Programy MATLAB do wykładu 21.XI. 2020

```
% name:RectStar_C15; Type: 3-phase
% star connected (3 Tyr) and controlled by alpha
clear vars;
tRange=[0:0.0001:0.2];
i0=[0.0 0.0 0.0];
global Um; Um=500;
f=50.0;
global omega; omega=2*pi*f;
global phi; phi=0.0;
Ls=0.05; Ld=0.5;
global E; E=120.0;
global Rs; Rs=1.0; global Rod; Rod=40.0;
global iprog; iprog=0.0002;
global Rblock; Rblock=1000;
 A=[Ld+Ls Ld Ld; Ld Ld+Ls Ld; Ld Ld Ld+Ls];
global A1; A1=inv(A);
global alpha; alpha=30/180*pi;
 [tSol,YSol]=ode45(@Rec3,tRange,i0);
 [ID]=[YSol(:,1)+YSol(:,2)+YSol(:,3)];
 plot(tSol,YSol(:,1),'Linewidth',0.5)
 grid on
 hold on
 plot(tSol,YSol(:,2))
 plot(tSol,YSol(:,3))
 plot(tSol,ID,'Linewidth',1.5)
 xlabel('time [s]')
 ylabel('Phase currents & DC current [A]')
 title('3-pulse, star connected rectifier. Alpha=170 deg,  E=-500v')
 legend('iA current','iB current','iC current',...
     'DC current','Location','northwest');
 hold off;
 [UA]=Um*sin(omega*tSol+phi);
 [UB]=Um*sin(omega*tSol+phi-2*pi/3);
 [UC]=Um*sin(omega*tSol+phi+2*pi/3);
 [P1]=YSol(:,1).*(UA)+YSol(:,2).*(UB)+YSol(:,3).*(UC);
 f2=figure;
 plot(tSol,P1,'Linewidth',1.5)
 grid on;
 xlabel('time [s]')
 ylabel('[power [W]')
 title('Source power -> alpha=30 deg,  E=120 V')
 function dYdt=Rec3(t,Y)
global Um; global omega; global phi;
global E; global Rs; global Rod; global iprog;
global Rblock; global A1; global alpha;
function Rez=rezystancja(x)
if x>0.1*iprog Rez=0.0002*Rblock;
elseif x>0.07*iprog Rez=0.05*Rblock;
elseif x>0.05*iprog Rez=0.1*Rblock;
elseif x>0.03*iprog Rez=0.2*Rblock;
elseif x>0.01*iprog Rez=0.5*Rblock;
elseif x>0 Rez=Rblock;
elseif x>-0.01*iprog Rez=2*Rblock;
elseif x>-0.03*iprog Rez=3*Rblock;
elseif x>-0.05*iprog Rez=4*Rblock;
elseif x>-0.07*iprog Rez=5*Rblock;
```

```
elseif x>-0.09*iprog Rez=6*Rblock;
elseif x>-0.11*iprog Rez=8*Rblock;
else Rez=10*Rblock;
end
end
 iA=Y(1); iB=Y(2); iC=Y(3);
 alpha1=alpha+pi/6;
 betaA=omega*t+phi;
 betaB=betaA-2*pi/3;
 betaC=betaA+2*pi/3;
 beta1=mod(betaA,2*pi);
 beta2=mod(betaB,2*pi);
 beta3=mod(betaC,2*pi);
 uA=Um*sin(betaA); uB=Um*sin(betaB);
 uC=Um*sin(betaC);
 RTA=10*Rblock; RTB=10*Rblock; RTC=10*Rblock;
if beta1>alpha1 RTA=rezystancja(iA);
end
if beta2>alpha1 RTB=rezystancja(iB);
end
if beta3>alpha1 RTC=rezystancja(iC);
end
 ps1=uA-E-Rod*(iB+iC)-iA*(RTA+Rod+Rs);
 ps2=uB-E-Rod*(iA+iC)-iB*(RTB+Rod+Rs);
 ps3=uC-E-Rod*(iB+iA)-iC*(RTC+Rod+Rs);
 diAdt=A1(1)*ps1+A1(4)*ps2+A1(7)*ps3;
 diBdt=A1(2)*ps1+A1(5)*ps2+A1(8)*ps3;
 diCdt=A1(3)*ps1+A1(6)*ps2+A1(9)*ps3;
 dYdt=[diAdt;diBdt;diCdt];
end
```

```
% name: RecBridge6_PFLG
% Bridge controlled by alpha
% Load Voltage+E computed
% Power computed
% efficiency computd
% Description included
clearvars;
step=0.000005;
tfin=0.15;
tRange=[0:step:tfin];
Nst=tfin/step+1; Su2=0;
global Um; Um=500;
f=50.0;
global omega; omega=2*pi*f;
global phi; phi=0;
Ls=0.003; Lm=0.001;
global Ld; Ld=0.015;
```

```
global E; E=400;
global Rs; Rs=0.1;
global Rl; Rl=0.6;
global RV; RV=1000000;
global iprog; iprog=0.002;
global Rblock; Rblock=1000;
A=[2*(Lm+Ls) Ls 2*Lm -Lm 2*Lm 0 0; Ls 3*Lm+2*Ls -2*Lm 2*Lm -Lm -Lm 0;...
2*Lm -2*Lm 4*Lm -2*Lm 3*Lm 0 0; -Lm 2*Lm -2*Lm 4*Lm -Lm -2*Lm 0;...
2*Lm -Lm 3*Lm -Lm Ld+3*Lm Ld Ld; 0 -Lm 0 -2*Lm Ld Ld+2*Lm Ld;...
0 0 0 0 Ld Ld Ld];
global A1; A1=inv(A);
global alpha; alpha=20/180*pi;
i0=[0.0 0.0 0.0 0.0 0.0 0.0 0.0];
[tSol,YSol]=ode23s(@Rec6,tRange,i0);
[ID]=[YSol(:,5)+YSol(:,6)+YSol(:,7)];
[iB]=[-YSol(:,1)-YSol(:,2)];
[sAB]=[sin(omega*tRange+phi)];
[sCB]=[sin(omega*tRange+phi+pi/3)];
[Power]=Um*(sAB(:).*YSol(:,1)+sCB(:).*YSol(:,2));
[Uodb]=-RV*YSol(:,7);
[URl]=Rl*ID;
[Uodb2]=URl+E;
[ULd]=[Uodb-Uodb2];
NST=round(Nst);
kp=round(NST/2);
kf=NST;
PowAv=sum(Power(kp:kf))/(kf-kp+1)
IDAv=sum(ID(kp:kf))/(kf-kp+1)
UodbAv=sum(Uodb(kp:kf))/(kf-kp+1)
PowLoad=Rl*IDAv^2+E*IDAv
for k=kp:kf
    Su2=Su2+(ID(k))^2;
end
PowLoad2=Rl*Su2/(kf-kp+1)+E*IDAv
% efficiency load case
%eta1=PowLoad2/PowAv
% efficiency generator case
 eta2=PowAv/(E*IDAv)
plot(tSol,Power/1000,'LineWidth',2)
grid on;
xlabel('time [s]')
ylabel('Power [kW]')
legend('Source Power')
title('Source Power')
figure;
plot(tSol,YSol(:,1))
grid on;
xlabel('time [s]')
ylabel('Currents [A]')
title('Phase Currents & DC Curent')
text(0.01,6.0,'alpha=160 deg; E=-476 V')
hold on
plot(tSol,YSol(:,2))
plot(tSol,iB)
plot(tSol,ID,'Linewidth',2)
hold off;
figure;
plot(tSol,Uodb,'Linewidth',1.5)
grid on;
title('Load voltage & Voltage drop on resistance');
xlabel('time [s]');
```

```
ylabel('voltages [V]');
hold on;
plot(tSol,URl)
plot(tSol,Uodb2,'Linewidth',1.5)
legend('UVolt','URload','URload+E');
hold off;
function dYdt=Rec6(t,Y)
global Um; global omega; global phi;
global E; global Rs; global Rl; global iprog;
global Rblock; global A1; global alpha;
global RV;
function Rez=rezystancja(x)
if x>0.1*iprog Rez=0.0002*Rblock;
elseif x>0.07*iprog Rez=0.05*Rblock;
elseif x>0.05*iprog Rez=0.1*Rblock;
elseif x>0.03*iprog Rez=0.2*Rblock;
elseif x>0.01*iprog Rez=0.5*Rblock;
elseif x>0 Rez=Rblock;
elseif x>-0.01*iprog Rez=2*Rblock;
elseif x>-0.03*iprog Rez=3*Rblock;
elseif x>-0.05*iprog Rez=4*Rblock;
elseif x>-0.07*iprog Rez=5*Rblock;
elseif x>-0.09*iprog Rez=6*Rblock;
elseif x>-0.11*iprog Rez=8*Rblock;
else Rez=10*Rblock;
end
end
i1=Y(1); i2=Y(2); i3=Y(3); i4=Y(4); i5=Y(5); i6=Y(6); i7=Y(7);
 betaA=omega*t+phi;
 betaC=betaA+pi/3;
 beta1=mod(betaA,2*pi);
 beta6=mod(betaA-pi/3,2*pi);
 beta3=mod(betaA-2/3*pi,2*pi);
 beta2=mod(betaA-pi,2*pi);
 beta5=mod(betaA-4/3*pi,2*pi);
 beta4=mod(betaA-5/3*pi,2*pi);
 uAB=Um*sin(betaA);
 uCB=Um*sin(betaC);
 iT1=i1+i3+i5; iT2=i3+i5; iT3=-i1-i3+i4;
 iT4=i2-i3+i4; iT5=i6-i4; iT6=-i2-i4+i6;
 RT1=10*Rblock; RT2=10*Rblock; RT3=10*Rblock;
 RT4=10*Rblock; RT5=10*Rblock; RT6=10*Rblock;
 alpha1=alpha+pi/3;
if beta1>alpha1
 RT1=rezystancja(iT1);
end
if beta6>alpha1 RT6=rezystancja(iT6);
end
if beta3>alpha1 RT3=rezystancja(iT3);
end
if beta2>alpha1 RT2=rezystancja(iT2);
end
if beta5>alpha1 RT5=rezystancja(iT5);
end
if beta4>alpha1 RT4=rezystancja(iT4);
end
 ps1=(-RT1-RT3-2*Rs)*i1-Rs*i2+(-RT1-RT3)*i3-RT1*i5+RT3*i4+uAB;
 ps2=-Rs*i1+(-RT4-RT6-2*Rs)*i2+RT4*i3+(-RT4-RT6)*i4+RT6*i6+uCB;
 ps3=(-RT1-RT3)*i1+RT4*i2+(-RT1-RT2-RT3-RT4)*i3+(RT3+RT4)*i4+(-RT1...
-RT2)*i5;
 ps4=RT3*i1+(-RT4-RT6)*i2+(RT3+RT4)*i3+(-RT3-RT4-RT5...
```

```
-RT6)*i4+(RT5+RT6)*i6;
 ps5=-RT1*i1+(-RT1-RT2)*i3+(-RT1-RT2-Rl)*i5-Rl*i6-E-Rl*i7;
 ps6=RT6*i2+(RT5+RT6)*i4-Rl*i5+(-RT5-RT6-Rl)*i6-E-Rl*i7;
 ps7=-Rl*(i5+i6)+(-Rl-RV)*i7-E;

di1dt=A1(1)*ps1+A1(8)*ps2+A1(15)*ps3+A1(22)*ps4+A1(29)*ps5+A1(36)*ps6+A1(43
)*ps7;

di2dt=A1(2)*ps1+A1(9)*ps2+A1(16)*ps3+A1(23)*ps4+A1(30)*ps5+A1(37)*ps6+A1(44
)*ps7;

di3dt=A1(3)*ps1+A1(10)*ps2+A1(17)*ps3+A1(24)*ps4+A1(31)*ps5+A1(38)*ps6+A1(4
5)*ps7;

di4dt=A1(4)*ps1+A1(11)*ps2+A1(18)*ps3+A1(25)*ps4+A1(32)*ps5+A1(39)*ps6+A1(4
6)*ps7;

di5dt=A1(5)*ps1+A1(12)*ps2+A1(19)*ps3+A1(26)*ps4+A1(33)*ps5+A1(40)*ps6+A1(4
7)*ps7;

di6dt=A1(6)*ps1+A1(13)*ps2+A1(20)*ps3+A1(27)*ps4+A1(34)*ps5+A1(41)*ps6+A1(4
8)*ps7;

di7dt=A1(7)*ps1+A1(14)*ps2+A1(21)*ps3+A1(28)*ps4+A1(35)*ps5+A1(42)*ps6+A1(4
9)*ps7;
 dYdt=[di1dt;di2dt;di3dt;di4dt;di5dt;di6dt;di7dt];
end



% name: Bridge6_Motor
% Bridge controlled by alpha
% Change of alpha -> exponential: time constant Ta
% Load : DC Motor
% Motor Data: 6.6 kW; 400 V; 21 A; 30.4 Nm;...
% 74.8% Pf=420 W; 3.33 Ohm; 39 mH; 0.03 kgm^2...
% cE=1.525; Damp=0.0075;
% Power computed; efficiency computed
format shortg;
c1=clock
step=0.001;
tfin=2.5;
tRange=[0:step:tfin];
Nst=tfin/step+1; Su2=0;
NST=round(Nst);
i0=[0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0];
global Um; Um=500;
f=50.0;
global omega; omega=2*pi*f;
global phi; phi=0;
Ls=0.003; Lm=0.001;
global Ld; Ld=0.039;
global cE; cE=1.525;
global Rs; Rs=0.1;
global Rl; Rl=3.33;
global J; J=0.1; Tn=30.4; kl=0.8;
global Tl; Tl=kl*Tn;
global Da; Da=0.0075;
global Ta; Ta=0.9;
vn=2070/9.56; In=21;
global RV; RV=100000000;
```

```
global iprog; iprog=0.0002;
global Rblock; Rblock=1000;
A=[2*(Lm+Ls) Ls 2*Lm -Lm 2*Lm 0 0; Ls 3*Lm+2*Ls -2*Lm 2*Lm -Lm -Lm 0;...
2*Lm -2*Lm 4*Lm -2*Lm 3*Lm 0 0; -Lm 2*Lm -2*Lm 4*Lm -Lm -2*Lm 0;...
2*Lm -Lm 3*Lm -Lm Ld+3*Lm Ld Ld; 0 -Lm 0 -2*Lm Ld Ld+2*Lm Ld;0 0 0 0 Ld Ld
Ld];
global A1; A1=inv(A);
global alpha; alpha=90/180*pi;
[tSol,YSol]=ode23s(@Rec6,tRange,i0);
[ID]=[YSol(:,5)+YSol(:,6)+YSol(:,7)];
[iB]=[-YSol(:,1)-YSol(:,2)];
[Volt]=[-RV*YSol(:,7)];
[sAB]=[sin(omega*tRange+phi)];
[sCB]=[sin(omega*tRange+phi+pi/3)];
for k=1:Nst Power1(k)=Um*sAB(k)*YSol(k,1);
Power2(k)=Um*sCB(k)*YSol(k,2);
Power(k)=Power1(k)+Power2(k);
end
kp=round(0.8*Nst);
kf=NST
vAv=sum(YSol(kp:kf,8))/(kf-kp);
rev=vAv*9.56
PowAv=sum(Power(kp:kf))/(kf-kp)
VoltAv=sum(Volt(kp:kf))/(kf-kp)
IDAv=sum(ID(kp:kf))/(kf-kp)
E=cE*vAv
PowLoad=Rl*IDAv^2+E*IDAv
Pm=Tl*vAv;
for k=kp:kf Su2=Su2+(ID(k))^2;
end
PowLoad2=Rl*Su2/(kf-kp+1)+E*IDAv
% efficiency load case
eta1=PowLoad2/PowAv
% efficiency motor case
eta2=Pm/PowAv
f1=figure; f2=figure; f3=figure; f4=figure;
figure(f1)
plot(tSol,Power/1000)
grid on;
xlabel('time [s]')
ylabel('Power [kW]')
legend('Source Power [kW]')
title('Source Power')
figure(f2)
plot(tSol,Volt)
grid on;
xlabel('time [s]')
ylabel('Load Voltage [V]')
legend('Load Voltage [V]','Location','northwest');
title('Load Voltage')
figure(f3)
plot(tSol,YSol(:,1))
grid on;
xlabel('time [s]')
ylabel('Currents')
title('Phase Currents & DC Curent; alpha=90->0 [deg]')
text(0.8,35.0,'alpha=90->>0 deg')
hold on
plot(tSol,YSol(:,2))
plot(tSol,iB)
plot(tSol,ID)
```

```
hold off;
figure(f4)
plot(tSol,YSol(:,8)/vn,'Linewidth',2.5)
grid on;
xlabel('time [s]')
ylabel('Speed & Current')
title('Speed & Armature Current; alpha=90->0 [deg]; Ta=0.9 [s]')
hold on
plot(tSol,ID/In)
legend('Rotary speed','Armature current');
hold off
c2=clock
czas=c2-c1
function dYdt=Rec6(t,Y)
global Um; global omega; global phi;
global cE; global Rs; global Rl; global iprog;
global Rblock; global A1; global alpha;
global Ld; global RV; global J; global Tl;
global Da; global Ta;
function Rez=rezystancja(x)
if x>0.1*iprog Rez=0.0002*Rblock;
elseif x>0.07*iprog Rez=0.05*Rblock;
elseif x>0.05*iprog Rez=0.1*Rblock;
elseif x>0.03*iprog Rez=0.2*Rblock;
elseif x>0.01*iprog Rez=0.5*Rblock;
elseif x>0 Rez=Rblock;
elseif x>-0.01*iprog Rez=2*Rblock;
elseif x>-0.03*iprog Rez=3*Rblock;
elseif x>-0.05*iprog Rez=4*Rblock;
elseif x>-0.07*iprog Rez=5*Rblock;
elseif x>-0.09*iprog Rez=6*Rblock;
elseif x>-0.11*iprog Rez=8*Rblock;
else Rez=10*Rblock;
end
end
i1=Y(1); i2=Y(2); i3=Y(3); i4=Y(4); i5=Y(5); i6=Y(6);
i7=Y(7); v=Y(8); id=i5+i6+i7;
 betaA=omega*t+phi;
 betaC=betaA+pi/3;
 beta1=mod(betaA,2*pi);
 beta6=mod(betaA-pi/3,2*pi);
 beta3=mod(betaA-2/3*pi,2*pi);
 beta2=mod(betaA-pi,2*pi);
 beta5=mod(betaA-4/3*pi,2*pi);
 beta4=mod(betaA-5/3*pi,2*pi);
 uAB=Um*sin(betaA);
 uCB=Um*sin(betaC);
 E=cE*v;
 iT1=i1+i3+i5; iT2=i3+i5; iT3=-i1-i3+i4;
 iT4=i2-i3+i4; iT5=i6-i4; iT6=-i2-i4+i6;
 RT1=10*Rblock; RT2=10*Rblock; RT3=10*Rblock;
 RT4=10*Rblock; RT5=10*Rblock; RT6=10*Rblock;
 alpha1=alpha+pi/3;
  alpha2=alpha1*exp(-t/Ta);
if beta1>alpha2
 RT1=rezystancja(iT1);
end
if beta6>alpha2 RT6=rezystancja(iT6);
end
if beta3>alpha2 RT3=rezystancja(iT3);
end
```

```
if beta2>alpha2 RT2=rezystancja(iT2);
end
if beta5>alpha2 RT5=rezystancja(iT5);
end
 if beta4>alpha2 RT4=rezystancja(iT4);
end
 ps1=(-RT1-RT3-2*Rs)*i1-Rs*i2+(-RT1-RT3)*i3-RT1*i5+RT3*i4+uAB;
 ps2=-Rs*i1+(-RT4-RT6-2*Rs)*i2+RT4*i3+(-RT4-RT6)*i4+RT6*i6+uCB;
 ps3=(-RT1-RT3)*i1+RT4*i2+(-RT1-RT2-RT3-RT4)*i3+(RT3+RT4)*i4+(-RT1-RT2)*i5;
 ps4=RT3*i1+(-RT4-RT6)*i2+(RT3+RT4)*i3+(-RT3-RT4-RT5-RT6)*i4+(RT5+RT6)*i6;
 ps5=-RT1*i1+(-RT1-RT2)*i3+(-RT1-RT2-Rl)*i5-Rl*i6-E-Rl*i7;
 ps6=RT6*i2+(RT5+RT6)*i4-Rl*i5+(-RT5-RT6-Rl)*i6-E-Rl*i7;
 ps7=Rl*-(i5+i6)-(Rl+RV)*i7-E;

di1dt=A1(1)*ps1+A1(8)*ps2+A1(15)*ps3+A1(22)*ps4+A1(29)*ps5+A1(36)*ps6+A1(43
)*ps7;

di2dt=A1(2)*ps1+A1(9)*ps2+A1(16)*ps3+A1(23)*ps4+A1(30)*ps5+A1(37)*ps6+A1(44
)*ps7;

di3dt=A1(3)*ps1+A1(10)*ps2+A1(17)*ps3+A1(24)*ps4+A1(31)*ps5+A1(38)*ps6+A1(4
5)*ps7;

di4dt=A1(4)*ps1+A1(11)*ps2+A1(18)*ps3+A1(25)*ps4+A1(32)*ps5+A1(39)*ps6+A1(4
6)*ps7;

di5dt=A1(5)*ps1+A1(12)*ps2+A1(19)*ps3+A1(26)*ps4+A1(33)*ps5+A1(40)*ps6+A1(4
7)*ps7;

di6dt=A1(6)*ps1+A1(13)*ps2+A1(20)*ps3+A1(27)*ps4+A1(34)*ps5+A1(41)*ps6+A1(4
8)*ps7;

di7dt=A1(7)*ps1+A1(14)*ps2+A1(21)*ps3+A1(28)*ps4+A1(35)*ps5+A1(42)*ps6+A1(4
9)*ps7;
 dvdt=(cE*id-Tl -Da*v)/J;
 dYdt=[di1dt;di2dt;di3dt;di4dt;di5dt;di6dt;di7dt;dvdt];
end
```

```
% test funkcji massMatrixForm
% test dla ode15s
% na podstawie przykładu z dokumentacji
% dla równania DAE
clearvars;
syms x1(t) x2(t) f(t,x1,x2) r m;
eqs=[m*x2(t)*diff(x1(t),t)+m*t*diff(x2(t),t)==f(t,x1(t),x2(t)),...
    x1(t)^2+x2(t)^2==r^2];
vars=[x1(t),x2(t)];
% [M,F]=massMatrixForm(eqs,vars)
%  M = [m*x2(t), m*t]
%      [      0,   0]
%  F =  f(t, x1(t), x2(t))
%       r^2 - x2(t)^2 - x1(t)^2
 M=[m*x2(t),m*t; 0,0];
 F=[f(t,x1(t),x2(t)); r^2 - x2(t)^2 - x1(t)^2];
 syms Y1 Y2;
M1=subs(M,[vars,m,r,f],[Y1,Y2,100,1,@(t,x1,x2) t+x1*x2])
F1=subs(F,[vars,m,r,f],[Y1,Y2,100,1,@(t,x1,x2) t+x1*x2])
```

```
MM=matlabFunction(M1,'vars',{t,[Y1;Y2]})
FF=matlabFunction(F1,'vars',{t,[Y1;Y2]})
tspan=[0 1.6];
Y0=[0.5;0.5*sqrt(3)];
opt=odeset('Mass',MM,'MassSingular','yes','InitialSlope',[0.05;0]);
[t,Y]=ode15s(FF,tspan,Y0,opt);
plot(t,Y(:,1))
grid on
hold on
plot(t,Y(:,2))
hold off;
f0=figure;
plot(Y(:,1),Y(:,2))
```