

3-1-2004

Symbolically Aided Model Development for an induction Machine in Virtual Test Bed

Wenzhong Gao

Mississippi State University, wgao@cavs.msstate.edu

Eugene V. Slovodnik

University of South Carolina - Columbia, solodovn@engr.sc.edu

Roger A. Dougal

University of South Carolina - Columbia, dougal@engr.sc.edu

Follow this and additional works at: https://scholarcommons.sc.edu/elct_facpub



Part of the [Electrical and Computer Engineering Commons](#)

Publication Info

Published in *IEEE Transactions on Energy Conversion*, Volume 19, 2004, pages 125-135.

<http://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=60>

© 2004 by IEEE

This Article is brought to you by the Electrical Engineering, Department of at Scholar Commons. It has been accepted for inclusion in Faculty Publications by an authorized administrator of Scholar Commons. For more information, please contact digres@mailbox.sc.edu.

Symbolically Aided Model Development for an Induction Machine in Virtual Test Bed

Wenzhong Gao, *Senior Member, IEEE*, Eugene V. Solodovnik, and Roger A. Dougal, *Senior Member, IEEE*

Abstract—A new phase-domain induction machine model for use in power system dynamic simulation is developed with the aid of a symbolic tool. The symbolic tool can automatically construct a time-domain power system component model in the resistive companion form (RCF) that is widely used in time-domain simulators. The automatic differentiation technique (ADT) is utilized within the context of a symbolic modeling language, and the tool has been implemented for the virtual test bed (VTB) simulation environment. The new induction machine model was used to study start-up transients of an induction motor. The new model was verified by comparing the simulation results to those obtained from a standard d - q model. This paper also demonstrates that a symbolically assisted technique can provide an efficient and rapid path for developing complex nonlinear device models for power system simulations.

Index Terms—Automatic differentiation, dynamic response, electric machines, induction machines, power system modeling, power system simulation, power systems, resistive companion form, symbolic tool, symbolically assisted simulation.

I. INTRODUCTION

INDUCTION machines are ubiquitous components of electric power systems. To perform dynamic simulation of a power system network, it is desirable to have a detailed and, at the same time, computationally efficient induction machine model suitable for dynamic studies of complex power systems. However, it is difficult to develop such a model because induction machine dynamics are highly nonlinear due to the heavily nonlinear mutual inductances between the stator and rotor windings. To simplify the model development, researchers have been trying to model induction machines in the d - q domain with the aid of reference frame transformation. But the d - q modeling technique introduces undesirable effects in dynamic simulation of a power system. Specifically, these effects include instability of interfacing the d - q model to the rest of the power system network, difficulties in modeling harmonics and unbalanced operation, and inefficiency in simulating multi-machine network since the d - q and inverse d - q transformations have to be performed at each time step. Therefore, there is a considerable research interest in phase-domain modeling for electric machines [7], [8], [12], [15], [16]. Although several phase-domain models

based on the Trapezoidal integration and Thevenin equivalent circuit modeling technique have been proposed, none of them can facilitate a system-wide simultaneous simulation solution.

This paper focuses on the development of a phase-domain resistive companion form (RCF) model of induction machine. The resistive companion network analysis technique is best described in [5], [6], and [11]. One of the merits of the RCF modeling technique is that natural coupling between neighboring components, which is based on the enforcement of physical conservation laws, can be used in a power system network. RCF technique also allows for easy object-oriented computer programming for a power system network simulator. Once a component model has been developed and tested, it can be used in any study case. For example, each component model can be implemented as a dynamically linked library (DLL) object.

The equations that describe the full nonlinear dynamics of an induction machine in the phase-domain are quite complex. Developing the simulation model by manual manipulation of the equations, performing numerical integration, and obtaining the Jacobian matrix as required for the RCF model is tedious, error-prone at best, and quite nearly unmanageable. Hence, it is desirable to have a high-level model development tool that will automatically generate the simulation model from “raw” equations that can be simply entered. This is described as a symbolically assisted modeling and simulation method [1], [2], [10].

In this paper, we describe how a new symbolic modeling tool was utilized to develop a complex and detailed phase-domain induction machine model. The tool rapidly calculates the sensitivity matrices by an automatic differentiation technique (ADT) [3], [9]. The ADT was originally developed in the late 1980s. It automatically generates exact derivative values at low computational cost. It has been shown that the computational cost ratio of evaluating the derivatives to the cost of evaluating the function is independent of the number of variables. Therefore, it can provide a very efficient way for developing a complex nonlinear device model for power system simulation. The symbolic tool is implemented in virtual test bed (VTB), which is an interactive simulation environment that allows analysis of mixed discipline systems and provides advanced visualization capabilities. The VTB architecture is described in detail in [4].

The paper is organized as follows. In Section II, the dynamic model for a general induction machine is derived in phase coordinates. In Section III, a brief description of the RCF modeling methodology is presented. In Section IV, the ADT is applied to the automatic creation of RCF models. In Section V, the phase-domain induction machine model is obtained with the aid of the symbolic tool. In Section VI, a start-up transient is simulated and the model is verified by comparison to results obtained from a standard d - q reference frame model; the model is

Manuscript received September 3, 2002. This work was supported by the ONR under Grant N00014-00-1-0368.

W. Gao was with the Electrical Engineering Department, University of South Carolina, Columbia, 29208 USA. He is now with the Center for Advanced Vehicular Systems, Mississippi State University, Mississippi State, MS 39762 USA (e-mail: wgao@cavs.msstate.edu).

E. Solodovnik and R. A. Dougal are with the Electrical Engineering Department, University of South Carolina, Columbia, 29208 USA (e-mail: gaowenz@engr.sc.edu; solodovnik@engr.sc.edu; dougal@engr.sc.edu).

Digital Object Identifier 10.1109/TEC.2003.822316

further used to simulate unbalanced operation of induction motors in a multi-machine network. Finally, conclusions are given in Section VII.

II. PHASE-DOMAIN MODEL DERIVATION

The dynamic equations of the electrical and mechanical subsystems of an induction machine are derived in this section. The induction machine is modeled as a set of mutually coupled inductors, which interact among themselves to generate the electromagnetic torque. Straightforward circuit analysis leads to the derivation of an appropriate mathematical model.

The following assumptions are made: 1) the machine is cylindrical; 2) space magnetomotive force (mmf) and flux waves are sinusoidally distributed (neglecting the teeth and slots effects); 3) saturation, hysteresis, and eddy currents are neglected. Fig. 1 illustrates the stator and rotor windings of an induction machine: three-phase stator windings and three-phase rotor windings. The stator is Y connected, the rotor may be squirrel cage or shorted wound rotor. Note that all inductors are mounted on the same magnetic circuit, and thus, they are all coupled. The rotor position $\theta_r(t)$ is defined as the electrical angle between the stator phase *a* magnetic axis (a stationary reference) and the rotor phase *a* magnetic axis. The mechanical rotor position angle is

$$\theta_m(t) = \frac{2}{p}\theta_r(t) \quad (1)$$

where p is the number of poles of the rotating magnetic field in the air gap.

Application of Kirchhoff's Voltage Law, Kirchhoff's Current Law, and Faraday's Law to the circuit of Fig. 1 yields

$$v_{as}(t) - v_{ns}(t) = r_{as}i_{as}(t) + \frac{d}{dt}\lambda_{as}(t) \quad (2)$$

$$v_{bs}(t) - v_{ns}(t) = r_{bs}i_{bs}(t) + \frac{d}{dt}\lambda_{bs}(t) \quad (3)$$

$$v_{cs}(t) - v_{ns}(t) = r_{cs}i_{cs}(t) + \frac{d}{dt}\lambda_{cs}(t) \quad (4)$$

$$i_{ns}(t) = -(i_{as}(t) + i_{bs}(t) + i_{cs}(t)) \quad (5)$$

$$0 = r_{ar}i_{ar}(t) + \frac{d}{dt}\lambda_{ar}(t) \quad (6)$$

$$0 = r_{br}i_{br}(t) + \frac{d}{dt}\lambda_{br}(t) \quad (7)$$

$$0 = r_{cr}i_{cr}(t) + \frac{d}{dt}\lambda_{cr}(t) \quad (8)$$

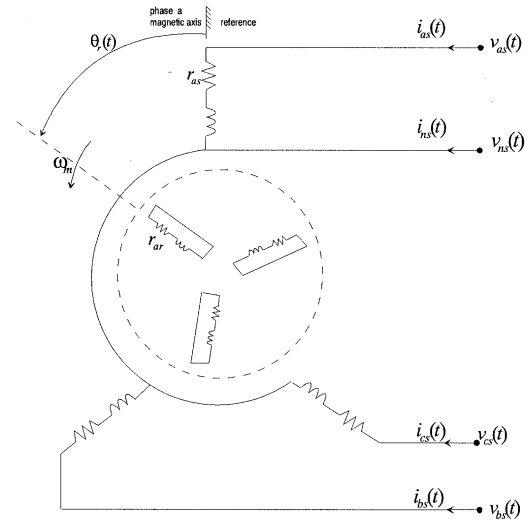


Fig. 1. General induction machine as a set of mutually coupled windings.

The magnetic flux linkages are complex functions of the rotor position and the electric currents flowing in the various windings of the machine. The magnetic flux linkages of the stator and rotor phase windings are

$$\begin{bmatrix} \lambda_{abcs}(t) \\ \lambda_{abcr}(t) \end{bmatrix} = \begin{bmatrix} L_{ss}(t) & L_{sr}(t) \\ L_{rs}(t) & L_{rr}(t) \end{bmatrix} \begin{bmatrix} i_{abcs}(t) \\ i_{abcr}(t) \end{bmatrix} \quad (9)$$

where (see expressions at the bottom of the page.)

The notation in the above equations is obvious. For example, $(L_{las} + L_{ms})$ is the self-inductance for phase *a* stator winding, and L_{ms} is the stator magnetizing inductance; $(L_{lar} + L_{mr})$ is the self-inductance for the phase *a* rotor winding, and L_{mr} is the rotor magnetizing inductance. L_{sr} is the maximum mutual inductance between a stator phase winding and a rotor phase winding. Notice that the mutual inductances in $L_{sr}(t)$ are dependent on the time-varying rotor position. This makes the overall system highly nonlinear and time-varying.

The electromagnetic (EM) torque can be computed by differentiating the field energy function $w_{fld}(t)$ with respect to the rotor mechanical position θ_m , $T_{em}(t) = (P_{em}(t)/\omega_m(t)) = (dw_{fld}(t)/d\theta_m)$. In particular, we have

$$T_{em}(t) = (i_{abcs}(t))^T \left(\frac{\partial L_{sr}(t)}{\partial \theta_m} \right) i_{abcr}(t).$$

$$\begin{aligned} L_{ss} &= \begin{bmatrix} L_{las} + L_{ms} & -\frac{1}{2}L_{ms} & -\frac{1}{2}L_{ms} \\ -\frac{1}{2}L_{ms} & L_{lbs} + L_{ms} & -\frac{1}{2}L_{ms} \\ -\frac{1}{2}L_{ms} & -\frac{1}{2}L_{ms} & L_{lcs} + L_{ms} \end{bmatrix} \\ L_{rr} &= \begin{bmatrix} L_{lar} + L_{mr} & -\frac{1}{2}L_{mr} & -\frac{1}{2}L_{mr} \\ -\frac{1}{2}L_{mr} & L_{lbr} + L_{mr} & -\frac{1}{2}L_{mr} \\ -\frac{1}{2}L_{mr} & -\frac{1}{2}L_{mr} & L_{lcr} + L_{mr} \end{bmatrix} \\ L_{sr}(t) &= L_{sr} \begin{bmatrix} \cos(\theta_r(t)) & \cos(\theta_r(t) + 2\pi/3) & \cos(\theta_r(t) - 2\pi/3) \\ \cos(\theta_r(t) - 2\pi/3) & \cos(\theta_r(t)) & \cos(\theta_r(t) + 2\pi/3) \\ \cos(\theta_r(t) + 2\pi/3) & \cos(\theta_r(t) - 2\pi/3) & \cos(\theta_r(t)) \end{bmatrix} \\ L_{rs}(t) &= L_{sr}^T(t) \end{aligned}$$

The remainder of the rotor equations are

$$J \frac{d\omega_m(t)}{dt} = (i_{abcs}(t))^T \left(\frac{\partial L_{sr}(t)}{\partial \theta_m} \right) i_{abcr}(t) - T_m(t) \quad (10)$$

$$\frac{d\theta_m(t)}{dt} = \omega_m(t) \quad (11)$$

where J is the moment of inertia of the shaft, $T_m(t)$ is the mechanical load torque, and $\omega_m(t)$ is the mechanical speed.

Note that if $R_s = \text{diag}(r_s \ r_s \ r_s)$, $R_r = \text{diag}(r_r \ r_r \ r_r)$, $L_{las} = L_{lbs} = L_{lcs} = L_{ls}$, $L_{lar} = L_{lbr} = L_{lcr} = L_{lr}$, then stator and rotor windings are balanced.

III. RESISTIVE COMPANION FORM MODELING METHODOLOGY

Consider a power system component with a number of terminals through which it can be interconnected to other components as it is illustrated in Fig. 2. Each terminal has an associated across and a through variable. If the terminal is electrical, these variables are the terminal voltage with respect to a common reference and the electrical current flowing into the terminal, respectively.

Such a component can be described with a set of algebraic-integral-differential equations of the following general form [14]:

$$\begin{bmatrix} \dot{\mathbf{i}} \\ \mathbf{0} \end{bmatrix} = \begin{bmatrix} \mathbf{f}_1(\dot{\mathbf{v}}, \dot{\mathbf{y}}, \dots, \int \mathbf{v}, \int \mathbf{y}, \dots, \mathbf{v}, \mathbf{y}, \mathbf{u}, t) \\ \mathbf{f}_2(\dot{\mathbf{v}}, \dot{\mathbf{y}}, \dots, \int \mathbf{v}, \int \mathbf{y}, \dots, \mathbf{v}, \mathbf{y}, \mathbf{u}, t) \end{bmatrix} \quad (12)$$

where $\mathbf{f}_1, \mathbf{f}_2$ are arbitrary vector functions; \mathbf{i} is a vector of through variables, \mathbf{v} is a vector of across variables, \mathbf{y} is a vector of internal state variables, and \mathbf{u} is a vector of independent controls. Note that the functions \mathbf{f}_1 and \mathbf{f}_2 define two sets of equations named *external* and *internal equations*. The through variables appear only in the external equations. Similarly, the device states are classified as *external states* $\mathbf{v}(t)$ (i.e., the across variables) and *internal states* $\mathbf{y}(t)$. Equation set (12) is consistent in the sense that the total number of states is equal to the total number of equations.

In order to create an RCF model of this component, (12) is integrated over the simulation time step using Trapezoidal rule or other numerical integration methods. For a nonlinear device, the resulting algebraic equations have to be solved by Newton's method iteratively at each time step. Hence, the RCF model is of the following form at each time step:

$$\begin{bmatrix} \dot{\mathbf{i}}(t) \\ \mathbf{0} \end{bmatrix} = \mathbf{G}(\mathbf{v}(t), \mathbf{v}(t-h), \mathbf{y}(t), \mathbf{y}(t-h), t) \begin{bmatrix} \mathbf{v}(t) \\ \mathbf{y}(t) \end{bmatrix} - \begin{bmatrix} \mathbf{b}_1(\mathbf{v}(t), \mathbf{v}(t-h), \mathbf{i}(t-h), \mathbf{y}(t), \mathbf{y}(t-h), t) \\ \mathbf{b}_2(\mathbf{v}(t), \mathbf{v}(t-h), \mathbf{i}(t-h), \mathbf{y}(t), \mathbf{y}(t-h), t) \end{bmatrix} \quad (13)$$

where \mathbf{G} is the Jacobian matrix of the discretized version of (12); $\mathbf{b}_1, \mathbf{b}_2$ are vectors depending on the through variables, across variables, and internal states of the component at current and previous time step. In case of linear device, vectors $\mathbf{b}_1, \mathbf{b}_2$ depend only on past history values of through, across, and internal states.

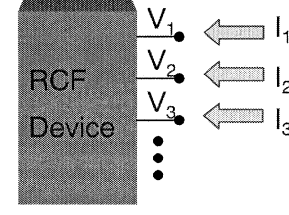


Fig. 2. Resistive companion form device terminal definition.

After all of the components in the power system are modeled as RCF models, the network solution can be obtained by application of energy conservation laws at each node of the system. This procedure results in the following set of equations:

$$\sum_k A^k i^k(t) = I_{inj} \quad (14)$$

where I_{inj} is a vector of nodal current injections, A^k is a component incidence matrix with: (Please see the expression at the bottom of the page.) and $i^k(t)$ are the terminal currents of component k . The component k terminal across variables $\mathbf{v}^k(t)$ is related to the nodal vector of across variables $\mathbf{v}(t)$ by

$$\mathbf{v}^k(t) = (A^k)^T \mathbf{v}(t). \quad (15)$$

Upon substitution of device equations (13), (14) becomes a set of nonlinear algebraic equations. At each time step, these equations are solved using Newton's method. The solution is given by iterative evaluation of the following expression:

$$\begin{bmatrix} \mathbf{v}(t) \\ \mathbf{y}(t) \end{bmatrix} = \begin{bmatrix} \mathbf{v}^0(t) \\ \mathbf{y}^0(t) \end{bmatrix} - \begin{bmatrix} \mathbf{G}_{11} & \mathbf{G}_{12} \\ \mathbf{G}_{21} & \mathbf{G}_{22} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{m}_1^0 \\ \mathbf{m}_2^0 \end{bmatrix} \quad (16)$$

where $\mathbf{v}^0(t), \mathbf{y}^0(t)$ are the values of the state variables at the previous iteration; $\mathbf{m}_1^0, \mathbf{m}_2^0$ represents the mismatch of the system equations of the previous iteration; $\mathbf{G}_{11}, \mathbf{G}_{12}, \mathbf{G}_{21}, \mathbf{G}_{22}$ are block matrices of the system Jacobian matrix.

The next section describes the application of the automatic differentiation technique to RCF model generation.

IV. AUTOMATIC GENERATION OF RCF MODEL

The major effort in the development of the RCF model equations (13) is the computation of the Jacobian matrix. This process involves the computations of many partial derivatives of the component model equations. The automatic differentiation technique can be used for this purpose to avoid manual computation. This technique is based on the fact that exact derivatives of a function can be obtained by applying the chain rule, a well-known method of differential calculus. The advantages of this approach are

- the process can be readily automated using a high-level computer language;
- the results are accurate, since they are identical to those obtained from analytic evaluation of the derivatives;

$$A_{ij}^k = \begin{cases} 1, & \text{if terminal } j \text{ of component } k \text{ is connected to node } i \\ 0, & \text{otherwise} \end{cases}$$

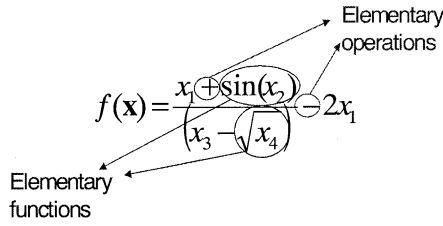


Fig. 3. Example function composition.

TABLE I
EXAMPLE FUNCTION-INTERMEDIATE VARIABLES, ELEMENTARY
FUNCTIONS, AND THEIR DERIVATIVES

Intermediate variables	Elementary functions	Derivatives of elementary functions
$x_5 = \sin(x_2)$	$x_5 = g_5(x_2)$	$g_{5,2} = \cos(x_2)$
$x_6 = \sqrt{x_4}$	$x_6 = g_6(x_4)$	$g_{6,4}(x_4) = 1/(2\sqrt{x_4})$
$x_7 = 2x_1$	$x_7 = g_7(x_1)$	$g_{7,1}(x_1) = 2$
$x_8 = x_1 + x_5$	$x_8 = g_8(x_1, x_5)$	$g_{8,1} = 1, g_{8,5} = 1$
$x_9 = x_3 - x_6$	$x_9 = g_9(x_3, x_6)$	$g_{9,3} = 1, g_{9,6} = -1$
$x_{10} = x_8 / x_9$	$x_{10} = g_{10}(x_8, x_9)$	$g_{10,8}(x_9) = 1/x_9$ $g_{10,9}(x_8, x_9) = -x_8 x_9^{-2}$
$x_{11} = x_{10} - x_7$	$x_{11} = g_{11}(x_{10}, x_7)$	$g_{11,10} = 1, g_{11,7} = -1$
$f(x) = x_{11}$		

- the run-time computational effort required is relatively small, and equivalent to manually derived analytical models.

ADT relies on the fact that most functions, no matter how complicated, can be considered as a composition of several elementary functions and elementary operations. For example, the expression shown in Fig. 3 consists of elementary functions of *sin* and *square root*, and the elementary operations of addition, subtraction, multiplication, and division. Such functions can be differentiated using the chain rule.

The automatic differentiation algorithm consists of two phases: forward and reverse. The forward phase consists of the evaluation of the elementary function derivatives. The results are stored as intermediate variables. Consider, for example, the function in Fig. 3. For this function, seven intermediate variables and associated elementary functions are defined as shown in Table I.

The third column of Table I contains the derivatives of the elementary functions, which are denoted as $g_{i,j}(x_{P_i})$, that is

$$g_{i,j}(x_{P_i}) = \frac{\partial g_i}{\partial x_j}, \quad i \in [n+1, N], \quad x_j \in P_i.$$

A flowchart of the forward phase algorithm is presented in Fig. 4. As soon as all of the intermediate variables are constructed and all of the derivatives of the elementary functions are obtained, the reverse phase of the algorithm follows.

The construction of the intermediate variables can be visualized using a graph as it is illustrated in Fig. 5. Each node of the graph represents a variable (independent or intermediate). Each

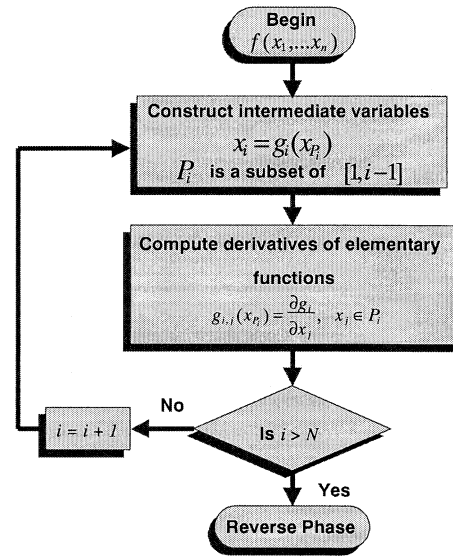


Fig. 4. ADT forward phase algorithm.

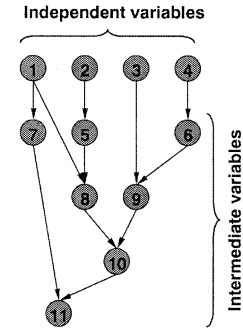


Fig. 5. Tree of intermediate variables.

line segment of the graph runs from the node x_i to the node x_j ($i < j$), if elementary function g_i depends on the variable x_j .

It has been shown [9] that the derivative of the function $f(x)$ with respect to the variable x_k can be obtained through adding all of the products of $g_{i,j}$ between nodes x_k and x_N . The reverse phase algorithm for this example is demonstrated in Table II. The reverse algorithm flowchart is illustrated in Fig. 6.

It can be seen that the algorithm introduces a number of redundant operations such as multiplication by 1, summation with zero, etc. These operations are searched and eliminated; thus improving the algorithm run-time efficiency.

The implementation of RCF model generator consists of three basic functions: discretization, initialization, and time-stepping function. Discretization uses a suitable numerical integration rule to convert the differential algebraic model equations into purely algebraic equations. The discretized equations also depend on the time step. The initialization function is called once at the beginning of the simulation. The time-stepping function is called repeatedly during the simulation. The ADT algorithm's computationally intensive parts, including all searches and optimizations, are completed during the initialization process. The time-stepping part includes only the evaluation of the derivative results, thus ensuring optimal computational efficiency. The flowchart of the ADT-based RCF model is illustrated in Fig. 7.

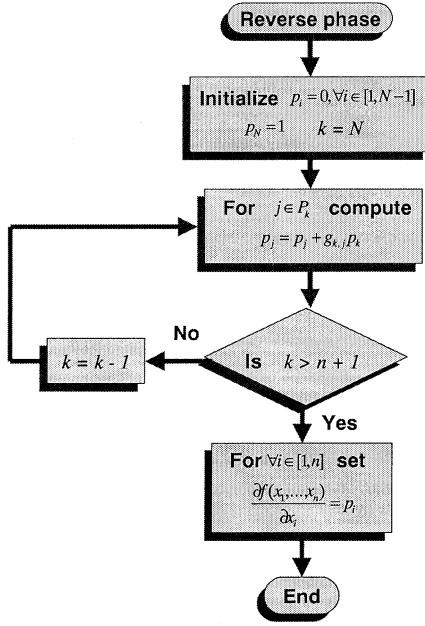


Fig. 6. ADT reverse phase algorithm.

TABLE II
EXAMPLE FUNCTION REVERSE PHASE ALGORITHM RESULTS

$p_{11} = 1$	$p_{11} = 1$
$p_{10} = p_{10} + g_{11,10} p_{11}$	$p_{10} = 1$
$p_7 = p_7 + g_{11,7} p_{11}$	$p_7 = -1$
$p_9 = p_9 + g_{10,9} p_{10}$	$p_9 = -x_8 x_9^{-2}$
$p_8 = p_8 + g_{10,8} p_{10}$	$p_8 = 1/x_9$
$p_6 = p_6 + g_{9,6} p_9$	$p_6 = x_8 x_9^{-2}$
$p_3 = p_3 + g_{9,3} p_9$	$p_3 = -x_8 x_9^{-2}$
$p_5 = p_5 + g_{8,5} p_8$	$p_5 = 1/x_9$
$p_1 = p_1 + g_{8,1} p_8$	$p_1 = 1/x_9$
$p_1 = p_1 + g_{7,1} p_7$	$p_1 = 1/x_9 - 2$
$p_4 = p_4 + g_{6,4} p_6$	$p_4 = 1/(2\sqrt{x_4}) x_8 x_9^{-2}$
$p_2 = p_2 + g_{5,2} p_5$	$p_2 = \cos(x_2)/x_9$

V. IMPLEMENTATION OF THE MODEL USING SYMBOLIC LANGUAGE

The induction machine model equations are manipulated and rearranged according to the format of (12) so as to facilitate RCF model generation. The rearranged phase-domain model for the induction machine is given in the Appendix. Note that the model is highly nonlinear and has 23 equations corresponding to 23 nodes. The Jacobian matrix for this model has $23^2 = 529$ elements. Though some of the elements are zero, there are still a large number of elements that have to be evaluated. Apparently, manual calculation of the Jacobian matrix would result in long development time of the model and even longer debugging time, since errors inevitably would be present if the RCF model is manually developed by a human being. Such complexity of the phase-domain model of the induction machine is

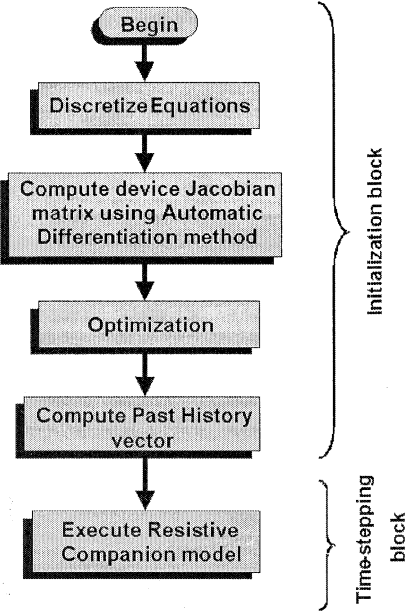


Fig. 7. Algorithm for ADT-based RCF model.

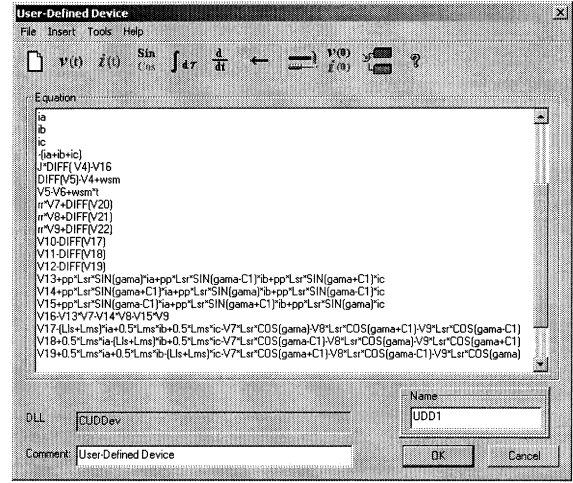


Fig. 8. Induction machine model in the symbolic tool.

the main obstacle to manual development of its RCF representation. However, the symbolically assisted model generator allows rapid development of the phase-domain RCF model of the induction machine. The automatic model generator ran just in fractions of a second and the resulting model was free of errors. Furthermore, computational efficiency of the automatically developed RCF model is comparable to that of the manually developed model.

The model equations are typed into a symbolic tool according to a high-level symbolic language, which is illustrated in the following Fig. 8. The symbolic tool automatically discretizes the model equations, computes the Jacobian, and generates the RCF model equations in the form of (13). It is implemented in the VTB software [4], which is a time-domain simulation environment that can be used for design, analysis, and virtual prototyping of electric power systems.

The developed RCF model generator can be used in two basic modes—runtime and offline. In the runtime regime, the RCF

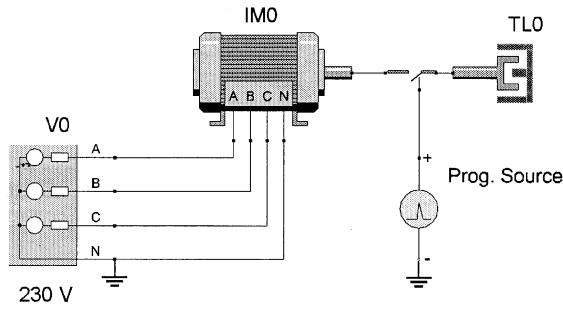
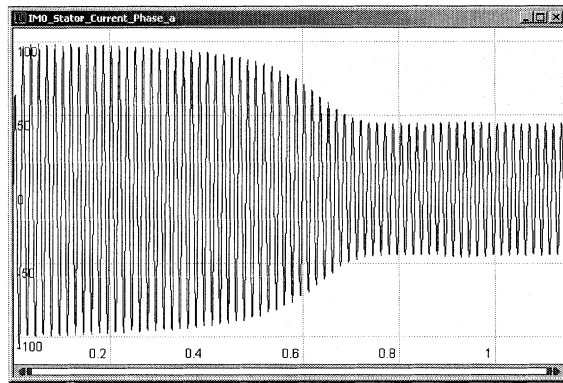
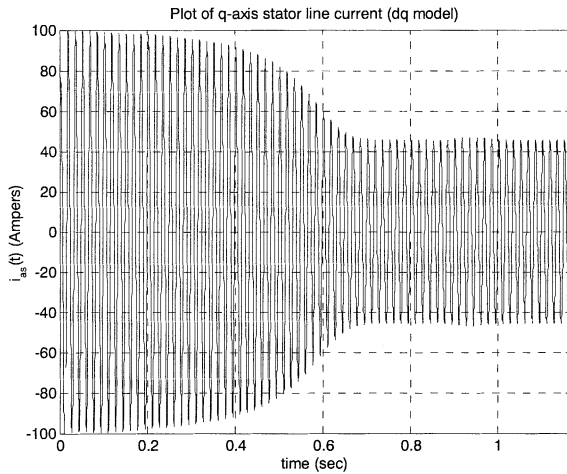


Fig. 9. Schematic diagram of the first test system in VTB.



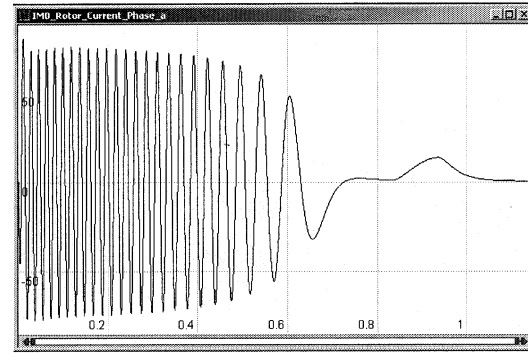
(a)



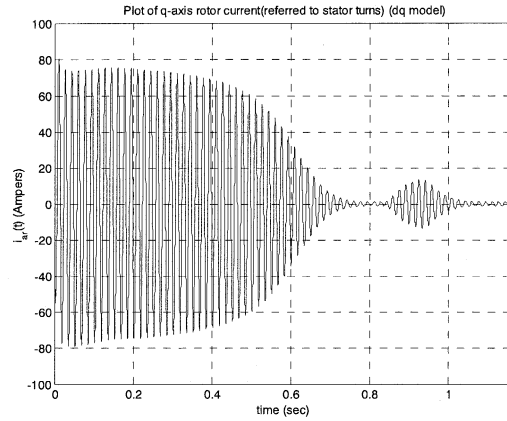
(b)

Fig. 10. Stator phase current: (a) New model. (b) $d-q$ model.

model is constructed and executed while the simulation engine runs. This mode has the advantage of rapid model development. The user is able to change the device equations and to immediately see the changes as reflected in waveforms. The graphical user interface illustrated in Fig. 8 corresponds to this mode. In the offline regime, the RCF model is automatically constructed in the form of C++ source code. The code can be compiled to create the RCF model, which is a DLL object compatible with VTB software. The offline mode is used for automatic generation of the most efficient permanent models that can be included in the VTB model library.



(a)



(b)

Fig. 11. Rotor current (referred to stator turns): (a) New model (b) $d-q$ model.

VI. APPLICATION OF THE NEW INDUCTION MACHINE MODEL

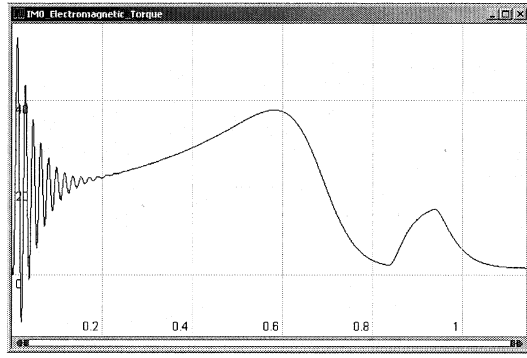
A. Simulation of Startup Transients of an Induction Machine

To verify the new model and demonstrate the power of the newly developed symbolic model development tool, the starting transients of an induction machine are simulated using the proposed model, and compared to the results obtained by using a $d-q$ reference model. The parameters of the three-phase Y-connected squirrel cage induction machine are as follows [13]:

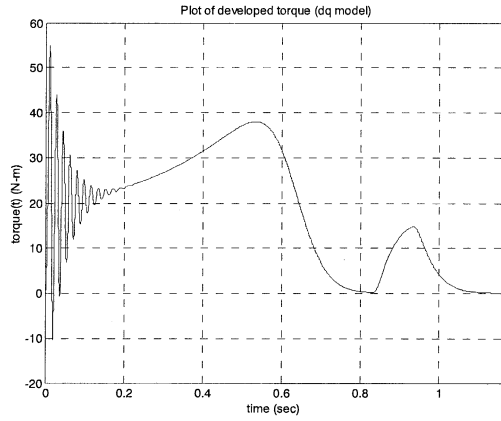
$$\begin{aligned} r_s &= 0.531 \, \Omega, \quad r_r = 0.408 \, \Omega, \quad J = 0.1 \, \text{kg} \cdot \text{m}^2, \\ L_{ls} &= L_{lr} = 2.52 \, \text{mH}, \quad L_M = 8.47 \, \text{mH}, \quad \text{poles} = 4, \\ V_{\text{rated}} &= 230 \, \text{V}, \quad P_{\text{rated}} = 3.73 \, \text{kW}. \end{aligned}$$

Note that rotor parameters are referred to the stator side. The induction motor is suddenly connected to a balanced three-phase, 60 Hz, and sinusoidal supply with rated line to line voltage of 230 V root mean square (rms). The effect of source impedance in series with the supply is not considered here. Fig. 9 illustrates this test system in VTB software environment.

The total simulation time is 1.2 s. The three-phase voltage source is suddenly applied to the motor stator terminals at time $t = 0$ s. A 0.85-p.u. load torque lasting 0.1 s is applied to the rotor shaft at time $t = 0.833$ s. This event is modeled using an ideal voltage-controlled switch and programmable voltage source in VTB.



(a)



(b)

Fig. 12. Electromagnetic torque: (a) New Model and (b) d - q model.

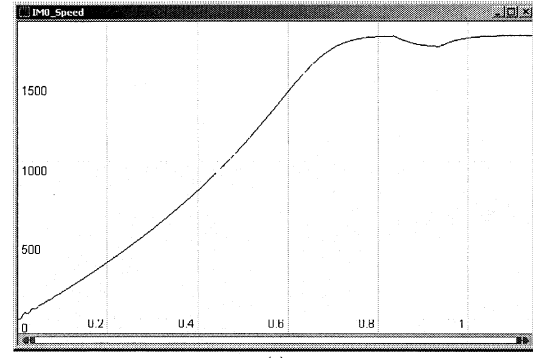
Simulation results obtained from VTB are shown in the plots together with results obtained from a standard d - q reference model within a stationary reference frame, which is implemented in MATLAB.

In particular, the simulation results of stator phase current are given in Fig. 10, rotor current in Fig. 11, electromagnetic torque in Fig. 12, and rotor speed in Fig. 13, respectively.

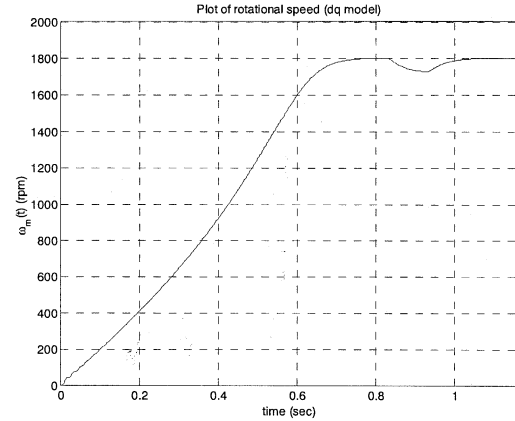
As can be seen, except for the rotor currents, the simulation results obtained from the new model in VTB and results from the standard d - q model are in good agreement. In Fig. 11, the rotor current obtained from the new model has the actual slip frequency in the machine rotor circuit, but the rotor current given by the d - q model has a frequency different from the actual frequency, which is related to the reference frame. This illustrates a clear advantage of the new abc model. However, the envelope of the rotor current waveform in Fig. 11(b) matches the waveform in Fig. 11(a), thereby proving that the new model is valid.

B. Simulation of Unbalanced Operation of Two Induction Machines

To demonstrate that the new model can be used to simulate unbalanced operation of induction machines and that the new abc model can be easily used to simulate a multi-machine power system network, another simulation is performed. The VTB schematic diagram of the test case is illustrated in Fig. 14. Two induction motors (with same parameters as in previous simulation study case) are fed by an infinite bus with 230 V (line-line) through a piece of transmission line. The line is 10 m



(a)



(b)

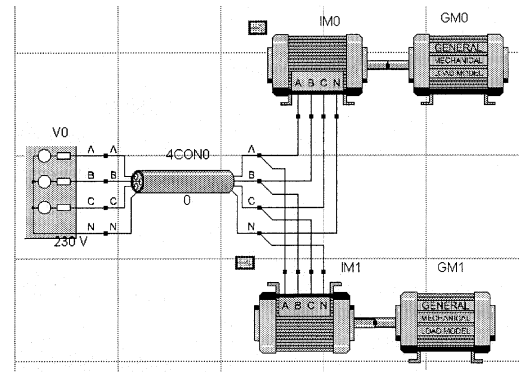
Fig. 13. Speed: (a) New Model (b) d - q model.

Fig. 14. Schematic diagram of the second test system in VTB.

long with a conductivity 5.714 mhos/m and is modeled as Pi equivalent circuits. The motors are accelerated with no load to steady state. Then, 1.0-p.u. load is applied to each motor's shaft and the system achieves a new steady state. Next, the rotor phase A winding of induction motor IM0 and the rotor phase B winding of induction motor IM1 are suddenly open-circuited (by assigning a big resistance 1 M Ω to r_{ar} of IM0 and r_{br} of IM1). The graphical user interface (GUI) of the induction machine model in VTB is illustrated in Fig. 15.

The simulation results are given in Figs. 16–20.

In particular, the simulation results of electromagnetic torque of IM0 are given in Fig. 16, rotor speed of IM0 is in Fig. 17, stator phase currents of IM0 are in Fig. 18, rotor phase currents

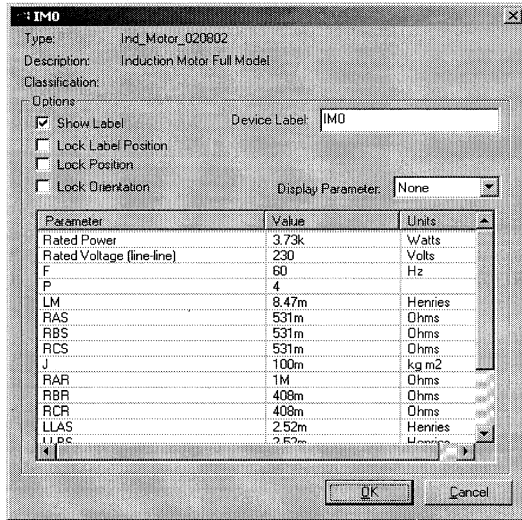


Fig. 15. GUI of the induction motor model in VTB.

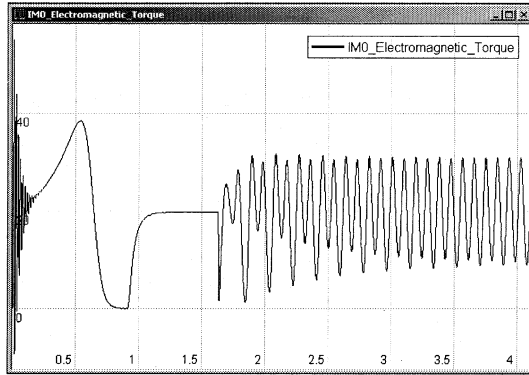


Fig. 16. Electromagnetic torque of IM0.

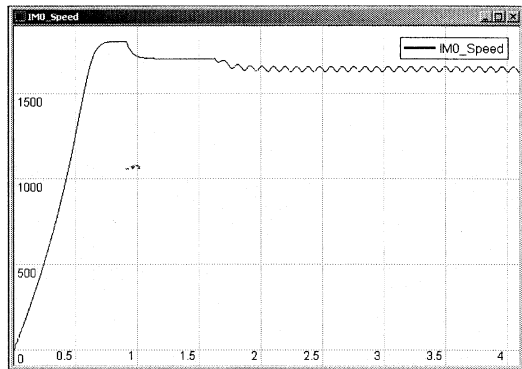


Fig. 17. Speed of IM0.

of IM0 are in Fig. 19, and rotor phase currents of IM1 are in Fig. 20, respectively.

As can be seen from the simulation results, the EM torque oscillates badly after one phase of the rotor is open-circuited. Also, the rotor speed and stator current oscillate. Rotor currents in the other phase windings are increased to accommodate the load. A prediction such as this given by the simulation is useful to gain an insightful understanding of different operation modes of induction machines.

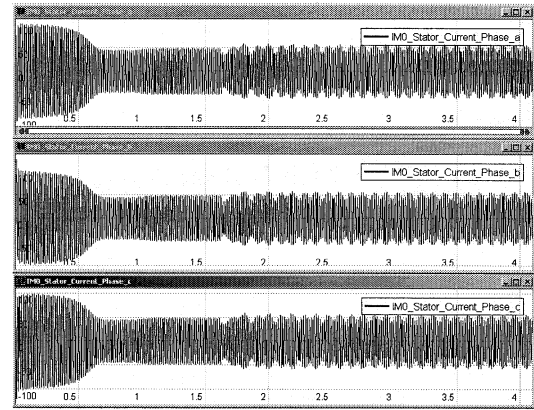


Fig. 18. Stator phase current of IM0.

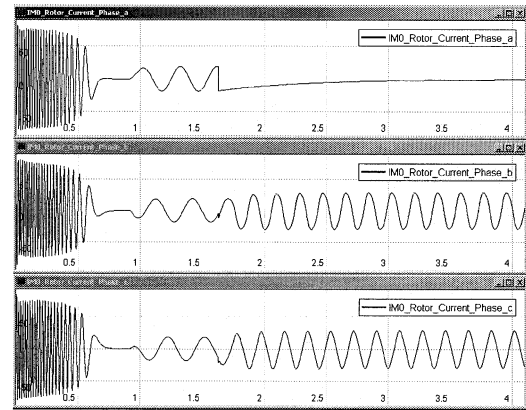


Fig. 19. Phase A, B, and C rotor currents of IM0 (referred to stator turns).

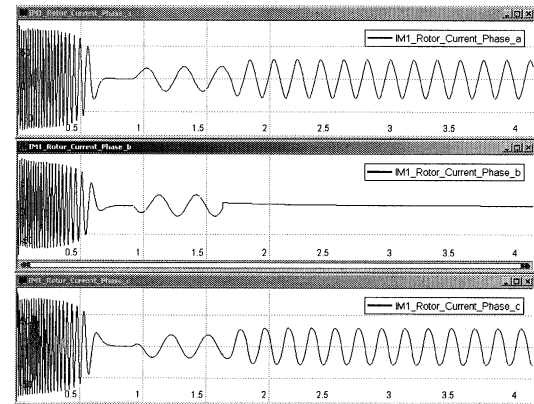


Fig. 20. Rotor current of IM1 (referred to stator turns).

VII. CONCLUSION

The time-domain model of a general induction machine was rapidly developed with the aid of a symbolic model development tool. The resulting model was validated by comparison to a standard model and was proven suitable for power system dynamic simulation. The symbolically assisted tool provides a fast and efficient way of developing complex nonlinear device models. The tool greatly reduces the effort required to develop a complex simulation model.

In a general power system simulation, unlike the conventional model in $d-q$ coordinates, the phase-domain model can be directly incorporated into the overall network (no transformations

are needed at each time step) to generate more accurate (no numerical instability or predictions for certain unknown variables are present) and more informative (simulation solutions of the unknown variables are more related to actual physical quantities) network simulation results.

Further, the new model of the induction machine is especially advantageous for simulation of multi-machine power systems since there is no need for multiple reference frames. The new model has the capability to model asymmetries and/or unbalanced operating conditions and can be extended to model non-linear saturation effects in induction machines.

APPENDIX

To facilitate the development of the RCF model, the phase-domain model equations of the induction machine are manipulated, rearranged, and cast into the following form. Note that some intermediate variables have been introduced so that each equation is simplified. This procedure also eases the automatic model generation by the symbolic tool.

$$i_{as}(t) = \frac{1}{r_{as}}v_{as}(t) - \frac{1}{r_{as}}v_{ns}(t) - \frac{1}{r_{as}}y_1(t) \quad (17)$$

$$i_{bs}(t) = \frac{1}{r_{bs}}v_{bs}(t) - \frac{1}{r_{bs}}v_{ns}(t) - \frac{1}{r_{bs}}y_2(t) \quad (18)$$

$$i_{cs}(t) = \frac{1}{r_{cs}}v_{cs}(t) - \frac{1}{r_{cs}}v_{ns}(t) - \frac{1}{r_{cs}}y_3(t) \quad (19)$$

$$i_{ns}(t) = -\frac{1}{r_{as}}v_{as}(t) + \frac{1}{r_{as}}v_{ns}(t) + \frac{1}{r_{as}}y_1(t) - \frac{1}{r_{bs}}v_{bs}(t) + \frac{1}{r_{bs}}v_{ns}(t) + \frac{1}{r_{bs}}y_2(t) - \frac{1}{r_{cs}}v_{cs}(t) + \frac{1}{r_{cs}}v_{ns}(t) + \frac{1}{r_{cs}}y_3(t) \quad (20)$$

$$T_m(t) = J \frac{d\omega_m(t)}{dt} - \gamma_T(t) \quad (21)$$

$$0 = \frac{d\varphi(t)}{dt} - \omega_m(t) + \omega_{sm} \quad (22)$$

$$0 = \varphi(t) - \theta_m(t) + \omega_{sm} \cdot t \quad (23)$$

$$0 = r_{ar}i_{ar}(t) + \frac{d}{dt}\lambda_{ar}(t) \quad (24)$$

$$0 = r_{br}i_{br}(t) + \frac{d}{dt}\lambda_{br}(t) \quad (25)$$

$$0 = r_{cr}i_{cr}(t) + \frac{d}{dt}\lambda_{cr}(t) \quad (26)$$

$$0 = y_1(t) - \frac{d}{dt}\lambda_{as}(t) \quad (27)$$

$$0 = y_2(t) - \frac{d}{dt}\lambda_{bs}(t) \quad (28)$$

$$0 = y_3(t) - \frac{d}{dt}\lambda_{cs}(t) \quad (29)$$

$$0 = y_4(t) + \frac{p}{2}L_{sr}\sin\left(\frac{p}{2}\varphi(t) + \frac{p}{2}\omega_{sm}t\right) \times \left(\frac{1}{r_{as}}v_{as}(t) - \frac{1}{r_{as}}v_{ns}(t) - \frac{1}{r_{as}}y_1(t)\right) + \frac{p}{2}L_{sr}\sin\left(\frac{p}{2}\varphi(t) + \frac{p}{2}\omega_{sm}t - \frac{2\pi}{3}\right)$$

$$\times \left(\frac{1}{r_{bs}}v_{bs}(t) - \frac{1}{r_{bs}}v_{ns}(t) - \frac{1}{r_{bs}}y_2(t)\right) + \frac{p}{2}L_{sr}\sin\left(\frac{p}{2}\varphi(t) + \frac{p}{2}\omega_{sm}t + \frac{2\pi}{3}\right) \times \left(\frac{1}{r_{cs}}v_{cs}(t) - \frac{1}{r_{cs}}v_{ns}(t) - \frac{1}{r_{cs}}y_3(t)\right) \quad (30a)$$

$$0 = y_5(t) + \frac{p}{2}L_{sr}\sin\left(\frac{p}{2}\varphi(t) + \frac{p}{2}\omega_{sm}t + \frac{2\pi}{3}\right) \times \left(\frac{1}{r_{as}}v_{as}(t) - \frac{1}{r_{as}}v_{ns}(t) - \frac{1}{r_{as}}y_1(t)\right) + \frac{p}{2}L_{sr}\sin\left(\frac{p}{2}\varphi(t) + \frac{p}{2}\omega_{sm}t\right) \times \left(\frac{1}{r_{bs}}v_{bs}(t) - \frac{1}{r_{bs}}v_{ns}(t) - \frac{1}{r_{bs}}y_2(t)\right) + \frac{p}{2}L_{sr}\sin\left(\frac{p}{2}\varphi(t) + \frac{p}{2}\omega_{sm}t - \frac{2\pi}{3}\right) \times \left(\frac{1}{r_{cs}}v_{cs}(t) - \frac{1}{r_{cs}}v_{ns}(t) - \frac{1}{r_{cs}}y_3(t)\right) \quad (30b)$$

$$0 = y_6(t) + \frac{p}{2}L_{sr}\sin\left(\frac{p}{2}\varphi(t) + \frac{p}{2}\omega_{sm}t - \frac{2\pi}{3}\right) \times \left(\frac{1}{r_{as}}v_{as}(t) - \frac{1}{r_{as}}v_{ns}(t) - \frac{1}{r_{as}}y_1(t)\right) + \frac{p}{2}L_{sr}\sin\left(\frac{p}{2}\varphi(t) + \frac{p}{2}\omega_{sm}t + \frac{2\pi}{3}\right) \times \left(\frac{1}{r_{bs}}v_{bs}(t) - \frac{1}{r_{bs}}v_{ns}(t) - \frac{1}{r_{bs}}y_2(t)\right) + \frac{p}{2}L_{sr}\sin\left(\frac{p}{2}\varphi(t) + \frac{p}{2}\omega_{sm}t\right) \times \left(\frac{1}{r_{cs}}v_{cs}(t) - \frac{1}{r_{cs}}v_{ns}(t) - \frac{1}{r_{cs}}y_3(t)\right) \quad (30c)$$

$$0 = y_7(t) - y_4(t)i_{ar}(t) - y_5(t)i_{br}(t) - y_6(t)i_{cr}(t) \quad (31)$$

$$0 = \lambda_{as}(t) - (L_{las} + L_{ms}) \times \left(\frac{1}{r_{as}}v_{as}(t) - \frac{1}{r_{as}}v_{ns}(t) - \frac{1}{r_{as}}y_1(t)\right) + 0.5L_{ms}\left(\frac{1}{r_{bs}}v_{bs}(t) - \frac{1}{r_{bs}}v_{ns}(t) - \frac{1}{r_{bs}}y_2(t)\right) + 0.5L_{ms}\left(\frac{1}{r_{cs}}v_{cs}(t) - \frac{1}{r_{cs}}v_{ns}(t) - \frac{1}{r_{cs}}y_3(t)\right) - i_{ar}(t)L_{sr}\cos\left(\frac{p}{2}\varphi(t) + \frac{p}{2}\omega_{sm}t\right) - i_{br}(t)L_{sr}\cos\left(\frac{p}{2}\varphi(t) + \frac{p}{2}\omega_{sm}t + \frac{2\pi}{3}\right) - i_{cr}(t)L_{sr}\cos\left(\frac{p}{2}\varphi(t) + \frac{p}{2}\omega_{sm}t - \frac{2\pi}{3}\right) \quad (32a)$$

$$0 = \lambda_{bs}(t) + 0.5L_{ms}\left(\frac{1}{r_{as}}v_{as}(t) - \frac{1}{r_{as}}v_{ns}(t) - \frac{1}{r_{as}}y_1(t)\right) - (L_{lbs} + L_{ms})\left(\frac{1}{r_{bs}}v_{bs}(t) - \frac{1}{r_{bs}}v_{ns}(t) - \frac{1}{r_{bs}}y_2(t)\right) + 0.5L_{ms}\left(\frac{1}{r_{cs}}v_{cs}(t) - \frac{1}{r_{cs}}v_{ns}(t) - \frac{1}{r_{cs}}y_3(t)\right) - i_{ar}(t)L_{sr}\cos\left(\frac{p}{2}\varphi(t) + \frac{p}{2}\omega_{sm}t - \frac{2\pi}{3}\right)$$

$$\begin{aligned}
& -i_{br}(t)L_{sr}\cos\left(\frac{p}{2}\varphi(t)+\frac{p}{2}\omega_{sm}t\right) \\
& -i_{cr}(t)L_{sr}\cos\left(\frac{p}{2}\varphi(t)+\frac{p}{2}\omega_{sm}t+\frac{2\pi}{3}\right) \quad (32b)
\end{aligned}$$

$$\begin{aligned}
0 &= \lambda_{cs}(t) + 0.5L_{ms}\left(\frac{1}{r_{as}}v_{as}(t) - \frac{1}{r_{as}}v_{ns}(t) - \frac{1}{r_{as}}y_1(t)\right) \\
&+ 0.5L_{ms}\left(\frac{1}{r_{bs}}v_{bs}(t) - \frac{1}{r_{bs}}v_{ns}(t) - \frac{1}{r_{bs}}y_2(t)\right) \\
&- (L_{lcs} + L_{ms})\left(\frac{1}{r_{cs}}v_{cs}(t) - \frac{1}{r_{cs}}v_{ns}(t) - \frac{1}{r_{cs}}y_3(t)\right) \\
&- i_{ar}(t)L_{sr}\cos\left(\frac{p}{2}\varphi(t) + \frac{p}{2}\omega_{sm}t + \frac{2\pi}{3}\right) \\
&- i_{br}(t)L_{sr}\cos\left(\frac{p}{2}\varphi(t) + \frac{p}{2}\omega_{sm}t - \frac{2\pi}{3}\right) \\
&- i_{cr}(t)L_{sr}\cos\left(\frac{p}{2}\varphi(t) + \frac{p}{2}\omega_{sm}t\right) \quad (32c)
\end{aligned}$$

$$\begin{aligned}
0 &= \lambda_{ar}(t) - L_{sr}\cos\left(\frac{p}{2}\varphi(t) + \frac{p}{2}\omega_{sm}t\right) \\
&\times \left(\frac{1}{r_{as}}v_{as}(t) - \frac{1}{r_{as}}v_{ns}(t) - \frac{1}{r_{as}}y_1(t)\right) \\
&- L_{sr}\cos\left(\frac{p}{2}\varphi(t) + \frac{p}{2}\omega_{sm}t - \frac{2\pi}{3}\right) \\
&\times \left(\frac{1}{r_{bs}}v_{bs}(t) - \frac{1}{r_{bs}}v_{ns}(t) - \frac{1}{r_{bs}}y_2(t)\right) \\
&- L_{sr}\cos\left(\frac{p}{2}\varphi(t) + \frac{p}{2}\omega_{sm}t + \frac{2\pi}{3}\right) \\
&\times \left(\frac{1}{r_{cs}}v_{cs}(t) - \frac{1}{r_{cs}}v_{ns}(t) - \frac{1}{r_{cs}}y_3(t)\right) \\
&- (L_{lar} + L_{mr})i_{ar}(t) + 0.5L_{mr}i_{br}(t) + 0.5L_{mr}i_{cr}(t) \quad (33a)
\end{aligned}$$

$$\begin{aligned}
0 &= \lambda_{br}(t) - L_{sr}\cos\left(\frac{p}{2}\varphi(t) + \frac{p}{2}\omega_{sm}t + \frac{2\pi}{3}\right) \\
&\times \left(\frac{1}{r_{as}}v_{as}(t) - \frac{1}{r_{as}}v_{ns}(t) - \frac{1}{r_{as}}y_1(t)\right) \\
&- L_{sr}\cos\left(\frac{p}{2}\varphi(t) + \frac{p}{2}\omega_{sm}t\right) \\
&\times \left(\frac{1}{r_{bs}}v_{bs}(t) - \frac{1}{r_{bs}}v_{ns}(t) - \frac{1}{r_{bs}}y_2(t)\right) \\
&- L_{sr}\cos\left(\frac{p}{2}\varphi(t) + \frac{p}{2}\omega_{sm}t - \frac{2\pi}{3}\right) \\
&\times \left(\frac{1}{r_{cs}}v_{cs}(t) - \frac{1}{r_{cs}}v_{ns}(t) - \frac{1}{r_{cs}}y_3(t)\right) \\
&+ 0.5L_{mr}i_{ar}(t) - (L_{lbr} + L_{mr})i_{br}(t) + 0.5L_{mr}i_{cr}(t) \quad (33b)
\end{aligned}$$

$$\begin{aligned}
0 &= \lambda_{cr}(t) - L_{sr}\cos\left(\frac{p}{2}\varphi(t) + \frac{p}{2}\omega_{sm}t - \frac{2\pi}{3}\right) \\
&\times \left(\frac{1}{r_{as}}v_{as}(t) - \frac{1}{r_{as}}v_{ns}(t) - \frac{1}{r_{as}}y_1(t)\right) \\
&- L_{sr}\cos\left(\frac{p}{2}\varphi(t) + \frac{p}{2}\omega_{sm}t + \frac{2\pi}{3}\right) \\
&\times \left(\frac{1}{r_{bs}}v_{bs}(t) - \frac{1}{r_{bs}}v_{ns}(t) - \frac{1}{r_{bs}}y_2(t)\right) \\
&- L_{sr}\cos\left(\frac{p}{2}\varphi(t) + \frac{p}{2}\omega_{sm}t\right)
\end{aligned}$$

$$\begin{aligned}
&\times \left(\frac{1}{r_{cs}}v_{cs}(t) - \frac{1}{r_{cs}}v_{ns}(t) - \frac{1}{r_{cs}}y_3(t)\right) \\
&+ 0.5L_{mr}i_{ar}(t) + 0.5L_{mr}i_{br}(t) - (L_{lcr} + L_{mr})i_{cr}(t) \quad (33c)
\end{aligned}$$

ACKNOWLEDGMENT

The authors would like to thank the reviewers and the editor for their helpful comments.

REFERENCES

- [1] F. L. Alvarado and Y. Liu, "General purpose symbolic tools for electric networks," *IEEE Trans. Power Syst.*, vol. 3, pp. 689–697, May 1988.
- [2] F. L. Alvarado, C. A. Canizares, A. Keyhani, and B. Coates, "Instructional use of declarative languages for the study of machine transients," *IEEE Trans. Power Syst.*, vol. 6, pp. 407–412, Feb. 1991.
- [3] B. M. Averick, J. J. More, C. H. Bischof, A. Carle, and A. Griewank, "Computing large sparse Jacobian matrices using automatic differentiation," in Mathematics and Computer Science Division, Argonne National Laboratory, Preprint MCS-P10-1088, Jan. 1993.
- [4] C. W. Brice, L. U. Gökder, and R. A. Dougal, "The virtual test bed: An environment for virtual prototyping," in *Proc. Int. Conf. Elect. Ship*, Istanbul, Turkey, Sept. 1998, pp. 27–31.
- [5] H. W. Dommel, "Digital computer solution of electromagnetic transients in single- and multiphase networks," *IEEE Trans. Power App. Syst.*, vol. PAS-88, pp. 388–399, Apr. 1969.
- [6] —, "Nonlinear and time-varying elements in digital simulation of electromagnetic transients," *IEEE Trans. Power App. Syst.*, vol. PAS-90, pp. 2561–2566, Nov./Dec. 1971.
- [7] W. Gao, "New methodology for power system modeling and its application in machine modeling and simulation," Ph.D. dissertation, School of Elect. Comput. Eng., Georgia Inst. Technol., 2002.
- [8] W. Gao and A. P. S. Meliopoulos, "Phase-domain model of a three-phase induction machine for dynamic simulation," in *Proc. 33rd Annu. North Amer. Power Symp.*, 2001.
- [9] M. Jerosolimski and L. Levacher, "A new method for fast calculation of Jacobian matrices: Automatic differentiation for power system simulation," *IEEE Trans. Power Syst.*, vol. 9, pp. 700–706, May 1994.
- [10] J. A. Martinez, "Educational use of EMTF MODELS for the study of rotating machine transients," *IEEE Trans. Power Syst.*, vol. 8, pp. 1392–1399, Nov. 1993.
- [11] A. P. S. Meliopoulos, *Power System Grounding and Transients: An Introduction*. New York: Marcel Dekker, 1988.
- [12] J. R. Marti and T. O. Meyers, "Phase-domain induction motor model for power system simulators," in *Proc. IEEE Western Canada Conf. Exhibition*, 1995, pp. 276–282.
- [13] D. W. Novotny and T. A. Lipo, *Vector Control and Dynamics of AC Drives*. New York: Oxford Univ. Press, 1996.
- [14] E. V. Solodovnik, A. P. S. Meliopoulos, and G. J. Cokkinides, "On stability of implicit numerical methods in nonlinear dynamical systems simulation," in *Proc. 30th IEEE Southeastern Symp. Syst. Theory*, Morgantown, WV, Mar. 8–10, 1998, pp. 27–31.
- [15] P. Subramaniam and O. P. Malik, "Digital simulation of a synchronous generator in direct phase quantities," *Proc. Inst. Elect. Eng.*, vol. 118, no. 1, pp. 153–160, Jan. 1971.
- [16] R. Takahashi, J. Tamura, Y. Tada, and A. Kurita, "Derivation of phase-domain model of an induction generator in terms of instantaneous values," in *Proc. IEEE Power Eng. Soc. Winter Meeting*, vol. 1, 2000, pp. 359–364.



Wenzhong Gao (SM'03) was born in Jiangxi, China, in 1968. He received the Bachelor degree in aeronautical propulsion control engineering from Northwestern Polytechnic University, Xi'an, China, in 1988, the Master's degree in industrial automation from Northeastern University, Shenyang, China, in 1991, and the Ph.D. degree in electrical and computer engineering from Georgia Institute of Technology, Atlanta, in 2002.

His research interests include power system modeling and simulation, electric machinery, and drive, etc.



Eugene V. Solodovnik was born in Taganrog, Russia, in 1967. He received the E.E. diploma from the Taganrog State University of Radio-Engineering, Taganrog, Russia, in 1991 and the Ph.D. degree from the University of South Carolina, Columbia, in 2001.

His research interests include control, modeling, analysis, and simulation of power systems, and computer-aided-design (CAD) software development, while his current research projects include hardware-in-the-loop system studies and distributed computer simulations.



Roger A. Dougal (SM'94) received the Ph.D. degree in electrical engineering from Texas Tech University, Lubbock, in 1983.

Currently, he is the Director of the virtual test bed project, which is developing an advanced simulation and virtual prototyping environment for multidisciplinary dynamic systems. He is also on the faculty of the Electrical Engineering Department at South Carolina University, Columbia.

Dr. Dougal received the Samuel Litman Distinguished Professor of Engineering award, and has been honored as a Carolina Research Professor.