

Tutoriál pre inteligentné adaptívne riadenie a Neural Network Toolbox

Cieľom tutoriálu pre inteligentné adaptívne riadenie bude:

1. *Inteligentné adaptívne riadenie.*
2. *Využitie príkazov Neural Network Toolboxu.*
3. *Neurónový model simulačného modelu zvoleného systému.*
4. *Implementácia dopredného neurónového modelu v adaptívnom riadení fyzikálnych systémov s časovo premenlivými parametrami.*

Úlohy:

1. **Zostavte matematicko-fyzikálny model nelineárneho dynamického systému M6 s využitím analytickej identifikácie**
2. **Natrénujte neurónový model nelineárneho dynamického systému M6.**
 - 2.a **Navrhňte štruktúru neurónového modelu**
 - 2.b **Postup identifikácie nelineárneho systému pomocou experimentu**
 - 2.c **Získanie tréningových a testovacích dát pre vytvorenie neurónového modelu**
 - 2.d **Využitím príkazov Neural Network Toolboxu vytvorte neurónový model systému**
3. **Implementujte neurónový model nelineárneho dynamického systému v adaptívnom riadení simulačného modelu fyzikálneho systému s časovo premenlivými parametrami**

Úloha č. 1 Zostavte matematicko-fyzikálny model nelineárneho dynamického systému M6 s využitím analytickej identifikácie

Odvodenie matematického popisu modelu M6 s využitím postupu analytickej identifikácie sa nachádza na stránke predmetu *Optimálne a nelineárne systémy* v časti *Simulačné modely*.

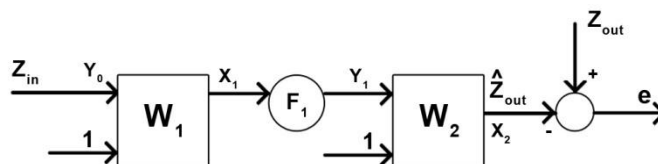
http://matlab.fei.tuke.sk/ons/pdfModely/M6_dveNadobyInter.pdf

Úloha č.2 Natrénujte neurónový model nelineárneho dynamického systému M6

Pre netrénovanie neurónového modelu dynamického systému je potrebné sa oboznámiť s teóriou neurónových sietí a následne s možnosťami trénovania, ktoré nám poskytuje Neural Network toolbox.

V teórii automatického riadenia sú v súčasnosti veľmi dobre spracované syntézy návrhu regulátorov pre lineárne modely. Väčšina metód návrhu klasických regulátorov vychádzajúcich z nelineárnych modelov vyžaduje úplnú znalosť riadeného systému čo je často nemožné. Jedným z možných riešení je použitie neurónového modelu systému, ktorý môže vďaka univerzálnym aproximačným schopnostiam zastávať funkciu modelu, regulátora alebo inteligentného člena v regulačnom obvode.

Viacvrstvovú perceptrónovú sieť s jednou skrytou vrstvou je možné vyjadriť v maticovej rotácii zobrazenej na nasledujúcom obrázku



Obr. 1 Viacvrstvová perceptrónová sieť s jednou skrytou vrstvou v maticovej rotácii

kde W_1, W_2 sú váhové matice, ktorých každý riadok reprezentuje váhy príslušného neurónu v danej vrstve a posledný stĺpec obsahuje hodnoty prahových koeficientov a X_i, Y_i sú vektory v i -tej vrstve. Vektor F_I predstavuje výstupnú (aktivačnú) funkciu neurónov. Počet neurónov v každej vrstve je daný počtom prvkov vstupného Z_{in} a výstupného vektora Z_{out} . Málo zvolených neurónov neumožní neurónovej sieti aproximovať všetky nelinearity zahrnuté v procese naopak príliš veľa neurónov zvyšuje riziko pretrénovania siete. Pri návrhu regulátora je potrebné poznať koeficienty prenosovej funkcie systému a tieto koeficienty sa dajú získať z matice zosilnení N ktorá sa vypočíta nasledovne:

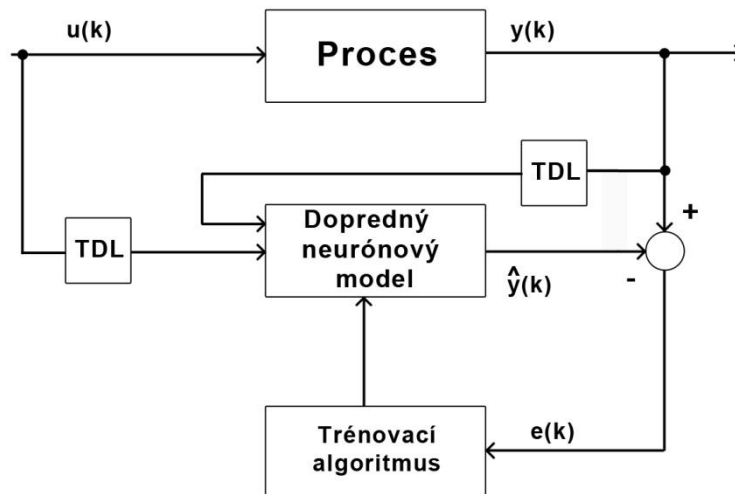
$$N = \frac{d\hat{z}_{out}}{dz_{in}^T} = \frac{dX_2}{dY_0^T} = \frac{dX_2}{dY_1^T} \cdot \frac{dY_1}{dX_1^T} \cdot \frac{dX_1}{dY_0^T} = W_2 F_1'(X_1) W_1 \quad (1)$$

Úloha č. 2.a Navrhnite štruktúru neurónového modelu

Poznáme dve štruktúry neurónových modelov:

1. dopredné – používajú sa pri identifikácii nelineárnych systémov
2. inverzné – používajú sa ako spätnoväzoný regulátor

Keďže dopredné modely sú vhodné pre identifikáciu nelineárnych procesov nižšie bude uvedená identifikačná štruktúra použitá pri identifikácii nelineárneho modelu hydraulického systému. Pri použití dopredných neurónových modeli je neurónový model umiestnený paralelne s identifikovaným procesom. Signál použitý na tréning neurónového modelu sa nazýva predikčná chyba, ktorá predstavuje rozdiel medzi výstupom systému a výstupom neurónového modelu. Identifikačná štruktúra použitá pri identifikácii hydraulického systému sa nazýva štruktúra so sériovo paralelným modelom zobrazená na nasledujúcom obrázku.



Obr. 2 Identifikačná štruktúra so sériovo-paralelným modelom

Bloky TDL sú oneskorovacie bloky vzoriek vstupov a výstupov procesu. Výhodou tejto štruktúry je zaručená konvergencia a stabilita modelu.

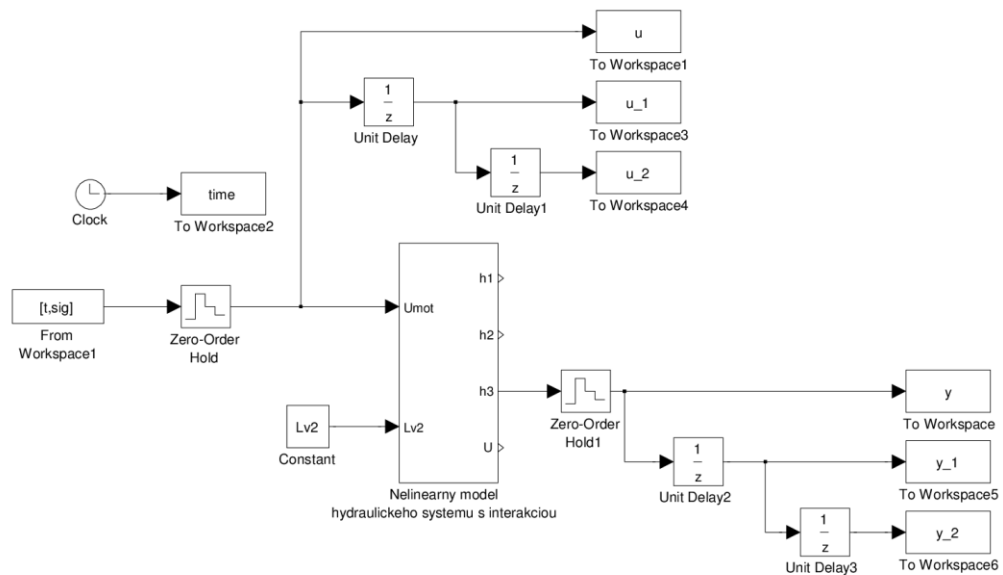
Úloha č. 2.b Postup identifikácie nelineárneho systému pomocou experimentu

Algoritmus pre identifikáciu nelineárneho modelu systému pozostáva z nasledujúcich krokov:

1. Inicializácia simulácie
2. Vykonalie experimentu pozostávajúceho zo zisku vstupných a výstupných tréningových dát nelineárneho modelu.
3. Výber štruktúry neurónového modelu – výber identifikačnej štruktúry nelineárneho modelu systému (štruktúra s paralelným alebo sériovo-paralelným modelom), nastavenie počtu neurónov skrytej vrstvy, tréningovej funkcie skrytej vrstvy.
4. Vytvorenie neurónového modelu – možnosťami ktoré poskytuje neural network toolbox natréňovať neurónový model systému.
5. Validácia neurónového modelu – porovnanie odozvy neurónového modelu systému a nelineárneho systému ak je neurónový model validný tak ukončiť algoritmus identifikácie a model je možné použiť v riadení nelineárneho systému. Ak model nie je validný môže to byť spôsobené zlou voľbou tréningových dát v tom prípade sa vrátíme v algoritme na krok 2. Chybné natréňovanie môže byť spôsobené aj zlým výberom štruktúry modelu v takom prípade v algoritme sa vrátíme na krok 3 a upravíme zvolenú štruktúru modelu.

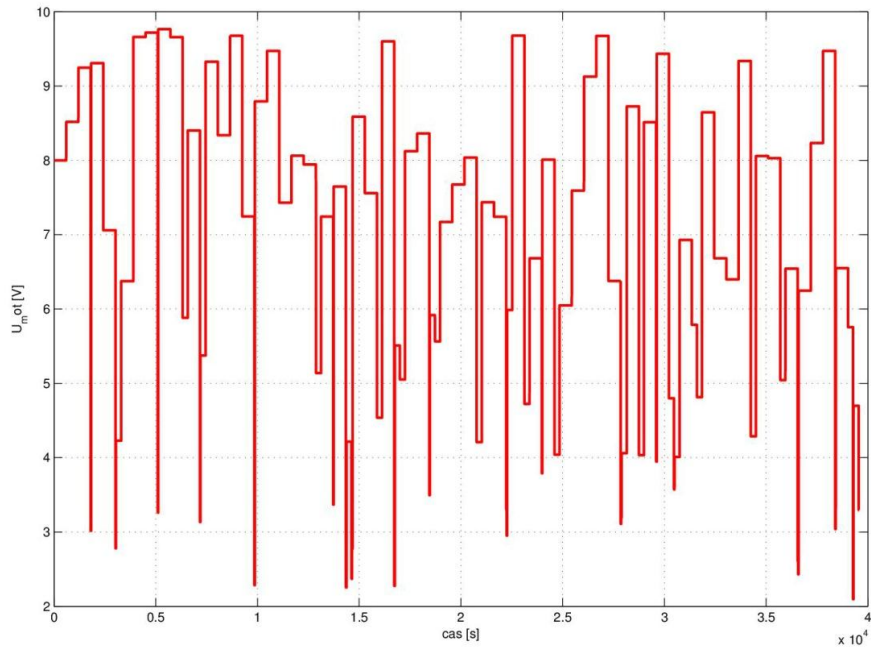
Úloha č. 2.c Získanie tréningových a testovacích dát pre vytvorenie neurónového modelu

Pred začiatkom tréningu neurónového modelu je potrebné pripraviť si tréningové dáta neurónového modelu. Nakoľko neurónová sieť nemá vlastnú dynamiku, tak sa na jej vstup privádzajú oneskorené vzorky vstupu a výstupu systému. Simulinkovská schéma pre získanie tréningových dát je na nasledujúcom obrázku.



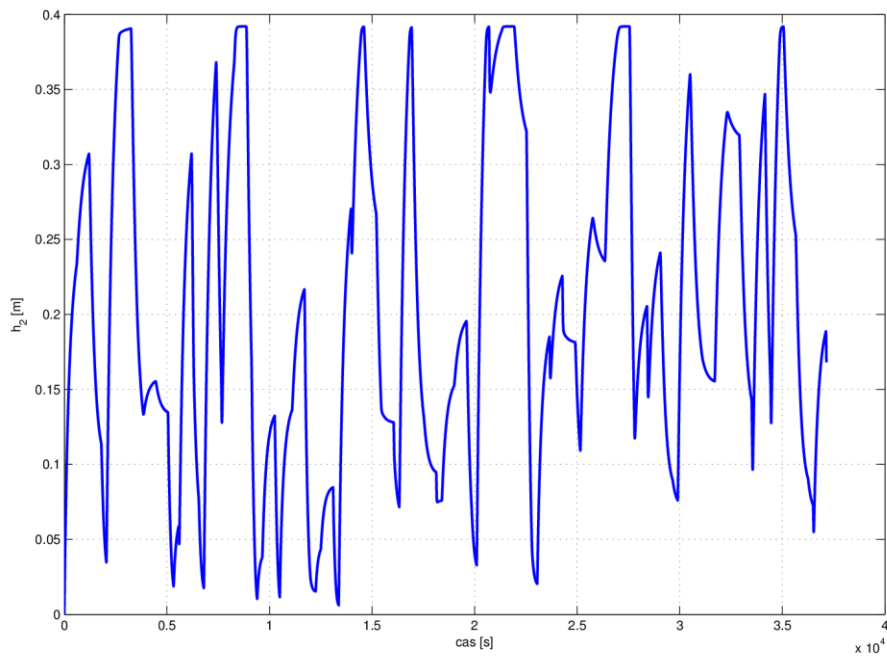
Obr. 3 Simulačná schéma pre získanie tréningových dát

Vstupným tréningovým signálom bol pseudonáhodný binárny signál zobrazený na nasledujúcom obrázku.



Obr. 4 Vstupný tréningový signál

Odozva systému na daný vstupný signál je zobrazená na nasledujúcom obrázku



Obr. 5 Odozva systému na pseudonáhodný binárny signál

Oba signály bolo nutné normovať pre potreby tréningovania neurónového modelu systému. Normované signály boli použité na tréningovanie neurónového modelu hydraulického systému.

Úloha č. 2.d Využitím príkazov neural network toolboxu vytvorte neurónový model

Pre natréňovanie neurónového modelu systému je nutné sa oboznámiť s neural network toolboxom a s príkazmi ktoré daný toolbox ponúka.

Neural network toolbox (NNT) je nástroj Matlabu, ktorým je možné vytvárať štruktúry neurónových sietí pre potreby aproximácie, riadenia, predikcie, atď. Na prácu s NNT máme k dispozícii príkazový riadok, grafické rozhranie aj prostredie Simulink.

Dôležitým krokom pri návrhu typu neurónového modelu je výber vhodnej architektúry.

Vlastnosti modelu, ktoré sa určujú sú nasledovné:

- počet vrstiev neurónového modelu
- počet neurónov v každej vrstve
- funkcie jednotlivých neurónov (typ aktivačnej funkcie)

Príkaz pre vytvorenie doprednej neurónovej siete v prostredí Matlab je nasledovný:

```
siet = newff(P,T,n,{ 'tansig' 'purelin'}, 'trainlm')
```

siet – objekt v prostredí Matlab reprezentujúci štruktúru neurónovej siete

P – signál vstupov

T – signál výstupov

n – počet neurónov skrytej vrstvy

tansig – nelineárna aktivačná funkcia skrytej vrstvy

purelin – lineárna aktivačná funkcia výstupnej vrstvy

trainlm – metóda tréňovania neurónovej siete (Levenberg-Marquardt)

Pred použitím príkazu *newff* je potrebné normovať získané vstupné a výstupné signály z nelineárneho modelu systému. Normovať znamená nájsť maximálnu hodnotu daného vektora a celý vektor vydeliť danou hodnotou. Tým získame signál v rozmedzí 0 až 1. Vhodným typom signálu pre tréňovanie je pseudonáhodný binárny signál.

Signály vstupov a výstupov majú tvar:

$$P = [u_{1n'}, u_{2n'}, y_{1n'}, y_{2n'}]$$

$$T = [y_n']$$

Po vytvorení štruktúry neurónového modelu musíme nastaviť ďalšie parametre ku ktorým prístupujeme cez takzvanú bodkovú konštrukciu:

siet.performFcn = 'sse' – suma štvorcov odchýlok

siet.trainParam.goal = 0.000001 – ukončovacia podmienka na chybu SSE

set.trainParam.show = 20 – frekvencia zobrazovania priebehu chyby

siet.trainParam.epochs = 1000 – maximálny počet tréningových cyklov (epoch)

siet.trainParam.mc = 0.95 – momentum konštanta

Po nastavení parametrov tréningu môžeme pristúpiť k samotnému tréningu neurónovej siete. Pomocou tréningového algoritmu sa modifikujú parametre neurónovej siete tak, aby sa dosiahla čo najlepšia aproximácia medzi skutočným výstupom siete a žiadaným výstupom siete. Toto učenie je učenie off-line, pretože sa vykonáva bez toho aby bola neurónová sieť v reálnom prostredí.

Tréning neurónovej siete vykonáme nasledujúcim príkazom:

$$[nsiet, tr, Y, E, Pf, Af] = \text{train}(siet, P, T)$$

siet – nenatréňovaná neurónová sieť

P – signál vstupov

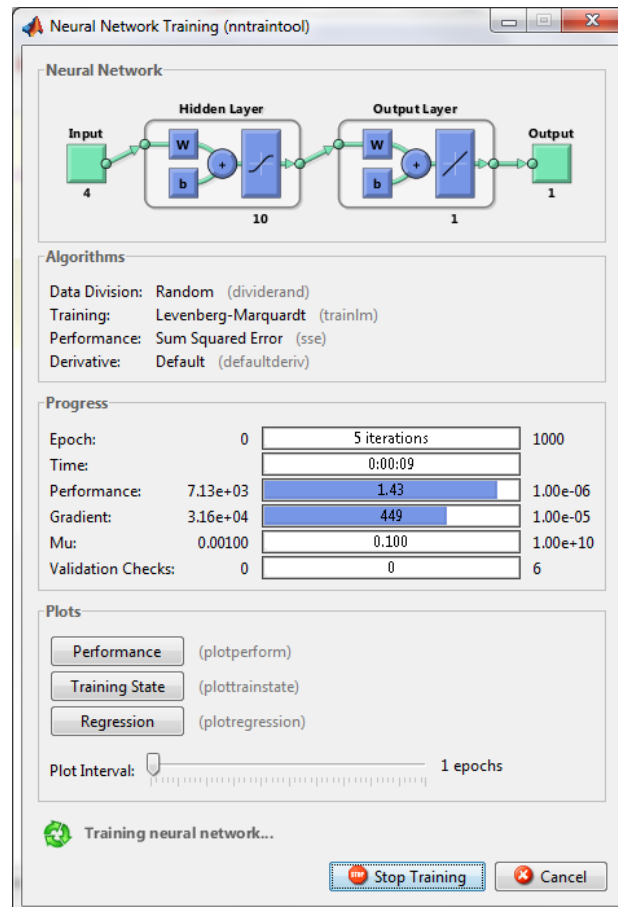
T – signál výstupov

nsiet – natréňovaná neurónová sieť

Y – výstup z natréňovanej neurónovej siete pri vstupe *P*

E – chyba siete

Po spustení príkazu *train* sa zobrazí grafické rozhranie pre tréovanie neurónového modelu kde môžeme vidieť nastavené parametre a priebeh tréovania neurónového modelu systému. Grafické rozhranie je zobrazené na nasledujúcom obrázku.



Obr. 6 Grafické rozhranie tréovania neurónového modelu

Na Obr. 6 vidno, že pri tréovaní bolo zvolených 10 neurónov na skrytej vrstve, počet cyklov bolo nastavený na hodnotu 1000 a ukončovacia podmienka na chybu SSE nastavená na 0,000001. Ostatné parametre sú nastavené rovnako ako bolo vyššie spomenuté. Ukončenie tréovania môže nastať v troch prípadoch a to:

1. vykonaním všetkých cyklov tréovania
2. ukončovacia podmienka na chybu SSE dosiahne nastavenú hodnotu
3. hodnota validation check dosiahne nastavenú hodnotu

Ak tréovanie je ukončené z dôvodu vykonania všetkých cyklov môže to znamenať že neurónový model nebude presne aproximovať nelinearity modelu. Ak nastane ukončenie tréovania, kvôli tomu že validation check dosiahne nastavenú hodnotu, tiež to môže znamenať, že neurónový model nebudevalidný. Najvhodnejším ukončením tréovania neurónového modelu je ak ukončovacia podmienka chyby SSE dosiahne nastavenú hodnotu.

Pre získanie odozvy na vstupný signál slúži nasledujúci príkaz:

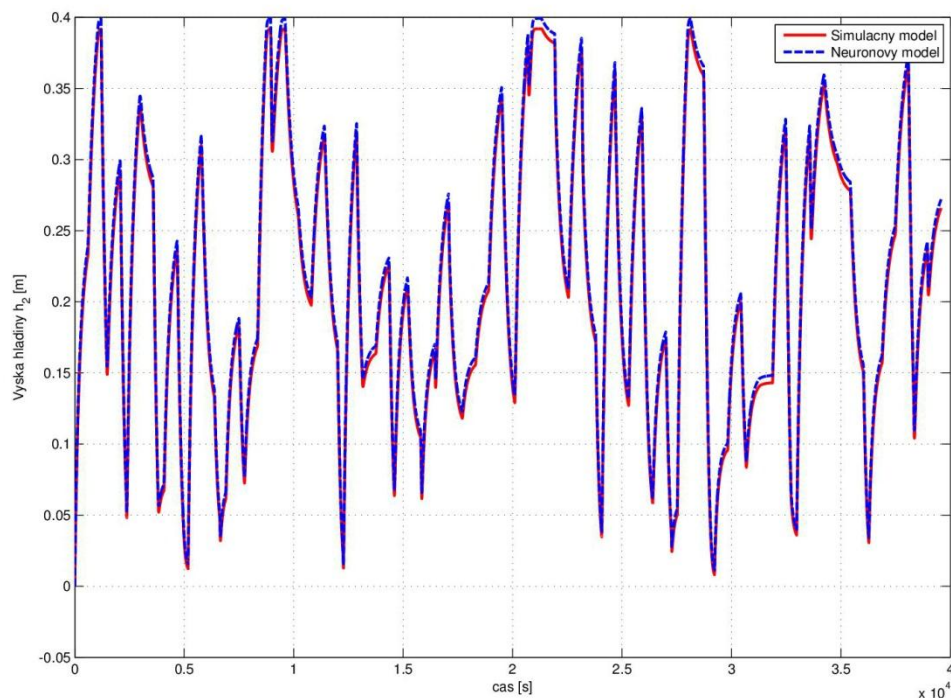
```
[Y2]=sim(nsiet,P2)
```

P2 – signál vstupov

Y2 – signál výstupov natrénovaného neurónového modelu

nsiet – natrénovaná neurónová sieť

Po získaní odozvy neurónového modelu na pseudonáhodný signál môžeme porovnať jeho odozvu s odozvou simulačného modelu daného systému. Porovnanie odoziev je na nasledujúcom obrázku:

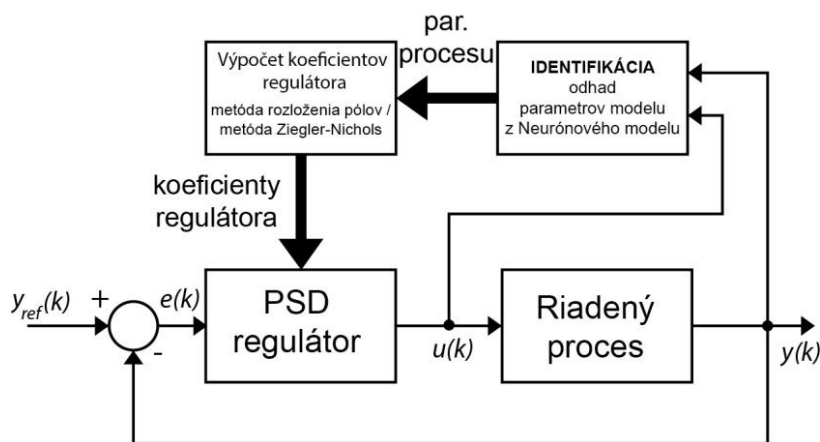


Obr. 7 Odozva simulačného a neurónového modelu na pseudonáhodný binárny signál

Odozva neurónového modelu má takmer totožný priebeh ako odozva simulačného modelu z čoho vyplýva, že neurónová sieť natrénovaná vyššie spomenutými príkazmi je vhodná pre použitie v riadení.

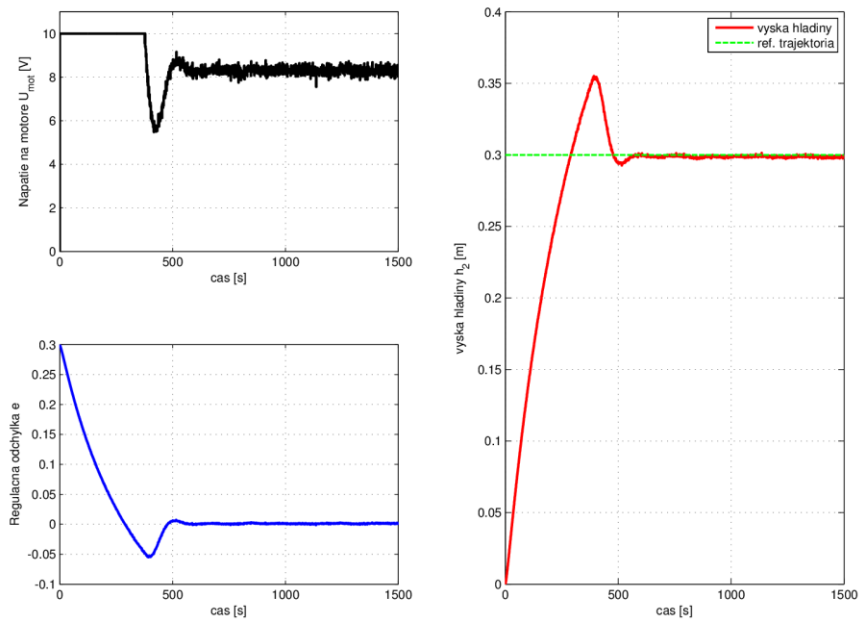
Úloha č. 3 Implementujte neurónový model nelineárneho dynamického systému v adaptívnom riadení simulačného modelu fyzikálneho systému s časovo premenlivými parametrami

Riadiaca štruktúra použitá pri implementácii neurónového modelu dynamického systému je zobrazená na nasledujúcom obrázku:



Obr. 8 Riadiaca štruktúra adaptívneho riadenia s odhadom parametrov z neurónového modelu

Neurónový model vytvorený v prechádzajúcich úlohách metódou rozloženia pólov popísanou v tutoriály pre klasické adaptívne riadenie. Neurónový model bol použitý pre odhad koeficientov prenosovej funkcie hydraulického systému. Potrebné je vypočítať maticu zosilnenia N podľa vzorca (1). Výsledky adaptívneho riadenia nelineárneho systému s odhadom parametrov z neurónového modelu sú zobrazené na nasledujúcom obrázku



Obr. 9 Inteligentné adaptívne riadenie nelineárneho modelu hydraulického systému

Adaptívne riadenie je vhodné použiť na nelineárny model s časovo premenlivými parametrami a tak bol k výstupnému signálu nelineárneho modelu pridaný vysokofrekvenčný šum. Neurónový model bol validný a tak ho bolo možné použiť v adaptívnom riadení. Použitím metódy rozloženia pólov nastal prekmit regulovanej veličiny nad požadovanú hodnotu. Kmitavý priebeh akčného zásahu je spôsobený pridaním vysokofrekvenčného šumu na výstupnom signále.