An Online Rotor Time Constant Estimator for the Induction Machine

Kaiyu Wang, *Member, IEEE*, John Chiasson, *Senior Member, IEEE*, Marc Bodson, *Fellow, IEEE*, and Leon M. Tolbert, *Senior Member, IEEE*

***Abstract*—Indirect field-oriented control of an induction machine requires knowledge of the rotor time constant to estimate the rotor flux linkages. Here, an online method is presented for estimating the rotor time constant and the stator resistance, both of which vary during operation of the machine due to ohmic heating. The method uses measurements of the stator voltages, stator currents, and their derivatives (first derivative of the voltages and both the first and second derivatives of the currents). The problem is formulated as finding those parameter values that best fit (in a least-squares sense) the model of the induction motor to the measured output data of the motor. This method guarantees that the parameter values are found in a finite number of steps. Experimental results of an online implementation are presented.**

***Index Terms*—Induction motor, parameter identification, rotor time constant.**

I. Introduction

THE field-oriented control method provides a means to obtain high-performance control of an induction machine for use in applications such as traction drives. This field-oriented control methodology requires knowledge of the rotor fluxes which are not usually measured [1], [2]. To circumvent this problem, the rotor fluxes are estimated using an observer, and this observer requires the value of the rotor time constant. The rotor time constant varies due to ohmic heating, and the work presented here is a method which allows the value of the rotor time constant to be updated during normal operation of the machine.

The induction motor parameters consist of M (the mutual inductance), L_S, L_R (the stator and rotor inductances), R_S, R_R (the stator and rotor resistances), and J (the inertia of the rotor). Standard methods for the estimation of induction motor parameters include the blocked rotor test, the no-load test, and the standstill frequency response test. However, these approaches cannot be used online, that is, during normal operation of the machine.

The interest here is in tracking the value of T_R as it changes. A model-based approach is considered here, which uses measurements of the stator currents, stator voltages, and rotor position to find the parameter values that best fit this data set to the model in a least-squares sense. Using these measurements, the first derivative of the stator voltages, both the first and second derivatives of the stator currents, and the derivative of the rotor angle are all computed (reconstructed) for use in the estimation algorithm. The method is implemented online, and experimental results of the tracking of T_R are presented.

Because the rotor state variables are not available measurements, the system identification model cannot be made linear in the parameters without overparameterizing the model (see [3] for a discussion of the problems encountered due to noise when trying to identify the induction motor parameters using an overparameterized model). In the work here, the model is reformulated so that it is a *nonlinear* system identification problem that is not overparameterized. It is shown that the parameter vector that minimizes the least-squares (residual) error can be found in a finite number of steps (this is in contrast to [3], where the induction motor parameters are found solving a constrained optimization problem via an iterative numerical algorithm).

In [4], the authors formulated and solved a nonlinear least-squares problem to estimate all of the identifiable parameters (σ, T_R, R_S, L_S) of the motor based on input/output data. In this brief, only the two parameters that change due to ohmic heating, namely T_R and R_S, are estimated and implemented online. This is an important special case because, as explained before, the rotor time constant is required in the standard flux estimation scheme used in field-oriented control. (The value of R_S is also estimated so that its effect on the estimation of T_R is accounted for properly.) Further, by only estimating these two parameters, the numerical conditioning of the problem is improved and the (sufficient) excitation requirements of the system are reduced. More specifically, it is shown next that constant speed measurements (with the motor under load) are sufficient to determine T_R and R_S, which is not true if all the parameters are estimated as in [4]. Thus, the ability of the algorithm to track variations in T_R is improved. Further, in contrast to [4], the algorithm is implemented *online* and the experiments are carried out using a pulse width modulation (PWM) inverter rather than the three phase voltages from a utility outlet. Preliminary versions of the present work have appeared in [5] and [6], but again, in both of these works, the algorithm was not implemented online and the experiments were not done using a standard PWM inverter. Finally, an error index and a parametric error index are derived and presented as measures of performance of the estimation algorithm, which was not done in [4].

Manuscript received January 2, 2006; revised May 25, 2005. Manuscript received in final form August 30, 2006. Recommended by Associate Editor S. Peresada. This work was supported in part by Oak Ridge National Laboratory under the UT/Battelle Contract 4000007596 and by the National Science Foundation under Contract NSF ECS-0093884.

K. Wang is with the Electronic Control Systems Group, Osram Sylvania, Danvers, MA 01923 USA (e-mail: kaiyu.wang@sylvania.com).

J. Chiasson is with the Electronics and Computer Engineering Department, Boise State University, Boise, ID 83725-2100 USA (e-mail: johnchiasson@boisestate.edu).

M. Bodson is with the Electronics and Computer Engineering Department, University of Utah, Salt Lake City, UT 84112 USA (e-mail: bodson@ece.utah.edu).

L. M. Tolbert is with the Electronics and Computer Engineering Department, The University of Tennessee, Knoxville, TN 37996 USA and also with Oak Ridge National Laboratory, Oak Ridge, TN 37831 USA (e-mail: tolbert@utk.edu; tolbertlm@ornl.gov).

Color versions of Figs. 2–10 are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCST.2006.886445

1063-6536/$25.00 © 2007 IEEE

A combined parameter identification and velocity estimation problem is discussed in [7]–[9], where the speed is assumed to be slowly varying. In [10] and [11] a linear least-squares approach was used for parameter estimation and solved by assuming a slowly varying speed. In the paper by Marino *et al.* [12], the rotor time constant (rotor resistance) and the stator resistance were estimated using a dynamic estimation scheme that was shown to converge under some mild conditions (rotating field in the air gap). Castaldi *et al.* [13] also developed a dynamic estimation scheme based on adaptive control/observer techniques. The approach presented here is not dynamic, and thus, the issue of convergence does not arise. However, sufficiency of excitation is a concern, and it will be shown that the conditions to ensure this are satisfied in normal operation of the machine. For a summary of the various techniques for tracking the rotor time constant, the reader is referred to the recent survey [14], the recent paper [15], and to the book [16].

This brief is organized as follows. Section II introduces a standard induction motor model expressed in the rotor coordinates. Then, an overparameterized model which is linear in the unknown parameters is derived and discussed. Section IV presents the identification scheme for the rotor time constant by reducing the overparameterized linear model to a nonlinear model, which is not overparameterized. An approach to solve the resulting nonlinear (in the parameters) least-squares identification problem is presented and shown to guarantee that the minimum least-squares solution is found in a finite number of steps. Section VI presents the results of the identification algorithm using experimental data.

II. Induction Motor Model

Standard models of induction machines are available in the literature. Parasitic effects such as hysteresis, eddy currents, magnetic saturation, and others are generally neglected. Consider a state-space model of the system given by (cf. [2], [17], [18])

$$\begin{aligned}
\frac{di_{Sa}}{dt} &= \frac{\beta}{T_R}\psi_{Ra} + \beta n_p\omega\psi_{Rb} - \gamma i_{Sa} + \frac{1}{\sigma L_S}u_{Sa}\\
\frac{di_{Sb}}{dt} &= \frac{\beta}{T_R}\psi_{Rb} - \beta n_p\omega\psi_{Ra} - \gamma i_{Sb} + \frac{1}{\sigma L_S}u_{Sb}\\
\frac{d\psi_{Ra}}{dt} &= -\frac{1}{T_R}\psi_{Ra} - n_p\omega\psi_{Rb} + \frac{M}{T_R}i_{Sa}\\
\frac{d\psi_{Rb}}{dt} &= -\frac{1}{T_R}\psi_{Rb} + n_p\omega\psi_{Ra} + \frac{M}{T_R}i_{Sb}\\
\frac{d\omega}{dt} &= \frac{Mn_p}{JL_R}(i_{Sb}\psi_{Ra} - i_{Sa}\psi_{Rb}) - \frac{\tau_L}{J}
\end{aligned} \tag{1}$$

where $\omega = d\theta/dt$ with θ as the position of the rotor, n_p is the number of pole pairs, i_{Sa}, i_{Sb} are the (two-phase equivalent) stator currents, ψ_{Ra}, ψ_{Rb} are the (two-phase equivalent) rotor flux linkages, and u_{Sa}, u_{Sb} are the (two-phase equivalent) stator voltages.

The parameters of the model are the five electrical parameters, R_S and R_R (the stator and rotor resistances), M (the mutual inductance), L_S and L_R (the stator and rotor inductances), and the two mechanical parameters, J (the inertia of the rotor) and τ_L (the load torque). The symbols

$$\begin{aligned}
T_R &= L_R/R_R\\
\sigma &= 1 - M^2/(L_S L_R)\\
\beta &= M/(\sigma L_S L_R)\\
\gamma &= R_S/(\sigma L_S) + M^2 R_R/\left(\sigma L_S L_R^2\right)
\end{aligned}$$

have been used to simplify the expressions. T_R is referred to as the rotor time constant while σ is called the total leakage factor.

This model is transformed into a coordinate system attached to the rotor. For example, the current variables are transformed according to

$$\begin{bmatrix} i_{Sx} \\ i_{Sy} \end{bmatrix} = \begin{bmatrix} \cos(n_p\theta) & \sin(n_p\theta) \\ -\sin(n_p\theta) & \cos(n_p\theta) \end{bmatrix} \begin{bmatrix} i_{Sa} \\ i_{Sb} \end{bmatrix}. \tag{2}$$

The transformation simply projects the vectors in the (a, b) frame onto the axes of the moving coordinate frame. An advantage of this transformation is that the signals in the moving frame [i.e., the (x, y) frame] typically vary slower than those in the (a, b) frame (they vary at the slip frequency rather than at the stator frequency). At the same time, the transformation does not depend on any unknown parameter in contrast to the field-oriented (or dq) transformation. The stator voltages and the rotor fluxes are transformed in the same way as the currents resulting in the following model ([10], [11]):

$$\frac{di_{Sx}}{dt} = \frac{u_{Sx}}{\sigma L_S} - \gamma i_{Sx} + \frac{\beta}{T_R}\psi_{Rx} + n_p\beta\omega\psi_{Ry} + n_p\omega i_{Sy} \tag{3}$$

$$\frac{di_{Sy}}{dt} = \frac{u_{Sy}}{\sigma L_S} - \gamma i_{Sy} + \frac{\beta}{T_R}\psi_{Ry} - n_p\beta\omega\psi_{Rx} - n_p\omega i_{Sx} \tag{4}$$

$$\frac{d\psi_{Rx}}{dt} = \frac{M}{T_R}i_{Sx} - \frac{1}{T_R}\psi_{Rx} \tag{5}$$

$$\frac{d\psi_{Ry}}{dt} = \frac{M}{T_R}i_{Sy} - \frac{1}{T_R}\psi_{Ry} \tag{6}$$

$$\frac{d\omega}{dt} = \frac{Mn_p}{JL_R}(i_{Sy}\psi_{Rx} - i_{Sx}\psi_{Ry}) - \frac{\tau_L}{J}. \tag{7}$$

III. Linear Overparameterized Model

As stated in the introduction, the interest here is in online tracking of the value of T_R as it changes due to ohmic heating so that an accurate value is available to estimate the flux for a field oriented controller. However, the stator resistance value R_S will also vary due to ohmic heating so that its variation must also be taken into account. The electrical parameters M, L_S, and σ are assumed to be known and not varying. Measurements of the stator currents i_{Sa}, i_{Sb} and voltages u_{Sa}, u_{Sb}, as well as the position θ of the rotor are assumed to be available; velocity is then reconstructed from the position measurements. However, the rotor flux linkages are not assumed to be measured.

Standard methods for parameter estimation are based on equalities where known signals depend *linearly* on unknown parameters. However, the induction motor model described before does not fit into this category unless the rotor flux linkages are measured. The first step is to eliminate the fluxes ψ_{Rx}, ψ_{Ry} and their derivatives $d\psi_{Rx}/dt, d\psi_{Ry}/dt$. The four equations

(3), (4), (5), and (6), can be used to solve for ψ_{Rx}, ψ_{Ry}, $d\psi_{Rx}/dt, d\psi_{Ry}/dt$, but one is left without another independent equation to set up a regressor system for the identification algorithm. Consequently, a new set of independent equations is found by differentiating (3) and (4) to obtain

$$\frac{1}{\sigma L_s}\frac{du_{Sx}}{dt} = \frac{d^2 i_{Sx}}{dt^2} + \gamma\frac{di_{Sx}}{dt} - \frac{\beta}{T_R}\frac{d\psi_{Rx}}{dt} - n_p\beta\omega\frac{d\psi_{Ry}}{dt} - n_p\beta\psi_{Ry}\frac{d\omega}{dt} - n_p\omega\frac{di_{Sy}}{dt} - n_p i_{Sy}\frac{d\omega}{dt} \quad (8)$$

and

$$\frac{1}{\sigma L_s}\frac{du_{Sy}}{dt} = \frac{d^2 i_{Sy}}{dt^2} + \gamma\frac{di_{Sy}}{dt} - \frac{\beta}{T_R}\frac{d\psi_{Ry}}{dt} + n_p\beta\omega\frac{d\psi_{Rx}}{dt} + n_p\beta\psi_{Rx}\frac{d\omega}{dt} + n_p\omega\frac{di_{Sx}}{dt} + n_p i_{Sx}\frac{d\omega}{dt}. \quad (9)$$

Next, (3), (4), (5), and (6) are solved for ψ_{Rx}, ψ_{Ry}, $d\psi_{Rx}/dt, d\psi_{Ry}/dt$ and substituted into (8) and (9) to obtain

$$\begin{aligned} 0 = &- \frac{d^2 i_{Sx}}{dt^2} + \frac{di_{Sy}}{dt}n_p\omega + \frac{1}{\sigma L_S}\frac{du_{Sx}}{dt} - \left(\gamma + \frac{1}{T_R}\right)\frac{di_{Sx}}{dt} \\ &- i_{Sx}\left(-\frac{\beta M}{T_R^2} + \frac{\gamma}{T_R}\right) + i_{Sy}n_p\omega\left(\frac{1}{T_R} + \frac{\beta M}{T_R}\right) \\ &+ \frac{u_{Sx}}{\sigma L_S T_R} + n_p\frac{d\omega}{dt}i_{Sy} - n_p\frac{d\omega}{dt}\frac{1}{\sigma L_S\left(1 + n_p^2\omega^2 T_R^2\right)} \\ &\times\Bigg(-\sigma L_S T_R\frac{di_{Sy}}{dt} - \gamma i_{Sy}\sigma L_S T_R - i_{Sx}n_p\omega\sigma L_S T_R \\ &- \frac{di_{Sx}}{dt}n_p\omega\sigma L_S T_R^2 - \gamma i_{Sx}n_p\omega\sigma L_S T_R^2 \\ &+ i_{Sy}n_p^2\omega^2\sigma L_S T_R^2 + n_p\omega T_R^2 u_{Sx} + T_R u_{Sy}\Bigg) \end{aligned} \quad (10)$$

$$\begin{aligned} 0 = &- \frac{d^2 i_{Sy}}{dt^2} - \frac{di_{Sx}}{dt}n_p\omega + \frac{1}{\sigma L_S}\frac{du_{Sy}}{dt} - \left(\gamma + \frac{1}{T_R}\right)\frac{di_{Sy}}{dt} \\ &- i_{Sy}\left(-\frac{\beta M}{T_R^2} + \frac{\gamma}{T_R}\right) - i_{Sx}n_p\omega\left(\frac{1}{T_R} + \frac{\beta M}{T_R}\right) \\ &+ \frac{u_{Sy}}{\sigma L_S T_R} - n_p\frac{d\omega}{dt}i_{Sx} + n_p\frac{d\omega}{dt}\frac{1}{\sigma L_S\left(1 + n_p^2\omega^2 T_R^2\right)} \\ &\times\Bigg(-\sigma L_S T_R\frac{di_{Sx}}{dt} - \gamma i_{Sx}\sigma L_S T_R + i_{Sy}n_p\omega\sigma L_S T_R \\ &+ \frac{di_{Sy}}{dt}n_p\omega\sigma L_S T_R^2 + \gamma i_{Sy}n_p\omega\sigma L_S T_R^2 \\ &+ i_{Sx}n_p^2\omega^2\sigma L_S T_R^2 - n_p\omega T_R^2 u_{Sy} + T_R u_{Sx}\Bigg). \end{aligned} \quad (11)$$

This set of equations may be rewritten in regressor form as

$$y(t) = W(t)K \quad (12)$$

where $W \in \mathbb{R}^{2\times 8}$, $K \in \mathbb{R}^8$ are given at the bottom of the page and $y \in \mathbb{R}^2$ is given by and

$$y \triangleq \begin{bmatrix} \frac{d^2 i_{Sx}}{dt^2} - n_p i_{Sy}\frac{d\omega}{dt} - n_p\omega\frac{di_{Sy}}{dt} - n_p^2\omega^2 M\beta i_{Sx} - \frac{du_{Sx}/dt}{\sigma Ls} \\ \frac{d^2 i_{Sy}}{dt^2} + n_p i_{Sx}\frac{d\omega}{dt} + n_p\omega\frac{di_{Sx}}{dt} - n_p^2\omega^2 M\beta i_{Sy} - \frac{dv_{Sy}/dt}{\sigma Ls} \end{bmatrix}.$$

As $M^2/L_R = (1-\sigma)L_S$, $M\beta = (1-\sigma)/\sigma$, $\gamma = (R_S/\sigma L_S) + (1/\sigma L_S)(1/T_R)(M^2/L_R) = (R_S/\sigma L_S) + (1/\sigma L_S)(1/T_R)(1-\sigma)L_S$ it is shown that y and W depend only on known quantities while the unknowns R_S, T_R are contained only within the parameter vector K. The components of the parameter vector K are related as

$$\begin{aligned} K_3 &= K_2^2 \\ K_4 &= K_1 K_2 \\ K_5 &= 1/K_2 \\ K_6 &= K_1/K_2 \\ K_7 &= K_1/K_2^2 \\ K_8 &= 1/K_2^2 \end{aligned} \quad (13)$$

showing that only the two parameters K_1, K_2 are independent. These two parameters determine R_S and T_R by

$$\begin{aligned} T_R &= 1/K_2 \\ R_S &= \sigma L_S K_1 - (1-\sigma)L_S K_2. \end{aligned} \quad (14)$$

Though the system regressor is linear in the parameters, it is overparameterized, resulting in poor numerical conditioning if

$$W \triangleq \begin{bmatrix} -\frac{di_{Sx}}{dt} & -\frac{di_{Sx}}{dt} + n_p\omega i_{Sy} + n_p\omega M\beta i_{Sy} + \frac{u_{Sx}}{\sigma Ls} & M\beta i_{Sx} & -i_{Sx} & n_p\frac{di_{Sy}}{dt}\frac{d\omega}{dt} + n_p^2\left(\omega i_{Sx}\frac{d\omega}{dt} - \omega^2\frac{di_{Sx}}{dt}\right) \\ -\frac{di_{Sy}}{dt} & -\frac{di_{Sy}}{dt} - n_p\omega i_{Sx} - n_p\omega M\beta i_{Sx} + \frac{u_{Sy}}{\sigma Ls} & M\beta i_{Sy} & -i_{Sy} & -n_p\frac{di_{Sx}}{dt}\frac{d\omega}{dt} + n_p^2\left(\omega i_{Sy}\frac{d\omega}{dt} - \omega^2\frac{di_{Sy}}{dt}\right) \end{bmatrix}$$

$$\begin{bmatrix} n_p\frac{di_{Sy}}{dt}\frac{d\omega}{dt} + n_p^2\left(\omega i_{Sx}\frac{d\omega}{dt} - \omega^2\frac{di_{Sx}}{dt}\right) + n_p^3\omega^3 i_{Sy}(1+M\beta) + \frac{1}{\sigma Ls}\left(n_p^2\omega^2 u_{Sx} - n_p u_{Sy}\frac{d\omega}{dt}\right) \\ -n_p\frac{di_{Sx}}{dt}\frac{d\omega}{dt} + n_p^2\left(\omega i_{Sy}\frac{d\omega}{dt} - \omega^2\frac{di_{Sy}}{dt}\right) - n_p^3\omega^3 i_{Sx}(1+M\beta) + \frac{1}{\sigma Ls}\left(n_p^2\omega^2 u_{Sy} + n_p u_{Sx}\frac{d\omega}{dt}\right) \end{bmatrix}$$

$$\begin{bmatrix} n_p i_{Sy}\frac{d\omega}{dt} - n_p^2\omega^2 i_{Sx} & n_p^2\left(i_{sx}\omega\frac{d\omega}{dt} - \omega^2\frac{di_{Sx}}{dt}\right) & n_p^2\left(\omega\frac{di_{Sx}}{dt}\frac{d\omega}{dt} - \omega^2\frac{d^2 i_{Sx}}{dt^2}\right) + \frac{di_{Sy}}{dt}n_p^3\omega^3 - \frac{n_p^2}{\sigma Ls}\left(\omega u_{Sx}\frac{d\omega}{dt} - \omega^2\frac{du_{Sx}}{dt}\right) \\ -n_p i_{Sx}\frac{d\omega}{dt} - n_p^2\omega^2 i_{Sy} & n_p^2\left(i_{sy}\omega\frac{d\omega}{dt} - \omega^2\frac{di_{Sy}}{dt}\right) & n_p^2\left(\omega\frac{di_{Sy}}{dt}\frac{d\omega}{dt} - \omega^2\frac{d^2 i_{Sy}}{dt^2}\right) - \frac{di_{Sx}}{dt}n_p^3\omega^3 - \frac{n_p^2}{\sigma Ls}\left(\omega u_{Sy}\frac{d\omega}{dt} - \omega^2\frac{du_{Sy}}{dt}\right) \end{bmatrix}$$

$$K \triangleq \begin{bmatrix} \gamma & \frac{1}{T_R} & \frac{1}{T_R^2} & \frac{\gamma}{T_R} & T_R & \gamma T_R & \gamma T_R^2 & T_R^2 \end{bmatrix}^T$$

standard least-squares techniques are used. A more serious difficulty with identifying such an overparameterized model is that the $K_i's$ that are obtained do not necessarily satisfy the constraints (13), leaving the engineer with the problem of extracting γ and T_R from these inconsistent parameter values. Note also that γ and T_R can be estimated from K_1 and K_2 as in (14), but K_5 and K_6 could also be used, as well as many other combinations of parameters. How should this be done, given that the estimates are not equally precise due to various sensitivities in the least-squares criterion? The method proposed here avoids this issue altogether by minimizing the error while ensuring satisfaction of the constraints. This is described in Section IV.

IV. Least-Squares Identification [19]–[21]

Equation (12) can be rewritten as

$$y(nT) = W(nT)K \tag{15}$$

where T is the sample period, nT is the nth sample time at which a measurement is taken, and K is the vector of unknown parameters. If the constraint (13) is ignored, then the system is an overparameterized linear least-squares problem. In this case, theoretically an exact unique solution for the unknown parameter vector K may be determined after several time instants. However, several factors contribute to errors which make (15) only approximately valid in practice. Specifically, both $y(nT)$ and $W(nT)$ are measured through signals that are noisy due to quantization and differentiation. Further, the dynamic model of the induction motor is only an approximate representation of the real system. These sources of error result in an inconsistent system of equations. To find a solution for such a system, the least-squares criterion is used. Specifically, given $y(nT)$ and $W(nT)$, where $y(nT) = W(nT)K$, one defines

$$E^2(K) = \sum_{n=1}^{N} |y(nT) - W(nT)K|^2 \tag{16}$$

as the *residual error* associated to a parameter vector K. Then, the least-squares estimate K^* is chosen such that $E^2(K)$ is minimized for $K = K^*$. The function $E^2(K)$ is quadratic and therefore, has a unique minimum at the point where $\partial E^2(K)/\partial K = 0$. Solving this expression for K^* yields the least-squares solution to $y(nT) = W(nT)K$ as

$$K^* = \left[\sum_{n=1}^{N} W^T(nT)W(nT)\right]^{-1} \left[\sum_{n=1}^{N} W^T(nT)y(nT)\right]. \tag{17}$$

When the system model is overparameterized (as in the application here), the expression (17) will lead to an ill conditioned solution for K^*. That is, small changes in the data $W(nT)$, $y(nT)$ lead to large changes in the value computed for K^*. To get around this problem, a *nonlinear* least-squares approach is taken which involves minimizing

$$E^2(K) = \sum_{n=1}^{N} |y(nT) - W(nT)K|^2 = R_y - 2R_{Wy}^T K + K^T R_W K \tag{18}$$

subject to the constraints (13), where

$$R_y \triangleq \sum_{n=1}^{N} y^T(nT)y(nT)$$
$$R_{Wy} \triangleq \sum_{n=1}^{N} W^T(nT)y(nT)$$
$$R_W \triangleq \sum_{n=1}^{N} W^T(nT)W(nT).$$

On physical grounds, the parameters K_1, K_2 are constrained to the region

$$0 < K_1 < \infty, \quad 0 < K_2 < \infty. \tag{19}$$

Also, based on physical grounds, the squared error $E^2(K)$ will be minimized in the interior of this region. Define the new error function $E_p^2(K_1, K_2)$ as

$$\begin{aligned} E_p^2(K_1, K_2) &\triangleq \sum_{n=1}^{N} |y(nT) - W(nT)K|^2 \Big|_{\substack{K_3 = K_2^2 \\ K_4 = K_1 K_2 \\ \vdots}} \\ &= R_y - 2R_{Wy}^T K \Big|_{\substack{K_3 = K_2^2 \\ K_4 = K_1 K_2 \\ \vdots}} \\ &\quad + (K^T R_W K)\Big|_{\substack{K_3 = K_2^2 \\ K_4 = K_1 K_2 \\ \vdots}}. \end{aligned} \tag{20}$$

As just explained, the minimum of (20) must occur in the interior of the region and therefore, at an extremum point. This then entails solving the two extrema equations

$$r_1(K_p) \triangleq \frac{\partial E_p^2(K_1, K_2)}{\partial K_1} = 0 \tag{21}$$

$$r_2(K_p) \triangleq \frac{\partial E_p^2(K_1, K_2)}{\partial K_2} = 0. \tag{22}$$

The partial derivatives in (21) and (22) are *rational* functions in the parameters K_1, K_2. Defining

$$p_1(K_1, K_2) \triangleq K_2^4 r_1(K_1, K_2) = K_2^4 \frac{\partial E_p^2(K_1, K_2)}{\partial K_1} \tag{23}$$

$$p_2(K_1, K_2) \triangleq K_2^5 r_2(K_1, K_2) = K_2^5 \frac{\partial E_p^2(K_1, K_2)}{\partial K_2} \tag{24}$$

results in the $p_i(K_1, K_2)$ being *polynomials* in the parameters K_1, K_2 and having the same positive zero set (i.e., the same roots satisfying $K_i > 0$) as (21) and (22). The degrees of the polynomials p_i are given in the following table:

	$\deg K_1$	$\deg K_2$
$p_1(K_1, K_2)$	1	7
$p_2(K_1, K_2)$	2	8

These two polynomials can be rewritten in the form

$$p_1(K_1, K_2) = a_1(K_2)K_1 + a_0(K_2) \tag{25}$$

$$p_2(K_1, K_2) = b_2(K_2)K_1^2 + b_1(K_2)K_1 + b_0(K_2). \tag{26}$$

A systematic procedure to find all possible solutions to a set of polynomials is provided by the elimination theory through the method of resultants [22] and [23]. (This method was used in [4].) However, in this special case, $p_1(K_1, K_2)$ is of degree 1 in K_1 and can be solved directly. Substituting $K_1 = -a_0(K_2)/a_1(K_2)$ from $p_1(K_1, K_2) = 0$ into $p_2(K_1, K_2) = 0$ and multiplying the result through by $a_1^2(K_2)$, one obtains the (resultant) polynomial

$$r(K_2) = a_0^2(K_2)b_2(K_2) - a_0(K_2)a_1(K_2)b_1(K_2) + a_1^2(K_2)b_0(K_2) \quad (27)$$

where $\deg_{K_2}\{r\} = 20$. The roots of (27) are the only possible candidates for the values of K_2 that satisfy $p_1(K_1, K_2) = p_2(K_1, K_2) = 0$ for some K_1. In the online implementation, the coefficients of the polynomials $a_1(K_2)$, $a_0(K_2)$, $b_2(K_2)$, $b_1(K_2)$, and $b_0(K_2)$ [whose explicit expressions in terms of the elements of the matrices R_W and R_{Wy} are known *a priori* vis-a-vis (20), (23), and (24)] are computed and stored during data collection. The coefficients of the polynomial $r(K_2)$ are then computed online according to (27) by vector convolution, addition, and subtraction. The positive roots K_{2i} of $r(K_2) = 0$ are next computed and then substituted into $p_1(K_1, K_{2i}) = 0$ and solved for its positive roots K_{1j}. By this method of back solving, all possible (finite number) candidate solutions (K_{1j}, K_{2i}) are found, and the one that gives the smallest squared error, i.e., the smallest value of $E_p^2(K_1, K_2)$, is chosen.

A. Numerical Issues

After finding the solution that minimizes $E_p^2(K_1, K_2)$, one needs to know if the solution makes sense. For example, in the *linear* least-squares problem, there is a unique well defined solution provided that the regressor matrix is nonsingular (i.e., the matrix R_W in (18) assuming it is not overparameterized). In the nonlinear case here, a Taylor series expansion about the computed minimum point (K_1^*, K_2^*) gives

$$E_p^2(K_1, K_2) = E_p^2(K_1^*, K_2^*) + \frac{1}{2}\sum_{i=1,j=1}^{2}(K_i - K_i^*) \times \frac{\partial^2 E_p^2(K_1^*, K_2^*)}{\partial K_i \partial K_j}(K_j - K_j^*) + \cdots. \quad (28)$$

One then checks that the Hessian matrix $\partial^2 E_p^2(K_1^*, K_2^*)/\partial K_i \partial K_j$ is positive definite to ensure that the data is sufficiently rich to identify the parameters.

Further, in the *linear* least-squares problem (i.e., solving $R_{Wy} = R_W K$ with the model *not* overparameterized) one checks that the condition number of the regressor matrix R_W is small enough to ensure that the solution vector K is not too sensitive to the data. In the nonlinear approach here, one could check the sensitivity of roots of (27) to its coefficients. That is, with $r(K_2) = \sum_{i=0}^{20} r_i K_2^i$, for each $i = 1, \ldots, 20$ one perturbs the ith coefficient $r_i \to r_i + \Delta r$ and computes $\partial K_2/\partial r_i \approx (K_2^i - K_2^*)/\Delta r$ (K_2^i is the root corresponding to K_2^* with r_i replaced by $r_i + \Delta r$).

V. Error Estimates

Naturally, it is desirable to have a meaningful way to evaluate the confidence in the identification scheme. More specifically, one would want to know how well $W(n)K$ matches the data $y(n)$ and also how sensitive the residual error is with respect to the parameters K_i. To treat these issues, a residual error index and a set of parametric error indices are defined. In judging the performance of the algorithm presented in this brief, these measures are used.

Note that the measures of uncertainties developed hereafter do not rely on typical statistical measures used in system identification [20]. Because of the nonlinearity of the model, the effect of measurement noise cannot be considered to be an additive term uncorrelated with the signals $W(n)$ and $y(n)$. In fact, even if the system was linear, the assumption of independent error statistics would be dubious because of modelling errors. For this reason, there has been a considerable effort in the literature to address the problem under new assumptions (see [24], [25], and the special issue [26]). The measures of uncertainties derived hereafter are comparable to those of [24].

A. Residual Error Index

Equation (20) defines the residual error $E_p^2(K_p)$ at K_p^* where a simple observation shows that $0 \le E^2(K_p^*) \le R_y$. To develop a relative measure of how well the data fits the model, define the *residual error index* to be (see [10])

$$E_I = \sqrt{\frac{E_p^2(K_p^*)}{R_y}} \quad (29)$$

which is zero if $E^2(K_p^*) = 0$, and 1 when $K_p^* = 0$ (so $E^2(K_p^*) = R_y$). Therefore, the residual error index E_I, ranges from 0 to 1, where $E_I = 0$ indicates a perfect fit between model and data. The residual error index E_I is usually nonzero due to noise, unmodeled dynamics and nonlinearities. In the worst case, $E_I = 1$, which would mean that the residual error has a magnitude comparable to that of the measurement $y(n)$.

B. Parametric Error Indices

In addressing the issue of sensitivity of K_p^* to errors, we recall that

$$\left[\frac{\partial E_p^2(K_p)}{\partial K_p}\right]_{K_p = K_p^*} = 0.$$

Therefore, it is not possible to use the derivative of the residual error as a measure of how sensitive the error is with respect to K. An alternative is to define δK as the variation in K such that the increase of error is, for example, equal to 25% of the residual error $E_p^2(K^*)$ itself (see Fig. 1). Specifically, define a parametric error index as the *maximum* value of δK_i for $i = 1$, 2 such that

$$E^2(K_p^* + \delta K_p) = 1.25 E^2(K_p^*) \quad (30)$$

where $\delta K_p \triangleq [\delta K_1 \ \delta K_2]^T$.

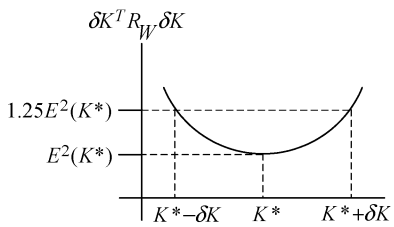

Fig. 1. $E_p^2(K^* + \delta K)$ versus δK.

In words, for all δK_p that result in a 25% increase in the residual error, find the maximum value of δK_i for $i = 1, 2$. The parametric error index δK_i indicates the amount by which K_i, the ith component of K, could vary without causing more than a 25% increase in the residual error. A large parametric error index indicates that the parameter estimate could vary greatly without a large change in the residual error. Thus, the accuracy of the parameter estimates would be in doubt. Likewise, a small parametric error indicates that the residual error is very sensitive to changes in the parameter estimates. In these cases, the parameter estimates may be considered more accurate. In any case, the error indices should not be considered as actual errors, but rather as orders of magnitude of the errors to be expected, to guide the identification process, and to warn about unreliable results.

Obviously, the choice of a parametric error index as corresponding to a 25% increase in the residual error is somewhat subjective. A different level of residual error would lead to a scaling of all the components of the parametric error index by a common factor. An alternative would be to select a residual error level corresponding to a known bound on the measurement noise (thus, the algorithm of [24]). While such an assumption leads to rigorous bounds on the parametric errors, the noise bound itself would still be highly subjective as it would have to account for modelling errors as well as measurement noise.

To compute the parametric error indices, for $i = 1, 2$ one maximizes δK_i subject to (30). This is straightforwardly setup as an unconstrained optimization using Lagrange multipliers by maximizing

$$\delta K_i + \lambda \left(E^2\left(K_p^* + \delta K_p\right) - 1.25E^2\left(K_p^*\right)\right) \tag{31}$$

over all possible $\delta K_p \triangleq [\delta K_1\ \delta K_2]^T$ and λ. For example, with $i = 1$, the extrema are solutions to

$$1 + \lambda \frac{\partial\left(E^2\left(K_p^* + \delta K_p\right) - 1.25E^2\left(K_p^*\right)\right)}{\partial \delta K_1} = 0 \tag{32}$$

$$\lambda \frac{\partial\left(E^2\left(K_p^* + \delta K_p\right) - 1.25E^2\left(K_p^*\right)\right)}{\partial \delta K_2} = 0 \tag{33}$$

$$E^2\left(K_p^* + \delta K_p\right) - 1.25E^2\left(K_p^*\right) = 0. \tag{34}$$

Equations (32)–(34) are rational functions in the three unknowns δK_1, δK_2, and λ. Multiplying through by powers δK_1 and δK_2, they can be transformed into three *polynomial* equations in the three unknowns of δK_1, δK_2, and λ, where elimination theory can then be used to solve this system [22], [23], [27].

VI. Experimental Results

A three phase, 230 V, 375 W (0.5 Hp), 1735 r/min ($n_p = 2$ pole-pair) induction machine was used for the experiments. A 4096 pulse/rev optical encoder was attached to the motor for position measurements. The stator inductance is $L_S = 0.2908$ H and the leakage factor is $\sigma = 0.096$. The motor was connected to a 10-kHz PWM Allen–Bradley inverter (cat no. 1305) used as a three-phase source. The real-time computing platform is from Opal-RT which has an AMD Athlon 2000+ (1.66 GHz) processor with 128 MB of RAM. The stator currents and voltages along with the rotor position were sampled at 4 kHz. Filtered differentiation (using digital filters) was used for calculating the acceleration and the derivatives of the voltages and currents. Specifically, the signals were filtered through a third-order low pass (500-Hz cutoff) Butterworth filter followed by reconstruction of the derivatives using $dx(t)/dt = (x(t) - x(t-T))/T$, where T is the sampling period. As filtering and differentiation are linear operations, their order of implementation does not matter. As discussed in [28], if one differentiates once or more the output of a low-pass Butterworth filter, the resulting signals are equivalent to the state variables of a state-space implementation of the filter (as long as the order of the filter is higher than the order of differentiation), which is equivalent to the implementation used here. At each time $t = nT$, the following is done.

1) The A/Ds for the voltages and currents are sampled and put through a 3-2 transformation to obtain $u_{Sa}(nT)$, $u_{Sb}(nT)$, $i_{Sa}(nT)$, and $i_{Sb}(nT)$. The encoder measurement (sample) $\theta(nT)$ is also read.
2) These voltage and current samples are rotated by the angle $n_p\theta$ using the sampled position measurement to obtain $u_{Sx}(nT)$, $u_{Sy}(nT)$, $i_{Sx}(nT)$, and $i_{Sy}(nT)$.
3) The computed quantities $u_{Sx}(nT)$, $u_{Sy}(nT)$, $i_{Sx}(nT)$, $i_{Sy}(nT)$, along with $\theta(nT)$ are each input to a low pass filter and their respective differentiation filters.
4) The outputs of all the filters at $t = NT$ (low pass and differentiation) are then used to compute $W(nT)$.
5) At each time step, R_W, R_{Wy}, R_y are updated according to $R_W((n-1)T) + W^T(nT)W(nT) \rightarrow R_W(nT)$, $R_{Wy}((n-1)T) + W^T(nT)y(nT) \rightarrow R_{Wy}(nT)$, and $R_y((n-1)T) + y^T(nT)y(nT) \rightarrow R_y(nT)$, respectively.

After the data collection period (1 s), the parameters are computed as follows.

1) The coefficients of (23) and (24) are computed directly from the components of R_W, R_{Wy}, R_y.
2) The coefficients of the polynomial $r(K_2)$ are then computed online according to (27) by vector convolution, addition, and subtraction.
3) The positive roots K_{2i} of $r(K_2) = 0$ are next computed, then substituted into $p_1(K_1, K_{2i}) = 0$ and solved for its positive roots K_{1j}. By this method of back solving, all possible (finite number) candidate solutions (K_{1j}, K_{2i}) are found.
4) The coefficients of $E_p^2(K_1, K_2)$ in (20) are then computed directly from components of R_W, R_{Wy}, R_y. The root pair that gives the smallest squared error, i.e., the smallest value

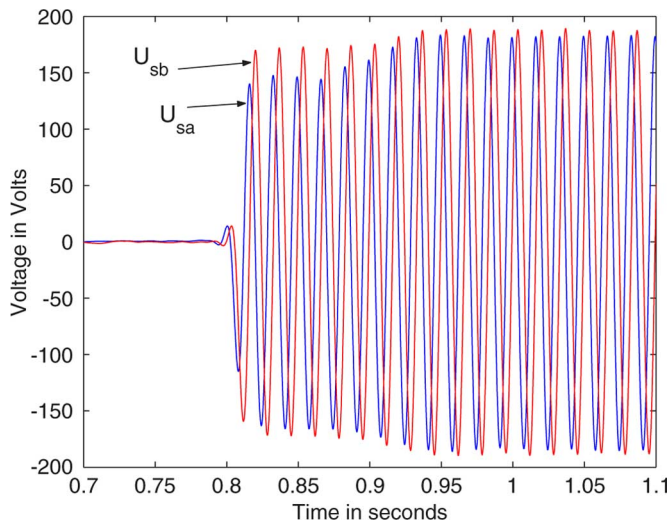

Fig. 2. Sampled two phase equivalent voltages u_{Sa} and u_{Sb}.

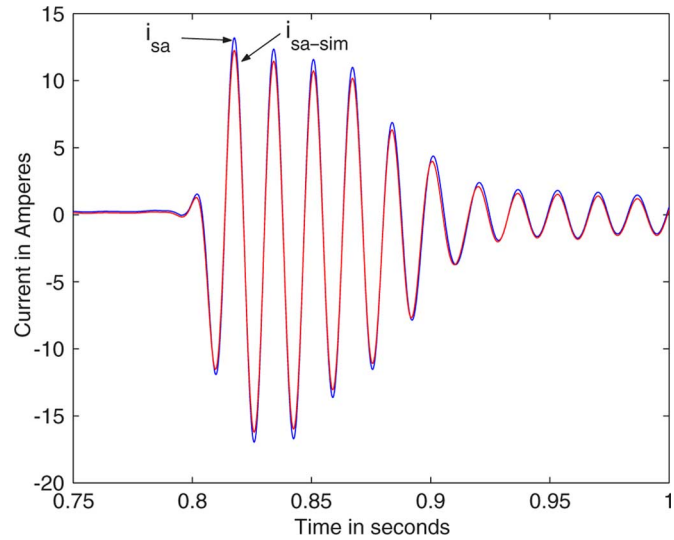

Fig. 3. Phase a current i_{Sa} and its simulated response i_{Sa_sim}.

of $E_p^2(K_1, K_2)$, is chosen. The values of T_R, R_S are found using (14).

The computation of the roots of the resultant polynomial (27) was programmed in C and embedded in a S-function model in Simulink. The Simulink file was converted into a real-time excutable file using RT-LAB from Opal-RT Technologies [29]. After collecting the data for 1 s, the S-function evaluated the resultant polynomial, computed its roots, and then completed the estimation algorithm to obtain T_R online (i.e., the parameter update is every second).

VII. Experiment Set 1

In the first experiment, a three-phase 60-Hz voltage was applied to the induction machine to bring it from zero speed up to rated speed with no load. The voltages and currents were put through a 3-2 transformation to obtain the two-phase equivalent voltages u_{Sa}, u_{Sb} plotted in Fig. 2. The sampled two-phase equivalent current i_{Sa} and its simulated response i_{Sa_sim} are shown in Fig. 3 (the simulated response is explained in Section VII-A). The phase b current i_{Sb} is similar, but shifted by $\pi/(2n_p)$. The calculated speed ω (from the position measurements) and the simulated speed ω_{sim} are shown in Fig. 4 (the simulated response is explained in Section VII-A).

In this case, the parameter values that resulted in the minimum least-squares error and their corresponding parametric error indices are shown in the table at the bottom of the page.

Using (14), it follows that

$$T_R = 0.12 \text{ s} \tag{35}$$

$$R_S = 5.04\ \Omega. \tag{36}$$

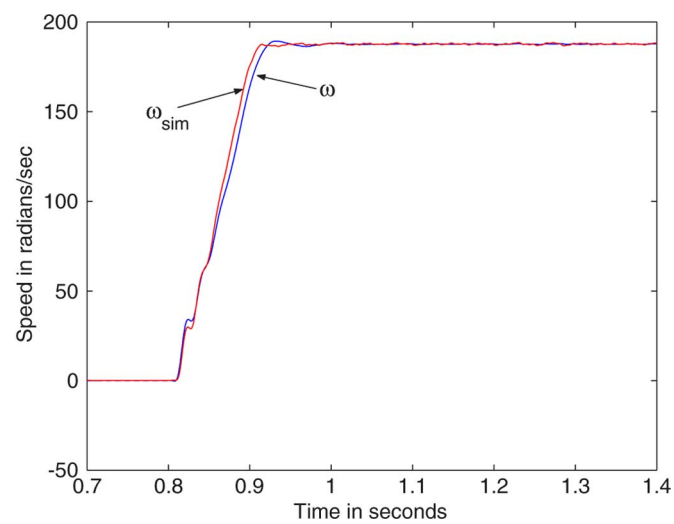

Fig. 4. Calculated speed ω and simulated speed ω_{sim}.

The Hessian matrix at the minimum point is [see (28)]

$$\left\{\frac{\partial^2 E^2(K_p)}{\partial K_i \partial K_j}\right\}_{|(K_1^*, K_2^*)} = \begin{bmatrix} 0.3105 & 0.000411 \\ 0.000411 & 104.95 \end{bmatrix}$$

which is positive definite.

A. Simulation of the Experimental Motor

Another useful way to evaluate the identified parameters (35) and (36) is to simulate the motor using these values and the measured voltages as input. The experimental voltages, shown in Fig. 2, were then used as input to a simulation of the model (1) using the parameter values from (35) and (36). The resulting

Parameter	Estimated Value	P.E. Index with $1.25E^2\left(K_p^*\right)$
K_1^*	243.51	100.17
K_2^*	8.06	2.21

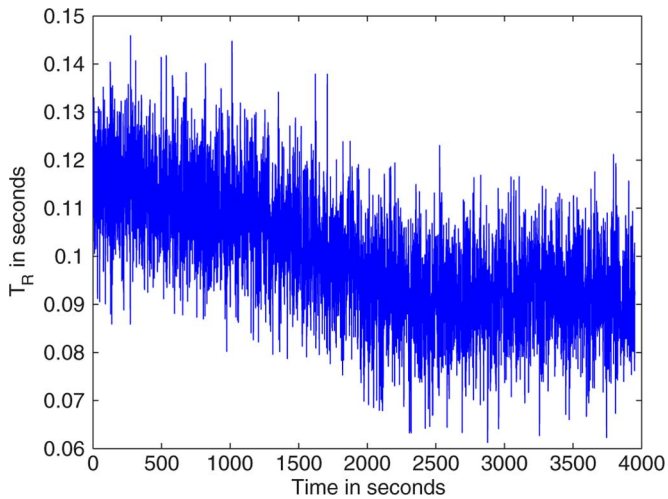

Fig. 5. T_R estimation recorded each second over 1 h.

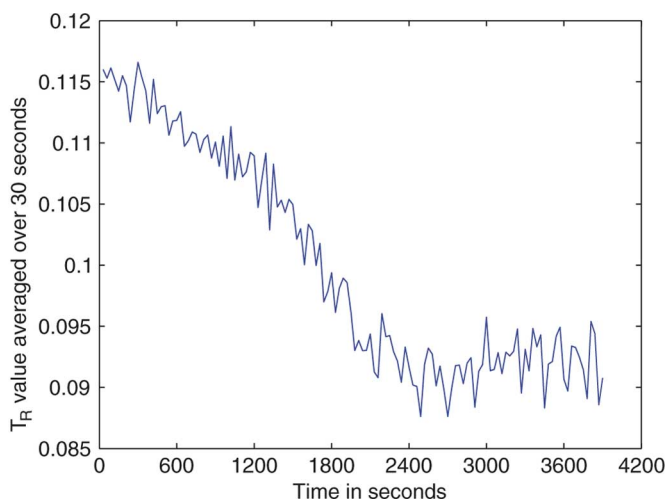

Fig. 6. T_R value averaged over previous 30 s.

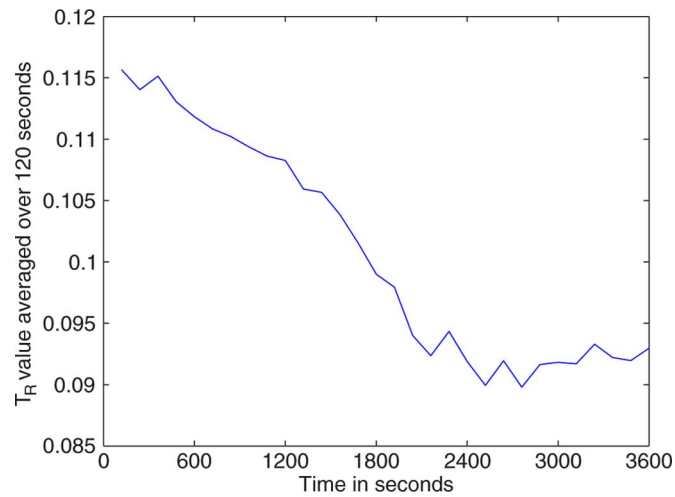

Fig. 7. T_R value averaged over previous 120 s.

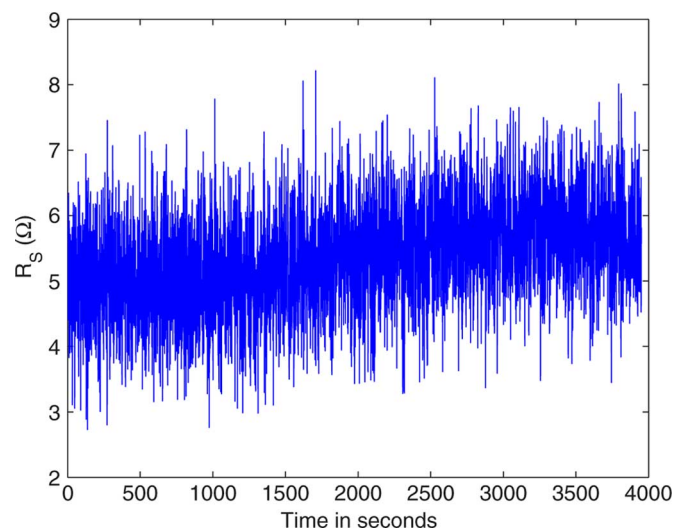

Fig. 8. R_S estimation recorded each second over 1 h.

phase a current i_{Sa_sim} from the simulation is shown in Fig. 3 and corresponds well with the actual measured current i_{Sa}. Similarly, the resulting speed ω_{sim} from the simulation is shown in Fig. 4, where the simulated speed is somewhat more oscillatory than the measured speed ω.

VIII. Experiment Set 2

In the second set of experiments the machine is run at constant speed under full load. The experimental platform is the same as in the first set of experiments. The induction machine was coupled with a dc machine to provide the load. The electrical frequency was chosen to be 30 Hz, and the rms stator current was 1.6 A (rated current at full load). There is sufficient information in the signals to determine T_R and R_S, even if the speed is constant; however, the sensitivity to noise is greater than in the test with varying speed. Nevertheless, the experiment shows that it is possible to track the rotor time constant in a situation that does not provide sufficient information for the determination of all the parameters of the machine (cf. [4]).

The induction machine was run with full rated load for about 1 h, where the temperature of the case of the induction machine changed from a room temperature of 22 to 44 °C as measured with an infrared thermometer. The value of T_R was estimated online each second and is plotted in Fig. 5. There is noticeable oscillation in the estimated values of T_R when the update is every second, but as expected, it is shown that the average value decreases as the temperature increases. One can average the estimated values of T_R shown in Fig. 5 over longer periods of time to discern the tendency. For example, if the averaging time interval is chosen to be 30 s, then the average over this time interval is given by

$$T_{R_30}(kT_{30}) = \frac{1}{30}\sum_{n=1}^{30} T_R\left([30(k-1)+n]\,T_1\right)$$

for $k = 1, 2, 3, \ldots$, with $T_{30} = 30T_1$ and $T_1 = 1$ s. This is plotted in Fig. 6.

Fig. 7 shows the calculated average value using 120 s for the averaging interval. Note that the plot shows that the estimated value of T_R begins with an average value of about 0.115 s when the machine is turned on and settles out at 0.09 s after the machine is heated up. Further, note that the "cold" value of T_R,

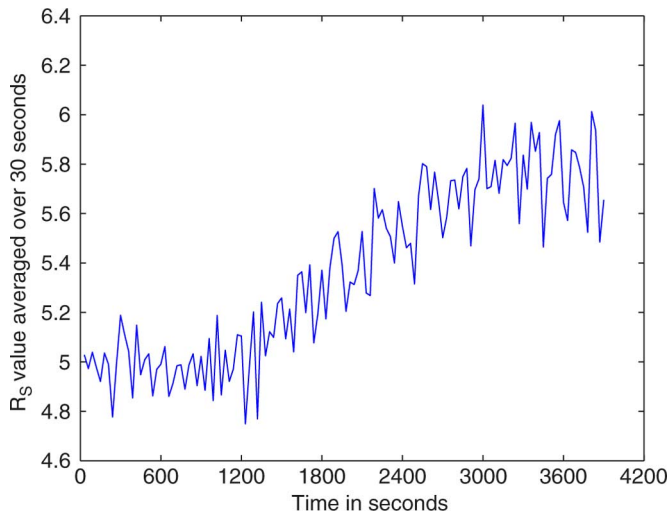

Fig. 9. R_S value averaged over previous 30 s.

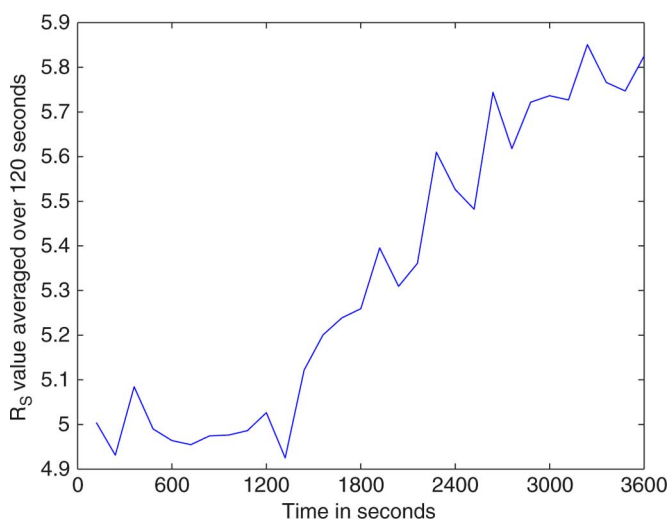

Fig. 10. R_S value averaged over previous 120 s.

namely 0.115 s, is less than 5% different from the (cold) value estimated when the motor was accelerating as given in (35). That is, two different types of experiments result in less than a 5% difference in the value estimated for T_R.

Fig. 8 gives the estimated values of R_S computed each second. Fig. 9 gives the result of averaging these values over a 30 s time interval while Fig. 10 gives the result of averaging these values over 120 s. Fig. 10 shows that the average value increases from 5 to 5.8 Ω approximately. The estimate of R_S is more sensitive to measurement noise because it appears in the equations only as $u_{Sa} - R_S i_{Sa}$, $u_{Sb} - R_S i_{Sb}$ and $U_{Sa} \gg R_S I_{Sa}$, $U_{Sb} \gg R_S I_{Sb}$ at constant speed under full load (U_{Sa} is the rms value of u_{Sa}, etc.). Note also that the "cold" value of R_S, namely 5 Ω, is essentially the same as the (cold) value obtained when the motor was accelerating as given in (36).

IX. Conclusion

In this brief, an online method for estimation of the rotor time constant and the stator resistance of an induction machine was presented. The problem was formulated as finding the parameters that best fit the model of the induction motor to measured output data of the motor in a least-squares sense. The method guarantees the parameters are found in a finite number of steps. Two different types of experiments were performed and their results were consistent. Important advantages of the proposed methodology include that it is an *online* method (so that the value of T_R can be continuously updated) and that it can estimate T_R under constant or varying speed conditions.

References

[1] W. Leonhard, *Control of Electrical Drives*, 3rd ed. Berlin, Germany: Springer-Verlag, 2001.

[2] J. Chiasson, *Modeling and High-Performance Control of Electric Machines*. New York: Wiley, 2005.

[3] C. Moons and B. D. Moor, "Parameter identification of induction motor drives," *Automatica*, vol. 31, no. 8, pp. 1137–1147, 1995.

[4] K. Wang, J. Chiasson, M. Bodson, and L. Tolbert, "A nonlinear least-squares approach for estimation of the induction motor parameters," *IEEE Trans. Autom. Contr.*, vol. 50, no. 10, pp. 1622–1628, Oct. 2005.

[5] ——, "An on-line method for tracking the rotor time constant of an induction machine," in *Proc. Amer. Contr. Conf.*, 2005, pp. 2739–2744.

[6] ——, "An on-line rotor time constant estimator for the induction machine," in *Proc. IEEE Int. Electric Mach. Drives Conf.*, 2005, pp. 608–614.

[7] M. Vélez-Reyes, K. Minami, and G. Verghese, "Recursive speed and parameter estimation for induction machines," in *Proc. IEEE Ind. Appl. Conf.*, 1989, pp. 607–611.

[8] M. Vélez-Reyes, W. L. Fung, and J. E. Ramos-Torres, "Developing robust algorithms for speed and parameter estimation in induction machines," in *Proc. IEEE Conf. Dec. Contr.*, 2001, pp. 2223–2228.

[9] M. Vélez-Reyes and G. Verghese, "Decomposed algorithms for speed and parameter estimation in induction machines," in *Proc. IFAC Nonlinear Contr. Syst. Des. Symp.*, 1992, pp. 156–161.

[10] J. Stephan, M. Bodson, and J. Chiasson, "Real-time estimation of induction motor parameters," *IEEE Trans. Ind. Appl.*, vol. 30, no. 3, pp. 746–759, May/Jun. 1994.

[11] J. Stephan, "Real-time estimation of the parameters and fluxes of induction motors," M.S. thesis, Carnegie Mellon Univ., Pittsburgh, PA, 1992.

[12] R. Marino, S. Peresada, and P. Tomei, "On line stator and rotor resistance estimation for induction motors," *IEEE Trans. Contr. Syst. Technol.*, vol. 8, no. 3, pp. 570–579, May 2000.

[13] P. Castaldi, W. Geri, M. Montanari, and A. Tilli, "A new adaptive approach for on-line parameter and state estimation of induction motors," *Contr. Eng. Practice*, vol. 13, pp. 81–94, Jan. 2005.

[14] H. A. Toliyat, E. Levi, and M. Raina, "A review of RFO induction motor parameter estimation techniques," *IEEE Trans. Energy Convers.*, vol. 18, no. 2, pp. 271–283, Jun. 2003.

[15] M. Véelez-Reyes, M. Mijalković, A. M. Stanković, S. Hiti, and J. Nagashima, "Output selection for tuning of field-oriented controllers: steady-state analysis," in *Conf. Record Ind. Appl. Society*, 2003, pp. 2012–2016.

[16] P. Vas, *Parameter Estimation, Condition Monitoring, and Diagnosis of Electrical Machines*. Oxford, U.K.: Clarendon, 1993.

[17] R. Marino, S. Peresada, and P. Valigi, "Adaptive input-output linearizing control of induction motors," *IEEE Trans. Autom. Contr.*, vol. 38, no. 2, pp. 208–221, Feb. 1993.

[18] M. Bodson, J. Chiasson, and R. Novotnak, "High performance induction motor control via input-output linearization," *IEEE Contr. Syst. Mag.*, vol. 14, no. 4, pp. 25–33, Aug. 1994.

[19] L. Ljung, *System Identification: Theory for the User*. Englewood Cliffs, NJ: Prentice-Hall, 1986.

[20] T. Söderström and P. Stoica, *System Identification*. Englewood Cliffs, NJ: Prentice-Hall, 1989.

[21] S. Sastry and M. Bodson, *Adaptive Control: Stability, Convergence, and Robustness*. Englewood Cliffs, NJ: Prentice-Hall, 1989.

[22] D. Cox, J. Little, and D. O'Shea, *IDEALS, VARIETIES, AND ALGORITHMS: An Introduction to Computational Algebraic Geometry and Commutative Algebra*, 2nd ed. Berlin, Germany: Springer-Verlag, 1996.

[23] J. von zur Gathen and J. Gerhard, *Modern Computer Algebra*. Cambridge, U.K.: Cambridge Univ. Press, 1999.

[24] J. Norton, "Identification of parameter bounds for ARMAX models from records with bounded noise," *Int. J. Contr.*, vol. 45, no. 2, pp. 375–390, 1987.

[25] R. Younce and C. Rohrs, "Identification with nonparametric uncertainty," *IEEE Trans. Autom. Contr.*, vol. AC-37, no. 6, pp. 715–728, Jun. 1992.

[26] "Special issue on system identification for robust control design," *IEEE Trans. Autom. Contr.*, vol. AC-37, no. 7, pp. 899–1008, Jul. 1992.

[27] K. Wang, "A methodology for solving the equations arising in nonlinear parameter identification problems: Application to induction machines," Ph.D. dissertation, The Univesity of Tennessee, Knoxville, TN, 2005.

[28] I.-J. Ha and S.-H. Lee, "An online identification method for both stator and rotor resistances of induction motors without rotational transducers," *IEEE Trans. Ind. Electron.*, vol. 47, no. 4, pp. 842–853, Aug. 2000.

[29] Opal-RT Technologies, Ann Arbor, MI, RT-LAB 2006. [Online]. Available: http://www.opal-rt.com