Massachusetts Institute of Technology
**Department of Electrical Engineering and Computer Science**
6.685 Electric Machines

Problem Set 12 Solutions Issued December 4, 2005

1. The first order model is the 'steady state' model, which assumes that the electrical states reach steady state much more quickly than the mechanical state (speed). Slip is $s = 1 - \frac{p\omega_m}{\omega_0}$, where we are using $\omega_m$ as the actual rotor speed. Then we can build up terminal impedance by:

$$\begin{aligned} Z_r &= jX_2 + \frac{R_2}{s} \\ Z_{ag} &= Z_r || jX_m \\ Z_t &= R_1 + jX_1 + Z_{ag} \end{aligned}$$

Terminal current is $I_a = \frac{V_t}{Z_t}$ and rotor branch current is found using a divider relationship:

$$I_2 = -\frac{jX_m}{jX_m + Z_r} I_a$$

If we take voltage (and consequently current) to be *peak*, torque is simply:

$$T_e = \frac{p}{\omega_0} P_{ag} = \frac{3}{2}\frac{p}{\omega_0}\frac{R_2}{s}|I_s|^2$$

We model load torque as:

$$T_\ell = T_0 \left(\frac{p\omega_m}{\omega_0}\right)^2$$

and the single state equation is:

$$\frac{d\omega_m}{dt} = \frac{1}{J}\left(T_e - T_\ell\right)$$

To get input power we simply multiply voltage by current:

$$P_{in} = \mathrm{Re}\left\{V_t I_a^*\right\}$$

This is implemented in the first set of scripts, appended and the results for this motor are shown in Figures 1 and 2.

2. For the second and third parts of this problem set we employ the induction motor model cast in terms of a coordinate system rotating at synchrous speed:

$$\begin{aligned} \frac{d\lambda_{ds}}{dt} &= V_{ds} + \omega_e \lambda_{qs} - r_s i_{ds} \\ \frac{d\lambda_{qs}}{dt} &= V_{qs} - \omega_e \lambda_{ds} - r_s i_{qs} \\ \frac{d\lambda_{dr}}{dt} &= \omega_s \lambda_{qr} - r_r i_{dr} \end{aligned}$$

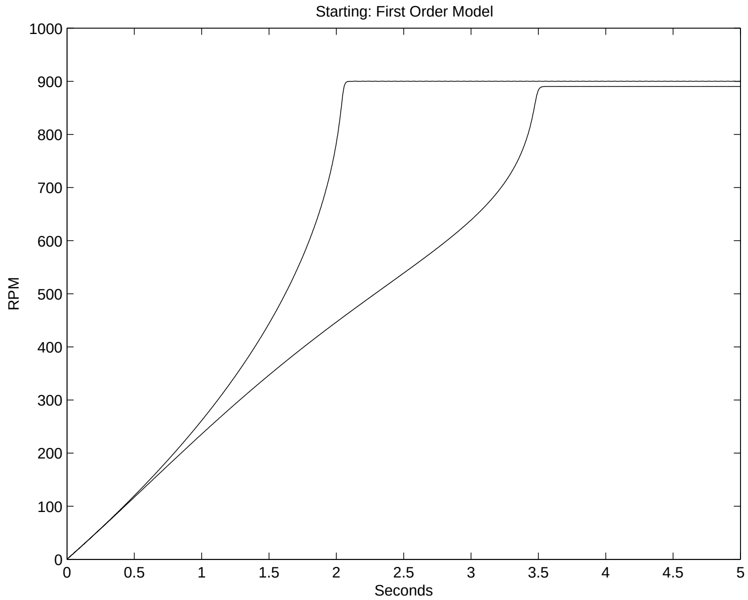


Figure 1: Starting Speed: First Order Model

$$\begin{aligned}
\frac{d\lambda_{qr}}{dt} &= -\omega_s \lambda_{dr} - r_r i_{qr} \\
T_e &= \frac{3}{2} p \left( \lambda_{ds} i_{qs} - \lambda_{qs} i_{ds} \right) \\
\frac{d\omega_m}{dt} &= \frac{1}{J} \left( T_e - T_\ell \right)
\end{aligned}$$

Our formulation for load torque $T_\ell$ is the same for these parts of the problem as for the first part. Now, we need to find the currents for both rotor and stator. Since the rotor is 'round' the quadrature axis behaves exactly like the stator. The flux/current relationship is:

$$\begin{bmatrix} \lambda_{ds} \\ \lambda_{dr} \end{bmatrix} = \begin{bmatrix} L_s & M \\ M & L_r \end{bmatrix} \begin{bmatrix} i_{ds} \\ i_{dr} \end{bmatrix}$$

Those inductances are:

$$\begin{aligned}
L_s &= \frac{X_1 + X_m}{\omega_0} \\
L_r &= \frac{X_2 + X_m}{\omega_0} \\
M &= \frac{X_m}{\omega_0}
\end{aligned}$$

Inverting that matrix yields the current/flux relationship:

$$\begin{aligned}
i_{ds} &= y_{11} \lambda_{ds} + y_{12} \lambda_{dr} \\
i_{dr} &= y_{12} \lambda_{ds} + y_{22} \lambda_{dr}
\end{aligned}$$

For the third order model we assume that the stator is essentially in steady state and that resistance is small enough. So that

$$\lambda_{ds} = V_q$$

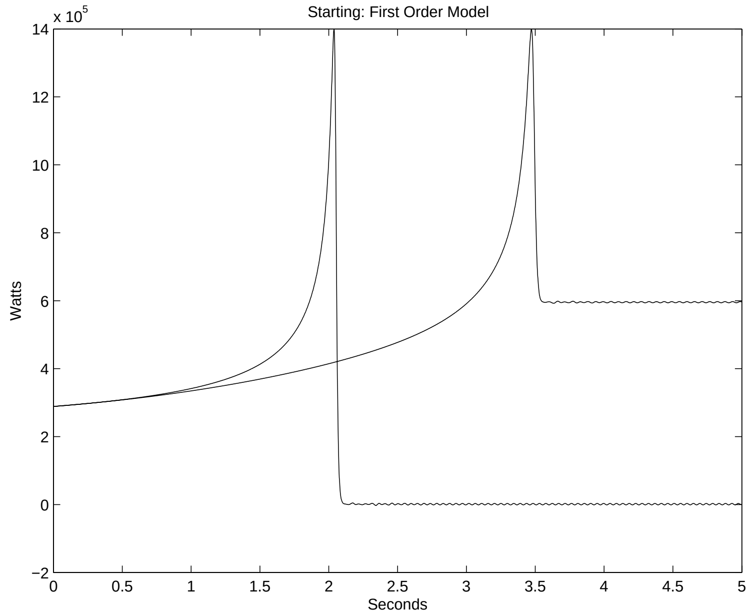


Figure 2: Starting Power: First Order Model

$$\lambda_{qs} = -V_d$$

Simulation of this case uses only the last three of the state equations. For our purposes here we assume that the terminal voltage is on the q- axis: $V_q = V_t$ and $V_d = 0$. For this problem, terminal voltage was stated as 4160 volts, rms, line-line. Phase voltage, peak is then:

$$V_t = \sqrt{\frac{2}{3}} \times 4160 \approx 3397V$$

At the end we can calculate input power by multiplying terminal voltage times current, but for this case we need to add back in the resistive loss:

$$P_{in} = \frac{3}{2} V_t i_{qs} + R_1 \left( i_{qs}^2 + i_{ds}^2 \right)$$

The third order model is implemented in two scripts which are appended. The results are shown in Figures 3 and 4

3. The fifth order model is even easier because we use the simulation model already stated directly. Power is simply voltage times current. The implementation of this in two scripts is appended and the results are shown in Figure 5 and 6.

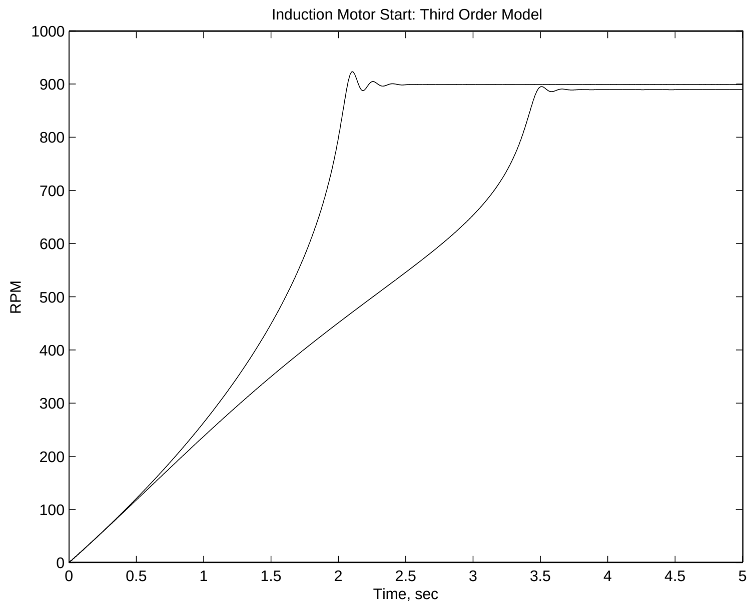


Figure 3: Starting Speed: Third Order Model

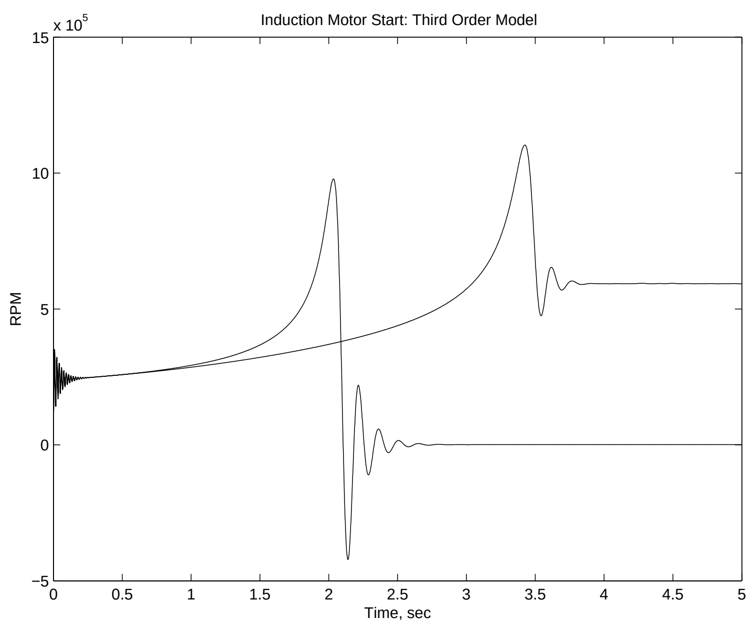


Figure 4: Starting Power: Third Order Model

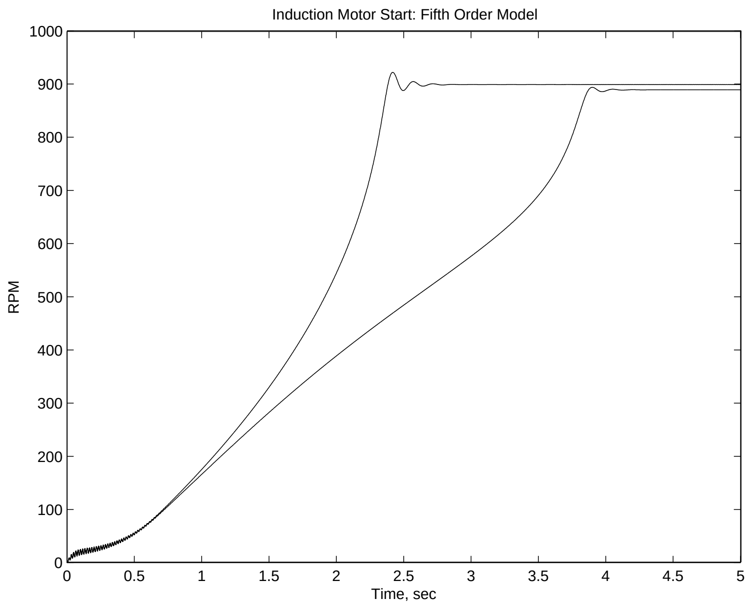


Figure 5: Starting Speed: Fifth Order Model

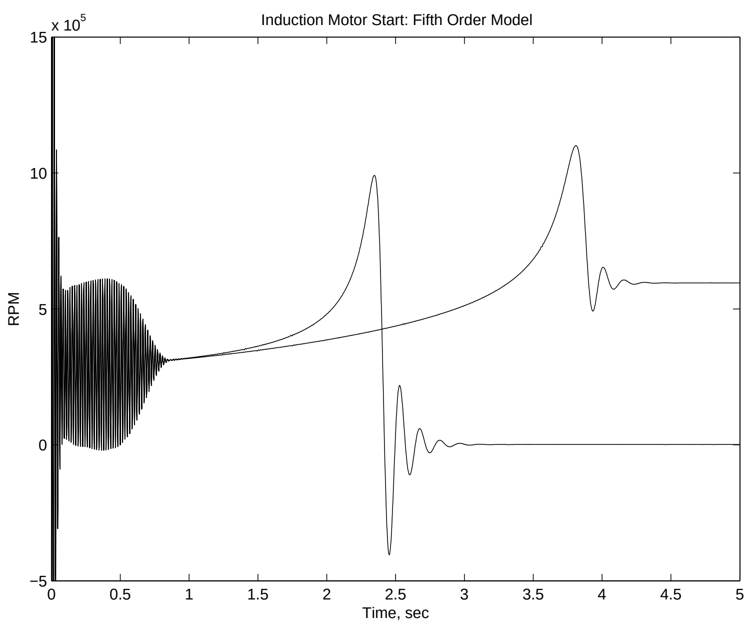


Figure 6: Starting Power: Fifth Order Model

```
% Sample Parameters for Induction Motor Starting Scripts
% Representative Parameters of a large 4160 v motor
p = 4;
omz = 120*pi;
vt = sqrt(2/3)*4160;
r1 = .295;
r2 = .277;
x1 = 2.61;
x2 = 3.24;
xad = 61.3;
J = 60;
T_0 = (p/omz)*6e5;

% 6.685 Problem Set 12, Problem 1
% First Order Model: Induction Motor Acceleration

global x1 x2 xad r1 r2 J p omz vt T_l
bparms

x0=[0];          % start from rest
T_l = 0;                       % unloaded start
tspan = 0:.001:5;
options = odeset('reltol',3e-5,'abstol',3e-6);

[t,x] = ode23('im1',tspan,x0, options);    % this is the simulator
T_l =T_0;                      % start under load
[tl, xl] = ode23('im1', tspan, x0, options);
rpm = (60/(2*pi)) .* x; % just translates units
rpml = (60/(2*pi)) .*xl;
figure(1)
clf
plot(t, rpm, tl, rpml); % draw picture
xlabel('Seconds');
ylabel('RPM');
title('Starting: First Order Model');

% OK: now we compute terminal current
omm = p .* x;
omml = p .* xl;
s = 1 - omm ./ omz;
sl = 1 - omml ./ omz;
Zr = j*x2 + r2 ./ s;
Zrl = j*x2 + r2 ./ sl;
Zt = r1 + j*x1 + j*xad .* Zr ./ (j*xad + Zr);
Ztl = r1 + j*x1 + j*xad .* Zrl ./ (j*xad + Zrl);
P = 1.5 .* real(vt^2 ./ Zt);
Pl = 1.5 .* real(vt^2 ./ Ztl);
```

```
figure(2)
clf
plot(t, P, tl, Pl)
xlabel('Seconds')
ylabel('Watts')
title('Starting: First Order Model')

function xdot = im1(t, x)        % First Order Acceleration

global x1 x2 xad r1 r2 J p omz vt T_l

omm = x;           % rotational frequency
s = 1-p*omm/omz;       % per-unit slip
zr = r2/s + j*x2;     % rotor impedance
zm = j*xad;          % magnetizing impedance
zag = zm*zr/(zm+zr);      % looking across the air-gap
zt = zag + j*x1 + r1;      % terminal impedance
ia = vt / zt;          % terminal current
ir = ia * zm / (zm+zr);      % rotor current
pag = 1.5*abs(ir)^2 * r2/s;      % air-gap power
Tl = T_l*(p*omm/omz)^2;         % load torque
xdot = ((pag/omz) * p -Tl)/ J; % torque over inertia
```

```
% 6.685 Problem Set 12, Part B
% Implementation of third order induction motor model
% simulation of across-the-line starting
global vt omz y11 y12 y22 r1 r2 p J T_l

bparms              % go get parameters

% first, generate admittances
Ls = (x1+xad)/omz;
Lr = (x2+xad)/omz;
M = xad/omz;

y11 = Lr/(Ls*Lr-M^2);
y22 = Ls/(Ls*Lr-M^2);
y12 = -M/(Ls*Lr-M^2);

% first simulated unloaded start
T_l = 0;
tspan = 0:.001:5;
X0 = [0 0 0]';
[t, X] = ode23('im3', tspan, X0);
omm = X(:,3);
rpm = (30/pi) .* omm;
iq = y12 .* X(:,2);                 % quadrature axis current
id = y11 * vt/omz + y12 .* X(:,1);      % direct axis current
Pin = 1.5*vt .* iq + r1 .* (id .^2 + iq .*2);
% now simulate loaded
T_l = T_0;
[tl, X] = ode23('im3', tspan, X0);
omml = X(:,3);
rpml = (30/pi) .* omml;
iq = y12 .* X(:,2);                 % quadrature axis current
id = y11 * vt/omz + y12 .* X(:,1);      % direct axis current
Pinl = 1.5*vt .* iq + r1 .* (id .^2 + iq .^2);

figure(3)
plot(t, rpm, tl, rpml)
title('Induction Motor Start: Third Order Model')
ylabel('RPM')
xlabel('Time, sec')

figure(4)
plot(t, Pin, tl, Pinl)
title('Induction Motor Start: Third Order Model')
ylabel('RPM')
xlabel('Time, sec')
axis([0 5 -5e5 1.5e6])
```

```
function  xdot  =  im3(t,  x)

global  vt  omz  y11  y12  y22  r1  r2  p  J  T_l

lamdad  =  vt/omz;
lamdaq  =  0;
lamdadr  =  x(1);
lamdaqr  =  x(2);
omm  =  x(3);

omme  =  p*omm;
oms  =  omz-omme;

Tl  =  T_l*(omme/omz)^2;

ids  =  y11*lamdad  +  y12*lamdadr;
iqs  =  y11*lamdaq  +  y12*lamdaqr;
idr  =  y12*lamdad  +  y22*lamdadr;
iqr  =  y12*lamdaq  +  y22*lamdaqr;

dlamdadr  =  oms*lamdaqr  - r2*idr;
dlamdaqr  =  -oms*lamdadr  - r2*iqr;
domm  =  (1.5*p/J)*(lamdad  *  iqs  - lamdaq  *  ids)-Tl/J;

xdot  =  [dlamdadr  dlamdaqr  domm]';
```

```
% 6.685 Problem Set 12, Part C
% Implementation of fifth order induction motor model
% simulation of across-the-line starting
global vt omz y11 y12 y22 r1 r2 p J T_l

bparms              % go get parameters

% first, generate admittances
Ls = (x1+xad)/omz;
Lr = (x2+xad)/omz;
M = xad/omz;

y11 = Lr/(Ls*Lr-M^2);
y22 = Ls/(Ls*Lr-M^2);
y12 = -M/(Ls*Lr-M^2);

% first simulated unloaded start
T_l = 0;
tspan = 0:.001:5;
X0 = [0 0 0 0 0]';
[t, X] = ode23('im5', tspan, X0);
omm = X(:,5);
rpm = (30/pi) .* omm;
iq = y11 .* X(:,2) + y12 .* X(:,4);
Pin = 1.5*vt .* iq;
% now simulate loaded
T_l = T_0;
[tl, X] = ode23('im5', tspan, X0);
omml = X(:,5);
rpml = (30/pi) .* omml;
iq = y11 .* X(:,2) + y12 .* X(:,4);
Pinl = 1.5*vt .* iq;

figure(5)
plot(t, rpm, tl, rpml)
title('Induction Motor Start: Fifth Order Model')
ylabel('RPM')
xlabel('Time, sec')

figure(6)
plot(t, Pin, tl, Pinl)
title('Induction Motor Start: Fifth Order Model')
ylabel('RPM')
xlabel('Time, sec')
axis([0 5 -5e5 1.5e6])

function xdot = im5(t, x)
```

```
global  vt  omz  y11  y12  y22  r1  r2  p  J  T_l

lamdad  =  x(1);
lamdaq  =  x(2);
lamdadr  =  x(3);
lamdaqr  =  x(4);
omm  =  x(5);

omme  =  p*omm;
oms  =  omz-omme;

Tl  =  T_l*(omme/omz)^2;

ids  =  y11*lamdad  +  y12*lamdadr;
iqs  =  y11*lamdaq  +  y12*lamdaqr;
idr  =  y12*lamdad  +  y22*lamdadr;
iqr  =  y12*lamdaq  +  y22*lamdaqr;

dlamdad  =  omz*lamdaq  - r1*ids;
dlamdaq  =  vt  - omz*lamdad  - r1*iqs;
dlamdadr  =  oms*lamdaqr  - r2*idr;
dlamdaqr  =  -oms*lamdadr  - r2*iqr;
domm  =  (1.5*p/J)*(lamdad  *  iqs  - lamdaq  *  ids)-Tl/J;

xdot  =  [dlamdad  dlamdaq  dlamdadr  dlamdaqr  domm]';
```