

A Transputer-Based Adaptive Speed Controller for AC Induction Motor Drives with Load Torque Estimation

Ming-Fa Tsai, *Student Member, IEEE* and Ying-Yu Tzou, *Member, IEEE*

Abstract—In this paper, we design and implement an adaptive speed controller that can estimate load torque for ac induction motor drives employing a transputer-based parallel processing technique. The adaptive speed controller, which precedes the field-oriented control loop, consists of a two-degree-of-freedom controller and a feedforward load-torque compensator. The two-degree-of-freedom controller is designed by a pole-placement technique with polynomial manipulations. Its parameters are adjusted adaptively in terms of estimated model parameters. Estimating the model parameters entails a second-order least-squares estimator with constant trace to avoid estimator windup. The design of the feedforward compensator is based on an estimated load-torque model. Estimating the load torque entails a first-order least-squares estimator with variable forgetting factor and covariance resetting, the purposes of which are to detect any slow or sudden changes of torque disturbance, respectively. The resulting adaptive controller is implemented in parallel by IMS T800-20 transputers. Experimental results demonstrate the robustness of the proposed control method in contending with varying load and torque disturbance.

Index Terms—Adaptive control, induction motor, load torque estimation, transputers.

NOMENCLATURE

R_s	Stator resistance per phase.
R_r	Rotor resistance per phase.
L_s	Stator inductance per phase.
L_r	Rotor inductance per phase referred to stator.
L_m	Mutual magnetizing inductance per phase.
P	Number of poles.
p	Differential operator.
ω_e	Synchronous rotating frame angular speed.
ω_r	Motor electrical angular speed.
ω_m	Motor rotor angular speed.
ω_{sl}	Slip angular speed.
$v_{qs}(v_{ds})$	q -axis (d -axis) stator voltage.
$i_{qs}(i_{ds})$	q -axis (d -axis) stator current.
$i_{qs}^*(i_{ds}^*)$	q -axis (d -axis) stator current command.
$i_{qr}(i_{dr})$	q -axis (d -axis) rotor current.

$v_\alpha(v_\beta)$	α -axis (β -axis) stator voltage.
$i_\alpha(i_\beta)$	α -axis (β -axis) stator current.
$i_\alpha^*(i_\beta^*)$	α -axis (β -axis) stator current command.
$\psi_{qs}(\psi_{ds})$	q -axis (d -axis) stator flux linkage.
$\psi_{qr}(\psi_{dr})$	q -axis (d -axis) rotor flux linkage.
T_e	Motor developed torque.
T_L	External load torque.
J_m	Motor rotor inertia.
J	System total inertia.
B_m	Motor viscous coefficient.
B	System total viscous coefficient.
K_T	Torque constant.
K_E	Back EMF voltage constant.
τ_m	Model mechanical time constant.
τ_e	Model electrical time constant.
τ_r	Motor rotor time constant.

I. INTRODUCTION

PRESENT-DAY automation industry increasingly demands high-performance electrical servo drives, particularly of the type that guarantee asymptotic tracking and regulation, irrespective of changes in load or motor parameters or of load torque disturbances. Adaptive control methods that take torque disturbance suppression into account have been utilized to improve servo drive performance and help maintain invariant control behavior in a wide range of variable operating conditions [1]–[4].

Basically, such speed control systems for ac servo drives come in two categories: a model reference adaptive system (MRAS) and a self-tuning regulator (STR). In a MRAS system, a control input drives the controlled plant, tracking the response produced by the reference model; research work on this aspect is not lacking [5]. On the other hand, the STR system explicitly estimates plant model parameters and, based on these, then redesigns the controller to control the invariant speed [6]–[8]. Although research to date into adaptive speed control for ac motor drives is not lacking, attention must be given to ascertain the effects of adaptive control schemes and of the compensation of load torque disturbance when the motor parameters change and load torque disturbance occurs, giving rise to improve the performance.

As illustrated in Fig. 1, the adaptive controller presented herein most closely relates to a self-tuning regulator which can additionally estimate external load torque for reducing the

Paper IPCSD 96–58, approved by the Industrial Drives Committee of the IEEE Industry Applications Society for presentation at the 1995 IEEE/IAS International Conference on Industrial Automation and Control, Taipei, Taiwan, R.O.C., May 22–27. Manuscript released for publication October 21, 1996. This work was supported by the National Science Council, Taipei, Taiwan, R.O.C., under Project NSC84-2213-E009-010.

The authors are with the Power Electronics and Motion Control Laboratory, Institute of Control Engineering, National Chiao Tung University, Hsinchu, 30050 Taiwan, R.O.C. (e-mail: yytzou@cc.nctu.edu.tw).

Publisher Item Identifier S 0093-9994(97)01543-0.

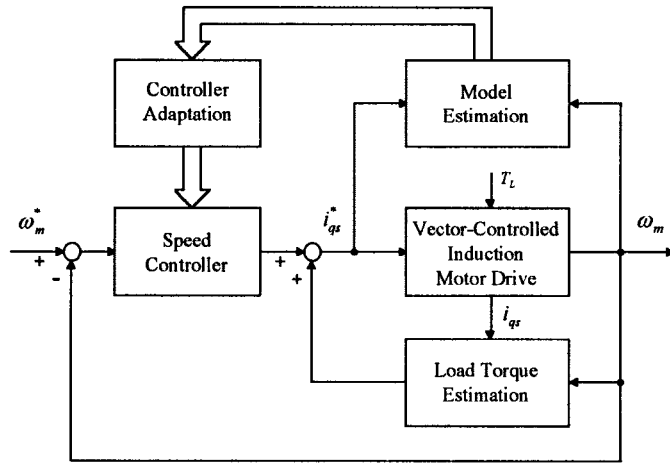


Fig. 1. The functional block diagram of an adaptive controller with load torque estimation for an ac induction motor drive.

torque disturbance effect by feedforward compensation. Estimating load torque requires a first-order least-squares estimator [9], rather than the conventional observation technique by a load torque observer [10]. In this way, the model parameters and load torque estimations are separate and the self-tuning regulation and feedforward compensation are decoupled and, hence, can be independently designed. Compensating torque disturbance can enhance the robustness of the system. However, applying the method to an ac induction motor drive for rapid response control has been somewhat restricted due mainly to the control algorithm, the complexity of which has proved a considerable obstacle in the way of microcontrollers and microprocessors. The advent of high-speed 32-b transputers, operating at a 20-MHz (or above) clock frequency [11], renders the complicated control algorithm amenable to implementation in parallel, resulting in a high sampling-rate operation.

This paper presents the design and implementation of a self-tuning adaptive speed controller that can additionally estimate and feedforward compensate load torque for ac induction motor drives, employing a transputer-based parallel processing technique. The system can rapidly compensate any output error caused either by variation of the motor parameters or by external torque disturbances. Experimental results demonstrating the proposed controller's effectiveness are presented.

II. DYNAMIC MODELING OF A VECTOR-CONTROLLED AC INDUCTION MOTOR DRIVE

The dynamic properties of an induction motor as a controlled plant may be described by a set of nonlinear differential equations linking the stator and rotor currents and the voltages with the mechanical quantities, such as torque, speed, and angular position. The analysis and control of such a plant appears complicated because of the intricate coupling of all the control inputs, but the problem can be overcome by controlling the field-oriented quantities, which reduce the control of an ac induction motor to that of a separated excited dc motor [12].

In deriving the dynamic model of a field-oriented or vector-controlled induction motor drive, the voltage and developed-

torque equations for an ac induction motor of the squirrel cage type can be expressed in terms of the d - q axes components in a synchronous rotating frame as follows:

$$v_{ds} = R_s i_{ds} + \frac{d\psi_{ds}}{dt} - \omega_e \psi_{qs} \quad (1)$$

$$v_{qs} = R_s i_{qs} + \frac{d\psi_{qs}}{dt} + \omega_e \psi_{ds} \quad (2)$$

$$0 = R_r i_{dr} + \frac{d\psi_{dr}}{dt} - \omega_{s1} \psi_{qr} \quad (3)$$

$$0 = R_r i_{qr} + \frac{d\psi_{qr}}{dt} + \omega_{s1} \psi_{dr} \quad (4)$$

$$T_e = \frac{3P}{4} L_m (i_{qs} i_{dr} - i_{ds} i_{qr}) \quad (5)$$

where

$$\psi_{ds} = L_s i_{ds} + L_m i_{dr} \quad (6)$$

$$\psi_{qs} = L_s i_{qs} + L_m i_{qr} \quad (7)$$

$$\psi_{dr} = L_r i_{dr} + L_m i_{ds} \quad (8)$$

$$\psi_{qr} = L_r i_{qr} + L_m i_{qs} \quad (9)$$

and

$$\omega_{s1} = \omega_e - \omega_r. \quad (10)$$

All symbols are listed in the Nomenclature at the beginning of this paper. Eliminating ψ_{ds} , ψ_{qs} , i_{dr} , and i_{qr} in (1)–(4) by employing (6)–(9), the voltage equations in terms of stator current and rotor flux linkage can be restated in matrix form as

$$\begin{bmatrix} v_{ds} \\ v_{qs} \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} R_s + L_\sigma p & -\omega_e L_\sigma & \frac{L_m p}{L_r} & -\frac{\omega_e L_m}{L_r} \\ \omega_e L_\sigma & R_s + L_\sigma p & \frac{\omega_e L_m}{L_r} & \frac{L_m p}{L_r} \\ -\frac{L_m R_r}{L_r} & 0 & \frac{R_r}{L_r} + p & -\omega_{s1} \\ 0 & -\frac{L_m R_r}{L_r} & \omega_{s1} & \frac{R_r}{L_r} + p \end{bmatrix} \times \begin{bmatrix} i_{ds} \\ i_{qs} \\ \psi_{dr} \\ \psi_{qr} \end{bmatrix} \quad (11)$$

where p is the differential operator and

$$L_\sigma = \frac{L_s L_r - L_m^2}{L_r}. \quad (12)$$

The developed-torque equation in (5) becomes

$$T_e = \frac{3P}{4} \frac{L_m}{L_r} (i_{qs} \psi_{dr} - i_{ds} \psi_{qr}). \quad (13)$$

Ideal decoupling control between d - and q -axes can be achieved by aligning the rotor flux vector to the d -axis and setting the rotor flux linkage to be a constant, which means

$$\psi_{qr} = \frac{d\psi_{qr}}{dt} = 0 \quad (14)$$

$$\psi_{dr} = \psi_r = \text{constant}. \quad (15)$$

Employing (14) on the third and fourth row of (11) yields

$$\frac{L_r}{R_r} \frac{d\psi_r}{dt} + \psi_r = L_m i_{ds} \quad (16)$$

and

$$\omega_{s1} = \frac{L_m R_r}{\psi_r L_r} i_{qs}. \quad (17)$$

Further, the torque equation in (13) becomes

$$T_e = \frac{3P}{4} \frac{L_m}{L_r} (i_{qs}^* \psi_{dr}). \quad (18)$$

Thus, with the condition of (15), which can be achieved by setting the i_{ds}^* command to be a constant when the motor is operating in its constant-torque region, the motor's torque generation, with decoupling, can be controlled by the stator current i_{qs} .

The vector control scheme, which employs a rotor flux model based on (16) and (17), is illustrated in Fig. 2. Note that the current-controlled PWM inverter has also been included in the system. To obtain the equivalent vector-controlled plant model for the adaptive controller design, the current control dynamics should be considered. Taking a traditional current-controlled sinusoidal PWM inverter drive, let K_I be the equivalent current controller gain; then, the torque-producing current control action can be modeled as

$$v_{qs}(t) = K_I (i_{qs}^*(t) - i_{qs}(t)). \quad (19)$$

Incorporating (10), (12), (14), (15), and (17) into the second row of (11) yields

$$v_{qs} = L_\sigma \frac{di_{qs}}{dt} + L_s \omega_r i_{ds} + \left(R_s + \frac{L_s}{L_r} R_r \right) i_{qs}. \quad (20)$$

The developed-torque equation and the rotor angular speed are related by

$$T_e - T_L = J \frac{d\omega_m}{dt} + B\omega_m = \frac{2}{P} \left(J \frac{d\omega_r}{dt} + B\omega_r \right). \quad (21)$$

Equations (18)–(21) consequently constitute the dynamic model of the vector-controlled ac induction motor drive represented by the block diagram in Fig. 3. The open-loop response of $\omega_m(s)$ can be expressed from Fig. 3 as

$$\omega_m(s) = G_1(s) i_{qs}^*(s) + G_2(s) T_L(s) \quad (22)$$

where $G_1(s)$ and $G_2(s)$ are as shown by (23) and (24), at the bottom of the page. In (23) and (24)

$$R_\sigma = R_s + \frac{L_s}{L_r} R_r \quad (25)$$

$$K_T = \frac{3P}{4} \frac{L_m^2}{L_r} i_{ds} \quad (26)$$

$$K_E = \frac{P}{2} L_s i_{ds} \quad (27)$$

where

- K_m steady-state gain;
- τ_m mechanical time constant;
- τ_e electrical time constant.

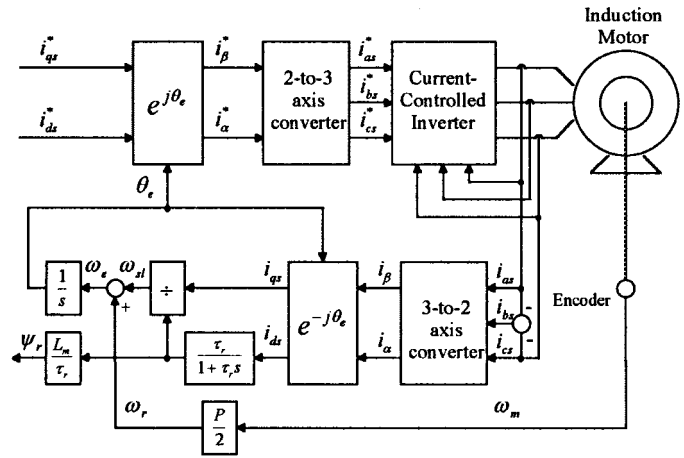


Fig. 2. Vector control scheme of an ac induction motor drive.

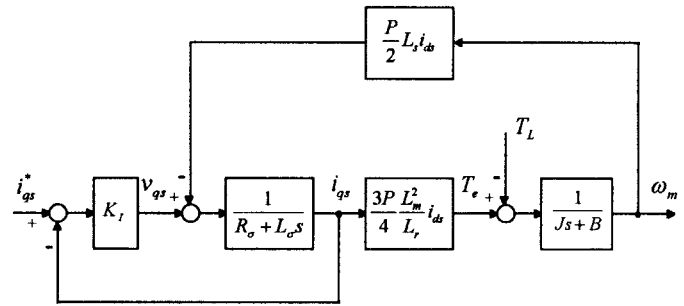


Fig. 3. Equivalent model of the vector-controlled induction motor drive.

Since the load torque is an inaccessible input, the transfer function $G_1(s)$ will be used as the plant model for the adaptive controller design. The magnitude of torque disturbance will be estimated by estimation theory and then eliminated by feedforward compensation.

III. ADAPTIVE SPEED CONTROL WITH LOAD TORQUE ESTIMATION

The adaptive controller for the ac induction motor drive consists of an on-line parameter estimator, a 2-degree-of-freedom controller adaptation, and an estimator-based load torque compensator. These are described as follows.

A. Parameter Estimation

For the parameter estimation, the pulse-transfer function of $G_1(s)$ in the z -domain can be derived through a zeroth-order hold device as

$$G_1(z) = \frac{B(z)}{A(z)} = \frac{b_1 z + b_2}{z^2 + a_1 z + a_2} \quad (28)$$

$$G_1(s) = \frac{K_I K_T}{L_\sigma J s^2 + (K_I J + R_\sigma J + L_\sigma B) s + (K_E K_T + K_I B + R_\sigma B)} = \frac{K_m}{(\tau_m s + 1)(\tau_e s + 1)} \quad (23)$$

$$G_2(s) = \frac{-(L_\sigma s + R_\sigma)}{L_\sigma J s^2 + (K_I J + R_\sigma J + L_\sigma B) s + (K_E K_T + K_I B + R_\sigma B)}. \quad (24)$$

where a_1 , a_2 , b_1 , b_2 are function of the motor parameters and can be obtained as

$$a_1 = -e^{-T/\tau_e} - e^{-T/\tau_m} \quad (29)$$

$$a_2 = e^{-(T/\tau_e + T/\tau_m)} \quad (30)$$

$$b_1 = K_m \left(1 + \frac{\tau_e}{\tau_m - \tau_e} e^{-T/\tau_e} - \frac{\tau_m}{\tau_m - \tau_e} e^{-T/\tau_m} \right) \quad (31)$$

$$b_2 = K_m \left(a_2 - \frac{\tau_m}{\tau_m - \tau_e} e^{-T/\tau_e} + \frac{\tau_e}{\tau_m - \tau_e} e^{-T/\tau_m} \right) \quad (32)$$

where T is the sampling time. The linear difference equation of the process is

$$\begin{aligned} \omega_m(k) = & -a_1\omega_m(k-1) - a_2\omega_m(k-2) + b_1i_{qs}^*(k-1) \\ & + b_2i_{qs}^*(k-2), \end{aligned} \quad (33)$$

To be able to estimate the parameters a_1, a_2, b_1 , and b_2 , the difference equation is expressed in a predictor form as

$$\hat{\omega}_m(k) = \varphi^T(k-1)\hat{\theta}(k) \quad (34)$$

where the regression vector φ is

$$\begin{aligned} \varphi(k) &= [-\omega_m(k-1) \quad -\omega_m(k-2) \quad i_{qs}^*(k-1) \quad i_{qs}^*(k-2)]^T \\ & \quad (35) \end{aligned}$$

and $\hat{\theta}(k)$ is an estimated vector of the system parameters

$$\theta = [a_1 \quad a_2 \quad b_1 \quad b_2]^T. \quad (36)$$

The estimation has mostly been done using the recursive least-squares method with exponential forgetting factor. However, this method works well only if the process is properly excited all the time. This is not realistic for servo systems because there may be long time intervals without excitation from the reference input and, hence, a problem called estimator windup will occur [13]. To avoid the estimator windup, one version of the regularized constant-trace algorithm is adopted for the model parameter estimation. The estimator is given as [14]

$$\hat{\theta}(k) = \hat{\theta}(k-1) + a(k)K(k)(\omega_m(k) - \varphi^T(k)\hat{\theta}(k-1)) \quad (37)$$

$$K(k) = \frac{P(k-1)\varphi(k)}{(1 + \varphi^T(k)P(k-1)\varphi(k) + c\varphi^T(k)\varphi(k))} \quad (38)$$

$$\bar{P}(k) = \bar{P}(k-1) - a(k)K(k)\varphi^T(k)P(k-1) \quad (39)$$

$$P(k) = c_1 \frac{\bar{P}(k)}{\text{tr}(\bar{P}(k))} + c_2 I \quad (40)$$

$$a(k) = \begin{cases} a' & |\omega_m(k) - \varphi^T(k)\hat{\theta}(k-1)| > 2\delta \\ 0 & \text{otherwise} \end{cases} \quad (41)$$

where $c_1 > 0, c_2 \geq 0, c \geq 0$; the variable $a(k)$ introduces a dead zone in the estimator, and δ is a magnitude estimate of the noise. In this way, the trace of the covariance matrix $P(k)$ is kept to be constant in each iteration to provide the adaptation capability for parameter change. Further, the dead zone prevents the estimator from drifting when the errors are small and when the system is not persistently excited. Typically, the estimator parameters are selected such that $a' \in [0, 1, 0.5]$ and $c_1/c_2 \approx 10^4$.

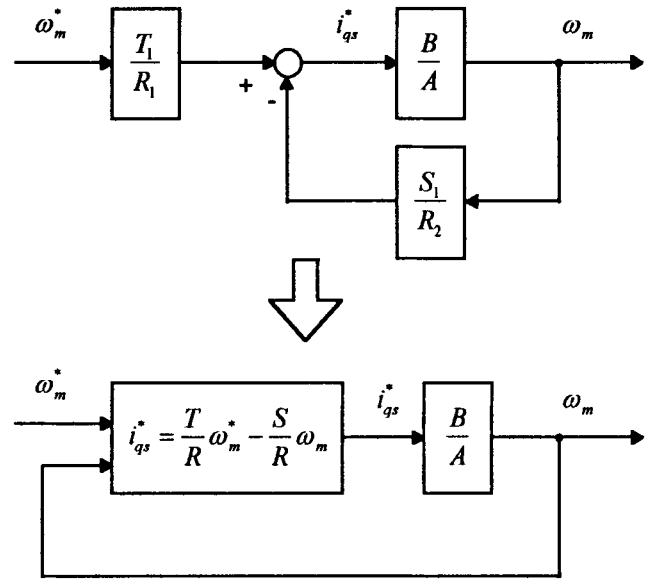


Fig. 4. The 2-degree-of-freedom structure of the pole-placement controller.

B. Controller Adaptation

The design of the speed controller is based on pole placement with polynomial manipulation technique for its simplicity and easy-to-use nature [14]. According to this method, the speed controller is considered to be a two-degree-of-freedom structure, as shown in Fig. 4. The controller has one output, the torque-producing current command, i_{qs}^* , and two inputs, the speed command, ω_m^* , and the motor speed, ω_m . The speed controller with these inputs and outputs can be represented by

$$i_{qs}^*(k) = \frac{T_1(z)}{R_1(z)}\omega_m^*(k) - \frac{S_1(z)}{R_2(z)}\omega_m(k) \quad (42)$$

where $R_1(z), R_2(z), T_1(z)$, and $S_1(z)$ are polynomials in z . The control law of (42) can be written as

$$R(z)i_{qs}^*(k) = T(z)\omega_m^*(k) - S(z)\omega_m(k) \quad (43)$$

where $R(z) = R_1(z)R_2(z), T(z) = T_1(z)R_2(z), S(z) = S_1(z)R_1(z)$, and $R(z)$ is monic, i.e., the coefficient of the highest power in $R(z)$ is unity. Combing the speed controller (43) with the pulse-transfer function of the motor drive (28) yields the closed-loop pulse-transfer function as follows:

$$G_{cl}(z) = \frac{B(z)T(z)}{A(z)R(z) + B(z)S(z)}. \quad (44)$$

To carry out the speed controller adaptation, the parameters of the controller polynomial are obtained by solving the following Diophantine equation:

$$A(z)R(z) + B(z)S(z) = A_m(z)A_o(z) \quad (45)$$

and

$$T(z) = \frac{B_m(z)A_o(z)}{B(z)} \quad (46)$$

where A_m is the desired closed-loop characteristic polynomial, A_o is the canceled-out polynomial in (44), and the controller polynomials are expressed as

$$R(z) = (z + r)(z - 1) \quad (47)$$

$$S(z) = s_0 z^2 + s_1 z + s_2 \quad (48)$$

$$T(z) = t_0 z^2 + t_1 z + t_2. \quad (49)$$

One should note that the $R(z)$ in (47) contains an integral factor which is required for zero steady-state error. Specification of the system can be expressed in terms of the desired closed-loop transfer function as

$$G_m(z) = \frac{B_m(z)}{A_m(z)} \quad (50)$$

where A_m and B_m do not have any common factors.

For ease of the design, the desired properties of the closed-loop system can be first specified in the s -domain by requiring that the response to a command signal has the characteristic polynomial

$$s^2 + 2\zeta\omega_n s + \omega_n^2$$

where ζ is the damping ratio and ω_n is the natural frequency. Then, the equivalent discrete-time polynomial with the sampling time T can be derived and given by

$$\begin{aligned} A_m(z) &= z^2 + 2\zeta e^{-\zeta\omega_n T} \cos(\omega_n T \sqrt{1 - \zeta^2}) z + e^{-2\zeta\omega_n T} \\ &= z^2 + p_1 z + p_2. \end{aligned} \quad (51)$$

The roots of the cancellation polynomial can be assigned at $s = -\alpha$, which is on the negative real axis with dynamic faster than the desired closed-loop response determined by A_m . Then, the discrete-time cancellation polynomial is

$$A_o(z) = (z - e^{-\alpha T})^2 = z^2 + q_1 z + q_2. \quad (52)$$

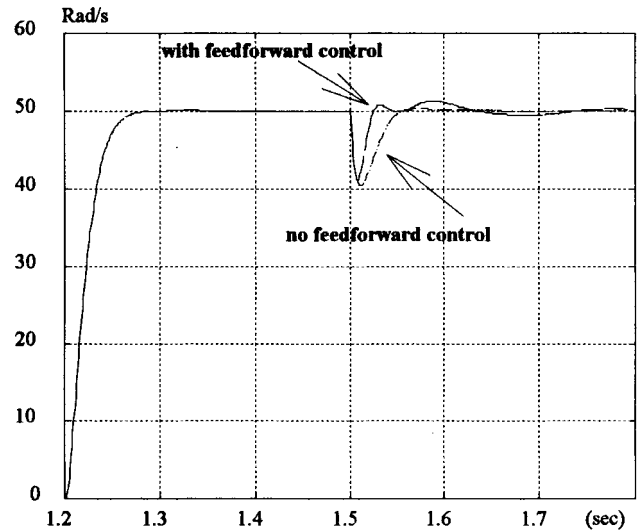
The controller parameters then can be obtained by solving the Diophantine equation (45) and (46). The control law derived from (42) becomes

$$\begin{aligned} i_{qs}^*(k) &= t_0 \omega_m^*(k) + t_1 \omega_m^*(k-1) + t_2 \omega_m^*(k-2) \\ &\quad - s_0 \omega_m(k) - s_1 \omega_m(k-1) - s_2 \omega_m(k-2) \\ &\quad - (r-1) i_{qs}^*(k-1) + r i_{qs}^*(k-2). \end{aligned} \quad (53)$$

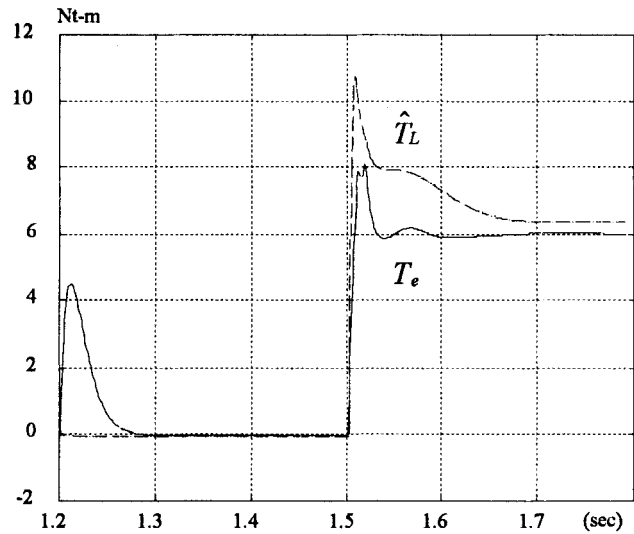
C. Load Torque Estimation and Compensation

Similar to the model parameter estimation method described above, estimating load torque entails a recursive least-squares algorithm. As can be seen from Fig. 3, the transfer function from the torque-producing current i_{qs} to the motor speed output ω_m is a first-order system with the form

$$G_p(s) = \frac{\omega_m(s)}{i_{qs}(s)} = \frac{K_T}{J s + B}. \quad (54)$$



(a)



(b)

Fig. 5. Simulation results for a load torque of 6 N·m suddenly applied at 1.5 s. (a) Speed response with feedforward compensation (solid line) and without feedforward compensation (dash line). (b) Producing-torque response with feedforward compensation (solid line) and the estimated load-torque convergence behavior (dash line).

TABLE I
PARAMETERS OF THE TESTED INDUCTION MOTOR.

Three Phases, Two Poles, Y-Winding, 800W					
L_s	0.145 H	R_s	1.1 Ω	J_m	0.0027 $\text{Kg} \cdot \text{m}^2$
L_r	0.145 H	R_r	1.3 Ω	B_m	0.000058 $\text{Kg} \cdot \text{m} / \text{rad} / \text{s}$
L_m	0.136 H				

The discrete-time equation with load torque disturbance can be obtained through a zeroth-order holder as follows:

$$\omega_m(k) + a\omega_m(k-1) = b i_{qs}(k-1) - b T_L(k-1) / K_T \quad (55)$$

where $a = -e^{-BT/J}$ and $b = (1+a)K_T/B$, in which T

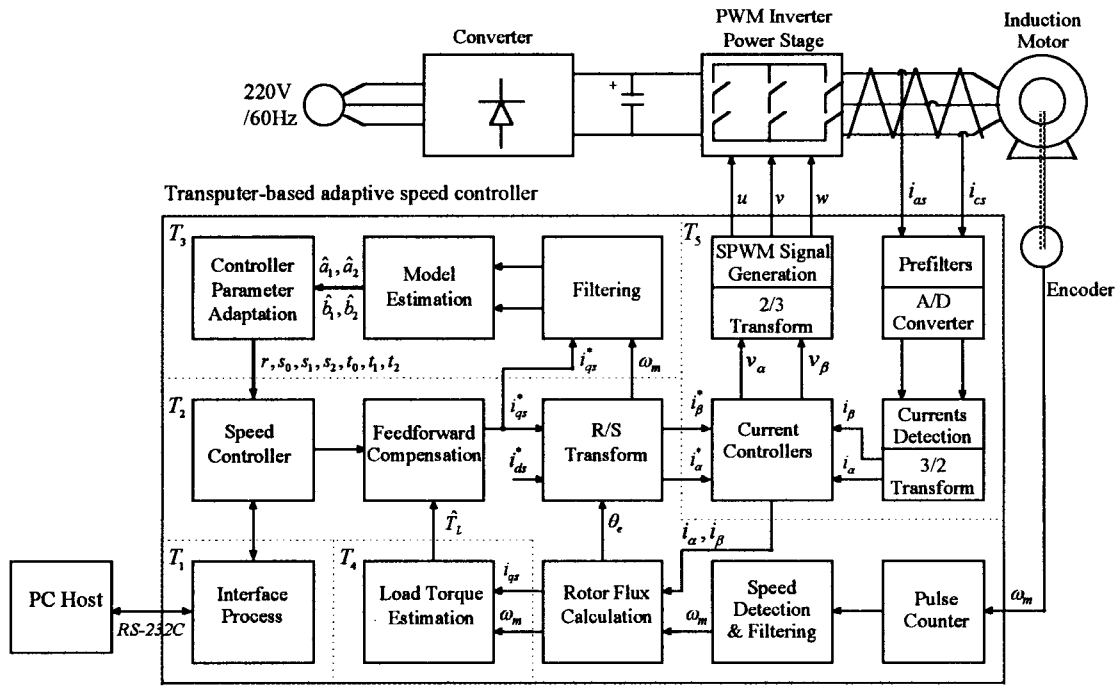


Fig. 6. Configuration of the transputer-based adaptive speed control system.

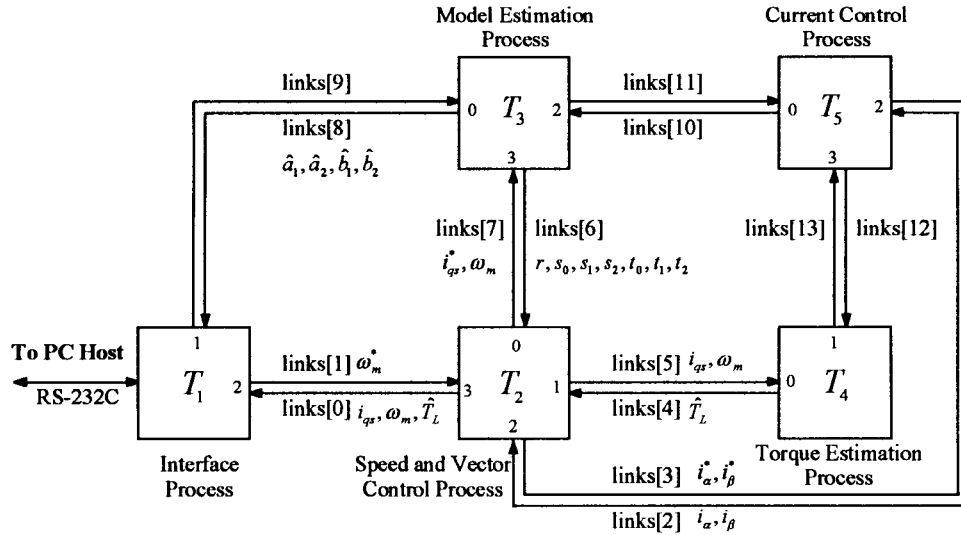


Fig. 7. Transputer topology for the parallel implementation.

is the sampling time. To estimate the load torque, we rewrite (55) into the same form as (34), but here the regression vector $\varphi(k)$ and estimated parameter vector $\hat{\theta}(k)$ are, respectively, as follows:

$$\varphi(k) = [-\omega_m(k-1) \quad i_{qs}(k-1) \quad -1]^T \quad (56)$$

and

$$\hat{\theta} = [\hat{a} \quad \hat{b} \quad \hat{c}]^T. \quad (57)$$

Then, the load torque can be estimated as

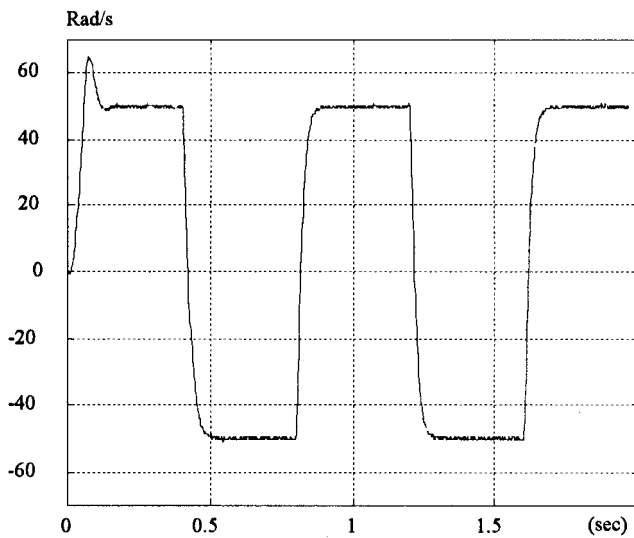
$$\hat{T}_L = \frac{\hat{c}K_T}{\hat{b}}. \quad (58)$$

In order to immediately detect when a slow or sudden change of torque disturbance has occurred and take appropriate

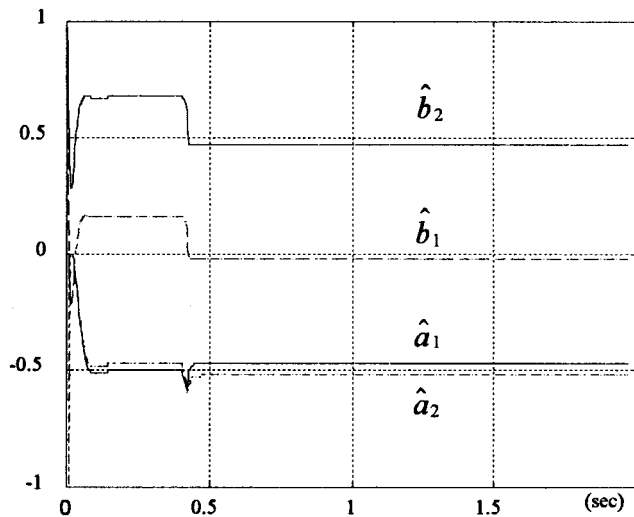
action, we adopt the least-squares algorithm with variable forgetting factor. This algorithm is

$$\hat{\theta}(k) = \hat{\theta}(k-1) + \frac{P(k-2)\varphi(k-1)}{\lambda(k-1) + \varphi^T(k-1)P(k-2)\varphi(k-1)} \cdot [\omega_m(k) - \varphi^T(k-1)\hat{\theta}(k-1)] \quad (59)$$

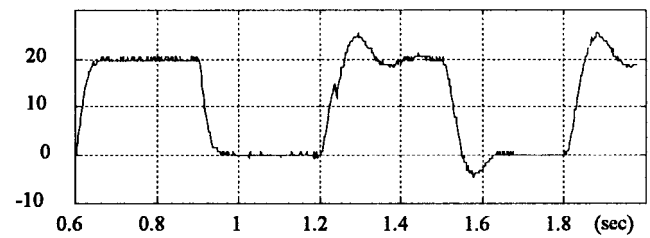
$$= \frac{1}{\lambda(k-1)} \times \left[P(k-2) - \frac{P(k-2)\varphi(k-1)\varphi^T(k-1)P(k-2)}{\lambda(k-1) + \varphi^T(k-1)P(k-2)\varphi(k-1)} \right] \quad (60)$$



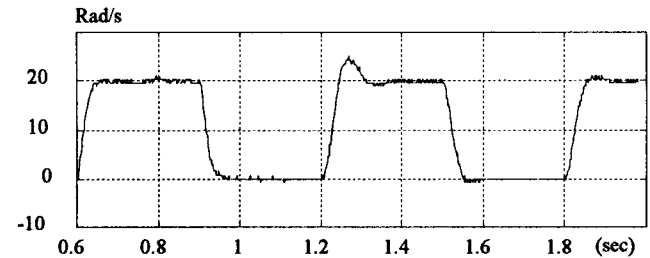
(a)



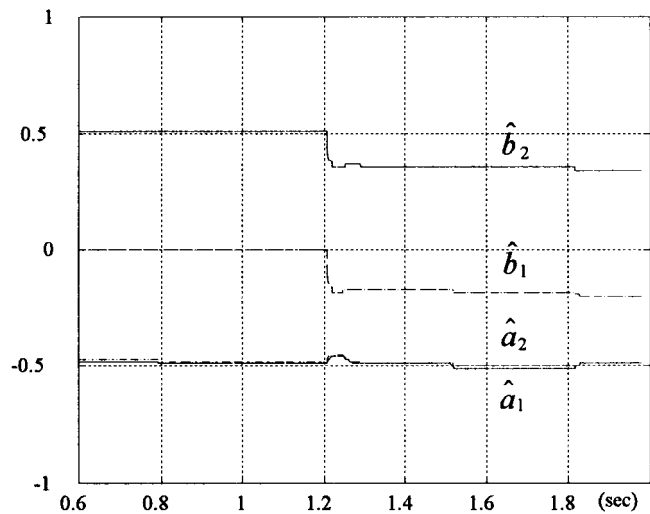
(b)



(a)



(b)



(c)

Fig. 8. Experimental results of the adaptive speed control response. (a) Motor shaft speed. (b) Model parameters' convergence behavior.

where the forgetting factor $\lambda(k)$ is

$$\lambda(k) = 1 - \alpha' \frac{\varepsilon(k)^2}{1 + \varphi^T(k-1)P(k-2)\varphi(k-1)}. \quad (61)$$

In (61), $\varepsilon(k)$ is the prediction error and α' is a tuning parameter. Thus, if a torque disturbance occurs, the prediction error square increases; this reduces $\lambda(k)$ temporarily but increases the covariance matrix $P(k)$ quickly so that rapid adaptation can occur. After adaptation, the prediction error square decreases and $\lambda(k)$ returns a value near one. The prediction error square can be further monitored to detect a sudden change of load torque. If the prediction error square exceeds a preset threshold value, we can reset the covariance matrix for quick adaptation to get the true parameter of the load torque estimation.

For getting fast convergence of the estimated parameters, the sampling rate can be set to be two or more times of that in the speed control loop. The estimated load torque then is

Fig. 9. Comparison of the performance for the system inertia suddenly increased just prior to 1.2 s. (a) Without adaptation. (b) With adaptation. (c) Model parameters' convergence behavior with adaptation.

directly added to output of the 2-degree-of-freedom controller for feedforward compensation.

Before the implementation of the proposed adaptive controller, the computer simulation is made to test the effectiveness of the control scheme. The motor parameters used are listed in Table I. The selected parameters of the model estimator (37)–(41) are $c = 0.1$, $c_1 = 10.0$, $c_2 = 0.001$, $\alpha' = 0.3$, and $\delta = 0.1$. The selected parameter of (61) in torque estimator is $\alpha' = 0.2$ and the threshold value of prediction error is 0.6. The design specifications are $\omega_n = 94.2$ Rad/s and $\zeta = 1.0$. Fig. 5(a) shows the comparison between the performance with and without feedforward load-torque compensation when a load torque of 6 N·m is suddenly applied at the time of 1.5 s. As can be seen, the speed response with reduced deviation is achieved by using the feedforward compensation. The corresponding load-torque estimation convergence behavior and the motor developed-torque response are shown in Fig. 5(b).

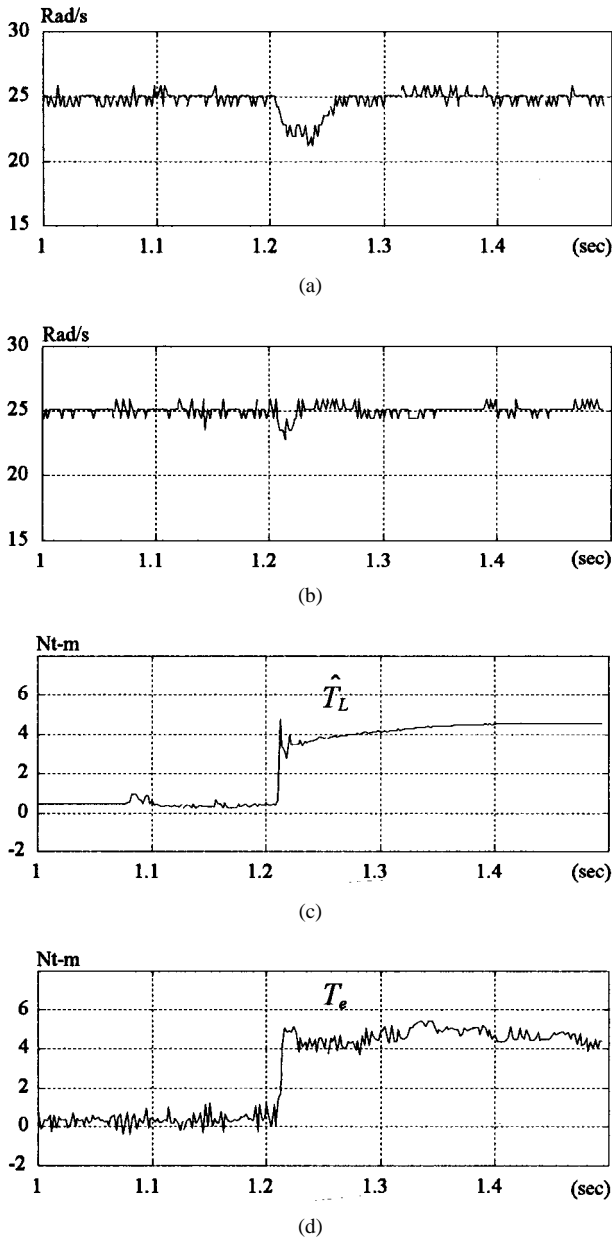


Fig. 10. Comparison of the performance for the load torque suddenly increased. (a) Without feedforward compensation. (b) With feedforward compensation. (c) The estimated load torque. (d) The motor developed torque with feedforward compensation.

IV. TRANSPUTER IMPLEMENTATION AND EXPERIMENTAL RESULTS

Fig. 6 illustrates the configuration of the transputer-based adaptive speed control system of an ac induction motor drive. The induction motor employed in this paper is a NIKKI product NA20-40F, the parameters of which are shown in Table I. The load torque change is implemented by employing a loaded dc generator that is directly coupled to the rotor shaft of the induction motor. A close contact on the output terminals of the dc generator, which is in series with a current-limiting resistor for protection, can produce a constant torque proportional to the motor speed on the counter rotary direction of the motor. The system inertia change is implemented by employing a rigid cylinder load that is connected to the

TABLE II
 COMPUTATION TIME OF THE PARTITIONED CONTROL TASKS.

Transputer (T800-20)	Task	Computation Time (μ sec)	Total (μ sec)
T_2	Speed Detection	20	238
	Speed Compensation	28	
	Coordinate Transform	168	
	Rotor Flux Calculation	22	
T_3	Model Estimation	120	172
	Controller Adaptation	52	
T_4	Load Torque Estimation	90	90
T_5	Current Detection	42	86
	Current Regulation	20	
	SPWM Generation	24	

induction motor shaft through an electromagnetic clutch. The load can be immediately attached or detached to the motor by controlling a dc power supply, which is applied to the clutch. The moment inertia of the rigid cylinder load is $0.0064 \text{ kg}\cdot\text{m}^2$, which is about 2.2 times that of the motor rotor. The position and speed of the motor shaft are detected by a rotary encoder with 2000 pulse/revolution. The stator currents are measured by using Hall-effect sensors and then converted to digital values by a multiplexed 12-b analog-to-digital converter (AD1678). The induction motor is driven by a current-controlled POWER-MOSFET-based voltage source inverter with the current limit of 18 A.

Instead of using INMOS-provided transputer boards, two transputer-based modules, one called I/O module (IOM) and the other called parallel computing module (PCM), were developed to build the system [15]. The transputer is not only a powerful microcomputer (10 MIPS and 1.5 MFLOPS for IMS T800-20), but also acts as a parallel processing element. Parallelism is supported by the transputer intrinsic hardware and software capabilities. Each transputer has four independent serial links which provide point-to-point connections between transputers with communication rate up to 20 Mb/s. In order to reduce the computation delay, we employed five transputers for the implementation. Transputer T_2 performs the speed detection and filtering, the speed compensation, and the vector control which contains the rotor flux calculation and the rotating-to-stationary coordinate transformation. T_3 performs the model estimation and the controller adaptation. The estimation of the external load torque is assigned to T_4 . T_5 performs the current detection, the current regulation, and the sinusoidal PWM signal generation. T_1 acts as the system supervisor and provides the user interface and data storage. Through an RS-232C communication interface, the software and data can be transmitted between the transputer-based adaptive control system and the host computer. The interprocess links of transputers are two-wire connections by which a grid structure is obtained, as shown in Fig. 7. The parallel structure planning is configured by the software in the high-level programming language, Occam [16].

The selection of the sampling time for each process invariably relates to the control dynamics and the computation time needed for the task execution. Table II shows the task computation time of the partitioned control algorithm. To achieve fast convergence of the load torque estimation, transputer T_4 performs the software routine every $200 \mu\text{s}$. To get fast current loop response, the sampling time in T_5 is $100 \mu\text{s}$. The software routine in T_2 and T_3 are executed every one ms.

Based on the constructed system and the selected parameters of the adaptive control scheme described above, the experimental results are shown in Figs. 8–10. Fig. 8 gives the dynamic motor speed response to a square wave command with period of 800 ms and magnitude from -50 to 50 rad/s to provide the necessary excitation for model estimation. One can see that on the first excitation the four parameters $(\hat{a}_1, \hat{a}_2, \hat{b}_1, \hat{b}_2)$ are convergent from the initial guess $(0, 0, 1, 1)$ to $(-0.50, -0.47, 0.16, 0.68)$ which makes that the speed output error is zero. On the next excitation, the four parameters are convergent to the true parameters $(-0.47, -0.52, -0.02, 0.47)$ and, hence, the speed output reaches the desired response. Fig. 9 presents a comparison between the performance with and without adaptation, if the load is suddenly applied just prior to the time of 1.2 s, at which the reference input has a step change, and the parameters transient behavior. The speed response without adaptation, as shown in Fig. 9(a), is kept well before the load is attached. But while the load is attached, overshoots occur when the reference input has step changes. On the contrary, as shown in the Fig. 9(b), for the performance with adaptation, only an overshoot occurs at the first excitation (1.2 s) while the load is attached because of the parameters transient behavior from $(-0.47, -0.52, -0.02, 0.47)$ to $(-0.49, -0.49, -0.17, 0.36)$. No more overshoots occur at the subsequent excitations while the load remains attached and, hence, the performance is improved. Fig. 10 gives a comparison between the performance with and without feedforward compensation for a sudden increase of load torque from 0.5 to 4.5 N·m, indicating that the performance with feedforward compensation is more robust than that without feedforward compensation in contending with the load torque disturbance.

V. CONCLUSION

This paper has developed a transputer-based adaptive speed controller with the addition of estimator-based load torque compensation for high-performance control of an ac induction motor. For reducing the computation delay and enhancing the overall system performance, we employed four transputers to execute the control algorithm in parallel, which contains the model parameter estimation, the controller adaptation, the speed compensation, the vector control, the current control and sinusoidal PWM, and the load torque estimation, and one transputer for the user interface processing. Estimating the load torque entails a recursive least-squares method rather than the conventional load-torque observation technique. Experimental results demonstrate the robustness of the proposed adaptive speed control method in contending with varying load and torque disturbance.

REFERENCES

- [1] K. Ohishi, K. Ohnishi, and K. Miyachi, "Adaptive DC servo drive control taking force disturbance suppression into account," *IEEE Trans. Ind. Applicat.*, vol. 24, pp. 171–176, Jan./Feb. 1988.
- [2] A. Chandra, L. Dessaint, M. Saad, and K. Al-Haddad, "Implementation of self-tuning algorithms for reference tracking of a DC drive using a DSP chip," *IEEE Trans. Ind. Electron.*, vol. 41, pp. 104–109, Feb. 1994.
- [3] K. H. Gurubasavaraj, "Implementation of a self-tuning controller using digital signal processor chips," *IEEE Contr. Syst. Mag.*, pp. 38–42, June 1989.
- [4] E. Bassi, F. Benzi, and R. Scattolini, "Design of a digital adaptive controller for electrical drives in industrial applications," *IEEE Trans. Ind. Electron.*, vol. 39, pp. 120–127, Apr. 1992.
- [5] C. M. Liaw, C. T. Pan, and Y. C. Chen, "Design and implementation of an adaptive controller for current-fed induction motor," *IEEE Trans. Ind. Electron.*, vol. 35, pp. 393–401, Aug. 1988.
- [6] R. B. Sepe and J. H. Lang, "Real-time adaptive control of the permanent-magnet synchronous motor," *IEEE Trans. Ind. Applicat.*, vol. 27, pp. 706–714, July/Aug. 1991.
- [7] Y. P. Yang and C. F. Fang, "Adaptive speed control of AC servo induction motors with on-line load estimation," in *Conf. Rec. R.O.C. Automatic Control Conf.*, Taoyuan, Taiwan, R.O.C., 1994, pp. 481–486.
- [8] H. J. Wu, M. F. Tsai, and Y. Y. Tzou, "Transputer-based fully digital adaptive control system for high-performance AC induction motor drives," in *Conf. Rec. IEEE PESC Conf.*, Taipei, Taiwan, R.O.C., 1994, pp. 1250–1256.
- [9] T. R. Fortescue, L. S. Kershenbaum, and B. E. Ydstie, "Implementation of self-tuning regulators with variable forgetting factors," *Automatica*, vol. 17, no. 6, pp. 831–835, 1981.
- [10] K. Ohishi, M. Nakao, K. Ohnishi, and K. Miyachi, "Microprocessor-controlled DC motor for load-insensitive position servo system," *IEEE Trans. Ind. Electron.*, vol. 34, pp. 44–49, Feb. 1987.
- [11] INMOS, *Transputer Reference Manual*, 1988.
- [12] F. Blaschke, "The principle of field orientation as applied to the new TRANSVECTOR closed-loop control system for rotating field machines," *Siemens Rev.*, vol. 39, no. 5, pp. 217–220, May 1972.
- [13] B. Wittenmark and K. Astrom, "Practical issues in the implementation of self-tuning control," *Automatica*, vol. 20, no. 5, pp. 595–605, 1984.
- [14] K. J. Astrom and B. Wittenmark, *Adaptive Control*. Reading, MA: Addison Wesley, 1989.
- [15] M. F. Tsai and Y. Y. Tzou, "A transputer-based flexible parallel-processing system architecture for high-performance control applications," in *Proc. AUTOMATION*, Taipei, Taiwan, R.O.C., 1994, vol. 3, pp. 145–151.
- [16] INMOS, *Occam 2 Reference Manual*, 1988.



Ming-Fa Tsai (S'93) was born in Taiwan, Republic of China, on April 17, 1958. He received the B.S. and M.S. degrees from the Department of Electronic Engineering and Control Engineering, National Chiao Tung University, Hsinchu, Taiwan, in 1980 and 1990, respectively. He is currently pursuing the Ph.D. degree at National Chiao Tung University and is engaged in the research of motor control.

In 1980, he joined the Chinese Army for a two-year period, after which he was with the Chung Shan Institute of Science and Technology, Lungtan, Taiwan, for four years. His research interests also include adaptive control theory, parallel processing techniques, and power electronic systems.



Ying-Yu Tzou (S'82–M'88) was born in Taiwan, Republic of China, on February 13, 1956. He received the B.S. and M.S. degrees in control engineering from National Chiao Tung University, Hsinchu, Taiwan, and the Ph.D. degree in electrical engineering from the Institute of Electronics Engineering, National Chiao Tung University, in 1978, 1983, and 1987, respectively.

During 1980–1981 he was with the Control System Department, Electronic Research and Service Organization (ERSO), Industry Technology Research Institute (ITRI), as a Design Engineer. During 1983–1986 he was with Microtek Automation, Inc. as a Project Manager working on the development of a computer numerical controller (CNC) for machine tools. He is currently an Associate Professor, Department of Control Engineering, National Chiao Tung University, and also serves as an Industrial Consultant for many local power electronics and automation companies. He was the Director of the Institute of Control Engineering during 1992–1994. His special interests are sensorless ac drives, intelligent UPS, FPGA based control IC's for motor drives, and DSP applications in power electronics and motion control.