1869

State-Space Fuzzy-Neural Network for Modeling of Nonlinear Dynamics

Yancho Todorov, Margarita Terziyska

yancho.todorov@iit.bas.bg, terziyska@dir.bg

*Institute of Information and Communication Technologies,
Bulgarian Academy of Science, Bulgaria*

Summary

- **Why State-Space Fuzzy-Neural approach?**
- **Improved Takagi-Sugeno FN State-Space Neural Network.**
- **Training method for the proposed FN SS Neural Network.**
- **Simulation Experiments.**
- **Conclusions.**

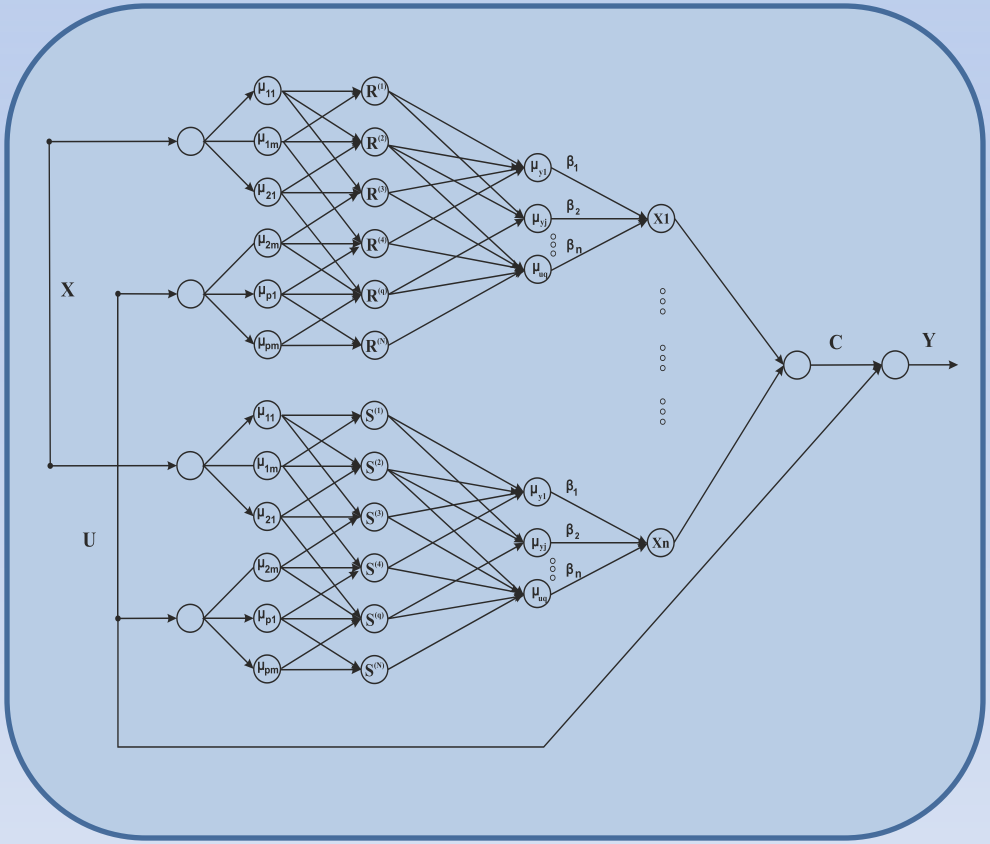
Why State-Space Fuzzy-Neural approach?

- State-Space approach involves less parameters identification compared to time domain system representation.
- Combining the State-Space description with simple TS fuzzy-neural approach, enables the possibility for nonlinear system modeling by using multiple model weighting.
- Using Fuzzy-Neural State-Space representation facilitate the further development of constrained optimization polices in purpose of Model Predictive Control.

Improved TS State-Space Fuzzy-Neural Network.

- **Defining a proper structure of the SS FFN is a crucial issue, not only for the model prediction capabilities but it affects the computational feasibility of the optimization task in MPC control.**
- **Our previous works show that the use of the standard TS approach in State-Space is inappropriate, due difficulties to coordinate the learning procedures in notion to different error terms, which may require a multiparametric optimization.**
- **Solution: We propose a simple hierarchical model structure which enables the possibility to avoid many optimization obstacles preserving the general State-Space notation in a multiple model manner.**

Structure of the proposed SS FNN



- ✓ Each parallel structure models one system state as a set of local linear models.
- ✓ The model output depends on those predicted system states - X and the direct system input - U.
- ✓ Thus the parameters of each parallel structure in notion to the respective state are adjusted using its error state term.
- ✓ The parameters associated with the output – Y are adjusted using the output error term.

Description of the SS FNN model

Mathematical description of the model

$$\begin{cases} \hat{x}_1(k+1) & = f_{x_1}(\hat{x}_1(k),.....\hat{x}_n(k),\mathbf{u}(k)) \\ \hat{x}_2(k+1) & = f_{x_2}(\hat{x}_1(k),.....\hat{x}_n(k),\mathbf{u}(k)) \\ \vdots & \vdots \\ \hat{x}_n(k+1) & = f_{x_n}(\hat{x}_1(k),.....\hat{x}_n(k),\mathbf{u}(k)) \end{cases}$$

where x$_i$ *is a predicted i-th system state, by fuzzyfication of its previous discter instance value and the actual system input*

$$R^{(i)} : if\ r_1(k)\, is\ M_1^{(i)} ...and\ ...r_p(k)\, is\ M_p^{(i)}$$

$$then \quad \left| \hat{x}_n(k+1) = A\hat{\mathbf{x}}(k) + B\mathbf{u}(k) \right|^i$$

where R is the i^{th} rule of the local rule base, r_p are the state regressors (the states and the output of the system), M_i is a membership function of a fuzzy set and $A^{(i)}$, $B^{(i)}$, are matrices in notion to i^{-th} fuzzy rule

Description of the SS FNN model

Mathematical description of model

From a given input vector, the output of the each fuzzy model is inferred by computing the following equation:

$$\hat{\mathbf{x}}(k+1) = \mathbf{f_x}^{(i)} \mathbf{g_{ui}}^{(i)} \text{where } \mathbf{g_{ui}} = \prod_{i=1}^{N} \boldsymbol{\mu}_{\mathbf{ui}}$$

where μ_{ui} are the degrees of fulfillment in notion to *i-th* activated fuzzy Gaussian membership function, defined as:

$$\mu_{r_{p,m}}^{(n)} = \exp \frac{-(r_p - c_{r_{p,m}})^2}{2\sigma_{r_{p,m}}^{\ 2}}$$

A common practical case is to define a system by second order model in the form:

$$\dot{x} = \begin{vmatrix} \dot{x}_1 \\ \dot{x}_2 \end{vmatrix}, \quad A = \begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix}, \quad B = \begin{vmatrix} b_1 \\ b_2 \end{vmatrix},$$

$$C = \begin{vmatrix} c_1 & c_2 \end{vmatrix}, \quad D = \begin{vmatrix} d \end{vmatrix}$$

$$\left| \begin{array}{l} \hat{x}_1(k+1) = \sum_{i=1}^{N} g_{ui_l}(a_{11}\hat{x}_1(k) + a_{12}\hat{x}_2(k) + b_1 u(k)) \\ \hat{x}_2(k+1) = \sum_{i=1}^{P} g_{ui_l}(a_{11}\hat{x}_1(k) + a_{12}\hat{x}_2(k) + b_2 u(k)) \\ y(k) = c_{11}\hat{x}_1(k) + c_{12}\hat{x}_2(k) + du(k) \end{array} \right.$$

Learning algorithm for the SS FNN

- The task of model identification is to determine both groups of parameters of the Gaussian membership functions in the rule premise part and the linear parameters (coefficients) in the rule consequent part of the local models.
- The learning algorithm for each state associated fuzzy-neural model is based on minimization of an instant error measurement function: *E=ε²/2*, between the real *x(k)* and the estimated by the fuzzy-neural model system state .

The general parameter learning rule for the consequent parameters is:

$$\beta_{si}(k+1)=\beta_{si}(k)+\eta\left(\frac{\partial E}{\partial \beta_{si}}\right)$$

where the gradient is calculated by a defined chain rule:

$$\frac{\partial E}{\partial \beta_{ij}}=\frac{\partial E}{\partial \hat{x}}\frac{\partial \hat{x}}{\partial f_x}\frac{\partial f_x}{\partial \beta_{si}}$$

The final recurrent predictions for each adjustable coefficient β_{ij} ($a^{(i)}$ or $b^{(i)}$) are obtained by the following equation:

$$\beta_{si}(k+1)=\beta_{si}(k)+\eta(x(k)-\hat{x}(k))\overline{g}_{ui_l}{}^{(j)}(k)r_p(k)$$

The rule premise parameter scheduling is achieved by:

$$\alpha_{si}(k+1)=\alpha_{si}(k)+\eta\left(\frac{\partial E}{\partial \alpha_{si}}\right)$$

where the gradient is being obtained by the following chain rule:

$$\frac{\partial E}{\partial \alpha_{pi}}=\frac{\partial E}{\partial \hat{x}}\frac{\partial \hat{x}}{\partial \mu_{pi}}\frac{\partial \mu_{pi}}{\partial \alpha_{pi}}$$

$$\sigma_{pi}(k+1)=\sigma_{pj}(k)+\eta\varepsilon(k)\overline{g}_{ui}(k)[f_x^{(i)}(k)-\hat{x}(k)]\frac{[r_p(k)-c_{pi}]^2}{\sigma_{pi}{}^3(k)}$$

$$c_{pi}(k+1)=c_{pi}(k)+\eta\varepsilon(k)\overline{g}_{ui_l}(k)[f_x^{(i)}(k)-\hat{x}(k)]\frac{[r_p(k)-c_{pi}]}{\sigma_{pi}{}^2(k)}$$

Adaptive learning rate scheduling

- In order to overcome the deficiencies of the Gradient Descent approach, a simple adaptive solution to define at each iteration step the learning rate *η,* has been employed*.* The idea lies on the estimation of the *Root Squared Error*:

$$\mathrm{E} = \sqrt{\sum_{k=1}^{M} (x(k) - \hat{x}(k))^2}$$

- Afterwards, the following condition is applied:

$$\begin{aligned} &\textit{if } \mathrm{E}_i > \mathrm{E}_{i-1} k_w \\ &\qquad \eta_{i+1} = \eta_i \tau_d \\ &\textit{if } \mathrm{E}_i \le \mathrm{E}_{i-1} k_w \\ &\qquad \eta_{i+1} = \eta_i \tau_i \end{aligned}$$

where τ_d =0.7 and τ_i=1.05 are scaling factors and k_w=1.41 is the coefficient of admissible error accumulation

Validation of the proposed SS FNN

Computer simulated pendulum system

An oscillation pendulum system model is used to test the modeling capabilities of the proposed State-Space Fuzzy-Neural Network.

A rigid zero-mass pole with length *L* connects a pendulum ball and a frictionless pivot at the ceiling. The mass of the pendulum ball is *M*, and its size can be omitted with respect to *L*.

The pole (together with the ball) can rotate around the pivot, against the friction *f* from the air to the ball, which can be simply quantified as

$$f = -sign(v)Kv^2$$

$$\ddot{\theta} = \frac{F}{ML}\cos\theta - \frac{g}{L}\sin\theta - sign(\dot{\theta})\frac{KL}{M}\dot{\theta}^2$$

Using two state variables x_1, x_2 to represent the position and the velocity, a State-Space representation is being obtained:

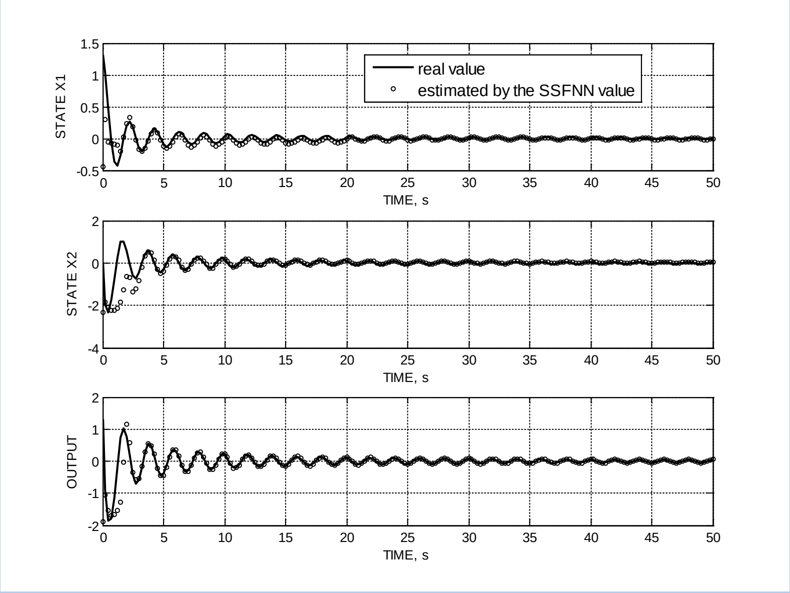
$$\mathbf{X} = (x_1,\ x_2)^T = (\theta,\ \dot{\theta})^T,\ \mathbf{U}=(\mathrm{F})$$

$$\dot{\mathbf{X}} = \begin{pmatrix} 0 & 1 \\ -g\sin x_1 / x_1 & -|x_2| \end{pmatrix}\mathbf{X} + \begin{pmatrix} 0 \\ \cos x_1 \end{pmatrix}\mathbf{U}$$

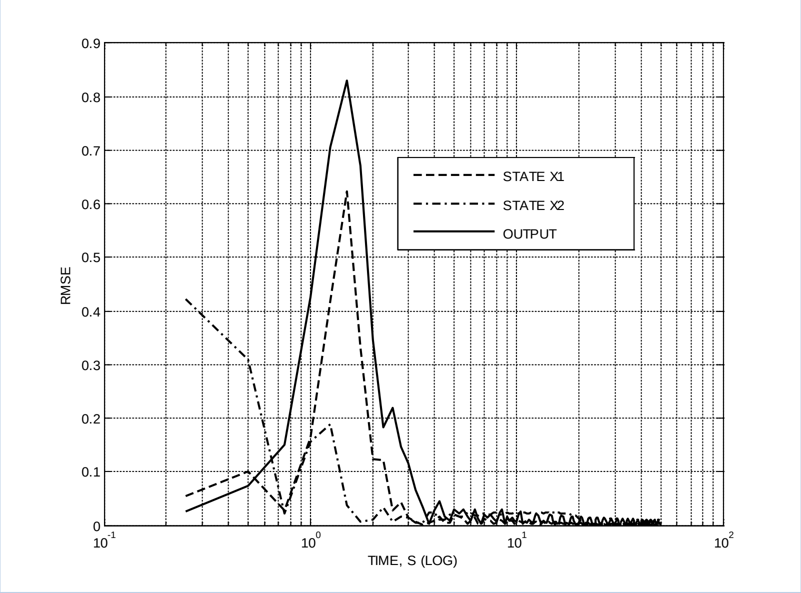
Applying Runge-Kutta method to (19), we can get the 'continuous' states of the testing system. The input (***U***) and states (***X***) are sampled every 0.25 second and for total 50 seconds, using the following conditions (F=0, $x_1(0)=5\pi/12$, $x_2(0)=0$).

Validation of the proposed SS FNN

Computer simulated pendulum system



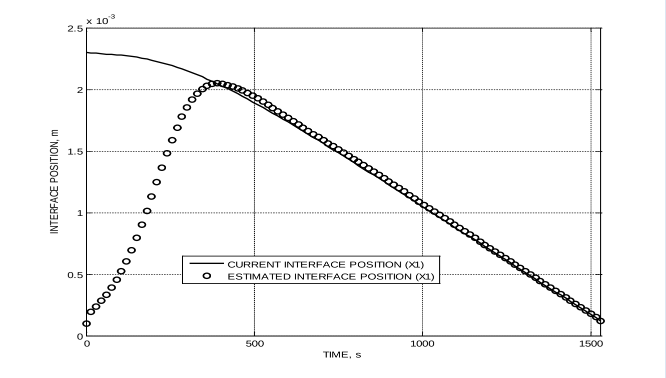
Estimation of the states and the output of the Pendulum Oscillating

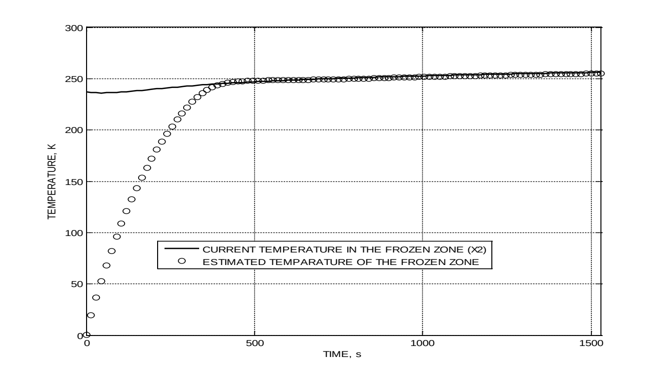


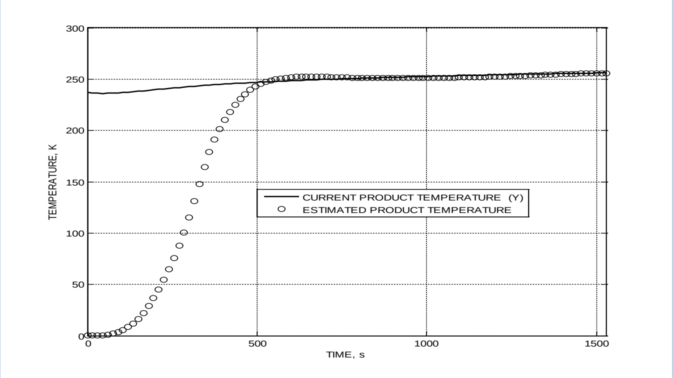
Estimation of Root Mean Squared Errors of the estimations.(In logaritmic scale)

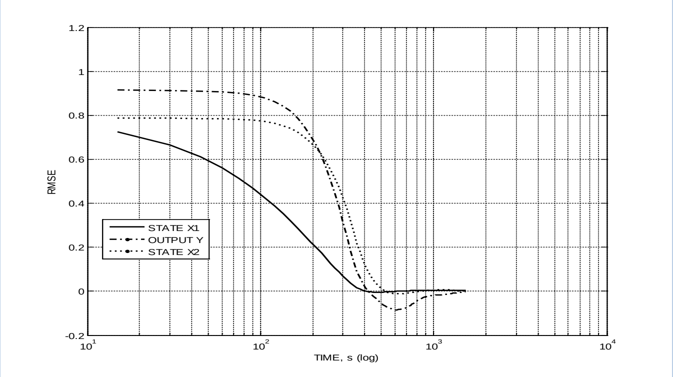
Validation of the proposed SS FNN

Computer simulated nonlinear Lyophilization plant









Estimation of the states and the output of the Lyophilization plant

Estimation of Output and the Root Mean Squared Errors (in logaritmic scale)

Conclusions

- It was shown the development of a novel structure of State-Space Fuzzy-Neural Network model with parallel units for states estimation.
- The numerical validation in modeling of two nonlinear processes (Oscillating pendulum and Lyophilization plant) have shown a good ability of the model to adapt accurately on different system dynamics (faster or slower).
- The obtained predictions of the system states and outputs are achieved with minimal error as demonstrated.
- An extension of the approach is the inclusion of the model in a Constrained Model Predictive Control scheme using the QP optimization strategy.

Aknowlegments

The research work reported in the paper is partly supported by the project ***AComIn*** "Advanced Computing for Innovation", grant 316087, funded by the FP7 Capacity Programme (Research Potential of Convergence Regions).

THANK YOU
FOR YOUR ATTENTION!