

Umělé neuronové sítě jako prostředek pro modelování nelineárních soustav

Ivan Taufer¹ a Oldřich Drábek¹

The Artificial Neural Network as means for modeling Nonlinear Systems

The paper deals with nonlinear system identification based on neural network. The topic of this publication is simulation of training and testing a neural network. A contribution is assigned to technologists which are good at the classical identification problems but their knowledges about identification based on neural network are only on the stage of theoretical bases.

Key words: Nonlinear system, Identification, Neural Network, Simulation.

Úvod

Umělé neuronové sítě NS jsou příslibem pro oblast identifikace a řízení nelineárních systémů. Např. neuronová síť může být použita v případě linearizace regulačního ventilu s použitím metody inverzního modelování, při inverzním řízení dynamického nelineárního systému, atd. Neuronové sítě jsou systémy, pracující obecně s libovolným počtem vstupů a výstupů a jsou tedy snadno použitelné pro identifikaci vícerozměrových systémů. Tento příspěvek je zaměřen na použití neuronové sítě k identifikaci nelineárního systému. K této problematice existuje řada odkazů, např. (Sjöberg et al., 1994; Hunt et al., 1992; Pach et al., 1995; Warwirck, 1996; Yamada & Yabuta, 1993).

Představy o výsledcích, dosažitelných v oblasti identifikace systémů prostřednictvím NS, mohou být pro začátečníky, obeznámené pouze se základy teorie NS, dost mlhavé. Proto jsme se rozhodli podělit se o některé zkušenosti z trénování a testování dvouvrstvé NS, která je modelem jednoduché nelineární soustavy. Soustava je tvořena sériovým spojením dynamického členu, statického členu s nelineárním průběhem statické charakteristiky a bloku nelinearity typu "nasycení". Síť je trénována algoritmem učení se zpětným šířením (Backpropagation - BP), viz např. (Sincák & Andrejková, 1996; Kvasnička et al., 1997; Fausset, 1994; Hertz, 1991; Bíla, 1995; Šnorek & Jiřina, 1996).

Stručný popis identifikace nelineárního systému

Uvažujme nelineární dynamický SISO systém, popsany nelineární diferenční rovnicí (Hunt et al., 1992)

$$y^s(k+1) = f[y^s(k), \dots, y^s(k-n_y+1), u(k), \dots, u(k-n_u+1)] \quad , \quad (2.1)$$

ve které pořadnice y^s na výstupu soustavy v časovém okamžiku $(k+1)$ závisí na minulých n_y pořadnicích výstupního signálu a na n_u pořadnicích vstupního signálu; $f[\cdot]$ reprezentuje nelineární závislost mezi vstupem a výstupem soustavy.

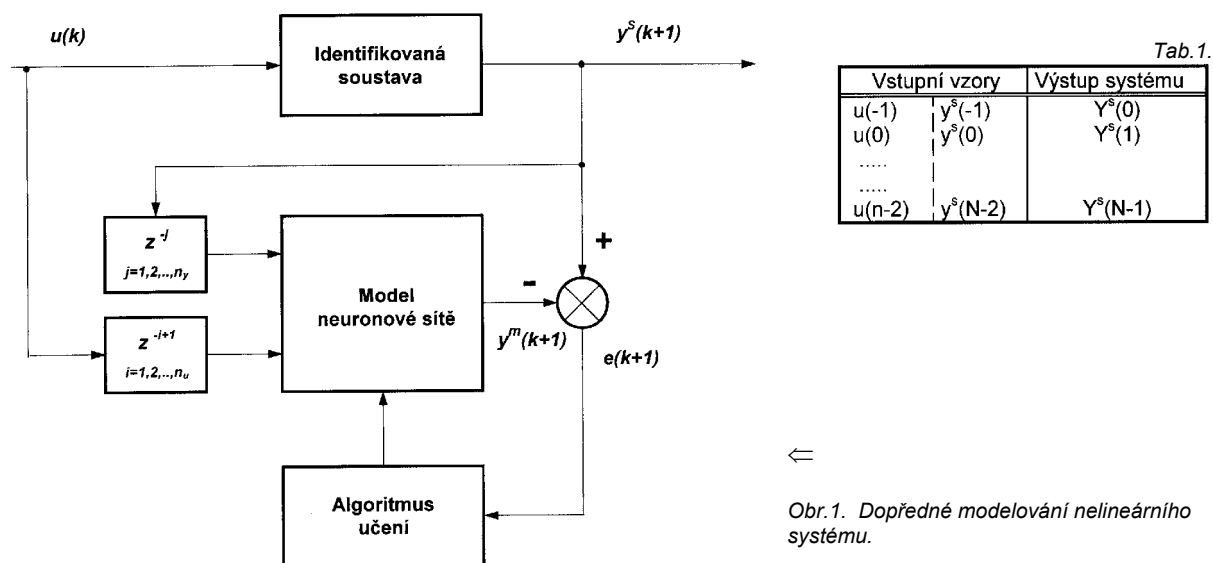
Samozřejmostí při výstavbě matematického modelu uvažované soustavy je volba takové vstupně - výstupní struktury neuronové sítě, která bude odpovídat struktuře identifikované soustavy. Označíme-li tedy výstup NS symbolem y^m , potom rovnice modelu nelineární soustavy

$$y^m(k+1) = \hat{f}[y^m(k), \dots, y^m(k-n_y+1), u(k), \dots, u(k-n_u+1)] \quad , \quad (2.2)$$

¹ Prof. Ing. Ivan Taufer, DrSc. a Prof. Ing. Oldřich Drábek, CSc. Katedra řízení procesů a výpočetní techniky, Fakulta chemicko – technologická, Univerzita Pardubice, 532 10 Pardubice, nám. Čs. Legií 565, tel.: +420 40 603 7123, fax: +420 40 603 7068, e-mail: ivan.taufer@upce.cz.

(Recenzovali: Prof. Ing. Karol Kostúr, CSc. a Prof. Ing. Imrich Košťál, CSc. Revidovaná verzia doručená 25.11.1998)

kde $\hat{f}[\cdot]$ reprezentuje aproximaci nelineární závislosti mezi vstupem a výstupem soustavy. Poznamenejme, že často je $n_u = n_y = n$. Uvažujme dopředné modelování, při kterém je model připojen k soustavě paralelně (obr.1).



⇐

Obr.1. Dopředné modelování nelineárního systému.

$$e(k+1) = y^s(k+1) - y^m(k+1) \quad (2.3)$$

je vstupem algoritmu učení neuronové sítě.

Vzorky vstupního signálu $u(k)$ (např. pořadnice skokové funkce) a odezvy soustavy $y^s(k)$, snímané v časových okamžicích $k = -n, -n+1, \dots, N-n$, tvoří spolu s posunutými pořadnicemi vstupní vzory neuronové sítě. Pořadnice na výstupu soustavy $y^s(k+1)$ je vzorem pro výstup modelu $y^m(k+1)$. Poznamenejme, že zmíněné vzory nemusíme vždy získat jako výsledek on-line experimentu, naznačeného na obr. 1. Lze použít i dobře známé metody off-line, při které není model fyzikálně propojený s identifikovanou soustavou. Jak v prvním, tak i v druhém případě jsou vstupní vzory neuronové sítě

$$\begin{aligned} z^{i+1} u(k), \quad i = 1, 2, \dots, n; \quad k = -n, -n+1, \dots, N-n-1, \\ z^i y^s(k+1), \quad i = 1, 2, \dots, n; \quad k = -n, -n+1, \dots, N-n-1 \end{aligned}$$

a výstupní vzory

$$y^s(k+1), \quad k = -n, -n+1, \dots, N-n.$$

Tyto vzory tvoří tzv. trénovací množinu. Např. pro nelineární soustavu, u které lze předpokládat popis nelineární diferenční rovnicí

$$y^m(k+1) = \hat{f}[y^s(k), u(k)] \quad , \quad (2.4)$$

je trénovací množina tvořena N trojicemi dat (tab.1). Z uvedených dat dvojice v prvním sloupci jsou vstupem neuronové sítě, zatímco pořadnice ve druhém sloupci jsou výstupem soustavy a slouží k vyjádření chyby (2.3) a k následnému učení neuronové sítě algoritmem se zpětným šířením v čase. Opa-kovaným průchodem trénovací množiny BP - algoritmem v p_p iteračních krocích, dochází k trénování sítě, k postupnému zpřesňování synaptických vah mezi neurony a ke snižování hodnoty trénovací účelové funkce

$$E_w(p_p) = \frac{1}{N} \sum_{k=0}^N [y^s(k) - y^m(k)]^2 \quad . \quad (2.5)$$

Jestliže v průběhu trénování $E_{tr}(p_p) < \varepsilon$, kde ε je zvolené malé číslo, je trénování sítě ukončeno a

$$y^m(k) \cong y^s(k) \quad (2.6)$$

s hodnotou trénovací účelové funkce $E_{tr}^{(e)}(p_p)$.

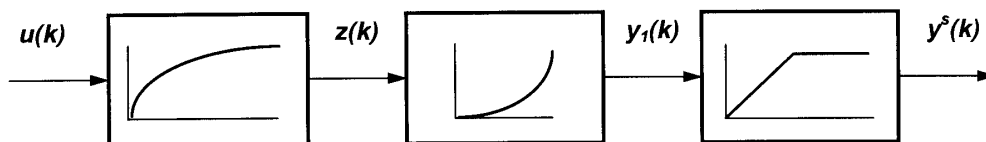
Účelem identifikace je však obvykle určení modelu ve tvaru neuronové sítě s takovými vlastnostmi, aby rovnice (2.6) platila pro *jakýkoliv* vstupní signál, tzn. nejen pro signál, který byl východiskem pro konstrukci trénovací množiny. Tzn., že požadujeme, aby neuronová síť byla schopna určitého stupně zobecnění, jinými slovy, aby bylo dosaženo jisté rovnováhy mezi odezvou neuronové sítě na trénovací množinu a odezvu sítě na testovací množinu. Konstrukce testovací množiny tedy vychází z jiného vstupního signálu, než z kterého vzešla trénovací množina, a to např. z exponenciální funkce, impulzu, rampy a pod. Zjišťování stupně zobecnění pak spočívá ve střídání procesu trénování a testování neuronové sítě při průběžném sledování kritéria testovací účelové funkce

$$E_{ts}(p_p) = \frac{1}{N} \sum_{k=0}^N [y_{ts}^s(k) - y_{te}^m(k)]^2, \quad (2.7)$$

kde $y_{ts}^s(k)$ je výstup soustavy a $y_{te}^m(k)$ je výstup neuronové sítě v k -tém kroku. Kritérium $E_{ts}(p_p)$ vykazuje v závislosti na p_p extrém (minimum). Po jeho dosažení testování ukončíme a hodnoty synaptických vah, vypočtené v tomto iteračním kroku, považujeme za optimální.

Specifikace identifikovaného systému, neuronové sítě a algoritmu učení

Předmětem identifikace je nelineární soustava popsána modelem Wienerova typu s nelineárním blokem nasycení na výstupu - viz obr.2. Soustava je simulována na číslicovém počítači prostřednictvím programu analogové simulace (Pivoňka & Sládeček, 1991).



Obr.2. Blokové schéma nelineárního systému.

Dynamická část soustavy je popsána přenosem

$$\frac{Z(p)}{U(p)} = \frac{K}{1 + pT_1}, \quad (K = 1; \quad T_1 = 5), \quad (3.1)$$

statická část parabolickým průběhem

$$y_1(k) = s_2 z^2(k), \quad (s_2 = 1), \quad (3.2)$$

a blok nasycení

$$y^s(k) = \begin{cases} A y_1(k), & \text{pro } 0 < y_1(k) < 1/A, \\ 1, & \text{pro } y_1(k) > 1/A. \end{cases} \quad (A = 1,1), \quad (3.3)$$

Modelem soustavy je dvouvrstvá neuronová síť s jednou skrytou vrstvou a s jedním výstupním neuronem. Simulační výpočty jsou provedeny pro různý počet vstupů neuronové sítě a pro různý počet neuronů ve skryté vrstvě. Za základ simulace je vzata neuronová síť NS 2-3-1 (neuronová síť s dvěma vstupy, s třemi neurony ve skryté vrstvě a s jedním výstupním neuronem). Výsledky trénování

a testování této sítě jsou porovnány s výsledky dosaženými s různými dalšími sítěmi, z nichž jsou dále prezentovány výsledky dosažené na sítích NS 4-10-1 a NS 6-5-1.

Síť je trénována algoritmem učení se zpětným šířením. Všechny neurony výše uvedených sítí mají přenosové binární funkce sigmoidálního tvaru a pracují v rozsahu (0; 1). Strmosti sigmoidálních funkcí jsou v průběhu učení neuronové sítě buď konstantní nebo adaptovatelné. Startovací hodnoty synaptických vah a strmostí lze zadávat jako deterministická data, nebo z generátoru náhodných čísel, nacházejících se pro synaptické váhy v rozsahu (-0,5; 0,5) a pro strmosti (0; 5). Trénovací množiny jsou konstruovány na základě vstupního signálu ve tvaru jednotkové skokové funkce a odpovídajících odezev. Konstrukce testovacích množin vychází ze vstupního signálu ve tvaru exponenciální funkce s časovou konstantou $T_{ex} = 2$, impulsu o výšce 1 a šířce $T_{im} = 10$ a rampy s integrační časovou konstantou $T_i = 10$.

Trénování a testování neuronových sítí

Dříve než začneme s trénováním neuronových sítí, je třeba zvolit určité parametry, které jsou mnoha autory přísně střeženy, jako jejich "know - how". Základní informace o volbě těchto parametrů (volba startovacích hodnot synaptických vah, doba trénování, vhodný počet vzorů, počet neuronů ve skryté vrstvě) lze nalézt např. v (Fausset, 1994). Poznamenejme, že většina profesionálních programových produktů, určených k trénování neuronových sítí, řeší některé z těchto problémů automaticky.

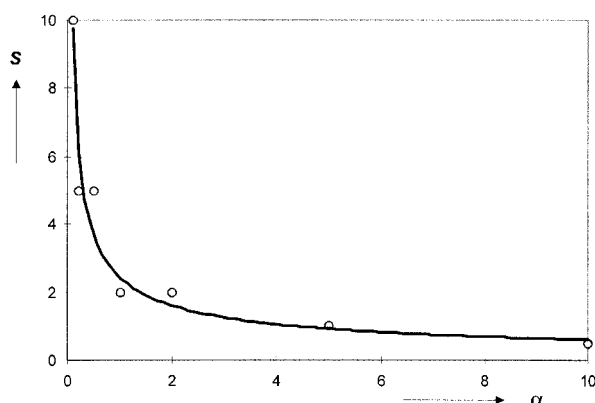
Jsou odzkoušeny dvě varianty trénování neuronových sítí:

- trénování s adaptivním nastavováním synaptických vah a s konstantní strmostí sigmoidálních přenosových funkcí,
- trénování s adaptivním nastavováním synaptických vah i strmostí.

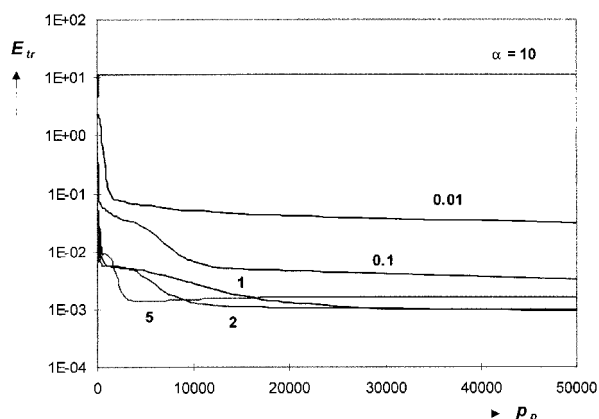
Trénování při konstantní strmosti sigmoidálních funkcí

K tomuto účelu je použita neuronová síť NS 2-3-1. Startovací hodnoty synaptických vah jsou zadávány z generátoru náhodných čísel v reprodukovatelném režimu, z důvodů možnosti porovnání dosažených výsledků. Pro jednotlivé výpočty jsou strmosti S a koeficient učení α zadávány jako determinované hodnoty. Neuronová síť je trénována pro různé hodnoty α a S . Maximální počet iteračních kroků byl zvolen $p_p = 50.000$. Trénováním jsou sledovány dva cíle:

- zjištění relativní minimální hodnoty trénovací účelové funkce $E_{tr}^{(min)}$,
- posouzení charakteru průběhu $E_{tr} = f(p_p)$.



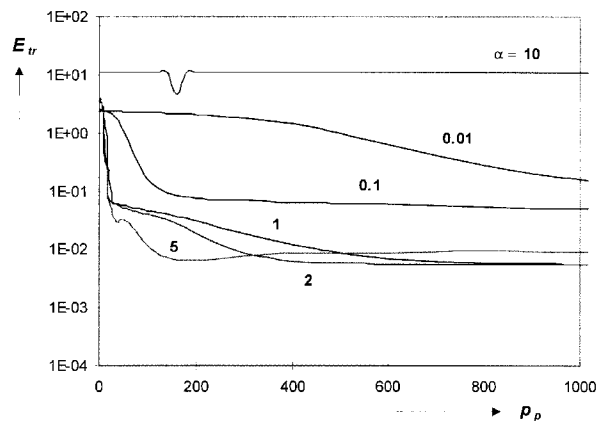
Obr.3. Geometrické místo bodů s relativní, minimální hodnotou $E_{tr}^{(min)}$.



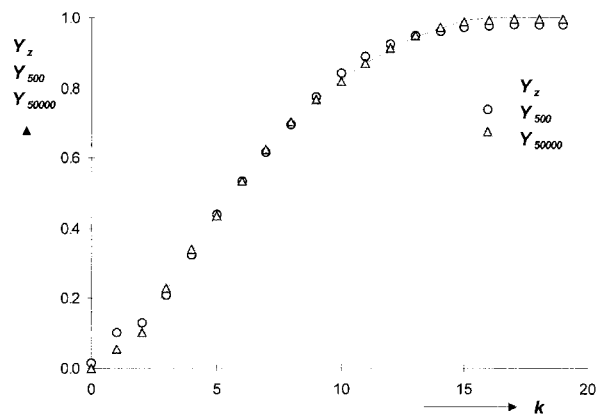
Obr.4. Závislost $E_{tr} = f(p_p)$ pro $S = 2$ a různé hodnoty α .

Výsledek vyhodnocení prvního dílčího cíle je patrný z grafu na obr. 3, kde je uvedena závislost $S = f(\alpha)$ pro $E_{tr}^{(min)}$. Graf na obr. 3 lze považovat za geometrické místo bodů relativních minimálních hodnot $E_{tr}^{(min)}$ pro odpovídající hodnoty strmosti a koeficienty učení. Při nevhodné volbě parametrů S a α může být rozdíl v hodnotách kritérií $E_{tr}^{(min)}$ velmi značný.

Charakter procesu trénování v rozsahu 0 - 50.000 iteračních kroků lze kvalitativně posoudit z průběhů na obr. 4. Jednotlivé průběhy účelových funkcí jsou podle očekávání klesající.



Obr. 5. Průběhy trénovacích procesů po odstartování algoritmu (S-konst.).



Obr. 6. Časový průběh odezvy soustavy a neuronové sítě pro vstupní signál ve tvaru skoku.

Pro dané startovací hodnoty strmostí existuje určitý interval koeficientů učení ($\alpha_{\min}$; $\alpha_{\max}$), ve kterém je trénování neuronové sítě neefektivní. Pokud $\alpha \notin (\alpha_{\min}, \alpha_{\max})$, nedochází k trénování neuronové sítě vůbec (viz obr. 3, $\alpha = 10$), nebo se nám sice podaří síť natrénovat, ale s velikou hodnotou kriteriální funkce $E_{tr}^{(min)}$ (viz obr. 3; $\alpha = 0,01$). Se zvyšováním α uvnitř intervalu se zvyšuje rychlost učení neuronové sítě. Z průběhů na obr. 4 a 5 je patrné, že v případě trénování dochází k výraznému poklesu E_{tr} ve dvou stupních: k prudkému snížení dochází v prvních několika stovkách průchodů trénovací množiny algoritmem učení a k druhému, mírnějšímu poklesu dochází v dalších několika tisícovkách iteračních kroků.

Pro ilustraci jsou na obr. 6 uvedené odezvy soustavy na jednotkový skok a odezvy neuronové sítě trénované s parametry $S = 2$, $\alpha = 1$ a $E_{tr}^{(min)}$, dosažené při $p_p = 500$ a $p_p = 50.000$.

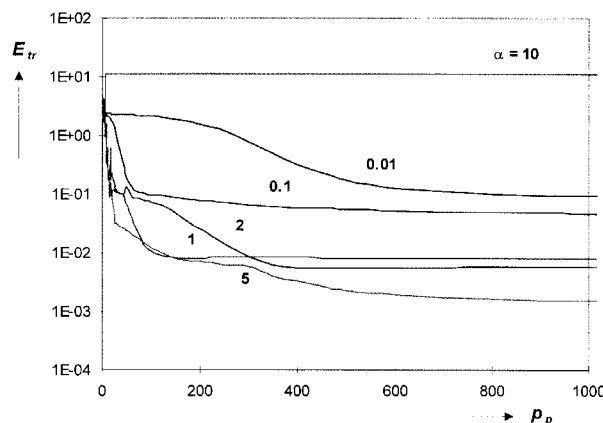
Z vizuálního posouzení průběhů na obr. 6 vyplývá, že

- neshoda mezi odezvami modelů a soustavy je značná v oblasti vysokých kmitočtů,
- teprve zvýšením počtu průchodů trénovací množiny algoritmem učení na velmi vysokou hodnotu se k chování soustavy přiblížíme výrazněji.

Trénování při adaptaci strmosti sigmoidálních funkcí

Za účelem porovnání procesů učení s konstantní a proměnnou strmostí sigmoidálních funkcí je proveden výpočet závislosti $E_{tr} = f(p_p)$ pro náhodné startovací hodnoty strmostí všech neuronů, dříve zmíněné koeficienty učení a pro $p_p = 50.000$. Posouzení výsledků těchto výpočtů ve vztahu k výsledkům graficky zobrazeným na obr. 4 vyznívá ve prospěch učení s adaptací strmostí:

- pro $p_p = 50.000$ dosáhneme cca o 20% nižší hodnoty $E_{tr}^{(min)}$,
- průběh adaptace po startu algoritmu je výrazně rychlejší, i když svým charakterem odpovídá průběhu bez adaptace (viz obr. 7, porovnej s obr. 5).



Dále z výsledků simulace vyplývá, že

- se zvyšujícím se počtem vstupů modelu a počtem neuronů ve skryté vrstvě se zvyšuje hodnota koeficientu učení, potřebná k dosažení minimální hodnoty účelové funkce,
- se zvyšujícím se počtem vstupů modelu a počtem neuronů ve skryté vrstvě klesá E_{tr} ,
- nejlepších výsledků trénování je dosaženo v případě NS 4-10-1 a NS 6-5-1.

Obr. 7. Průběhy trénovacích procesů po odstartování algoritmu (S-adapt.).

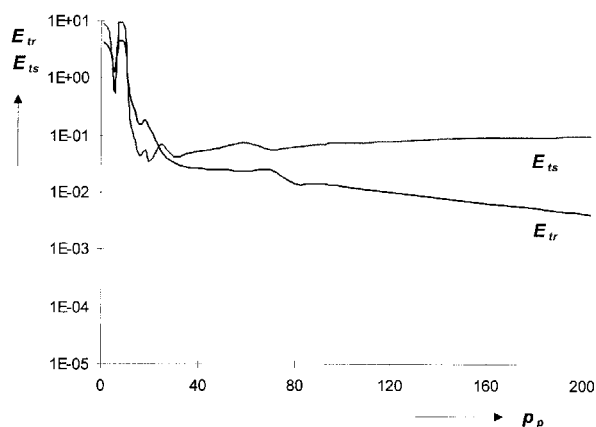
Testování NS

Testování je realizováno na čtyřech neuronových sítích s testovacími množinami konstruovanými ze zkušebních signálů ve tvaru exponenciály, impulzu a rampy. Analýzou výsledků simulace jsme dospěli k následujícím kvalitativním závěrům:

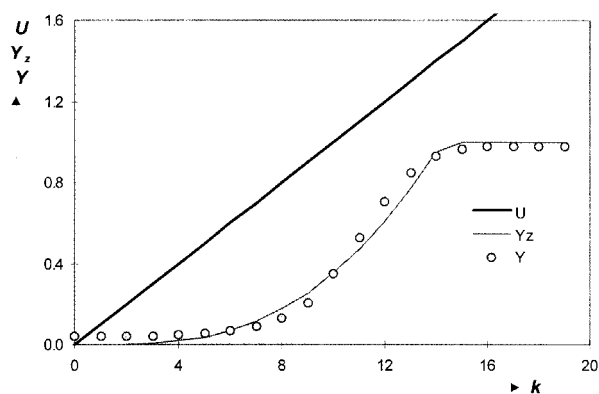
- v průbězích $E_{tr}=f(p_p)$ jsou patrné lokální extrémy,
- pro určitou testovací množinu vzorů se hodnota E_{ts}^{opt} se zvyšujícím se počtem vstupů, resp. neuronů ve skryté vrstvě výrazně nemění. K výraznější změně dochází v optimálním počtu iteračních kroků,
- pro danou neuronovou síť je $E_{tr}^{(opt)}$ tím menší a počet $p_p^{(opt)}$ tím vyšší, čím více se testovací množina svým charakterem přibližuje k množině trénovací.

Pro ilustraci je na obr.8 uveden průběh $E_{ts}=f(p_p)$ a $E_{tr}=f(p_p)$ pro trénovací množinu ve tvaru skoku a testovací množinu ve tvaru rampy pro NS 4-10-1.

Z průběhu na obr.9 si lze učinit představu o míře shody odezvy soustavy a modelu na vstup ve tvaru rampy pro NS 4-10-1.



Obr.8. Závislost $E_{tr}=f(p_p)$ a $E_{ts}=f(p_p)$ při trénování skokem a testování rampou.



Obr.9. Časový průběh odezvy soustavy a NS pro vstupní signál ve tvaru rampy.

Závěr

Z výsledků popsané simulace trénování a testování konkrétní, poměrně jednoduché nelineární soustavy nelze vyvozovat obecně platné kvantitativní závěry. Lze však získat určitou kvalitativní představu o vlivu různých parametrů na přesnost procesu trénování, na potřebnou dobu trénování a rychlost konvergence.

Výstavba modelu ve tvaru neuronové sítě vyžaduje tedy provedení řady simulačních výpočtů, které jsou předpokladem pro vytvoření solidního modelu natrénovaného na určitou trénovací množinu vzorů a schopného zobecnění na testovací množinu jiných vzorů, které mohou přijít v úvahu při provozování neuronové sítě. Při této příležitosti znovu připomeňme, že u profesionálních programových produktů jsou mnohé problémy spojené s nastavením potřebných parametrů řešeny automaticky.

Literatura

- Bíla, J.: Umělá inteligence a neuronové sítě v aplikacích. ČVUT, Praha, 1995.
- Fausset, L.: Fundamentals of Neural Networks. Prentice-Hall, Inc., 1994.
- Hertz, J. et al.: Introduction to the Theory of Neural Computation. Santa Fe Institute, 1991.
- Hunt, K.J. et al.: Neural Networks for Control Systems - A Survey. Automatica, Vol. 28, No. 6, 1992.
- Kvasnička, V. et al.: Úvod do teorie neuronových sítí. IRIS, Bratislava, 1997.
- Pach, Ch. et al.: Dynamic System Identification by Neural Networks. JSME Int. Jour., Vol.38/4, 1995.
- Pivoňka, P. & Sládeček, L.: Simulace dynamických systémů. Automatizace, 34, č.3, 1991.
- Sinčák, P. & Andrejková, G.: Neuronové siete, (2.diel). Elfa, s.r.o., Košice, 1996.
- Sjöberg, J., et al.: Neural Network in System Identification. Report LiTH-ISY-R-1622, 1994.
- Šnorek, M. & Jiřina, M.: Neuronové sítě a neuropočítače. ČVUT, Praha, 1996.
- Warwick, K.: Identification in Engineering Systems. In: Proce. of the Conf.e helh at Swansea, M, 1996.

Yamada, T. & Yabuta, T.: Dynamic System Identification Using Neural Networks. *IEEE Trans. on Systems, Man and Cybernetics*, Vol. 23, No. 1, 1993.