

Využití neuronové sítě pro identifikaci reálného systému

Pišan Radim · Elektrotechnika

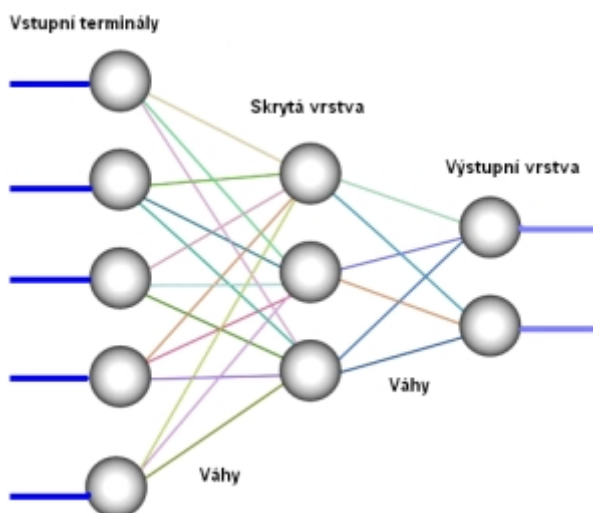
20.06.2011



Identifikace systémů je proces, kdy z naměřených dat můžeme odvodit parametry matematického modelu. Pro buzení těchto soustav je vhodné použít různé vstupní signály jako jsou např. náhodný, pseudonáhodný binární signál, bílý šum. Odezvou na tento vstupní signál je výstupní signál. Asi nejznámější a nejpoužívanější klasickou metodou pro výpočet odhadu parametrů modelu je metoda nejmenších čtverců. Cílem tohoto příspěvku ale není popis klasické identifikace, ale využití umělé inteligence a to neuronových sítí. Existuje několik typů neuronových sítí. Pro potřeby tohoto příspěvku byla vybrána dopředná neuronová síť a algoritmus učení je použit Marquardt-Levenberg.

Neuronová síť

Na Obr. 1 je schéma dopředné neuronové sítě. Síť se skládá ze vstupních terminálů (vstupy do sítě), pak následuje skrytá vrstva (může jich být i několik), poslední vrstva se nazývá výstupní. Každá vrstva obsahuje jednotlivé neurony. Spojení mezi vrstvami obstarávají váhy. Jejich počáteční hodnota je vygenerována jako náhodné číslo nejčastěji v rozsahu $(-1,1)$. Váhy se dále využívají pro učení neuronové sítě.



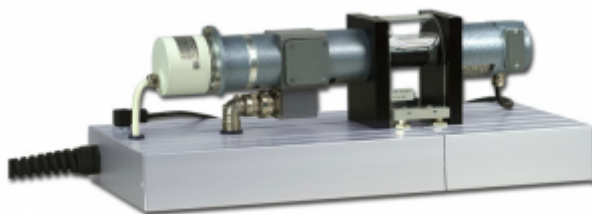
Obr. 1 Schéma dopředné neuronové sítě

Průchod neuronovou sítí lze rozdělit na dvě etapy. V první fázi jsou přivedena vstupní data na vstupní terminály a jsou ohodnocena příslušnými váhami. Dále je udělána suma a takto vniklé data vstupují do neuronu. V neuronu je implementována přenosová funkce, která převádí vstup do neuronu na jeho výstup. Mezi

nejpoužívanější funkce patří: gaussovská, hyperbolicko tangenciální, lineární saturaovaná, lineární, sigmoidální, skoková. Takto se postupuje až na výstup neuronové sítě, kde se porovná hodnota výstupu s hodnotou v tréninkové množině, vzniká chyba pro aktuální pár tréninkových hodnot. Celková chyba se určí jako suma jednotlivých chyb. Ta je použita ve druhé fázi tzv. zpětného šíření, kde probíhá proces učení a to tak, že učicí algoritmus přenastavuje váhy. Jde se tedy směrem od výstupu ke vstupu.

Servomechanismus

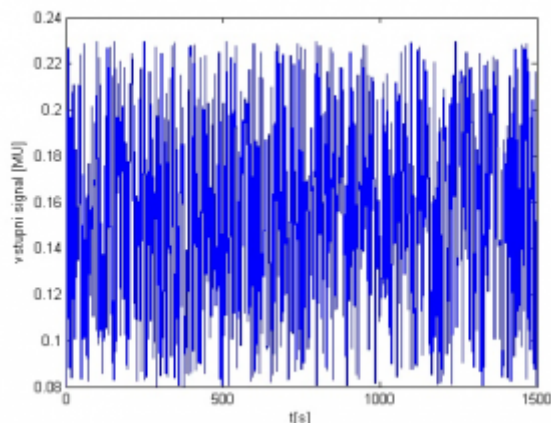
Pro ověření neuronové sítě na reálných datech, byl vybrán servomechanismus Amira DR300. Tento systém je tvořen trvale buzeným DC-motorem. Tento motor pracuje ve čtyřkvadrantním režimu. Pro snímání otáček je použito tachodynamo a inkrementální snímač. Motor je připojen k technologické kartě, která je umístěna v PC. Pro ovládání motoru je využíváno programové prostředí MATLAB-SIMULINK a jeho Real-time toolbox. Na Obr. 1 je vidět servomechanismus Amira DR300. Tato soustava má pásmo necitlivosti saturační pásmo a je také mírně nelineární.



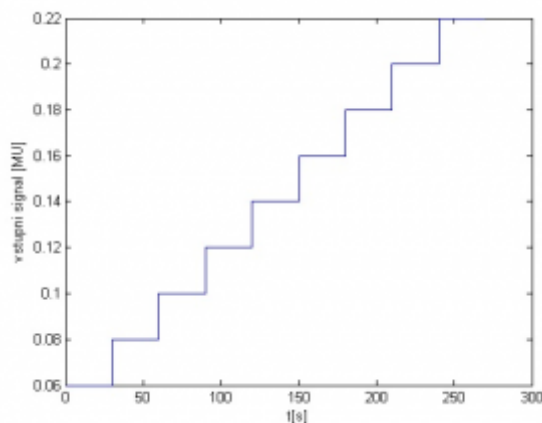
Obr.2 Servomechanismus Amira DR300

Experimentální část

Jako vstupní signál, který slouží k buzení motoru (vstupní napětí) byl vybrán náhodný signál. Tento signál byl použit jen v jednom pracovním kvadrantu systému Amira DR 300, byl tvaru obdelníků s periodou 1,5 s a byl přiveden jako vstup do motoru Obr. 2. Výstupem jsou otáčky motoru θ - byly snímány s periodou 0,1s. Takto vzniklá tréninková množina byla použita pro učení neuronové sítě. Pro ověření naučené neuronové sítě, je vhodné použít jiný vstupní signál, v tomto případě byl použit signál tvaru skoků Obr. 3. Rozměrové jednotky uváděné u jednotlivých veličin [MU], nepopisují reálné hodnoty veličin, ale pouze jednotky implementované v Real-time toolboxem.

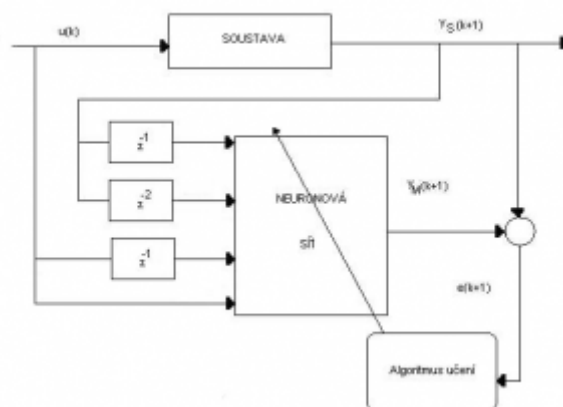


Obr. 3 Náhodně vygenerovaný signál pro tréninkovou množinu



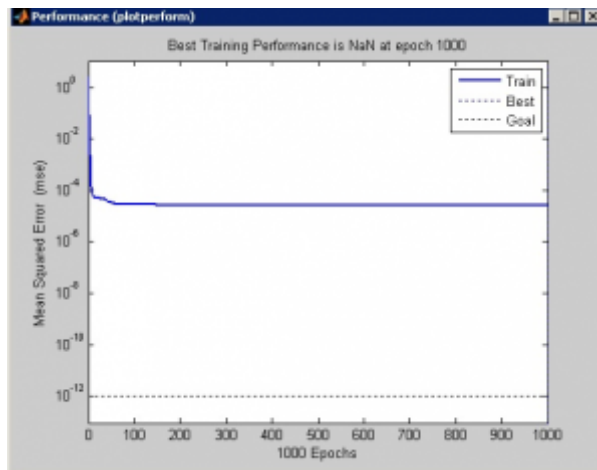
Obr. 4 Testovací signál tvaru skoků

Dalším parametrem, který je potřeba určit, je struktura neuronové sítě. Tento parametr ovlivňuje schopnost neuronové sítě se učit. Pro soustavu Amira DR300 se jako velmi dobrá jevila síť v konfiguraci 4-4-1, což představuje 4 vstupní terminály, 4 neurony ve skryté vrstvě a jeden výstupní neuron. Přenosová funkce ve skryté byla sigmoidální a ve výstupní vrstvě funkce lineární. Na Obr. 5 lze najít schéma zapojení neuronové sítě k soustavě.



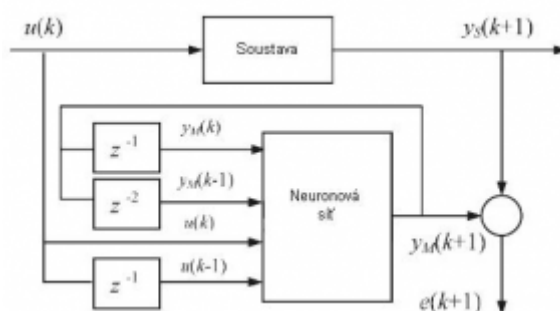
Obr. 5 Schéma zapojení neuronové sítě k soustavě

Bloky z^{-1} a z^{-2} reprezentují zpoždění o jeden, respektive o dva kroky. Jak lze vidět z Obr. 5 na vstup sítě jsou přivedeny i výstupy ze soustavy. Takto vytvořená tréninková množina je připojena k neuronové síti. Pro učení sítě byl použit Levenberg-Marquardt algoritmus, který je implementován v programovém prostředí MATLAB, konkrétně v Neural Network Toolboxu. Parametry pro učení byly následující: počet epoch trénování 1000, chyba, při které dojde k ukončení učení $1 \cdot 10^{-7}$. Na Obr. 6 můžeme vidět vývoj chyby. Výsledná chyba se zastavila na hodnotě $2,38 \cdot 10^{-5}$.



Obr. 6 Vývoj celkové chyby

K ověření natrénované neuronové sítě bylo použito schéma na Obr. 7. Jak můžeme vidět, jednotlivé výstupy z neuronové sítě vstupují zpět na vstupní terminály a to se zpožděním jednoho respektive dvou kroků.

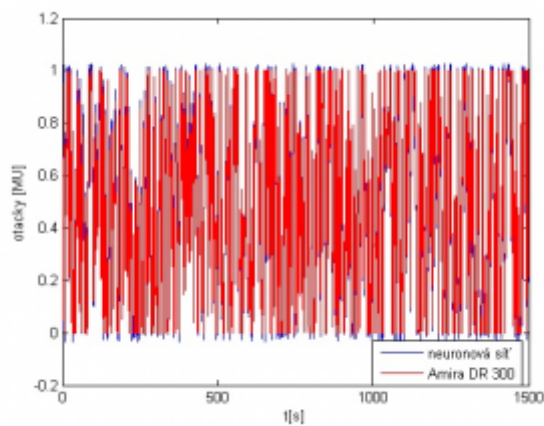


Obr. 7 Schéma testování neuronové sítě

Nejprve byla vybrána tréninková množina, která sloužila k učení sítě. Ta byla přivedena na vstupní terminály podle Obr. 7. Tyto vstupní data jsou tvořena jen vstupy do soustavy (jen vstupní napětí, které bylo přivedeno na vstup motoru). Výstupní hodnoty z neuronové sítě (otáčky motoru), byly porovnávány se skutečnými hodnotami otáček motoru a byla vypočítána odchylka těchto hodnot a to podle vzorce:

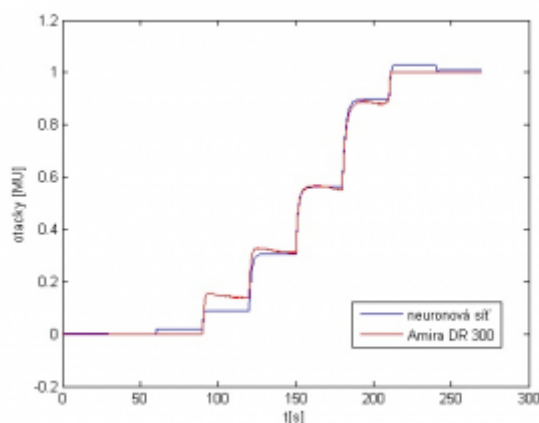
$$S = \frac{1}{N} \sum_{i=1}^N (o_{ni} - o_{ri})^2 \quad (1)$$

kde N - počet hodnot tréninkové množiny, o_n - i -tý výstup z neuronové sítě, o_r - i -tý změřený výstup ze soustavy Amira DR 300



Obr. 8 Testování na tréninkovou množinu

Obr. 8 ukazuje odezvu soustavy a neuronové sítě na první testovací signál. Z obrázku je patrné, že shoda je poměrně velká. Odchylka činila $9,54 \cdot 10^{-4}$.



Obr. 9 Testování na skokové změny

Jako druhá testovací funkce byla použita funkce skoková. Na Obr. 9, můžeme vidět odezvy reálného systému Amira DR 300 a neuronové sítě na druhý testovací signál. Jde vidět, že neneuronová síť docela dobře aproximuje daný systém. Odchylka byla vypočítána na hodnotu $5,50 \cdot 10^{-4}$.

Poděkování

Tento příspěvek vznikl za podpory interní grantové agentury Univerzity Tomáše Bati ve Zlíně s číslem IGA/55/FAI/10/D

Závěr

Cílem toho příspěvku bylo, ukázat možnost identifikace reálných zařízení za pomoci neuronové sítě. Z provedených simulací je patrné, že neuronové sítě dovedou docela dobře aproximovat tyto reálné soustavy. Umějí se vypořádat i s nelineárními systémy a také se systémy, které mají např. pásmo necitlivosti, popřípadě pásmo saturace, jak je ukázáno v tomto příspěvku. Značnou nevýhodou je sestavení tréninkové množiny. Jedná se především o periodu vzorkování dat a výběr vhodného budícího signálu. Další problém lze nalézt ve volbě struktury sítě a počtu vrstev. Zde by se daly uplatnit především genetické algoritmy, které by tento problém mohly řešit.

Literatura

1. DREYFUS, G. Neural networks methodology and applications, Editions Eyrolles, 2005, XVIII, 497 p. 212 illus., Hardcover, ISBN: 978-3-540-22980-3
 2. NORGAARD, M., RAVN, O., POULSEN, N.K., HANSEN, L.K., Neural networks for modelling and control of dynamic systems, Softcover, 2000, XIV, 246 p. 84 illus., ISBN: 978-1-85233-227-3
 3. HAGAN, Martin; MENHAJ, Mohammad. Training Feedforward Networks with the Marquardt Algorithm. Transactions of neural network. 1994, 5, 6, s. 989-993.
 4. TAUFER, I., DRÁBEK, O., SEIDL, P., Umělé neuronové sítě-teorie a aplikace (6). CHEMagazín, 6 (XVI), 2006, s. 31-33. ISSN 1210-7409
 5. Amira measuring and control technology [online]. 2006 [cit. 2011-06-13]. Laboratory equipment. Dostupné z WWW: <http://www.amira.de/dr300_engl.html>.
-