1-1-2009

Online Identification of the Rotor Time Constant of an Induction Machine

Ahmed Oteafy
Boise State University

John Chiasson
Boise State University

Marc Bodson
University of Utah

2009 American Control Conference
Hyatt Regency Riverfront, St. Louis, MO, USA
June 10-12, 2009

Online Identification of the Rotor Time Constant of an Induction Machine

Ahmed Oteafy, *Graduate Student Member, IEEE*, John Chiasson, *Senior Member, IEEE*, and Marc Bodson, *Fellow, IEEE.*

***Abstract*— Indirect field oriented control of an induction machine requires knowledge of the rotor time constant to estimate the rotor flux linkages. An online method is described 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 formulates the problem using a nonlinear least-squares criterion and is guaranteed to find the minimizing solution (parameter values) in a finite number of steps. In this work the algorithm has been implemented online in simulation with the results demonstrating its application and efficacy.**

***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 rotor flux linkages, which are not usually measured [1][2]. In order to circumvent this problem, the rotor flux linkages are estimated using an observer, and this observer requires the value of the rotor time constant T_R. However, the rotor time constant varies due to ohmic heating. The work presented here implements in simulation an online method which allows the value of the rotor time constant to be updated during normal operation of the machine.

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 $T_R = L_R/R_R$ as it changes. A model based approach is considered here, which uses measurements of the stator currents, stator voltages, and rotor speed to find the parameter values that best fit this data set to the model in a least-squares sense. The method is implemented online, and simulation results of the tracking of T_R are presented.

In [3][4], the authors developed a regressor model that was *nonlinear* in the unknown parameters and used elimination theory (resultants) to solve for the parameter values that minimized the squared error. That methodology, though computationally intensive, made no simplifying assumptions on the system model with the only restriction being that the system was sufficiently excited, which is true as long as there is a load on the motor or it is accelerating. Here we consider the special case in which the machine is running at constant speed during the data collection, which is not a restrictive requirement in most industrial variable speed drives. This constant speed assumption allows the implementation of a method to estimate the rotor time constant and stator resistance that is less computationally intensive than the method proposed in [4][5].

The method is a simplification of the *nonlinear* least-squares method presented in [4][5] where it is shown that the computational complexity of this algorithm is significantly reduced. In particular, the (reduced complexity) method presented here comes down to solving a 5-th order polynomial rather than a 20-th order polynomial as in [4][5]. The estimation method is used in combination with an input-output linearization controller ([2][6][7]).

A combined parameter identification and velocity estimation problem is discussed in [8][9] where the speed is assumed to be slowly varying. In [10][11] a linear least-squares approach was used for parameter estimation and solved by assuming a slowly varying speed. For a summary of the various techniques for tracking the rotor time constant, the reader is referred to the recent survey [12].

The paper is organized as follows. Section II introduces a standard induction motor model. Section III develops the linear regressor model assuming the motor is at constant speed. Section IV summarizes an algorithmic implementation of the method, with simulation results presented in Section V. Finally, concluding remarks and future work are discussed in Section VI.

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.

A. Oteafy is with the ECE Department, Boise State University, Boise ID 83725, ahmedoteafy@u.boisestate.edu.

J. Chiasson is with the ECE Department, Boise State University, Boise ID 83725, johnchiasson@boisestate.edu.

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

978-1-4244-4524-0/09/$25.00 ©2009 AACC

[6][7][2])

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

where $\omega = d\theta/dt$ with θ 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

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

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])

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

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

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

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

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

III. Regressor 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, 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. Consequently, a new set of independent equations is found by differentiating equations (3) and (4) to obtain

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

and

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

Next, equations (3), (4), (5), (6) are solved for ψ_{Rx}, ψ_{Ry}, $d\psi_{Rx}/dt, d\psi_{Ry}/dt$. These in turn are substituted into equations (8) and (9) with the assumption of constant speed to obtain

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

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

As $\gamma = R_S/\left(\sigma L_S\right) + \beta M/T_R$, it follows that $-\beta M/T_R^2 + \gamma/T_R = \left(R_S/T_R\right)/\left(\sigma L_S\right)$ and $\gamma + 1/T_R = R_S/\left(\sigma L_S\right) +$

$(\beta M+1)/T_R$ so that (10) and (11) become

$$0 = -\frac{d^2 i_{Sx}}{dt^2} + \frac{di_{Sy}}{dt}n_p\omega + \frac{1}{\sigma L_S}\frac{du_{Sx}}{dt} - \Big(R_S/(\sigma L_S) + (\beta M+1)/T_R\Big)\frac{di_{Sx}}{dt} - i_{Sx}(\frac{R_S}{T_R}\frac{1}{\sigma L_S}) + i_{Sy}n_p\omega((\beta M+1)/T_R) + \frac{u_{Sx}}{\sigma L_S T_R} \tag{12}$$

$$0 = -\frac{d^2 i_{Sy}}{dt^2} - \frac{di_{Sx}}{dt}n_p\omega + \frac{1}{\sigma L_S}\frac{du_{Sy}}{dt} - \Big(R_S/(\sigma L_S) + (\beta M+1)/T_R\Big)\frac{di_{Sy}}{dt} - i_{Sy}(\frac{R_S}{T_R}\frac{1}{\sigma L_S}) - i_{Sx}n_p\omega(\beta M+1)/T_R + \frac{u_{Sy}}{\sigma L_S T_R}. \tag{13}$$

A. Regressor Model

With

$$y(t) \triangleq \begin{bmatrix} \frac{d^2 i_{Sx}}{dt^2} - \frac{di_{Sy}}{dt}n_p\omega - \frac{1}{\sigma L_S}\frac{du_{Sx}}{dt} \\ \frac{d^2 i_{Sy}}{dt^2} + \frac{di_{Sx}}{dt}n_p\omega - \frac{1}{\sigma L_S}\frac{du_{Sy}}{dt} \end{bmatrix} \tag{14}$$

$$W(t) \triangleq \begin{bmatrix} -\frac{di_{Sx}}{dt}\frac{1}{\sigma L_S} & (\beta M+1)\left(-\frac{di_{Sx}}{dt} + i_{Sy}n_p\omega\right) + \frac{u_{Sx}}{\sigma L_S} & -\frac{i_{Sx}}{\sigma L_S} \\ -\frac{di_{Sy}}{dt}\frac{1}{\sigma L_S} & (\beta M+1)\left(-\frac{di_{Sy}}{dt} - i_{Sx}n_p\omega\right) + \frac{u_{Sy}}{\sigma L_S} & -\frac{i_{Sy}}{\sigma L_S} \end{bmatrix} \tag{15}$$

and

$$K = \begin{bmatrix} K_1 \\ K_2 \\ K_3 \end{bmatrix} \triangleq \begin{bmatrix} R_S \\ 1/T_R \\ R_S/T_R \end{bmatrix}, \tag{16}$$

equations (12) and (13) are written in regressor form as

$$y(t) = W(t)K. \tag{17}$$

This model is over-parameterized in that the parameters are not independent of each other as they must satisfy the constraint

$$K_3 = K_1 K_2. \tag{18}$$

Replacing K_3 by K_1K_2 in (17) results in the model not being over-parameterized, but it is then nonlinear in the parameters. This is considered next.

B. Nonlinear Least-Squares Identification [13]

Equation (17) can be rewritten as

$$y(nT) = W(nT)K \tag{19}$$

where T is the sample period, nT is the n-th sample time at which a measurement is taken, and $K = \begin{bmatrix} K_1 & K_2 & K_3 \end{bmatrix}^T$ is the vector of unknown parameters. If the constraint (18) is ignored, then the system is an over-parameterized 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 (19) 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

$$E^2(K) = \sum_{n=1}^{N}\Big|y(nT) - W(nT)K\Big|^2 \tag{20}$$

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

$$K^* = \left[\sum_{n=1}^{N} W^T(nT)W(nT)\right]^{-1}\left[\sum_{n=1}^{N} W^T(nT)y(nT)\right]. \tag{21}$$

When the system model is over-parameterized (as in the application here), the expression (21) 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

$$E^2(K) = \sum_{n=1}^{N}\Big|y(nT) - W(nT)K\Big|^2 = R_y - 2R_{Wy}^T K + K^T R_W K \tag{22}$$

subject to the constraint (18), where

$$R_y \triangleq \sum_{n=1}^{N} y^T(nT)y(nT), \tag{23}$$

$$R_{Wy} \triangleq \sum_{n=1}^{N} W^T(nT)y(nT), \tag{24}$$

$$R_W \triangleq \sum_{n=1}^{N} W^T(nT)W(nT). \tag{25}$$

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

$$0 < K_1 < \infty, 0 < K_2 < \infty. \tag{26}$$

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

$$E_p^2(K_1,K_2) \triangleq \sum_{n=1}^{N}\Big|y(nT) - W(nT)K\Big|^2_{K_3=K_1K_2} = R_y - 2R_{Wy}^T K\Big|_{K_3=K_1K_2} + \left(K^T R_W K\right)\Big|_{K_3=K_1K_2}. \tag{27}$$

As the minimum of (27) must occur in the interior of the region, it is therefore at an extremum point. This then entails solving the two extrema equations

$$p_1(K_1, K_2) \triangleq \frac{\partial E_p^2(K_1, K_2)}{\partial K_1} \quad (28)$$

$$p_2(K_1, K_2) \triangleq \frac{\partial E_p^2(K_1, K_2)}{\partial K_2} \quad (29)$$

which are *polynomials* in the parameters K_1, K_2. The degrees of the polynomials p_i are given in the table below

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

These two polynomials can be rewritten in the form

$$p_1(K_1, K_2) = a_1(K_2)K_1 + a_0(K_2) \quad (30)$$

$$p_2(K_1, K_2) = b_2(K_2)K_1^2 + b_1(K_2)K_1 + b_0(K_2). \quad (31)$$

A systematic procedure to find all possible solutions to a set of polynomials is provided by elimination theory through the method of resultants [14]. (This method was used in [3].) 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

$$\begin{aligned} r(K_2) = {} & a_0^2(K_2)b_2(K_2) - a_0(K_2)a_1(K_2)b_1(K_2) \\ & + a_1^2(K_2)b_0(K_2) \end{aligned} \quad (32)$$

where $\deg_{K_2}\{r\} = 5$. The roots of (32) 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 $a_1(K_2), a_0(K_2), b_2(K_2), b_1(K_2), b_0(K_2)$ (whose explicit expressions in terms of the elements of the matrices R_W and R_{Wy} are known a priori vis-a-vis (27), (28), and (29)) are computed and stored during data collection. The coefficients of the polynomial $r(K_2)$ are then computed online according to (32) 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 (finite number) of the possible 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.

IV. Implementation

The aforementioned method is now implemented in this work by designing the necessary blocks in synthesizable m-code on the Simulink® platform. Hence, in addition to the core algorithm of the regressor, several other blocks are required to mitigate numerical issues arising from the implementation of the algorithm, and these will be briefly discussed in what follows. Figure 1 depicts the procedure.

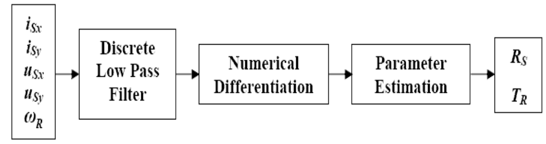

Fig. 1. Main process

A. Signal Filtering and Differentiation

The regressor model of (14) and (15) requires first and second order derivatives of the signals. This is achieved by numerical differentiation, which is preceded by a discrete filter with an appropriately selected cutoff frequency to reduce noise caused by the differentiation process. Moreover, the signals $\theta, i_{Sx}, i_{Sy}, u_{Sx}$, and u_{Sy} are passed through the same filter so that they all have the same delay due to the filter.

The numerical differentiation is implemented using the centered difference approach for the first and second order derivatives, and is given by

$$\frac{dx(nT)}{dt} \approx \frac{x((n+1)T) - x((n-1)T)}{2T} \quad (33)$$

$$\frac{d^2x(nT)}{dt^2} \approx \frac{x((n+1)T) - 2x(nT) + x((n-1)T)}{T^2} \quad (34)$$

Note that the centered difference approach has a smaller error bound than the forward or backward difference approaches yet with the same computational cost (see [15] pp. 168–176).

B. Parameter Estimation Process

The parameter estimation process utilizes the filtered, and differentiated signals to construct, at every time step, $Y(nT)$ of (14) and $W(nT)$ of (15). Then, for the next N iterations an accumulation process constructs R_y, R_{Wy}, and R_W, of (23). At the conclusion of every accumulation cycle of N iterations the values of R_y, R_{Wy}, and R_W are presented to the subsequent step before resetting them and restarting the accumulation process. Following the accumulation process, the coefficients of the 5^{th} order polynomial $r(K_2)$ of (32) are computed directly from R_y, R_{Wy}, and R_W, and a root finding algorithm is employed to determine its real roots. The root finding algorithm is summarized as follows

1) Scale the coefficients of the polynomial $r(K_2)$ by scaling its roots by Cauchy's bound.
2) Apply Weyl's quadtree root-finding approach (see [16] and [17]).
3) Numerically determine the real and complex roots, and remove repeated roots.
4) Rescale the real roots.
5) For every real root for K_2 compute the corresponding value K_1. Out of the finite possible pairs (K_1, K_2), choose the one that minimizes the error function $E_p^2(K_1, K_2)$ of (27).

Hence, the root finding algorithm produces the parameters $R_S = K_1$ and $T_R = 1/K_2$ that minimize the error function $E_p^2(K_1, K_2)$.

The combination of root normalization and the Quadtree approach result in a consistent and stable root finding algorithm even when the coefficients of $r(K_2)$ are badly scaled.

V. Simulation Results

The estimation method is applied to a two-phase equivalent model of the induction machine that is under closed-loop control. An input-output linearizing controller is applied to the induction machine. The simulation carried out in this section highlights the gain in performance that is achieved by applying the online estimation process to update the rotor time constant to the controller.

The parameters of the induction machine under study are (see [2]): $M = 0.0117$ H, $L_R = 0.014$ H, $L_S = 0.014$ H, $R_S = 1.7\ \Omega$, $R_R = 3.9\ \Omega$, $\tau_{L0} = 0.15$ N-m, $f = 0.00014$ N-m/rad/sec, $J = 0.00011$ Kg-m^2, and $n_P = 3$. The controller sets the desired rotor speed at $\omega_R = 2\pi \times 75$ rad/sec, while the load torque is defined to be $\tau_L \triangleq \tau_{L0} + f\omega$ to incorporate the viscous friction $f = 0.00014$ N-m/rad/sec and an output load torque $\tau_{L0} = 0.15$ N-m. The data is collected at a sampling rate of $f_S = 4$ kHz. The filter was a 2^{nd} order low pass Butterworth filter with a cutoff frequency of 70 Hz.

In the simulation the resistor values are increased by 50% after 3 seconds of operation with the estimator updating the value of T_R every 0.5 seconds. Figure 2 is the rotor speed and its reference profile that was run during the collection of the data (voltages, currents, and position). Figure 3 shows the plot of $K_1 = R_S$ and its reference versus time. After the update at 3.5 secs the estimator gives the correct value of K_1 within 0.03%. Similarly, Figure 4 is a plot of $K_2 = 1/T_R$ and its reference versus time showing that after the update the estimator gives the correct value of K_2 within 2%. Figure 5 is a plot of the real power $P(t) = u_{Sa}i_{Sa} + u_{Sb}i_{Sb}$ vs time. As the figure shows, the real power jumps up to 66.9 W at 3 sec. After the rotor time constant is updated at 3.5 seconds, the real power comes down to 63.7 W, which is a 5% decrease. Figure

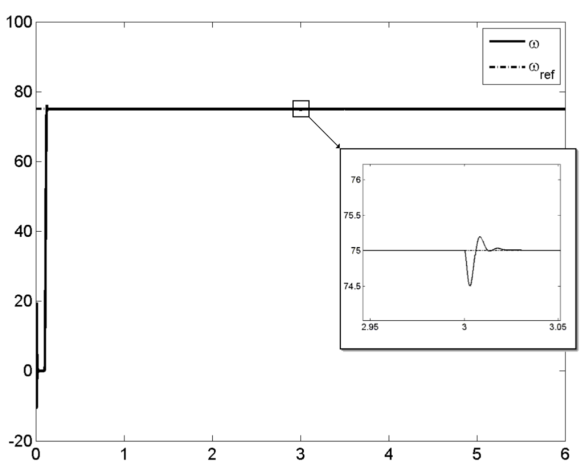

Fig. 2. ω and ω_{ref} vs. time in sec

6 is a plot of the reactive power $Q(t) = u_{Sa}i_{Sb} - u_{Sa}i_{Sb}$

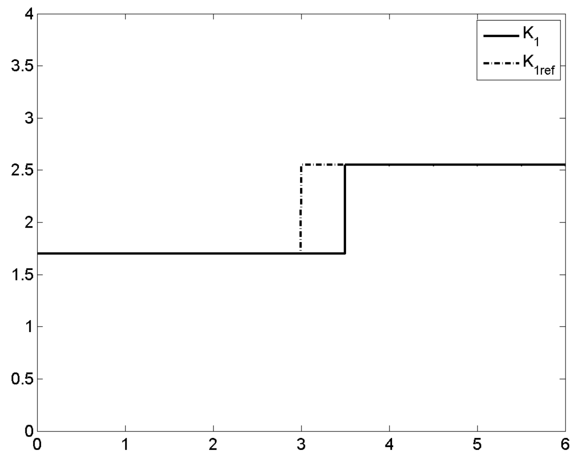

Fig. 3. $K_1 = R_S$ and K_{1ref} vs. time in sec

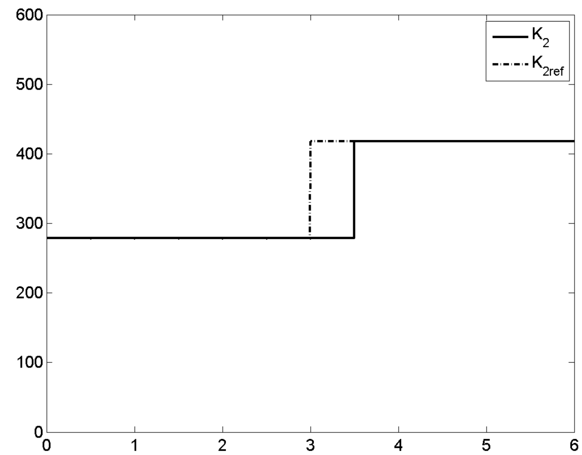

Fig. 4. $K_2 = 1/T_R$ and K_{2ref} vs. time in sec

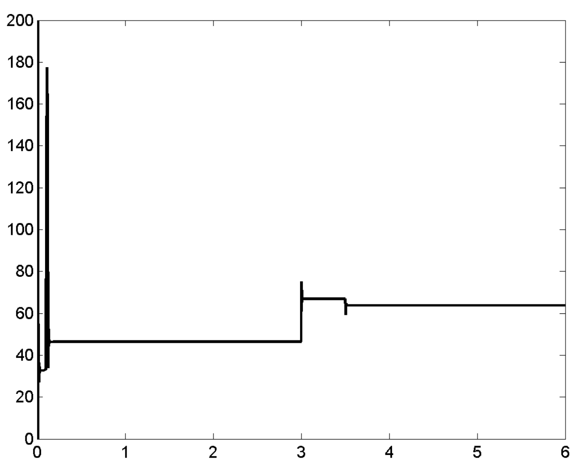

Fig. 5. Real power P in W vs. time in sec.

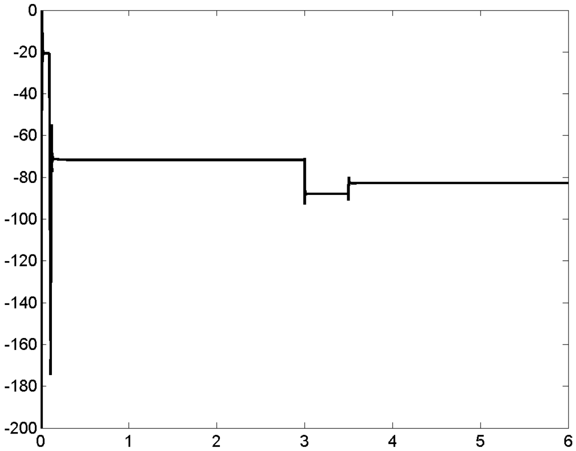

Fig. 6. Reactive power Q in W vs. time in sec

vs time. The figure shows that the reactive power jumps in magnitude to 88.0 W at $t = 3$ sec. After the rotor time constant is updated at 3.5 seconds, its magnitude comes down to 82.8 W, which is a 6% decrease. As mentioned in the introduction, the rotor time constant $T_R = 1/K_2$ is used to estimate the rotor fluxes which in turn are used to to estimate the direct and quadrature currents for use in field oriented control. In field oriented control the motor torque is given by $\tau = \mu\psi_d i_q$ ($\mu = \frac{M n_p}{J L_R}$) which at constant speed is given by $\tau = \mu M i_d i_q$ [2]. For a given torque, the current magnitude $i_d^2 + i_q^2$ is minimized if $i_d = i_q$. Consequently, if an incorrect value of the time constant is used, then non optimum values of the currents will be commanded by the controller resulting in increased Ohmic losses in the stator and rotor resistors. This then results in a higher power usage for the same torque requirement. Thus it is important to know the correct value of T_R as it varies for energy efficient operation of the motor.

VI. Conclusions and Further Work

An approach for the online identification of the values of the rotor time constant and the stator resistance that was originally developed by the authors in [18] has been simulated online in this work. A practical benefit of updating the rotor time constant is the power savings as shown above. Work is now proceeding on implementing the above algorithm in an experimental setting and results are expected by the time of the conference.

References

[1] Werner Leonhard, *Control of Electrical Drives*, 3rd Edition, Springer-Verlag, Berlin, 2001.

[2] John Chiasson, *Modeling and High-Performance Control of Electric Machines*, John Wiley & Sons, 2005.

[3] Kaiyu Wang, John Chiasson, Marc Bodson, and Leon Tolbert, "A nonlinear least-squares approach for estimation of the induction motor parameters", *IEEE Transactions on Automatic Control*, vol. 50, no. 10, pp. 1622–1628, October 2005.

[4] Kaiyu Wang, John Chiasson, Marc Bodson, and Leon M. Tolbert, "An on-line rotor time constant estimator for the induction machine", in *Proceedings of the IEEE International Electric Machines and Drives Conference*, May 2005, pp. 608–614, San Antonio TX.

[5] Kaiyu Wang, John Chiasson, Marc Bodson, and Leon Tolbert, "An online rotor time constant estimator for the induction machine", *IEEE Transactions on Control Systems Technology*, vol. 15, no. 2, pp. 339–348, March 2007.

[6] R. Marino, S. Peresada, and P. Valigi, "Adaptive input-output linearizing control of induction motors", *IEEE Transactions on Automatic Control*, vol. 38, no. 2, pp. 208–221, February 1993.

[7] M. Bodson, J. Chiasson, and R. Novotnak, "High performance induction motor control via input-output linearization", *IEEE Control Systems Magazine*, vol. 14, no. 4, pp. 25–33, August 1994.

[8] Miguel 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*, 2001, pp. 2223–2228, Orlando, Florida.

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

[10] J. Stephan, M. Bodson, and J. Chiasson, "Real-time estimation of induction motor parameters", *IEEE Transactions on Industry Applications*, vol. 30, no. 3, pp. 746–759, May/June 1994.

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

[12] Hamid A. Toliyat, Emil Levi, and Mona Raina, "A review of RFO induction motor parameter estimation techniques", *IEEE Transactions on Energy Conversion*, vol. 18, no. 2, pp. 271–283, June 2003.

[13] Shankar Sastry and Marc Bodson, *Adaptive Control: Stability, Convergence, and Robustness*, Prentice-Hall, Englewood Cliffs, NJ, 1989.

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

[15] R. L. Burden and J. D. Faires, *Numerical Analysis*, 8th Edition, Thomson Brooks/Cole, 2005.

[16] J.-C Yakoubsohn, "Numerical analysis of a bisection-exclusion method to find zeros of univariate analytic functions", *Journal of Complexity*, vol. 21, no. 5, pp. 652–690, October 2005.

[17] H. Weyl, "Randbemerkungen zu hauptproblemen der mathematik", *Mathematische Zeitschrift*, vol. 20, no. 1, pp. 131–150, December 1924.

[18] John Chiasson and Marc Bodson, "Estimation of the rotor time constant of an induction machine at constant speed", in *Proceedings of the European Control Conference ECC'07*, July 2-5 2007, pp. 4673–4678, Kos Greece.