

Prednáška 9: INTELIGENTNÉ RIADENIE NELINEÁRNYCH PROCESOV S VYUŽITÍM NEURÓNOVÝCH SIETÍ

Riadenie a Umelá Inteligencia

doc. Ing. Anna Jadlovská, PhD.,

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

LS 2015/2016

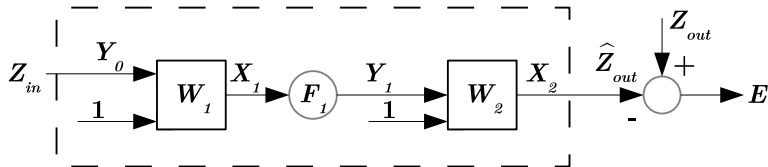
Inteligentné riadenie nelineárnych procesov s využitím neurónových sietí

- **Neurónová sieť** je masívne paralelný distribuovaný procesor, ktorý má schopnosť uchovávať poznatky získané skúsenosťou a mať ich k dispozícii pre použitie
- **Sieť** sa podobá mozgu z dvoch hľadísk:
 - ▶ poznatky získava procesom učenia
 - ▶ na uchovanie poznatkov sa používajú medzineurónové spojenia \cong synaptické váhy
- **Procedúra**, ktorou sa sieť učí nazývame algoritmus učenia \cong tréningový algoritmus \Rightarrow základom výpočtovej sily NN:
 - ▶ masívna paralelná a distribuovaná štruktúra,
 - ▶ schopnosť učiť sa a zovšeobecňovať

\rightarrow Pre využitie teórie NN v **identifikácii** a **riadení** nelineárnych dynamických systémov je najčastejšie používaným typom **VIACVRSTVOVÁ PERCEPTRÓNOVÁ SIETĚ** (NN typu MLP) a na tréningovanie sa používa **algoritmus spätného šírenia chyby** (Back-Propagation Algorithm - BPA)

9.1 Viacvrstvá perceptrónová sieť s jednou skrytou vrstvou

NN typu MLP s jednou skrytou vrstvou sa dá vyjadriť pomocou maticovej notácie → Obr. 1



Obr. 1: Architektúra NN typu MLP s jednou skrytou vrstvou vyjadrená v maticovej notácii

Pozn.: Sieťe typu MLP iba s jednou skrytou vrstvou postačujú k realizácii funkcie univerzálneho aproximátora (počet vrstiev).

9.1 Viacvrstvá perceptrónová sieť s jednou skrytou vrstvou

- počet neurónov v každej vrstve - počet neurónov m_0 vo vstupnej vrstve a počet neurónov m_3 vo výstupnej vrstve \rightarrow je daný počtom prvkov vstupného (Z_{in}) a výstupného vektora (Z_{out})

[málo zvolených neurónov nedovolí NN MLP aproximovať všetky nelinearity zahrnuté v procese; príliš veľa neurónov zvyšuje riziko pretrénovania siete]

- výstupné (aktivačné) funkcie neurónov - časté použitie má SIGMODIÁLNA AKTIVAČNÁ FUNKCIA:

$$Y = f(x) = \frac{1}{(1 + \exp(-x))}$$

aktivačná funkcia hyperbolický tangens: $Y = f(x) = \tanh(x)$

9.1 Viacvrstvová perceptrónová sieť s jednou skrytou vrstvou

Vektor X (reprezentujúci vnútornú aktivitu neurónu) - v danej vrstve MLP siete prenáša informáciu z každého výstupu neurónov predchádzajúcej vrstvy pri konštantnom externom vstupe (bias) obr.1

$W_1, W_2 \rightarrow$ **váhové matice** (každý riadok reprezentuje váhy príslušného neurónu v danej vrstve a posledný stĺpec obsahuje hodnoty prahových koeficientov)

$X_i, Y_i \rightarrow$ sú vektory v i -tej vrstve

Táto štruktúra (obr. 1) bude aplikovaná na riešenie úloh identifikácie a riadenia NDS:

- výstup siete:

$$X_2 = W_2 F_1 \left(W_1 \begin{Bmatrix} Y_0 \\ 1 \end{Bmatrix} \right) \quad (1)$$

9.1 Viacvrstvá perceptrónová sieť s jednou skrytou vrstvou

Pre neurónovú sieť s jednou skrytou vrstvou:
BPA v maticovej notácii:

1. výpočet výstupného vektora:

$$\begin{aligned} X_1 &= W_1 \begin{Bmatrix} Y_0 \\ 1 \end{Bmatrix}, & Y_1 &= F_1(X_1) \\ X_2 &= W_2 \begin{Bmatrix} Y_1 \\ 1 \end{Bmatrix}, & Y_2 &= X_2 \end{aligned} \quad (2)$$

2. výpočet lokálneho gradientu δ a spätné šírenie chyby:

$$\begin{aligned} \delta_2 &= (-E) \\ \delta_1 &= F_1'(X_1) W_2^T \delta_2 \\ \delta_0 &= (W_1)^T \delta_1 \end{aligned} \quad (3)$$

9.1 Viacvrstvová perceptrónová sieť s jednou skrytou vrstvou

3. zmena váh v maticiach W_1 a W_2 :

$$\begin{aligned} G_2 &= \delta_2 Y_1'; & W_2(k) &= W_2(k-1) - \eta G_2 + \alpha(W_2(k-1) - W_2(k-2)) \\ G_1 &= \delta_1 \begin{Bmatrix} Y_0 \\ 1 \end{Bmatrix}^T; & W_1(k) &= W_1(k-1) - \eta G_1 + \alpha(W_1(k-1) - W_1(k-2)) \end{aligned} \quad (4)$$

Predpokladáme, že pred zahájením algoritmu BPA sú inicializované **váhy** a **prahy** siete MLP. Výpočet matice zesilnenia N :

$$\begin{aligned} N &= \frac{d\hat{Z}_{out}}{d\hat{Z}_{in}^T} = \frac{dX_2}{dY_0^T} = \frac{dX_2}{dY_1^T} \frac{dY_1}{dX_1^T} \frac{dX_1}{dY_0^T} = \\ &= W_2 F_1'(X_1) W_1 \end{aligned} \quad (5)$$

9.2 Dynamické neurónové modely

- NN - reprezentujú nový trend v modelovaní DS, pretože na rozdiel od analytických modelov nepotrebujú predchádzajúci fyzikálny náhľad na proces → vychádzajú z nameraných vstupno-výstupných dát
- napriek rôznym snahám o úplné nahradenie riadiaceho systému prvkami NN treba konštatovať, že ťažisko efektívneho využitia NN v riadení tkvie v tvorbe takých štruktúr riadenia, kde sa zúžitkujú kladné vlastnosti NN (aproximácia ľub. nelineárnej funkcie, silne paralelná štruktúra, schopnosť zovšeobecnenia štruktúr ukrytých v dátach, spoľahlivá aproximácia viacrozmerných systémov)

9.2 Dynamické neurónové modely

- najpodstatnejšou vlastnosťou NN z pohľadu TAR je ich schopnosť modelovať nelineárne procesy \Rightarrow pri využití NN v modelovaní NDS nie je potrebné stanoviť **matematickú formu modelu**, ale stačí nájsť štruktúru dát reprezentujúcu vstupy/výstupy procesu v jeho **pracovnom rozsahu** a potom natrénovať NN, ktorá bude plniť funkciu nelineárneho **vstupno-výstupného** alebo **stavového modelu** riadeného systému. Trénovanie NN použitím vstupno-výstupných dát z procesu \cong problém **nelineárnej regresie**

9.2 Dynamické neurónové modely

- v modelovaní reálnych procesov rozlišujeme tri základné prístupy:
 - ⇒ **deterministické modely** - **white-box models** - štruktúra aj dynamika procesu je popísaná deterministickým modelom → sústava diferenciálnych (diferenčných) rovníc: V/V opis; stavový priestor - na základe známych fyzikálnych vlastností
 - ⇒ **čiastočne známe modely** - **grey-box models** - proces je čiastočne určený pomocou známych fyzikálnych zákonitostí, kým neurčené parametre procesu sa zidentifikujú z nameraných V/V dát $y(k)/u(k)$
 - ⇒ **modely typu čierna skrinka** - **black-box models** - neotrebnú pre získanie modelu poznanie fyzikálnych súvislostí a zákonov, predpokladajú iba známu postupnosť nameraných V/V dát na reálnom procese

Posledné 2 oblasti modelovania sú odlišné od prvej a patria z pohľadu mat. metód do teórie ESTIMÁCIE A NELINEÁRNEJ REGRESIE

9.3 Prehľad rôznych typov identifikačných štruktúr

- riešenie problému identifikácie DS sa skladá z 2 krokov:
 - ▶ z výberu **identifikačného modelu**
 - ▶ z výberu **metódy** pre odhad parametrov modelu na báze výstupnej chyby identifikácie

- neurónové modely poznáme:
 - A. **DOPREDNÉ** → používame v **identifikácii** nelineárnych procesov
 - B. **INVERZNÉ** → vo funkcii spätnoväzobných regulátorov

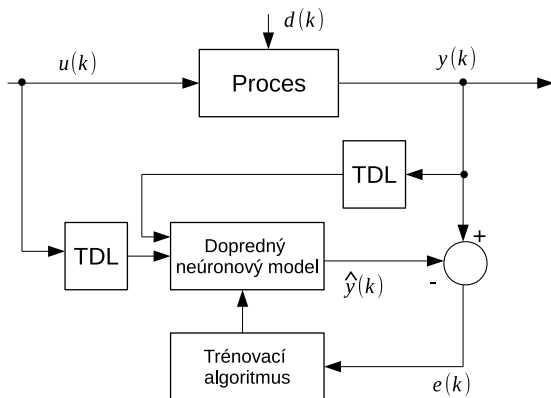
9.3 Prehľad rôznych typov identifikačných štruktúr

A. DOPREDNÉ neurónové modely (Forward modelling)

- neurónový model, kt. aproximuje dynamiku procesu \cong dopredný model
- NN je umiestnená **paralelne** s identifikovaným procesom a chyba medzi výstupom NN $\hat{y}(k)$ a výstupom procesu $y(k)$, tzv. **predikčná chyba** $e(k)$ je použitá na trénovanie NN ako trenovací signál (Obr. 2)
- trénovanie NN je klasické **trénovanie s učiteľom** \cong kontrolované učenie
→ **pričom učiteľom je reálny proces**
- ako NN môže byť použitá **dopredná sieť typu MLP** s algoritmom BPA
→ popísaná štruktúra \cong SÉRIOVO-PARALELNÝ MODEL

9.3 Prehľad rôznych typov identifikačných štruktúr

Identifikačná štruktúra so sériovo-paralelným modelom, TDL - bloky oneskorenia



$\hat{y}(k)$ - výstup z neuronového modelu, $y(k)$ - výstup procesu,
 $e(k)$ - predikčná chyba

9.3 Prehľad rôznych typov identifikačných štruktúr

Ak predpokladáme, že NDS je popísaný nelineárnou diferenčnou rovnicou:

$$y(k) = f\left[\underbrace{y(k-1), \dots, y(k-n)}_{\text{n-výstupných hodnôt procesu}}, \underbrace{u(k-1), \dots, u(k-m)}_{\text{m-vstupných hodnôt procesu}} \right] \quad (6)$$

$\underbrace{\hspace{15em}}_{\text{vstup do NN}}$

vstup do NN môže zahŕňať predchádzajúce hodnoty vstupu a výstupu reálneho systému.

Ak označíme výstup z neurónového modelu $\hat{y}(k)$, aproximácia sa dá zapísať:

$$\hat{y}(k) = \hat{f}[y(k-1), \dots, y(k-n), u(k-1), \dots, u(k-m)] \quad (7)$$

\hat{f} - nelineárne vstupno-výstupné zobrazenie neurónovým modelom (NN typu MLP) \rightarrow algoritmus tréningu BPA, G-L alg., Levenberg - Marg. alg.

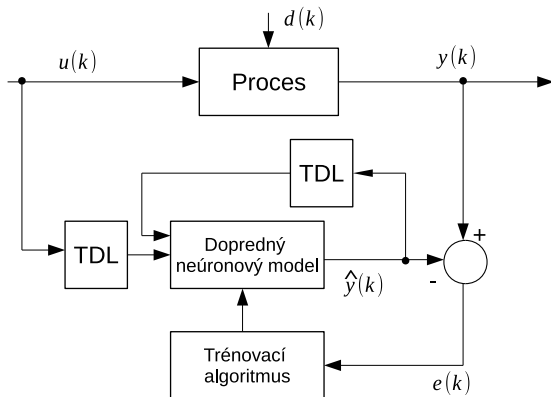
9.3 Prehľad rôznych typov identifikačných štruktúr

Ak po natrénovaní NN (MLP) je chyba identifikácie medzi výstupom NN $\hat{y}(k)$ a výstupom systému $y(k)$ malá ($\hat{y}(k) \sim y(k)$), môžeme výstup neurónového modelu a jeho oneskorené vzorky vrátiť späť na jeho vstup \Rightarrow sériovo-paralelný model nahradzujeme paralelným modelom

[aplikačné využitie je vtedy, ak má byť identifikačný model použitý off-line, t.j. nezávisle na procese].

9.3 Prehľad rôznych typov identifikačných štruktúr

Identifikačná štruktúra s paralelným modelom, TDL - bloky oneskorenia



$\hat{y}(k)$ - výstup z neurónového modelu, $y(k)$ - výstup procesu,
 $e(k)$ - predikčná chyba

9.3 Prehľad rôznych typov identifikačných štruktúr

Neurónový model môže byť popísaný:

$$\hat{y}(k) = \hat{f}[\hat{y}(k-1), \dots, \hat{y}(k-n), u(k-1), \dots, u(k-m)] \quad (8)$$

Uvedený postup môže byť použitý aj na učenie neurónovej siete. Použitie paralelného modelu v identifikačnej štruktúre sa s výhodou prejaví pri problémoch so šumom na výstupe procesu.

Výhoda sériovo-paralelného modelu \rightarrow zaručená konvergencia a stabilita.

9.3 Prehľad rôznych typov identifikačných štruktúr

B. INVERZNÉ neurónové modely (Inverse modelling, INM)

- majú významnú úlohu v teórii riadenia, avšak získanie analytického tvaru je náročná úloha
- ak dopredný model bol popísaný rovnicou (7), inverzný model môžeme vyjadriť v tvare:

$$u(k) = \hat{f}^{-1}[y(k+1), y(k), \dots, y(k-n+1), u(k-1), \dots, u(k-m+1)] \quad (9)$$

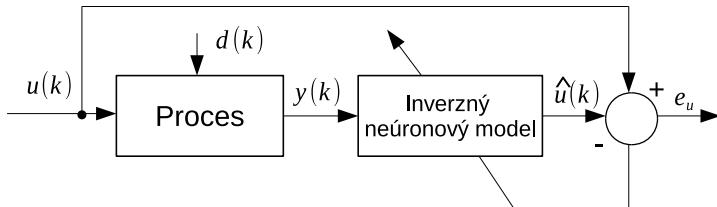
kde $y(k+1)$ nie je známa hodnota a preto ju nahrádzame referenčnou hodnotou riadiacej veličiny $r(k+1)$

- v praxi sa používajú dve architektúry tréovania NN ako inverzného modelu procesu
 - 1 architektúra GENERAL TRAINING (GT)
 - 2 architektúra SPECIALIZED TRAINING (ST)

9.3 Prehľad rôznych typov identifikačných štruktúr

1. Architektúra GENERAL TRAINING

- v architektúre GT je NN (MPL) tréňovaná ako inverzný model procesu učiacim signálom e_u
- pri metóde GT je aplikovaný na vstup procesu signál u za účelom získania odpovedajúceho výstupu procesu y , pričom NN je tréňovaná chybou e_u , ktorá je získaná ako rozdiel výstupu neuronového modelu \hat{u} a vstupného signálu u do procesu



9.3 Prehľad rôznych typov identifikačných štruktúr

- po natrénovaní NN v štruktúre GT predstavuje inverziu k danému procesu, t.j. **inverzný neurónový model**, ktorý môže byť v spojení s klasickým regulátorom použitý pre kompenzáciu nelinearít

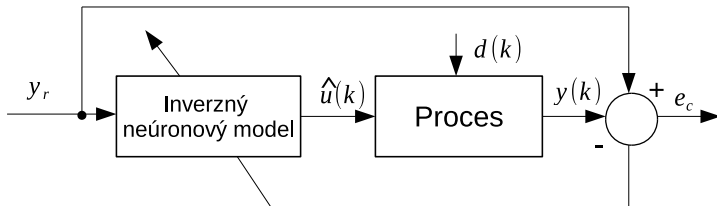
Nevýhody štruktúry GT:

- výber rôznych hodnôt vstupného signálu u pre natrénovanie nezaistí, aby sa výstup procesu y nachádzal po natrénovaní v tých oblastiach, ktoré sú zaujímavé pre úspešnú aplikáciu v riadení
- ak riadený systém je mnohorozmerný, získaný identifikovaný model nemusí odpovedať reálnemu procesu. (nepoužiteľné pre procesy s integračným charakterom)

9.3 Prehľad rôznych typov identifikačných štruktúr

2. Architektúra SPECIALIZED TRAINING

- v architektúre ST je NN (MPL) tréňovaná ako inverzný model procesu učiacim signálom e_c (táto štruktúra eliminuje nedostatky GT)
- zo štruktúry ST vyplýva, že hodnota riadiacej veličiny $r(k)$ predstavujúca požadovaný výstup procesu tvorí vstup do NN, ktorá je tréňovaná tak, aby na svojom výstupe generovala taký signál \hat{u} , ktorý zabezpečí, že chyba e_c medzi aktuálnym výstupom procesu y a požadovaným výstupom $r \approx y_r$ bola minimálna



9.3 Prehľad rôznych typov identifikačných štruktúr

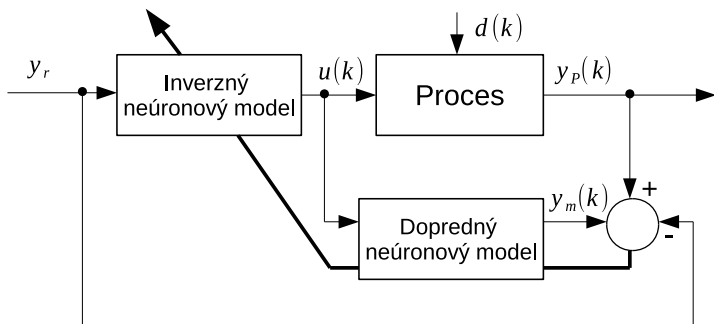
- Pri tejto metóde ST je tréningovým signálom na získanie INM procesu chyba e_c

V porovnaní s GT má ST tieto výhody:

- metóda je určená priamo na riadenie (tréningový signál $e_c = y_r - y$)
- v prípade mnohorozmerného DS je identifikovaný skutočný inverzný model (jednoznačnosť úlohy)

9.3 Prehľad rôznych typov identifikačných štruktúr

3. Aplikácia naučeného NN modelu pri inverznom modelovaní



Spätné šírenie chyby cez dopredný neurálny model procesu na adaptáciu NN pre získanie inverzného modelu procesu

$$e_c = y_r - y_p, \quad e_u = N(k)e_c$$

$N(k)$ - vypočítame z dopredného modelu procesu

9.3 Prehľad rôznych typov identifikačných štruktúr

- najprv použijeme dopredný neurónový model na identifikáciu nelineárneho procesu (sériovo-paralelný model)
- dopredný neurónový model procesu je trénovaný použitím $e = y_p - y_m$, y_p - výstup procesu, y_m - výstup dopredného neurónového modelu
- inverzný neurónový model procesu je trénovaný použitím $e_c = y_r - y_p$, y_r - požadované chovanie procesu, y_m - aktuálne chovanie procesu, ktorá sa použije na výpočet chyby $e_u = N(k)e_c$, (e_u sa spätne šíri cez inverzný neurónový model)
- schéma ilustruje spôsob, v ktorom sa chyba e_c spätne šíri cez neurálny dopredný model a je použitá pomocou Jacobiánu procesu $N(k)$ na výpočet chyby e_u , ktorý trénuje neurónový inverzný model