Table of Contents

Using ode15s 1
Set up 2 area, 39bus per area case. 1
Set up long term 2
Set up x0/y0 by running one time step of ED to get PGs 2
Set limits for Diffeq Limiter 3
form the load 3
Simulate the steady state 3
do some plots 3

Using ode15s

```
%clc; clear all; close all

global Load_spline
C = psconstants_will;

% ps = case9_ps_lk_perm;
% ps = updateps(ps);
```

Set up 2 area, 39bus per area case.

```
ps = case39_ps_will;
ps = replicate_case_parallel_gencost_change(ps,2);
ps = updateps(ps);
ps.bus(40:end,C.bu.area) = 2;
ps.gov(:,C.gov.R)        = ps.gen(:,C.ge.Pmax).*0.05/ps.baseMVA % reg constant is
ps.mac(:,C.ma.Tg)        = ps.gov(:,C.gov.Tg);
ps.mac(:,C.ma.R)         = ps.gov(:,C.gov.R);

load_buses = ps.bus_i(ps.shunt(:,1));
bus_areas  = ps.bus(load_buses,C.bu.area);
```

```
ps =

      baseMVA: 100
          bus: [78x19 double]
       branch: [94x22 double]
          gen: [28x22 double]
        shunt: [38x10 double]
          mac: [28x15 double]
          gov: [28x9 double]
        areas: []
      gencost: [28x7 double]
    frequency: 60
        bus_i: [239x1 double]
```

Set up long term

```
day_in_s     = 24*60*60; %24hrs*60min/hr*60s/min
fivemin_in_s = 5*60;
day_in_5min  = day_in_s/fivemin_in_s;
tmax=60;
tmin=1;

perc_reg  = 1;
ps0       = ps;
nmacs     = size(ps.gen,1);
n         = size(ps.bus,1);
ix        = get_indices_will(n,nmacs); % index to help us find stuff
ps        = find_areas(ps);
ps        = set_ramp_rates(ps);
```

Set up x0/y0 by running one time step of ED to get PGs

```
initial_load      = ps.shunt(:,C.sh.P);
%timestep_check   = [initial_load,initial_load*1.2,initial_load*0.8, initial_load*
[Pgs_sbs,Rgs_sbs] = Econ_Dispatch_fn(ps,(initial_load),perc_reg);
ps.gen(:,C.ge.Pg) = Pgs_sbs %Use first time step's optimized Pg's for
ps                = dcpf(ps)

% prepare the machine state variables
ps.mac = get_mac_state(ps,'linear');
```

```
Optimization terminated.
Optimization terminated.

ps =

          baseMVA: 100
              bus: [78x19 double]
           branch: [94x22 double]
              gen: [28x22 double]
            shunt: [38x10 double]
              mac: [28x23 double]
              gov: [28x9 double]
            areas: []
          gencost: [28x7 double]
        frequency: 60
            bus_i: [239x1 double]
       tie_lines_T: {[2x1 double]  [0x1 double]}
       tie_lines_F: {[0x1 double]  [2x1 double]}
    bus_tie_locs_T: {[2x1 double]  [0x1 double]}
    bus_tie_locs_F: {[0x1 double]  [2x1 double]}


ps =
```

```
       baseMVA: 100
           bus: [78x19 double]
        branch: [94x22 double]
           gen: [28x22 double]
         shunt: [38x10 double]
           mac: [28x23 double]
           gov: [28x9 double]
         areas: []
       gencost: [28x7 double]
     frequency: 60
         bus_i: [239x1 double]
  tie_lines_T: {[2x1 double]  [0x1 double]}
  tie_lines_F: {[0x1 double]  [2x1 double]}
bus_tie_locs_T: {[2x1 double]  [0x1 double]}
bus_tie_locs_F: {[0x1 double]  [2x1 double]}
            B: [78x78 double]
```

Set limits for Diffeq Limiter

```
ps.gen(:,C.ge.reg_ramp_up)   = Rgs_sbs;
ps.gen(:,C.ge.reg_ramp_down) = -Rgs_sbs;
ps.gov(:,C.gov.LCmax)        = ones(nmacs,1); %include the rest of ps.gov?
ps.gov(:,C.gov.LCmin)        = -ones(nmacs,1);
```

form the load

```
[Load_spline,ps] = Load_Type(4,ps,tmax,bus_areas);
total_load = ppval(Load_spline,0:tmax);
ps         = get_ps_areas_libby(ps,bus_areas,load_buses,total_load);
```

Simulate the steady state

```
[t,theta,delta,omega,Pm,ps] = simgrid_lti_lk_perm(ps,[tmin,tmax],1);
```

```
k =

    0.0060


num_pos_evals =

     0
```

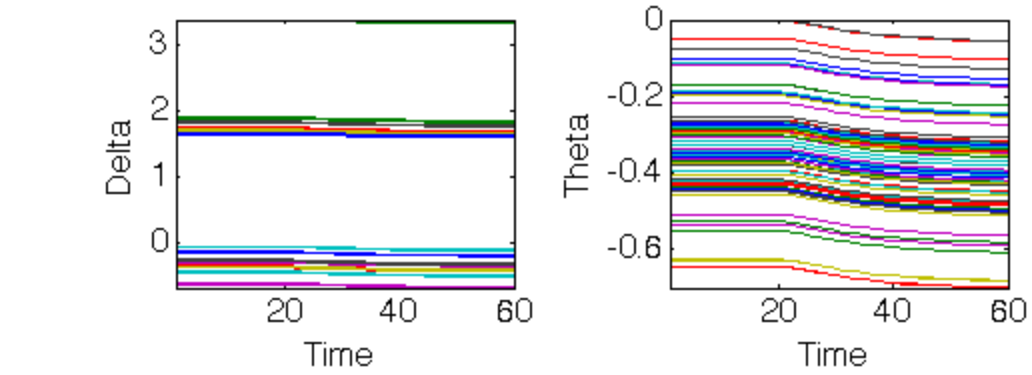
do some plots

```
subplot_row = 2;
```

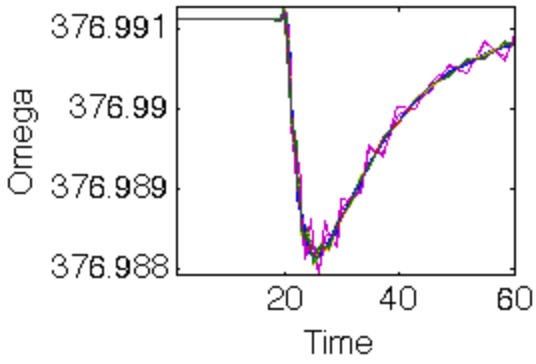
```
subplot_col = 2;
fontsize = 16;

figure(7); clf;
subplot(subplot_row,subplot_col,1)
plot(t,delta);
axis([tmin tmax -Inf Inf])
set(gca,'FontSize',fontsize)
xlabel('Time')
ylabel('Delta')


%figure(2);clf;
subplot(subplot_row,subplot_col,2)
plot(t,theta);
axis([tmin tmax -Inf Inf])
set(gca,'FontSize',fontsize)
xlabel('Time')
ylabel('Theta')

%figure(3);clf
subplot(subplot_row,subplot_col,3)
plot(t,omega);
axis([tmin tmax -Inf Inf])
set(gca,'FontSize',fontsize)
xlabel('Time')
ylabel('Omega')

%figure(4);clf;
subplot(subplot_row,subplot_col,4)
plot(t,Pm);
axis([tmin tmax -Inf Inf])
set(gca,'FontSize',fontsize)
xlabel('Time')
ylabel('Pm')

figure(3);clf;
subplot(3,1,1)
plot(t,Pm(:,1))
set(gca,'FontSize',fontsize)
xlabel('Time')
ylabel('Pm')
subplot(3,1,2)
plot(t,Pm(:,2),'g')
set(gca,'FontSize',fontsize)
xlabel('Time')
ylabel('Pm')
subplot(3,1,3)
plot(t,Pm(:,3),'r')
set(gca,'FontSize',fontsize)
xlabel('Time')
ylabel('Pm')

figure(4);clf;
```

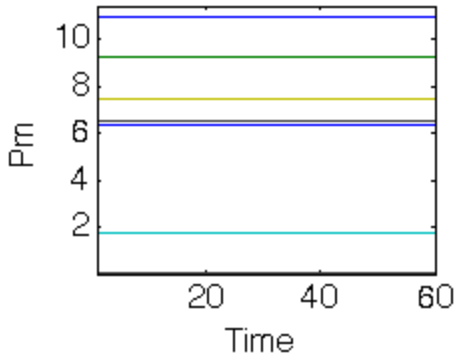
```
%subplot(subplot_row,subplot_col,5)
plot(t, ppval(Load_spline(1),t),'k')
axis([tmin tmax -Inf Inf])
set(gca,'FontSize',fontsize)
xlabel('Time')
ylabel('Load')

figure;clf;
libby=[5 19];
plot(t,omega(:,libby));
axis([tmin tmax -Inf Inf])
set(gca,'FontSize',fontsize)
xlabel('Time')
ylabel('Omega')
title(['K = ',num2str(ps.areas(1,1))])
ylim([376.988,376.992])


ps.gen(libby,:)
```

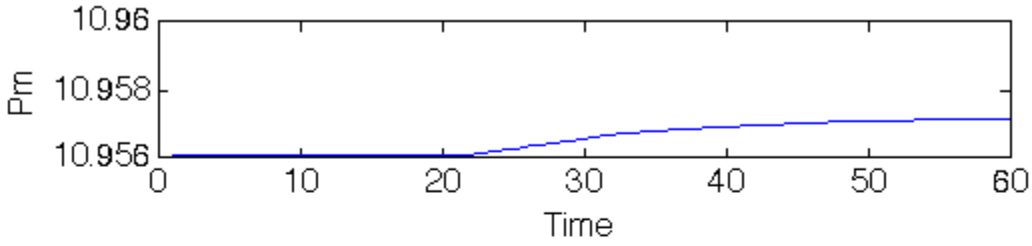
```
ans =

   1.0e+03 *

  Columns 1 through 7

    0.1340    0.0044         0    9.9990   -9.9990    0.0010    0.1000
    0.2340    0.0044         0    9.9990   -9.9990    0.0010    0.1000

  Columns 8 through 14

    0.0010    0.6080         0         0         0         0         0
    0.0010    0.6080         0         0         0         0         0

  Columns 15 through 21

    0.0020    0.0082         0    0.1824   -0.1824    0.0044   -0.0044
    0.0020    0.0082         0    0.1824   -0.1824    0.0044   -0.0044

  Column 22

    0.0030
    0.0030
```

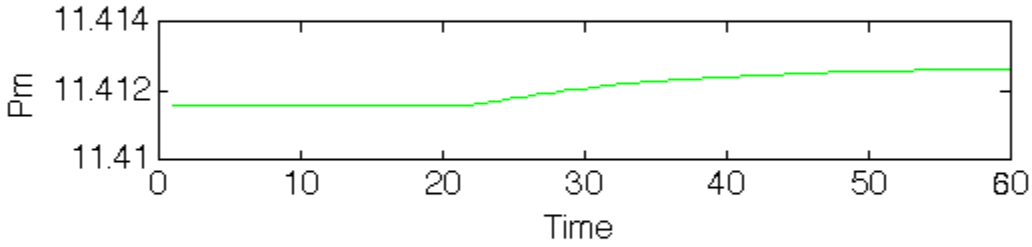

Delta
3
2
1
0
20
40
60
Time

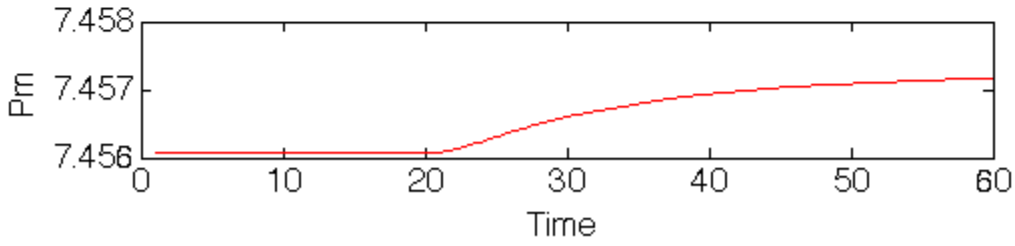
Theta
0
-0.2
-0.4
-0.6
20
40
60
Time

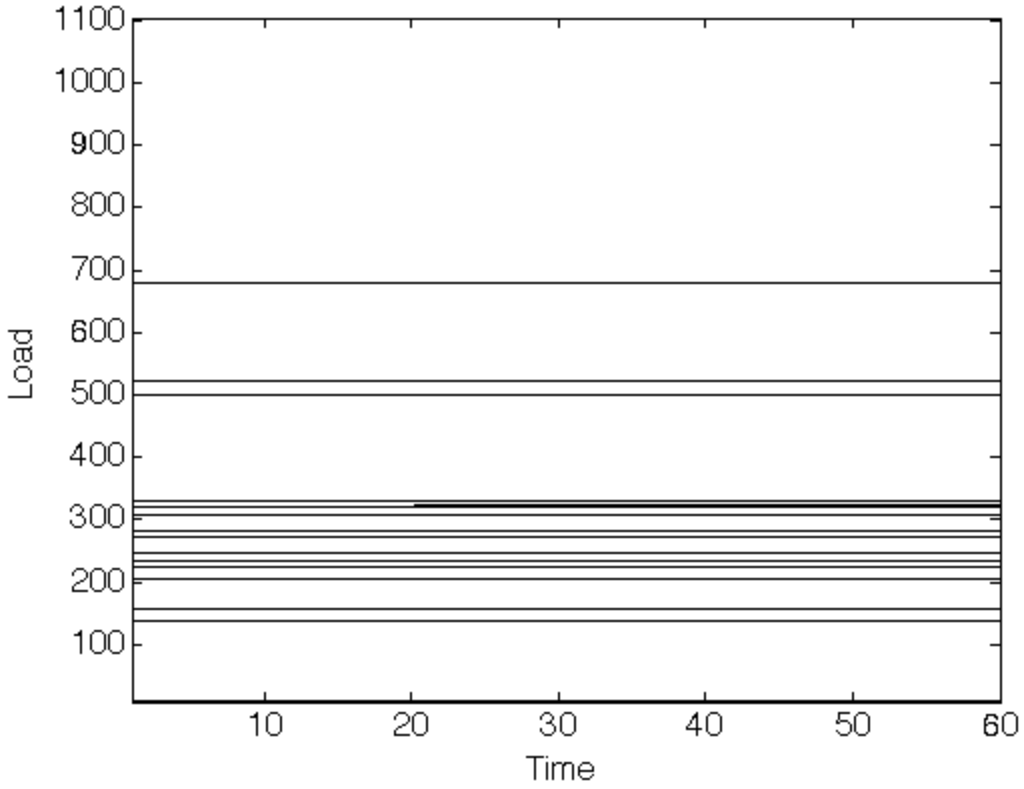

Omega
376.991
376.99
376.989
376.988
20
40
60
Time

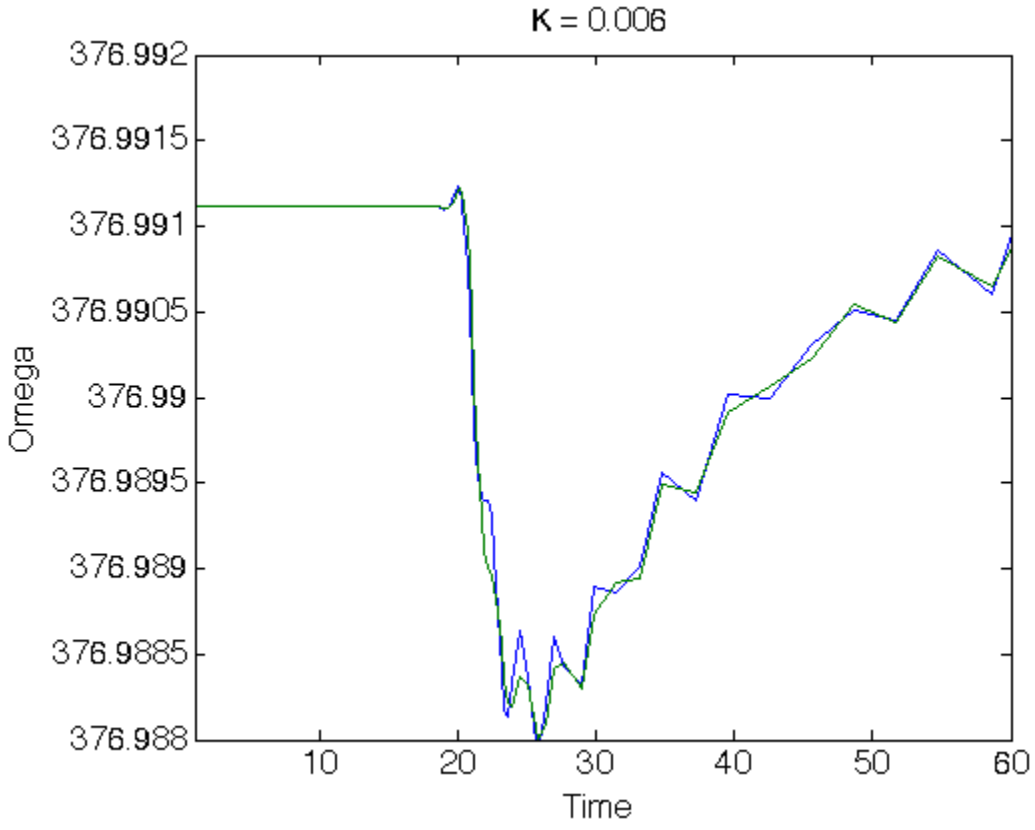

Pm
10
8
6
4
2
20
40
60
Time


Pm
10.96
10.958
10.956
0
10
20
30
40
50
60
Time


Pm
11.414
11.412
11.41
0
10
20
30
40
50
60
Time


Pm
7.458
7.457
7.456
0
10
20
30
40
50
60
Time


1100
1000
900
800
700
600
500
400
300
200
100
Load
10
20
30
40
50
60
Time


K = 0.006
376.992
376.9915
376.991
376.9905
376.99
376.9895
376.989
376.9885
376.988
Omega
10
20
30
40
50
60
Time

Published with MATLAB® R2013b