**POLITECNICO DI MILANO**

**Scuola di Ingegneria Industriale e dell'Informazione**

**Corso di Laurea in Ingegneria Elettrica**

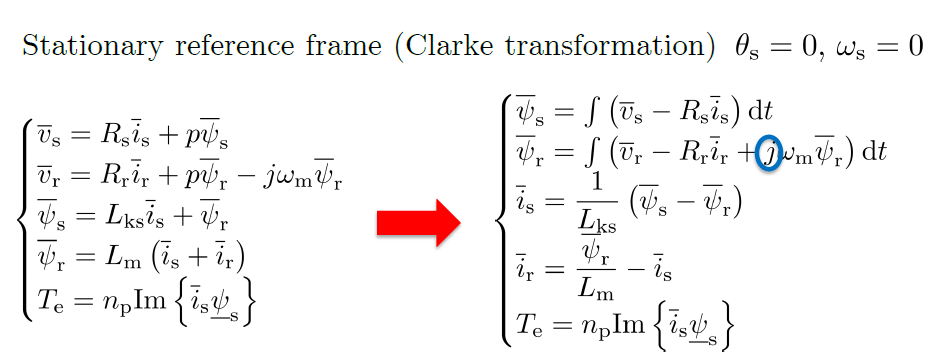


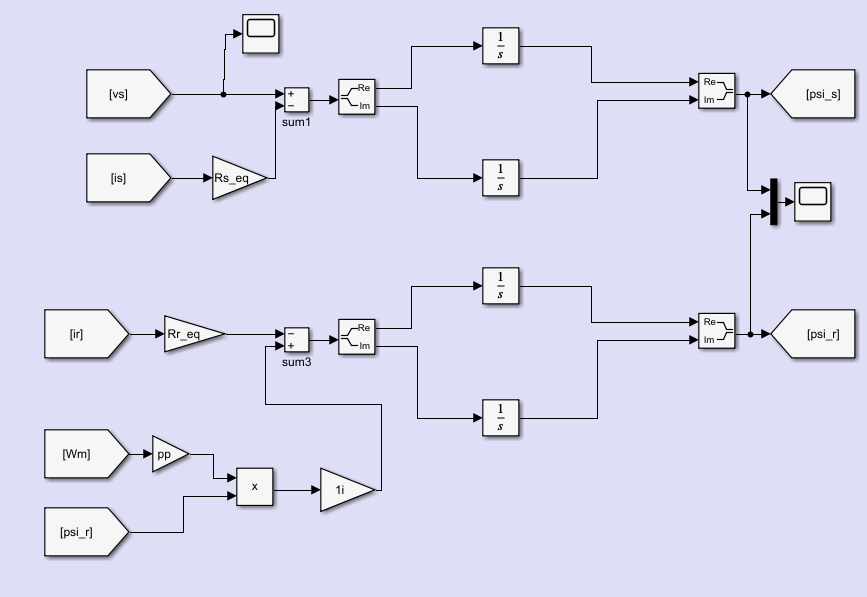
Field Oriented Control of an induction motor of a train

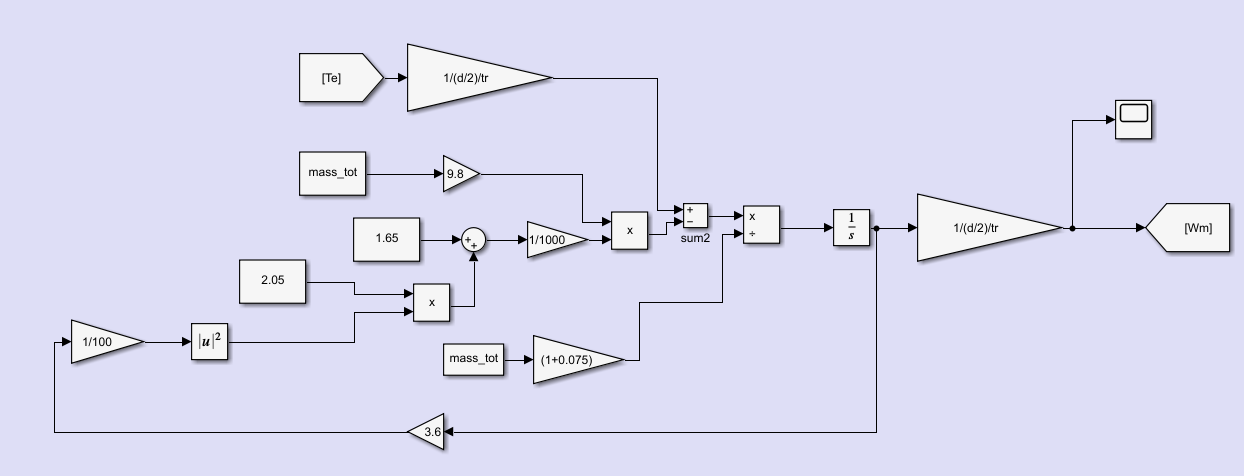
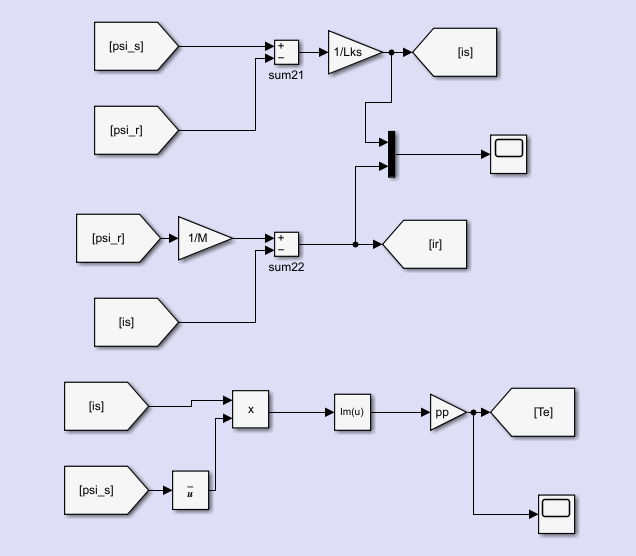
Alessandro Secchi

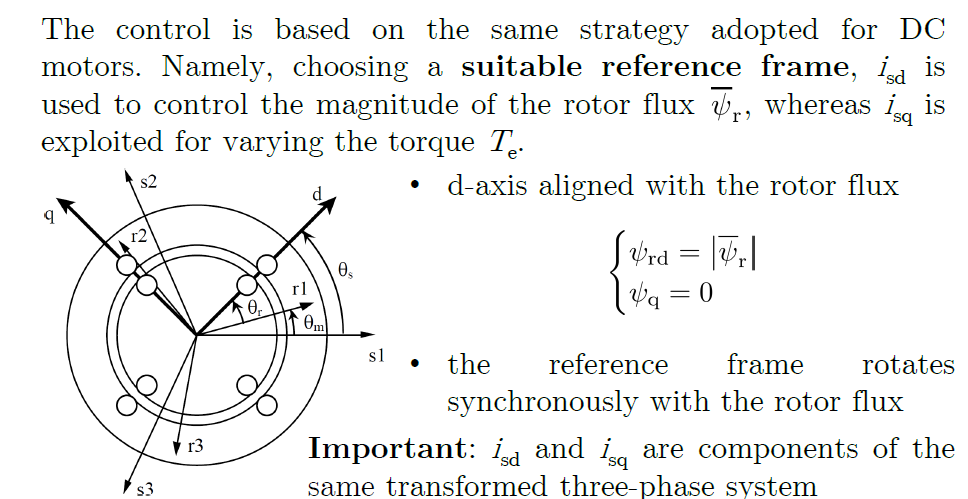
Matr. 944668

For the motor model, a stationary reference frame has ben chosen:

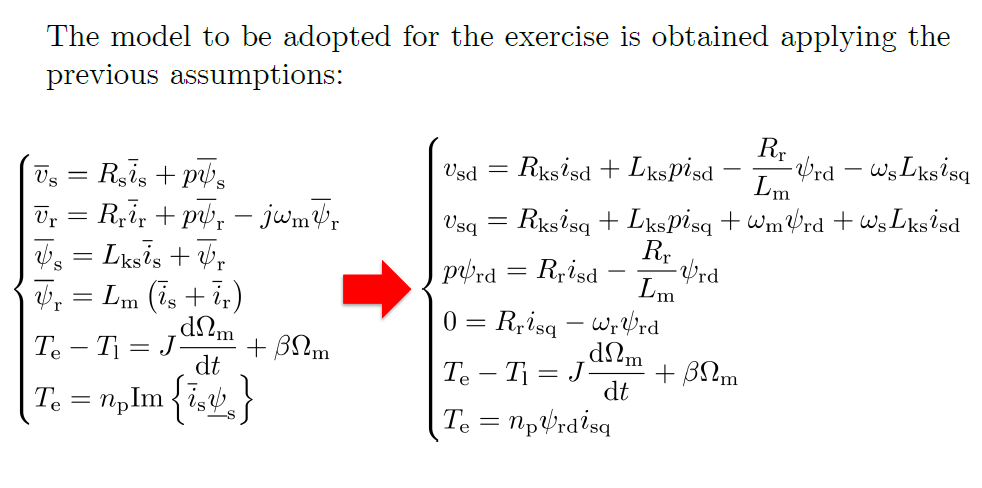




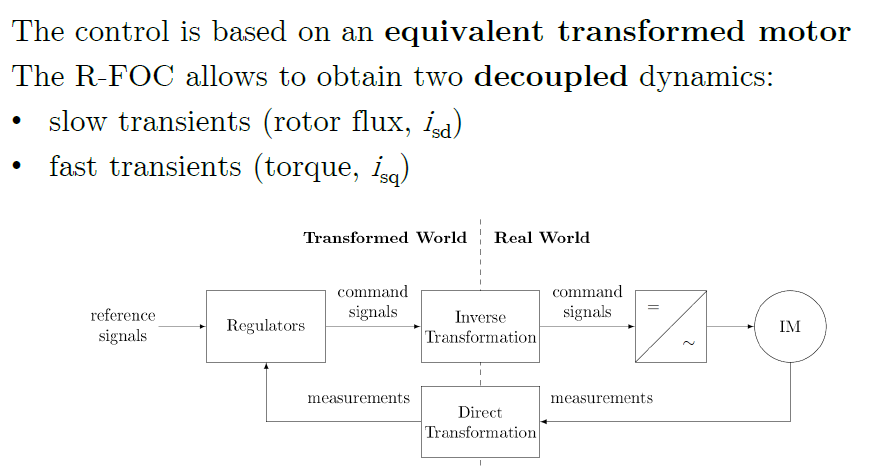




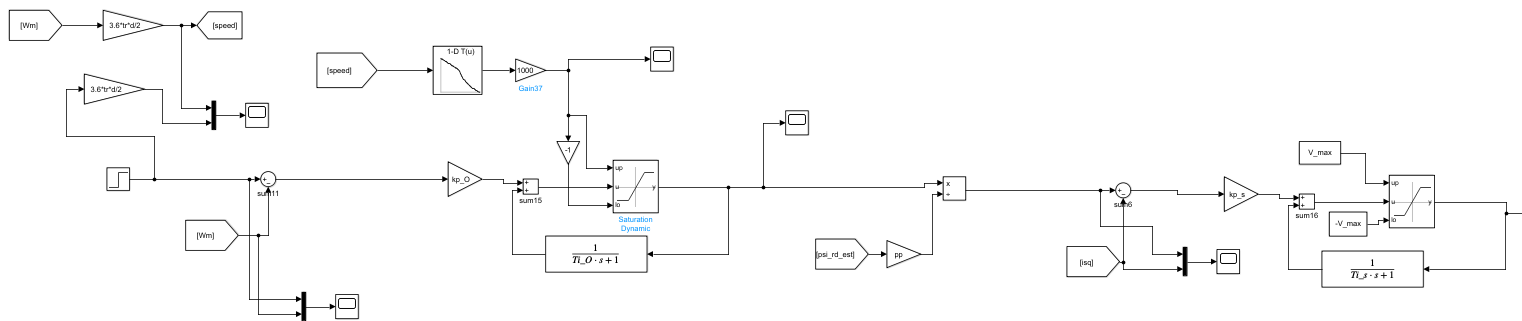
For the control scheme, instead:

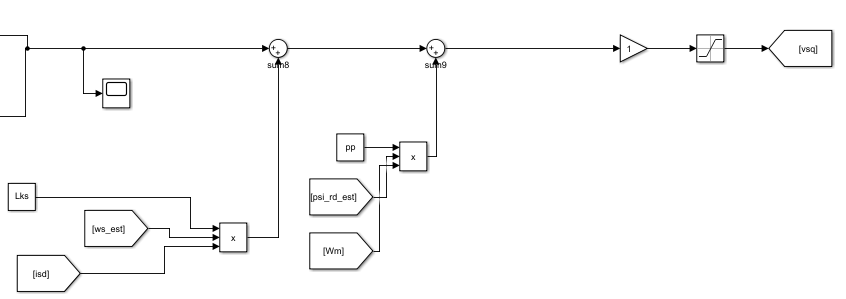


The vector control try to follow the idea of the DC drive.

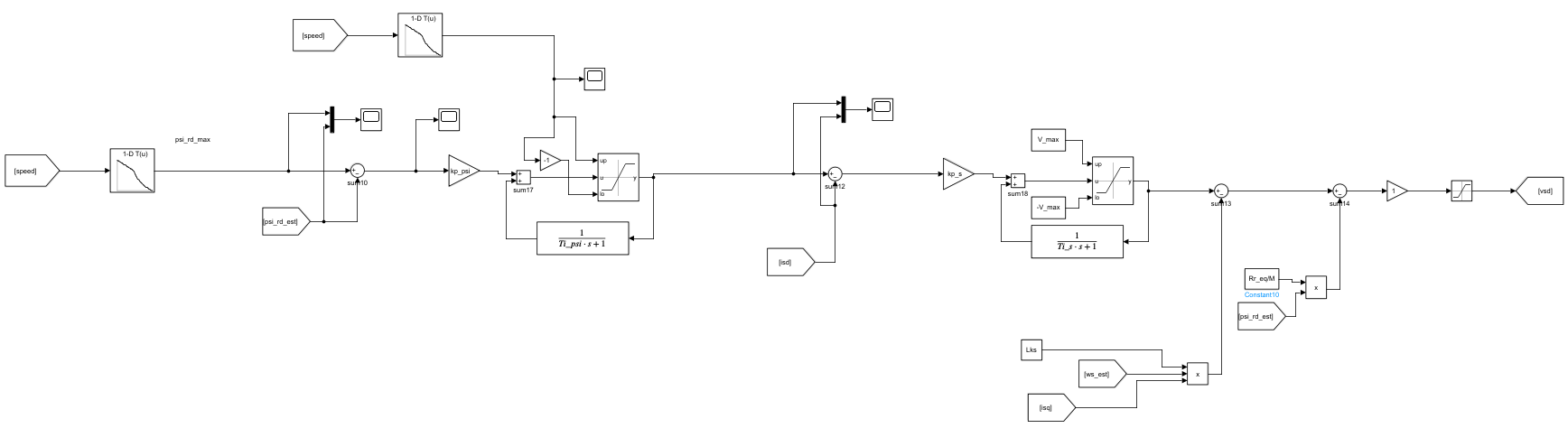


Speed and isd regulators with the decoupling terms:

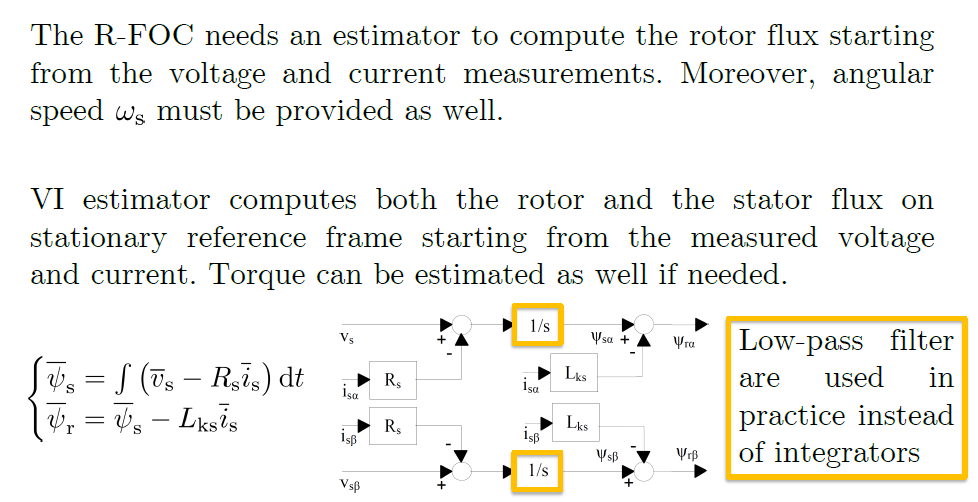


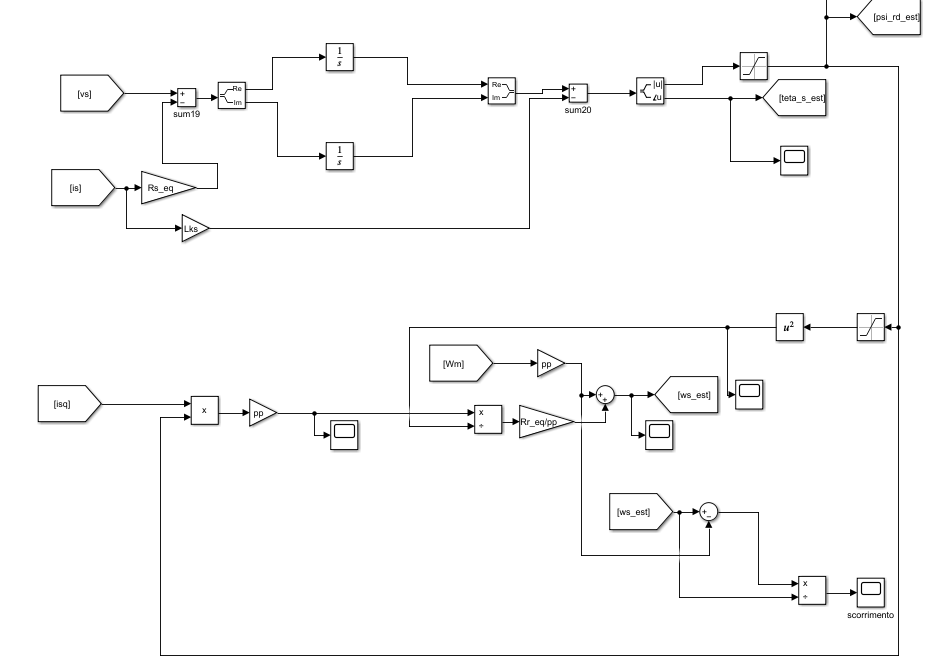


Flux and isq regulator with decoupling terms:

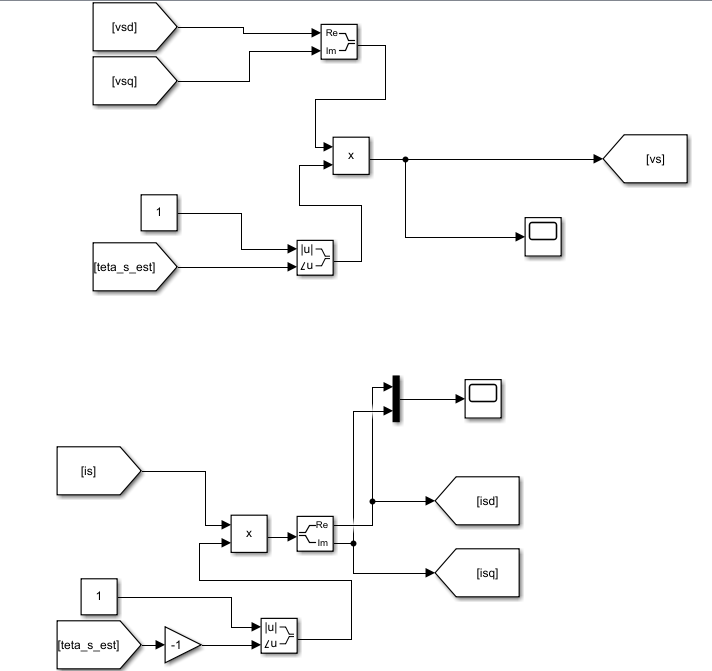


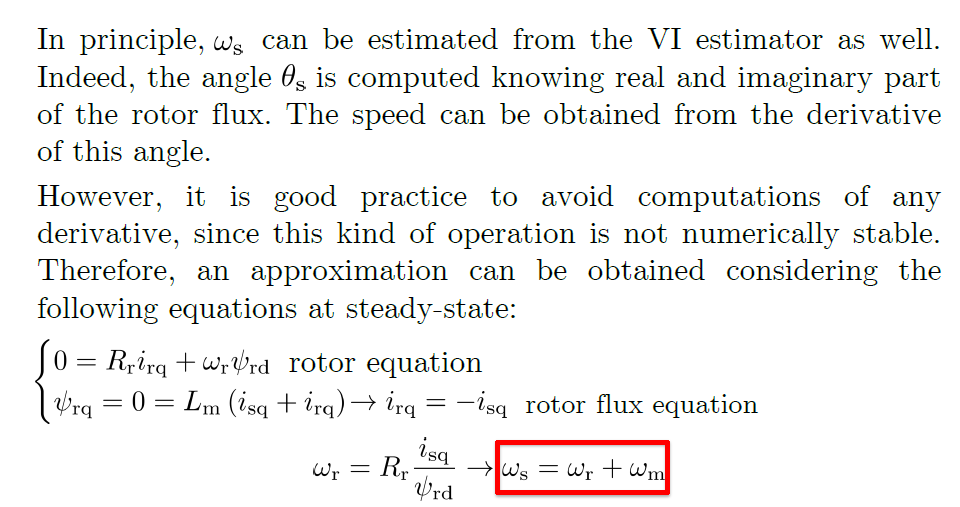
The estimators are a fundamental part of the control scheme.



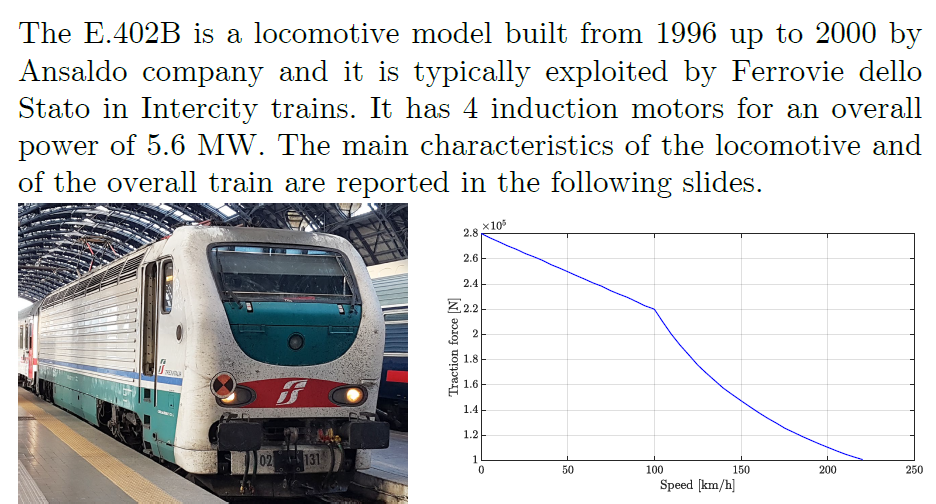


In order to pass from dq reference frame to space phasor and viceversa, the following blocks are also needed:



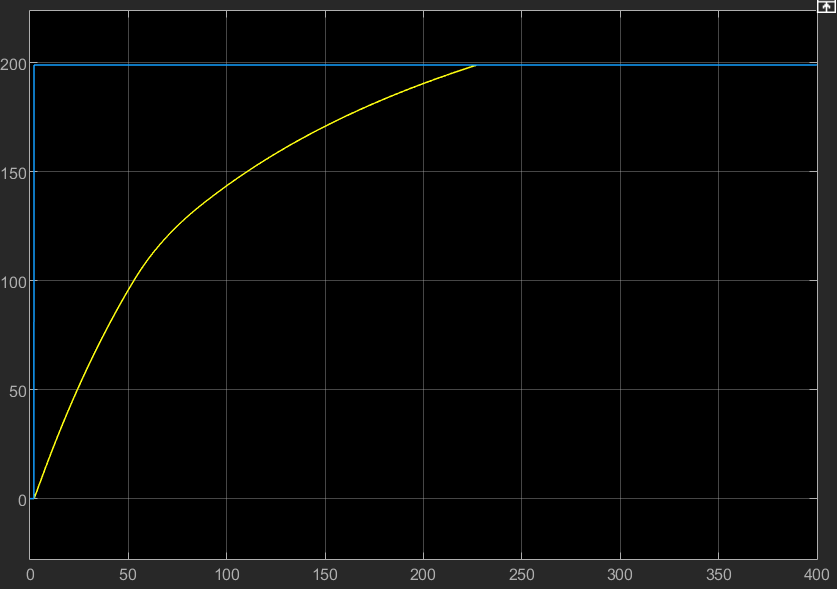


Exercise and data (please refer to the code at the end of the report):



Results:

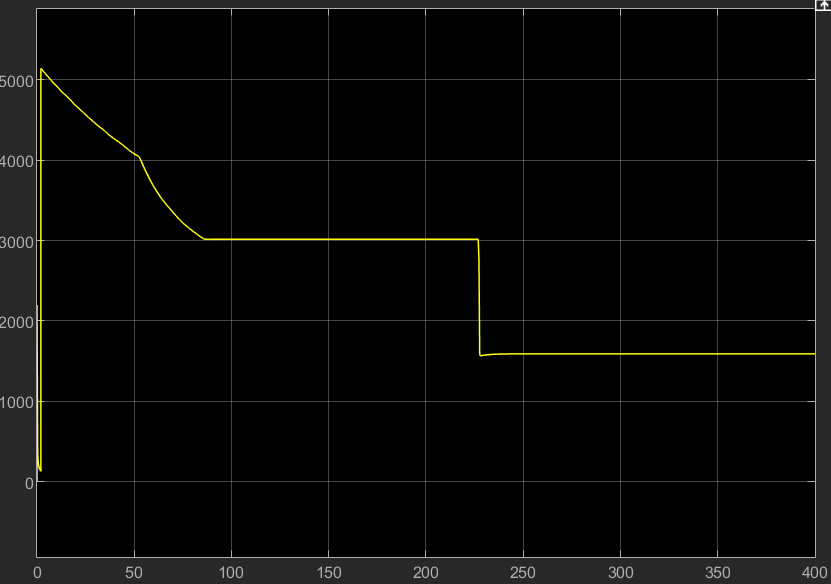
Speed profile in km/h:



Note:

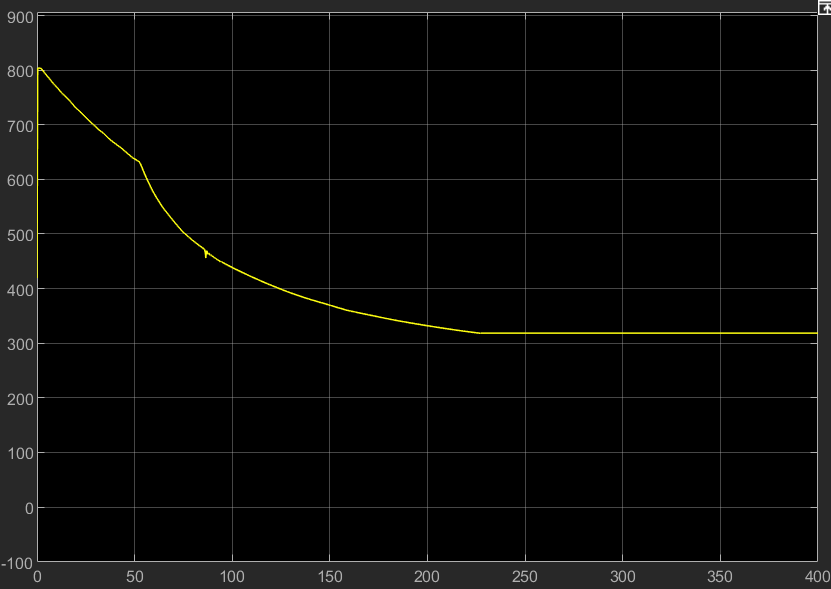
* The profile simulates a max acceleration and the step reference start from 2 sec, this because in those 2 sec there is time to create some rotor flux and have available starting torque.
* The reference is followed slower after 100km/h, infact there is lower accelleration since the train is running over its based speed.
* In the reality, in case of starting without the presence of rotor flux, there is a buttom on train that applied just isd current to create some initial flux.
* Actually, when the train reaches a certain velocity, it switches off the traction force up to a min speed sensed and then it turns on and so on: coasting mode.

Isq current in A:

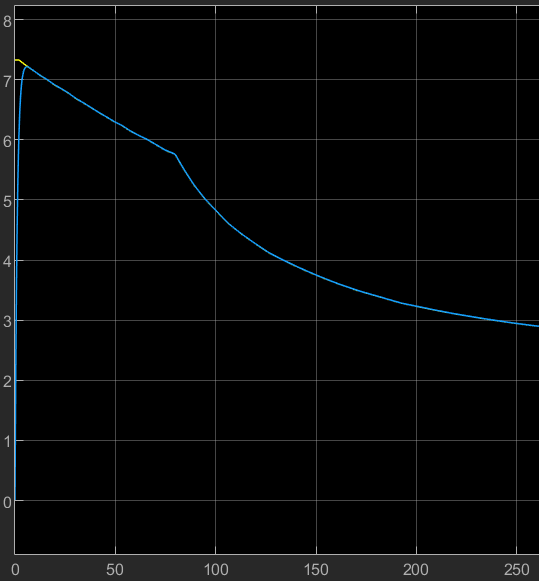


The current at the beginning decrease due to the flux weakening and then it is stabilized just to win the friction forces.

The profile of the isd current is:



While the profile of the rotor flux has still the same shape (due to the flux weakening) and with max value set at almost 7Wb:



Code:

clc

clear all

%% DC machine parameters

load('Traction\_e402B\_online');

%parameters of 1 motor = equivalent motor

V\_motor = 1860; %voltage of each motor

V\_max=V\_motor\*sqrt(3);

pp=2; %pole pairs of each motor

cos\_fi=0.85; %power factor of each motor

Rs=0.01; %p.u.

Rr=0.01; %p.u.

X\_lock=0.1; %p.u.

X\_noload=3.5; %p.u.

d=1250\*10^-3; %diameter wheels

tr=23/91; %trasmission ratio

mass\_tot=89\*10^3+11\*47\*10^3+45\*80+45\*15; %kg

v\_max=200\*1000/3600; %max speed m/s

omega\_max=v\_max/tr/d\*2; %rad/s

eff=0.9; %efficiency

J=mass\_tot\*v\_max^2/omega\_max^2;

%J\_eq=mass\_tot\*(tr\*d)^2/4;

Pn=5.6\*10^6; %tot mechanical power [W]

Pe\_tot=Pn/eff;

I\_motor=Pe\_tot/sqrt(3)/V\_motor/cos\_fi;

Z=V\_motor/I\_motor;

Rs\_eq=Rs\*Z;

Rr\_eq=Rr\*Z;

X\_lock\_eq=X\_lock\*Z;

X\_noload\_eq=X\_noload\*Z;

Rks=Rs\_eq+Rr\_eq;

Lks=X\_lock\_eq/(2\*pi\*50);

M=(X\_noload\_eq)/(2\*pi\*50);

F\_friction=62000; %N at v\_max

T\_friction=F\_friction/2\*d/2\*tr; %at 100km/h base speed

omega\_b=100\*1000/3600/tr/d\*2;

psi\_r\_max=V\_motor\*sqrt(3)\*0.8/(omega\_b\*pp);

k=traction(1,2)/psi\_r\_max;

psi\_rd\_ref = traction(:,2)/k;

speed\_ref = traction(:,1);

id\_ref=psi\_rd\_ref/M;

Tn = Pn/omega\_b; % Nominal torque provided by the machine

iq\_max=1000\*traction(1,2)\*d/2\*tr/pp/psi\_r\_max;

tau\_s=Lks/Rs;

B=T\_friction/omega\_b;

tau\_O=J/B;

tau\_psi=M/Rr\_eq;

% B = Tfriction/rated\_speed\_motor; % friction coefficient < Or

% tau\_mec=J\_eq/B;

%% PI controller design parameters

s=tf('s');

%Gi

tau\_s\_desired=tau\_s/100;

wc\_s=2\*pi/tau\_s\_desired;

%GO

tau\_O\_desired=tau\_O/1000;

wc\_O=2\*pi/tau\_O\_desired;

%Gpsi

tau\_psi\_desired=tau\_psi/1000;

wc\_psi=2\*pi/tau\_psi\_desired;

%tf

Gi = 1/(Rs+Lks\*s);

GO = 1/(B+J\*s);

Gpsi = (1/M)/(1+(M/Rr\_eq)\*s);

%% Zero Pole cancellation (90 phase margin)

% %PI parameters ia

% kp\_a=wc\_a\*La;

% ki\_a=wc\_a\*Ra;

% Regi=kp\_a+ki\_a/s

% Ti\_a=kp\_a/ki\_a;

% %tf open loop

% Li=Regi\*Gi;

% %tf close loop

% Fi=Li/(1+Li);

% % figure

% % bode(Li)

% % figure

% % bode(Fi)

% %PI parameters ie

% kp\_e=wc\_e\*Le;

% ki\_e=wc\_e\*Re;

% Rege=kp\_e+ki\_e/s

% Ti\_e=kp\_e/ki\_e;

% %tf open loop

% L\_e=Rege\*Ge;

% %tf close loop

% Fe=L\_e/(1+L\_e);

% % figure

% % bode(L\_e)

% % figure

% % bode(Fe)

% %PI parameters speed

% kp\_O=wc\_O\*J\_eq;

% ki\_O=wc\_O\*B;

% RegO=kp\_O+ki\_O/s

% Ti\_O=kp\_O/ki\_O;

% %tf open loop

% LO=RegO\*GO;

% %tf close loop

% FO=LO/(1+LO);

% % figure

% % bode(LO)

% % figure

% % bode(FO)

%% Pidtool

%otherwise use pidtool

phase\_m=90;

%pidtool(Gi)

opt=pidtuneOptions('PhaseMargin', phase\_m);

par\_regi=pidtune(Gi,'PI',wc\_s,opt);

ki\_s=par\_regi.Ki;

kp\_s=par\_regi.Kp;

Regi=kp\_s+ki\_s/s

Ti\_s=kp\_s/ki\_s;

%tf open loop

Li=Regi\*Gi;

%tf close loop

Fi=Li/(1+Li);

% figure

% bode(Li);

% figure

% bode(Fi);

% figure

% margin(Li);

%pidtool(GO)

par\_reg\_speed=pidtune(GO,'PI',wc\_O,opt);

ki\_O=par\_reg\_speed.Ki;

kp\_O=par\_reg\_speed.Kp;

RegO=kp\_O+ki\_O/s

Ti\_O=kp\_O/ki\_O;

%tf open loop

LO=RegO\*GO;

%tf close loop

FO=LO/(1+LO);

% figure

% bode(LO);

% figure

% bode(FO);

% figure

% margin(LO);

%pidtool(Gpsi)

par\_reg\_speed=pidtune(Gpsi,'PI',wc\_psi,opt);

ki\_psi=par\_reg\_speed.Ki;

kp\_psi=par\_reg\_speed.Kp;

Regpsi=kp\_psi+ki\_psi/s

Ti\_psi=kp\_psi/ki\_psi;

%tf open loop

Lpsi=Regpsi\*Gpsi;

%tf close loop

Fpsi=Lpsi/(1+Lpsi);

% figure

% bode(LO);

% figure

% bode(FO);

% figure

% margin(LO);