Massachusetts Institute of Technology  
Department of Electrical Engineering and Computer Science  

6.11s: Mathematical Assistant Aided Design of Electric Machines and Drives  

Class Notes 7: Induction Machine Simulation and Control  

March 24, 2002

1 Introduction

The simulation models developed in Chapter 4 of these notes can be easily adapted to induction machines, which are for many purposes easier to handle analytically. In this chapter we develop models for simulation and control of induction motors. The derivation is quite brief for it relies on what we have already done for synchronous machines. In this chapter, however, we will stay in “ordinary” variables, skipping the per-unit normalization.

One of the more useful impacts of modern power electronics and control technology has enabled us to turn induction machines into high performance servomotors. In this note we will develop a picture of how this is done. Quite obviously there are many details which we will not touch here. The objective is to emulate the performance of a DC machine, in which (as you will recall), torque is a simple function of applied current. For a machine with one field winding, this is simply:

\[ T = GI_f I_a \]

This makes control of such a machine quite easy, for once the desired torque is known it is easy to translate that torque command into a current and the motor does the rest.

Of course DC (commutator) machines are, at least in large sizes, expensive, not particularly efficient, have relatively high maintenance requirements because of the sliding brush/commutator interface, provide environmental problems because of sparking and carbon dust and are environmentally sensitive. The induction motor is simpler and more rugged. Until fairly recently the induction motor has not been widely used in servo applications because it was thought to be "hard to control". As we will show, it does take a little effort and even some computation to do the controls right, but this is becoming increasingly affordable.

2 Elementary Model:

We return to the elementary model of the induction motor. In ordinary variables, referred to the stator, the machine is described by flux-current relationships (in the d-q reference frame):

\[
\begin{bmatrix}
\lambda_{dS} \\
\lambda_{dR} \\
\lambda_{qS} \\
\lambda_{qR}
\end{bmatrix} =
\begin{bmatrix}
L_S & M \\
M & L_R
\end{bmatrix}
\begin{bmatrix}
i_{dS} \\
i_{dR} \\
i_{qS} \\
i_{qR}
\end{bmatrix}
\]

Note the machine is symmetric (there is no saliency), and since we are referred to the stator, the stator and rotor self-inductances include leakage terms:

\[ L_S = M + L_{St} \]
\[ L_R = M + L_{Rd} \]

The voltage equations are:

\[ v_{dS} = \frac{d\lambda_{dS}}{dt} - \omega \lambda_{qS} + r_s i_{dS} \]
\[ v_{qS} = \frac{d\lambda_{qS}}{dt} + \omega \lambda_{dS} + r_s i_{qS} \]
\[ 0 = \frac{d\lambda_{dR}}{dt} - \omega_s \lambda_{qR} + r_R i_{dR} \]
\[ 0 = \frac{d\lambda_{qR}}{dt} + \omega_s \lambda_{dR} + r_R i_{qR} \]

Note that both rotor and stator have “speed” voltage terms since they are both rotating with respect to the rotating coordinate system. The speed of the rotating coordinate system is \( \omega \) with respect to the stator. With respect to the rotor that speed is \( \omega_s \), where \( \omega_m \) is the rotor mechanical speed. Note that this analysis does not require that the reference frame coordinate system speed \( \omega \) be constant.

Torque is given by:

\[ T^e = \frac{3}{2} p (\lambda_{dS} i_{qS} - \lambda_{qS} i_{dS}) \]

### 2.1 Simulation Model

As a first step in developing a simulation model, see that the inversion of the flux-current relationship is (we use the d- axis since the q- axis is identical):

\[ i_{dS} = \frac{L_R}{L_S L_R - M^2} \lambda_{dS} - \frac{M}{L_S L_R - M^2} \lambda_{dR} \]
\[ i_{dR} = \frac{M}{L_S L_R - M^2} \lambda_{dS} - \frac{L_S}{L_S L_R - M^2} \lambda_{dR} \]

Now, if we make the following definitions (the motivation for this should by now be obvious):

\[ X_d = \omega_0 L_S \]
\[ X_{kd} = \omega_0 L_R \]
\[ X_{ad} = \omega_0 M \]
\[ X_d' = \omega_0 \left( L_S - \frac{M^2}{L_R} \right) \]

the currents become:

\[ i_{dS} = \frac{\omega_0}{X_d} \lambda_{dS} - \frac{X_{ad}}{X_{kd} X_d'} \lambda_{dR} \]
\[ i_{dR} = \frac{X_{ad} \omega_0}{X_{kd} X_d} \lambda_{dS} - \frac{X_d \omega_0}{X_d' X_{kd}} \lambda_{dR} \]

The q- axis is the same.
Torque may be, with these calculations for current, written as:

\[ T_e = \frac{3}{2} p (\lambda_{ds} i_{qs} - \lambda_{qs} i_{ds}) - \frac{3}{2} \frac{\omega_0 X_{gd}}{X_{kd} X_d} (\lambda_{ds} \lambda_{qR} - \lambda_{qs} \lambda_{dR}) \]

Note that the usual problems with ordinary variables hold here; the foregoing expression was written assuming the variables are expressed as peak quantities. If RMS is used we must replace 3/2 by 3!

With these, the simulation model is quite straightforward. The state equations are:

\[
\begin{align*}
\frac{d\lambda_{ds}}{dt} &= V_{ds} + \omega \lambda_{qS} - R_S i_{ds} \\
\frac{d\lambda_{qs}}{dt} &= V_{qs} - \omega \lambda_{ds} - R_S i_{qs} \\
\frac{d\lambda_{dR}}{dt} &= \omega \lambda_{qR} - R_R i_{dR} \\
\frac{d\lambda_{qR}}{dt} &= -\omega \lambda_{dR} - R_S i_{qR} \\
\frac{d\Omega_m}{dt} &= \frac{1}{J} (T_e + T_m)
\end{align*}
\]

where the rotor frequency (slip frequency) is:

\[ \omega_s = \omega - p\Omega_m \]

For simple simulations and constant excitation frequency, the choice of coordinate systems is arbitrary, so we can choose something convenient. For example, we might choose to fix the coordinate system to a synchronously rotating frame, so that stator frequency \( \omega = \omega_0 \). In this case, we could pick the stator voltage to lie on one axis or another. A common choice is \( V_d = 0 \) and \( V_q = V \).

3 Control Model

If we are going to turn the machine into a servomotor, we will want to be a bit more sophisticated about our coordinate system. In general, the principle of field-oriented control is much like emulating the function of a DC (commutator) machine. We figure out where the flux is, then inject current to interact most directly with the flux.

As a first step, note that because the two stator flux linkages are the sum of air-gap and leakage flux,

\[
\begin{align*}
\lambda_{ds} &= \lambda_{agd} + L_{St} i_{ds} \\
\lambda_{qs} &= \lambda_{agq} + L_{St} i_{qs}
\end{align*}
\]

This means that we can re-write torque as:

\[ T_e = \frac{3}{2} p (\lambda_{agd} i_{qs} - \lambda_{agq} i_{ds}) \]
Next, note that the rotor flux is, similarly, related to air-gap flux:
\[
\lambda_{agd} = \lambda_{dR} - L_R i_dR \\
\lambda_{agq} = \lambda_{qR} - L_R i_qR
\]

Torque now becomes:
\[
T^e = \frac{3}{2^p} \left( \lambda_{dR} i_qS - \lambda_{qR} i_dS \right) - \frac{3}{2^p} L_R (i_dR i_qS - i_qR i_dS)
\]

Now, since the rotor currents could be written as:
\[
i_dR = \frac{\lambda_{dR}}{L_R} - M \frac{i_dS}{L_R} \\
i_qR = \frac{\lambda_{qR}}{L_R} - M \frac{i_qS}{L_R}
\]

That second term can be written as:
\[
i_dR i_qS - i_qR i_dS = \frac{1}{L_R} (\lambda_{dR} i_qS - \lambda_{qR} i_dS)
\]

So that torque is now:
\[
T^e = \frac{3}{2^p} \left( 1 - \frac{L_R}{L_R} \right) (\lambda_{dR} i_qS - \lambda_{qR} i_dS) = \frac{3}{2^p} M \frac{L_R}{L_R} (\lambda_{dR} i_qS - \lambda_{qR} i_dS)
\]

4 Field-Oriented Strategy:

What is done in field-oriented control is to establish a rotor flux in a known position (usually this position is the d- axis of the transformation) and then put a current on the orthogonal axis (where it will be most effective in producing torque). That is, we will attempt to set
\[
\lambda_{dR} = A_0 \\
\lambda_{qR} = 0
\]

Then torque is produced by applying quadrature-axis current:
\[
T^e = \frac{3}{2^p} M \frac{L_R}{L_R} A_0 i_qS
\]

The process is almost that simple. There are a few details involved in figuring out where the quadrature axis is and how hard to drive the direct axis (magnetizing) current.

Now, suppose we can succeed in putting flux on the right axis, so that \( \lambda_{qR} = 0 \), then the two rotor voltage equations are:
\[
0 = \frac{d\lambda_{dR}}{dt} - \omega_s \lambda_{qR} + r_R i_dR \\
0 = \frac{d\lambda_{qR}}{dt} + \omega_s \lambda_{dR} + r_R i_qR
\]
Now, since the rotor currents are:

\[
i_{dR} = \frac{\lambda_{dR}}{L_R} - \frac{M}{L_R} i_{dS}
\]
\[
i_{qR} = \frac{\lambda_{qR}}{L_R} - \frac{M}{L_R} i_{qS}
\]

The voltage expressions become, accounting for the fact that there is no rotor quadrature axis flux:

\[
0 = \frac{d\lambda_{dR}}{dt} + r_R \left( \frac{\lambda_{dR}}{L_R} - \frac{M}{L_R} i_{dS} \right)
\]
\[
0 = \omega_s \lambda_{dR} - r_R \frac{M}{L_R} i_{qS}
\]

Noting that the rotor time constant is

\[
T_R = \frac{L_R}{r_R}
\]

we find:

\[
T_R \frac{d\lambda_{dR}}{dt} + \lambda_{dR} = M i_{dS}
\]
\[
\omega_s = \frac{M}{T_R} \frac{i_{qS}}{\lambda_{dR}}
\]

The first of these two expressions describes the behavior of the direct-axis flux: as one would think, it has a simple first-order relationship with direct-axis stator current. The second expression, which describes slip as a function of quadrature axis current and direct axis flux, actually describes how fast to turn the rotating coordinate system to hold flux on the direct axis.

Now, a real machine application involves phase currents \(i_a\), \(i_b\), and \(i_c\), and these must be derived from the model currents \(i_{dS}\) and \(i_{qS}\). This is done with, of course, a mathematical operation which uses a transformation angle \(\theta\). And that angle is derived from the rotor mechanical speed and computed slip:

\[
\theta = \int (p\omega_m + \omega_s) \, dt
\]

A generally good strategy to make this sort of system work is to measure the three phase currents and derive the direct- and quadrature-axis currents from them. A good estimate of direct-axis flux is made by running direct-axis flux through a first-order filter. The tricky operation involves dividing quadrature axis current by direct axis flux to get slip, but this is now easily done numerically (as are the trigonometric operations required for the rotating coordinate system transformation). An elementary block diagram of a (possibly) plausible scheme for this is shown in Figure 1.

In this picture we start with commanded values of direct- and quadrature-axis currents, corresponding to flux and torque, respectively. These are translated by a rotating coordinate transformation into commanded phase currents. That transformation (simply the inverse Park’s transform) uses the angle \(q\) derived as part of the scheme. In some (cheap) implementations of this scheme the commanded currents are used rather than the measured currents to establish the flux and slip.

We have shown the commanded currents \(i^*_a\), etc. as inputs to an “Amplifier”. This might be implemented as a PWM current-source, for example, and a tight loop here results in a rather high performance servo system.
Figure 1: Field Oriented Controller