

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

Problem Set 11 Solutions

November 27, 2011

Problem 1: Variable Reluctance Motor

Much of the solution to this problem is keeping track of where the inductance is varying. Note for some value of flux λ , if $\frac{\lambda}{L(\theta)} < I_s$, then:

$$\begin{aligned} I &= \frac{\lambda}{L(\theta)} \\ W'_m &= \frac{1}{2}L(\theta)I^2 \\ T_e &= \frac{I^2}{2} \frac{\partial L}{\partial \theta} \end{aligned}$$

If the flux is greater than that:

$$\begin{aligned} I &= \frac{\lambda}{L_{\min}} + I_s \frac{L(\theta) - L_{\min}}{L_{\min}} \\ W'_m &= \frac{1}{2}I_s^2 L_{\min} + (L(\theta) - L_{\min}) I_s I \\ T_e &= II_s \frac{\partial L}{\partial \theta} \end{aligned}$$

The details are shown in the scripts attached. I do not presume to present these as masterpieces of slick programming, but they seem to work. A subroutine is used to present inductance and rate of change of inductance and then a second subroutine calculates current and torque, given flux, inductance and rate of change of inductance. Here are some pictures. First, Figure 1 shows the (unsaturated) inductance and rate of change of inductance as a function of rotor angle, for one stator pole.

Next, Figure 2 shows current as a function of rotor position for several values of flux, from 0.2 T to 2.0 T in increments of 0.2 T. This is computed easily as:

$$\begin{aligned} i &= \frac{\lambda}{L(\theta)} && \text{if } \frac{\lambda}{L(\theta)} < I_{\text{sat}} \\ &= I_{\text{sat}} \left(1 - \frac{L(\theta)}{L_{\min}} \right) + \frac{\lambda}{L_{\min}} && \text{otherwise} \end{aligned}$$

To obtain torque, we note that the magnetic coenergy is:

$$\begin{aligned} w'_m &= \frac{1}{2}L(\theta)i^2 && \text{if } \frac{\lambda}{L(\theta)} < I_{\text{sat}} \\ &= \frac{1}{2}L(\theta)I_{\text{sat}}^2 + L(\theta)(i - I_{\text{sat}}) && \text{otherwise} \end{aligned}$$

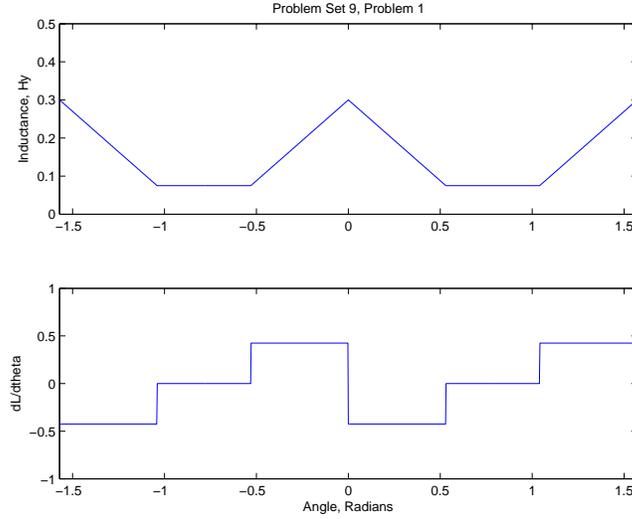


Figure 1: Inductance and rate of change of inductance vs. angle

Then torque is:

$$\begin{aligned}
 T^e &= \frac{i^2}{2} \frac{\partial L}{\partial \theta} && \text{if } \frac{\lambda}{L(\theta)} < I_{\text{sat}} \\
 &= \left(I_{\text{sat}} i - \frac{1}{2} I_{\text{sat}}^2 \right) \frac{\partial L}{\partial \theta} && \text{otherwise}
 \end{aligned}$$

This is shown in Figure 3

To find average torque note there are three phases and four rotor poles, so there are $3 \times 4 = 12$ cycles per revolution. Thus

$$\langle T_e \rangle = \frac{12}{2\pi} \times W$$

where W is work done per 'stroke'. We approximate this by:

$$W = \int T_e d\theta \approx \sum T_e \Delta\theta$$

and we have taken $\Delta\theta$ to be a suitably small value. Note we have used only the duration of one flux pulse to calculate this work.

Finally, we have picked out two starting angles, one in the motoring range and one in the generating range and have plotted instantaneous flux, current and torque. Note that current in the winding is a positive number, but, particularly in generating operation, current may be freewheeling through the diodes back into the power supply, and so this current can appear to be negative to the supply.

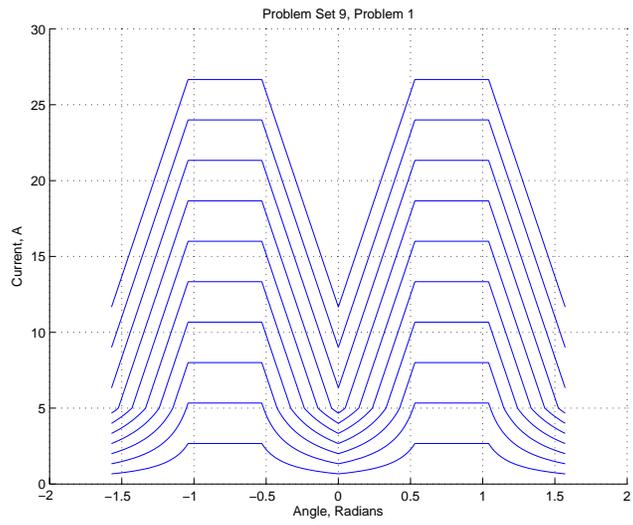


Figure 2: Current vs. position with flux as a parameter

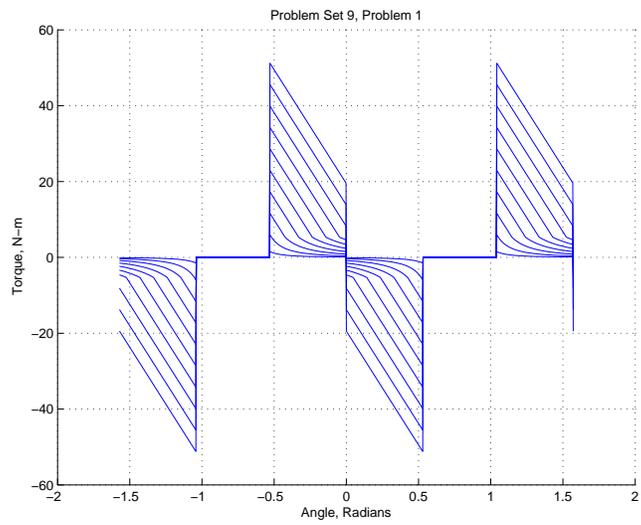


Figure 3: Torque vs. position with flux as a parameter

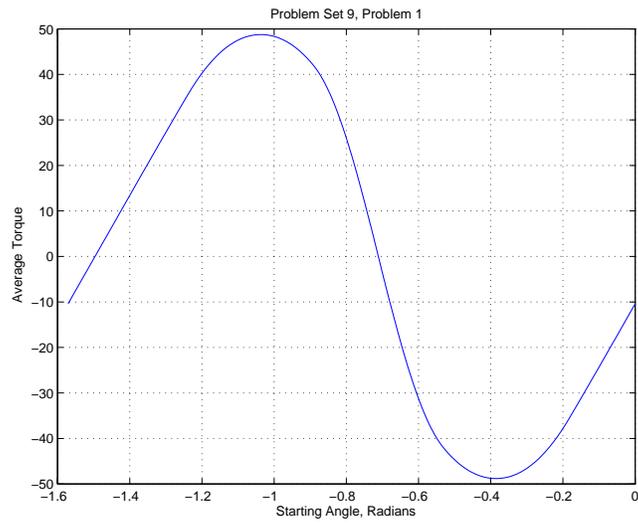


Figure 4: Average torque as a function of pulse start angle

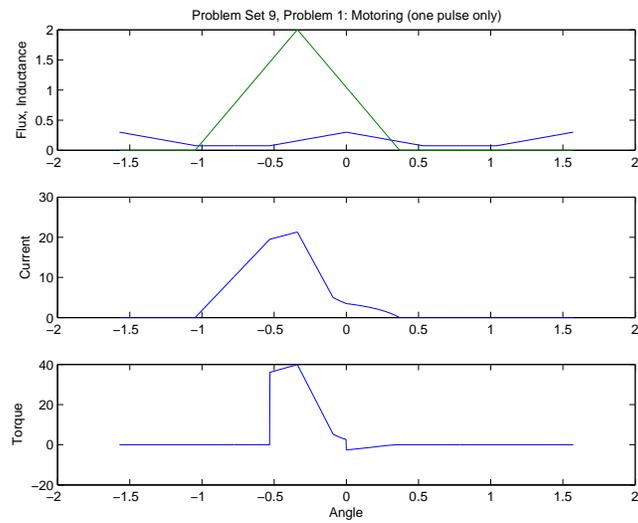


Figure 5: Motoring Operation

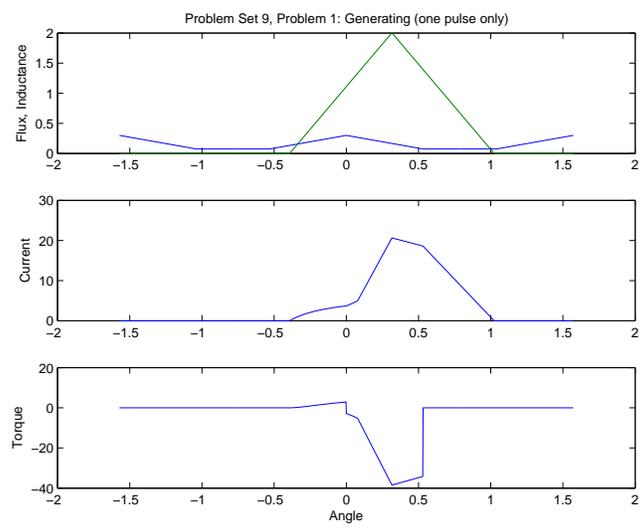


Figure 6: Generating Operation

Problem 2: Damping

The objective here is to understand how to linearize the synchronous machine model, which is decidedly nonlinear (but not pathologically so). We seek a model that looks like this:

$$\frac{2H}{\omega_0} \frac{d^2\delta}{dt^2} + B \frac{d\delta}{dt} = T_e + T_m$$

considering only damping from the quadrature axis damper winding. If the torque angle is doing this:

$$\delta = \delta_0 \text{Re} \left\{ \underline{\delta}_1 e^{j\Omega t} \right\}$$

then a damping torque would be produced:

$$-B \frac{d\delta}{dt} = \text{Re} \left\{ -j\Omega B \underline{\delta}_1 e^{j\Omega t} \right\}$$

The damper in question has the following state equation:

$$\frac{d\psi_{kq}}{dt} = -\omega_0 r_{kq} i_{kq}$$

The quadrature axis armature current and q-axis damper currents are:

$$\begin{aligned} i_q &= \frac{x_{kq}}{x_q x_{kq} - x_{aq}^2} \psi_q - \frac{x_{aq}}{x_q x_{kq} - x_{aq}^2} \psi_{kq} \\ i_{kq} &= -\frac{x_{aq}}{x_q x_{kq} - x_{aq}^2} \psi_q - \frac{x_q}{x_q x_{kq} - x_{aq}^2} \psi_{kq} \end{aligned}$$

A little bit of manipulation is required to coerce the first of these into:

$$i_q = \frac{\psi_q + e'_d}{x'_q}$$

where

$$x'_q = \frac{x_q x_{kq} - x_{aq}^2}{x_{kq}}$$

Then, using the same notation,

$$i_{kq} = \frac{x_{aq}}{x_{kq}} \frac{\psi_q}{x'_q} + \frac{x_q}{x_{kq}} \frac{\psi_{kq}}{x'_q}$$

if torque angle is varying as postulated above, the variation in flux is:

$$\underline{\psi}_{q1} = -v \sin \delta_0 \underline{\delta}_1$$

The differential equation for damper flux becomes:

$$\frac{d\psi_{kq1}}{dt} + \omega_0 \frac{r_{kq}}{x_{kq}} \frac{x_q}{x_{kq}} \psi_{kq1} = \omega_0 \frac{r_{kq}}{x_{kq}} \frac{x_{aq}}{x'_q} \psi_{q1}$$

Recognizing the damper time constant:

$$T_{kq} = T'_{qo} = \frac{x_{kq}}{\omega_0 r_{kq}}$$

The q- axis short circuit time constant is:

$$T'_q = T'_{qo} \frac{x'_q}{x_q}$$

The damper winding differential equation becomes, using complex notation:

$$(j\Omega T'_q + 1) \psi_{kq1} = \frac{x_{aq}}{x_q} \psi_{q1}$$

This means that the voltage behind q- axis reactance is:

$$e'_{d1} = -\frac{x_{aq}}{x_{kq}} \psi_{kq1} = -\frac{x_{aq}^2}{x_q x_{kq}} \frac{\psi_{q1}}{1 + j\Omega T'_q}$$

The quadrature axis current variation is:

$$i_{q1} = \frac{\psi_{q1} + e'_{dq}}{x'_q} = \frac{\psi_{q1}}{x'_q} \left(1 - \frac{x_{aq}^2}{x_q x_{kq}} \frac{\psi_{q1}}{1 + j\Omega T'_q} \right)$$

The first-order torque variation (at least the part that has damping in it will be:

$$T_1 = \frac{1}{2} \text{Re} \left\{ \psi_{d1} i_{q1}^* \right\} = \frac{v^2 \cos^2 \delta_0}{2} \frac{1}{x'_q} \text{Re} \left\{ 1 - \frac{x_{aq}^2}{x_q x_{kq}} \frac{\psi_{q1}}{1 + j\Omega T'_q} \right\}$$

Matching terms, we get the equivalent of the damping coefficient:

$$\Omega B = \frac{1}{2} \frac{v^2 \cos^2 \delta_0}{x'_q} \left(\frac{x_{aq}^2}{x_q x_{kq}} \frac{\Omega T'_q}{1 + (\Omega T'_q)^2} \right)$$

This at least has the right sign, indicating that there is positive damping, but it does not have the right frequency dependence. It works only for values of $\Omega T'_q < 1$.

Problem 3: Single Phase Motor

Essentially all of the story is told in the code here. I have set the motor up in the same way for both running and starting conditions. As in class, the starting condition involves this matrix equation:

$$\begin{bmatrix} V \\ V \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} j(x_1 + x_\phi) & 0 & \frac{jx_\phi}{2} & \frac{jx_\phi}{2} \\ 0 & Z_e + j(x_1 + x_\phi) & \frac{-x_\phi}{2} & \frac{x_\phi}{2} \\ \frac{jx_\phi}{2} & \frac{x_\phi}{2} & \frac{jx_\phi}{2} + \frac{r_2}{s} & 0 \\ \frac{jx_\phi}{2} & \frac{-x_\phi}{2} & 0 & \frac{jx_\phi}{2} + \frac{r_2}{2-s} \end{bmatrix} \begin{bmatrix} I_a \\ I_b \\ I_F \\ I_R \end{bmatrix}$$

The solution consists of evaluating this equation for the full range of slip, asking Matlab to solve it for I_F and I_R and then getting torque:

$$T = \frac{p}{\omega} \left(|I_F|^2 \frac{r_2}{s} - |I_R|^2 \frac{r_2}{2-s} \right)$$

The impedance Z_e is, respectively, that of a capacitance:

$$Z_e = \frac{1}{j\omega c}$$

or simply a resistance.

For the 'running' condition, this same expression can be used, but without the line and column that describe the starting winding. That is, strike the second row and second column of the matrix.

The results are shown in Figure 7

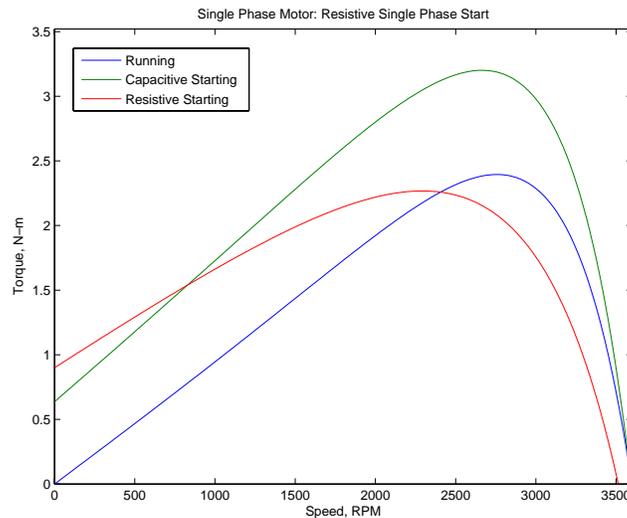


Figure 7: Split Phase Motor Run and Start

Script for Problem 1

```
% 6.685 2013 Problem Set 11, Problem 1
% uses ssrind.m and cur.m

global Ns Nr tho Lmax Lmin Isat

% variables
Ns = 6;           % number of stator poles
Nr = 4;           % number of rotor poles
Lmax = .30;       % maximum inductance
Lmin = .075;      % minimum inductance
tho = 9*pi/20;    % overlap angle
Isat = 5;         % saturation current (Kirtley's trivial model)
lam_peak = 2;     % operating maximum flux

% first, test inductance procedure
dth = pi/1200;    % we will see dth again: a small angle
theta = -pi/2:dth:pi/2;
L = zeros(size(theta));
dL = zeros(size(theta));
for i=1:length(theta),
    [L(i), dL(i)] = srind(theta(i));
end

figure(1)
subplot 211
plot(theta, L)
title('Problem Set 9, Problem 1')
ylabel('Inductance, Hy')
axis([-pi/2 pi/2 0 0.5])
subplot 212
plot(theta, dL)
ylabel('dL/dtheta')
xlabel('Angle, Radians')
axis([-pi/2 pi/2 -1 1])

% now we compute current over a range of fluxes
lambda = 0.2:0.2:2; % given these values of flux

I = zeros(size(theta)); % placeholders
T = zeros(size(theta));

figure(2) % set up figures for the results
clf
```

```

hold on
figure(3)
clf
hold on

for k = 1:length(lambda),
    lam = lambda(k);
    for i = 1:length(theta),
        [I(i), T(i)] = cur(L(i), dL(i), lam);
    end
    figure(2)
    plot(theta, I)
    figure(3)
    plot(theta, T)
end

figure(2)
hold off
title('Problem Set 9, Problem 1')
ylabel('Current, A')
xlabel('Angle, Radians')
grid on

figure(3)
hold off
title('Problem Set 9, Problem 1')
ylabel('Torque, N-m')
xlabel('Angle, Radians')
grid on

% now that we know what is up, we construct a steady state waveform
% we use only one pulse of flux
theta_s = -pi/2:dth:0; % voltage pulse starts at these locations
theta_1 = -tho/2:dth:0; % angles of the voltage pulse: start
theta_2 = 0:dth:tho/2; % and finish
theta_v = [theta_1 theta_2]; % array of angles
lambda_1 = 2*lam_peak/tho .* (theta_1 + tho/2); % flux as a function of angle
lambda_2 = 2*lam_peak/tho .* (tho/2 - theta_2); % resulting from square voltage pulse
lambda_v = [lambda_1 lambda_2]; % this is the flux waveform
Tav = zeros(size(theta_s)); % placeholder for average torque
for it = 1:length(theta_s); % now sweep over starting angle
    Lambda = zeros(size(theta)); % now we must have flux as function of rotor angle
    I_l = zeros(size(theta)); % and current
    T_l = zeros(size(theta)); % and local torque
    for il = 1:length(lambda_v), % this sets up flux waveform

```

```

        Lambda(it+il) = lambda_v(il);           % which is zero outside this range
        [I_l(it+il), T_l(it+il)] = cur(L(it+il), dL(it+il), lambda_v(il)); % bingo: all is d
    end
    Tav(it) = (.5*Ns*Nr/pi)*dth*sum(T_l);      % and here we take the average
end

figure(4)
plot(theta_s, Tav)
title('Problem Set 9, Problem 1')
ylabel('Average Torque')
xlabel('Starting Angle, Radians')
grid on

% then we pick out a motoring and generating case
im = 200;
    Lambda = zeros(size(theta));
    I_l = zeros(size(theta));
    T_l = zeros(size(theta));
    for il = 1:length(lambda_v),
        Lambda(im+il) = lambda_v(il);
        [I_l(im+il), T_l(im+il)] = cur(L(im+il), dL(im+il), lambda_v(il));
    end

figure(5)
subplot 311
plot(theta, L, theta, Lambda)
title('Problem Set 9, Problem 1: Motoring (one pulse only)')
ylabel('Flux, Inductance')
subplot 312
plot(theta, I_l)
ylabel('Current')
subplot 313
plot(theta, T_l)
ylabel('Torque')
xlabel('Angle')

ig = 450;
    Lambda = zeros(size(theta));
    I_l = zeros(size(theta));
    T_l = zeros(size(theta));
    for il = 1:length(lambda_v),
        Lambda(ig+il) = lambda_v(il);
        [I_l(ig+il), T_l(ig+il)] = cur(L(ig+il), dL(ig+il), lambda_v(il));
    end
figure(6)

```

```

subplot 311
plot(theta, L, theta, Lambda)
title('Problem Set 9, Problem 1: Generating (one pulse only)')
ylabel('Flux, Inductance')
subplot 312
plot(theta, I_1)
ylabel('Current')
subplot 313
plot(theta, T_1)
ylabel('Torque')
xlabel('Angle')
-----
function [L, dL] = srmind(th)

% some horrible programming practice
global Ns Nr tho Lmax Lmin Isat
% returns VRM inductance
%Ns (not used) is number of stator poles
%Nr is number of rotor poles
%tho is overlap angle
% Lmax is max inductance
% Lmin is min inductance

thr = 2*pi/Nr; % repetition angle
th1 = tho/2; % Overlap angle
thd = thr/2-th1*(1-Lmin/Lmax); % angle over which L does not vary!
thz = thr/2-th1; % position of zero projection of L
thm = mod(th, thr); % angle with respect to primary
thp = abs(thr/2-thm); % variation angle
if thp<thd, % is the angle where L does not vary?
    L = Lmin; % then we are at mininum inductance
else
    L = Lmax*(thp-thz)/th1; % this works because there is no region of max inductance
end

% now to get derivative
thc1 = th1*(1-Lmin/Lmax); % lower corner
thc2 = thr-thc1; % upper corner

dLdt = Lmax/th1; % inductance slope
if thm < thc1, % where is angle with respect to corners?
    dL = -dLdt; % if less than fiirst corner, slope is negative
elseif thm > thc2, % if greater than second corner it is positive
    dL = dLdt;
else dL = 0; % or in the middle it is constant

```

```
end
-----
function [I,T] = cur(L, dL, lam)
global Ns Nr tho Lmax Lmin Isat

if lam/L < Isat,
    I = lam/L;
    T = .5*I^2 * dL;
else
    I = lam/Lmin - Isat*(L/Lmin-1);
    T = (Isat*I-.5*Isat^2)* dL;
end
```

Script for Problem 3

```
% 6.685 Problem Set 11, Problem 3
% single phase motor: starting and running

xm = 98;           % magnetizing reactance
x1 = 1.25;        % stator leakage
x2 = 0.75;        % rotor leakage
r1 = 2.0;         % stator resistance
r2 = 1.20;        % rotor resistance

v = 120;          % RMS stator voltage
p = 1;           % pole pairs
om = 2*pi*60;    % radian frequency
C = 50e-6;       % starting capacitor
Ze = 1/(j*om*C); % impedance of starting capacitor
zer = 10;        % split phase resistance for starting

% first, do the run case

S = logspace(-3, 0, 500);
N = (om*60/(2*pi)) .* (1-S);

Tr = zeros(size(S));
If = zeros(size(S));
Ir = zeros(size(S));
% running torque calculation
for k = 1:length(S)
    s = S(k);
    Ar = [j*x1+j*xm+r1 .5*j*xm .5*j*xm;
          .5*j*xm j*x2+.5*j*xm+r2/s 0;
          .5*j*xm 0 j*x2+.5*j*xm+r2/(2-s)];
    V = [v 0 0]';
    I = Ar\V;
    If(k) = I(2);
    Ir(k) = I(3);
    Tr(k) = (p/om) * (abs(If(k))^2 * r2/s - abs(Ir(k))^2 * r2/(2-s));
end

Ts = zeros(size(S));
Tsr = zeros(size(S));

% starting torque calculation: Capacitor Start
for k = 1:length(S)
    s = S(k);
```

```

As = [j*x1+j*xm+r1 0 .5*j*xm .5*j*xm;
      0 Ze+j*x1+j*xm+r1 -.5*xm .5*xm;
      .5*j*xm .5*xm .5*j*xm+r2/s 0;
      .5*j*xm -.5*xm 0 .5*j*xm+j*x2+r2/(2-s)];
V = [v v 0 0]';
I = As\V;
If(k) = I(3);
Ir(k) = I(4);
Ts(k) = (p/om) * (abs(If(k))^2 * r2/s - abs(Ir(k))^2 * r2/(2-s));
end
maxt = max(max(Tr), max(Ts));
figure(1)
plot(N, Tr, N, Ts)
title('Single Phase Motor')
ylabel('Torque, N-m')
xlabel('Speed, RPM')
axis([0 3600 0 1.1*maxt])
legend('Running', 'Starting')

% starting torque calculation: Resistance Split Phase Start
Zer = 10;

for k = 1:length(S)
s = S(k);
As = [j*x1+j*xm+r1 0 .5*j*xm .5*j*xm;
      0 Zer+j*x1+j*xm+r1 -.5*xm .5*xm;
      .5*j*xm .5*xm .5*j*xm+r2/s 0;
      .5*j*xm -.5*xm 0 .5*j*xm+j*x2+r2/(2-s)];
V = [v v 0 0]';
I = As\V;
If(k) = I(3);
Ir(k) = I(4);
Tsr(k) = (p/om) * (abs(If(k))^2 * r2/s - abs(Ir(k))^2 * r2/(2-s));
end

%maxt = max(max(Tr), max(Ts));

figure(2)
plot(N, Tr, N, Ts, N, Tsr)
title('Single Phase Motor: Resistive Single Phase Start')
ylabel('Torque, N-m')
xlabel('Speed, RPM')
axis([0 3600 0 1.1*maxt])
legend('Running', 'Capacitive Starting', 'Resistive Starting')

```

MIT OpenCourseWare
<http://ocw.mit.edu>

6.685 Electric Machines
Fall 2013

For information about citing these materials or our Terms of Use, visit: <http://ocw.mit.edu/terms>.