

# APLIKAČNÉ VYUŽITIE MATLAB-DDE KOMUNIKÁCIE S MULTIPROGRAMOVÝM PRÍSTUPOM A VYKRESĽOVANÍM V REÁLNO M ČASE PRE IDENTIFIKÁCIU A RIADENIE LABORATÓRNEHO MODELU

J. Čerkala, A. Jadlovská

Katedra kybernetiky a umelej inteligencie, Fakulta elektrotechniky a informatiky,  
Technická univerzita v Košiciach, Slovenská republika

## Abstrakt

V článku sa budeme venovať popisu funkcionalít programového modulu "*Aplikácia pre riadenie hydraulického modelu*", ktorý je určený pre experimenty na laboratórnom modeli hydraulického systému. Programový modul pozostáva zo skriptov a funkcií programovacieho jazyka Matlab a blokov prostredia Simulink, ktoré umožňujú komunikáciu na báze DDE s programom Excel ako aj s riadiacim PLC laboratórneho modelu a zároveň umožňujú efektívnu správu experimentov a dát meraní. Modul ďalej obsahuje navrhnuté jednoduché grafické rozhranie, v ktorom si používateľ môže zvoliť typ experimentu a dáta, na základe ktorých vykoná identifikáciu parametrov modelu alebo typ algoritmu riadenia na základe metódy syntézy.

## 1 Úvod

Programový modul *Aplikácia pre riadenie hydraulického modelu* (ARHM) bol vytvorený v rámci diplomovej práce [1] s cieľom využiť laboratórny model hydraulického systému pre výuku. Zvolili sme prístup pre jednoduché ovládanie bez nutnosti nastavovania parametrov modelu alebo experimentov. Základným cieľom pri tvorbe programového modulu bolo vytvoriť jednoduché rozhranie, ktoré ponúka veľa možností využitia modelu – od simulácii až po reálne experimenty.

Súčasťou programového modulu ARHM sú demo príklady pre jednoduchú demonštráciu práce s laboratórnym modelom hydraulického systému a zahŕňajú tieto okruhy:

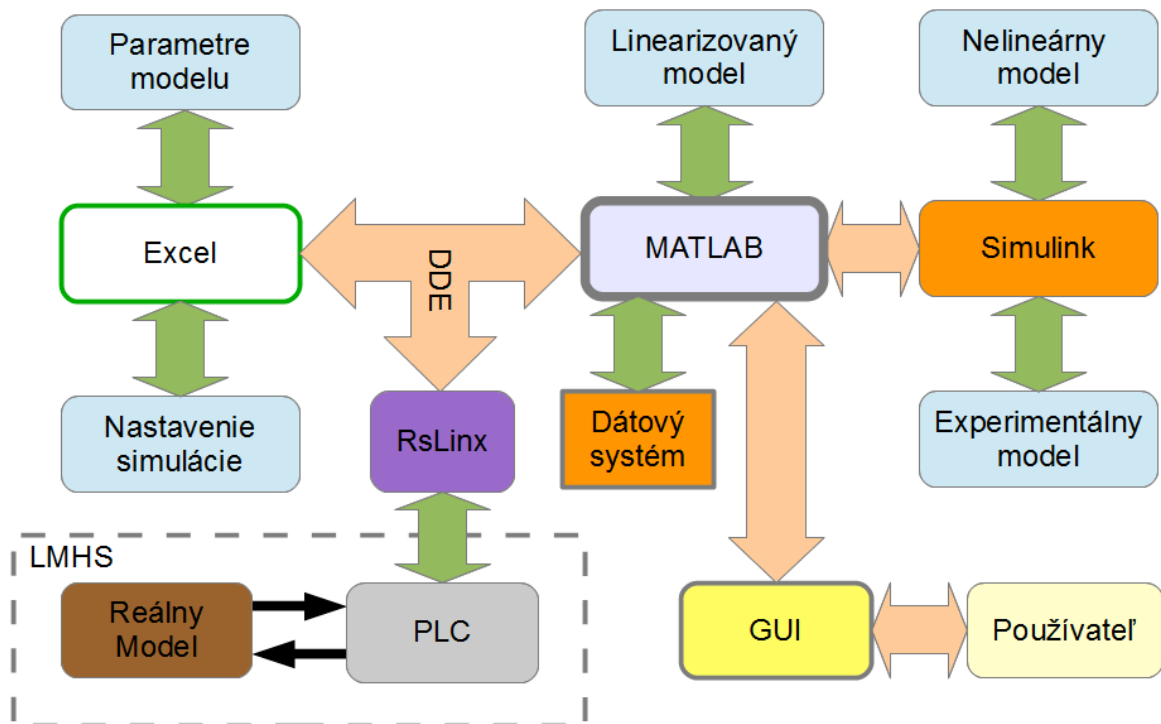
- DDE komunikácia,
- analytická identifikácia – nelineárny model systému,
- experimentálna identifikácia – stochastický regresný model (ARX, ARMAX),
- vstupno-výstupné riadenie – metóda umiestnenia pólov,
- stavové riadenie – princíp minimalizácie kvadratického kritéria

DDE komunikácii pre daný hydraulický model sa venuje publikácia [2], ktorá hlbšie popisuje princíp tohto typu komunikácie [2]. Avšak tento článok si kladie za cieľ prezentovať možnosti DDE komunikácie a jej aplikačné využitie v rámci programového modulu.

## 2 Koncepcia DDE komunikácie

Nakoľko našou motiváciou bolo zjednodušenie ovládania laboratórneho modelu hydraulického systému (LHMS), ktorý je podrobnejšie popísaný v 3. časti, zvolili sme Matlab ako hlavný programovací jazyk, s ktorým používateľ priamo komunikuje, pretože obsahuje grafické rozhranie pre ovládanie (GUI) a umožňuje vykresľovanie efektívnych a dobre popísaných grafov – funkcie 2D a 3D grafiky. Programový modul umožňuje plnú kontrolu nad dátami a rýchlu modifikovateľnosť kódu.

Ďalej v článku podrobnejšie popíšeme väzby, význam a funkciu jednotlivých blokov programového modulu (Obr. 1).



Obr. 1.: Schéma DDE komunikácie programového modulu

### Väzby



Hlavná väzba medzi blokmi – zložitejší prenos dát medzi programami alebo komunikácia pomocou DDE.



Priama dátová výmena, jednoduchá realizácia.



Snímanie hodnôt modelu respektíve budenie akčného člena

### DDE komunikácia, RsLinx a PLC

*DDE komunikačný kanál* riadený funkciami programovacieho jazyka Matlab si vymieňa údaje s programami Excel a RsLinx. Hlavná komunikačná slučka je naprogramovaný v skripte, kde pre každú periódu vzorkovania zosníma hodnoty riadiaceho PLC laboratórneho modelu hydraulického systému a zapíše budenie respektíve vypočítaný akčný zásah.

Blok *RsLinx* je reprezentovaný programom pre spojenie s PLC, ktorý je taktiež založený na prenose dát komunikáciou DDE. V PLC automate je spustený základný riadiaci program, ktorý zabezpečuje chod modelu, škálovane a obmedzovanie hodnôt a teda program zároveň slúži aj ako poistka pre prípad pretečenia nádrží, ak by došlo v riadiacej slučke k chybe alebo by došlo k zmene parametrov LMHS, ktoré by umožnili takéto nežiadúce chovanie. Funkcie pre komunikáciu nášho programového modulu komunikujú práve s týmto programom a je ho možné sledovať v programe *RsLogix 5000*, v ktorom taktiež existuje možnosť vykresľovania aktuálneho stavu výšky hladín alebo budenia.

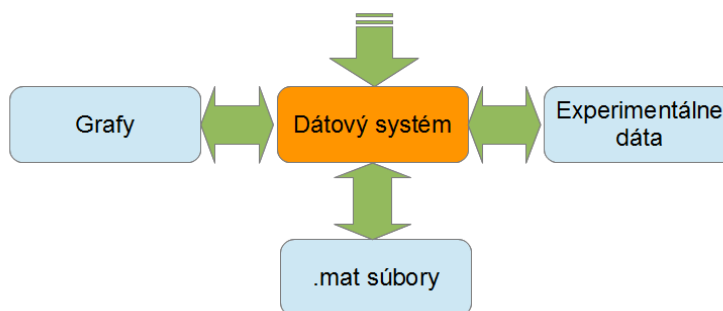
**Blok****MATLAB**

Blok *Matlab* pomocou programovacieho jazyka Matlab spravuje celý programový modul ARHM. Obsahuje jednoduché grafické rozhranie, ktoré umožňuje komunikáciu s používateľom. Práve Matlab vykonáva takmer všetky numerické výpočty a kľúčové úlohy ako napríklad

- výpočet parametrov regresného modelu (ARX, ARMAX) metódou najmenších štvorcov,
- inicializuje a riadi komunikáciu s ostatnými programami,
- sleduje stav LMHS a spúšťa experimenty.

**Blok****Dátový systém**

Blok *Matlab* priamo riadi blok *Dátový systém* a to tak, že po vykonaní experimentu sú grafy vždy ukladané do priečinka daného experimentu a to v priečinku so stručným pomenovaním experimentu a dátumom vykonania experimentu. Výsledky týchto experimentov ukladá vo forme .fig a .emf súborov, tak ako aj .mat súborov dát. V prípade experimentu pre získanie tréningových dát sú tieto dáta automaticky zvolené ako pracovné dáta pre experimenty identifikácie (Obr. 2).

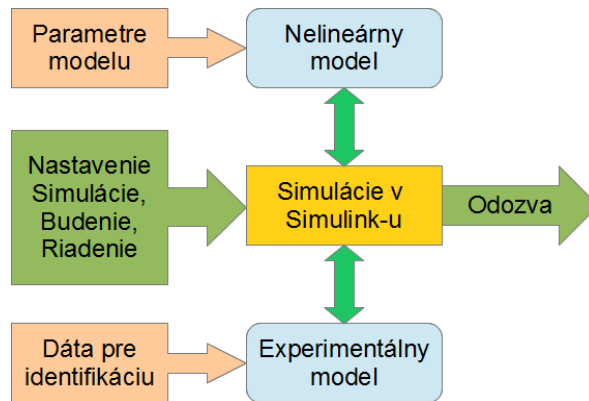


Obr. 2.: Správa dátových zdrojov programového modulu

**Blok****Simulink**

Blok *Simulink* slúži pre simuláciu experimentov s nelineárnym modelom hydraulického systému. Taktiež obsahuje naprogramované moduly pre experimentálnu identifikáciu pozostávajúce zo schém v prostredí Simulink, ktoré sú vyskladané z built-in funkcií a blokov pre experimentálnu identifikáciu. Vstupom tejto časti je nastavenie simulácie, ktoré je v rámci modulu predvolené. Úlohy bloku *Simulink* v programovom module ARHM je možné rozdeliť podľa typu aproximácie LMHS s ktorým sa pracuje (Obr. 3). Ak sa pracuje s nelineárnym simulačným modelom na báze diferenciálnych rovníc, jeho vstupmi sú aj parametre, pre ktoré je tento model navrhovaný a nastavujú sa už v programovom module *Matlab* pomocou masky subsystému.

V prípade experimentálneho modelu, ktorý je získavaný experimentálnou identifikáciou, jeho vstupom je dátový súbor budenia a odozvy, pre ktorý sa metódou najmenších štvorcov vypočítavajú parametre regresného modelu. Výstupné dáta odoziev a modelu sa posielajú priamo do programového modulu *Matlab* spolu s časovým vektorom, kde sa následne analyzujú, upravujú a archivujú.



Obr. 3.: Vázby medzi objektmi v Simulink-u

V rámci bloku *Simulink* sú využívané aj s-funkcie spadajúce pod *System Identification Toolbox*.

## Blok

Excel

Blok *Excel* obsahuje uložené parametre LMHS a nastavenia experimentov. Ak je nutné, používateľ má možnosť vstúpiť do týchto hárkov a nastavovať hodnoty. Veľkou výhodou Excel-u je dynamické prepojenie hodnôt. Ak sa teda zmení perióda vzorkovania, automaticky sa prepočítajú nastavenia simulácii či počty vzoriek pre experiment. Takto sú hárku programu Excel realizované rozličné nastavenia:

- parametre modelu – číselné hodnoty pre ktoré sa odvádza analytický model,
- experiment – nastavenie simulácie, pracovného bodu a spôsobu nábehu do tohto bodu,
- budiaci signál – signál, ktorý bude použitý pre experiment a jeho nastavenia,
- DDE komunikácia – nastavenie adres pre PLC a virtuálny model v Excel-i a zároveň nastavenia predvolených hodnôt,
- syntéza – voľba pólov pre syntézu metódou rozloženia pólov a voľba kvadratického kritéria pre stavové riadenie,
- informácie – rôzne všeobecné informácie o programovom module.

Ďalšiu dôležitú úlohu zohral Excel aj pri vytváraní programového modulu. V samostatnom hárku sa nachádzal virtuálny model pamäte PLC, ktorý slúžil na testovanie DDE komunikácie a testy vykresľovania stavových hodnôt LMHS v reálnom čase (Obr.4). V samostatnom hárku sa nachádzajú adresy polí pre PLC a buniek virtuálneho modelu PLC ako aj inicializačné hodnoty, ktoré sa vždy po spustení komunikácie zapísali.

Virtuálny model pre test DDE komunikácie						
ID	Premenná	Hodnota	Jednotka	Komentár	Typ	
1	H1a	180	m	Aktuálna výška hladiny v prvej nádrži	Čítaj	
2	H2a	70	m	Aktuálna výška hladiny v druhej nádrži	Čítaj	
3	Uina	0	V	Aktuálne budenie motora	Zápis	
4	H1max	280	m	Limit hladiny v prvej nádrži	Zápis	
5	H2max	280	m	Limit hladiny v druhej nádrži	Zápis	
6	wa	0	V	Riadenie	Zápis	
7	topic	Riadenie	-	Topic pre komunikáciu	Zápis	
8	CnBit	0	bit	Kontrolovací bit	Zápis	
9	StBit	0	bit	Štartovací bit	Zápis	

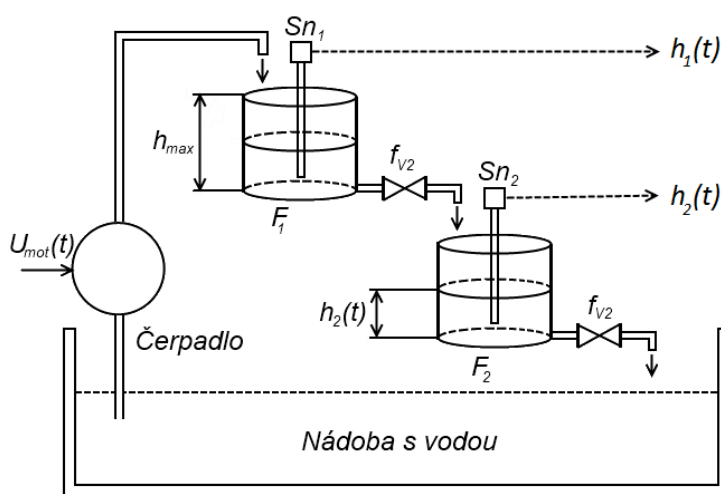
Obr. 4.: Virtuálny model hydraulického systému realizovaný v programe Excel formou tabuľky

K týmto nastaveniam sme pristupovali pomocou príkazu *xlsread*, ktorý umožňuje priame čítanie buniek hárku programu Excel. Pre zmenu týchto hodnôt slúži príkaz *xlswrite*.

Ak si používateľ vytvorí viacero kópií nastavovacieho súboru, môže ich rýchlo meniť – stačí nahradiť súbor. Týmto spôsobom používateľ počas behu skriptu nikdy nenastavuje parametre, len základné voľby ako typ budenia alebo referenčnej trajektórie.

### 3 Laboratórny model hydraulického systému

Na *Katedre kybernetiky a umelej inteligencie Fakulty elektrotechniky a informatiky* sa v *Laboratóriu Mechatronických Systémov* (<http://kyb.fei.tuke.sk/laboratoria/miest/V142.php>) nachádza laboratórny model hydraulického systému (<http://kyb.fei.tuke.sk/laboratoria/modely/hyd.php>), ktorý vyrobila firma *Kybernetika s. r. o* – (Obr. 5). Model pozostáva z dvoch valcových nádrží, ktoré sú zapojené kaskádne – výtok prvej je prítokom druhej. Voda druhej nádrže vyteká do vane na spodnej strane modelu, kde ju lopatkové čerpadlo pumpuje znova do prvej nádrže. Lopatkové čerpadlo je poháňané asynchrónnym motorom, ktorý je riadený cez frekvenčný menič pomocou PLC automatu.



Obr. 5.: Laboratórny model hydraulického systému - náčrt

Snímanie výšky hladín zabezpečujú kapacitné snímače, ktoré boli podrobnejšie popísané v článku [3].

Na základe meraní a experimentov sme získali parametre modelu a hlavne konštantu čerpadla. Programový modul ARHM pristupuje priamo do Excel-u počas inicializácie jeho hlavného skriptu. Tento prístup nám zároveň umožňuje vytvoriť viac nastavení modelov, ktoré je možné rýchlo meniť, čo je výhodné najmä v prípade zmeny výtokových ventilov LMHS.

### 4 Identifikácia modelu

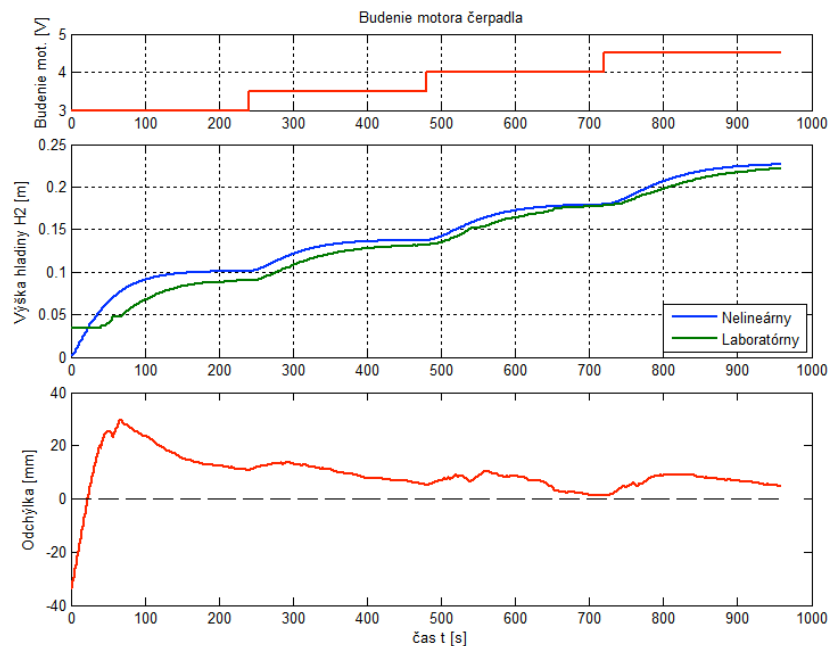
Pri tvorbe programového modulu ARHM sme využili prístup matematicko-fyzikálneho modelovania (metóda analytickej identifikácie) a metódy a algoritmy pre experimentálnu identifikáciu s využitím stochastického regresného modelu (ARX, ARMAX).

#### 4.1 Analytická identifikácia

Pomocou rovníc materiálovej bilancie a fyzikálnych zákonitostí sme odvodili nelineárny model hydraulického systému, ktorý sme následne naprogramovali v prostredí Simulink.

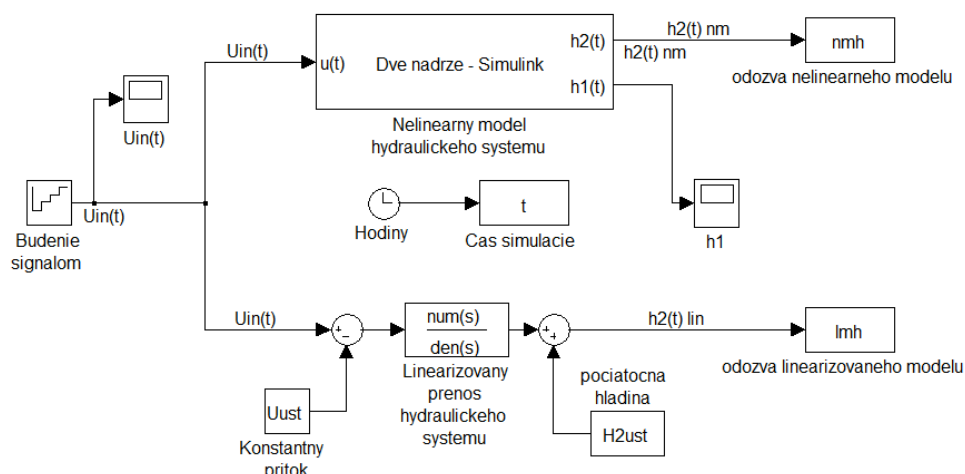
$$\begin{aligned} \frac{dh_1(t)}{dt} &= \frac{u_{in}(t) \cdot K_{\text{čer}}}{F} - \frac{K_{11}}{F} \cdot \sqrt{h_1(t)}, \\ \frac{dh_2(t)}{dt} &= \frac{K_{11}}{F} \cdot \sqrt{h_1(t)} - \frac{K_{22}}{F} \cdot \sqrt{h_2(t)}. \end{aligned} \quad (1)$$

Po dosadení rozmerov a experimentálne získaných parametrov sme model podrobili experimentu vzájomného porovnania s reálnym modelom hydraulického systému (Obr.6). Experiment pozostával z porovnávania nameraných dát pre zvolené rovnaké budenie na LMHS so simulačne získanými dátami zo simulácie na nelineárnom modeli. Takto sme dokázali, že použité parametre v analytickom modeli sú správne.



Obr. 6.: Porovnanie nelineárneho a laboratórneho modelu odozvou na skokovú zmenu budenia motora čerpadla – 2. nádrž

Programový modul ARHM umožňuje po vhodnej voľbe pracovného bodu výpočet parametrov linearizovaného modelu a použitie vypočítaných hodnôt takto získaného linearizovaného modelu pre syntézu riadenia a taktiež ponúka možnosť tento linearizovaný model podrobiť vzájomnému porovnaniu s analytickým nelineárnym modelom naprogramovaným v prostredí Simulink (Obr. 7.). Výsledky opäť potvrdili, že modely navzájom korešpondujú.



Obr. 7.: Schéma v prostredí Simulink pre porovnanie nelineárneho analytického modelu s jeho linearizovanou verzou

## 4.2 Experimentálna identifikácia

V rámci experimentálnej identifikácie sme využili funkcie *System Identification toolbox*, ktorého aplikačné využitie je presnejšie opísané v článkoch [4],[5] a systémovom manuáli [6].

Využili sme štruktúry stochastických regresných modelov ARX a ARMAX, ich parametre boli počítané metódou najmenších štvorcov (MNS). V programovom modeli využívame built-in funkcie *System Identification toolbox-u* v simulačnom jazyku Matlab, naprogramovanú funkciu MNS a bloky v prostredí Simulink.

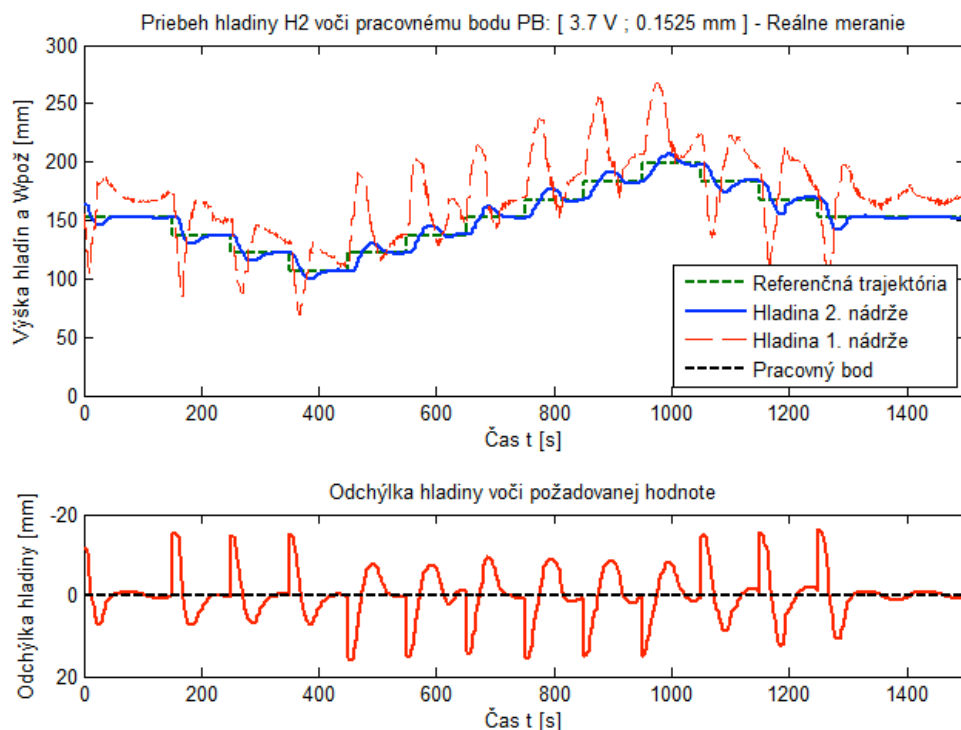
Programový modul ARHM obsahuje aj voľby pre získavanie trénovacích dát pre tento typ identifikácie a taktiež je v ňom implementované aj automatické predspracovanie dát. Používateľ má možnosť voľby medzi preddefinovanými modelmi aj pre syntézu riadenia.

## 5 Vykresľovanie v grafoch v reálnom čase pre Matlab

Programový modul ARHM od začiatku predpokladal vykresľovanie hodnôt v reálnom čase experimentu. V prostredí Simulink to je samozrejmosť, avšak pre ARHM mali byť grafy realizované iba funkciami 2D grafiky programovacieho jazyka Matlab. Našou motiváciou bolo mať plnú kontrolu nad grafom, pričom graf sa mal dynamicky nastavovať podľa daného experimentu a jeho prednastavení. Umožňuje to funkcia *set*.

Vzhľadom na to, že LMHS hydraulický systém má veľmi pomalú dynamiku, mohli sme si dovoliť aj menej presné vzorkovanie, ako ponúka simulácia v prostredí Simulink. V riadiacej slučke sme použili funkcie *tic* a *toc*. Funkcia *tic* spustí interné počítadlo, ktoré počíta uplynutý čas a funkcia *toc* zasa vracia dobu, ktorá uplynula od posledného zadania *tic*. Týmto spôsobom sme mohli určiť čas jednej periódy, v našom prípade 1 sekundy, pričom na začiatku periódy sme začali stopovať, vykonali sme DDE výmenu údajov s PLC laboratórneho modelu, prípadne výpočet akčného zásahu, aktualizovali sme grafy, overili, či nastali chyby a zvyšný čas sme čakali do začiatku ďalšej periódy.

Zároveň sme sledovali aj výpočtový čas – ak došlo k presiahnutiu času pre jednu periódu, do príkazového riadku bol vypísaný čas a posun. K posunu mohlo dôjsť v prípade zvolenia malej periódy vzorkovania, ktorá nestačila na aktualizáciu grafu, čo je pomerne výpočtovo náročné. Taktiež ho mohol spôsobiť počítač, ktorý by v čase simulácie vykonával iné výpočtové úlohy.



Obr. 8.: Graf, ktorý bol vykreslený priebežne počas experimentu riadenia sledovania referenčnej trajektórie pre metódu syntéz umiestnenia pólov

Toto vykresľovanie je použité napríklad v experimentoch testovania algoritmu riadenia, kde sa sleduje zvolená referenčná trajektória (Obr. 8). Taktiež je možné v grafe vykresliť predpokladané chovanie sa systému, ktoré sme získali simuláciou. Po spustení je zobrazený graf v celej šírke

simulácie, pričom sa zobrazuje aj “waitbar” – graf priebehu experimentu, kde je možné experiment počas behu zastaviť. Ak došlo k prerušeniu buď z dôvodu chyby alebo voľbou používateľa, zobrazovaný graf sa uložil s už zaznamenaným priebehom.

## 6 Záver

V článku sme predstavili programový modul “Aplikácia pre riadenie hydraulického modelu”, ktorý je určený ako výukový modul pre študentov programu *Kybernetika a informačno-riadiace systémy*, na *Katedre kybernetiky a umelej inteligencie* pre ilustráciu a testovanie metód a algoritmov pre identifikáciu modelu fyzikálneho systému pričom zároveň umožňuje implementovať získané výsledky (parametre) modelu vo vhodných algoritmoch riadenia. Programový modul ARHM je vhodný pre generovanie dátových súborov, ktoré môžu vo výukovom procese poslúžiť ako pracovné dáta pre experimentálnu identifikáciu v rámci predmetov *Optimálne a nelineárne systémy* a *Riadenie a umelá inteligencia* na druhom stupni štúdia.

Taktiež je tento programový modul demonštráciou možností simulačného jazyka Matlab v rôznych smeroch – od DDE komunikácie, blokových schém v prostredí Simulink po výpočty riadiacich zásahov a grafy.

## PodĎakovanie

Tento článok bol podporený Vedeckou grantovou agentúrou Slovenskej Republiky v projekte Vega č..1/0286/11 Dynamické hybridné architektúry v multiagentových sieťových riadiacich systémoch (70%) a projektu Kega č. 021TUKE-4/2012 pod názvom CyberLabTrainSystem - demonštrátor a trenážer informačno-riadiaceho systému (30%).

## Použitá literatúra

- [1] ČERKALA, J.: Aplikácia výsledkov experimentálnej identifikácie v návrhu algoritmov riadenia reálneho modelu hydraulického systému. Diplomová práca (vedúca práce: doc. Ing. Anna Jadlovská, PhD), Košice: Fakulta elektrotechniky a informatiky, Technická univerzita v Košiciach, 2012. 83 s.
- [2] JAJČIŠIN, Š.: Verification of Control Algorithms with DDE Communication on Real Hydraulic System. In: SCYR 2011 : 11th Scientific Conference of Young Researchers of Faculty of Electrical Engineering and Informatics Technical University of Košice: proc. - Košice : FEI TU, 2011 pp. 388-391. - ISBN 978-80-553-0644-5.
- [3] JAJČIŠIN, Š. – JADLOVSKÁ, A.: Riadenie laboratórneho modelu hydraulického systému. In: Electroscopie – online časopis pro elektrotechniku, ročník 2011, číslo III. ISSN 1802-4564, Dostupné na internete: [http://147.228.94.30/index.php?option=com\\_content&view=article&id=280:riadenie-laboratorneho-modelu-hydraulickeho-systemu-&catid=34:cislo-32011-&Itemid=48](http://147.228.94.30/index.php?option=com_content&view=article&id=280:riadenie-laboratorneho-modelu-hydraulickeho-systemu-&catid=34:cislo-32011-&Itemid=48) (2011).
- [4] ČERKALA, J. - JADLOVSKÁ, A.: Experimentálna identifikácia nelineárneho dynamického systému pomocou IDENT Tool v prostredí Matlab. In: Electrical Engineering and Informatics 3 : Proceeding of the Faculty of Electrical Engineering and Informatics of the Technical University of Košice. - Košice : FEI TU, 2012 s. 46-51. - ISBN 978-80-553-0890-6.
- [5] ČERKALA, J. – JADLOVSKÁ A.: Aplikácia System Identification Toolboxu v experimentálnej identifikácii lineárnych dynamických systémov. Electrical Engineering and Informatics: Proceeding of the Faculty of Electrical Engineering and Informatics of the Technical University of Kosice. 2010, s. 540-545. ISSN 978-80-553-0460-1.
- [6] LJUNG, L.: System Identification Toolbox, Používateľská príručka, 6. verzia

---

Ing. Jakub Čerkala  
jakub.cerkala@tuke.sk

Doc. Ing. Anna Jadlovská, Phd.  
anna.jadlovská@tuke.sk