The Pennsylvania State University

The Graduate School

Department of Electrical Engineering

NUMERICALLY EFFICIENT FINITE ELEMENT METHOD SIMULATION OF VOLTAGE DRIVEN SOLID ROTOR SYNCHRONOUS MACHINES

A Thesis in

Electrical Engineering

by

Pasha S. Petite

Submitted in Partial Fulfillment
of the Requirements
for the Degree of

Master of Science

May 2008

The thesis of Pasha S. Petite was reviewed and approved* by the following:

Heath Hofmann
Associate Professor of Electrical Engineering
Thesis Advisor

Jeffrey Mayer
Associate Professor of Electrical Engineering

Kenneth Jenkins
Professor of and Head of Department of Electrical Engineering

*Signatures are on file in the Graduate School

ABSTRACT

This thesis applies a numerically efficient finite element method to the simulation of a two dimensional cross-section of a solid rotor synchronous machine. The finite element method is capable of determining the steady-state behavior of the machine in the presence of a periodic but otherwise arbitrary voltage input. This is achieved by iteratively simulating the system over a single period of the input voltage and modifying the initial state vector of the system until an initial state vector is found that results in a sufficiently close final state vector. The steady-state solution is solved using a shooting-Newton method. The shooting-Newton method is made more efficient through the use of a generalized minimum residual linear solver (GMRES). Results show that the shooting-Newton/GMRES method can be used to determine steady-state behavior in the presence of voltage inputs with generally non-uniform time steps. The results also show that the shooting-Newton/GMRES method is faster than conventional transient simulation at determining the steady state behavior of the machine model used.

TABLE OF CONTENTS

LIST OF FIGURES

ACKNOWLEDGEMENTS

I wish to sincerely thank Dr. Hofmann for his invaluable mentorship throughout this work. His patience, good nature, and teaching ability are exemplary among instructors I have encountered in my experience as a student.

I also wish to thank Dr. Mayer for his service as a committee member for this thesis.

Finally, I wish to thank the faculty, students, and staff in this department for their friendship and encouragement during my time at Penn State.

Chapter 1

Motivation and Background for the Numerical Simulation Method

1.1 Motivation

This thesis concerns a method for simulating the behavior of electromechanical systems. The need for improved methods to simulate electromechanical systems, including generators and motors, continues to grow due to increasing demands on the performance of such systems. While good simulation methods exist for electromechanical systems, there is room for improvement in the computational efficiency of such methods as well as the type of systems that can be simulated.

Engineers are interested in accurately and efficiently modeling and simulating electromechanical systems for a variety of applications. The need for electric power generation continues to grow to meet the increasing demand of consumers, and thus drives the development of new electromechanical machines. New requirements for electric power, including electric motors for automotive applications, also drive such development. The cost of electric power and increased efficiency and performance requirements mean that engineers require improved tools to design electromechanical systems.

The simulation of electromechanical systems is not trivial. As more accurate models for systems are developed, the computational complexity needed to simulate such systems grows rapidly. The most general tool that engineers use to simulate the behavior

of electromechanical systems is the family of finite element methods. Such methods divide a system into a large number of small parts called “elements”. The differential equations that govern the interaction of these elements are then determined. The collection of elements and the related system of differential equations constitutes a mathematical system. This system can be simulated using any number of available numerical methods. Typically, such methods become more accurate when more elements are used. An excellent reference addressing both methods and challenges pertaining to the simulation of electric machines is provided by **[1]**.

While these methods are highly accurate, it is not unusual for days or even weeks to be required in order to simulate a few seconds of the behavior of a motor or generator. The length of time needed to perform the simulation grows with the number of elements used. For a given simulation it is possible to choose the number of elements such that it would take years or more for the simulation to complete! Clearly, it is desirable to decrease the computational cost of such simulations so that increasingly accurate simulations can be performed within a reasonable time frame.

This thesis concerns the efficient simulation of solid rotor synchronous machines, a particular type of electromechanical system. In particular, it concerns an efficient method for determining the steady-state behavior of this type of machine.

1.2 Numerical Simulation Method

An intuitive method of determining the steady-state dynamics of a system driven with a periodic input is simply to choose an initial condition for the system and then

simulate the system until the transient response due to the initial condition has dissipated to within a satisfactory tolerance. That is, the simulation can be run until such a time as the Euclidian norm of the difference between the system state vector $\vec{x}(t)$ at time $t_0 = (n-1)T$ and at time $t_1 = nT$ is within a chosen tolerance ε. The simulation is complete when

$$\left\|\vec{x}(nT) - \vec{x}((n-1)T)\right\| < \varepsilon \qquad \textbf{(1)}$$

This requires running the simulation for n periods. Without prior knowledge of the speed at which the transient dynamics of the system will decay it is impossible to predict the value of n for a given value of ε. Many systems, including electromechanical systems, may have very slow transient dynamics; hence a better method for determining the steady-state behavior of a system is desired.

An alternative is to use a "shooting" method to determine the initial condition such that the state at time $t_0 = 0$ is equal to the state at time $t_1 = T$, where T is the period of the input. The method is called a shooting method because it searches (or "shoots") for the desired initial condition. Specifically, the shooting-Newton method is shown in **[2]** to be a useful algorithm for solving the steady-state initial condition problem. A shooting method uses an iterative process to determine the initial condition $\vec{x}_0 = \vec{x}(0)$ that results in the desired $\vec{x}(T)$

$$\left\|\vec{x}(T) - \vec{x}(0)\right\| < \varepsilon \qquad \textbf{(2)}$$

An initial guess is made for $\vec{x}(0)$ and the system is simulated over period T. The value of $\|\vec{x}(T) - \vec{x}(0)\|$ is then computed and the guess is updated.

In general, practical systems have non-linear dynamics and are hence referred to as non-linear systems. A non-linear system can be characterized by a state transition function that specifies how the state of the system changes between an initial time t_0 and final time t.

$$\vec{x}(t) = \Phi(\vec{x}(t_0), t_0, t) \tag{3}$$

If the state transition function Φ were known then simulation of the system would be unnecessary. As Φ is unknown, the Newton-Raphson method is used to determine the value of the state vector $\vec{x}(t)$ at each time step as in **[3]**.

Applying Newton-Raphson yields the following equation relating the j^{th} guess $\vec{x}(0)^j$ for the initial condition and the previous guess $\vec{x}(0)^{j-1}$

$$\vec{x}(0)^j = \vec{x}(0)^{j-1} + \left[J_\Phi(\vec{x}(0)^{j-1}, 0, T) - I\right]^{-1}\left[\vec{x}(0)^{j-1} - \Phi(\vec{x}(0)^{j-1}, 0, T)\right] \tag{4}$$

where J_Φ is the Jacobian of the state transition function Φ. Each iteration involves updating $\vec{x}(0)$ as in Eq. **[4].** This continues until Eq. **[2]** is satisfied. The computation of $\Phi(\vec{x}(0)^{j-1}, 0, T)$ is carried out throughout the simulation using backward Euler integration, hence $\Phi(\vec{x}(0)^{j-1}, 0, T)$ is defined implicitly. Unfortunately, this means that $J_\Phi(\vec{x}(0)^{j-1}, 0, T)$ cannot be computed directly by differentiation of $\Phi(\vec{x}(0)^{j-1}, 0, T)$.

An efficient alternative to computing this Jacobian involves the use of a Krylov subspace method. Specifically, the generalized minimum residual method (GMRES) is

shown in **[4]** to be an efficient method of solving for $\vec{x}(0)^j$ in Eq. **[4]**. The GMRES method, developed in **[5],** solves for the vector $\vec{v}$ in $A\vec{v} = \vec{b}$ without knowledge of the matrix A. Instead, this method requires that the matrix-vector product $A\vec{v}$ can be generated for arbitrary $\vec{v}$.

The GMRES method is an iterative method that seeks to minimize the residual

$$r_i := \left\| \vec{b} - A\vec{v}_i \right\| \tag{5}$$

for each iteration (starting with $i = 1$): The i^{th} Krylov subspace is defined as

$$K_i = span\left\{ \vec{b}, A\vec{b}, ..., A^{i-1}\vec{b} \right\} \tag{6}$$

The i^{th} approximation $\vec{v}_i$ is within the space spanned by K_i. Gram-Schmidt orthagonalization is used to determine a set of orthogonal basis vectors $\vec{p}_1, \vec{p}_2, ..., \vec{p}_i$ for K_i. A vector $\vec{v}_i$ that is within the span of this basis is chosen to minimize r_i. The value of *i* is incremented and this process is repeated until the value of r_i is sufficiently small. That is, the GMRES method halts and returns the i^{th} approximation of *v* when

$$\left\| \vec{b} - A\vec{v}_i \right\| < \gamma \tag{7}$$

where γ is the desired tolerance.

Rewriting Eq. **[4]** as

$$\left[J_\Phi(\vec{x}(0)^{j-1}, 0, T) - I \right] \left(\vec{x}(0)^j - \vec{x}(0)^{j-1} \right) = \vec{x}(0)^{j-1} - \Phi(\vec{x}(0)^{j-1}, 0, T) \tag{8}$$

yields a linear equation of the desired form. The GMRES method can then be applied after each single-period simulation of the system to obtain the updated guess for the initial condition.

In two previous works by the advisor of this thesis **[6**, **7]** the shooting-Newton/GMRES method was applied to determine the steady-state dynamics of electromechanical machines in the presence of eddy current losses. This method was then expanded to include the effect of insulating barriers within the rotor of a machine **[8]**. In **[7]** and **[8]** the method was shown to be more computationally efficient than using a transient simulation method that is run until steady-state convergence.

In the work thus far the simulation has been designed to accept as inputs currents that are constant in the reference frame that is synchronous with the electrical angle of the rotor. This simplifies the simulation somewhat by minimizing the number of time-steps required for the simulation of each period and also allowing constant time steps to be used in the simulation. However, such inputs are not generally realistic for practical machines. Instead, the simulation method can be modified to accept arbitrary voltage inputs with non-uniform time steps. This allows the modeling of rotor losses due to the harmonics resulting from pulse-width modulation generated inputs.

It should be noted that literature searches related to efficient finite-element methods for finding the steady-state solution for synchronous machines produce somewhat limited results. The area remains fairly open for additional work. In **[9]** a different finite-element method for determining the periodic steady-state solution for a machine with voltage inputs is presented. However, this method is valid only for the case where the steady-state currents have an average value of zero. Also, this method cannot

be used to calculate the rotor eddy current losses. In certain applications it is desirable to minimize rotor losses in a machine design; hence an alternative approach to this simulation method is necessary. The method described in this thesis takes these rotor losses into account.

Chapter 2

Machine Model and Voltage Inputs

2.1 Machine Model

This thesis expands upon the shooting-Newton/GMRES method to allow the use of periodic but otherwise arbitrary voltage inputs in the simulation of a synchronous machine. This includes the ability not only to define the voltages arbitrarily, but also to allow any choice of time steps, including dynamically-varying time steps, in the simulation of the system. This method has the advantage of being able to account for rotor losses and has the capacity to simulate practical voltage inputs.

The method is developed here in general and is then applied to simulate the machine for the case of a practical voltage input. The physical model is developed for the stator and for the rotor, and then the finite-element model is developed from these dynamics.

2.1.1 Materials Model

The model for the magnetic properties of the materials in the machine is chosen as in **[7]**. That is, the magnetization $\vec{M}$ of the materials is modeled as an equivalent current density. The mapping between the magnetization and the magnetic flux density can be

modeled as non-linear. The direction of $\vec{M}$ is assumed to be the same as the direction of $\vec{B}$ and the magnitude of the magnetization is described by this nonlinear mapping.

$$\vec{M} = \frac{\left\|\vec{M}\right\|\left(\left\|\vec{B}\right\|\right)}{\left\|\vec{B}\right\|}\vec{B} \qquad (\mathbf{9})$$

Note that the magnetic flux density $\vec{B}$ is a function of the magnetic vector potential $\vec{A}$.

$$\vec{B} = \nabla \times \vec{A} \qquad (\mathbf{10})$$

which implies that the magnetization can alternatively be considered as a function of the magnetic vector potential. The ability to model non-linear magnetization is used in this thesis to model saturation. However, it is also possible to model hysteresis using this method.

A two-dimensional cross section of the machine will be modeled, hence the Jacobian of the $\vec{M} - \vec{B}$ relationship is required in the two-dimensional form:

$$\begin{bmatrix} \frac{dM_x}{dB_x} & \frac{dM_x}{dB_y} \\ \frac{dM_y}{dB_x} & \frac{dM_y}{dB_y} \end{bmatrix} = \begin{bmatrix} \left(\frac{B_x}{\left\|\vec{B}\right\|}\right)^2\left(\frac{d\left\|\vec{M}\right\|}{d\left\|\vec{B}\right\|} - \frac{\left\|\vec{M}\right\|}{\left\|\vec{B}\right\|}\right) + \frac{\left\|\vec{M}\right\|}{\left\|\vec{B}\right\|} & \frac{B_x B_y}{\left\|\vec{B}\right\|^2}\left(\frac{d\left\|\vec{M}\right\|}{d\left\|\vec{B}\right\|} - \frac{\left\|\vec{M}\right\|}{\left\|\vec{B}\right\|}\right) \\ \frac{B_x B_y}{\left\|\vec{B}\right\|^2}\left(\frac{d\left\|\vec{M}\right\|}{d\left\|\vec{B}\right\|} - \frac{\left\|\vec{M}\right\|}{\left\|\vec{B}\right\|}\right) & \left(\frac{B_y}{\left\|\vec{B}\right\|}\right)^2\left(\frac{d\left\|\vec{M}\right\|}{d\left\|\vec{B}\right\|} - \frac{\left\|\vec{M}\right\|}{\left\|\vec{B}\right\|}\right) + \frac{\left\|\vec{M}\right\|}{\left\|\vec{B}\right\|} \end{bmatrix} \qquad (\mathbf{11})$$

2.1.2 Stator Model

Magnetic field intensity $\vec{H}$ and current density $\vec{J}$ are related by the quasi-static form of Ampere's Law:

$$\nabla \times \vec{H} = \vec{J} \tag{12}$$

Substituting the expression for the magnetic flux density in terms of field intensity and magnetization

$$\vec{B} = \mu_0(\vec{H} + \vec{M}) \tag{13}$$

yields

$$\vec{J} = \nabla \times \left(\frac{\vec{B}}{\mu_0} - \vec{M} \right) \tag{14}$$

Applying the relationship between the flux density and the magnetic vector potential to Eq. **[14]** yields

$$\vec{J} = \frac{1}{\mu_0} \nabla \times \nabla \times \vec{A} - \nabla \times \vec{M} \tag{15}$$

Using the Coulomb gauge $\left(\nabla \cdot \vec{A} = 0\right)$ and applying the following identity

$$\nabla \times \nabla \times \vec{A} = -\nabla^2 \vec{A} + \nabla(\nabla \cdot \vec{A}) \tag{16}$$

for the case where the currents are enforced, Eq. **[15]** becomes

$$-\frac{1}{\mu_0} \nabla^2 \vec{A} = \vec{J} + \nabla \times \vec{M} \tag{17}$$

A model of a two-dimensional cross-section of the machine is desired and the currents are modeled as orthogonal to this cross-section. Thus the magnetic vector potential and current density can be reduced to a scalar:

$$-\frac{1}{\mu_0} \nabla^2 A = J + \nabla \times \vec{M} \tag{18}$$

Applying the method of weighted residuals to Eq. **[18]**:

$$\int_{\Omega}\left(-\frac{1}{\mu_0}\nabla^2 A - J + \nabla\times\vec{M}\right)w_i d\Omega = 0, \quad \forall\ w_i \tag{19}$$

and applying Green's identity

$$w_i\nabla^2 A = \nabla\cdot(w_i\nabla A) - \nabla w_i\cdot\nabla A \tag{20}$$

yields

$$\frac{1}{\mu_0}\int_{\Omega}\nabla w_i\cdot\nabla A d\Omega = \int_{\Omega} w_i J d\Omega + \int_{\Omega}(\nabla\times\vec{M})w_i d\Omega + \frac{1}{\mu_0}\int_{\Omega}\nabla\cdot(w_i\nabla A)d\Omega \tag{21}$$

where

$$\int_{\Omega}\nabla\cdot(w_i\nabla A)d\Omega = \oint_{\Gamma} w_i\nabla A d\Gamma \tag{22}$$

The magnetic vector potentials are represented as a weighted sum, i.e. $A = \sum_j a_j w_j$. Note that Dirichlet boundary conditions are chosen for the problem, hence $w_i = 0$ at the machine boundaries. This results in $\oint_{\Gamma} w_i\nabla A d\Gamma = 0$ in Eq. **[22]**. This representation yields

$$\sum_j \frac{1}{\mu_0}\int_{\Omega}\sum_j A_i\nabla w_j\nabla w_i\cdot d\Omega = \int_{\Omega} w_i J d\Omega + \int_{\Omega}(\nabla\times\vec{M})w_i d\Omega \quad , \quad \forall i \tag{23}$$

where the $\int_{\Omega}(\nabla\times\vec{M})w_i d\Omega$ term can be simplified using the identity

$$\nabla\times(w_i\vec{M}) = w_i\nabla\times\vec{M} - \vec{M}\times\nabla w_i \tag{24}$$

to

$$\int_{\Omega}(\nabla\times\vec{M})w_i d\Omega = \int_{\Omega}\nabla\times(w_i\vec{M})d\Omega + \int_{\Omega}\vec{M}\times\nabla w_i d\Omega \tag{25}$$

and noting that

$$\int_{\Omega} \nabla \times (w_i \vec{M}) d\Omega = \int_{\Gamma} w_i \vec{M} d\vec{\Gamma} = 0 \qquad (26)$$

by the curl theorem, yields

$$\int_{\Omega} (\nabla \times \vec{M}) w_i d\Omega = \int_{\Omega} \vec{M} \times \nabla w_i d\Omega \qquad (27)$$

The finite-element model for the stator is then formed using triangular elements and linear interpolation as in **[10]**.

2.1.3 Rotor Model

The rotor model is determined by applying Faraday's law

$$\nabla \times \vec{E} = \frac{-d\vec{B}}{dt} \qquad \Leftrightarrow \qquad \nabla \times \left(\vec{E} + \frac{d\vec{A}}{dt} \right) = 0 , \qquad (28)$$

which implies that

$$\vec{E} + \frac{d\vec{A}}{dt} = -\nabla \phi , \qquad (29)$$

where ϕ is a scalar. There is no potential applied to the rotor, thus the current density can be expressed in terms of the magnetic vector potential as

$$\vec{J} = \sigma \vec{E} = -\sigma \frac{d\vec{A}}{dt} \qquad (30)$$

which implies

$$\sigma \frac{d\vec{A}}{dt} - \frac{1}{\mu_0} \nabla^2 \vec{A} = \nabla \times \vec{M} \qquad (31)$$

Again, the magnetic vector potential can be reduced to a scalar

$$\sigma \frac{dA}{dt} - \frac{1}{\mu_0} \nabla^2 A = \nabla \times \vec{M} \tag{32}$$

Applying the method of weighted residuals to Eq. **[32]** yields

$$\frac{d}{dt} \int_{\Omega} \sigma \sum_j A_j w_j w_i d\Omega + \frac{1}{\mu_0} \int_{\Omega} \sum_j A_j \nabla w_j \cdot \nabla w_i d\Omega = \int_{\Omega} \vec{M} \times \nabla w_i d\Omega \tag{33}$$

As was the case for the stator, the finite-element model for the rotor is also formed using triangular elements and linear interpolation as in **[10]**.

The dynamic equations for the finite element model are thus given by

$$\begin{aligned} S_a \frac{d}{dt} \vec{a} + K_a \vec{a} &= \vec{I}_m(\vec{a}) + T_n \vec{i}_m \\ -S_\phi^T \frac{d}{dt} \vec{a} + G \overrightarrow{\nabla \phi} &= T_c \vec{i}_m \end{aligned} \tag{34}$$

where $\vec{a}$ is the magnetic vector potentials at the nodes in the FEM mesh, $\overrightarrow{\nabla \phi}$ is a vector consisting of winding potential gradients, and $\vec{i}_m$ is the two phase winding current vector in the stator reference frame.

The matrix *G* is used to multiply the conductivity and area of each conductor by $\overrightarrow{\nabla \phi}$. The matrix T_n is the product of two matrices. The first converts the two phase equivalent winding current vector to three phase currents, and the second distributes these currents appropriately to the finite element nodes. Similarly, the matrix T_c converts the two phase equivalent current vector to three phase currents and then converts these currents into the appropriate currents in each conductor in the stator slots. The matrix S_ϕ^T is used to calculate the average derivative of the magnetic vector potentials in each

conductor and multiply these by the each conductor's conductivity. The matrix S_a represents the conductivity of each element. The matrix K_a follows from the magnetic permeability of the elements. The elements of S_a and K_a are determined as

$$\begin{aligned} S_{ij} &= \int_{\Omega} \sigma w_i w_j d\Omega \\ K_{ij} &= \frac{1}{\mu_0} \int_{\Omega} \nabla w_i \cdot \nabla w_j d\Omega \end{aligned} \quad \textbf{(35)}$$

Note that from Eq. **[17]** and Eq. **[31]** it can be seen that the magnetization is a function of the magnetic vector potential. Thus the magnetization in the finite element model can be written as $\vec{M}(\vec{a})$. $\vec{I}_m(\vec{a})$ is determined as

$$\vec{I}_{m,i}(\vec{a}) = \int_{\Omega} \vec{M}(\vec{a}) \times \nabla w_i d\Omega \quad \textbf{(36)}$$

The dynamics associated with the leakage inductance and resistance are represented as

$$L_{leak} \frac{d}{dt} \vec{i}_m + R_{leak} \vec{i}_m + T_v \overrightarrow{\nabla \phi} = \vec{v}_m , \quad \textbf{(37)}$$

where R_{leak} and L_{leak} are the leakage resistance and leakage inductance (respectively), and T_v is the matrix used to determine the equivalent two-phase voltages from $\overrightarrow{\nabla \phi}$.

The equations above are combined into a single differential algebraic equation that is suitable for use in numerical simulation.

$$S \frac{d\vec{x}}{dt} + K\vec{x} = \vec{u} , \quad \textbf{(38)}$$

where the state vector is represented as

$$\vec{x} = \begin{bmatrix} \vec{a} \\ \overrightarrow{\nabla \phi} \\ \vec{i}_m \end{bmatrix},$$

and the S and K matrices are

$$S = \begin{bmatrix} S_a & 0 & 0 \\ S_\phi^T & 0 & 0 \\ 0 & 0 & L_{leak} I \end{bmatrix},$$

$$K = \begin{bmatrix} K_a & 0 & -T_n \\ 0 & G & T_c \\ 0 & T_v & R_{leak} I \end{bmatrix},$$

where $I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$. The input is represented as

$$\vec{u} = \begin{bmatrix} \vec{I}_m(\vec{a}) \\ 0 \\ \vec{v}_m \end{bmatrix}$$

Because the current vector $\vec{I}_m(\vec{a})$ is a non-linear function of the magnetic vector potentials it is convenient to write the dynamics in this abbreviated form. Of course, the input to the machine is simply the commanded voltage vector $\vec{v}_m$.

Writing the backward Euler equation for the finite element dynamics

$$S\left(\frac{\vec{x}_j - \vec{x}_{j-1}}{h_n}\right) + K\vec{x}_n = \vec{u}_n(\vec{x}_n) \qquad \textbf{(39)}$$

and taking the derivative with respect to the initial state vector $\vec{x}_0$ yields

$$\left(\frac{S}{h_n} + K - \frac{d\vec{u}_n}{d\vec{x}_n}\right)\frac{d\vec{x}_n}{d\vec{x}_0} = \frac{S}{h_n}\frac{d\vec{x}_{n-1}}{d\vec{x}_0} \qquad \textbf{(40)}$$

It is possible to determine $J_\Phi(x(0)^{j-1},0,T)$ by assigning $\vec{x}_0 = \vec{x}(0)^{j-1}$ and iteratively solving Eq. **[40]** over the entire period to get $J_\Phi(x(0)^{j-1},0,T) = \frac{d\vec{x}(T)}{d\vec{x}_0}$. However, in order to utilize GMRES it is only necessary to be able to compute arbitrary matrix vector products $J_\Phi(x(0)^{j-1},0,T)\vec{v}$, thus the Jacobian is never actually computed explicitly. These matrix vector-products can be generated by substituting the vector $\vec{v}$ for $\frac{d\vec{x}_0}{d\vec{x}_0}$ and iteratively solving Eq. **[40]** for the total number of time steps in one period.

2.2 Voltage Inputs

For variable-speed operation the machine requires corresponding variable-frequency voltage inputs. In order to convert available utility voltages to the desired frequency it is necessary to utilize power electronics. Typically a diode rectifier circuit is used to produce a DC bus voltage V_{bus} from the available utility voltages. This bus voltage is then used to power a three-phase inverter circuit. A typical inverter circuit cannot directly produce an arbitrary waveform. Instead, it is capable of switching quickly between two distinct output voltage levels. The inverter switches between these voltages once per switching period T_s. The value of T_s is chosen to be much smaller than the period of the desired input voltage signal.

For an inverter capable of outputting voltage levels $\frac{V_{bus}}{2}$ and $\frac{-V_{bus}}{2}$ the average value of the signal over one switching period is given by

$$\bar{v}(t) = \frac{1}{T_s}\int_0^{T_s} v_{PWM}(t)dt = V_{bus}\left(D - \frac{1}{2}\right) \quad \textbf{(41)}$$

where D is the duty cycle for that switching period.

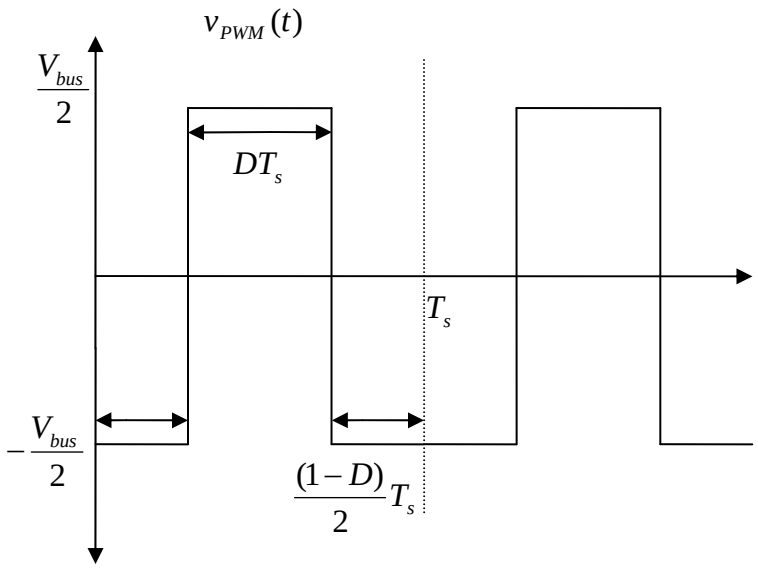

Figure **1**: Example of center-based pulse-width modulation waveform with switching period T_s and duty cycle D

In order to generate an arbitrary signal the duty cycle is varied over time to achieve the desired average value of the signal over each switching period. Choosing the duty cycle as

$$D(t) = \frac{1}{2} + A\cos(\omega_{re}t) \quad \textbf{(42)}$$

yields an average-value signal

$$v_{ave} = AV_{bus}\cos(\omega_{re}t) \qquad (\mathbf{43})$$

It is obvious that $v_{ave}(t) \neq v_{PWM}(t)$. The actual signal $v_{PWM}(t)$ provided by pulse-width modulation contains the desired signal $v_{ave}(t)$ as well as higher frequency harmonics. It is the effect of these harmonics that has motivated the work presented here.

The use of PWM input voltages requires that variable time steps be used in the simulation. For simulation of a three-phase machine it is necessary to determine a PWM input voltage signal for each phase, hence there are three (generally) distinct duty cycle values corresponding to each switching period.

Because the finite element model is simulated numerically in discrete time steps it is necessary to represent $v_{PWM}(t)$ in discrete time. At a minimum, the voltage signal must be represented each time the voltage switches in any of the three phases of the PWM generated voltage. Clearly the value of the duty cycle during a given switching period is a continuously varying value. This means that the time simulation step size should be continuously variable in order to accurately simulate the exact times that switching occurs.

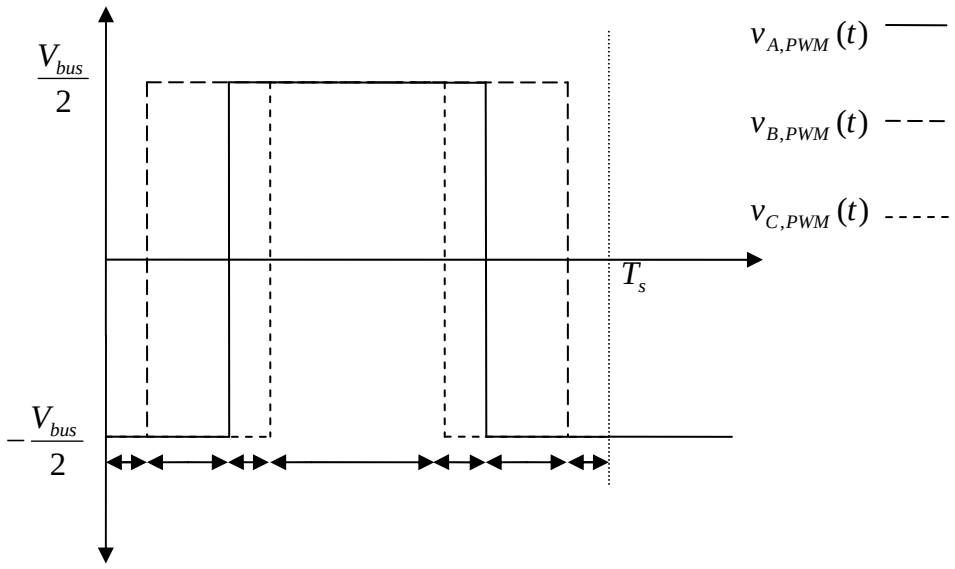

Figure **2**: Example of three-phase center-based PWM signals with distinct duty cycles. A minimum of six simulation time-steps are required per switching period T_s if the duty cycles are distinct.

Note that using more time steps will result in a more accurate simulation. In the simulation described here the compulsory time steps are further partitioned into smaller time steps in order to provide better resolution.

Chapter 3

Simulation and Results

3.1 Simulation with Ideal Voltage Inputs: Minimum Flux-Linkage Operating Point

The shooting-Newton/GMRES method is demonstrated first for sinusoidal voltage inputs with uniform time steps. That is, the voltage inputs are calculated for a desired operating point of the synchronous reluctance machine and the response to these inputs is determined using the shooting-Newton/GMRES method.

For a synchronous reluctance machine one operating point of particular interest is the minimum flux linkage operating point. This operating point is desirable because it minimizes core losses and minimizes saturation of the magnetic flux density within the machine. This operating point is also of interest because it maximizes the output power of the machine in the presence of voltage constraints. At this operating point the equivalent two-phase currents in the rotor reference frame are given by:

$$I^r = \begin{bmatrix} I_d^r \\ I_q^r \end{bmatrix} = \begin{bmatrix} I_{pk} \Big/ \sqrt{1 + L_d / L_q} \\ I_{pk} \Big/ \sqrt{1 + L_q / L_d} \end{bmatrix}, \qquad (\mathbf{44})$$

where I_{pk} is a given peak value of the current that we wish to achieve. The rotor frequency Ω_r is chosen to be 5445.4 *rad/s* (52000 *rpm*). The machine parameters are set to a 4 pole machine with 36 stator teeth and 3 rotor segments. The resulting rotor electrical frequency Ω_{re} is 10890.9 *rad/s*.

The steady-state voltage inputs corresponding to these desired steady-state currents are determined as

$$\vec{V}^{\,r} := \begin{bmatrix} V_d^r \\ V_q^r \end{bmatrix} = R\vec{I}^{\,r} + \Omega_{re} JL\vec{I}^{\,r}, \qquad \textbf{(45)}$$

where R is the two-phase equivalent resistance of the stator windings, Ω_{re} is the electrical frequency of the voltage inputs, $J = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$ is the two-dimensional equivalent to the imaginary unit, $L = \begin{bmatrix} L_d & 0 \\ 0 & L_q \end{bmatrix}$ is the matrix of inductances corresponding to the direct and quadrature axis components of the inductance, and $\vec{I}^{\,r} := \begin{bmatrix} I_d^r \\ I_q^r \end{bmatrix}$ is the vector of two-phase equivalent currents. The *r* superscript denotes that the elements are in the rotor electrical angle reference frame.

The vector $\vec{V}^{\,r}$ is then converted to the stator reference frame for use as the input in the simulation. This conversion is achieved using the rotation matrix:

$$\vec{V}(t) := \begin{bmatrix} V_d(t) \\ V_q(t) \end{bmatrix} = e^{J\theta_{re}(t)}\vec{V}^{\,r} \qquad \textbf{(46)}$$

Note that for this choice of reference frame the elements of $\vec{I}^{\,r}$ and $\vec{V}^{\,r}$ are constant.

The initial guess for x_0 is calculated by simply running a static simulation with the input vector $\vec{V}(0)$ determined using Eq. **[46]** at angle $\theta_{re}(0) = \theta_d$, where θ_d is the angle of the direct axis of the two-phase equivalent machine. The system is then simulated for one period with input $\vec{V}(t)$ and initial state x_0. The tolerance value for

convergence to steady-state is selected to be $\varepsilon = 10^{-6}$. The tolerance value for the GMRES method is selected to be $\gamma = 10^{-6}$. A PC with a 2.99 gigahertz CPU and 504 megabytes of RAM is used. After one period the guess of the initial state vector is updated as in Eq. **[8]**. This is repeated until the condition in Eq. **[2]** is satisfied.

The shooting-Newton/GMRES method converges after 8 iterations. The error between the final state and the initial state for the last iteration is 9.2928e-8. The simulation required 27385.468 seconds (approximately 7 hours and 37 minutes) to run completely. This includes time spent running the initial static simulation and calculating the input voltage $\vec{V}(t)$.

A plot of the magnetic flux density in the two-dimensional cross-section of the machine is shown in Figure **[3]** . The magnetic flux density is determined using the state vector at the start of the steady-state mechanical period.

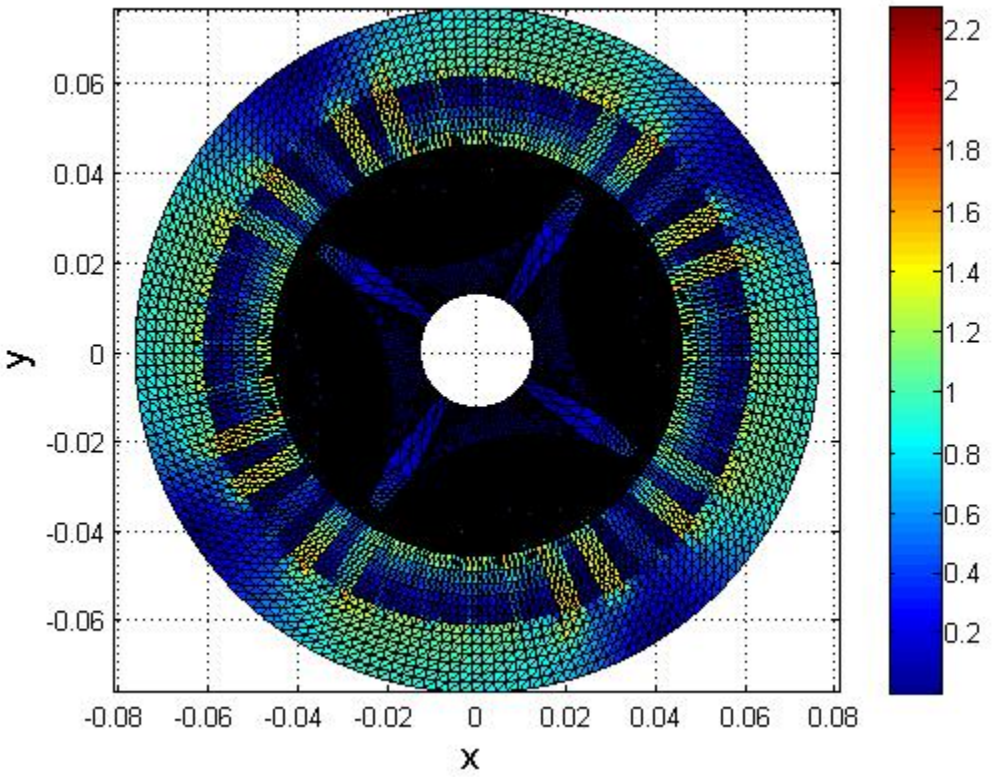

Figure **3**: The finite element mesh with magnetic flux density shown in units of Teslas. The flux density values correspond to the initial stator position θ_d in the steady-state solution (determined using the shooting-Newton/GMRES method).

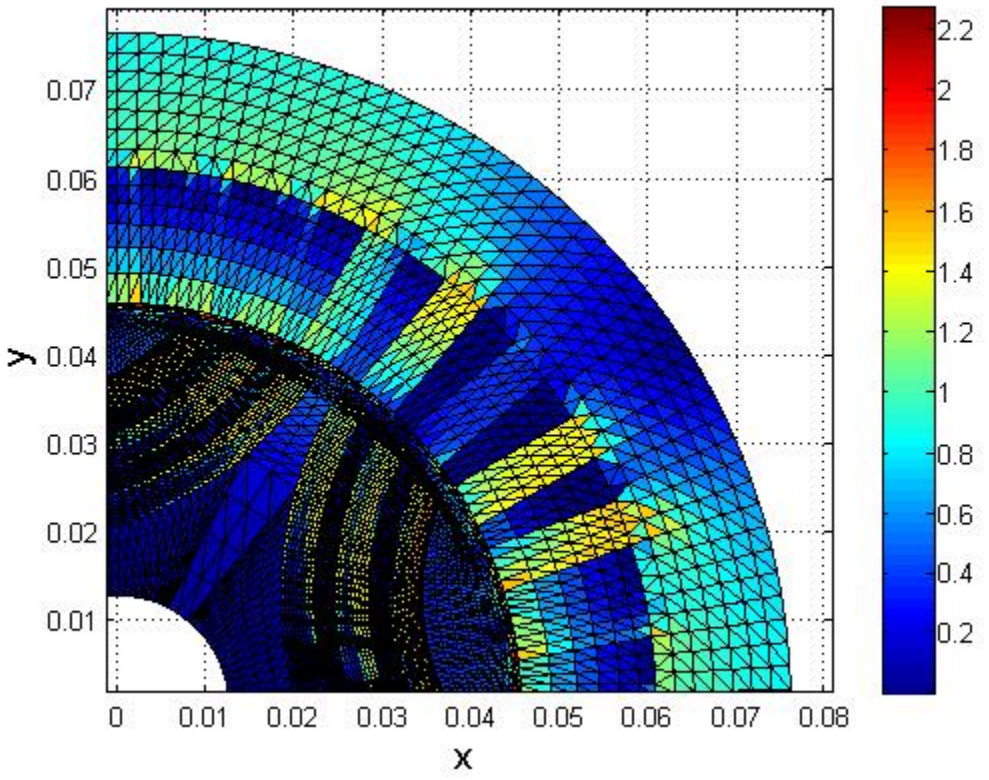

Figure **4**: Close-up of one quarter of the finite element mesh in Figure **[3]**

For purposes of comparison, a transient simulation is run with the same parameters and time steps. The same PC is used. It is unknown how quickly the transient simulation will converge to steady-state, so the simulation is initially run for 25 periods and the error is observed. This number of iterations is chosen because it is estimated to take approximately 24 hours to run (during debugging it was observed that slightly less than one hour is typically required to simulate one period). The first 25 periods of simulation require 75438.163 seconds (approximately 20 hours and 58 minutes) to run. After 25 periods the error between the final and initial states is 587.72.

The error value is not within the desired tolerance, so an additional 25 periods are run, this time requiring 74359.580 seconds (approximately 20 hours and 39 minutes). After 50 periods of transient simulation the error is 546.45, only a 7.02% decrease with respect to the error after 25 periods.

From the plots of magnetic flux density it can be seen that the flux densities from the transient simulation are typically higher than the steady-state flux densities determined using the shooting-Newton/GMRES method. This difference is not unexpected when considering the fact that the transient simulation has not yet converged to steady-state. Unfortunately, due to the extremely slow convergence of the transient simulation method it is impractical to continue this method until satisfactory convergence is obtained.

The error after 50 periods of transient simulation was a factor of 5.88×10^9 greater than the error determined using the shooting-Newton/GMRES method after only 8 iterations. The time taken for 50 periods of transient simulation was 149797.743 seconds, much more time than the 27385.468 seconds required for the shooting-Newton/GMRES method. Clearly this system converges very slowly to steady state and hence the conventional transient simulation method is poorly suited for this particular machine.

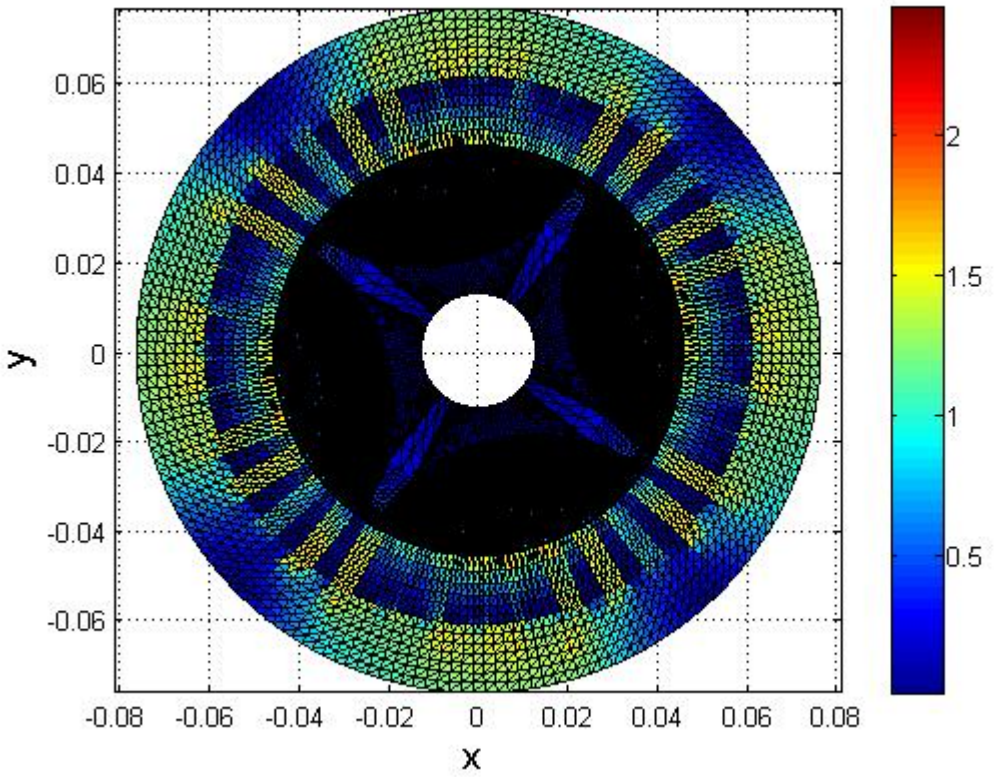

Figure 5: The finite element mesh with magnetic flux density at the end of the 50th period of transient simulation. Magnetic flux density values are in units of Teslas.

From these simulations it can be seen that the shooting-Newton/GMRES method works well for this machine in terms of speed of convergence to steady-state. The method can now be applied to a case where the voltage input vector $\vec{V}(t)$ is not composed of inputs with uniform time steps. Specifically, the input is generated using pulse-width modulation.

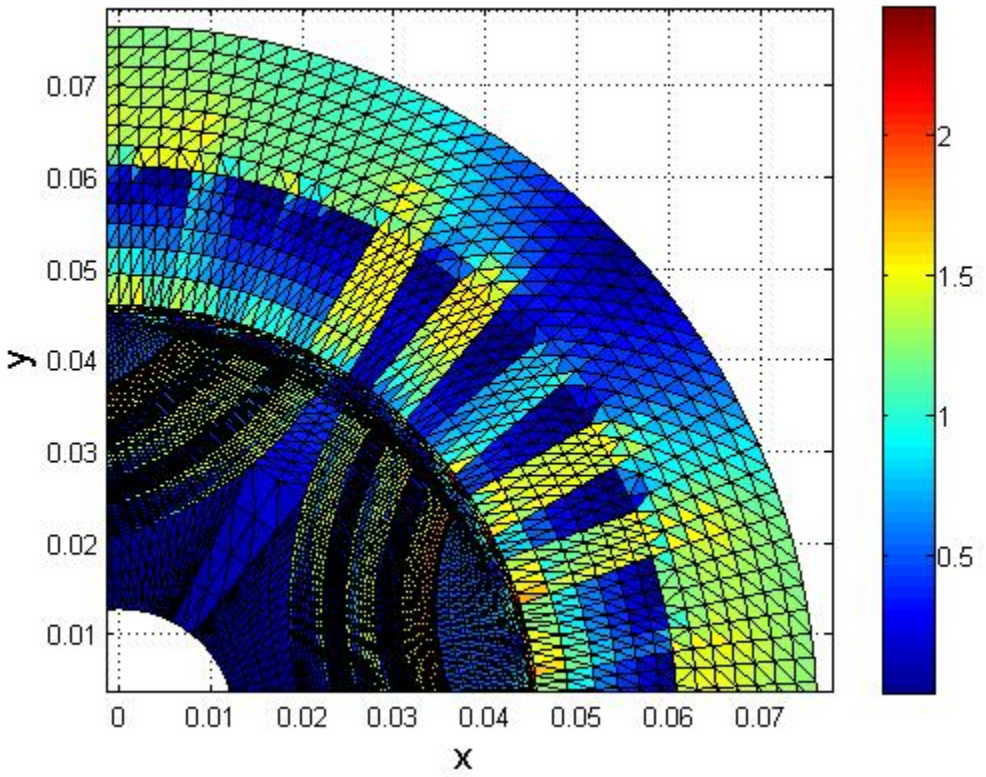

Figure **6**: Close-up of one quarter of the finite element mesh in Figure **[5]**

3.2 Simulation with Pulse-Width Modulation Voltage Inputs

Voltage inputs are generated using pulse-width modulation based upon the corresponding ideal voltages. That is, the method in section Error! Not a valid link. is followed to generate PWM voltage inputs corresponding to the input voltage vector $\vec{V}(t)$ from the previous simulations.

Using PWM requires the number of time steps used to be increased. The number of switching periods is chosen to be 20 and the number of time steps within each

switching period is chosen to be 10. To produce the PWM inputs, the signal $\vec{V}(t)$ is first sampled at 20 uniformly spaced sample times. This discrete-time vector is then converted to three-phase representation using the transformation:

$$\begin{bmatrix} V_c(t) \\ V_b(t) \\ V_c(t) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 \\ -1/2 & \sqrt{3}/2 & 1 \\ -1/2 & -\sqrt{3}/2 & 1 \end{bmatrix} \begin{bmatrix} V_d(t) \\ V_q(t) \\ 0 \end{bmatrix}, \quad \textbf{(47)}$$

where $\begin{bmatrix} V_a & V_b & V_c \end{bmatrix}^T$ is the vector of line-to-neutral voltage inputs for the machine. The duty cycle at each sample time is calculated using Eq. **[42]** for each of the three phases. The interval in between each of these sample times is then divided into 10 non-uniform time steps. The starting time for two, four, or six of these time steps are determined according to the duty cycles for each of the three phases. There will be six compulsory starting times if the three phase duty cycles are all different, four if two duty cycles are the same, and only two if all three are identical. The starting times of each of the remaining time steps are distributed uniformly within the sampling period. If the start time of one of these non-compulsory time steps coincides with the start time of one of the compulsory time steps then its start time is changed to fall in between that time and the previous one. This avoids the divide-by-zero error that would occur if a time step had zero length.

The accuracy of the simulation improves when more time steps are used. Unfortunately, there is always some constraint imposed by the available computing resources. In this case, the main limitation is the 4 gigabyte limitation on addressable memory that is imposed by the MATLAB software used to run the simulation. Virtual memory is used by MATLAB to increase the available memory beyond 504 megabytes.

However, the sum of virtual memory and physical RAM cannot exceed 4 gigabytes due to this limitation. Choosing 20 sample times and 10 time steps within each sample time results in a total number of 200 time steps. This is the largest total number of time steps that was found not to cause the software to run out of memory. Despite this constraint on the number of time steps that could be used the simulation produces useful results. This is good news because it shows that the computational cost associated with simulating the system with realistic voltage inputs is comparable to the cost of simulating the system with ideal voltage inputs.

The commanded output voltages at each time step are determined as in section **2.2** and the resulting three phase voltages are converted to the two-phase equivalent voltages using the inverse transformation:

$$\vec{V}_{PWM}(t) := \begin{bmatrix} V_{d,PWM}(t) \\ V_{q,PWM}(t) \\ V_{0,PWM}(t) \end{bmatrix} = \begin{bmatrix} 2/3 & -1/3 & -1/3 \\ 0 & \sqrt{3}/3 & -\sqrt{3}/3 \\ 1/3 & 1/3 & 1/3 \end{bmatrix} \begin{bmatrix} V_{c,PWM}(t) \\ V_{b,PWM}(t) \\ V_{c,PWM}(t) \end{bmatrix} \qquad (\mathbf{48})$$

As in the previous simulations, the initial guess for x_0 is calculated by simply running a static simulation with the input vector $\vec{V}(0)$. The system is then simulated for one period with input $\vec{V}_{PWM}(t)$ and initial state x_0. The same tolerance value is used and the same PC is used. The shooting-Newton/GMRES method converges after 16 iterations. The error between the final state and the initial state for the last iteration is 4.686×10^{-7}. The simulation required 63914.718 seconds (approximately 17 hours and 45 minutes) to run completely. This includes time spent running the initial static simulation and calculating the input voltage $\vec{V}_{PWM}(t)$.

The two-phase equivalent currents are shown in Figure **[7]**. The effect of PWM voltage inputs can be seen in the ripple that is added to the otherwise sinusoidal currents. Part of the motivation for this work was to be able to determine the effects of realistic input voltages on these currents.

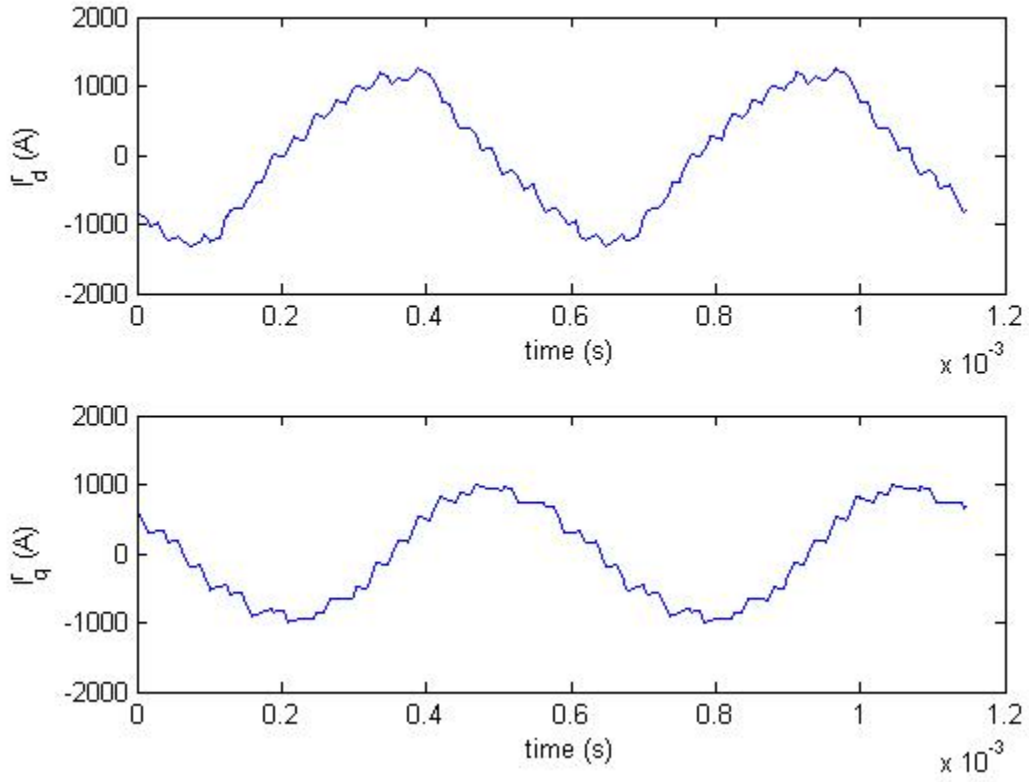

Figure **7**: Steady-state two phase equivalent currents due to input voltages produced using pulse-width modulation. Solved using the shooting-Newton/GMRES method with non-uniform time steps.

A plot of the magnetic flux density in the two-dimensional cross-section of the machine is shown in Figure **[8]**. Again, the magnetic flux density is determined using the state vector at the start of the steady-state mechanical period.

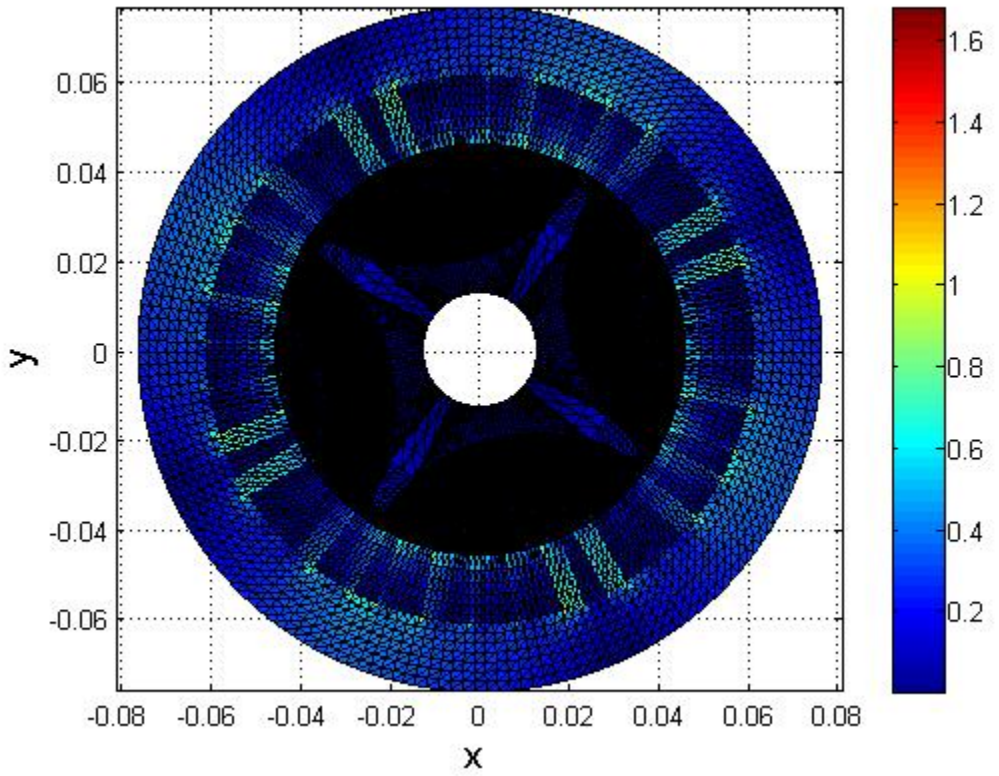

Figure **8**: The finite element mesh with magnetic flux density shown in units of Teslas. Solved using the shooting-Newton/GMRES method with voltage inputs determined using pulse-width modulation. The flux density values correspond to the initial stator position θ_d in the steady-state solution (determined using the shooting-Newton/GMRES method).

From these results it can be seen that shooting-Newton/GMRES method can be used to solve the steady-state behavior of the machine for the case where a non ideal input with non-uniform time steps is used. The method is very efficient, in this case

requiring only slightly more than twice the amount of time to run compared to the ideal case.

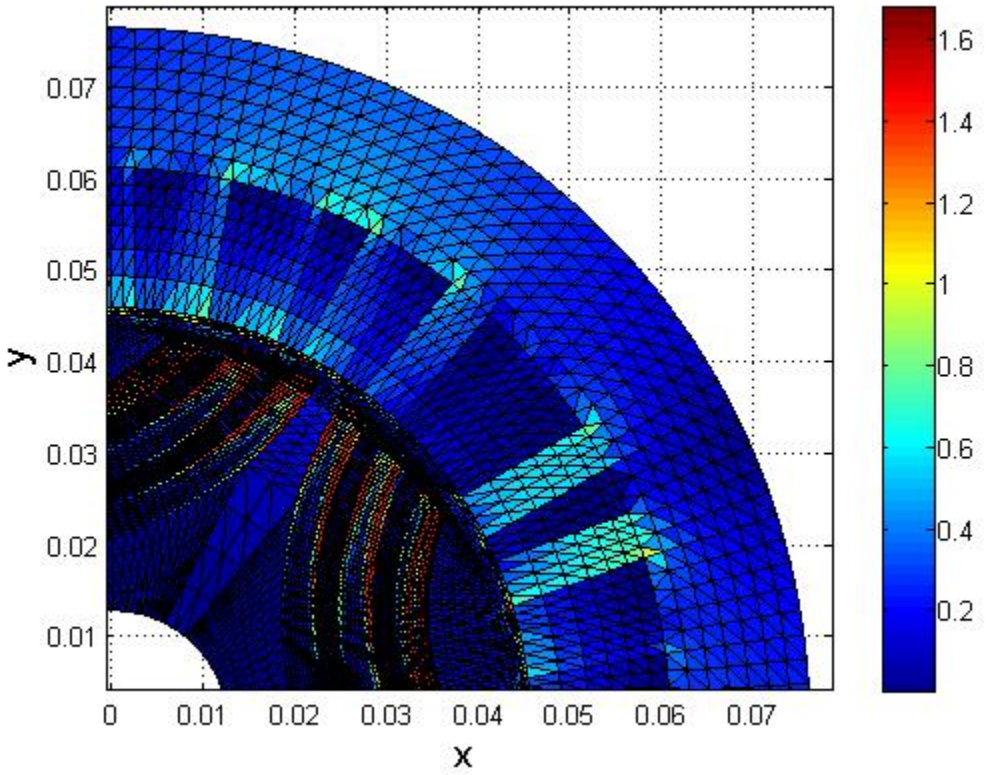

Figure **9**: Close-up of one quarter of the finite element mesh in Figure **[8]**

Chapter 4

Conclusions

The need for improved tools for electric machine design, including more efficient numerical simulation methods, has provided the motivation for this thesis. This thesis has expanded upon the shooting-Newton/GMRES method to show that it can be used to solve the steady-state behavior of solid-rotor synchronous machines with voltage inputs, including voltage inputs that require a non-uniform time-step size to be used in the simulation.

A two dimensional cross-section of a solid rotor synchronous machine was represented using a finite element mesh. The shooting-Newton/GMRES method was used to minimize the number of times that a period of motion of the machine needed to be simulated until the steady-state solution could be found.

This method was first applied to the machine model for the case where an ideal voltage input was used. That is, the voltage input was determined for a desired operating point and the resulting steady-state solution was determined. This voltage input was also used in a conventional transient simulation of the machine. The shooting-Newton/GMRES method was shown to converge quickly with respect to the speed of the transient simulation method. For this particular machine the transient simulation method converged so slowly that the error between the initial and final state vectors over the 50^{th} period was 5.4645×10^{8} times greater than the desired tolerance value of 10^{-6}, thus

making it an impractical method to use for this machine. The shooting-Newton/GMRES method provided an efficient and practical alternative to transient simulation.

The shooting-Newton/GMRES method was then used to solve the steady-state behavior for the case where the voltage inputs were generated using pulse-width modulation. The use of pulse-width modulation voltage inputs required that the step size of the simulation be non-uniform. The shooting-Newton/GMRES method was able to determine the steady-state behavior of the machine to within a tolerance of 4.686×10^{-7} after 16 iterations.

Several applications for this method exist. For example, from the steady-state solution determined using this method the rotor losses can be calculated and thus machines can be more effectively designed to minimize these losses. Also, the steady-state behavior of the machine in the presence of other practical voltage inputs can be determined using this method, thus allowing designers to optimize the machine parameters to yield improved performance in the presence of practical voltage inputs. The ability to efficiently determine the steady-state solution for solid rotor synchronous reluctance machines with voltage inputs will thus provide a useful tool in the design of these machines.

References

[**1**] Salon, Sheppard J, *Finite Element Analysis of Electrical Machines*. London: Kluwer Academic Publishers, 1995.

[**2**] M. Kakisaki and T. Sugawara, "A modified Newton method for the steady-state analysis," *IEEE Tran. Computer-Aided Design*, vol. CAD-4, pp. 662-667, Oct. 1985.

[**3**] J. D. Lambert, *Numerical Methods for Ordinary Differential Systems: The Initial Value Problem*. West Sussex, U.K.: John Wiley and Sons, 1991, pp. 12-13.

[**4**] R. Telichevesky, K Kundert, and J. White, "Efficient steady-state analysis based on matrix-free Krylov-subspace methods," in *Proc. 32nd Design Automation Conf.*, San Francisco, CA, June 1995.

[**5**] Y. Saad and M. H. Schultz, "GMRES: A generalized minimum residual algorithm for solving nonsymmetric linear systems," *SIAM J. Sci. Stat. Comput.*, vol. 7, pp. 856-869, 1986.

[**6**] H. Hofmann and S. R. Sanders, "High-speed synchronous reluctance machine with minimized rotor losses," *IEEE Trans. Ind. Applicat.*, vol. 36, pp. 531-539, Mar.-Apr. 2000.

[**7**] S. Li and H. Hofmann, "Numerically efficient steady-state finite element analysis of magnetically saturated electromechanical devices using a shooting-Newton/GMRES approach," *IEEE Trans. Magn.*, vol. 39, no. 6, pp. 3481-3485, Nov. 2003.

[**8**] D. Zhong and H. Hofmann, "Steady-state finite-element solver for rotor eddy currents in permanent-magnet machines using a shooting-Newton/GMRES approach," *IEEE Trans. Magn.*, vol. 40, no. 5, pp. 3249-3253, Sep. 2004.

[**9**] D. Dyck and P. J. Weicker, "Periodic steady-state solution of voltage driven magnetic devices," *IEEE Trans. Magn.*, vol. 43, no.4, pp. 1533-1536, April 2007.

[**10**] H. Hofmann, "High-Speed synchronous reluctance machine for flywheel applications," Ph.D. dissertation, Dep. Elect. Eng. Comput. Sci., Univ. California, Berkely, Dec. 1998.