On the Design of the DISO Controllers for a Neuro-Fuzzy Vector Control. The training data processing and some results.

Dan S. Mihai
Department of Electrical Drives and Industrial Informatics
University of Craiova
Blvd. Decebal 107, 200 440, Craiova, Romania
E-mail: dmihai@em.ucv.ro

KEYWORDS

Neuro-fuzzy Controller, Vector Control, Electrical Drives.

After some good previous results with neuro-fuzzy controllers SISO type for the speed or for the current loops in a vector control system, now the design is orientated to the DISO units. Also, the coexistence of both controllers type as neuro-fuzzy units are considered: the speed controller - with a slow bias and the current - with a high commutation rate for driving directly the inverter. The aim is also to provide a performing and integral intelligent-based solution for applications were the mechanical shock must be controlled (such as trams-trains, personal elevators). The design of such controllers is focused on three aspects: the conditions of a good data acquisition for training by simulation of a well tuned standard system, the data pre-processing for the training of the neural networks and the tuning of the synthesized controllers. The ANFIS methods is used for generating Sugeno fuzzy controllers. Different design details and tuning procedures are taken into account. The paper is continuing a previous work.

INTRODUCTION

Neural networks and fuzzy systems are different approaches to introducing humanlike reasoning to knowledge-based intelligent systems. The integration of fuzzy logic and neural networks seems natural and full of benefits (Kasabov, 1998). Artificial Neural Network (ANN) learns from scratch by adjusting the interconnections between layers. Fuzzy Inference System (FIS) is a computing framework based on the concept of fuzzy set theory, fuzzy if-then rules, and fuzzy reasoning. Integrating ANN and FIS have attracted the growing interest of researchers due to the growing need of adaptive intelligent systems to meet the real world requirements. (Nauk, 1995) proposes a taxonomy to describe different combinations of neural networks and fuzzy systems by Fuzzy Neural Networks, Concurrent Neuro - Fuzzy Models, Cooperative Neuro - Fuzzy Models, Hybrid Neuro - Fuzzy Models. An example for the last architecture is ANFIS (Adaptive-Network-Based Fuzzy Inference System) - (Jang, 1993), (Jantzen, 1998). The first applications of fuzzy neural networks to consumer products appeared on the Japanese and Korean market in 1991. Some examples include (Fuller, 1995): air conditioners, electric carpets, electric fans, electric thermo-pots, desk-type electric heaters, forced-flue kerosene fan heaters, kerosene fan heaters, microwave ovens, refrigerators, rice cookers, vacuum cleaner, washing machines, clothes dryers, photocopying machines, and word processors. (Vieiraa, 2004) provides a comparison of artificial neural networks and neuro-fuzzy systems applied for modeling and controlling a real system. Most of studies were made in the field of the modeling / identification / estimation / fault diagnosis for the mentioned system – (Woei, 2005). Fewer papers are allowed for a VC strategy combined with fuzzy or neuro-fuzzy controller, and only some samples concern the current loop.

For designing neuro-fuzzy controllers (NFC), an initial standard vector control structure (rotor flux variant) was considered – (Doumbia, 1997). In some previous works – (Mihai 2006), (Mihai 2007), the author performed the synthesis of SISO neuro-fuzzy (N-F) controllers that could replace the classical controllers (PI and hysteresis type for speed, respectively for current) with several benefits. Unlike other studies and solutions that implement a Mamdani controller, it is about Sugeno type controllers generated by the ANFIS (Jang, 1993) program. But the author found that the designer must be advised on several aspect concerning the best operating conditions of the system for data acquisition, the training data collection preparing and the results tuning. Continuing the paper (Mihai, 2009), several designing details for these aspects are followed both for the speed and current NFCs, in the DISO variant (considering the error variation of the variables). The results are selected from a wide range of operating conditions and induction motors taken into account (in the power range 2–37 kW, having as target application the transportation systems with low mechanical shock).

THE TRAINING DATA PROCESSING

The study carried (Mihai, 2009) has revealed that the expected (from the cycle type) non-monotonical evolution of the speed is completed by more or less powerful ringing phenomena that could have several explanations. It is unavoidable a commutations between the generated fuzzy rules (sets) as well as the commutations effect. However, every effort paid in order to reduce such oscillation should improve the overall behavior of the system. And the next (natural) idea is to tray for a good (declared) data training set, some additional processing like:

a. the reduction of the samples for increasing the speed of the training procedure (for a set of 3 x 110,000 data

Proceedings 23rd European Conference on Modelling and Simulation ©ECMS Javier Otamendi, Andrzej Bargiela, José Luis Montes, Luis Miguel Doncel Pedrera (Editors)
ISBN: 978-0-9553018-8-9 / ISBN: 978-0-9553018-9-6 (CD)

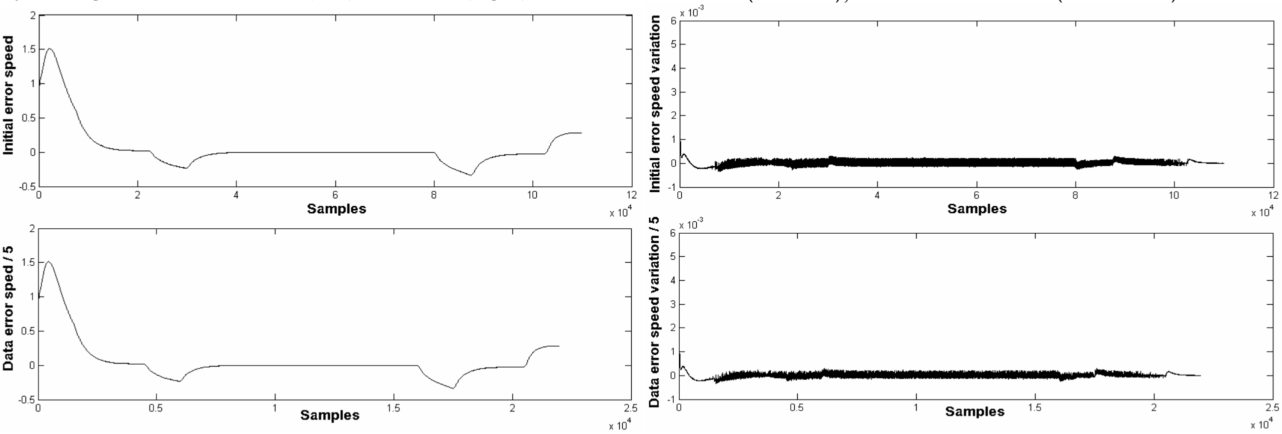
and a standard 2 GHz PC, this duration can reach many hours..);

b. a filtering of the samples, so that some discrete noises could be eliminated.

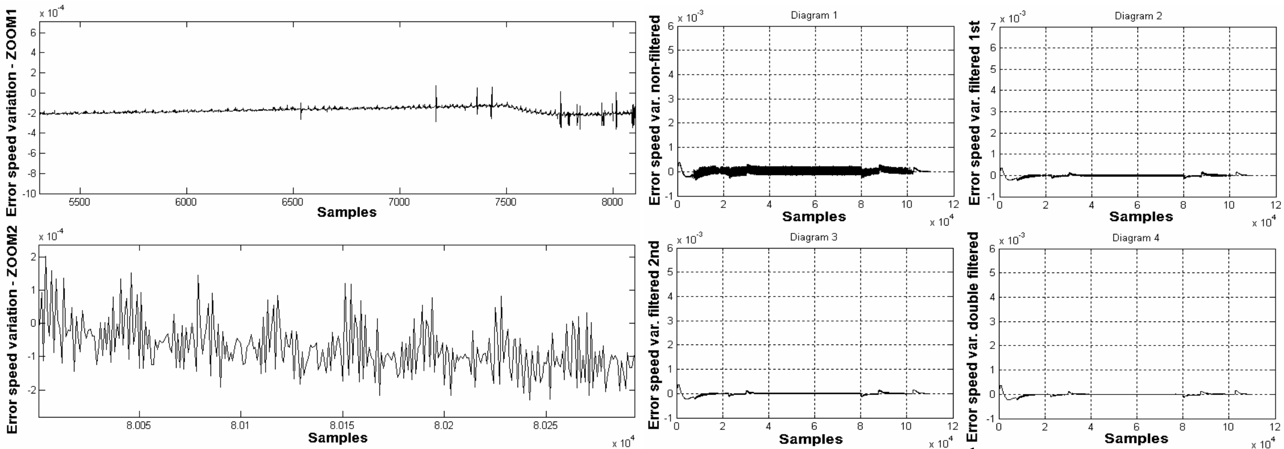
For the first step, a program was made for selecting from a N component vector V another one V_s, taking only the m^{th} samples; the new vector has N_m components, as follows:

$$V_S(i) = V(m \cdot i - m - 1)\,;$$
$$i = 1 \ldots N_m = \left[\frac{N}{m} - \varepsilon\right] + 1\,; \quad \varepsilon < \frac{1}{m} \qquad (1)$$

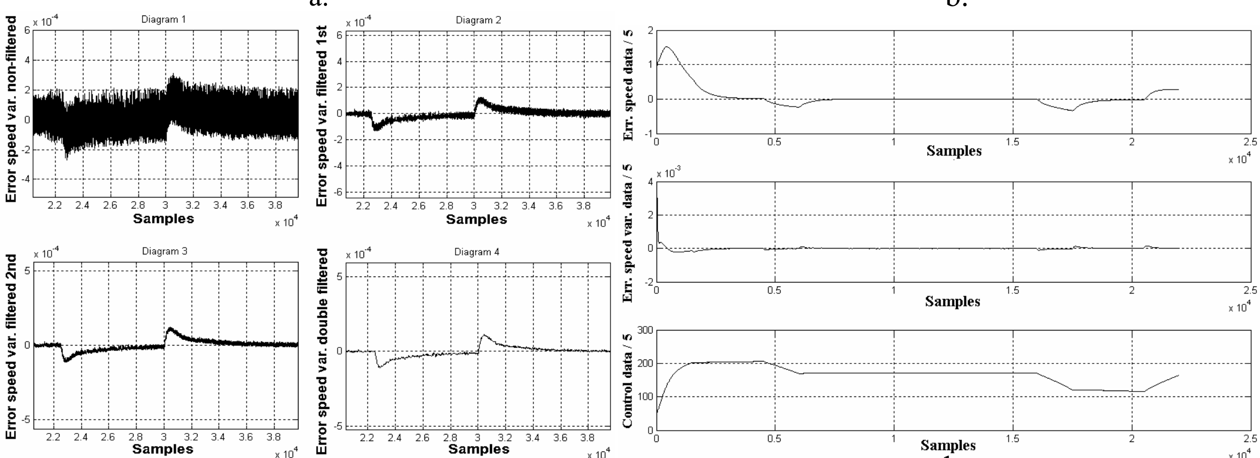
A comparison image of the initial training data (speed loop, 110,000 sets) and the selected ones (m=5) is given by the fig. 1, both for the ε_ω (left) and $\Delta\,\varepsilon_\omega$ (right). Some small differences are visible only for $\Delta\,\varepsilon_\omega$ For a bigger m (like 10), the samples of $\Delta\,\varepsilon_\omega$ become spaced out in an obvious manner. As for the second action (the data filtering), it has a bigger role for $\Delta\,\varepsilon_\omega$, where the high rate commutations could be in a direct relation with the control ringing. The question is if the filtering does not modify the useful data substance. Because the second NFC speed controller input was added as a supplementary one in order to improve the training process, what is important is the bias of this variable and not the high rate commutations and the local big slopes. In this meaning, the filtering could be useful, indeed. The fig. 2 b and c present the filtered results using a Savitzky-Golay (polynomial) FIR smoothing filter (filter 1); the other filter (denoted 2) is a fist



Figures 1: A set of initial trainig data and the reduced set by a factor of 5.



a. b.



c. d.

Figures 2: Zoomed images of $\Delta\,\varepsilon_\omega$ and the effects of a single or multiple filter.

order one. Both filters have been tuned by several tests. The aspect of the training data in these conditions (after a selection procedure and a double filtering) is given by the fig. 2d. The fig. 3 certifies a good quality of the new training data set by the similarity between the trajectory of the initial system and the trajectory associated with the data generated for the speed NFC. A direct comparison between the control variable from the initial PI speed controller and the fuzzy controller delivered by training the neural network is presented by the fig. 4.

Now, joining the experience for the previous stage, a good (better) global behavior of the system with the speed NFC is expected. The fig. 5 brings the characteristic elements (fuzzy sets, graphical correlations between the variables) for a tuned speed NFC in the next conditions:

- 1 operation quadrant;
- selected and double filtered training data;
- saturation of the output for the speed controller;
- 7 fuzzy sets, 100 training epochs, max-prod-wtaver operators for the Sugeno NF controller;

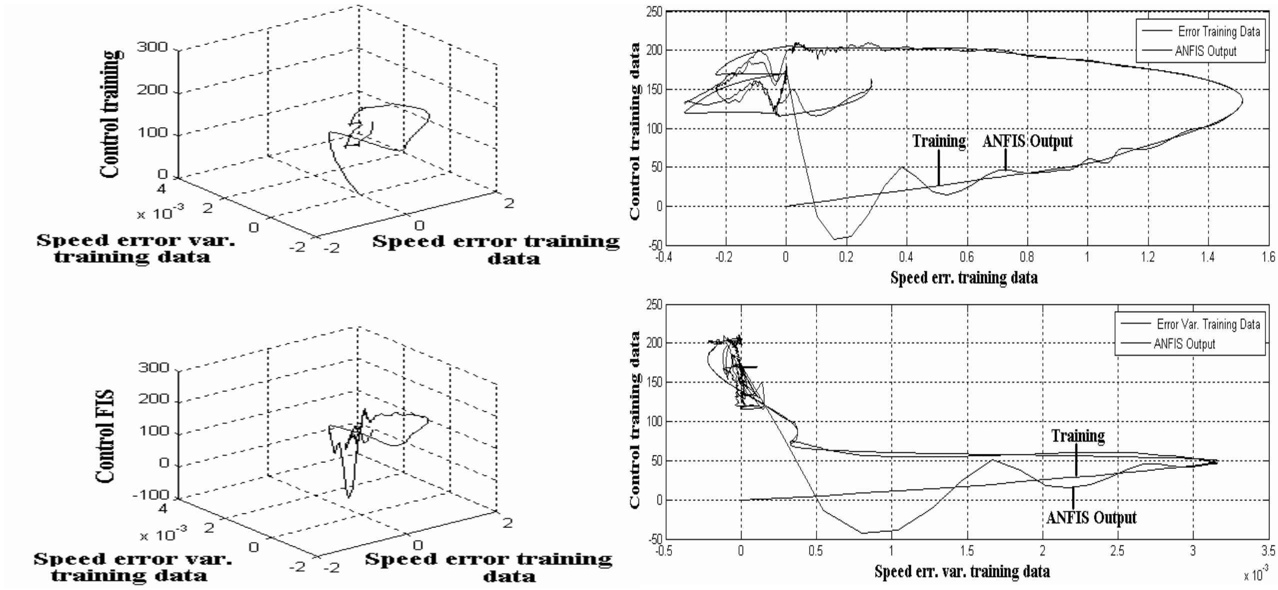
A supplementary tuning of the generated fuzzy controller was made in order to reduce some spikes of the electromagnetic torque (although with minor effects on the final speed evolution). This tuning is based on the supposition that this phenomenon could be related with (generated by) some big slopes in the inputs of the FIS controller. Such zones are identified in the projection speed error - output and the fuzzy sets are slightly modified following the effect on the slope attenuation. The fuzzy sets for the speed error variation have an unusual form but the dependence output ($\Delta \varepsilon_{\omega}$) became, obviously, a smooth one.

A global results for the system with the new speed NFC is shown by the fig. 6a and the quality of the controller is certified by the fig. 6b, where a different (much shorter than in the training step) speed cycle was imposed.

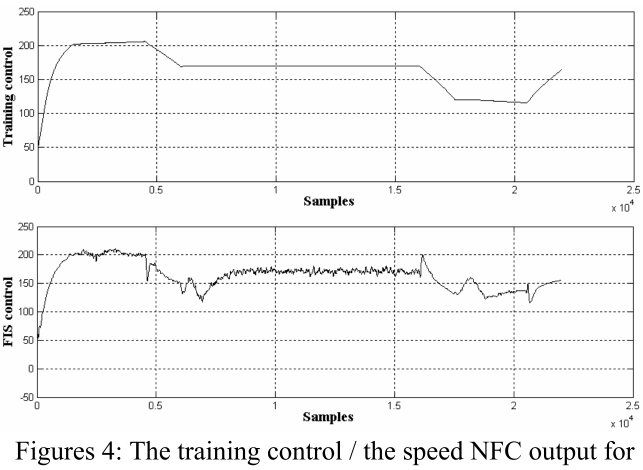
The same speed NFC is tested for different operation conditions (simultaneously applied) than those from the training step:

- a step speed input;
- a much higher steady-state speed value;
- an inertia bigger with 50 %;
- the motor resistance bigger with 25 % (like for a heating effect);
- a random variation of the load torque around the training values.

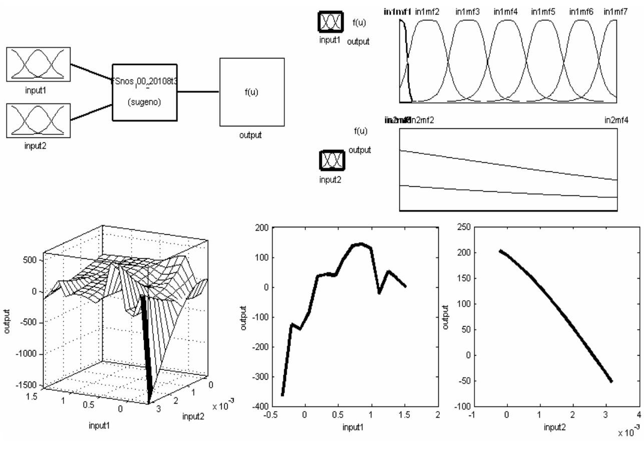
The results from the fig. 7b could give, again, an excellent mark for the NF controller, compared with the initial PI speed controller (fig. 7.a) - considering that the training of the NF solution was made in very different



Figures 3: The state-space trajectories for the selected and filtered data and for the generated speed NFC.



Figures 4: The training control / the speed NFC output for selected and filetered data: 22001 data, 100 epochs.



Figures 5: The fuzzy description of the speed controller.

operating conditions. Other test was made with no load torque (only a friction component, depending on speed); the results had the same quality.

3. THE SYNTHESIS OF THE CURRENT NFC.

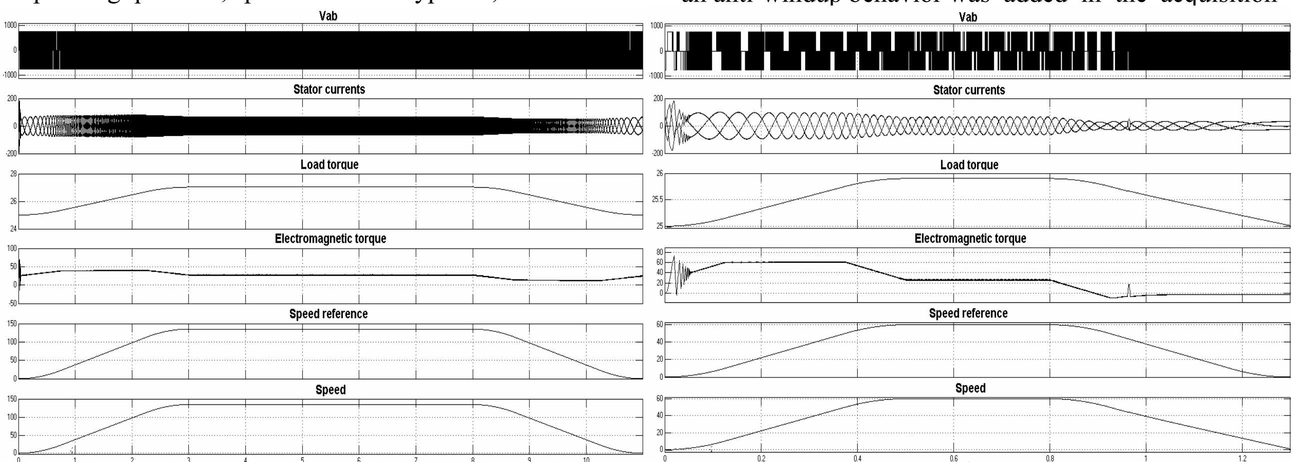
The substitution of the standard current controllers (hysteretic type) that deliver directly the firing pulses to the inverter by NFC units is very challenging by several reasons:

- the high rate commutations (with sudden rise and fall edges) that could be not quite appropriate for the "continuous " approach specific to the fuzzy logic;
- the training data does not obey a bias that could characterize the IN-OUT functional relation;
- the sampling is much higher than for the current.

However, the author has succeeded to obtain a viable current SISO NFC in a system having a classical speed controller (Mihai, 2007), designing a special block for the training data acquisition (and, later, for the pulse generation by the NFC). The fig, 8 gives the image of such a block for 3 DISO current NFC, one for each phase. The aim of the study at this level is to answer and overcome the next problems:

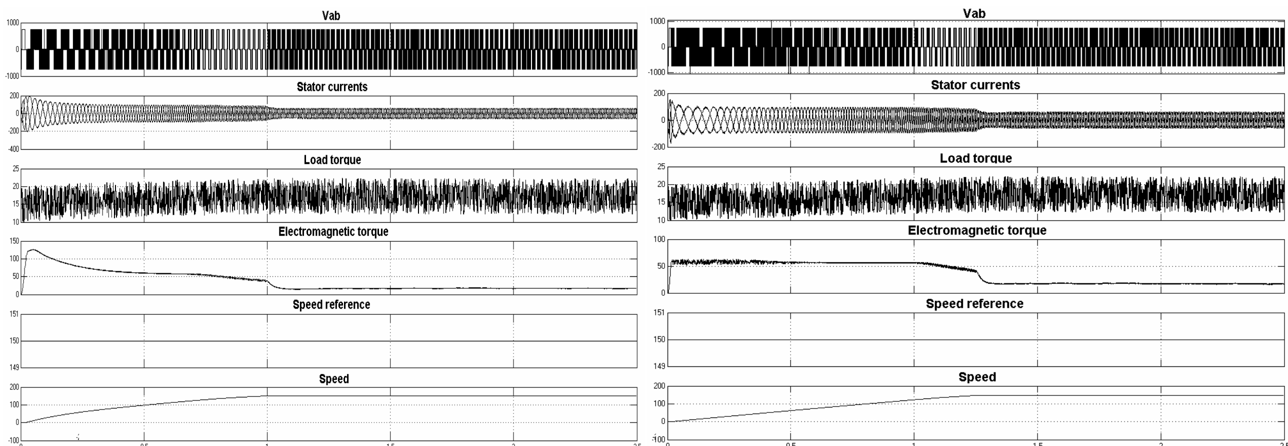
- which is the image of the current error variation and how important is to manage this variable;
- which are the best operation conditions for the data training sampling: initial tuning of the current loops, operating quadrant, speed reference type etc;
- if it is possible to avoid some big spikes (commutations) on the power part, due, mainly, to the extrapolation type computations that the training of the NN brings especially in the edge regions of the variables range;
- if the reduction of these spikes is a matter of the training step or is related with a post-tuning procedure;
- which is the best sampling rate for the training data;
- how long must be the training time interval?
- if a values selection and filtering is useful for the current too;

According to the simulations of the initial model (and the one having a speed NFC) gives a PWM commutation frequency under 25 kHz. Then, a 30 μs sampling rate seems good. A shorter (than for the speed controller synthesis) cycle of 5 sec leads to 166,000 samples for each phase. The acquisition block from fig. 8 stores both saturated and non-saturated variables - (Mihai, 2007). The fig. 9 presents the samples c_k, c_{k-1}, Δc_k non-saturated (the upper windows) and the same saturated variables (lower windows), justifying the necessity of the saturation procedure. There is, always, a high first values zone that makes the controller insensible to the common operating values during the most part of the regime. As a result, the obtained fuzzy controller is not able to realize a (suitable) commutation. That is why an additional bloc (D-sat) for an anti-windup behavior was added in the acquisition



a. b.

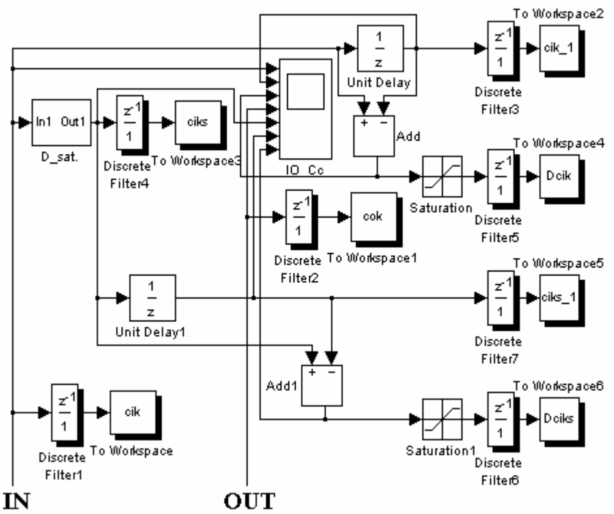
Figures 6. Some global results with the non-implicit speed NFC.



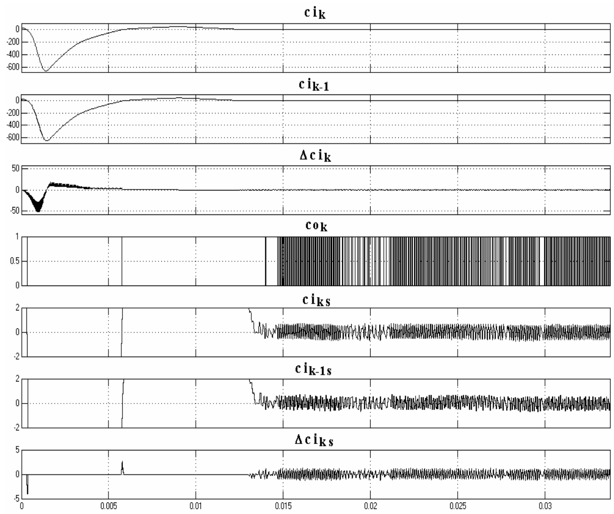
a. b.

Figures 7: A comparison between standard and NF speed controller.

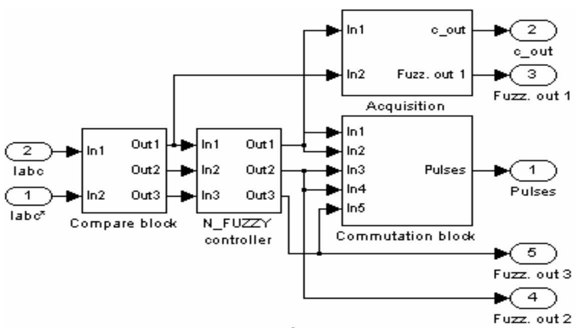
model, inserted also, later, in front of the final fuzzy controller (Mihai, 2007). Now, the FLC output is variable but not in a proper manner for the pulses needed by the inverter. A specific block must be added also after the fuzzy controller in order to realize its triggering 0 – 1 (fig. 10).



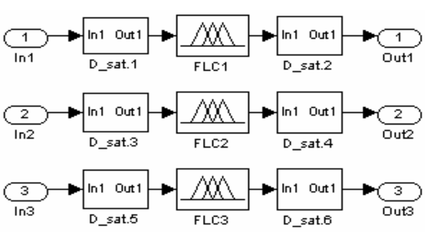
Figures 8: Acquisition block for the current NFC training.



Figures 9: A local image of the current training data.



a.



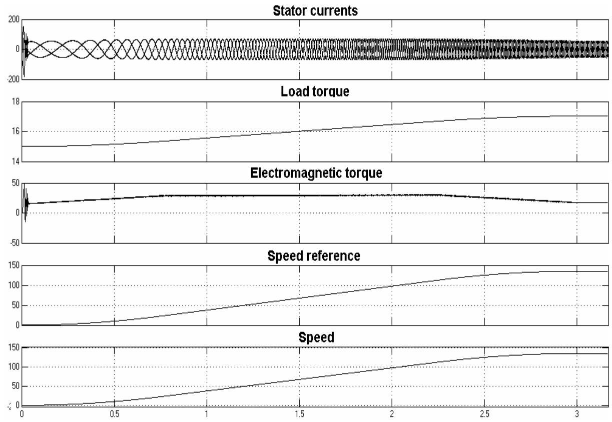
b.

Figures 10: The current NFCs with its elements.

A comparison between the initial commutations of the current controller and those after a selection procedures of samples (division by 5) shows that this kind of data preparing could totally change the commutation regime of the inverter:

cok;	coks div 5
1	1
1	0
0	1
1	0
1	1
0	0
1	0
1	0

A test with the tuned speed NFC inserted in a system having the current NFCc was made and the results are encouraging - fig. 11. More, the "coexistence" of both NFCs (speed and 3 x current) seems not having some bad effects on the input-output variable evolution for the speed controller - see fig. 12a (only the speed controller is a NF type) and b (both speed and current controllers are NF type).



Figures 11: The results for a system with an optimized DISO speed NFC and SISO current NFCs (no shock operation).

CONCLUSIONS

The author was interested in obtaining a robust vector control for asynchronous motors by means of neuro-fuzzy controllers both for the speed and the current loops. The interesting aspects of the research (and the challenge too) concern the passing from a simple controller (with a few or even a single tuning parameter) to a much more complex controller, the neuro-fuzzy units having, practically, an infinite tuning parameters. Although are available methods and programs able to deliver neuro-fuzzy controller based on training input - output data, the designer must pay a special attention to some important aspects: the most relevant operating conditions / regimes for the training data acquisition, the pre-processing of these data (by selection, filtering), the training parameters (epochs number, fuzzy sets as number and type etc and, finally, a post-tuning of the neuro-fuzzy controller. A double-input-single output neuro-fuzzy controller is able to bring better results for the system, in term of a solid robustness for different operating conditions. The design of the current neuro-fuzzy controller is much more complex because of the request to having the output in commutation for driving the power inverter. It is important to mention that the synthesis of a neuro-fuzzy controller operating in commutation mode is possible but requires additional blocks and must be made with lot of care. The tuning of the N-F controllers consists, in fact, in the choice of the fuzzy sets number, their distribution, the operator type and the selected defuzzification method. Some tests with several values for the epochs number proved a minor or no visible influence on the results in term of the training method error. Their number, however, could influence directly the surface control surface and, by that, some accuracy details of the system behavior. The influence of the AND/OR operators was not detectable in the shape of the surface control. The most important difference in the results validity comes from the IN-OUT training data that have a major influence on the distribution of the fuzzy sets; very seldom they have an uniform distribution. Also, different motor parameter sets generate N-F controller having very distinctive characteristics.

REFERENCES

Abraham A. 2001. It is time to Fuzzify Neural Networks!, Tutorial, *ICIMADE*, 1-3 Jun, Fargo, USA, pp 253 - 272.

Altrock C. von. 1995. *Fuzzy Logic and NeuroFuzzy Applications Explained*, Prentice Hall PTR.

Bose B.K. 2000. "Fuzzy logic and neural networks in power electronics and drives"*, IEEE Ind. Applic. Magazine*, pp.57 –63, Vol.6, May/Jun.

Doumbia M. L.; Roy G. and Rajagopalan V. 1997. "An Integrated Solution for Simulating Electrical Drive Systems with Matlab / Simulink", *Int. Symp. on Ind. Electronics*.

Fuller R. 1995. *Neural Fuzzy Systems*, Abo Akademi University, Abo.

Jantzen J. 1998. "Neurofuzzy Modelling", Technical Univ. of Denmark, Depart. of Autom., Lyngby, DENMARK, Tech. report no 98-H-874.

Kasabov N.K. 1998. *Foundations of Neural Networks, Fuzzy Systems, and Knowledge Engineering*, A Bradford Book, The MIT Press, Cambridge, Massachusetts.

Lin F.C. and Yang S.M. 2003. "On-line Tuning of an Efficiency–Optimized VectorControlled Induction Motor Drive", *Tamkang Journal of Science and Engineering,* Vol. 6, No. 2, pp. 103-110.

Mihai D. 2006, "Using a Hidden Mathematical Model Generated by Input-Output Data. The Reliability of a Neuro-Fuzzy Controller for a Servodrive", *Proc. of the 8th WSEAS Int. Conf. on Math. Methods and Comput. Techniques in Electrical Eng.*, Bucharest - Romania, paper 518-150, pp. 288-283.

Mihai D. and Mihai R. 2007. "Neuro-Fuzzy Versus Hysteresis Current Controller for a Vector Control Based Electrical Drive System"*, 11th IEEE International Conference on Intelligent Engineering Systems*, 29 June – 1 Jul, Budapest, Hungary, pp 241 – 246.

Mihai D.. 2009. "On the Design of the DISO Controllers for a Neuro-Fuzzy Vector Control. The training data acquisition", *23rd European on Conference on Modelling and Simulation*, Madrid, June.

Nauck D. 1995. "Beyond Neuro-Fuzzy: Perspectives and Directions", *Third Europ. Congr. on Intell. Techn. and Soft Comput., (EUFIT*'), Aachen, pp. 1159 -1164.

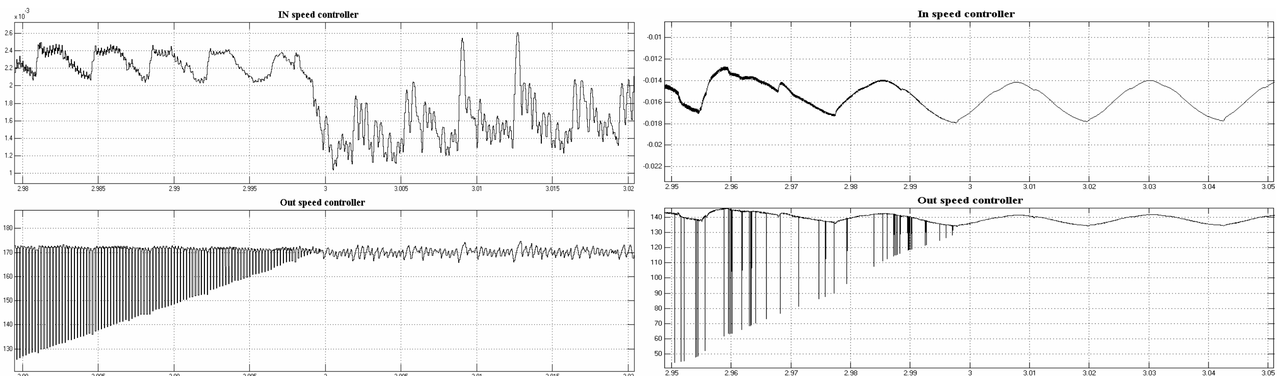
Vieiraa J.; Diasb F. M. and Mota A. 2004. "Artificial neural networks and neuro-fuzzy systems for modelling and controlling real systems: a comparative study", *Eng. Appl. of Artif. Intell.* 17, pp 265–273.

Wishart M. T. and Harley R. G. 1993. "Identification and control of an induction machine using artificial neural networks", *Conf. Rec. IEEE IAS Annu. Meeting,* Toronto, Ont., Canada, Oct., pp. 703–709.

Woei Wan Tan and Hong Huo. 2005. "A Generic Neurofuzzy Model-Based Approach for Detecting Faults in Induction Motors", *IEEE Trans.on Ind. Electr.*, VOL. 52, NO. 5, Oct., pp. 1420-1427.

Wishart M. T. and Harley R. G. 1993. "Identification and control of an ind. machine using artificial neural networks", *Conf. Rec. IEEE IAS Annu. Meeting,* Toronto, Ont., Canada, Oct., pp. 703–709.

DAN S. MIHAI was born in Brabova, Romania and went to the University of Craiova, where he studied electrotechnics and electrical drives, obtaining his degree in 1979. He worked for several years in automotive industry before moving in 1984 to the University of Craiova. He is now PhD Professor leading the subjects in the digital control field for the Faculty for Engineering in Electromechanics, Environment and Industrial Informatics. Professional skills and the recent research interest fields are available on http://www.em.ucv.ro/Organizare/PaginiPersonale/DMihai/DMihai_eng/Default.htm.



a. b.

Figures 12: The In - Out evolution (local) for the DISO optimized speed NFC in conjunction with current NFC.