK na panelu nástrojů 
- pomocí Launch Pad
- napsání *simulink* v příkazovém okně


Pro vytvoření simulačního schématu otevřeme okno (*model*) v hlavním menu **File – New – Model**, nebo pomocí ikony *Create a new model*. Všechny potřebné součásti k vytvoření simulačního schématu jsou k dispozici v blocích. Jednotlivé bloky uchytíme levým tlačítkem myši a přenášíme do vytvořeného okna. Bloky spojujeme jednoduchým způsobem, kliknutím na vstup (výstup) jednoho bloku a tažením čáry ke vstupu (výstupu) jiného bloku. Bloky lze kopírovat, otáčet, měnit barvy atd.



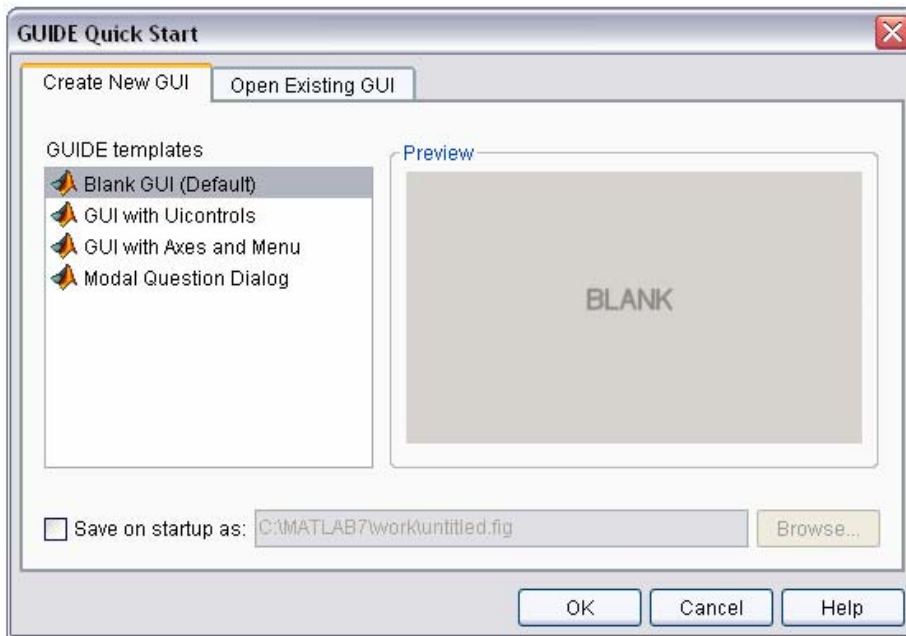
Obrázek 3 – Simulink

2.1.2 GUIDE

Nástroj Guide je v Matlabu využíván k tvorbě grafického uživatelského rozhraní.

- Guide můžeme spustit
- ikonou GUIDE na panelu nástrojů 
 - pomocí Launch Pad
 - napsání *guide* v příkazovém okně

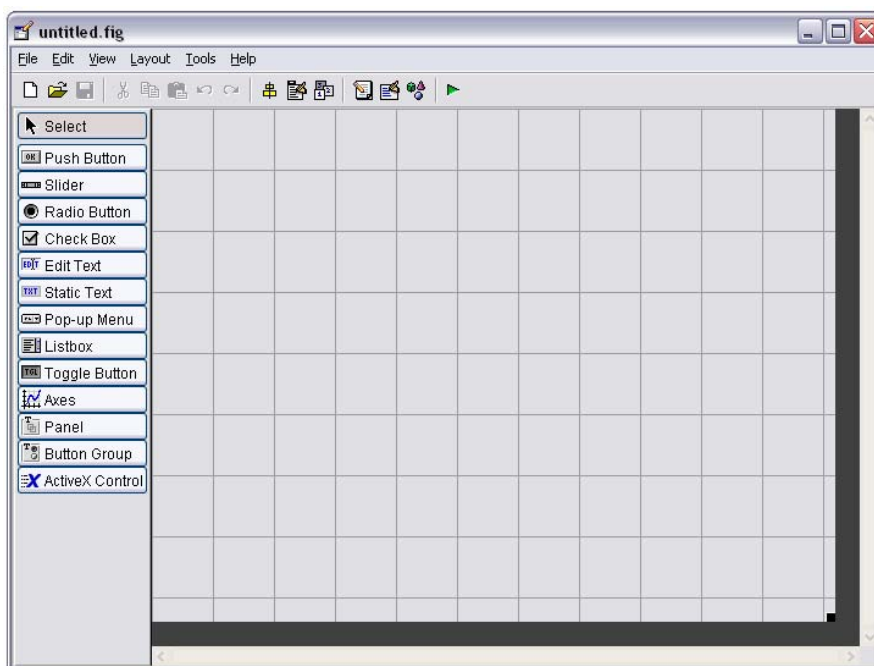
Po spuštění Guide je spuštěn dialog, který nabízí ze dvou listů. V prvním listu **Create New GUI** je možnost vytvoření nového dialogu a druhý list **Open Existing GUI** nabízí otevření již existujícího dialogu. V prvním listu je výběr ze 4 šablon **Blank GUI (Default)**, **GUI with Uicontrols**, **GUI with Axes and Menu** a **Modal Question Dialog**.



Obrázek 4 – GUIDE Quick Start

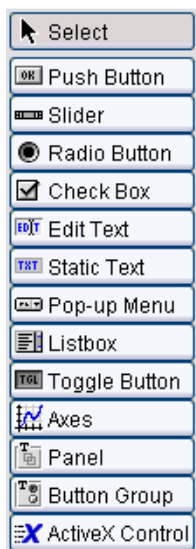
2.1.2.1 *Blank GUI (Default)*

Při spuštění první šablony **Blank GUI (Default)** se vytvoří prázdné dialogové okno bez jakýchkoliv existujících ovládacích prvků. Máme tedy možnost vytvořit dialog s téměř neomezenými možnostmi. V tomto okně je definována pouze velikost nového dialogu a i tuto velikost je možno měnit.



Obrázek 5 – GUIDE Quick Start

Lišta s ovládacími prvky (*component palette*) obsahuje všechny možné objekty, které je možné do dialogu vložit.



Obrázek 6 – Lišta s ovládacími prvky hlavního okna pro vytváření dialogů

Select - ikona pro označení již existujících objektů

Push Button - klasické tlačítko

Slider - posuvník

Radio button - zatrhávací tlačítko ve tvaru tečky

Check box – zatrhávací tlačítko ve tvaru fajfky

Edit text – editační okno

Static text – statický text

Pop-up menu – seznam položek, ze kterých je vždy zobrazena pouze jedna. Ostatní položky lze prohlížet a vybrat po rozbalení

List box – seznam položek, ze kterých lze vybrat aktuální položku

Toggle button – přepínací tlačítko, přepínač.

Axes – vytvoří se okno, kde mohou být zobrazeny obrázky nebo grafické průběhy

Panel – jedná se o panel (rám), do kterého můžeme vkládat jednotlivé ovládací prvky

Button Group – speciální panel určen pro zatrhávací tlačítka (**Radio button**), v tomto panelu může být zatrhnuo pouze jedno zmíněné tlačítko


ActiveX Control – pomocí tohoto prvku je možné vkládat do objektu ActiveX komponenty

Po vložení prvku do nového dialogu a následného uložení, se spustí *Editor* se souborem *untitled.m*, což je vlastně zdrojový kód dialogu. S tímto kódem se pak dále pracuje. Vkládáním různých funkcí, vlastností prvků atd.

3 SILAU

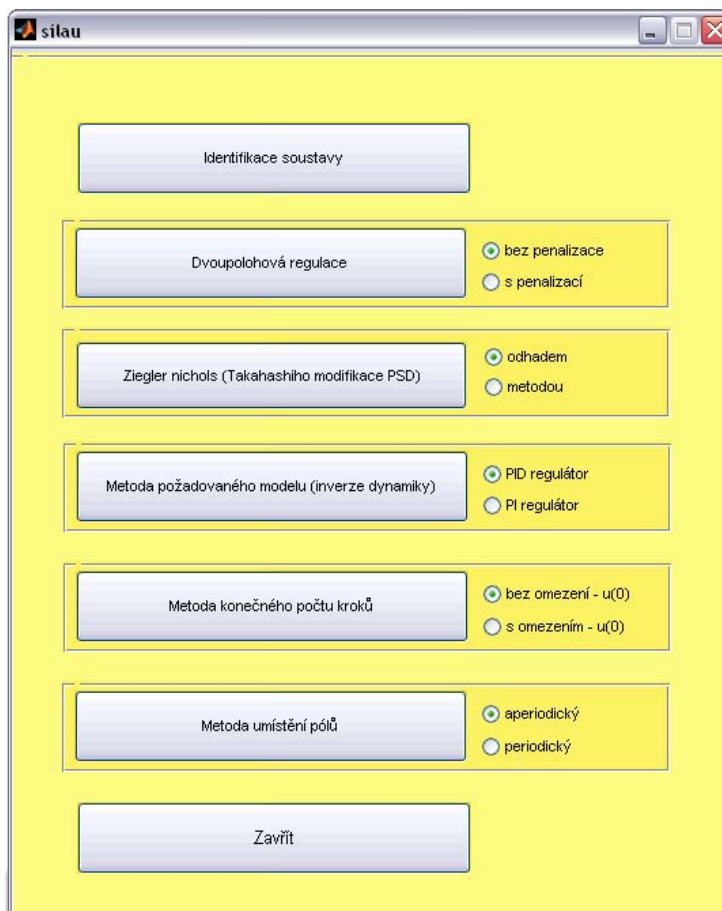
Program s názvem Silau (simulace laboratorních úloh) - slouží jako programová podpora jednotlivých laboratorních úloh v předmětu *Teorie automatického řízení II*. Ve zmíněném programu je možno simulovat a provádět jednotlivé výpočty k daným úlohám dle zadání. Program byl vytvořen v prostředí MATLAB a v jeho částích GUIDE a SIMULINK.

3.1 Spuštění programu

V programu Matlab otevřeme soubor start.m a necháme proběhnout (stiskneme tlačítko *Run*) . Naběhne hlavní okno *silau*.

3.2 Hlavní okno

Hlavní okno *silau* obsahuje 5 panelů (button group), 7 tlačítek (push button), a 10 přepínačů (radio button).



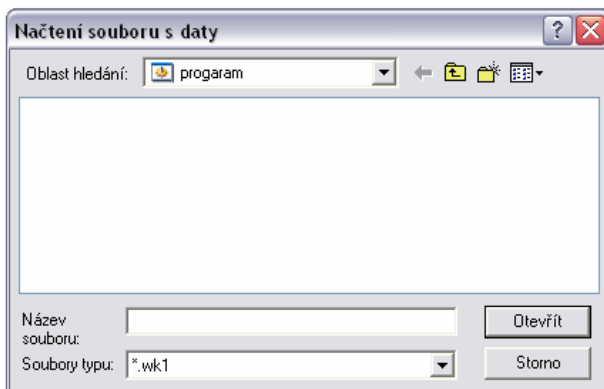
Obrázek 7 – Hlavní panel silau

3.2.1 Panely (button group) hlavního okna

Tlačítka *Dvoupolohová regulace*, *Ziegler Nichols (Takahashiho modifikace PSD)*, *Metoda požadovaného modelu (inverze dynamiky)*, *Metoda konečného počtu kroků* a *Metoda umístění pólů* jsou umístěny v jednotlivých panelech. V těchto panelech jsou vždy umístěny mimo již zmíněné tlačítka i 2 přepínací tlačítka (radio button), které slouží k dalšímu určení metody simulace. Vždy je možno označit kuličkou pouze jedno. Po označení metody stiskneme tlačítko a otevře se další okno s předem zvolenou metodou.

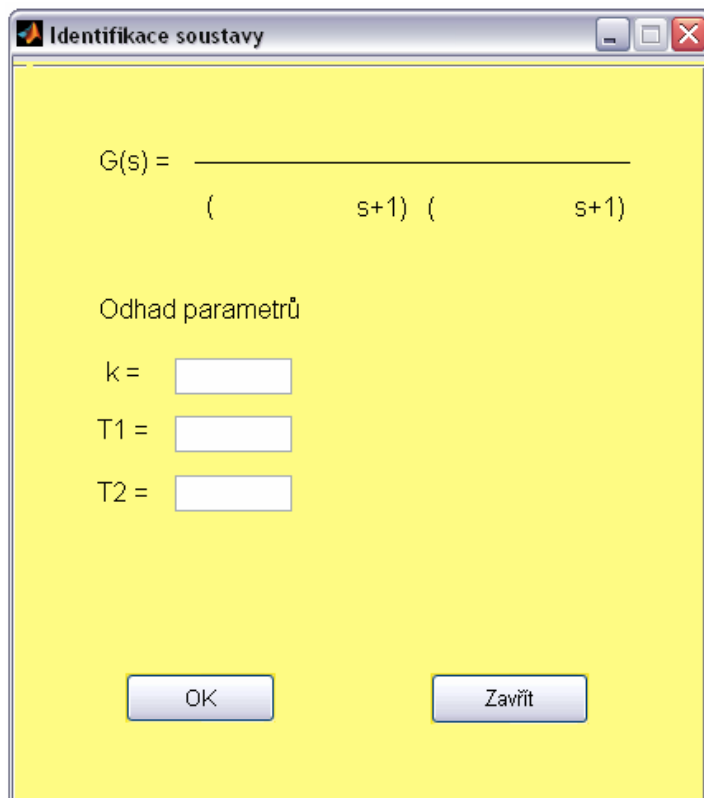
3.3 Identifikace soustavy

Po stisknutí tlačítka *Identifikace soustavy* v hlavním menu, naběhne okno *Načtení souboru s daty*.



Obrázek 8 – Načtení souboru s daty

Zde vybereme příslušný soubor **.wkl* s naměřenou přechodovou charakteristikou soustavy. Potvrdíme tlačítkem *Otevřít*. Spustí se okno pro vypsání parametrů. Do kolonek $k =$, $T1 =$, $T2 =$, zapíšeme teoretický odhad parametrů a spustíme tlačítkem *OK*. Proběhne výpočet parametrů a do přenosu $G(s) =$ se vypíšou vypočítané parametry. Tlačítkem *Zavřít* uzavřeme okno.



Identifikace soustavy

$$G(s) = \frac{\quad}{(s+1)(s+1)}$$

Odhad parametrů

k =

T1 =

T2 =

OK Zavřít

Obrázek 9 – Identifikace soustavy

3.4 Dvoupolohová regulace

Nejprve zvolíme druh Dvoupolohové regulace - bez penalizace nebo s penalizací.

3.4.1 Dvoupolohová regulace bez penalizace

Pokud je zaškrtnuto *bez penalizace*, tak se po stisknutí tlačítka *Dvoupolohová regulace* otevře okno *Dvoupolohová regulace bez penalizace*.

Obrázek 10 – Dvoupolohová regulace bez penalizace

Toto okno obsahuje rovnici přenosu $G(s)$, kde je možno zvolit zesílení přenosu k a také jednotlivé časové konstanty T_1 a T_2 .

$$G(s) = \frac{k}{(T_1s + 1)(T_2s + 1)}$$

Dále pak musí být zvolena perioda vzorkování T a konstanta ze statické charakteristiky. Tyto parametry jsou základní a zadávají se téměř v každém jednotlivém okně programu.

V tomto okně zadáme ještě Pásmo necitlivosti $+a$ a $-a$. Veškeré tyto hodnoty musí být vyplněny, nebo nebude možné pokračovat v simulaci.

3.4.2 Dvoupolohová regulace s penalizací

Pokud je zaškrtnuto *s penalizací*, tak se po stisknutí tlačítka *Dvoupolohová regulace* v hlavním okně *silau*, otevře okno *Dvoupolohová regulace s penalizací*.

Dvoupolohová regulace s penalizací

$$G(s) = \frac{[]}{([] s + 1) ([] s + 1)}$$

T = [] Konst. ze statické charakteristiky []

Pásmo necitlivosti (+a, -a)

+a = 0 °C Kpmax = []

-a = 0 °C ppp = [] %

OK Zavřít

Obrázek 11 – Dvoupolohová regulace s penalizací

V okně *Dvoupolohová regulace s penalizací* není nutno již zadávat Pásmo necitlivosti, ale k dalšímu měření se zde musí zadat parametr K_{pmax} a pásmo působnosti penalizace ppp .

Po vyplnění všech údajů a potvrzení tlačítka *OK* se spustí simulační schéma v prostředí Simulink. Veškeré zadané nebo vypočítané parametry se použijí v jednotlivých blocích.

3.5 Ziegler Nichols (Takahashiho modifikace PSD)

Další z metod je *Ziegler Nichols*, kde se simuluje regulace soustavy diskretním PID regulátorem. V hlavním okně *silau*, jsou opět dvě možnosti pro zmíněnou metodu.

3.5.1 Ziegler Nichols odhadem

První z možností je *Ziegler Nichols odhadem*, zde je nutno zadat opět základní parametry viz 3.4.1 a dále hodnoty K_P K_I K_D , které volíme odhadem.

Ziegler Nichols

$$G(s) = \frac{\text{[]}}{(\text{[] } s + 1) (\text{[] } s + 1)}$$

T = [] Konst. ze statické charakteristiky

Parametry []

Kp = []

Ki = []

Kd = []

OK Zavřít

Obrázek 12 – Ziegler Nichols odhadem

3.5.2 Ziegler Nichols metodou

V druhé možnosti *Ziegler Nichols metodou* je nutno zadat základní parametry viz 3.4.1. Hodnoty parametrů K_p , K_I a K_D budou vypočítány automaticky po stisknutí tlačítka *OK*. Pokud uživatel bude chtít změnit některý z parametrů, využije kolonek *změna K_p* , *změna K_i* nebo *změna K_d* , pokud ne, zapíše zde 0. Změna je zapisována v procentech.

Ziegler Nichols

$$G(s) = \frac{\text{[]}}{(\text{[] } s + 1) (\text{[] } s + 1)}$$

T = [] Konst. ze statické charakteristiky []

Parametry	Změna parametrů
Kp =	změna Kp = [] %
Ki =	změna Ki = [] %
Kd =	změna Kd = [] %

OK Zavřít

Obrázek 13 – Ziegler Nichols metodou

3.6.2 Metoda požadovaného modelu (inverze dynamiky) PI regulátor

U metody s PI regulátorem je nutno v okně nastavit překmit a základní parametry viz 3.4.1, mimo periody vzorkování T , ta bude vypočítána spolu s parametry $q0$ a $q1$.

Inverze dynamiky - PI regulátor

$$G(s) = \frac{\text{[]}}{(\text{[] } s+1) (\text{[] } s+1)}$$

Překmit [] % Konst. ze statické charakteristiky []

Parametry

q0 = T =

q1 =

OK Zavřít

Obrázek 15 – Inverze dynamiky PI – regulátor

3.7 Metoda konečného počtu kroků

3.7.1 Metoda konečného počtu kroků bez omezení $u(0)$

Metoda konečného počtu kroků bez omezení $u(0)$ využívá základních parametrů viz 3.4.1, a parametrů regulátoru $p1, p2, q0, q1$ a $q2$. Nutno zadat pouze základní parametry.

Metoda konečný počet kroků bez omezení $u(0)$

$G(s) = \frac{\text{[input]}}{(\text{[input]} s+1) (\text{[input]} s+1)}$

$T = \text{[input]}$ Konst. ze statické charakteristiky [input]

Parametry

$p1 = \text{[input]}$ $q0 = \text{[input]}$

$p2 = \text{[input]}$ $q1 = \text{[input]}$

$q2 = \text{[input]}$

OK Zavřít

Obrázek 16 – Metoda konečného počtu kroků bez omezení $u(0)$

3.7.2 Metoda konečného počtu kroků s omezením $u(0)$

V metodě konečného počtu kroků s omezením $u(0)$ se liší v nastavení od předešlé metody bez omezení $u(0)$ pouze tím, že je nutno zadat hodnotu parametru q_0 . Dále ještě přibyly parametry p_3 a q_3 , které jsou vypočítány s ostatními parametry po stisknutí tlačítka *OK*.

Metoda konečný počet kroků s omezením $u(0)$

$G(s) = \frac{\text{[]}}{(\text{[] } s+1) (\text{[] } s+1)}$

$T = \text{[]}$ Konst. ze statické charakteristiky []

Parametry

$q_0 = \text{[]}$

$p_1 =$ $q_1 =$

$p_2 =$ $q_2 =$

$p_3 =$ $q_3 =$

OK Zavřít

Obrázek 17 – Metoda konečného počtu kroků s omezením $u(0)$

3.8 Metoda umístění pólů

Poslední z použitých metod je určení koeficientů PSD regulátoru na základě umístění pólů.

3.8.1 Metoda umístění pólů pro aperiodický průběh

V okně této metody se nastaví základní parametry viz 3.4.1 a kořeny charakteristického polynomu r_1 , r_2 , r_3 a r_4 v rozsahu $0 \leq r_i \leq 1$. Další parametry p_1 , q_0 , q_1 a q_2 jsou vypočítány automaticky po stisknutí tlačítka OK.

Metoda umístění pólů - aperiodický

$G(s) = \frac{\text{[input]}}{(\text{[input]} s+1) (\text{[input]} s+1)}$

$T = \text{[input]}$ Konst. ze statické charakteristiky [input]

Kořeny polynomu
volíme v rozsahu $0 \leq r_i \leq 1$

$r_1 = \text{[input]}$ $r_3 = \text{[input]}$
 $r_2 = \text{[input]}$ $r_4 = \text{[input]}$

Parametry

$q_0 =$ $p_1 =$
 $q_1 =$
 $q_2 =$

OK Zavřít

Obrázek 18 – Metoda umístění pólů pro aperiodický průběh

3.8.2 Metoda umístění pólů pro periodický průběh

Nastavení okna *metody umístění pólů pro periodický průběh* se liší od předešlé metody pouze v nastavení kořenů charakteristického polynomu. Hodnoty r_1 , r_2 , r_3 a r_4 se volí v rozsahu $0 \leq r_i \leq 1$ a dva z nich komplexně sdružené, nebo se volí v rozsahu $-1 \leq r_i \leq 0$.

Metoda umístění pólů - periodický

$G(s) = \frac{\text{[input]}}{(\text{[input]} s+1) (\text{[input]} s+1)}$

$T = \text{[input]}$ Konst. ze statické charakteristiky [input]

Kořeny polynomu
 volíme v rozsahu $0 \leq r_i \leq 1$
 a dva z nich zvolíme komplexně sdružené
 nebo volíme v rozsahu $-1 \leq r_i \leq 0$

$r_1 = \text{[input]}$ $r_3 = \text{[input]}$
 $r_2 = \text{[input]}$ $r_4 = \text{[input]}$

Parametry

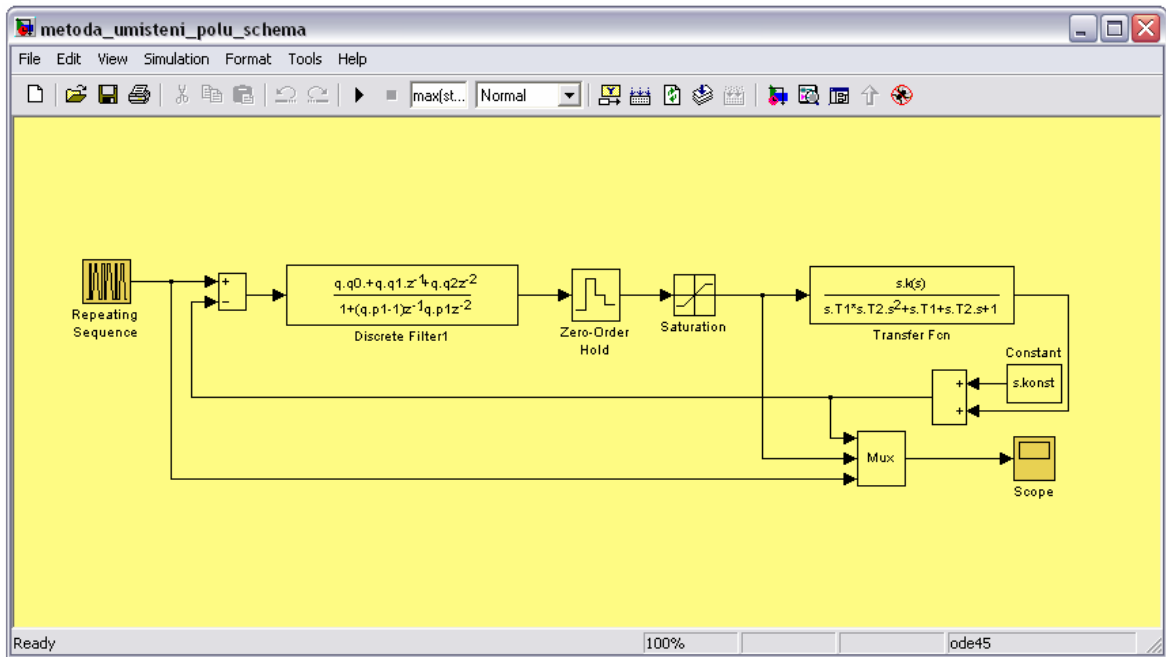
$q_0 = \text{[input]}$ $p_1 = \text{[input]}$
 $q_1 = \text{[input]}$
 $q_2 = \text{[input]}$

OK Zavřít

Obrázek 19 – Metoda umístění pólů pro periodický průběh

3.9 Simulační schémata

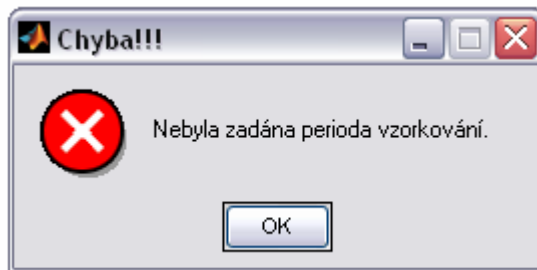
Nedílnou součástí každé metody je i simulační schéma. Veškerá schémata byly vytvořeny v Simulinku viz 2.1.1. Každé schéma využívá zadaných či vypočítaných hodnot danou metodou regulace. V simulačním schématu je nutno zadat žádanou hodnotu w *Repeating Sequence*, blok je označen tmavě žlutou barvou. Taktéž je vybarven i blok pro otevření simulačního průběhu *Scope*. Ostatní bloky není nutno jakkoli měnit, či nastavovat.



Obrázek 20 – Simulační schéma

3.10 Ochrana

Další částí programu je ochrana jednotlivých oken před spuštěním simulace, aniž by byli zadány všechny potřebné hodnoty. Stane-li se tak a uživatel se pokusí o spuštění bez vyplnění všech parametrů, vyskočí výstražné okno s informací, který parametr ještě nebyl zadán.



Obrázek 21 – Chyba

ZÁVĚR

Hlavní náplní této bakalářské práce bylo vytvořit programovou podporu laboratorních úloh předmětu *Teorie automatického řízení II*. Byl vytvořen program SILAU, sloužící k návrhu parametrů regulátoru za pomoci zvolených diskrétních metod syntézy. Program je přímo určen k návrhu parametrů regulátoru pro soustavu 2. řádu bez dopravního zpoždění, dle uvedeného zadání k jednotlivým laboratorním úlohám předmětu. Programová podpora je vytvořena v prostředí MATLAB + SIMULINK.

Dalším cílem bylo vytvořit vzorové protokoly k jednotlivým laboratorním úlohám předmětu. Vytvořené protokoly demonstrují využití programu SILAU a jsou pomůckou studentům ve zpracování samostatných prací ve zmíněném předmětu. Protokoly obsahují jednotlivé zdrojové kódy, které byly použity v programovém prostředí k výpočtu parametrů regulátoru. Tyto materiály jsou uvedeny v přílohách PI.

Jak program SILAU, tak i vypracované protokoly by měly ulehčit studentům práci v laboratořích. V uvedeném programu je možno ověřit veškerou správnost vypočítaných parametrů regulátoru k jednotlivým úlohám.

ZÁVĚR V ANGLIČTINĚ

The main goal of this bachelor thesis was to create a program support for the *Theory of Automatic Control II* course. A program named SILAU was created to generate possible parameters of regulators with the help of chosen discrete methods of synthesis. The program is designed to generate parameters of a regulator for order 2 system without traffic delay, in accordance with the specific assignment of the laboratory tasks of the course. The program support is designed in MATLAB + SIMULINK environment.

Another goal was to create master protocols for each laboratory task of the course. The created protocols illustrate the usage of SILAU program. They can also be used as a tool by students when working on individual assignments in this course. The protocols contain individual source codes, which were used to calculate the parameters of regulators in the program environment. These materials are included in the PI attachments.

Both the SILAU program and the protocols should simplify the work of students in the laboratories. The validity of the calculated regulators' parameters can be verified in the above mentioned program.

SEZNAM POUŽITÉ LITERATURY

- [1] Balátě, J. *Automatické řízení*. 2. přeprac. vyd. Praha : BEN - technická literatura, 2004. 664 s. ISBN 80-7300-148-9.
- [2] Gerlich, V.: Podpora cvičení předmětu *Teorie automatického řízení II*. Bakalářská práce, FAI UTB ve Zlíně 2006
- [3] Šulc, B. Vítěčková, M.: *Teorie a praxe návrhu regulačních obvodů*. Vydavatelství ČVUT, Praha, 2004
- [4] Modrák, O.: *Syntéza číslicových regulátorů – studijní materiály Technická Univerzita v Liberci*, 2004
- [5] Navrátil, P.: *Teorie automatického řízení II – diskrétní systémy* [online]. Dostupné z URL:.
- [6] Švarc, I.: *Automatizace Automatického řízení*. Akademické nakladatelství CERM, Brno, 2002.
- [7] The MathWorks – Online Documentation [online]. Dostupné z URL. <http://www.mathworks.com/>
- [8] Vašek, V.: *Teorie automatického řízení II*. Skriptum VUT v Brně, 1990.
- [9] HUMUSOFT – Online Documentation [online]. Dostupné z URL: <http://www.humusoft.cz/matlab/simulink.htm>

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

a	pásmo necitlivosti
G	přenosová funkce
$G(s)$	L-přenos (Laplaceův přenos)
k	zesílení regulátoru
K_D	zesílení derivační složky
K_I	zesílení integrační složky
K_P	zesílení proporcionální složky
K_{Pmax}	maximální hodnota penalizační konstanty
p_i	parametry jmenovatele regulátoru
PI	regulátor se s členy P – proporcionální, I – integrační
PID	regulátor se s členy P – proporcionální, I – integrační, D – derivační
ppp	pásmo působnosti penalizace
q_i	parametry čitatele regulátoru
r_i	kořeny polynomu
s	komplexní proměnná, algebraický doplněk
T	perioda vzorkování
T_i	časové konstanty
T_n	doba náběhu
T_u	doba průtahu
u	akční veličina
u	akční zásah
w	řídící veličina (žádaná hodnota)

SEZNAM OBRÁZKŮ

Obrázek 1 – Normovaná přechodová charakteristika statické soustavy vyššího řádu	10
Obrázek 2 – Hlavní okno Matlabu.....	16
Obrázek 3 – Simulink	17
Obrázek 4 – GUIDE Quick Start	18
Obrázek 5 – GUIDE Quick Start	18
Obrázek 6 – Lišta s ovládacími prvky hlavního okna pro vytváření dialogů.....	19
Obrázek 7 – Hlavní panel silau.....	21
Obrázek 8 – Načtení souboru s daty	22
Obrázek 9 – Identifikace soustavy.....	23
Obrázek 10 – Dvoupolohová regulace bez penalizace	24
Obrázek 11 – Dvoupolohová regulace s penalizací.....	25
Obrázek 12 – Ziegler Nichols odhadem	26
Obrázek 13 – Ziegler Nichols metodou.....	27
Obrázek 14 – Inverze dynamiky PID – regulátor	28
Obrázek 15 – Inverze dynamiky PI – regulátor	29
Obrázek 16 – Metoda konečného počtu kroků bez omezení $u(0)$	30
Obrázek 17 – Metoda konečného počtu kroků s omezením $u(0)$	31
Obrázek 18 – Metoda umístění pólů pro aperiodický průběh	32
Obrázek 19 – Metoda umístění pólů pro periodický průběh	33
Obrázek 20 – Simulační schéma.....	34
Obrázek 21 – Chyba	35

SEZNAM TABULEK

Tabulka 1 - Poloha kořenů charakteristické rovnice.....	13
---	----

SEZNAM PŘÍLOH

- P I Identifikace regulované soustavy
- P II Číslicový dvoupolohový regulátor s penalizací
- P III Diskrétní PID regulátor Ziegler Nichols
- P IV Diskrétní PID regulátor - Metoda požadovaného modelu (inverze dynamiky)
- P V Zpětnovazební diskrétní regulátor Konečný počet kroků regulace – klasický přístup
- P VI Zpětnovazební diskrétní PID regulátor Určení koeficientů PSD regulátoru na základě umístění póly