Load Variation Compensated Neural Network Speed Controller for Induction Motor Drives

Won Seok Oh*, Kyu Min Cho, Young Tae Kim***, and Hee Jun Kim******

Abstract - In this paper, a recurrent artificial neural network (RNN) based self-tuning speed controller is proposed for the high-performance drives of induction motors. The RNN provides a nonlinear modeling of a motor drive system and could provide the controller with information regarding the load variation, system noise, and parameter variation of the induction motor through the on-line estimated weights of the corresponding RNN. Thus, the proposed self-tuning controller can change the gains of the controller according to system conditions. The gain is composed with the weights of the RNN. For the on-line estimation of the RNN weights, an extended Kalman filter (EKF) algorithm is used. A self-tuning controller is designed that is adequate for the speed control of the induction motor. The availability of the proposed controller is verified through MATLAB simulations and is compared with the conventional PI controller.

Keywords: EKF, induction motor, neural network, load variation, self-tuning, on-line estimation

1. Introduction

For high-performance drives of induction motors, the use of modern control methods is required. The control systems are needed to ensure optimal performance under environmental and load variations and structural perturbations. Many kinds of modern control theories exist in the control area of induction motors. Among them, the self-tuning control is a good method for tuning the controller parameters according to load variations and structural perturbations of the induction motor. However, the self-tuning controller is basically a linear controller. The state estimation and parameter identification are based on the linear models. Thus, there are some drawbacks in the control of induction motors that have nonlinear characteristics [1].

In recent years, the artificial neural network (ANN) has gained wide attention in control applications [1,2,3, 4]. The ANN provides nonlinear modeling of the motor drive system with no knowledge of the predetermined model and thus makes the drive system robust to noise, parameter variations, and load changes. The concept of model reference adaptive control is used in training the ANN to achieve trajectory control of the induction motor. Most of the ANN-based adaptive control approaches use off-line system identification. Because of this separation, it is impossible to effectively cope with system parameters that are changed dynamically during operation, which makes the tuning of the respective controller parameters difficult. Thus, an on-line learning process is desirable [5].

In this paper, an on-line self-tuning speed controller is proposed for nonlinear induction motors. This controller structure is based on the minimum variance control theory and controller parameters are composed of system parameters. The recurrent neural network (RNN) model is used to identify the variation of motor parameters and load conditions. The proposed control theory possesses the capabilities of simultaneous online identification and control. For their training, the extended Kalman filter (EKF) algorithm is used The ability of the proposed controller is verified in the presence of external load variations using simulations.

2. Self-Tuning Controller

The mathematical model of the system that is to be controlled in this paper, under the influence of load variation and motor parameter variation, can be expressed with the single-input and single-output ARMAX (autoregressive moving average model with auxiliary input) [6].

$$A(q^{-1})y(k) = q^{-d}B(q^{-1})u(k) + C(q^{-1})\omega(k) \tag{1}$$

where $\{y(k)\}$ and $\{u(k)\}$ are the output and input of the system, $\omega(k)$ is uncorrelated white noise sequence, k

* Dept. of Elec. Eng., Yuhan College, Korea (oppa@yuhan.ac.kr)
** Dept. of Info. and Comm. Eng., Yuhan College, Korea (limsa@yuhan.ac.kr)
*** Dept. of Elec. Eng., Wonju College, Korea (ytkim@sky.wonju.ac.kr)
**** Division of Elec. and Comp. Eng., Hanyang University, Korea (hjkim@hanyang.ac.kr)
Received October 19, 2002; Accepted May 23, 2003

is sampling instant (k = 1,2,3,⋯), and q^{-d} is unit time delay operator. Thus, for example, $q^{-d}y(k) = y(k-d)$. $d(d \geq 1$, integer$)$ is time delay of the system. A, B, and C are polynomials of q^{-1} and can be expressed as (2).

$$\begin{aligned} A(q^{-1}) &= 1 + a_1 q^{-1} + \cdots + a_n q^{-n} \\ B(q^{-1}) &= b_0 + b_1 q^{-1} + \cdots + b_m q^{-m} \quad ; b_0 \neq 0 \\ C(q^{-1}) &= 1 + c_1 q^{-1} + \cdots + c_l q^{-l} \end{aligned} \tag{2}$$

where n, m, l is the order of the polynomial A, B, C. The accuracy of the ARMAX is determined by the choice of process time delay d and the order of A, B, C that should be selected according to the physical characteristics of the system under control. To design a speed controller of the induction motor, we define the following performance index.

$$\begin{aligned} J(u,k) &= E\{[y(k+d) - y_{ref}(k+d)]^2 \\ &\quad + \rho_u [Ru(k)]^2 + \rho_v [\upsilon_e(k+d)]^2\} \\ &= E\{\zeta(k)\} \end{aligned} \tag{3}$$

where $E\{\cdot\}$ is the expectation at time $(k+d)$, which is a function of data acquired up to time k and $y_{ref}(k)$ is the desired reference trace. To reduce the damaging effect on the system resulting from abrupt changes in the control signal, we select R with an integration effect.

$$R = 1 - q^{-1} \tag{4}$$

To have robust control performance, we include the v_e that is the integration of trace error.

$$v_e(k) = v_e(k-1) + e(k) \tag{5}$$

$$e(k) = y_{ref}(k) - y(k) \tag{6}$$

ρ_u, ρ_v are the weighting factors used to adjust the trade-off between control signal and accuracy of system response. $J(u,k)$ is minimized if the derivative of $\zeta(k)$ with respect to the control signal at the time k is equal to zero. Thus, the control signal $u(k)$ is obtained.

3. Induction Motor Drive System

The mechanical load-torque equation of the induction motor system is represented as in (7) [7].

$$\frac{dw_r}{dt} = \frac{P}{2J}(T_e - T_L) \tag{7}$$

where w_r is angular speed, P is pole number, J is inertia of the motor, and T_e and T_L are generating torque of the induction motor and load torque, respectively.

In the synchronously rotating reference frame, the generating torque is expressed as in (8).

$$T_e = \frac{3}{2}\left(\frac{P}{2}\right)\frac{L_m}{L_r} I_{qs} \psi_{dr} \tag{8}$$

where L_m, L_r are the magnetizing inductance and rotor inductance, respectively, and I_{qs}, ψ_{dr} are the stator current of the q axis and the rotor flux of the d axis.

We assume the load torque as shown in (9) [6].

$$T_L = K_1 + K_2 w_r + K_3 w_r^2 + J_L \frac{dw_r}{dt} \tag{9}$$

where K_1, K_2, K_3 are constant and J_L is the inertia of the load. The square term of (9) is a common characteristic for most fan-type loads.

The system of differential equations, (7), (8), and (9), can be rewritten as a single second-order nonlinear ordinary differential equation and can be discretized as shown in (10).

$$\begin{aligned} w_r(k+1) &= c_1 w_r(k) + c_2 w_r(k-1) + c_3 w_r^2(k) + c_4 w_r^2(k-1) \\ &\quad + c_0[I_{qs}(k) - I_{qs}(k-1)] \end{aligned} \tag{10}$$

where c_0, c_1, c_2, c_3, c_4 are the constant values, which could be expressed in terms of system parameters. Through (10), we can conclude that there exists a nonlinear relation between speed w_r and control input current I_{qs}. Thus, we need to design a nonlinear controller using the nonlinear modeling method, and we introduce a neural network model that is adequate to fulfill the system equation in (10)

4. RNN-Based Modeling

The dynamic system representation capabilities of RNNs have been shown to be considerably greater than those of purely static networks [2,3]. Fig. 1 shows the general architecture of an RNN, where Z^{-1} indicates one sampling time delay, $U_M(k)$ is the network input, $Y_N(k+1)$ is the output of the network and is produced one step ahead at the discrete time $(k+1)$, and $W_{N(N+M)}(k)$ are weights of the network. The net internal activity of neuron j at discrete time k is given by

$$net_j(k) = \sum_{i=1}^{N+M} W_{ji}(k)U_i(k). \tag{11}$$

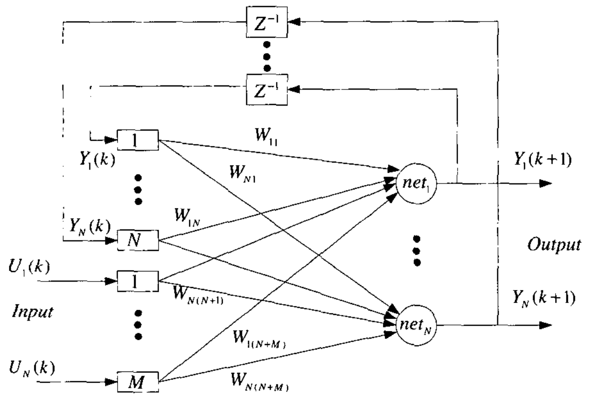


Fig. 1 General structure of the RNN

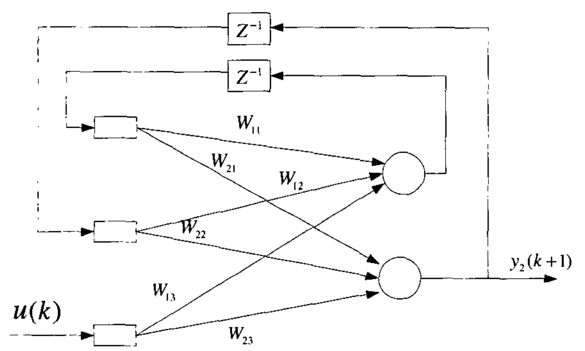


Fig. 2 RNN structure used in this paper

The RNN desired output $Y(k+1)$ is given by

$$Y_j(k+1) = \gamma(net_j(k)) \tag{12}$$

where $\gamma(\cdot)$ is the activation function.

We can design an induction motor model as shown in Fig. 2 that has one input, $u(k)$, which is current $I_{qs}(k)$, and two outputs, $y_1(k)$ and $y_2(k)$. $y_2(k)$ is the speed output $w_r(k)$ of the induction motor and $y_1(k)$ is intermediate virtual output.

5. Extended Kalman Filter Training

The EKF algorithm has shown significant merit for training both feedforward and recurrent neural networks. The training algorithm based on the EKF is shown to require significantly smaller training data than the pure gradient descent algorithms [2][4] and is simpler than gradient descent algorithms although it requires the same derivative information. The EKF requires no batch processing of data and, therefore, is very suitable for on-line training.

Let the desirable or target vector at time k be given by (13).

$$Y_r(k) = [y_{r1}(k) \ldots y_{rN}(k)]^T \tag{13}$$

where N is the length of the vector. If the output $y_i(k)$ of the network can be represented by the vector $Y(k)$, the error vector is (14).

$$\xi(k) = Y_r(k) - Y(k) \tag{14}$$

The RNN weights can be expressed into an M- dimensional vector $W(k)$. The components of $Y(k)$ are the partial derivatives of output $y_i(k)$ of network with respect to the weights of the network. The derivatives are represented in the form of $M \times N$ matrix $H(k)$ as (15).

$$H(k) = \begin{bmatrix} \frac{\partial(y_1(k))}{\partial W_1} & \cdots & \frac{\partial(y_N(k))}{\partial W_1} \\ \vdots & \ddots & \vdots \\ \frac{\partial(y_1(k))}{\partial W_M} & \cdots & \frac{\partial(y_N(k))}{\partial W_M} \end{bmatrix} \tag{15}$$

where y_i is respective neuron output. Then, $W(k)$ and $P(k)$ are updated using the following EKF recursion equations.

$$A(k) = [(\eta(k)S(k))^{-1} + H(k)^T P(k)H(k)]^{-1} \tag{16}$$

$$K(k) = P(k)H(k)A(k) \tag{17}$$

$$W(k+1) = W(k) + K(k)\xi(k) \tag{18}$$

$$P(k+1) = P(k) - K(k)H(k)^T P(k) \tag{19}$$

where $\eta(k)$ is the learning parameter. The EKF algorithm works in such a way that at discrete sampling time k, the input signals and recurrent nodes outputs are propagated through the network and the function $Y(k)$ is calculated. After the error vector $\xi(k)$ is computed, the Kalman gain

K(k) is updated and we can get the new weights *W(k+1)*. New *W(k+1)* go to the controller as new gains. After that, *P* is updated for the next calculation. Fig. 3 shows the computation flow chart of the EKF training algorithm.

6. Design of Induction Motor Speed Control System

From (10), we could conclude that the speed output model of the induction motor is a function of speed and input like (20).

$$y_2(k+1) = f[y_2(k), y_2(k-1), u(k), u(k-1)] \tag{20}$$

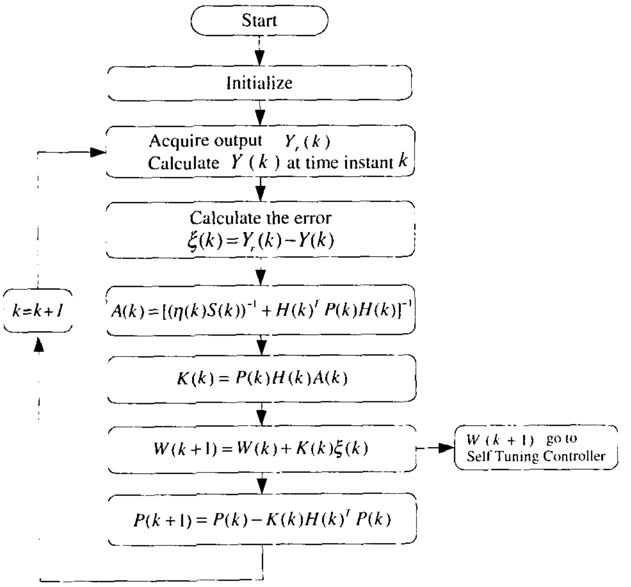


Fig. 3 Flow chart for the EKF training algorithm

Thus, it can be shown that $W_{11}=0$ in Fig. 2.

To minimize the performance index $J(u,k)$ of (3), the derivative of $\zeta(k)$ with respect to the control signal at the time *k* should be equal to zero. Then we can find the control input $u(k)$ of the induction motor speed control system that satisfies (21).

$$\frac{d\zeta(k+1)}{du(k)} = 0 \tag{21}$$

The control input is shown as in (22).

$$u(k) = \frac{1}{b_0}[a_1 y_2(k) + a_2 y_2(k-1) + a_0 y_{ref}(k+1) + b_1 u(k-1) + b_2 \upsilon_e(k)] \tag{22}$$

where $a_1 = -W_{22}W_{23}(1+\rho_v)$, $a_2 = -W_{12}W_{21}W_{23}(1+\rho_v)$, $a_0 = W_{23}(1+\rho_v)$, $b_0 = W_{23}{}^2(1+\rho_v)+\rho_u$, $b_1 = \rho_u - W_{13}W_{21}W_{23}(1+\rho_v)$, and $b_2 = W_{23}\rho_v$.

Using the EKF algorithm, we could get the weighting matrix W . Then, we have control input value $u(k)$ through (22) that is composed of weighting values. $u(k)$ is the torque current component of induction motor, $I_{qs}(k)$.

7. Results

To verify the validity of the proposed self-tuning neural network controller, several simulations are carried out using MATLAB and Simulink software. Fig. 4 shows the block diagram of the proposed control system. In the RNN, the weights of the neuron are calculated using the EKF algorithm and go to the self-tuning neural network controller to tune its gains.

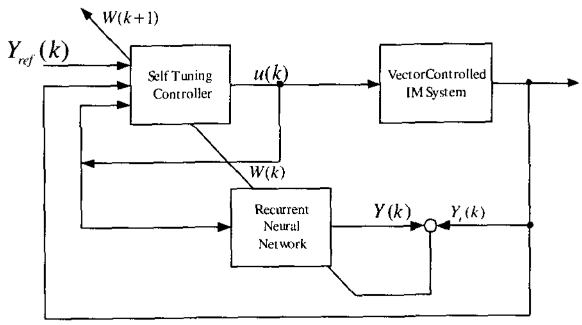


Fig. 4 Block diagram of self-tuning neural network control system

Simulations are factually focused on whether or not the proposed algorithm is better than PI linear controller. For the comparison, simulations of the speed response are performed in case of speed command, load, and inertia variation of the induction motor.

Table 1 shows the parameters of the tested induction motor whose general specifications are 5hp, 230V, and 4 pole. The sampling time of the controller is 100 μs.

Figs. 5 and 6 show speed response waveforms in the case of load variation using a conventional PI speed controller in which PI gains are designed to be optimal and using the proposed self-tuning neural controller, respectively. The command speed is in increased from zero to 180rad/sec and the 50% disturbance load of the rated torque (T_{LN}) is applied at 7sec.

Table 1 Parameters of the tested induction motor

Stator resistance	0.5814 Ω
Rotor resistance	0.4165 Ω
Stator leakage inductance	3.479 mH
Rotor leakage inductance	4.15 mH
Magnetizing inductance	78.25 mH
Rotor inertia	0.1 $Kg.m^2$
Base frequency	377 rad/s
Rated torque	19.8 Nm

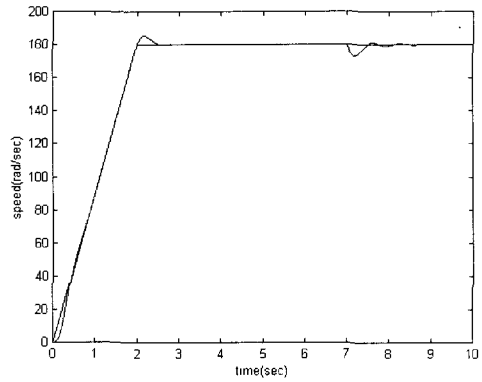


Fig. 5 Speed response using PI controller ($T_L = 0.5T_{LN}$)

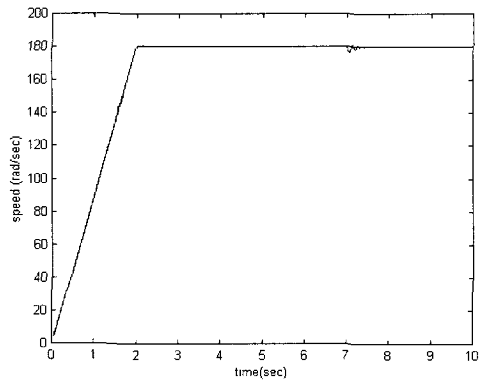


Fig. 6 Speed response using proposed controller ($T_L = 0.5T_{LN}$)

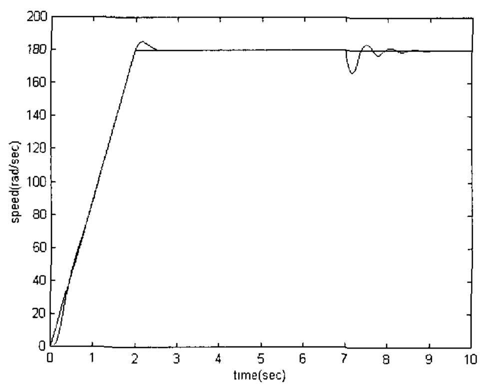


Fig. 7 Speed response using PI controller ($T_L = T_{LN}$)

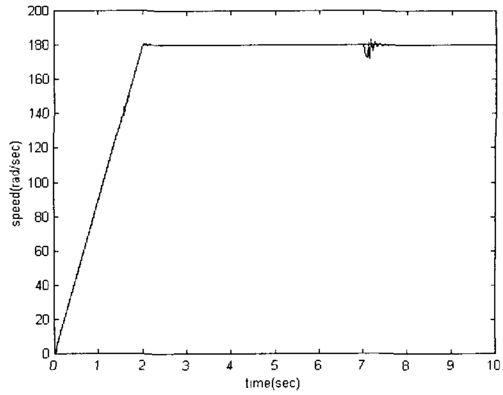


Fig. 8 Speed response using proposed controller ($T_L = T_{LN}$)

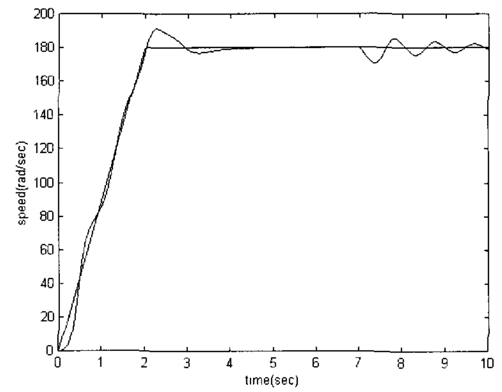


Fig. 9 Speed response using PI controller ($J_L = 4J_{LN}$)

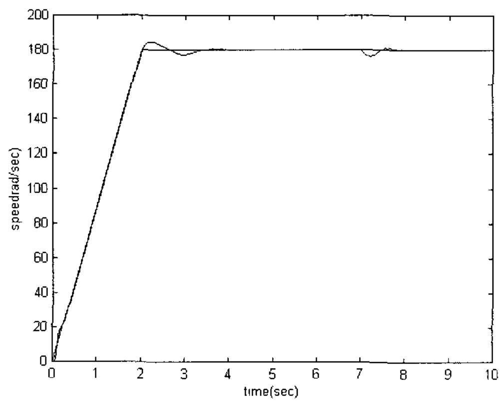


Fig. 10 Speed response using proposed controller ($J_L = 4J_{LN}$)

Figs. 7 and 8 show speed response waveforms under full load. From all points of view, such as variable speed characteristics and speed recovery time, the proposed neural controller is better than the conventional PI controller. In particular, in the full load case, the PI controller response has more overshoot and vibration than the proposed neural controller. Figs. 9 and 10 show speed response with inertia variation, where the load moment of inertia J_L is changed to four times the normal value J_{LN} and full load is applied at 7sec. We can observe that the proposed neural controller is more robust to the inertia variation.

7. Conclusion

In this paper, an on-line self-tuning neural network controller for induction motor drives was presented. The structure of the proposed controller is based on the self-tuning controller that can tune its control gain according to system parameter variations to be autonomous. An RNN induction motor model is used to identify the variation of the induction motor system. The EKF algorithm is used to train the RNN weights that represent parameter variations. The designed controller also can compensate for uncertainties of the nonlinear induction motor control system since the real output values are directly used for parameter identification and tuning. Simulation results using MATLAB verify the effectiveness of the proposed controller. Implementation is currently in progress.

References

[1] M. A. El-Sharkawi and S. Weerasooriya, "Development and Implementation of Self-Tuning Tracking Controller for DC Motors," *IEEE Trans. on Energy Conversion*, pp. 122–128, March 1990.

[2] Joao O. P. Pinto, Bimal K. Bose, and Luiz E. B. Silva, "A Stator Flux Oriented Vector-Controlled Induction Motor Drive with Space Vector PWM and Flux Vector Synthesis by Neural Networks," *IAS*, pp. 1605–1612, Sep. 2000.

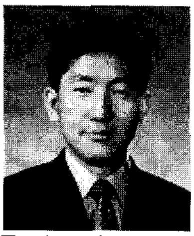
[3] K. S. Narendra and K. Parthasarathy, "Identification and Control of Dynamic Systems using Neural Networks," *IEEE Trans. Neural Networks*, vol. 1, pp. 4–27, Jan. 1990.

[4] M. A. Hoque, M. R. Zaman and M. A. Rahman, "Artificial Neural Network Based Permanent Magnet DC Motor Drives," *IAS*, pp. 98—103, 1995.

[5] G. V. Puskorius and L. A. Feldkamp, "Neurocontrol of Nonlinear Dynamic Systems with Kalman Filter Trained Recurrent Networks," *IEEE Trans. Neural Networks*, vol. 5, pp. 279–297, Mar. 1994.

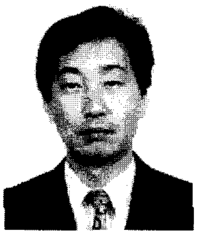
[6] Goodwin, G. C. and K. S. Sin, *Adaptive Filtering Prediction and Control*, Prentice Hall, 1984.

[7] B. K. Bose, *Power Electronics and AC Drives*, Prentice Hall. 1986.



Won Seok Oh

He received the B.S., M.S., and Ph.D. degrees in Electrical Engineering from Hanyang University, Seoul, Korea, in 1985, 1987, and 1994, respectively. He was a visiting scholar at the University of Tennessee in 2001. Presently, he is an associate professor of Electrical Engineering at Yuhan College, Pucheon, Korea. His research interests are AI-based motor control and photovoltaic generation systems.



Kyu Min Cho

He received his B.S., M.S., and Ph.D. degrees in Electrical Engineering from Inha University, Incheon, Korea, in 1985, 1991, and 1994, respectively. He is an associate professor of the Department of Information and Communications at Yuhan College, Pucheon, Korea. His research interests include communications power supply, electronic ballasts, switching techniques of power converters, and high-frequency power sources for plasma applications.

Young Tae Kim

He received his B.S, M.S., and Ph.D. degrees in Electrical Engineering from Hanyang University, Seoul, Korea, in 1984, 1989, and 1996, respectively. He is an assistant professor of Electrical Engineering at Wonju National College, Wonju, Korea. His research interests are DC-DC converters, motor control, and power systems of spacecraft

Hee Jun Kim

He received the B.S. and M.S. degrees in Electronics Engineering from Hanyang University, Seoul, Korea, in 1976 and 1978, respectively. He received his Ph.D. degree in Electronics Engineering from Kyushu University, Kyushu, Japan, in 1986. He is a professor of the School of Electrical Engineering and Computer Science, Hanyang University, Ansan, Korea. From 1991 to 1992, he was with Virginia Polytechnic Institute and State University as a visiting scholar. Since 1988, he has been a technical advisor with the Dong-Ah Elecomm Company, Korea. His interests include switching power converters, electronic ballasts, soft switching techniques, and analog signal processing.