A Neuro–Fuzzy-Based On-Line Efficiency Optimization Control of a Stator Flux-Oriented Direct Vector-Controlled Induction Motor Drive

Bimal K. Bose, Nitin R. Patel, and Kaushik Rajashekara

***Abstract*—Fuzzy logic-based on-line efficiency optimization control has been described in the literature [1] for an indirect vector-controlled induction motor drive. The purpose of this paper is to extend the same control to a stator flux-oriented electric vehicle induction motor drive and then implement the fuzzy controller by a dynamic back propagation neural network-based controller. The principal advantage of fuzzy control, i.e., fast convergence with adaptive step size of the control variable, is retained. The neural network adds the advantage of fast control implementation, either by a dedicated hardware chip or by digital signal processor (DSP)-based software.**

***Index Terms*— Drive, efficiency, fuzzy logic, induction motor, neural network.**

I. Introduction

Fuzzy logic and neural network techniques are now being increasingly applied to power electronics and variable frequency drives. One interesting application of fuzzy logic is on-line search-based efficiency optimization control of a vector-controlled induction motor drive [1]. The foundation of such a control can be described as follows [2]. A machine is normally operated at the rated flux, in order to give the best transient response. However, at light loads, the rated flux operation gives excessive core loss, thus impairing the efficiency of the drive. Since drives operate at light load most of the time, optimum efficiency can be obtained by programming the flux. The on-line efficiency optimization control on the basis of search, where the flux is decremented in steps until the measured input power for a certain load torque and speed condition settles down to the lowest value, is very attractive. The control does not require any knowledge of machine parameters, is completely insensitive to parameter changes, and the algorithm is applicable universally to any arbitrary drive. The control can be conveniently implemented by fuzzy logic [1], which is described later [see Fig. 1(b)]. The principal advantage of fuzzy control is the fast convergence with adaptive step size of the control variable. This means that the machine flux decrementation starts in the beginning with a large step size, which then gradually decreases so that the optimum flux condition is attained quickly. The additional advantage of fuzzy control is that it can accept inaccurate signals corrupted with noise.

In this paper, the fuzzy efficiency optimization control is extended to a stator flux-oriented direct vector-controlled electric vehicle (EV) induction motor drive of 100-kW power. The fuzzy controller input–output transfer characteristics are then used to train a feedforward neural network with delayed feedback, which then replaces the fuzzy controller in the drive system. The basic implementation of a fuzzy logic controller by a static feedforward neural network is described in [4]. In such an implementation, all the advantages of the fuzzy control are retained. Besides, the neural network adds the advantage of fast computation, either by a dedicated hardware chip or by digital signal processor (DSP)-based software routine. In a complex control system, relief of such a heavy computational burden to the DSP appears be very attractive. Extensive simulation study indicates excellent performance of the neuro–fuzzy control.

Manuscript received March 6, 1996; revised September 17, 1996. This work was supported by Delphi Energy & Engine Management Systems-GM Corporation.

B. K. Bose and N. R. Patel are with the Department of Electrical Engineering, The University of Tennessee, Knoxville, TN 37996-2100 USA.

K. Rajashekara is with Delphi Energy & Engine Management Systems, Indianapolis, IN 46250 USA.

Publisher Item Identifier S 0278-0046(97)01822-4.

II. Control System Description

Fig. 1(a) shows the system control block diagram incorporating the proposed neuro–fuzzy controller. The power circuit of the EV drive consists of a 300-V battery, IGBT bridge inverter and a 100-kW (peak power) induction motor. Basically, it is a sensorless, stator flux-oriented direct vector-controlled system [3], [5], where the developed torque is being controlled in the outer loop. The advantage of stator flux-oriented control is that it is primarily sensitive to stator resistance variation, which can be compensated somewhat easily. Besides, near zero speed, the stator voltage signal behind the stator resistance (R_s) is somewhat larger and easier to process. However, the disadvantage is that it introduces coupling, for which decoupling compensation (i_{dq}) is required. The torque loop generates command for the i_{qs} (torque component of current) loop. The stator flux command Ψ_s^* is constructed by subtracting the neuro–fuzzy controller output $(\Sigma\Delta\Psi_s^*)$ from the rated flux (Ψ_{sr}^*). The flux loop then generates the i_{ds} (flux component of current) command after adding the compensation current (i_{dq}), as shown. Both torque and flux loops then generate the voltage commands which are then rotated and pulsewidth modulation (PWM)-controlled to drive the inverter.

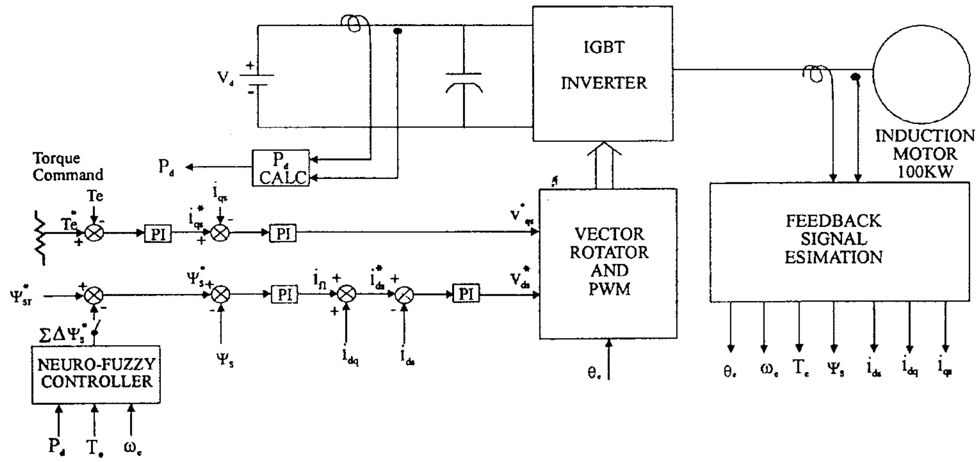
III. Neuro–Fuzzy Control

Fig. 1(b) gives the detailed functional diagram of the neuro–fuzzy controller indicated in Fig. 1(a). Note that this control becomes effective only at steady-state condition, which can be detected by the torque loop error and the frequency signals (not shown). In the beginning, a complete fuzzy controller characterized by membership functions of $\Delta P_d(\text{pu})$, $L\Delta\Psi_s(\text{pu})$ and $\Delta\psi_s^*(\text{pu})$, and the corresponding rule table was developed. The operation principle of Fig. 1(b) can be described as follows. At a certain steady-state speed and load torque conditions, the input dc power (P_d) is sampled and compared with the previous value to determine the decrement ΔP_d. In addition, the last stator flux decrement $(\Delta L\Psi_s(\text{pu}))$ is reviewed. On these bases, the flux decrement step $\Delta\Psi_s^*(\text{pu})$ is generated from the fuzzy membership functions and rule table through fuzzy inference and defuzzification. The adjustable gains S_p and $S_{\Psi s}$, generated by the scale factor computation block, convert the input variable and control variable, respectively, to and from per-unit values, as indicated. The scale factors are given by

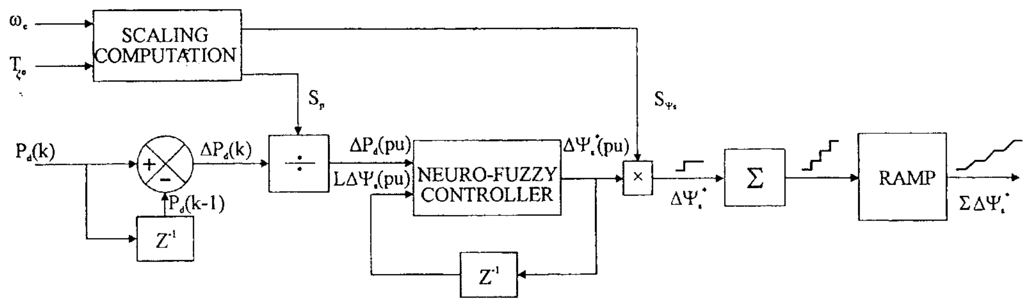
$$S_p = A_1 * \frac{\omega_e}{\omega_e(\text{rated})} + A_2 \tag{1}$$

$$S_{\psi s} = C_1 * \frac{\omega_e}{\omega_e(\text{rated})} + C_2 * \frac{T_e}{T_e(\text{rated})} + C_3 \tag{2}$$

where the symbols and constants are self-explanatory. The $\Delta\psi_s(\text{pu})$ decrementation stops at the minimum P_d (or by the constraint of stator current limit), since any additional $\Delta\psi_s(\text{pu})$ in the same direction will reverse the polarity of $\Delta P_d(\text{pu})$. In fact, the on-line search mechanism makes the operation oscillating about the minimum P_d point. The output flux decrements $(\Delta\Psi_s^*)$ are summed and ramped before coupling to the system in Fig. 1(a). The ramping of flux along



(a)



(b)

Fig. 1. (a) Control system block diagram incorporating the neuro–fuzzy-based efficiency optimization controller. (b) Details of the neuro–fuzzy controller.

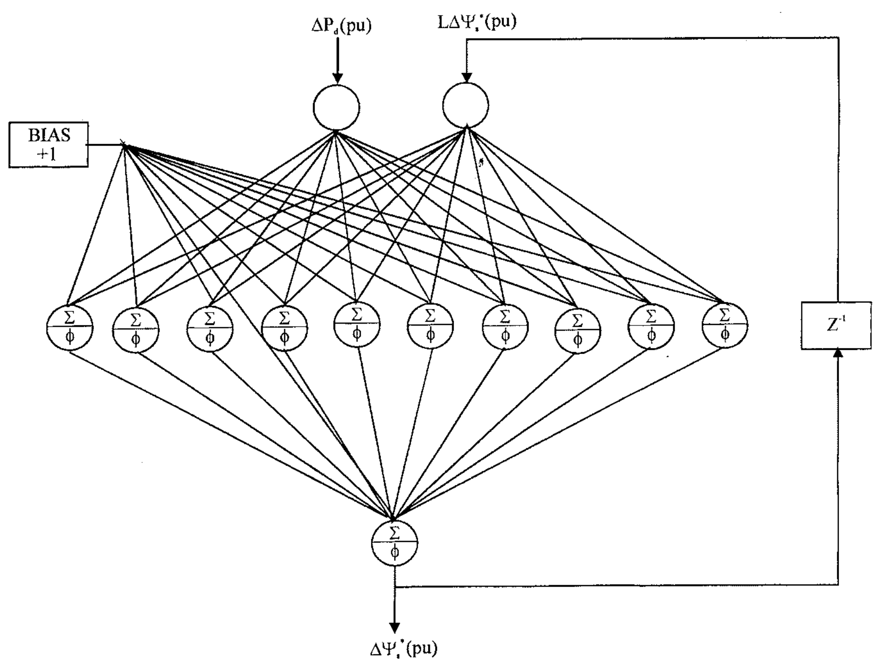


Fig. 2. Topology of feedforward neural network with delayed feedback.

with a high-gain torque loop heavily attenuates the pulsating torque. If any transient condition of the drive is detected, the fuzzy control is abandoned, and the rated flux is established to get the optimal transient response. Note that at the minimum P_d point with the stator current limit condition, the drive cannot withstand any sudden load torque jump because of the sluggishness of the flux loop response. For the same reason, the increase of speed response is somewhat slowed down. However, these limitations do not affect EV-type drives.

Once the fuzzy controller was developed, it was simulated with the complete drive system and iteratively tuned until the best performance was obtained. A dynamical feedforward neural network was then trained to emulate the fuzzy controller.

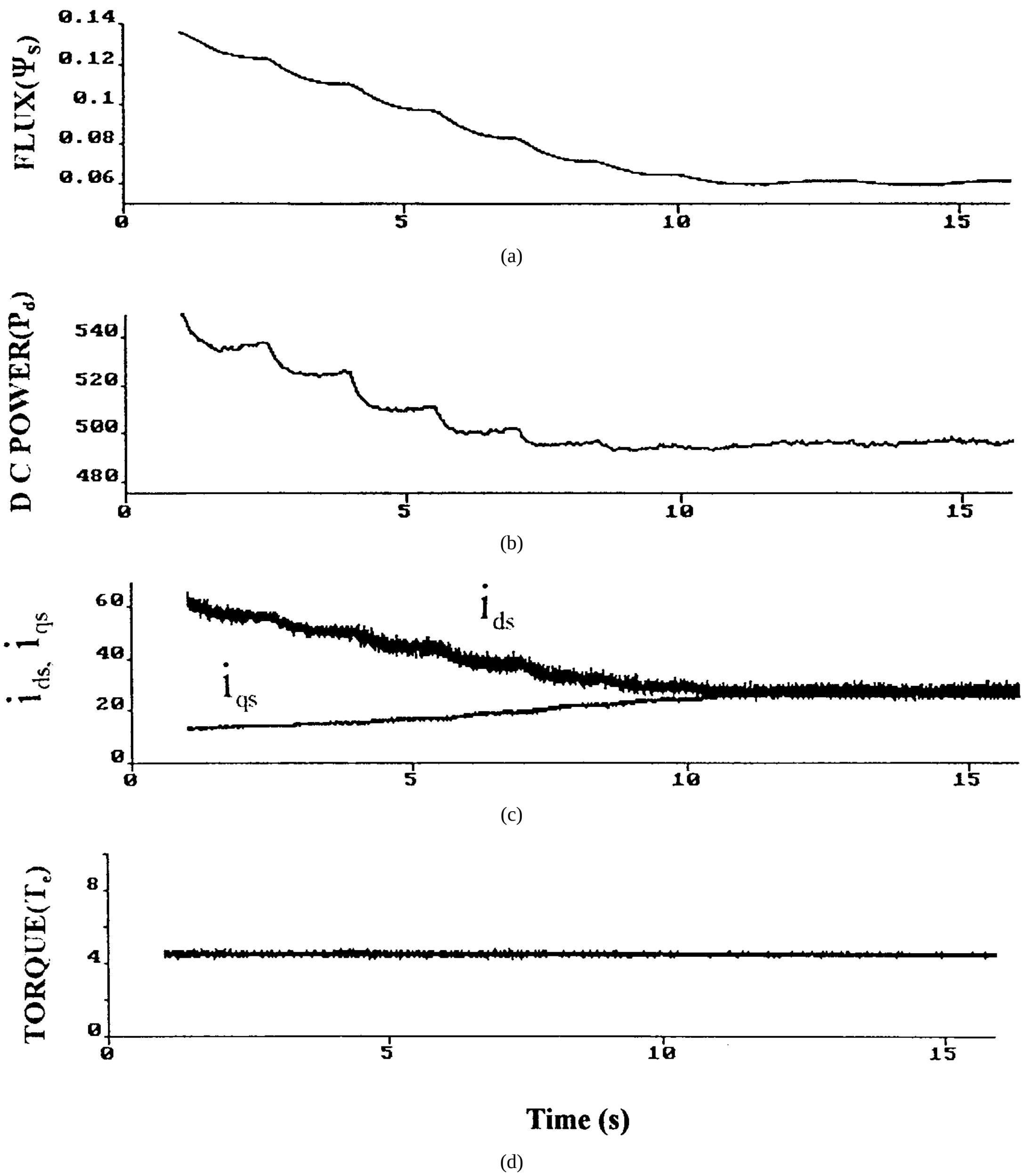


Fig. 3. Time domain optimum efficiency search curves at speed $(N) = 1000$ r/min and torque (Te) $= 4.5$ N·m.

Fig. 2 shows the feedforward neural network topology, which uses one hidden layer, and the respective number of neurons at the input, hidden, and output layer are two, ten, and one. Basically, it represents an input–output nonlinear pattern matching network where the nonlinearity is introduced by hyperbolic–tan-type transfer function (ϕ) at the hidden and output layer neurons. The training data were generated by simulating the fuzzy controller alone with the corresponding input–output signals. NeuralWare Professional II+ software based on backpropagation training algorithm was used to train the network. After extensive training, the fuzzy controller was replaced by the neural network controller in the drive system simulation, and its performance was then evaluated. Fig. 3 shows the performance of the neural network controller at the operating condition of 1000 r/min and load torque of 4.5 N·m. As the steady-state condition is detected by the torque loop error and frequency, the rated flux is decremented in steps. For constant speed and load torque (i.e., for constant output power), the input dc power decreases, which indicates the improvement of efficiency. Note that as i_{ds} decreases with flux, i_{qs} increases, so that the developed torque always balances the load torque. The flux decrementation steps progressively decrease until optimum efficiency (i.e., minimum P_d) condition is attained. At steady state, the operation will oscillate about the optimum point. At every decrementation of flux, a pulsating torque is likely to develop which is not acceptable to EV drive. As mentioned before, the ramping of flux signal along with high-gain torque loop practically eliminates any pulsating torque. Other operating points were found to give similarly good performance.

IV. Conclusion

The fuzzy logic-based on-line efficiency optimization control has been extended to a stator flux-oriented vector-controlled induction motor drive, and the controller is then translated to a dynamical feedforward neural network. Such a neuro–fuzzy control combines the advantages of fuzzy and neural controls. The control attains fast convergence with inherent adaptive step size signals of fuzzy control.

The neural network implementation permits fast computation and can be implemented by a dedicated hardware chip or by DSP-based software. Extensive simulation study verifies excellent performance of the controller. The controller is being implemented in a laboratory on a TMS320C30-type DSP and will be tested with a 100-kW EV drive.

References

[1] G. C. D. Sousa, B. K. Bose, and J. G. Cleland, "Fuzzy logic based on-line efficiency optimization control of an indirect vector-controlled induction motor drive," *IEEE Trans. Ind. Electron.*, vol. 42, pp. 192–198, Apr. 1995.

[2] D. S. Kirschen, D. W. Novotny, and T. A. Lipo, "On-line efficiency optimization of a variable frequency induction motor drive," in *Conf. Rec. IEEE-IAS Annu. Meeting,* 1984, pp. 488–492.

[3] X. Xu, R. W. De Donker, and D. W. Novotny, "A stator flux oriented induction motor drive," in *Conf. Rec. IEEE/PESC,* 1988, pp. 870–876.

[4] G. S. Buja and F. Todesco, "Neural network implementation of a fuzzy logic controller," *IEEE Trans. Ind. Electron.*, vol. 41, pp. 663–665, Dec. 1994.

[5] B. K. Bose, M. G. Simoes, D. R. Crecelius, K. Rajashekara, and R. Martin, "Speed sensorless hybrid vector controlled induction motor drive," in *Conf. Rec. IEEE-IAS Annu. Meeting,* 1995, pp. 137–143.

[6] B. K. Bose and N. R. Patel, "A programmable cascaded low-pass filter based flux synthesis for a stator flux oriented vector controlled induction motor drive," *IEEE Trans. Ind. Electron.*, to be published.

Discussion on "Simulation of Power Electronic Circuits Using Sparse Matrix Techniques"[1]

A. Chandrasekaran

In the above paper,[1] the authors have claimed the speeding-up of a modified nodal method developed by Sudha *et al.* (Ref. [3] in the above paper[1]) by using sparsity programming and resistance modeling of the switches, instead of the inductor modeling used in Ref. [3]. As a co-author of Ref. [3], I want to point out that Krishna *et al.* have not implemented the algorithm in Ref. [3] in the proper manner, while making comparisons for computational speedup. Even though a binary-valued inductor model is used in Ref. [3], the value of the inductor in the open condition of the switch is infinity, and the so-called "elaborate matrix manipulations" mentioned by Krishna *et al.* enable easy accounting of this fact. The matrix to be inverted excludes the switches, and so no refactorization is necessary at any stage of the solution. Krishna *et al.* use a large value of inductance or resistance for the open-switch model, and so the numerical values of the matrix change for different switch status. They do not mention the values used under open-switch conditions in the paper. If sparsity methods are used in the algorithm given in Ref. [3], the computation times can be reduced much more than what is given in the paper under discussion. Resistance switch model is used in the electromagnetic transient program (EMTP) of Prof. Dommel. Krishna *et al.* deserve to be commended for implementing the sparse techniques in the analysis.

Manuscript received January 12, 1996; revised January 15, 1996.

The author is with the Department of Electrical Engineering, Tennessee Technological University, Cookeville, TN 38505-0001 USA.

Publisher Item Identifier S 0278-0046(97)01823-6.

[1] K. Vasudevan, P. S. Rao, and K. S. Rao, *IEEE Trans. Ind. Electron.*, vol. 42, no. 4, pp. 409–413, Aug. 1995.

Authors' Reply

K. Vasudevan, P. S. Rao, and K. S. Rao

We thank Dr. A. Chandrasekaran for his comments and the interest shown in our paper. Our response to his comments is as follows.

1) The algorithm of Ref. [3] of the above paper[1] is implemented by us as per the various steps outlined by the authors in Sections 2.3 and 2.4 of their paper. It is, indeed, mentioned therein that sparsity techniques can be used to find $\boldsymbol{A}^{-1}$, the inverse of matrix $\boldsymbol{A}$. Hence, in our implementation of their algorithm, sparsity techniques have, indeed, been used. In fact, the sparse solution routines are the very same ones that have been used to implement our algorithms, except for the change in the solution routine, which, of course, is necessary.
2) The discusser mentions that, in their algorithm, "The matrix to be inverted excludes the switches, and so no refactorization is necessary at any stage of the solution." Perhaps the discusser is referring to the matrix $\boldsymbol{A}$, which, indeed, is not affected by switch states. This has been taken care of in our implementations, as discussed above. However, the matrix does need to be calculated and inverted whenever switch states change. This fact is clearly mentioned in two places (Sections 2.3 and 4) of Ref. [3].
3) The values of inductances or resistances used for the OFF state in the subject paper do not deserve any special mention. Any arbitrary high values could be chosen, while taking care that the solution procedure does not break down due to matrix illconditioning. The discusser is correct in stating that the numerical values of the system matrix keep changing with changes in switch states.
4) The discusser's comment that the electromagnetic transient program (EMTP) uses the resistance model appears to be an unfinished statement. The EMTP, as well as several other programs, have used resistor models before. We have clearly mentioned in Section IV-A of our paper that Ref. [1] mentions several such cases.

In summary, we feel that all the salient points regarding the algorithm of Ref. [3] have been taken care of in our implementation. In our opinion, comparisons given in our paper, therefore, do give a fair estimate of the computational times for the various algorithms in question.

Manuscript received July 16, 1996; revised November 7, 1996.

K. Vasudevan and P. S. Rao are with the Department of Electrical Engineering, Indian Institute of Technology, Madras, India.

K. S. Rao is with the Department of Electrical Engineering, North Dakota State University, Fargo, ND 58105-5285 USA.

Publisher Item Identifier S 0278-0046(97)01824-8.