An Online Rotor Time Constant Estimator for the Induction Machine

Kaiyu Wang, John Chiasson, Marc Bodson and Leon M. Tolbert

Abstract— **Indirect field oriented control for induction machine requires the knowledge of rotor time constant to estimate the rotor flux linkages. Here an online method for estimating the rotor time constant and stator resistance is presented. The problem is formulated as a nonlinear least-squares problem and a procedure is presented that guarantees the minimum is found in a finite number of steps. Experimental results are presented. Two different approaches to implementing the algorithm online are discussed. Simulations are also presented to show how the algorithm works online.**

Index Terms— **Induction Motor, Rotor Time Constant, Parameter Identification**

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 machine parameters, and in particular the rotor time constant which can vary due to ohmic heating. The problem is further complicated by the fact that rotor variables are not usually available for measurement.

The induction motor parameters, which are required for field oriented control, 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 locked 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. For example, field oriented control requires knowledge of the rotor time constant $T_R = L_R/R_R$ (which varies significantly due to ohmic heating) in order to estimate the rotor flux linkages. The interest here is in tracking the value of T_R as it changes. The approach is a nonlinear least-squares method using measurements of the stator currents and voltages along with the rotor speed. Due to the nature of this technique, it lends itself directly to an online implementation and therefore can be used to track the rotor time constant.

K. Wang, J. Chiasson and L. M. Tolbert are with the ECE Department, University of Tennessee, Knoxville, TN 37996. wkaiyu@utk.edu, chiasson@utk.edu, tolbert@utk.edu.

M. Bodson is with the ECE Department, University of Utah, Salt Lake City, UT 84112 bodson@ece.utah.edu

L. M. Tolbert is also with Oak Ridge National Laboratory, Oak Ridge TN. tolbertlm@ornl.gov

Drs. Chiasson and Tolbert would like to thank Oak Ridge National Laboratory for partially supporting this work through the UT/Battelle contract no. 4000007596. Dr. Tolbert would also like to thank the National Science Foundation for partially supporting this work through contract NSF ECS-0093884.

Because the rotor state variables are not available measurements, the system identification model cannot be made linear in the parameters without overparameterizing the model. In the work here, the model is reformulated so that it is a *nonlinear* system identification problem that is not overparameterized. Further, it is shown how to actually solve for parameter vector that minimizes the residual error.

This proposed method improves upon the *linear* least-squares approach formulated in [1][2]. The work in [1][2] was limited in that the acceleration was required to be small and that the iterative method used to solve the least squares problem was not guaranteed to converge nor necessarily achieve the minimum. Here, elimination theory [3][4] is used to solve the nonlinear least squares problem which in turn guarantees the minimum is found without any requirements on the machine's speed or acceleration when collecting the data; the data need only be sufficiently rich as described in the paper. Experimental results are presented to demonstrate the validity of the approach. The authors first proposed this method in [5] and the present work discusses the online implementation of the algorithm.

A combined parameter identification and velocity estimation problem is discussed in [6][7][8]. Here the velocity estimation problem is not considered, but the velocity is allowed to vary. For a summary of the various techniques for tracking the rotor time constant, the reader is referred to the recent survey [9], the recent paper [10] and to the book [11].

The paper 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 least-squares identification problem is presented and shown to guarantee the minimum least-squares solution is found. Section V presents the results of the identification algorithm with both simulated and 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 the state space model of the system given by (cf.

0-7803-8987-5/05/$20.00 ©2005 IEEE.

[12][13])

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

where $\omega = d\theta/dt$ with θ the position of the rotor, n_p is the number of pole pairs, and i_{Sa}, i_{Sb} are the (two phase equivalent) stator currents and 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

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

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

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

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 d/q transformation. The stator voltages and the rotor fluxes are transformed as the currents resulting in the following model ([2])

$$\begin{aligned}
\frac{di_{Sx}}{dt} &= \frac{u_{Sx}}{\sigma L_S} - \gamma i_{Sx} + \frac{\beta}{T_R}\psi_{Rx} + n_p\beta\omega\psi_{Ry} + n_p\omega i_{Sy} \qquad (3) \\
\frac{di_{Sy}}{dt} &= \frac{u_{Sy}}{\sigma L_S} - \gamma i_{Sy} + \frac{\beta}{T_R}\psi_{Ry} - n_p\beta\omega\psi_{Rx} - n_p\omega i_{Sx} \qquad (4) \\
\frac{d\psi_{Rx}}{dt} &= \frac{M}{T_R}i_{Sx} - \frac{1}{T_R}\psi_{Rx} \qquad (5) \\
\frac{d\psi_{Ry}}{dt} &= \frac{M}{T_R}i_{Sy} - \frac{1}{T_R}\psi_{Ry} \qquad (6) \\
\frac{d\omega}{dt} &= \frac{Mn_p}{JL_R}(i_{Sy}\psi_{Rx} - i_{Sx}\psi_{Ry}) - \frac{\tau_L}{J}. \qquad (7)
\end{aligned}$$

III. Linear Overparameterized Model

As stated in the introduction, the interest here is in tracking 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 it must also be taken into account. The electrical parameters M, L_S, 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 above does not fit in 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), (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. A new set of independent equations are found by differentiating equations (3) and (4) to obtain

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

and

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

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

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

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

This set of equations may be rewritten in regressor form as

$$y(t) = W(t)K \tag{12}$$

where $W \in \Re^{2\times 8}, K \in \Re^8$ and $y \in \Re^2$ are given by

$$W = \left[\begin{array}{ccc} -\frac{di_{Sx}}{dt} & -\frac{di_{Sx}}{dt} + n_p\omega i_{Sy} + n_p\omega M\beta i_{Sy} + \frac{u_{Sx}}{\sigma Ls} \\ -\frac{di_{Sy}}{dt} & -\frac{di_{Sy}}{dt} - n_p\omega i_{Sx} - n_p\omega M\beta i_{Sx} + \frac{u_{Sy}}{\sigma Ls} \end{array}\right.$$

$$\begin{array}{ccc} M\beta i_{Sx} & -i_{Sx} & n_p\frac{di_{Sy}}{dt}\frac{d\omega}{dt} + n_p^2(\omega i_{Sx}\frac{d\omega}{dt} - \omega^2\frac{di_{Sx}}{dt}) \\ M\beta i_{Sy} & -i_{Sy} & -n_p\frac{di_{Sx}}{dt}\frac{d\omega}{dt} + n_p^2(\omega i_{Sy}\frac{d\omega}{dt} - \omega^2\frac{di_{Sy}}{dt}) \end{array}$$

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

$$\begin{array}{cc} n_p i_{Sy}\frac{d\omega}{dt} - n_p^2\omega^2 i_{Sx} & n_p^2(i_{sx}\omega\frac{d\omega}{dt} - \omega^2\frac{di_{Sx}}{dt}) \\ -n_p i_{Sx}\frac{d\omega}{dt} - n_p^2\omega^2 i_{Sy} & n_p^2(i_{sy}\omega\frac{d\omega}{dt} - \omega^2\frac{di_{Sy}}{dt}) \end{array}$$

$$\begin{array}{c} n_p^2(\omega\frac{di_{Sx}}{dt}\frac{d\omega}{dt} - \omega^2\frac{d^2 i_{Sx}}{dt^2}) + \frac{di_{Sy}}{dt}n_p^3\omega^3 \\ n_p^2(\omega\frac{di_{Sy}}{dt}\frac{d\omega}{dt} - \omega^2\frac{d^2 i_{Sy}}{dt^2}) - \frac{di_{Sx}}{dt}n_p^3\omega^3 \end{array}$$

$$\left.\begin{array}{c} -\frac{n_p^2}{\sigma Ls}(\omega u_{Sx}\frac{d\omega}{dt} - \omega^2\frac{du_{Sx}}{dt}) \\ -\frac{n_p^2}{\sigma Ls}(\omega u_{Sy}\frac{d\omega}{dt} - \omega^2\frac{du_{Sy}}{dt}) \end{array}\right],$$

$$K \triangleq \left[\begin{array}{cccccccc} \gamma & \frac{1}{T_R} & \frac{1}{T_R^2} & \frac{\gamma}{T_R} & T_R & \gamma T_R & \gamma T_R^2 & T_R^2 \end{array}\right]^T$$

and

$$y \triangleq \left[\begin{array}{c} \frac{d^2 i_{Sx}}{dt^2} - n_p i_{Sy}\frac{d\omega}{dt} - n_p\omega\frac{di_{Sy}}{dt} - n_p^2\omega^2 M\beta i_{Sx} \\ \frac{d^2 i_{Sy}}{dt^2} + n_p i_{Sx}\frac{d\omega}{dt} + n_p\omega\frac{di_{Sx}}{dt} - n_p^2\omega^2 M\beta i_{Sy} \end{array}\right.$$

$$\left.\begin{array}{c} -\frac{du_{Sx}/dt}{\sigma Ls} \\ -\frac{dv_{Sy}/dt}{\sigma Ls} \end{array}\right].$$

As $\frac{M^2}{L_R} = (1-\sigma)L_S$, $M\beta = (1-\sigma)/\sigma$, $\gamma = \frac{R_S}{\sigma L_S} + \frac{1}{\sigma L_S}\frac{1}{T_R}\frac{M^2}{L_R} = \frac{R_S}{\sigma L_S} + \frac{1}{\sigma L_S}\frac{1}{T_R}(1-\sigma)L_S$ it is seen that y and W depend only on known quantities while the unknowns R_S, T_R are contained only within K.

Though the system regressor is linear in the parameters, one cannot use standard least-squares techniques as the system is overparameterized. Specifically,

$$\begin{aligned} K_3 &= K_2^2, K_4 = K_1K_2, K_5 = 1/K_2, K_6 = K_1/K_2, \\ K_7 &= K_1/K_2^2, K_8 = 1/K_2^2 \end{aligned} \tag{13}$$

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

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

IV. Least-Squares Identification [14][15][16]

Equation (12) can be rewritten as

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

where n is the time instant 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 equation (15) only approximately valid in practice. Specifically, both $y(n)$ and $W(n)$ 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 algorithm is used. Specifically, given $y(n)$ and $W(n)$ where $y(n) = W(n)K$, one defines

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

as the *residual error* associated to a 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 $y(n) = W(n)K$ as

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

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 $W(n), y(n)$ 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

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

subject to the constraints (13), where

$$\begin{aligned} R_y &\triangleq \sum_{n=1}^{N} y^T(n)y(n), R_{Wy} \triangleq \sum_{n=1}^{N} W^T(n)y(n), \\ R_W &\triangleq \sum_{n=1}^{N} W^T(n)W(n). \end{aligned}$$

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

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

Also, based on physical grounds, the squared error $E^2(K)$ will be minimized in the interior of this region. Let

$$E^2(K_p) \triangleq \sum_{n=1}^{N} \left| y(n) - W(n)K \right|^2_{\substack{K_3=K_2^2 \\ K_4=K_1K_2 \\ \vdots}} \tag{20}$$

$$= R_y - 2R_{Wy}^T K\Big|_{\substack{K_3=K_2^2 \\ K_4=K_1K_2 \\ \vdots}} + \left(K^T R_W K\right)\Big|_{\substack{K_3=K_2^2 \\ K_4=K_1K_2 \\ \vdots}}$$

where

$$K_p \triangleq \begin{bmatrix} K_1 & K_2 \end{bmatrix}^T.$$

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 equations

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

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

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

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

$$p_2(K_p) \triangleq K_2^5 r_2(K_p) = K_2^5 \frac{\partial E^2(K_p)}{\partial K_2} \tag{24}$$

results in the $p_i(K_p)$ 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 the system (21)-(22). The degrees of the polynomials p_i are given in the table below.

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

All possible solutions to this set may be found using elimination theory as is now summarized.

Solving Systems of Polynomial Equations [3][4]

The question at hand is "Given two polynomial equations $a(K_1, K_2) = 0$ and $b(K_1, K_2) = 0$, how does one solve them simultaneously to eliminate (say) K_2?". A systematic procedure to do this is known as *elimination theory* and uses the notion of *resultants*. Briefly, one considers $a(K_1, K_2)$ and $b(K_1, K_2)$ as polynomials in K_2 whose coefficients are polynomials in K_1. Then, for example, letting $a(K_1, K_2)$ and $b(K_1, K_2)$ have degrees 3 and 2, respectively in K_2, they may be written in the form

$$a(K_1, K_2) = a_3(K_1)K_2^3 + a_2(K_1)K_2^2 + a_1(K_1)K_2 + a_0(K_1)$$
$$b(K_1, K_2) = b_2(K_1)K_2^2 + b_1(K_1)K_2 + b_0(K_1).$$

The $n \times n$ *Sylvester* matrix, where $n = \deg_{K_2}\{a(K_1, K_2)\} + \deg_{K_2}\{b(K_1, K_2)\} = 3 + 2 = 5$, is defined by

$$S_{a,b}(K_1) = \begin{bmatrix} a_0(K_1) & 0 & b_0(K_1) & 0 & 0 \\ a_1(K_1) & a_0(K_1) & b_1(K_1) & b_0(K_1) & 0 \\ a_2(K_1) & a_1(K_1) & b_2(K_1) & b_1(K_1) & b_0(K_1) \\ a_3(K_1) & a_2(K_1) & 0 & b_2(K_1) & b_1(K_1) \\ 0 & a_3(K_1) & 0 & 0 & b_2(K_1) \end{bmatrix} \tag{25}$$

The *resultant* polynomial is then defined by

$$r(K_1) = \text{Res}\Big(a(K_1, K_2), b(K_1, K_2), K_2\Big) \triangleq \det S_{a,b}(K_1) \tag{26}$$

and is the result of *eliminating* the variable K_2 from $a(K_1, K_2)$ and $b(K_1, K_2)$. In fact, the following is true.

Theorem 1: Any solution (K_1^0, K_2^0) of $a(K_1, K_2) = 0$ and $b(K_1, K_2) = 0$ must have $r(K_1^0) = 0$. [3][4].

Though the converse of this theorem is not necessarily true, the finite number of solutions of $r(K_1) = 0$ are the *only* possible candidates for the first coordinate (partial solutions) of the common zeros of $a(K_1, K_2)$ and $b(K_1, K_2)$. Whether or not such a partial solution results in a full solution is simply determined by back solving and checking the solution.

Using the polynomials (23)-(24), the variable K_1 is eliminated to obtain

$$r(K_2) \triangleq \text{Res}\Big(p_1(K_1, K_2), p_2(K_1, K_2), K_1\Big) \tag{27}$$

where $\deg_{K_1}\{r(K_2)\} = 20$. The parameter K_2 was chosen as the variable *not* eliminated because its degree is much higher than K_1 meaning it would have a larger (in dimension) Sylvester matrix. The positive roots of $r(K_2) = 0$ are found which are then substituted into $p_1 = 0$ (or $p_2 = 0$) to find the positive roots in K_2, etc. By this method of back solving, all possible (finite number) candidate solutions are found and one simply chooses the one that gives the smallest squared error.

Conditioning of the Nonlinear Least-Squares Problem

After finding the solution that gives the minimal value for $E^2(K_p)$, 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 R_W is nonsingular (or in practical terms, its condition number is not too large). In the nonlinear case here, a Taylor series expansion about the computed minimum point $K_p^* = [K_1^*, K_2^*]^T$ gives $(i, j = 1, 2)$

$$E^2(K_p) = E^2(K_p^*) + \frac{1}{2}\left[K_p - K_p^*\right]^T \frac{\partial^2 E^2(K_p^*)}{\partial K_i \partial K_j}\left[K_p - K_p^*\right] + \cdots. \tag{28}$$

One then checks that the Hessian matrix $\frac{\partial^2 E^2(K_p^*)}{\partial K_i \partial K_j}$ is positive definite as well as its condition number to ensure that the data is sufficiently rich to identify the parameters.

V. Experimental Results

A three phase, 230 V, 0.5 Hp, 1735 rpm ($n_p = 2$ pole-pair) induction machine was used for the experiments. A 4096 pulse/rev optical encoder was attached to the motor for position measurements. The motor was connected to a three-phase 60 Hz source through a switch. When the switch was closed, 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 with a lowpass digital Butterworth filter followed by reconstruction of the derivatives using $dx(t)/dt = (x(t) - x(t-T))/T$ where T is the sampling interval. The voltages and currents were put through a $3 - 2$ transformation

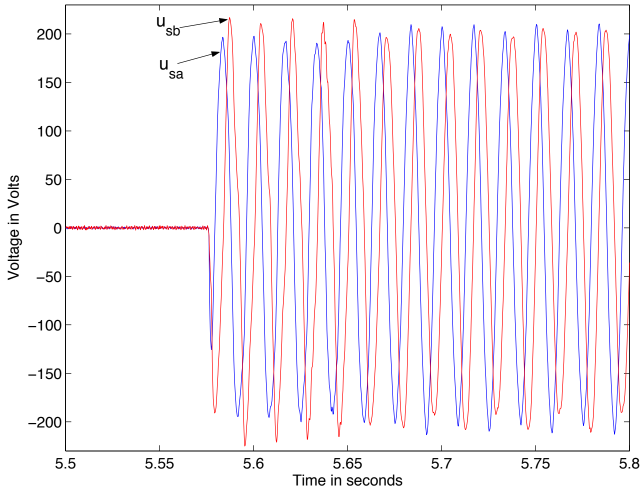

Fig. 1. Sampled two phase equivalent voltages u_{Sa} and u_{Sb}.

to obtain the two phase equivalent voltages u_{Sa}, u_{Sb} which are plotted in Fig. 1.

The sampled two phase equivalent current i_{Sa} and its simulated response i_{Sa_sim} are shown in Fig. 2 (The simulated current will be discussed below). The phase b current i_{Sb} is similar, but shifted by $\pi/(2n_p)$.

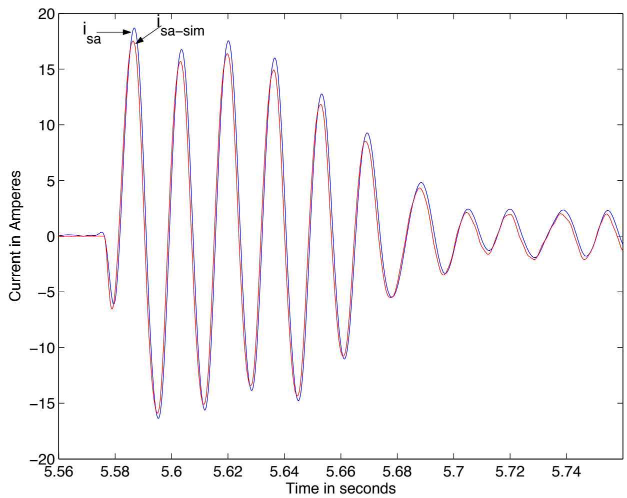

Fig. 2. Phase a current i_{Sa} and its simulated response i_{Sa_sim}.

The calculated speed ω (from the position measurements) and the simulated speed ω_{sim} are shown in Fig. 3 (the simulated speed ω_{sim} will be discussed below). Using the data $\{u_{Sa}, u_{Sb}, i_{Sa}, i_{Sb}, \theta\}$ collected between $5.57\,\text{sec}$ to $5.8\,\text{sec}$, the quantities $u_{Sx}, u_{Sy}, du_{Sx}/dt,\ du_{Sy}/dt, i_{Sx}, i_{Sy}$ $di_{Sx}/dt, di_{Sy}/dt,\ d^2i_{Sx}/dt^2, d^2i_{Sy}/dt^2,\ \omega = d\theta/dt, d\omega/dt$ were calculated and the regressor matrices R_W, R_y and R_{Wy} were computed. The procedure explained in Section IV was then carried out to compute K_1, K_2. In this case, there were three sets of extrema points that had *positive* values for all the K_i. The extremum value for K_1, K_2 that resulted in the minimum least-squares error was

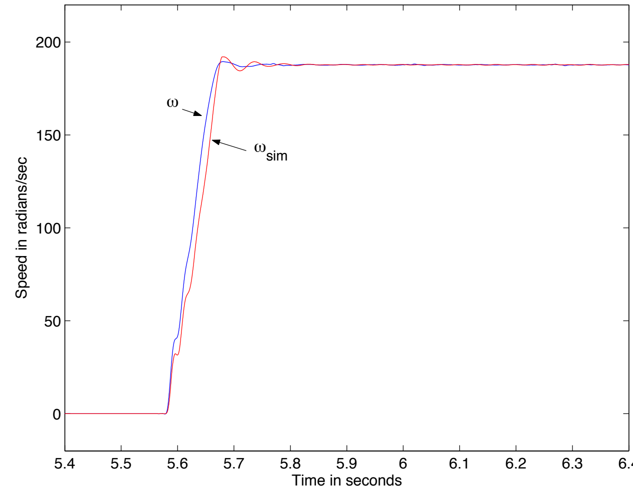

Fig. 3. Calculated speed ω and simulated speed ω_{sim}.

$$\begin{aligned} K1 &= 241.1024 \\ K2 &= 7.5988 \end{aligned}$$

Using (14), the motors electrical parameters are then

$$T_R = 0.1316 \text{ sec} \tag{29}$$

$$R_S = 5.0923\ \Omega \tag{30}$$

By way of comparison, the stator resistance was measured using an Ohmmeter giving the value of 4.9 Ohms. The Hessian matrix was calculated at the minimum point according to (28) resulting in

$$\left\{ \frac{\partial^2 E^2(K_p^*)}{\partial K_i \partial K_j} \right\} = \begin{bmatrix} 1.9123 & 0.00412 \\ 0.00412 & 570.0418 \end{bmatrix}$$

which is positive definite and has a condition number of 2.98×10^2.

A. Simulation of the Experimental Motor

Another useful way to evaluate the identified parameters (29) and (30) is to simulate the motor using these values and the measured voltages as input. The model (1) is now in terms of the parameters that can be estimated. The experimental voltages shown in Fig. 1 were then used as input to a simulation of the model (1) using the parameter values from (29) and (30). The resulting phase a current i_{Sa_sim} from the simulation is shown in Fig. 2 and corresponds well with the actual measured current i_{Sa}. Similarly, the resulting speed ω_{sim} from the simulation is shown in Fig. 3 where it is seen that the simulated speed is somewhat more oscillatory than the measured speed ω.

VI. Online Implementation

An online version of the rotor time constant estimator was simulated. Two different approaches to implementing the algorithm online are discussed.

The first approach consists of storing all the coefficients of the final resultant polynomial (27) in memory. These coefficients are functions of the entries of the data matrices $Ry \in \mathbb{R}$, $R_{Wy} \in \mathbb{R}^{8\times 1}$ and $R_W \in \mathbb{R}^{8\times 8}$ and they are stored (symbolically) as functions of these entries. In the online implementation, the data is collected, the matrices Ry, R_{Wy} and R_W are computed and the resulting numerical values of these entries are substituted into the expressions for coefficients of (27). In this way, the resultant polynomial is not be computed online. The computation of the roots of the resultant polynomial was written using the C language code, embedded in a S-function model. The measured variables (voltages, currents, and position) were sampled at $4\ kHz$ in the simulation. After collecting the data for one second, the S-function evaluated the resultant polynomial, computed its roots and then completed the estimation algorithm to obtain T_R, which was updated every second.

The results of an online simulation is shown in Figure 4. In that simulation, the rotor time constant T_R in the motor model was changed abruptly from $T_R = 0.067$ sec to $T_R = 0.078$ sec at $t = 5$ seconds; the estimation algorithm was then able to update the value of T_R one second later.

A second approach to online estimation is considered in order to circumvent the problem of the symbolic computation of the Sylvester matrices to compute the resultant polynomial which is then stored in memory. As the degrees of the polynomials to be solved increase, the dimension of the corresponding Sylvester matrices increase, and therefore the *symbolic* computation of their determinants becomes more intensive. The recent work of [17][18] is promising for the efficient symbolic computation of the determinants of large Sylvester matrices. The idea of this algorithm is based on polynomial methods in control and the discrete Fourier transform. To summarize, recall that the problem is to *symbolically* compute the determinant of the Sylvester matrix (25) to obtain the resultant polynomial (26). Another way to look at this problem is to write (26) as

$$r(K_1) = \sum_{i=0}^{N} p_i K_1^i \tag{31}$$

where the unknowns p_i and N are to be found. Any upper bound of the actual degree of $r(K_1)$ can be used for N. Such an upper bound is easily computed by finding the minimum of the sum of either the row or the column degrees of the Sylvester matrix [19]. Let $K_{1k} = e^{-j\frac{2\pi k}{N+1}}$ for $n = 0, 1, ..., N$ be $N + 1$ different values of K_1. Then the Discrete Fourier Transform (DFT) of the set of numbers $[p_0, p_1, ..., p_N]$ is

$$\begin{aligned} y_k &= \sum_{i=0}^{N} p_i e^{-j\frac{2\pi k}{N+1}i} = \sum_{i=0}^{N} p_i \left(e^{-j\frac{2\pi}{N+1}}\right)^k \\ p_i &\triangleq \frac{1}{N+1}\sum_{k=0}^{N} y_k e^{j\frac{2\pi i}{N+1}k}. \end{aligned}$$

Here y_k is just (31) evaluated at $K_{1k} = e^{-j\frac{2\pi k}{N+1}}$. That is, one computes the *numerical* determinant of (25) at the $N+1$ points K_{1k} (this is fast) and obtains the DFT of the coefficients of (31). Then the p_i are computed using the inverse DFT. That is, the symbolic calculation of the determinant is reduced to a finite number of fast *numerical* calculations. Such an approach has been shown to be as much as 500 times faster than existing methods [17].

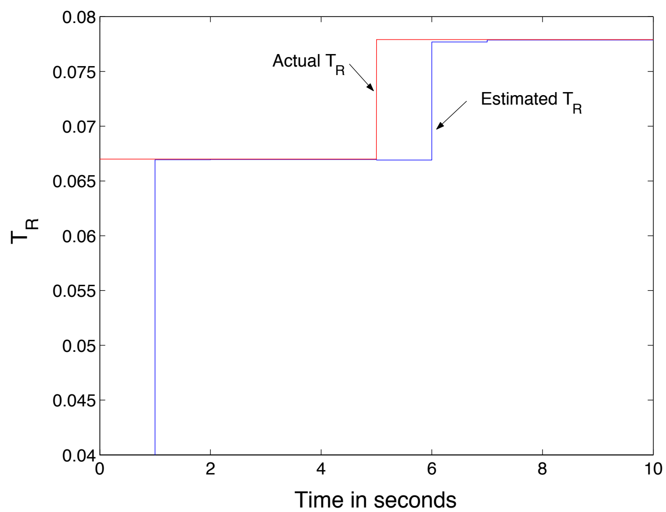

Fig. 4. Actual T_R versus estimated T_R.

VII. Conclusions

In this paper, a method for estimating the rotor time constant and stator resistance of an induction machine was presented. The parameter model was formulated as a nonlinear least-squares problem and then solved using elimination theory. Experimental results showed a close correlation with simulations based on the identified parameters. An important advantage of the procedure is that it can be used *online*, i.e., during regular operation of the machine, its parameter values can be continuously updated assuming sufficient excitation of the machine. Two approaches to online implementation of the algorithm were presented.

References

[1] J. Stephan, M. Bodson, and J. Chiasson, "Real-time estimation of induction motor parameters," *IEEE Transactions on Industry Applications*, vol. 30, pp. 746–759, May/June 1994.

[2] J. Stephan, "Real-time estimation of the parameters and fluxes of induction motors," Master's thesis, Carnegie Mellon University, 1992.

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

[4] Joachim von zur Gathen and Jürgen Gerhard, *Modern Computer Algebra*. Cambridge University Press, Cambridge, UK, 1999.

[5] K. Wang, J. Chiasson, M. Bodson, and L. M. Tolbert, "Tracking the rotor time constant of an induction motor traction drive for HEVs," in *Proceedings of the IEEE Workshop on Power Electronics in Transportation (WPET)*, pp. 83–88, October 2004.

[6] M. Vélez-Reyes, K. Minami, and G. Verghese, "Recursive speed and parameter estimation for induction machines," in *Proceedings of the IEEE Industry Applications Conference*, pp. 607–611, 1989. San Diego, California.

[7] M. Vélez-Reyes, W. L. Fung, and J. E. Ramos-Torres, "Developing robust algorithms for speed and parameter estimation in induction machines," in *Proceedings of the IEEE Conference on Decision and Control*, pp. 2223–2228, 2001. Orlando, Florida.

[8] M. Vélez-Reyes and G. Verghese, "Decomposed algorithms for speed and parameter estimation in induction machines," in *Proceedings of the IFAC Nonlinear Control Systems Design Symposium*, pp. 156–161, 1992. Bordeaux, France.

[9] H. A. Toliyat, E. Levi, and M. Raina, “A review of RFO induction motor parameter estimation techniques,” *IEEE Transactions on Energy Conversion*, vol. 18, pp. 271–283, June 2003.
[10] M. Vélez-Reyes, M. Mijalković, A. M. Stanković, S. Hiti, and J. Nagashima, “Output selection for tuning of field-oriented controllers: Steady-state analysis,” in *Conference Record of Industry Applications Society*, pp. 2012–2016, October 2003. Salt Lake City, UT.
[11] P. Vas, *Parameter estimation, condition monitoring, and diagnosis of electrical machines*. Oxford: Clarendon Press, 1993.
[12] R. Marino, S. Peresada, and P. Valigi, “Adaptive input-output linearizing control of induction motors,” *IEEE Transactions on Automatic Control*, vol. 38, pp. 208–221, February 1993.
[13] M. Bodson, J. Chiasson, and R. Novotnak, “High performance induction motor control via input-output linearization,” *IEEE Control Systems Magazine*, vol. 14, pp. 25–33, August 1994.
[14] L. Ljung, *System Identification: Theory for the User*. Prentice-Hall, 1986.
[15] T. Söderström and P. Stoica, *System Identification*. Prentice-Hall International, 1989.
[16] S. Sastry and M. Bodson, *Adaptive Control: Stability, Convergence, and Robustness*. Prentice-Hall, Englewood Cliffs, NJ, 1989.
[17] M. Hromcik and M. Šebek, “New algorithm for polynomial matrix determinant based on FFT,” in *Proceedings of the European Conference on Control ECC’99*, August 1999. Karlsruhe Germany.
[18] M. Hromcik and M. Šebek, “Numerical and symbolic computation of polynomial matrix determinant,” in *Proceedings of the 1999 Conference on Decision and Control*, pp. 1887–1888, 1999. Tampa FL.
[19] T. Kailath, *Linear Systems*. Prentice-Hall, Englewood Cliffs, NJ, 1980.