Model-Based Current Control of AC Machines Using the Internal Model Control Method

Lennart Harnefors, *Member, IEEE,* and Hans-Peter Nee, *Member, IEEE*

***Abstract*— In this paper, the internal model control (IMC) method is applied to ac machine current control. Permanent magnet synchronous machines and induction machines are considered. The result is synchronous-frame proportional integral (PI) or PI-type controllers, the parameters (gain and integration time) of which are expressed directly in certain machine parameters and the desired closed-loop bandwidth. This simplifies the controller design procedure, eliminating or reducing the need for trial-and-error steps, and is the main purpose for using IMC.**

***Index Terms*—Current control, internal model control, inverter-fed machines.**

I. Introduction

THERE have been several strategies proposed for ac machine current control. Some well-known and simple methods are hysteresis control, stator-frame proportional integral (PI) control, and synchronous-frame PI control [1, ch. 4, 5]. Of these, the latter has been acknowledged as being superior, as unlike hysteresis control, it allows a fixed switching frequency and, unlike stator-frame PI control, it yields zero control error at steady state.

More advanced schemes—in the sense that improved performance in some respect, e.g., faster response or lower harmonic content is gained at the expense of higher controller complexity than for the above ones—have also been proposed, including predictive control [2], generalized predictive control [3], state-variable regulator [4], state feedback [5], and the promising new concept of minimum time control [6].

In this paper, the internal model control (IMC) method [7] is introduced and applied to ac machine current control. A permanent magnet synchronous machine (PMSM) is the working example. It is also shown how IMC can be applied to induction machines (IM's). The main benefits of IMC are the following.

1) Synchronous-frame PI or PI-type current controllers are obtained.
2) The controller parameters (gain and integration time) are expressed directly in certain machine parameters and the desired closed-loop bandwidth. This simplifies the design procedure and trial-and-error steps are avoided.

Another benefit is that the cross coupling which exists between the d and q components of the stator voltage and current in synchronous coordinates can be compensated for easily. For standard synchronous-frame PI control, this cross coupling has the following effect. When the setpoint (reference) for i_q is changed, a transient error in i_d results, since the PI controllers work in two single loops and are capable of suppressing the cross coupling only at steady state. To combat this, decoupling networks in the current controllers have been suggested [8]. However, as pointed out in [6], a transient error in i_d may not be very serious. For a field-oriented IM, it goes unnoticed due to slow flux dynamics. Neither does it affect the torque for a round rotor PMSM (there is no reluctance torque). The approach in [6] is, therefore, to allow a fairly large transient error in i_d in exchange for a faster response in i_q. Yet, there are certainly cases where removal of cross coupling is desirable and/or extremely fast response of i_q is not required. We shall, therefore, aim at removing the dq cross coupling, but not make this endeavor a major issue. (A good topic for future research would be to try to combine the simplicity of IMC design with the good properties of the control scheme in [6].)

IMC has been considered for current control previously [9]. The main difference between [9] and this paper is stated in Section III-A.

This paper is organized as follows. The PMSM model used for controller design is introduced in Section II. IMC is introduced in Section III, and it is shown how the method can be applied to controller design for the PMSM and the IM. IMC is also analyzed with respect to sensitivity. Discrete-time implementation issues, including selection of the sampling and switching frequencies, are considered in Section IV. Simulations and experimental results are presented in Sections V and VI, respectively.

II. The Machine Model

In this section, we consider a model for the permanent magnet synchronous machine which is suitable for controller design. In Section III-C, it will be shown how the obtained model can be modified and applied also to the induction machine. The PMSM is modeled in rotor coordinates (the dq

Paper IPCSD 97–62, presented at the 1995 Industry Applications Society Annual Meeting, Lake Buena Vista, FL, October 8–12, and approved for publication in the IEEE Transactions on Industry Applications by the Industrial Drives Committee of the IEEE Industry Applications Society. This work was supported in part by the NUTEK Center of Excellence in Electric Power Engineering. Manuscript released for publication September 12, 1997.

L. Harnefors is with the Division of Electrical Machines and Drives, Department of Electric Power Engineering, Royal Institute of Technology, SE-100 44 Stockholm, Sweden (e-mail: lennarth@ekc.kth.se) and with the Department of Electrical Engineering, Mälardalen University, SE-721 23 Västerås, Sweden.

H.-P. Nee is with the Division of Electrical Machines and Drives, Department of Electric Power Engineering, Royal Institute of Technology, SE-100 44 Stockholm, Sweden.

Publisher Item Identifier S 0093-9994(98)01210-9.

frame). From [10], we obtain

$$v_d(t) = R_s i_d(t) - \omega_r L_q i_q(t) + L_d \frac{di_d(t)}{dt} \quad (1)$$

$$v_q(t) = R_s i_q(t) + \omega_r L_d i_d(t) + L_q \frac{di_q(t)}{dt} + \omega_r \psi_m \quad (2)$$

where v_d and v_q are the applied stator voltages, i_d and i_q the stator currents, R_s the stator resistance, ω_r the rotor speed, and L_d and L_q the d and q direction inductances, respectively. The term $\omega_r \psi_m$ is the back EMF. In (1) and (2), we clearly see that cross-coupling (interaction) terms exist, namely, $-\omega_r L_q i_q(t)$ and $\omega_r L_d i_d(t)$.

The system (1) and (2) is *multivariable;* there are two inputs and outputs (d and q voltages and currents, respectively). We shall rewrite (1) and (2) using a *transfer function matrix* [11], which is a generalization for multivariable systems of the standard transfer function description. This technique is rarely used for electrical machines and drives. However, we shall shortly see that matrix descriptions are very convenient. Let us apply the Laplace transform to (1) and (2), regarding the permanent magnet flux term as a quasi-constant auxiliary input signal which is included in the q direction voltage: $\boxed{v'_q = v_q - \omega_r \psi_m,}$ while letting $i_d(0) = i_q(0) = 0$. The rotor speed ω_r is considered as a slowly varying parameter as seen from the electrical dynamics.[1] Then

$$V_d(s) = (R_s + sL_d) I_d(s) - \omega_r L_q I_q(s) \quad (3)$$

$$V'_q(s) = (R_s + sL_q) I_q(s) + \omega_r L_d I_d(s). \quad (4)$$

Introducing Laplace transformed input and output vectors as follows (where U and Y are chosen instead of V and I for a notation consistent with that used in automatic control):

$$U(s) = \begin{bmatrix} V_d(s) \\ V'_q(s) \end{bmatrix}, \quad Y(s) = \begin{bmatrix} I_d(s) \\ I_q(s) \end{bmatrix} \quad (5)$$

we obtain

$$Y(s) = G(s) U(s) \quad (6)$$

where

$$\begin{aligned} G(s) &= \begin{bmatrix} sL_d + R_s & -\omega_r L_q \\ \omega_r L_d & sL_q + R_s \end{bmatrix}^{-1} \\ &= \frac{1}{p(s)} \begin{bmatrix} sL_q + R_s & \omega_r L_q \\ -\omega_r L_d & sL_d + R_s \end{bmatrix} \end{aligned} \quad (7)$$

and $p(s) = s^2 L_d L_q + s(L_d + L_q) R_s + R_s^2 + \omega_r^2 L_d L_q$. We also introduce the following notation, which will be used in the following:

$$G^{-1}(s) = \underbrace{\begin{bmatrix} sL_d + R_s & 0 \\ 0 & sL_q + R_s \end{bmatrix}}_{D(s)} + \underbrace{\begin{bmatrix} 0 & -\omega_r L_q \\ \omega_r L_d & 0 \end{bmatrix}}_{W}. \quad (8)$$

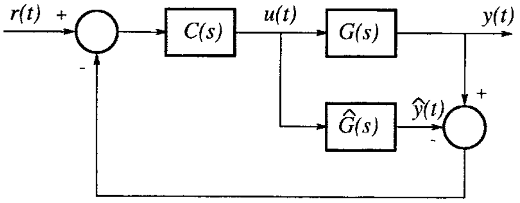


Fig. 1. IMC structure.

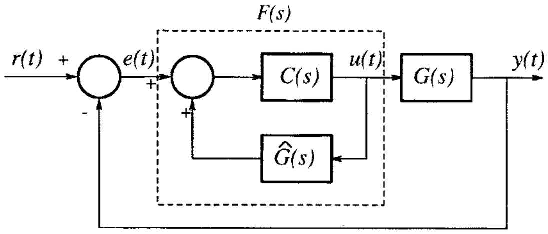


Fig. 2. Classic control system structure.

III. Internal Model Control

IMC was originally developed for chemical engineering applications [7] and is considered as a *robust control* method. Before applying IMC to the current control problem, a general presentation of the method is given.

The IMC structure is depicted in Fig. 1. The structure uses an internal model $\hat{G}(s)$ in parallel with the controlled system (plant) $G(s)$. For an ac machine, u and y are, thus, the stator voltage and current vectors, respectively, while $r = [i_d^{\text{ref}}, i_q^{\text{ref}}]^T$ is the current setpoint (reference) vector. The control loop is augmented by a block $C(s)$; the so-called *IMC controller.* $G(s)$, $\hat{G}(s)$, and $C(s)$ are all transfer function matrices.

There are some interesting facts worth noting about the IMC structure.

1) IMC can be considered as a special case of the classic structure shown in Fig. 2. The controller $F(s)$ in this system is related to the internal model $\hat{G}(s)$ and the IMC controller $C(s)$ in the following way:

$$F(s) = [I - C(s)\hat{G}(s)]^{-1} C(s) \quad (9)$$

where I is the identity matrix. Thus, $U(s) = F(s)[R(s) - Y(s)]$. To obtain integral action, which is a necessity if steady-state errors are to be avoided, $I - C(0)\hat{G}(0)$ should be zero. We thus need to let $C(0)\hat{G}(0) = I$.

2) It should be noted that, if the internal model is perfect, i.e., $\hat{G}(s) = G(s)$, there is no feedback in Fig. 1, and the closed-loop system has the transfer function matrix

$$G_c(s) = G(s)C(s). \quad (10)$$

Hence, the closed-loop system is stable if and only if $G(s)$ and $C(s)$ each are stable, implying that we have found *all* stabilizing controllers $F(s)$—given by (9)—for the stable plant $G(s)$. This is a special case of the *Youla parameterization* [11]. In this case, it is tempting to let $C(s) = G^{-1}(s)$, giving $G_c(s) = I$, i.e., all plant dynamics would be canceled and the output signal $y(t)$

[1] Current rise times of about 1 ms are common, whereas speed rise times as short as 10 ms are rare. Hence, there is normally a separation in dynamics of at least one decade between stator current and rotor speed, so the mechanical dynamics can be disregarded, as seen from the electrical dynamics.

would attain the setpoint $r(t)$ instantaneously. It is clear that this optimal result cannot be accomplished due to a variety of reasons:

a) If $G(s)$ is not minimum phase, i.e., there are zeros in the right-half plane, $G^{-1}(s)$ will be unstable.

b) $G^{-1}(s)$ is hardly ever *proper*, i.e., the degrees of the numerators of the elements are higher than the degrees of the denominators. $G^{-1}(s)$ cannot then be implemented.

c) Any attempt to cancel the plant dynamics will give large magnitudes of the control signals.

d) The method is highly sensitive to model errors.

However, with a few modifications, the idea can be used. Let us factor $G(s)$ as

$$G(s) = G_A(s)G_M(s) \tag{11}$$

where $G_A(s)$ is the *allpass* part of $G(s)$, including all zeros in the right-half plane and all time delays. We could then make

$$C(s) = G_M^{-1}(s) \tag{12}$$

which is the choice of $C(s)$ obtained by the H_2 *optimization procedure* [7]. However, this resolves only the first item on the above list. We therefore *detune* the optimal controller with a low-pass filter $L(s)$

$$C(s) = G_M^{-1}(s)L(s). \tag{13}$$

This is generally chosen diagonal

$$L(s) = \operatorname{diag}\left[\frac{\alpha_1^n}{(s+\alpha_1)^n}, \frac{\alpha_2^n}{(s+\alpha_2)^n}, \cdots, \frac{\alpha_{n_y}^n}{(s+\alpha_{n_y})^n}\right] \tag{14}$$

where the positive integer n is chosen sufficiently large, so that $C(s)$ becomes proper. Now, the closed-loop system can be made arbitrarily robust simply by making α_i smaller.

A. Controller Design for the PMSM

Let us now use the IMC method to design a suitable current controller for the PMSM. Since $G(s)$ has no right-half-plane zeros and behaves as a first-order system for high frequencies, we can let

$$C(s) = G^{-1}(s)L(s) \tag{15}$$

where all diagonal elements of $L(s)$ may be selected equal,

$$L(s) = \frac{\alpha}{s+\alpha}I. \tag{16}$$

Herein lies the main benefit of using IMC.

The tuning problem, which for a PI controller involves adjustment of two parameters, is reduced to the selection of one parameter only, the desired closed-loop bandwidth α.

Since, for a first-order system, the 10%–90% rise time t_r is related to α as $t_r = \ln 9/\alpha$, a specification of the rise time immediately yields the desired bandwidth and, in turn, a suitable controller. Using (9), with $\hat{G} = G$, we find that the equivalent classic controller becomes

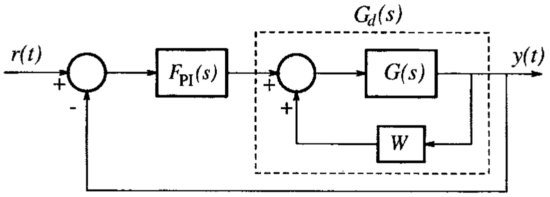


Fig. 3. Decoupling and diagonal IMC (DIMC).

$$\begin{aligned} F(s) &= \left[I - \frac{\alpha}{s+\alpha}I\right]^{-1} G^{-1}(s)\frac{\alpha}{s+\alpha} = \frac{\alpha}{s}G^{-1}(s) \\ &= \frac{\alpha}{s}\begin{bmatrix} sL_d + R_s & -\omega_r L_q \\ \omega_r L_d & sL_q + R_s \end{bmatrix} \\ &= \alpha \begin{bmatrix} L_d\left(1 + \dfrac{R_s}{sL_d}\right) & -\dfrac{\omega_r L_q}{s} \\ \dfrac{\omega_r L_d}{s} & L_q\left(1 + \dfrac{R_s}{sL_q}\right) \end{bmatrix}. \end{aligned} \tag{17}$$

It is, thus, more straightforward to implement the classic structure (Fig. 2) than the IMC structure (Fig. 1). A comparison with two standard PI controllers (one each for the d and q loops)

$$F_{\mathrm{PI}}(s) = \begin{bmatrix} K_d\left(1 + \dfrac{1}{sT_{id}}\right) & 0 \\ 0 & K_q\left(1 + \dfrac{1}{sT_{iq}}\right) \end{bmatrix} \tag{18}$$

shows that (17) is an extension of PI control with integrators added in the antidiagonal elements of $F(s)$ in order to remove the cross coupling, with

$$K_d = \alpha L_d, \quad K_q = \alpha L_q, \quad T_{id} = \frac{L_d}{R_s}$$

and

$$T_{iq} = \frac{L_q}{R_s}. \tag{19}$$

(The controller (18) is, for obvious reasons, also called a *diagonal* controller, see below.)

In [9], the IMC structure was implemented rather than the classic structure. Hence, the connection between IMC and PI control was lost.

B. Decoupling and Diagonal IMC (DIMC)

An alternative to direct implementation of IMC is to first decouple the machine dynamics by letting

$$\begin{bmatrix} v_d(t) \\ v_q(t) \end{bmatrix} = \begin{bmatrix} u_d(t) \\ u_q(t) \end{bmatrix} + \begin{bmatrix} -\omega_r L_q i_q(t) \\ \omega_r L_d i_d(t) \end{bmatrix} \tag{20}$$

where $u_d(t)$ and $u_q(t)$ are the outputs of the d and q direction PI controllers. This decoupling can be regarded as an *inner feedback loop,* as shown in Fig. 3. The decoupling matrix W is defined as in (8). This yields

$$G_d(s) = \begin{bmatrix} \dfrac{1}{sL_d + R_s} & 0 \\ 0 & \dfrac{1}{sL_q + R_s} \end{bmatrix} = D^{-1}(s). \tag{21}$$

Now, the standard PI controller (18) can be used in the outer feedback loop with the same parameter selection as for IMC, i.e., (19). If decoupling is not necessary, the inner loop can be left out and standard synchronous-frame PI control is obtained. Even in this case, (19) can be used for parameter selection.

IMC and DIMC are compared in Section III-D.

C. Controller Design for the Rotor-Flux-Oriented IM

Consider the complex space vector equations for the IM in synchronous coordinates [12]:

$$\begin{aligned}\frac{d}{dt}\overline{\psi}_s(t) &= -R_s\overline{i}_s(t) - j\omega_1\overline{\psi}_s(t) + \overline{v}(t),\\ \overline{\psi}_s(t) &= L_s\overline{i}_s(t) + L_m\overline{i}_r(t)\end{aligned} \quad (22)$$

$$\begin{aligned}\frac{d}{dt}\overline{\psi}_r(t) &= -R_r\overline{i}_r(t) - j\omega_2\overline{\psi}_r(t),\\ \overline{\psi}_r(t) &= L_m\overline{i}_s(t) + L_r\overline{i}_r(t).\end{aligned} \quad (23)$$

Here, $\overline{i}_s$, $\overline{\psi}_s$ and $\overline{i}_r$, $\overline{\psi}_r$ are the stator and rotor current and flux space vectors, respectively, while $\overline{v}$ is the impressed stator voltage space vector. R_s, R_r and L_s, L_r are the stator and rotor resistances and self-inductances, respectively. L_m is the magnetizing inductance, while ω_1 and $\omega_2 = \omega_1 - \omega_r$ are the stator and slip frequencies, respectively. Eliminating $\overline{i}_r$ and $\overline{\psi}_s$ among the above equations, we obtain

$$\begin{aligned}L_\sigma\frac{d\overline{i}_s(t)}{dt} + \underbrace{\left[R_s + \left(\frac{L_m}{L_r}\right)^2 R_r\right]}_{R_{\mathrm{IM}}}\overline{i}_s(t) + j\omega_1 L_\sigma\overline{i}_s(t)\\ = \overline{v}(t) + \frac{L_m}{L_r}\left(\frac{R_r}{L_r} - j\omega_r\right)\overline{\psi}_r(t)\end{aligned} \quad (24)$$

where $L_\sigma = L_s - L_m^2/L_r$. But rotor flux orientation implies that $\overline{\psi}_r$ is real and constant. Even though perfect field orientation never can be accomplished in practice, $\overline{\psi}_r$ will at least vary slowly and can be considered as a quasi-constant "disturbance" which is compensated for by integral control action. Hence, including the flux term in a modified stator voltage vector $\overline{v}'$, we can write

$$\overline{I}_s(s) = \underbrace{\frac{1}{(s + j\omega_1)L_\sigma + R_{\mathrm{IM}}}}_{\overline{G}(s)}\overline{V}'(s). \quad (25)$$

The complex-valued transfer function $\overline{G}$ corresponds to the inverse transfer function matrix

$$G^{-1}(s) = \begin{bmatrix} sL_\sigma + R_{\mathrm{IM}} & -\omega_1 L_\sigma \\ \omega_1 L_\sigma & sL_\sigma + R_{\mathrm{IM}}\end{bmatrix}. \quad (26)$$

Thus, the current controller for the rotor-flux-oriented IM is given by

$$F(s) = \frac{\alpha}{s}\begin{bmatrix} sL_\sigma + R_{\mathrm{IM}} & -\omega_1 L_\sigma \\ \omega_1 L_\sigma & sL_\sigma + R_{\mathrm{IM}}\end{bmatrix} \quad (27)$$

from which $F_{\mathrm{PI}}(s)$ for DIMC or standard PI control is obtained by putting $\omega_1 = 0$. The controller for the IM is, thus, identical to that for the PMSM with the following parameter substitutions:

$$\{L_d, L_q\} \to L_\sigma, \quad R_s \to R_{\mathrm{IM}}, \quad \omega_r \to \omega_1. \quad (28)$$

Due to this similarity, the notation for the PMSM is used exclusively in the following. All results derived hold for the IM as well, with the above substitutions.

D. Sensitivity Analysis

We analyze, for IMC and DIMC, the relative sensitivity of the loop transfer function matrix $G_k(s)$ (which is the transfer function of the controller–plant cascade) with respect to a small deviation Δa_i of the ith model parameter, a_i, from its true value, i.e.,

$$\frac{G_k^{-1}(s)\Delta G_k(s)}{a_i^{-1}\Delta a_i} \to a_i G_k^{-1}(s)\frac{\partial G_k(s)}{\partial a_i} \text{ as } \Delta a_i \to 0. \quad (29)$$

We can express this as

$$S_i(s) = a_i G_k^{-1}(s)\frac{\partial G_k(s)}{\partial a_i}\bigg|_{\hat{G}=G}. \quad (30)$$

That is, after the partial derivative is taken with respect to the model parameter a_i, the parameters of the model are made equal to those of the machine.

For *IMC*, the loop transfer function is $G_k^{\mathrm{IMC}}(s) = G(s)F(s) = (\alpha/s)G(s)\hat{G}^{-1}(s)$, so we immediately obtain

$$\begin{aligned}S_i^{\mathrm{IMC}}(s) &= a_i\left[\frac{\alpha}{s}G(s)\hat{G}^{-1}(s)\right]^{-1}\frac{\alpha}{s}G(s)\times\frac{\partial\hat{G}^{-1}(s)}{\partial a_i}\bigg|_{\hat{G}=G}\\ &= a_i G(s)\frac{\partial G^{-1}(s)}{\partial a_i}.\end{aligned} \quad (31)$$

For *DIMC*, we have $G_k(s) = G_d(s)F(s) = (\alpha/s)G_d(s)\hat{D}(s)$ where $G_d(s) = [I - G(s)\hat{W}]^{-1}G(s)$, giving

$$S_i^{\mathrm{DIMC}}(s) = a_i\left[\frac{\partial G_d(s)}{\partial a_i}\hat{D}(s) + G_d(s)\frac{\partial\hat{D}(s)}{\partial a_i}\right]_{\hat{G}=G}. \quad (32)$$

Differentiating the relation $[I - G(s)\hat{W}]G_d(s) = G(s)$ implicitly yields

$$\begin{aligned}\frac{\partial G_d(s)}{\partial a_i}\bigg|_{\hat{G}=G} &= G_d(s)\frac{\partial\hat{W}}{\partial a_i}G_d(s)\bigg|_{\hat{G}=G}\\ &= D^{-1}(s)\frac{\partial W}{\partial a_i}D^{-1}(s)\end{aligned} \quad (33)$$

and

$$\begin{aligned}S_i^{\mathrm{DIMC}}(s) &= a_i D^{-1}(s)\left[\frac{\partial W}{\partial a_i}D^{-1}(s)D(s) + \frac{\partial D(s)}{\partial a_i}\right]\\ &= a_i D^{-1}(s)\frac{\partial G^{-1}(s)}{\partial a_i}.\end{aligned} \quad (34)$$

From (8), it follows that the two functions can be written as

$$S_i^{\mathrm{IMC}}(s) = a_i G(s)\frac{\partial G^{-1}(s)}{\partial a_i} \quad (35)$$

$$S_i^{\mathrm{DIMC}}(s) = a_i G(s)|_{\omega_r=0}\frac{\partial G^{-1}(s)}{\partial a_i}. \quad (36)$$

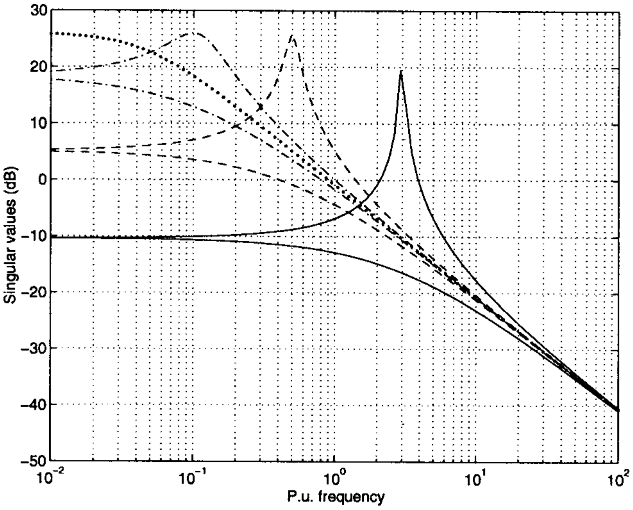


Fig. 4. Singular values of the PMSM model for $\omega_r = \{3, 0.5, 0.1, 0\}$ p.u. (solid, dashed, dash–dotted, and dotted, respectively).

The sensitivity functions of the two cases are, hence, scaled by the machine transfer function matrix at the operating speed and zero speed, respectively.

Let us make an evaluation in the frequency domain. As G is a transfer function matrix, the gain $|G(j\omega)|$ is undefined. Instead, *singular values* have to be utilized; see [11], [13] and Section A of the Appendix. The singular values of $G(j\omega)$ for different ω_r are depicted in Fig. 4 for a PMSM having the parameter values $L_d = L_q = 1$ and $R_s = 0.05$ p.u. We can note the following characteristics.

- For low frequencies ($\omega < 0.1$) and high speeds, IMC has *significantly lower sensitivity* than DIMC. Furthermore, the low-frequency sensitivity of IMC *decreases with increasing speed*. This is good, since the cross coupling increases with ω_r. IMC can, therefore, be expected to remove cross coupling better than DIMC. On the other hand, integral control action takes care of low-frequency cross coupling, so the difference between IMC and DIMC may not be overwhelming.
- The drawback of IMC is the large sensitivity peak at $\omega \approx \omega_r$, which is equal in magnitude to the sensitivity of DIMC at $\omega = 0$. This large peak is due to the attempt to cancel the machine dynamics (which are characterized by a complex pole pair, the imaginary parts of which follow the rotor speed [14]), and shows that oscillatory behavior can be expected at high speeds.

The sensitivity analysis can be concluded by the following recommendation.

> *DIMC is more desirable than IMC, as the sensitivity peak at $\omega = \omega_r$ is avoided. IMC has lower low-frequency sensitivity than DIMC, but the performance improvement is not expected to be overwhelming.*

Hence, IMC is of interest mainly not for direct implementation, but as a design method for standard PI controllers, with or without an inner cross-coupling removal loop added (DIMC).

IV. Discrete-Time Implementation

We consider DIMC only. A sample software implementation of the algorithm derived here—suitable for a digital signal processor (DSP)—is presented in Section C of the Appendix.

The controller is given by

$$F_{\mathrm{PI}}(s) = \frac{\alpha}{s}\begin{bmatrix} sL_d + R_s & 0 \\ 0 & sL_q + R_s \end{bmatrix}. \tag{37}$$

The controller can be implemented on state space. A realization is straightforwardly obtained as

$$\begin{aligned} \dot{x}_c(t) &= \underbrace{\alpha\begin{bmatrix} R_s & 0 \\ 0 & R_s \end{bmatrix}}_{B_c} e(t) \\ u(t) &= x_c(t) + \underbrace{\alpha\begin{bmatrix} L_d & 0 \\ 0 & L_q \end{bmatrix}}_{D_c} e(t) + Wy(t). \end{aligned} \tag{38}$$

where $x_c(t)$ is the controller state vector, $e(t) = r(t) - y(t)$, and $Wy(t)$ is the inner feedback for cross-coupling removal.

A. Discretization

Backward difference discretization [15] is suggested, which yields a simple algorithm:

$$x_{c,\,k} = x_{c,\,k-1} + TB_c e_k \tag{39}$$

$$u_k = x_{c,\,k} + D_c e_k + Wy_k \tag{40}$$

where T is the sampling period and k the sampling instant (at time $t = kT$).

B. Inverter Saturation and Antiwindup

A problem which has to be dealt with is that the output signal u in practice must be limited, as the inverter will saturate whenever $|u|$ reaches the maximum available voltage. Let $\overline{u}$ be the limited output:

$$\overline{u} = f(u) = \begin{cases} u, & |u| \leq |u|_{\max} \\ u \cdot \dfrac{|u|_{\max}}{|u|}, & |u| > |u|_{\max}. \end{cases} \tag{41}$$

However, only limiting the controller output signal will inevitably lead to poor performance.

- *Integrator windup* results if the integration of the control error e is not stopped when u saturates. This is known to give large overshoots [16].
- The controller output limitation can be considered as a disturbance $\tilde{u} = \overline{u} - u$ entering between the controller and the plant. IMC is sensitive to such disturbances, since the canceled plant dynamics are “activated” [7], [16]. This is undesirable also for DIMC, as the poles of the decoupled system are located close to the origin and will give rise to slow transients.

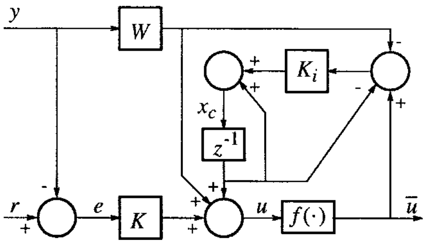


Fig. 5. Vector block diagram of the DIMC current controller with anti-windup. ($W = 0$ for PI control.)

The above problems can be dealt with in the following way, which is known as *back-calculation* [16].

Step 1: At the sampling instant k, compute the ideal controller output,

$$\begin{aligned} u_k &= x_{c,k} + D_c e_k + W y_k \\ &= x_{c,k-1} + \underbrace{(D_c + TB_c)}_{K} e_k + W y_k \end{aligned} \tag{42}$$

and limit the signal, $\overline{u}_k = f(u_k)$. This is the command vector to the pulsewidth modulator (PWM) [1], [17].

Step 2: In order to avoid the problems discussed above, the saturation disturbance has to be "moved" from the controller output to the controller input. Hence, by inverting (42), compute the error signal $\overline{e}_k$ which would yield $\overline{u}_k$:

$$\overline{e}_k = (D_c + TB_c)^{-1}[\overline{u}_k - x_{c,k-1} - W y_k]. \tag{43}$$

Then use $\overline{e}_k$ to update the controller state vector (i.e., the integrators):

$$x_{c,k} = x_{c,k-1} + TB_c \overline{e}_k. \tag{44}$$

If u does not saturate, $\overline{e}_k = e_k$. The latter two equations can be slightly simplified by combining them as follows:

$$\begin{aligned} x_{c,k} &= x_{c,k-1} + \underbrace{TB_c(D_c + TB_c)^{-1}}_{K_i}[\overline{u}_k \\ &\quad - x_{c,k-1} - W y_k]. \end{aligned} \tag{45}$$

The controller matrices K and K_i can be simplified by making suitable approximations. We have

$$K = D_c + TB_c \approx D_c = \begin{bmatrix} \alpha L_d & 0 \\ 0 & \alpha L_q \end{bmatrix} \tag{46}$$

and

$$\begin{aligned} K_i &= TB_c(D_c + TB_c)^{-1} \approx TB_c D_c^{-1} \\ &= \begin{bmatrix} \dfrac{TR_s}{L_d} & 0 \\ 0 & \dfrac{TR_s}{L_q} \end{bmatrix}. \end{aligned} \tag{47}$$

This results in the block diagram depicted in Fig. 5.

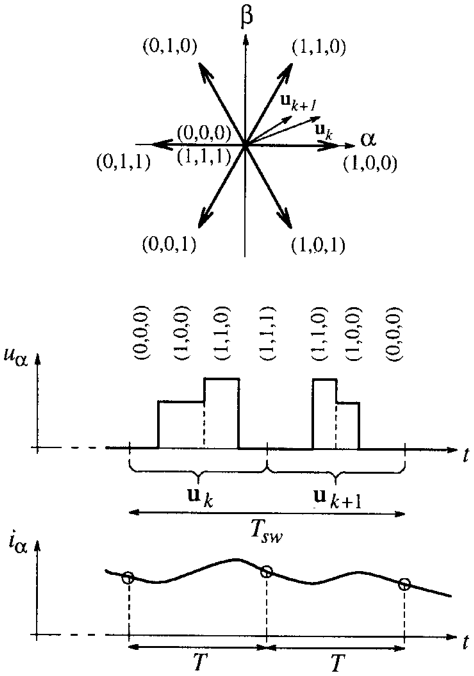


Fig. 6. Space vector modulation. T and T_{sw} are the sampling and switching periods, respectively.

C. Selection of the Sampling and Switching Frequencies

It is important to select the sampling and switching frequencies high enough, so that the system performance does not degrade. Selecting the angular sampling frequency ω_s as at least 10 times the closed-loop bandwidth is a good recommendation [15]:

$$\omega_s \geq 10\alpha. \tag{48}$$

As for the switching frequency, two independent voltage space vectors can be generated per switching period using space vector modulation of a voltage-source inverter [1, ch. 4], as illustrated in Fig. 6. (This holds for the suboscillation PWM method as well.) The angular switching frequency ω_{sw} should, hence, not be lower than half the sampling frequency. This yields

$$\omega_{sw} \geq 5\alpha. \tag{49}$$

Using this formula and the relation $t_r = \ln 9/\alpha$, we can relate the switching frequency (in hertz) to the rise time:

$$f_{sw} \geq \frac{5 \ln 9}{2\pi t_r} \approx \frac{1.75}{t_r}. \tag{50}$$

A 1-ms rise time thus requires a switching frequency of at least 1.75 kHz. This is a low figure for insulated gate bipolar transistor IGBT and MOSFET inverters [17]; it is often desirable to switch faster for lower harmonic content and reduced acoustical noise. The recommendation $\omega_{sw} \geq 5\alpha$ obviously does not put excessive demands on the switching frequency. Neither is the recommended sampling frequency of 3.5 kHz too demanding.

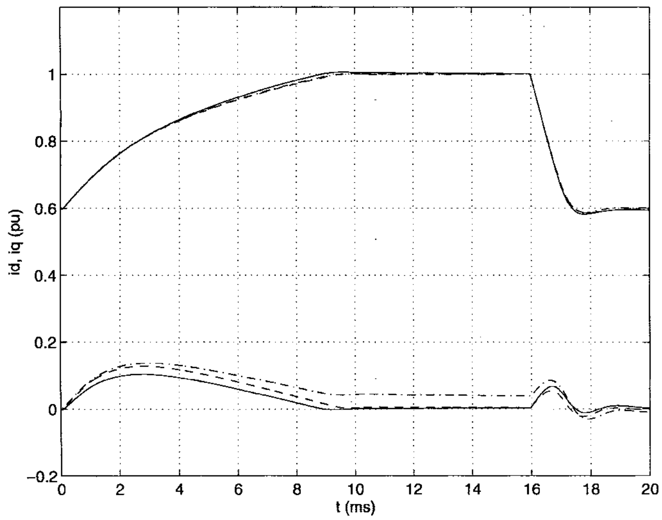


Fig. 7. PMSM simulation. Step responses for IMC (solid), DIMC (dashed), and PI (dash–dotted).

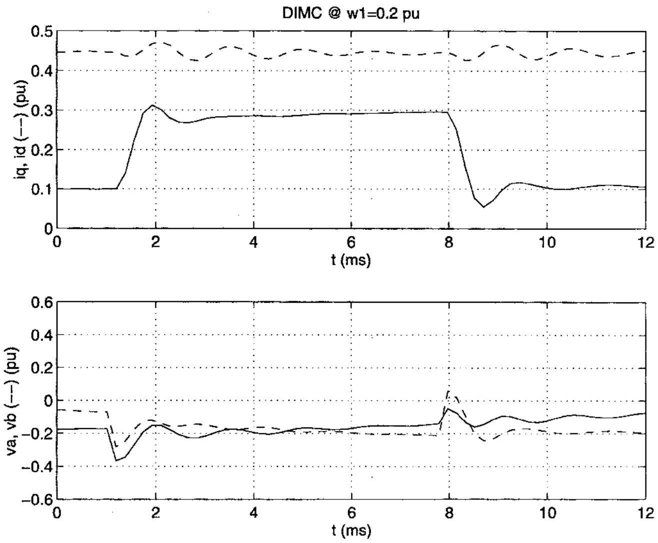


Fig. 8. IM experiment. Step responses for DIMC.

V. Simulation

Consider a PMSM with the data $L_d = 1.0$, $L_q = 1.4$, $R_s = 0.05$ in per-unit values and with a base frequency of 50 Hz, i.e., 314 rad/s. The permanent magnet flux $\psi_m = 1$ p.u. and the maximum stator voltage $|u|_{\max} = 1$ p.u. The machine model is slightly incorrect: $L_d = 1.2$, $L_q = 1.2$, $R_s = 0.08$. From the machine model data we design a current controller for a closed-loop system rise time of 1 ms.

Bandwidth Selection: The desired rise time corresponds to $10^{-3} \cdot 314 = 0.314$ p.u., so the bandwidth α should be selected as $\ln 9/0.314 = 7$ p.u.

Sampling Rate Selection: The "10 times bandwidth" rule gives a sampling frequency of 70 p.u., i.e., 3.5 kHz. A one-sample delay is simulated to account for inverter and computational dead times.

Evaluation: At $\omega_r = 0.5$ p.u. and $t = 0$, we simulate the response to a setpoint step change; i_q^{ref} goes from 0.6 to 1.0 p.u. and back again at $t = 16$ ms, while $i_d^{\text{ref}} = 0$ (see Fig. 7). At the first step, the inverter saturates during the transient, hence, the rise time is much longer than the desired. IMC is, as seen, slightly better than DIMC at removing the decoupling, which was predicted by theory. The difference is, however, not significant. Standard PI control gives a control error in i_d which very slowly decays to zero.

VI. Experimental Results

The DIMC algorithm is evaluated experimentally on an induction motor drive, the data of which are given in Section B of the Appendix. The IM is controlled using indirect rotor flux orientation [1, ch. 5].

It was found that $\alpha = 8$ p.u. is a good tradeoff between speed and robustness. A further increase was found to give too oscillatory a response. The nominal rise time is then $\ln 9/(8 \cdot 2\pi \cdot 50) = 0.88$ ms. The control algorithm has a delay of one sampling period. Suboscillation method PWM generation is used, so the inverter is not utilized to its full capacity. Note in the following that the vectors are scaled for rms value correspondence; $|i_s| = 1$ p.u. corresponds to rated current and similarly for the stator voltage vector.

A. Experiment 1

The motor is operated at no load at approximately 80% of rated flux and $\omega_r = 0.2$ p.u. At $t \approx 1$ ms, the setpoint for i_q is stepped up 0.2 p.u. and stepped down again at $t \approx 7.8$ ms. The setpoint for i_d is held constant at 0.44 p.u. The step responses for DIMC are shown in Fig. 8. (It should be noted that i_q has been adjusted vertically slightly, so that the step starts at 0.1 p.u. and the voltages v_α and v_β in stator coordinates are denoted as va and vb, respectively.) The step responses for PI are virtually identical and are not shown. This was to be expected due to the low rotor speed. The rise time is slightly shorter than the theoretical 0.88 ms. The settling time on the other hand is fairly long, which is an indication of slightly too low integral action.

B. Experiment 2

The field is weakened ($i_d^{\text{ref}} = 0.11$ p.u.) and the speed increased to 1.6 p.u. The same experiment is then repeated (see Figs. 9 and 10). This time, DIMC performs slightly better than PI. The slow decay of the control error in i_d which was acknowledged in the simulation is also present here (although the error is not particularly large).

VII. Conclusion

In this paper, the concept of internal model control was applied to synchronous-frame current control for permanent magnet synchronous machines and induction machines. It was shown that the controller resulting from the IMC design method corresponds to two PI controllers with cross-coupling-removing integrators added. The additional cost of implementing IMC compared to PI control is negligible. An alternative structure, DIMC, using an inner decoupling loop and two outer standard PI loops was introduced as an alternative. DIMC is the structure preferable to use. Although IMC has

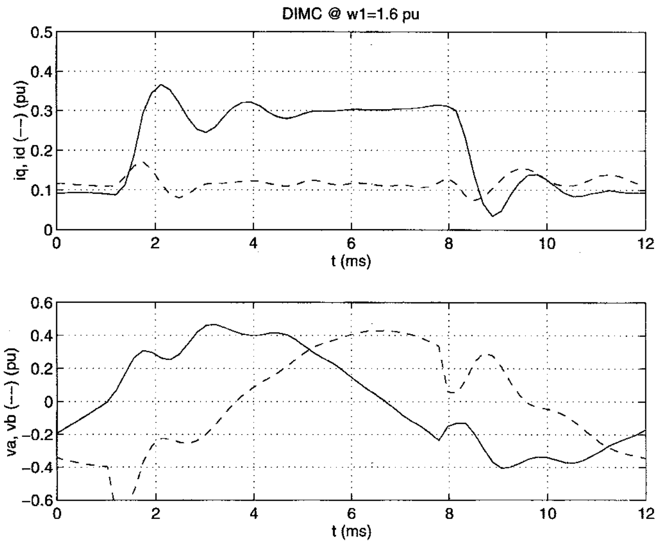


Fig. 9. IM experiment. Step responses for DIMC.

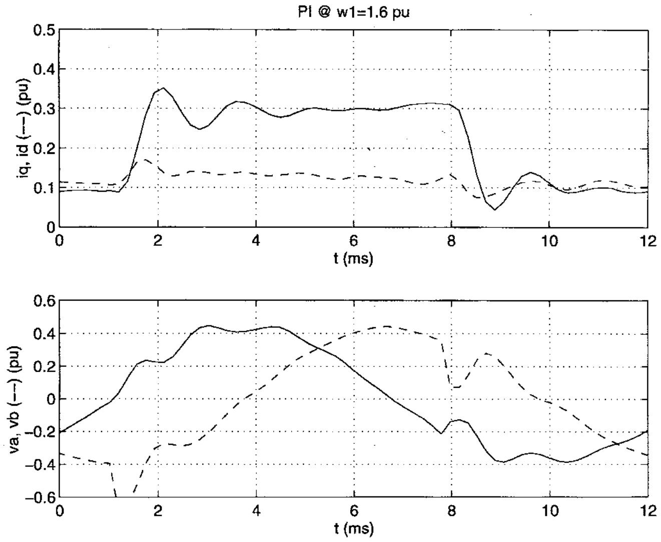


Fig. 10. IM experiment. Step responses for PI.

TABLE I
MACHINE DATA

Name-plate:	
Type	ELIN LKM-309, L04 F3B
Number of poles	4
Power	1.5 kW
Phase-to-phase voltage	220/380 V
Phase current	6.6/3.8 A
Speed	1415 rpm
Base values:	
Frequency	50 Hz
Current	3.8 A
Flux	0.62 Wb
Voltage	195 V
Impedance	52 Ω
Inductance	0.16 H
Machine parameters:	
L_m	264 mH ∼ 1.65 p.u.
$L_s = L_r$	279 mH ∼ 1.74 p.u.
R_s	5.5 Ω ∼ 0.11 p.u.
R_r	4.0 Ω ∼ 0.077 p.u.

lower parameter sensitivity, there is also a risk for oscillatory behavior which is avoided with DIMC.

The benefit of the IMC design method is that the controller parameters are expressed directly in the machine parameters and the desired closed-loop bandwidth. Hence, the design procedure is simple, and trial and error can largely be avoided. It should be stressed that, even if removal of cross coupling is not an important objective (see also [6]), IMC is still very useful for standard PI controller design.

Rules for sampling and switching frequency selection were given and discrete-time implementation issues were considered. A backcalculation algorithm was suggested in order to avoid integrator windup (and, thus, degraded performance) when the inverter saturates at transients.

We finally emphasize that the controllers are implemented in synchronous coordinates, which has been shown to yield the best performance among low-complexity current controllers [1, chs. 4, 5]. Therefore, coordinate transformations are needed in the implementation. With a state-of-the-art DSP, this is usually no obstacle, however, since trigonometric operations can be implemented using a lookup table or the CORDIC algorithm [18]. If coordinate transformations are still not desirable, the controller can be transformed to and implemented in stator coordinates, as shown in [19].

For further information, see [14].

APPENDIX

A. Singular Values

For a multivariable dynamic system, one can obtain a frequency function by substituting $s \rightarrow j\omega$ in any transfer function matrix, as for any single variable (scalar) transfer function. However, the gain $|G(j\omega)|$ is not as straightforward to express. It can be shown [11] that

$$\underline{\sigma}[G(j\omega)] \leq \frac{|Y(\omega)|}{|U(\omega)|} \leq \overline{\sigma}[G(j\omega)] \qquad (51)$$

where $\underline{\sigma}$ and $\overline{\sigma}$ are the *minimum and maximum singular values*. The singular values for a frequency function matrix are defined as

$$\sigma_i[G(j\omega)] = \sqrt{\lambda_i[G^T(-j\omega)G(j\omega)]} \qquad (52)$$

where λ_i is the ith eigenvalue. There are as many singular values as there are inputs/outputs. (We consider square G only.) It is, hence, not possible to talk about a *fixed* gain but a gain *spread*, bounded by $\underline{\sigma}$ and $\overline{\sigma}$. The actual gain depends on the direction of the input vector U. See also [13].

B. Data for Laboratory Induction Motor Drive

1) Machine: The data are given in Table I.

2) Inverter: IGBT voltage-source inverter, 540-V dc link, 20-kW rated power, three-phase diode rectifier on mains side, analog suboscillation method PWM generation with 5.3-kHz switching frequency.

3) Digital Signal Processor: Texas Instruments TMS320C40 floating-point DSP, sampling frequency 5.3 kHz, DSpace development system on Pentium PC host. All algorithms are implemented in C code. Stator current and resolver signal sampling is made synchronously with inverter switchings, at the peaks of the PWM suboscillation triangular wave. This gives significantly reduced noise levels in the samples.

C. A Sample Software Implementation of DIMC

A C program implementing the DIMC control algorithm follows below. It should be self explanatory to anyone with some knowledge of programming. Constants are written in capital letters. Declarations of variables, constants, and functions are not shown.

```
// Acquire signals & dq transform
read_ad(&ia, &ib, &wr, &theta);
co = cos(theta);
si = sin(theta);
id = co*ia + si*ib;
iq = -si*ia + co*ib;

// Compute ideal voltages:
ed = idref - id;
eq = iqref - iq;
vd = ALPHA*LD*ed - wr*LQ*iq + xd;
vq = ALPHA*LQ*eq + wr*LD*id + xq;

// Limit voltages:
uabs = sqrt(vd*vd + vq*vq);
if(uabs > UMAX) {
        vd = vd*UMAX/uabs;
        vq = vq*UMAX/uabs;
}

// Back-calculate:
xd += T*RS/LD*(vd - xd + wr*LQ*iq);
xq += T*RS/LQ*(vq - xq - wr*LD*id);

//ab transform and call PWM:
va = co*id - si*iq;
vb = si*id + co*iq;
PWM(va, vb);
```

REFERENCES

[1] B. K. Bose, Ed., *Power Electronics and Variable Frequency Drives.* Piscataway, NJ: IEEE Press, 1996.

[2] J. Holtz and S. Stadtfeld, "A predictive current controller for the stator current vector of ac machines fed from a switched voltage source," in *Proc. IPEC,* Tokyo, Japan, 1983, pp. 1665–1675.

[3] L. Zhang, R. Norman, and W. Shepherd, "Long-range predictive control of current regulated PWM for induction motor drives using the synchronous reference frame," *IEEE Trans. Contr. Syst. Technol.,* vol. 5, pp. 119–126, Jan. 1997.

[4] R. D. Lorenz and D. B. Lawson, "Performance of feedforward current regulators for field-oriented induction machine controllers," *IEEE Trans. Ind. Applicat.,* vol. 23, pp. 597–602, July/Aug. 1987.

[5] D.-C. Lee, S.-K. Sul, and M.-H. Park, "High performance current regulator for a field-oriented controlled induction motor drive," *IEEE Trans. Ind. Applicat.,* vol. 23, pp. 1247–1257, Sept./Oct. 1994.

[6] J.-W. Choi and S.-K. Sul, "New control concept-minimum time current control in the three-phase PWM converter," *IEEE Trans. Power Electron.,* vol 21, pp. 124–131, Jan. 1997.

[7] M. Morari and E. Zafiriou, *Robust Process Control.* Englewood Cliffs, NJ: Prentice-Hall, 1989.

[8] I. Boldea and S. A. Nasar, *Vector Control of AC Drives.* Boca Raton, FL: CRC Press, 1992.

[9] J. L. Thomas and M. Boidin, "An internal model control structure in field oriented controlled v.s.i. induction motors," in *Proc. EPE,* Florence, Italy, 1991, vol. 2, pp. 202–207.

[10] P. Vas, *Electrical Machines and Drives: A Space-Vector Theory Approach.* Oxford, U.K.: Clarendon, 1992.

[11] J. M. Maciejowski, *Multivariable Feedback Design.* Reading, MA: Addison-Wesley, 1989.

[12] W. Leonhard, *Control of Electrical Drives.* Berlin: Springer-Verlag, 1985.

[13] L. Harnefors and H.-P. Nee, "On the dynamics of ac machines and sampling rate selection for discrete-time vector control," in *Proc. ICEM,* Vigo, Spain, Sept. 1996, vol. 2, pp. 251–256.

[14] L. Harnefors, "On analysis, control and estimation of variable-speed drives," Ph.D. dissertation, Div. Electr. Machines and Drives, Dep. Elect. Power Eng., Royal Inst. Technol., Stockholm, Sweden, 1997.

[15] R. H. Middleton and G. C. Goodwin, *Digital Control and Estimation: A Unified Approach.* Englewood Cliffs, NJ: Prentice-Hall, 1990.

[16] K. J. Åström and T. Hägglund, *PID Controllers: Theory, Design and Tuning,* 2nd ed. Research Triangle Park, NC: Instrument Society of America, 1995.

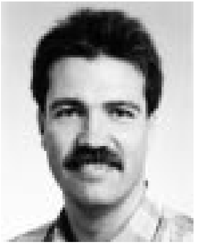
[17] J. G. Kassakian, M. F. Schlecht, and G. C. Verghese, *Principles of Power Electronics.* Reading, MA: Addison-Wesley, 1991.

[18] S. Y. Kung, *VLSI Array Processors.* Englewood Cliffs, NJ: Prentice-Hall, 1988.

[19] T. M. Rowan and R. J. Kerkman, "A new synchronous current regulator and an analysis of current-regulated PWM inverters," *IEEE Trans. Ind. Applicat.,* vol. 22, pp. 678–690, July/Aug. 1986.

Lennart Harnefors (S'94–M'98) received the M.S., Licentiate, and Ph.D. degrees in electrical engineering from the Royal Institute of Technology, Stockholm, Sweden, in 1993, 1995, and 1997, respectively.

Since 1994, he has been with the Department of Electrical Engineering, Mälardalen University, Västerås, Sweden, where he is a Senior Lecturer. He is also an Affiliate Senior Lecturer with the Division of Electrical Machines and Drives, Department of Electric Power Engineering, Royal Institute of Technology, Stockholm, Sweden. His research interests include circuits, systems, and control, particularly as applied to variable-speed drives.



Hans-Peter Nee (S'91–M'96) received the M.S., Licentiate, and Ph.D. degrees in electrical engineering from the Royal Institute of Technology, Stockholm, Sweden, in 1987, 1992, and 1996, respectively.

Since 1988, he has been with the Division of Electrical Machines and Drives, Department of Electric Power Engineering, Royal Institute of Technology, Stockholm, Sweden, where he is a Research Associate and Director of the Permanent Magnet Drives Research Program. His research interests are permanent magnet and induction motor drives.

Dr. Nee is the recipient of several awards, including a Best Paper Award at the ICEM'94 Conference.