

Monika Bakošová
Miroslav Fikar
Ľuboš Čirka

Základy automatizácie

Laboratórne cvičenia zo základov automatizácie

STU v Bratislava, 2003

© doc. Ing. Monika Bakošová, CSc., doc. Dr.-Ing. Miroslav Fikar, Ing. Ľuboš Čírka

Lektori: prof. Ing. I. Taufer, DrSc.
doc. Ing. B. Rohaľ-Ilkiv, CSc.

Publikácia neprešla redakčnou úpravou

Schválilo vedenie FCHPT STU v Bratislave - č. 6/2002

ISBN 80-227-1831-9

LCZA na Internet
<http://www.ka.chtf.stuba.sk/lcza>

Obsah

Predhovor	7
1 Úvod do MATLABu a Simulinku	9
1.1 Pracovný adresár	9
1.1.1 Vytvorenie pracovného adresára	9
1.1.2 Otvorenie MATLABu a nastavenie sa do pracovného adresára	9
1.2 Základy MATLABu	10
1.2.1 Úlohy z MATLABu	13
1.3 Základy Simulinku	14
1.3.1 Otvorenie simulinkovej schémy	14
1.3.2 Nastavenie parametrov blokov v schéme	14
1.3.3 Spustenie a zastavenie simulácie	14
1.3.4 Úlohy zo Simulinku	15
Literatúra	17

Predhovor

Publikácia „Laboratórne cvičenia zo základov automatizácie“ je určená hlavne poslucháčom III. ročníka študujúcim na Fakulte chemickej a potravinárskej technológie STU v Bratislave. Je zameraná na úvod do modelovania a riadenia procesov chemických a potravinárskych technológií, návrh jednoduchých regulačných obvodov ako aj na možnosti simulácie dynamiky procesov a ich riadenia.

Obsah publikácie je rozdelený do desiatich kapitol, z ktorých každá tvorí základ minimálne jedného laboratórneho cvičenia. Prvá kapitola sa venuje úvodu do výpočtového systému MATLAB, ktorý sa na laboratórnych cvičeniach intenzívne využíva. Ďalšie kapitoly sú venované trom väčším tematickým celkom.

V prvej časti je rozpracovaný matematický aparát potrebný k základom automatizácie. Vysvetľované sú Laplaceova transformácia, riešenie diferenciálnych rovníc, prenosové funkcie a základy ich algebry, prechodové charakteristiky a vzťah dynamiky procesov s pólmí a nulami prenosovej funkcie.

Druhá časť sa zaoberá modelovaním vybraných procesov chemických a potravinárskych technológií a simuláciou ich dynamiky na počítači v prostredí MATLABu a jeho podporného systému na modelovanie a simuláciu na báze blokových diagramov – Simulinku.

Tretia časť je venovaná problematike riadenia procesov. Vychádza z uzavretého regulačného obvodu a princípu spätnej väzby. Analyzuje stabilitu obvodov a vysvetľuje niektoré postupy identifikácie procesov a návrhu jednoduchých regulátorov.

Kvôli bezproblémovému zvládnutiu preberanej témy sú príklady v skriptách rozdelené na riešené príklady, ktoré názorne objasňujú teóriu, neriešené príklady na samostatné precvičenie učiva a úlohy, ktoré sú náplňou práce na cvičeniach.

Dodatok sa zaoberá sa možnosťou riešenia úloh laboratórnych cvičení pomocou Internetu prostredníctvom MILABu (MATLABu cez Internet). Celé skriptá a všetky úlohy sú on-line prístupné na internetovej stránke Katedry informatizácie a riadenia procesov FCHPT STU (<http://www.ka.chtf.stuba.sk>). Predmet „Laboratórne cvičenia zo základov automatizácie“ je jedným z prvých na FCHPT, ktorý poskytuje túto možnosť, a veríme, že sa stane zdrojom inšpirácie pre pedagógov a obohatí možnosti vzdelávania študentov. Na konci skoro každej kapitoly možno nájsť tak príkazy a funkcie MATLABu, týkajúce sa danej problematiky, ako aj opis postupu riešenia úlohy cez Internet. Prakticky je teda možné riešiť úlohy laboratórnych cvičení bez fyzickej prítomnosti v počítačových učebniach, či bez nutnosti inštalácie MATLABu na počítač. Týmto spôsobom je možné overiť si pred cvičením správnosť doma vypracovanej prípravy, či nechať si vypočítať niektoré čiastkové kroky riešenia.

Autori ďakujú prof. Ing. I. Tauferovi, DrSc. a doc. Ing. B. Rohaľovi-Ilkivovi, CSc. za cenné pripomienky, ktoré pomohli odstrániť mnohé nedostatky rukopisu. Vďaka tiež patrí kolektívu pracovníkov Katedry informatizácie a riadenia procesov a zvlášť Ing. M. Ondrovičovej a Ing. M. Kvasnicovi, ktorých návrhy a pripomienky pomohli pri vytvorení konečnej podoby rukopisu.

Kapitola 1

Úvod do MATLABu a Simulinku

Cieľom cvičenia je vytvorenie pracovných adresárov a oboznámenie sa s výpočtovým systémom MATLAB – Simulink tak, aby ho bolo možné používať v laboratórnych cvičeniach zo základov automatizácie.

1.1 Pracovný adresár

1.1.1 Vytvorenie pracovného adresára

Vytvorte vlastný pracovný adresár `ddhh`, kde `ddhh` = deň a hodina cvičenia, keď cesta do neho je `d:\user\za\ddhh`.

Postup

Na vytvorenie pracovného adresára použijeme ikonu TENTO POČÍTAČ na pracovnej ploche displeja. Otvoríme ju ĽK (ĽK = dvojité kliknutím ľavým tlačidlom myši) na ikone. (Kto je zvyknutý používať PRIESKUMNÍK, môže ho použiť rovnakým spôsobom.) V otvorenom okne nájdeme diskovú jednotku D, opäť ju otvoríme ĽK, nájdeme adresár USER, ĽK, nájdeme adresár ZA, ĽK. Okno, ktoré máme teraz otvorené je buď prázdne alebo sú tam už vytvorené adresáre niektorých skupín. Vytvoríme si ten svoj. V menu okna ĽK (ĽK = jedným kliknutím ľavým tlačidlom myši) otvoríme ponuku SÚBOR. V nej sa kurzorom myši nastavíme na možnosť NOVÝ OBJEKT a ĽK vyberieme PRIEČINOK (ZLOŽKA). V tejto chvíli sa v okne objaví žltá ikona s názvom NOVÝ PRIEČINOK (NOVÁ ZLOŽKA), ktorá je orámovaná a vysvietená namodro. Bez akéhokoľvek ďalšieho stláčania tlačidla myši alebo klávesov jednoducho začneme písať názov nášho pracovného adresára vo forme `ddhh` (krúžok, ktorý začína cvičenie v pondelok o 7 h napíše `po07`, krúžok, ktorý začína cvičenie v utorok o 13 h, napíše `ut13` a pod.). Po vložení názvu stlačíme kláves ENTER.

Upozornenia

- Doporučujeme do názvu adresára nevkladať žiadne iné znaky ani medzery (nie všetky znaky akceptovateľné pre názvy adresárov v rámci operačného systému sú použiteľné v MATLABe).
- Adresár, ktorý bol takto vytvorený, sa bude používať celý semester, preto si treba cestu do pracovného adresára zapamätať.

1.1.2 Otvorenie MATLABu a nastavenie sa do pracovného adresára

Pre prácu s MATLABom a Simulinkom treba vždy na začiatku cvičenia MATLAB otvoriť. To znamená, že na pracovnej ploche displeja nájdeme ikonu MATLAB a otvoríme ju ĽK. Pri otváraní MATLABu musíme byť trpezliví, lebo pri viacnásobnom ĽK sa nám otvorí MATLAB viackrát.

Preto vždy po otvorení pracovného okna MATLABu skontrolujeme hlavný panel (lišta v dolnej časti displeja), či na ňom máme len jedno tlačidlo MATLABu. Ak ich tam vidíme dve alebo viac, zbytočne otvorené programy zatvoríme. Po správnom otvorení MATLABu vidíme na hlavnom paneli len jedno tlačidlo MATLABu a v pracovnom okne MATLABu zas cestu do aktuálneho pracovného adresára v podobe buď

```
ans =  
D:\USER  
>>
```

alebo

```
ans =  
D:\USER\ZA  
>>
```

To znamená, že po otvorení MATLABu je ešte potrebné nastaviť sa do vlastného pracovného adresára `ddhh`. Urobí sa to napísaním príkazu

```
>>cd za\ddhh
```

ak sa MATLAB nastavil do adresára `USER`, alebo napísaním príkazu

```
>> cd ddhh
```

ak sa MATLAB nastavil do adresára `ZA`. Vykonanie príkazu sa zabezpečí stlačením ENTER. V prípade, že je všetko v poriadku, odozva MATLABu má len tvar

```
>>
```

s blikajúcim kurzorom za šípkami. Ak sme urobili nejakú chybu, MATLAB ju signalizuje výpisom chybového hlásenia. Overenie správnosti nastavenia do vlastného pracovného adresára sa urobí príkazom

```
>> pwd
```

na ktorý MATLAB odpovie

```
>> ans =  
D:\USER\ZA\ddhh  
>>
```

Teraz už možno začať pracovať s MATLABom.

Upozornenie

Postup nastavenia do vlastného pracovného adresára sa opakuje po každom novom spustení MATLABu, preto si ho treba zapamätať.

1.2 Základy MATLABu

Na laboratórnych cvičeniach zo základov automatizácie v MATLABe potrebujeme vedieť:

1. priradiť číselnú hodnotu premennej, napr. priradenie číselných hodnôt do premenných a_1 , b_1 :

```
>> a1=2;  
>> b1=3-2i  
b1 =  
3.0000 - 2.0000i
```

1.2. ZÁKLADY MATLABU

2. načítať maticu alebo vektor, napr. načítanie matice m_1 s 2 riadkami a 3 stĺpcami:

```
>> m1=[1 2 3;4 5 6]
m1 =
     1     2     3
     4     5     6
```

3. načítať koeficienty polynómu, napr. načítanie koeficientov polynómu $p_1 = 7s^4 + 6s^3 + s - 5$ vo vektorovom tvare:

```
>> p1=[7 6 0 1 -5]
p1 =
     7     6     0     1    -5
```

alebo v polynomickej tvare¹

```
>> p1=7*s^4+6*s^3+s-5
p1 =
      p1 = -5 + s + 6*s^3 + 7*s^4
```

4. urobiť základné aritmetické operácie súčet (+), rozdiel (−), súčin (*), podiel (/), mocninu (^) s číslami, maticami a vektormi, urobiť transpozíciu matice alebo vektora (') ('= apostrof), napr. transponovaná matica k matici m_1 :

```
>> m1'
ans =
     1     4
     2     5
     3     6
```

5. zapísať aritmetické výrazy, pričom treba mať na pamäti, že pri vyhodnocovaní výrazu MATLAB postupuje zľava doprava a priorita operácií klesá v poradí (), ^, * / (sú rovnocenné), + − (sú rovnocenné)
6. vypísať číselnú hodnotu premennej na obrazovku, a to jednoduchým napísaním jej názvu do príkazového riadku a stlačením ENTER.

V MATLABe potrebujeme poznať základné funkcie pre výpočet:

1. odmocniny čísla (`sqrt(a)`)
2. exponenciálnej funkcie (`exp(a)`)
3. absolútnej hodnoty (`abs(a)`)
4. goniometrických funkcií v radiánoch (napr. `sin(a)`, `cos(a)`, `tan(a)`)
5. inverzie matice (`inv(m)`), determinantu matice (`det(m)`)
6. koreňov polynómu (`roots(p)`)
7. polynómu z jeho daných koreňov (`poly(k)`)
8. súčinu polynómov (`conv(p1, p2)`) – ak sa používajú polynómy vo vektorovom tvare, napr. `[3 1 2]`, alebo `p1*p2` – ak sa používajú polynómy v polynomickej tvare, napr. `3*s^2+s+2`)

¹s nainštalovaným Polynomickým toolboxom

9. podielu polynómov (`deconv(p1, p2)` – ak sa používa tvar `[3 1 2]`, alebo `p1/p2` – ak sa používa tvar `3*s^2+s+2` a polynómy sú deliteľné bez zvyšku).

Ďalšie užitočné príkazy pre prácu s MATLABom:

1. Na získanie informácie o definovaných premenných slúži príkaz `who`, napr.:

```
>> who
Your variables are:
```

```
a1      ans      m1      p1
a2      b1      m2
```

alebo príkaz `whos`, napr.:

```
>> whos
```

Name	Size	Bytes	Class
a1	1x1	8	double array
a2	1x1	8	double array
ans	1x3	6	char array
b1	1x1	16	double array (complex)
m1	3x3	72	double array
m2	2x5	80	double array
p1	1x6	48	double array

Grand total is 31 elements using 238 bytes

2. Na vypísanie matlabovských súborov (s rozšírením `.m`) a simulinkových schém (s rozšírením `.mdl`) nachádzajúcich sa v pracovnom adresári slúži príkaz `what`, napr.:

```
>> what
```

M-files in the current directory

```
reak
```

MDL-files in the current directory

```
reaktor      schema
```

3. Na získanie pomoci v MATLABe sa používa príkaz `help`.

Upozornenia

- V predošlom texte ale i v ďalších kapitolách kvôli názornosti uvádzame nielen samotný príkaz, ktorý píše užívateľ, ale i odozvu MATLABu. Všetko, čo nasleduje medzi značkami `>>` a koncom riadku, je príkaz, ktorý píše užívateľ. MATLAB o vykonaní príkazu alebo neinformuje (v prípade, že posledný zadaný znak príkazu bola bodkočiarka), alebo vypíše informáciu o vykonaní príkazu (ak príkaz nekončil bodkočiarkou) alebo vypíše chybové hlásenie (ak sa v príkaze vyskytla chyba). Po odozve MATLABu sa na novom riadku vždy objavajú znaky `>>`, za ktoré sa píše nový príkaz.
- MATLAB používa desatinnú bodku.
- MATLAB pracuje aj s komplexnými číslami.

1.2. ZÁKLADY MATLABU

- MATLAB rozlišuje malé a veľké písmená v názvoch premenných. To znamená, že premenná `a` je iná ako premenná `A`.
- V prípade, že v MATLABe vykonáte nejakú aritmetickú operáciu a nepriradíte ju do konkrétnej premennej, MATLAB jej výsledok automaticky uloží do premennej `ans`.
- Pri sčítaní polynómov načítaných pomocou koeficientov v hranatých zátvorkách treba polynóm nižšieho stupňa doplniť nulami tak, aby počet koeficientov sčítovaných polynómov bol rovnaký.

1.2.1 Úlohy z MATLABu

1. Priradte premenným a_1, a_2, a_3 , postupne číselné hodnoty 5; 0,5; -20,05.
2. Priradte premenným b_1, b_2 číselné hodnoty $2 + i$, $-2 - 5i$.
3. Uložte do premenných m_1, m_2, m_3 nasledovné matice a vektor a presvedčte sa výpisom ich hodnôt na obrazovku, že ste ich načítali správne.

$$\begin{pmatrix} -2 & 0,5 & 16 \\ -4 & 25,1 & 2 \\ 8 & 19,6 & -5 \end{pmatrix}, \quad \begin{pmatrix} 1 & 0,54 & -123 & 0 & 12 \\ -23 & 123,7 & 1 & -9,01 & 3 \end{pmatrix}, \quad \begin{pmatrix} 1 \\ 2 \\ 6 \end{pmatrix}$$

4. Uložte do premenných p_1, p_2 koeficienty nasledovných polynómov.

$$s^5 + 4s^4 + 12s^3 + 6s^2 + 49s + 4$$
$$4s^4 + 2s^2 + s - 8$$

5. Sčítajte premenné a_1, a_2 . Výsledok priradte do premennej c_1 .
6. Odčítajte od premennej a_3 premennú a_2 . Výsledok priradte do premennej c_2 .
7. Vynásobte premenné b_1, b_2 . Výsledok priradte do premennej c_3 .
8. Vydeľte premennú a_1 premennou a_2 . Výsledok priradte do premennej c_4 .
9. Vypočítajte číselnú hodnotu výrazu

$$\frac{a_1^3 + a_2(a_3 - a_1)}{a_1 + a_2}.$$

10. Vypočítajte číselnú hodnotu výrazu $5(1 - e^{-0,2})$.
11. Do premennej n_1 priradte transponovaný vektor m_3 .
12. Vynásobte vektor m_3 sprava vektorom n_1 . Výsledok priradte do premennej n_2 .
13. Vynásobte vektor n_1 sprava vektorom m_3 . Výsledok priradte do premennej n_3 .
14. Sčítajte matice m_1 a n_2 . Výsledok priradte do premennej n_4 .
15. Vypočítajte inverznú maticu k matici n_4 . Výsledok priradte do premennej n_5 .
16. Vypočítajte determinant matice n_2 . Výsledok priradte do premennej d_1 .
17. Vypočítajte korene polynómov p_1, p_2 . Výsledky postupne priradte do premenných k_1, k_2 .
18. Z koreňov uložených do premennej k_1 určite polynóm, ktorému tieto korene patria. Výsledok vložte do premennej p_3 .

19. Porovnajte polynómy p_1 , p_3 ich výpisom na obrazovku.
20. Určite polynóm, ktorého korene sú -2 , $-4 + 2i$, $-4 - 2i$. Priradte ho do premennej p_4 .
21. Sčítajte polynómy p_1 , p_2 .
22. Vypočítajte súčin polynómov p_1 , p_2 , p_4 . (Ak použijete funkciu `conv()`, potom to treba robiť na dvakrát, pretože jej argumentom môžu byť len 2 polynómy.)

1.3 Základy Simulinku

Simulink sa bude v rámci predmetu laboratórne cvičenia zo základov automatizácie používať na simuláciu dynamických vlastností systémov len užívateľsky. To znamená, že blokové schémy budú pripravené a požiadavky na ich použitie sú nasledovné:

1. vedieť otvoriť konkrétnu simulinkovú schému,
2. vedieť nastaviť parametre blokov,
3. vedieť spustiť a zastaviť simuláciu,
4. vedieť prezrieť a spracovať grafické a numerické výsledky.

1.3.1 Otvorenie simulinkovej schémy

Simulinková schéma sa otvorí napísaním jej mena do príkazového riadku okna MATLABu a následným stlačením ENTER. Meno schémy sa píše bez rozšírenia `.mdl`, napr.:

```
>> schema
```

1.3.2 Nastavenie parametrov blokov v schéme

Pre nastavenie parametrov príslušného bloku je potrebné tento blok najskôr otvoriť L2K. V otvorenom bloku už vidno riadky, do ktorých sa vpíše jedna z nasledovných možností

1. konkrétne čísla, matice, vektory alebo polynómy, pričom treba dodržať formát zápisu,
2. mená príslušných premenných. V tomto prípade treba pred spustením simulácie použité premenné definovať v okne MATLABu.

Pre dobrú orientáciu je nad každým riadkom (a prípadne i v ňom) v otvorenom bloku napísané, aké premenné treba doňho vložiť. Takisto si treba všímať mená jednotlivých blokov v schéme, uľahčí to orientáciu v nej. V prípade, že je simulácia príliš pomalá, rýchla, dlhá alebo krátka, máme možnosť urobiť zmeny v rýchlosti simulácie zmenou veľkosti integračného kroku numerickej integračnej metódy (**Fixed step size** pri použití metódy s konštantným integračným krokom alebo **Max step size** pri použití metódy s premenným integračným krokom) a zmeny v trvaní simulácie zmenou konečného času simulácie (**Stop time**). Robí sa to v ponuke **Simulation** v menu okna simulačnej schémy a v nej v ponuke **Parameters**.

1.3.3 Spustenie a zastavenie simulácie

Simulácia sa spustí vybratím ponuky **Simulation – Start** v menu okna simulačnej schémy, stlačením čiernej šípky smerujúcej doprava na paneli nástrojov alebo stlačením kombinácie klávesov CTRL+T. Simulácia sa korektne ukončí vybratím ponuky **Simulation – Stop**, alebo stlačením kombinácie klávesov CTRL+T, alebo stlačením príslušného tlačidla na paneli nástrojov.

1.3. ZÁKLADY SIMULINKU

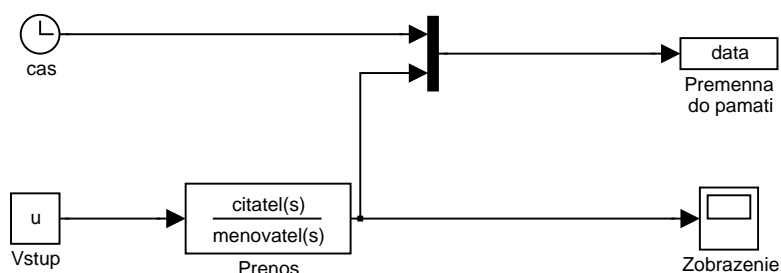
Upozornenia

- V schémach je použité grafické zobrazenie s automatickým nastavením vertikálnej osi. Po skončení simulácie sa automaticky upraví rozsahy osí stlačením tlačidla „ďalekohľad“ na paneli nástrojov grafu.
- Keby sa po spustení simulácie neobjavilo grafické znázornenie simulácie na obrazovke, otvoríme si ho Ľ2K na ikone **Zobrazenie** v schéme.
- Nenastavujú sa parametre blokov **Cas**, **Mux**, **Demux**, **Sum**.
- Pri simuláciách na laboratórnych cvičeniach zo základov automatizácie je vhodné kvôli spracovaniu výsledkov použiť pri nastavovaní parametrov simulácie metódu s konštantným integračným krokom.

1.3.4 Úlohy zo Simulinku

Príklad 1.3.1:

Sledujte pomocou simulinkového programu **schema** (obr. 1.1) dynamické vlastnosti systému opísaného prenosom.



Obr. 1.1 Schéma na sledovanie dynamických vlastností systému opísaného prenosom - **schema.mdl**

Poznámka

Pojem prenos sa preberie na prednáškach v 2. týždni. Pre objasnenie uvádzame jeho definíciu: prenos je definovaný ako podiel Laplaceovho obrazu výstupnej veličiny a Laplaceovho obrazu vstupnej veličiny pri nulových začiatkových podmienkach. Po odvodení má tvar zlomku s polynómom v čitateli i menovateli, pričom u fyzikálne realizovateľných systémov stupeň polynómu v čitateli je menší alebo rovný stupňu polynómu v menovateli.

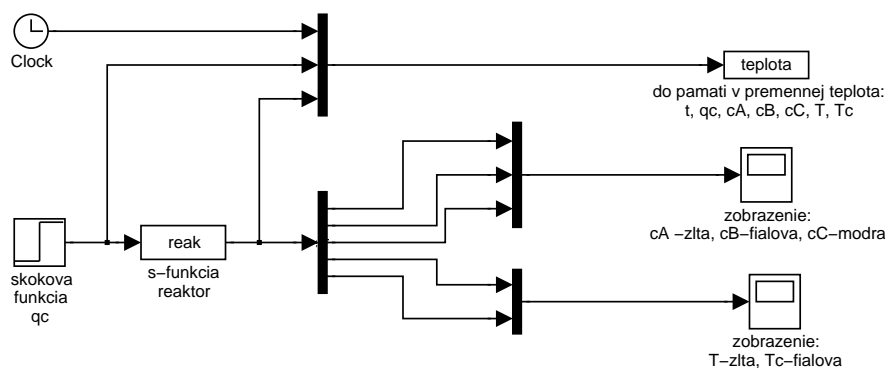
Postup

1. Otvorte program **schema**.
2. Definujte parametre blokov **Vstup** a **Prenos**. V bloku **Vstup** to znamená priradiť číselnú hodnotu premennej u (napr. 2), v bloku **Prenos** treba definovať premennú **citatel** (napr. $2s + 1$) a premennú **menovatel** (napr. $4s^3 + 3s^2 + 6s + 2$) vo vektorovom tvare. Ostatné bloky sú definované.
3. Spustite simuláciu.
4. Pozrite si grafické výsledky simulácie (blok **Zobrazenie**).
5. Pozrite si numerické výsledky simulácie vypísaním obsahu premennej **data** na obrazovku.

6. Vyskúšajte si simuláciu s iným vstupom a prípadne i iným prenosom.
7. Zmeňte čas simulácie a prípadne i krok integračnej metódy a simuláciu znova spustite.

Príklad 1.3.2:

Sledujte pomocou simulinkovej schémy **reaktor** (obr. 1.2) dynamické vlastnosti chemického reaktora najskôr v ustálenom stave a potom po skokovej zmene vstupnej veličiny, ktorou je prietok chladiaceho média q_c vstupujúceho do plášťa reaktora. Na výstupe z reaktora sa sledujú koncentrácie reagujúcich zložiek A (reaktant), B (hlavný produkt), C (vedľajší produkt), teplota reakčnej zmesi T a teplota v plášti reaktora T_c .



Obr. 1.2 Schéma na sledovanie dynamických vlastností reaktora - **reaktor.mdl**

Postup

1. Otvorte program **reaktor**.
2. Spustite simuláciu s hodnotou **Initial value** aj **Final value** v bloku **skokova funkcia** rovnou 0,004.
3. Pozrite si grafické výsledky simulácie.
4. Pozrite si numerické výsledky simulácie vypísaním obsahu premennej **teplota** na obrazovku.
5. Realizujte v čase 10 min (položka **Step time** v bloku **skokova funkcia** sa nastaví na hodnotu 10) skokovú zmenu prietoku chladiaceho média z hodnoty 0,004 na hodnotu $0,008 \text{ m}^3 \text{ min}^{-1}$ (položka **Final value** v bloku **skokova funkcia** sa nastaví na hodnotu 0,008).
6. Pozrite si opäť grafické a numerické výsledky simulácie.
7. Ďalšími simuláciami sledujte vplyv prietoku chladiaceho média na sledované veličiny v chemickom reaktore.

Literatúra

Mészáros, A., Danko, J., Mikleš, J. a Bakošová, M. *Základy automatizácie*. STU Press, Bratislava, 1997.

Mikleš, J., Dostál, P. a Mészáros, A. *Riadenie technologických procesov*. STU, Bratislava, 1994.