

Využití Matlabu ke zkvalitnění výuky odborných předmětů

Janeček Josef, MTI FM TU Liberec

Abstrakt :

V příspěvku si autor klade za cíl upozornit odbornou veřejnost, zejména učitele technických předmětů středních odborných škol zaměřených na elektroniku, elektrotechniku, strojírenství, automatizaci a mechatroniku, na možnosti zkvalitnění výuky s podporou moderního analytického a simulačního prostředí Matlab-Simulink.

Byl bych velmi rád, aby nebyl tento příspěvek chápán jako komerční, propagační text. Zdůrazňuji, že nejsem v žádném případě dealerem tohoto produktu, nemám žádný finanční zájem na zvýšení prodeje tohoto programu a rozhodně nemám žádný profit z rozšíření tohoto prostředí i do středních škol, kromě ambice zvýšení úrovně výuky a sledování celosvětového trendu.

Od roku 1975 pracuji na liberecké technice, věnuji se výuce předmětů podobného charakteru, které jsem studoval již na vysoké škole. Zabývám se problematikou dynamických systémů, zejména zpětnovazebních, a jejich průmyslovými aplikacemi. Přiznávám, že jsem patrně tímto oborem (který je nesporně součástí všech mechatronických aplikací) mírně profesně poznamenán.

V průběhu mého odborného života jsem zaznamenal dva výrazné zlomy, které posunuly podle mého mínění skokově úroveň oboru. Prvním bylo zavedení analogových počítačů pro simulaci dynamických dějů. Byl to krok od čistě abstraktní teoretické práce s diferenciálními rovnicemi k vizualizaci a reálné práci s dynamikou. Druhým výrazným skokem bylo zavedení číslicových počítačů pro analýzu, syntézu a modelování rozsáhlých nelineárních systémů. O své "místo na slunci" se ucházela řada jednoduchých i velmi propracovaných programových produktů. Nedokážu dost dobře posoudit zda zaslouženě, ale v každém případě velmi razantně si svoji dominanci zajistil programový produkt MATLAB firmy The MathWorks, Inc., který svou modulární koncepcí a nabídkou řady různých specializovaných modulů (tzv. toolboxů) pokrývá širokou škálu technické

činnosti. V současné době je používán takřka na všech technických vysokých školách po celém světě, prosazuje se v průmyslových aplikacích a rozšiřuje se ve vyspělých státech i do středních škol.

Matlab tvoří integrované prostředí pro vědeckotechnické výpočty, modelování, návrhy algoritmů, simulaci, analýzu a prezentaci dat. Umožňuje měření reálných dat fyzikálních systémů, zpracování signálů, návrhy řídicích a komunikačních systémů. Je nástrojem jak pro pohodlnou interaktivní práci, tak pro vývoj širokého spektra aplikací. Původně byl určen pro analýzu signálů a syntézu regulačních obvodů, ale škála specializovaných fakultativních modulů (které lze samostatně selektivně dokoupit) brzy rozšířila jeho použití i do jiných, odlišných oborů, jako např. finančnictví, biologie, chemie, atd. Matlab se stal i účinným nástrojem formální matematiky, včetně propracované vizualizace dat s dokonalými grafickými možnostmi. Simulink je nadstavba Matlabu, hlavní toolbox určený pro modelování a simulaci dynamických systémů. Výhradním distributorem Matlabu pro ČR je firma HUMUSOFT s.r.o., o jejíž obchodní politice by bylo možné diskutovat, pro školy však nabízí výrazné slevy.

Důvody pro využití Matlabu ve výuce odborných předmětů :

- umožňuje v souladu se světovým trendem posunout výuku na novou, kvalitativně vyšší úroveň s interaktivními prvky výuky
- zvýhodňuje absolventy středních škol (kteří přišli alespoň formálně do styku s Matlabem) při studiu na vysoké škole technického směru
- dává možnost pedagogům a učitelům odborných předmětů zvýšit svoji kvalifikaci a využít nástrojů Matlabu pro lepší demonstraci a zpřístupnění probírané látky
- využití grafických možností pro publikační činnost a zpracování referátů (v současné době většina referátů na konferencích a seminářích oboru využívá Matlabu k simulačnímu ověření výsledků včetně grafických výstupů)

Bariéry zavedení Matlabu ve výuce na středních školách :

- přetrvávající jistá konzervativnost některých učitelů a vedení škol, pracnost spojená s inovací studijních plánů
- obava z přílišné "vědeckosti" prostředí a nutnost pracovat s prostředky vyšší matematiky a diferenciálního počtu (skutečně, Matlab předpokládá, ale nevyžaduje hlubší zvládnutí vyšší matematiky, záleží na úrovni práce s ním,

pro středoškolského učitele by neměl být diferenciální počet strašákem, pro studenta může být dokonale uživatelsky skryt)

- zvládnutí příkazů a ovládání rozsáhlého softwarového nástroje (pro práci v konkrétním odborném předmětu se však zpravidla využívá jen malý zlomek všech možností, základní ovládání je prosté, není třeba kupovat zbytečně drahou kompletní konfiguraci, modulární charakter Matlabu umožňuje nakoupit pouze toolboxy, které ke své práci budeme používat)
- manuály, helpy, systémová hlášení a popisy příkazů jsou většinou v angličtině (v současné době je však již dostupná celá řada učebnic a literatura, která je věnována aplikaci Matlabu v češtině, např. [1,2] aj., detailněji viz např. [3])
- cena (cenová nabídka fy HUMUSOFT s.r.o. viz [3], pro školní účely nabízí firma výrazné slevy, před nákupem příslušné licence doporučuji však konzultaci, týkající se výběru nejvhodnější varianty a nákupu pouze těch modulů, které budeme využívat, může to výrazně ovlivnit cenu nejen základní konfigurace, ale i následných aktualizací)

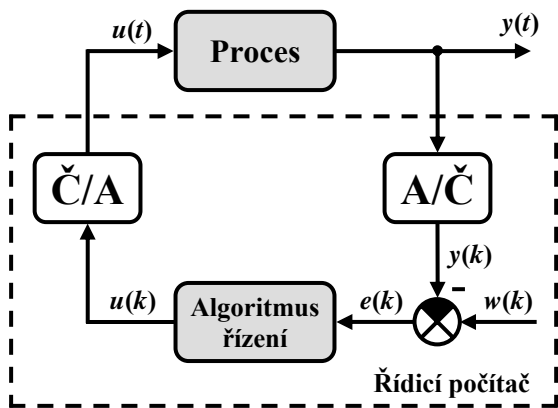
Není v možnostech tohoto krátkého příspěvku ukázat možné použití Matlabu ve středoškolské výuce různých odborných předmětů v plné šíři. Jako inspiromat s řadou jednoduše modifikovatelných příkladů z různých oblastí matematiky, fyziky, elektrotechniky, elektroniky a samozřejmě (vzhledem ke zmíněnému odbornému zaměření autora) i z oblasti automatického řízení a regulace, mohou např. posloužit speciálně pro střední školy zpracované publikace [1,2].

Jako jediný, zato podle mého soudu velmi sugestivní a motivující příklad jak řešit s využitím prostředků Matlabu i velmi komplikované úlohy bez složité teorie, uvádím na tomto místě princip iterační optimalizace parametrů dynamického systému. Přiznávám, že jsem tímto principem (odchován analytickými přístupy a metodami) dodnes velmi okouzlen. Jeho relativní jednoduchostí, téměř žádným omezením vzhledem ke složitosti úlohy, ale hlavně širokým aplikačním záběrem využitelným v různých oborech pro optimalizaci libovolné dynamické funkce určené svými parametry. Příkladem chci ilustrovat snadno aplikovatelný princip i na celou škálu problémů (identifikace, aproximace, ...) bez složité matematiky s velmi silným praktickým dopadem. Velmi doporučuji Vaší pozornosti !

!! VAROVÁNÍ !!

Nenechte se odradit zdánlivou složitostí.
Princip je jednoduchý, snadno ho pochopíte a stane se Vám velmi účinným pomocníkem v řadě technických aplikací.

Jako příklad uvádím úlohu optimalizace parametrů číslicového regulátoru (diskrétní regulátor řídí spojitou část) se zvoleným kvadratickým kritériem s penalizací akčních zásahů.



$$y'''(t) + 6y''(t) + 11y'(t) + 6y(t) = 6u(t)$$

$$J = \sum_{k=0}^{K_h} e^2(kT) + \kappa \nabla u^2(kT) = J(b_0, b_1) \rightarrow \min.$$

$$\nabla u(kT) = b_0 e(kT) + b_1 e[(k-1)T]$$

$$u(kT) = u[(k-1)T] + \nabla u(kT)$$

$$u(k) = u(kT) \quad , \quad e(k) = e(kT)$$

$$T = 0,1 \text{ sec.}$$

Regulátor představuje diskrétní variantu PI regulátoru se zvoleným krokem regulace T , K_h představuje počet kroků regulačního pochodu, ve kterých kritérium posuzujeme (tzv. kritériální horizont) a κ váhu zásahů ∇u v kritériu.

Řešení úlohy analytickými prostředky možné je, je však velmi pracné a vyžaduje zvládnutí aparátu Z-transformace a složitých optimalizačních metod. Jako výrazný kontrast uvádím aplikaci iterační simulační procedury, která velmi jednoduše úlohu řeší bez použití obtížných matematických postupů s minimálním teoretickým vybavením. Navíc nepřináší vůbec žádné obtíže ani složitější dynamika soustavy vzhledem k řádu, nelinearitám, omezením nebo komplikovanému kritériu, která by analytické řešení zcela znemožnila.

Metoda je založena na opakovaném (iteračním) výpočtu zadané úlohy s různými parametry a výběrem jejich nejlepší kombinace vzhledem ke zvolenému kritériu. Jedná se tak ve své podstatě o úlohu hledání extrému (minima) funkce více proměnných, pro kterou má numerická matematika řadu účinných metod (Simplex, Newton, Partan, gradientní metody, atd.). Matlab je jimi jako standardními procedurami vybaven.

Optimalizační proces je založen na spolupráci tří programů (tzv. skriptů, souborů příkazů, které Matlab postupně vykonává). Programy musejí být umístěny v aktuální adresáři Matlabu. V tomto případě model regulačního obvodu „OptPSmdl.mdl” (vytvořený v Simulinku) rozšířený o výpočet kritériálního funkcionálu J je opakovaně spouštěn v prostředí Matlab – Simulink s různými parametry regulátoru. Proces je řízen hlavním programem „OptPS.m“, který využívá standardní proceduru pro hledání extrému funkcí více proměnných. Optimalizovanou funkci definuje „KritPS.m“.

Program „OptPS.m“, skript řídicí průběh optimalizace :

Poznámka : vše co je uvedeno znakem % chápe Matlab jako komentář.

```
%
% OptPS      Optimalizace parametrů PS regulátoru
%            Kvadratické kritérium s penalizací akčních zásahů

disp(' ')
disp(' ')
disp('*****')
disp('Minimalizace kvadratického kritéria')
disp('Kvadratické kritérium s penalizací akčních zásahů')
disp('*****')
disp(' ')
disp(' ')
clear all;    close all;    hold off; format long;

global B0 B1 Tmax Kappa % globální proměnné určené pro přenos hodnot mezi
                        spolupracujícími programy

% ~~~~~
% BLOK ZADÁVANÝCH PARAMETRŮ, KTERÉ ŘÍDÍ VÝPOČET
Tmax=10; % Tmax ... zadaná doba simulace (kriteriální horizont)
dT=0.001; % dT ... tvrdě zadaný simulační krok pro „OptPSmdl“
          (formál.zlepšení simulace - eliminuje
          dynam.volbu simulačního kroku Simulinkem)
Kappa=1; % Kappa ... zadaná váha akčních zásahů v kritériu
          (Kappa malé-rychlé reg.pochody a naopak)
w=1; d=0; % zadané buzení regulačního obvodu (varianta)
%w=0; d=1; % zadané buzení regulačního obvodu (varianta)
B0=4.5; B1=-4.215; % výchozí seřízení regulátoru na začátku optimalizace
x0=[B0 B1]; % naplnění vektoru regulátoru na počátku optimalizace
% ~~~~~
sim('OptPSmdl'); % spouští z AKTUÁLNÍHO ADRESÁŘE simulaci "OptPSmdl" s
                  počátečním nastavením regulátoru
                  % pozn.:názvy m-file "OptPS" a simul.modelu „OptPSmdl“ se
                  musejí lišit i když mají různé přípony

% ~~~~~
% Výstup textu na obrazovku počítače
disp(' Výchozí situace :')
disp(' ')
disp('      B0, B1 ... parametry PS regulátoru')
disp('      Kappa ... váha akčních zásahů,      J ... hodnota kriteriálního
funkcionálu'), B0, B1, Kappa, J
disp(' ')
OptPSmdl % otevře informační okno se simulačním schématem
"OptPSmdl"
figure(1); % otevře okno figure a vykreslí regulační pochod s
           počátečním nastavením

plot([0 Tmax],[0 0], 'k', 'LineWidth',1); hold on;
plot(Time,Y, 'B', 'LineWidth',1); hold on;
% ~~~~~
pause % přeruší výpočet a čeká na "Enter" z klávesnice pro
      pokračování výpočtu

% ~~~~~
% Výstup textu na obrazovku počítače
disp(' ')
disp(' ')
disp('*****')
disp('Začátek iterace')
disp(' ')
% ~~~~~

% ~~~~~
% HLAVNÍ ČÁST OPTIMALIZAČNÍHO VÝPOČTU
OPTIONS=optimset('LargeScale','off','MaxIter',20,'Display','iter');
```

```

% nastavení parametrů optimalizace (4.par.= max.počet kroků iterace)
x=fminsearch('KritPS',x0,OPTIONS);
% varianta : vybraná optimalizace simplexovou metodou
% (1.par.= minimalizovaná účelová funkce definovaná scriptem "KritPS")
%x = fminunc('KritPS',x0,OPTIONS)
% (blokována)varianta : optimalizace gradientní metodou
% ~~~~~

% ~~~~~
% Závěrečný výstup na obrazovku + info
disp(' ')
disp('Konec iterace')
disp('*****')
disp(' ')
disp(' ')
B0opt=x(1); Blopt=x(2);
B0=B0opt; B1=Blopt; sim('OptPSmdl');
disp(' Výsledek optimalizace :')
disp(' ')
disp(' B0opt., Blopt. ... optimální parametry PS regulátoru J ...
optimální hodnota kritériálního funkcionálu'),B0opt, Blopt, J
disp(' ')
OptPSmdl % otevře informační okno se simulačním schématem "OptPSmdl"
figure(1); % otevře okno figure a vykreslí reg.pochod s opt.nastavením
plot(Time,Y,'R','LineWidth',2); hold on;
disp('*****')
disp('Konec příkladu optimalizace kvadratického funkcionálu')
disp('*****')
disp(' ')

```

Průběh výpočtu (výpis monitoru počítače) :

```

*****
Minimalizace kvadratického kritéria
Kvadratické kritérium s penalizací akčních zásahů
*****

```

Výchozí situace :

B0, B1 ... parametry PS regulátoru

Kappa ... váha akčních zásahů, J ... hodnota kritériálního funkcionálu

B0 = 4.500000000000000

B1 = -4.215000000000000

Kappa = 1

J = 5.51754703397270

```

*****

```

Začátek iterace

Iteration	Func-count	min f(x)	Procedure
1	3	3.55176	initial
2	5	3.37381	contract outside
3	7	3.37381	contract inside
4	9	3.37381	contract inside
5	11	3.37381	contract inside
6	13	3.30405	reflect
7	15	3.30405	contract inside
8	17	3.27281	expand
9	19	3.10336	expand
10	21	3.04325	expand
11	23	2.61162	expand
12	24	2.61162	reflect
13	26	1.9294	expand
14	27	1.9294	reflect
15	29	1.34946	expand



16	31	1.22409	reflect
17	33	1.16306	contract outside
18	35	1.16306	contract outside
19	37	1.16306	contract inside
20	39	1.15837	contract inside

Exiting: Maximum number of iterations has been exceeded - increase MaxIter option.

Current function value: 1.158374

Konec iterace

Výsledek optimalizace :

B0opt., B1opt. ... optimální parametry PS regulátoru

J ... optimální hodnota kritériálního funkcionálu

B0opt. = 1.34289493560781

B1opt. = -1.24081651210787

J = 1.15837446304773

Konec příkladu optimalizace kvadratického funkcionálu

Výsledek :

Po ukončení iterace byl jako optimální vybrán řídicí algoritmus minimalizující zadané kritérium :

$$\nabla u(kT) = 1,343 e(kT) - 1,241 e[(k-1)T]$$

$$u(kT) = u[(k-1)T] + \nabla u(kT)$$

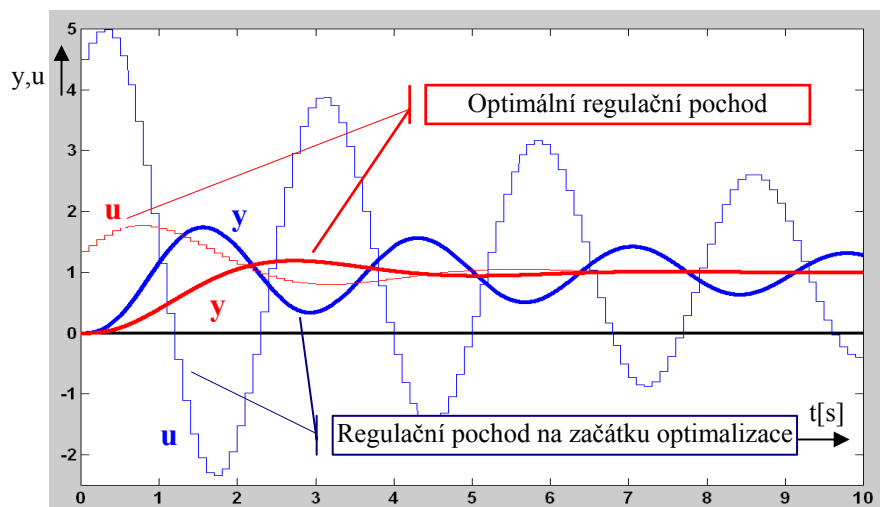
V průběhu optimalizace došlo k vývoji parametrů PS regulátoru a kritéria jakosti :

$$b_0 = 4,5 \rightarrow 1,343$$

$$b_1 = -4,215 \rightarrow -1,241$$

$$J = 5,518 \rightarrow 1,158$$

Odpovídající regulační pochody :



□ Konec příkladu

Literatura :

- [1] Janeček, J., Kupka, L.: *MATLAB & Simulink : řešené příklady*. 1.vyd. Lanškroun: SOŠ a SOU, 2007. ISBN 978-80-239-9532-9.
- [2] Kupka, L.: *MATLAB & Simulink : úvod do použití*. 1.vyd. Lanškroun: SOŠ a SOU, 2007. ISBN 978-80-239-8871-0.
- [3] WEB stránky HUMUSOFT s.r.o., MATLAB, Simulink. [cit.7.2009.] Dostupné na URL: <http://www.humusoft.cz/produkty/matlab/>

Velmi doporučuji jít touto cestou a vyhnout se složité matematice. Tento princip studenti snadno pochopí na rozdíl od složitostí Z-transformace a práce s ní.

Pokud se vyskytnou jakékoliv problémy, neváhejte mne kontaktovat :

josef.janecek@tul.cz,

tel.: 605 53 6879

Autor : doc., Ing. Josef Janeček, CSc., Ústav mechatroniky a technické informatiky, Fakulta mechatroniky, informatiky a mezioborových studií, Technická univerzita v Liberci

