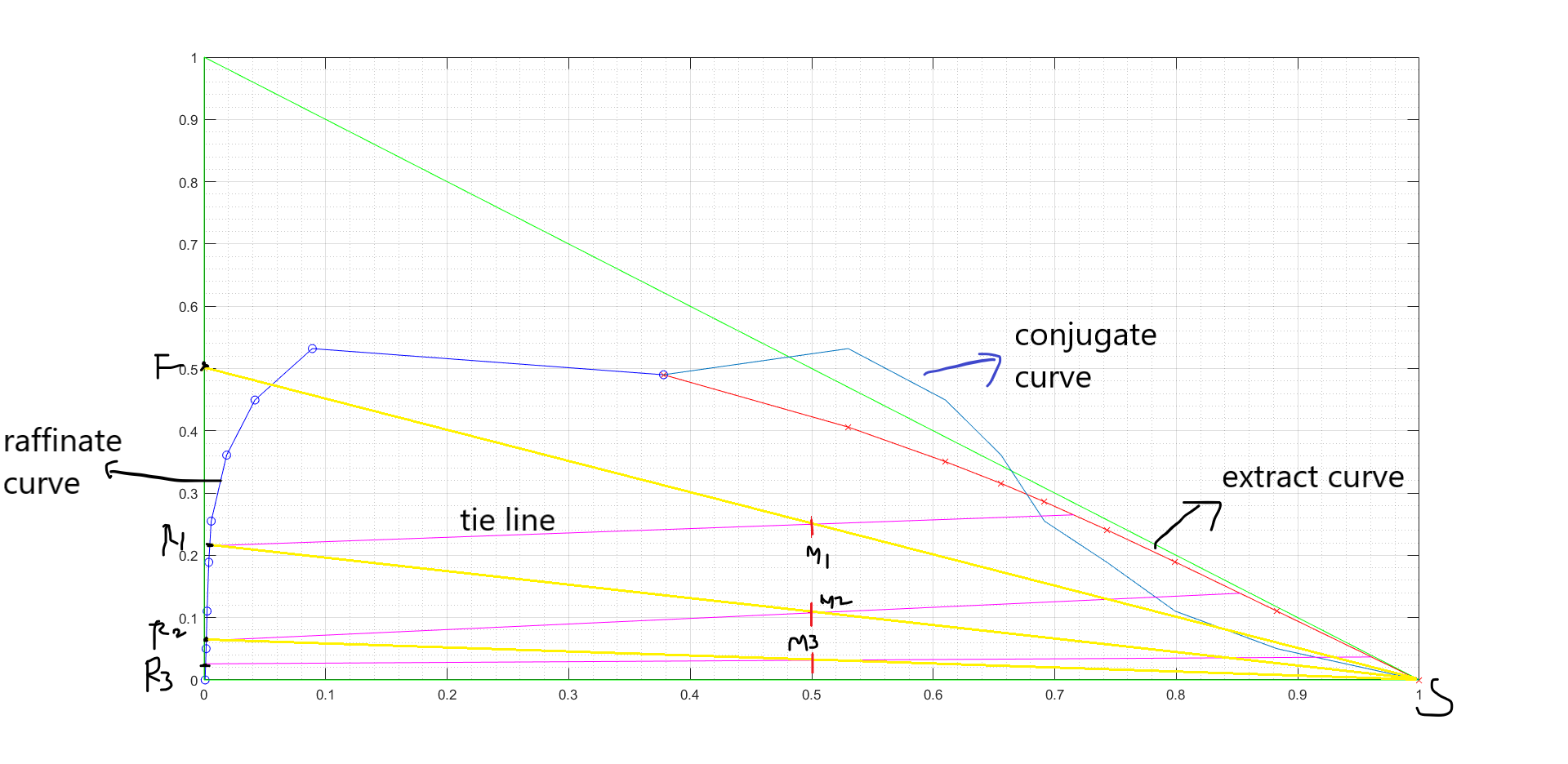
# Question 1

## **Stepping process**

**X coordinate**- composition of **chlorobenzene**

**Y coordinate**- composition of **Pyridine**



## **Code**

clear; close all;

%% Equilibrium Data

a\_e = [0,11.05,18.95,24.1,28.6,31.55,35.05,40.6,49]\*0.01;

c\_e = 0.01\*[0.05 0.67 1.15 1.62 2.25 2.87 3.95 6.4 13.2];

s\_e = 0.01\*[99.95 88.28 79.9 74.28 69.15 65.58 61 53 37.8];

a\_r = 0.01\*[0 5.02 11.05 18.9 25.5 36.1 44.95 53.2 49];

s\_r = 0.01\*[0.08 0.16 0.24 0.38 0.58 1.85 4.18 8.9 37.8];

c\_r = 0.01\*[99.92 94.82 88.71 80.72 73.92 62.05 50.87 37.9 13.2];

plot(s\_e,a\_e,'red',s\_r,a\_r,'blue',s\_e,a\_e,'rx',s\_r,a\_r,'bo');

hold on;

plot([0,1],[0,0],'green',[0,0],[0,1],'green',[1,0],[0,1],'green');

% %Plotting the tie lines

% for i = 1:9

% plot([s\_r(i),s\_e(i)],[a\_r(i),a\_e(i)],'m');

% end

grid on;grid minor;

rx = spline(a\_r(1:8),s\_r(1:8));

ey = spline(s\_e(1:8),a\_e(1:8));

%% conjugate curve

cc\_y = spline(s\_e,a\_r);

plot(s\_e,a\_r);

cc\_x = spline(a\_r(1:8),s\_e(1:8));

%% Feed data

mF1 = 2000;

yF1 = 0.5;

yS = 0;

S1 = mF1;

%% Mixing point 1

yM = yF1/2;

xM = 0.5;%Always

plot(xM,yM,'+');

sol = fsolve(@(x) cctie(x,xM,yM,rx,ey,cc\_y),0.6);

[xR1,yR1,xE1,yE1] = endpts(sol,rx,ey,cc\_y);

plot([xR1,xE1],[yR1,yE1],'m',0,yF1,'x');

%% Stage 2

yM2 = yR1/2;

plot(xM,yM2,'r+');

sol2 = fsolve(@(x) cctie(x,xM,yM2,rx,ey,cc\_y),0.7);

[xR2,yR2,xE2,yE2] = endpts(sol2,rx,ey,cc\_y);

plot([xR2,xE2],[yR2,yE2],'m');

%% Stage 3

yM3 = yR2/2;

plot(xM,yM3,'r+');

sol2 = fsolve(@(x) cctie(x,xM,yM3,rx,ey,cc\_y),0.7);

[xR3,yR3,xE3,yE3] = endpts(sol2,rx,ey,cc\_y);

plot([xR3,xE3],[yR3,yE3],'m');

% %% Stage 4

% yM3 = yR3/2;

% plot(xM,yM3,'r+');

% sol2 = fsolve(@(x) cctie(x,xM,yM3,rx,ey,cc\_y),0.7);

% [xR3,yR3,xE3,yE3] = endpts(sol2,rx,ey,cc\_y);

% plot([xR3,xE3],[yR3,yE3],'m');

%% mass balance

raff\_m = zeros(1,3)+mF1;

R = mF1;

y\_E = [yE1 yE2 yE3];

y\_R = [yF1 yR1 yR2 yR3];

for i = 1:2

R = R\*(2\*y\_E(i)-y\_R(i))/(y\_E(i)-y\_R(i+1));

raff\_m(i+1) = R;

end

%% Function to get end pts of tie line given pt on cc

function [xR,yR,xE,yE] = endpts(x,rx,ey,cc\_y)

xE = x;

yR = ppval(cc\_y,x);

xR = ppval(rx,yR);

yE = ppval(ey,xE);

end

%% function to return slope difference given point on cc and mid pt

function val = cctie(x,xM,yM,rx,ey,cc\_y)

xE = x;

yR = ppval(cc\_y,x);

xR = ppval(rx,yR);

yE = ppval(ey,xE);

val = (yE-yR)\*(xE-xM) - (yE-yM)\*(xE-xR);

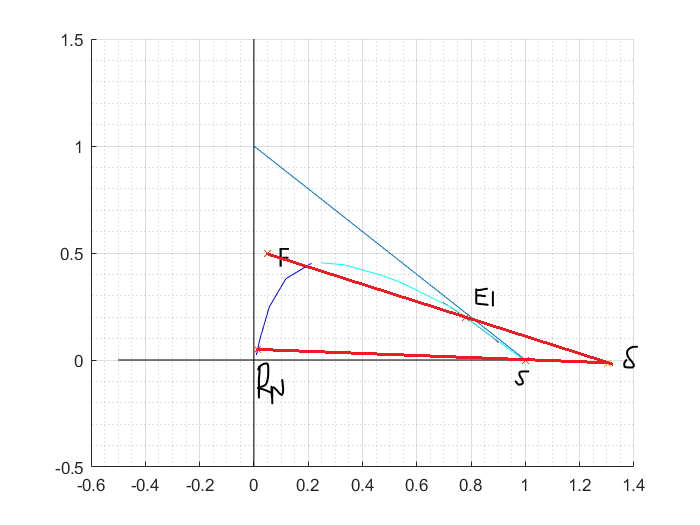
end

# Question 2

## **Determining Delta point.**

**X coordinate**- composition of **water**

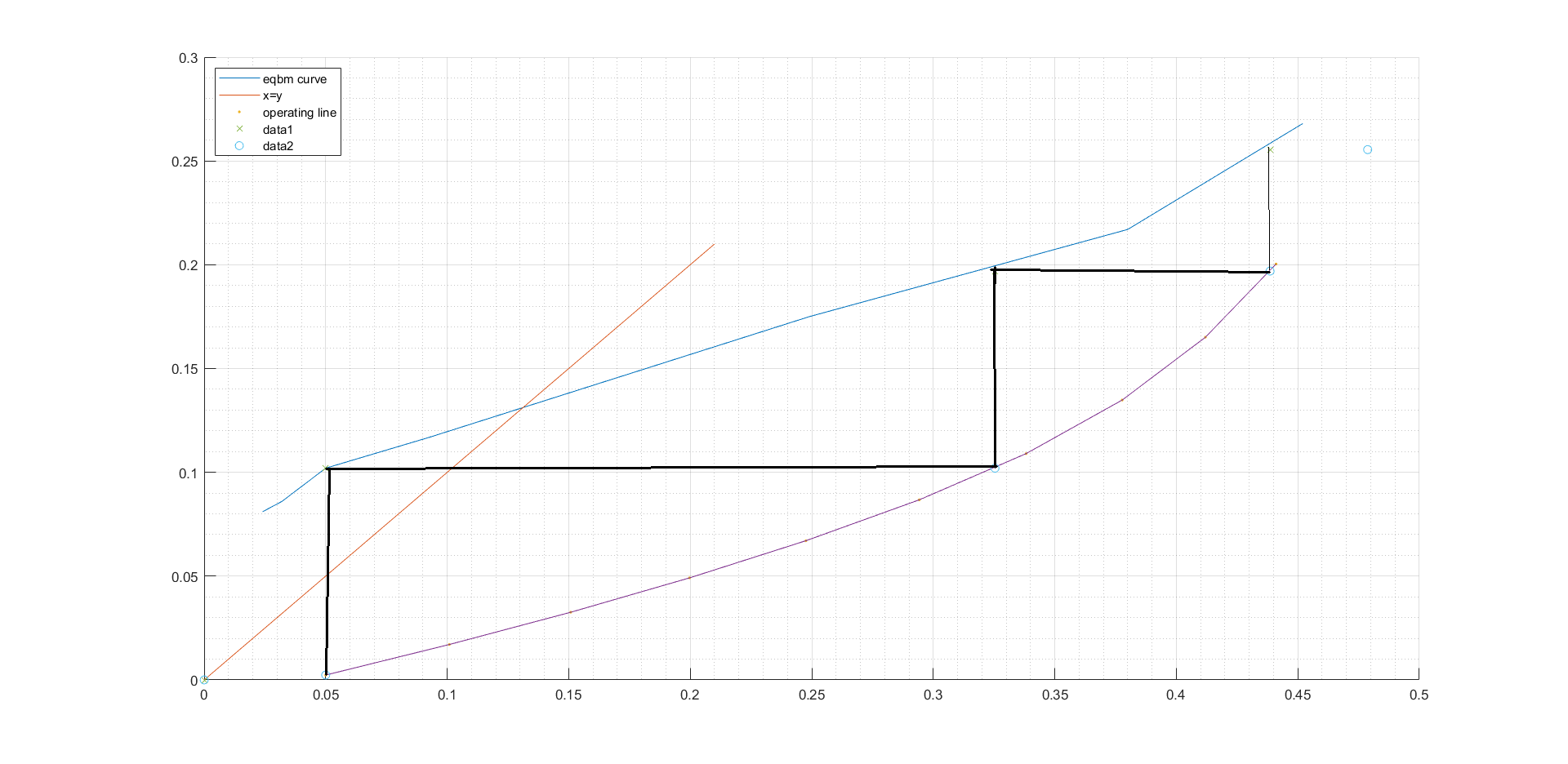
**Y coordinate**- composition of **alcohol**



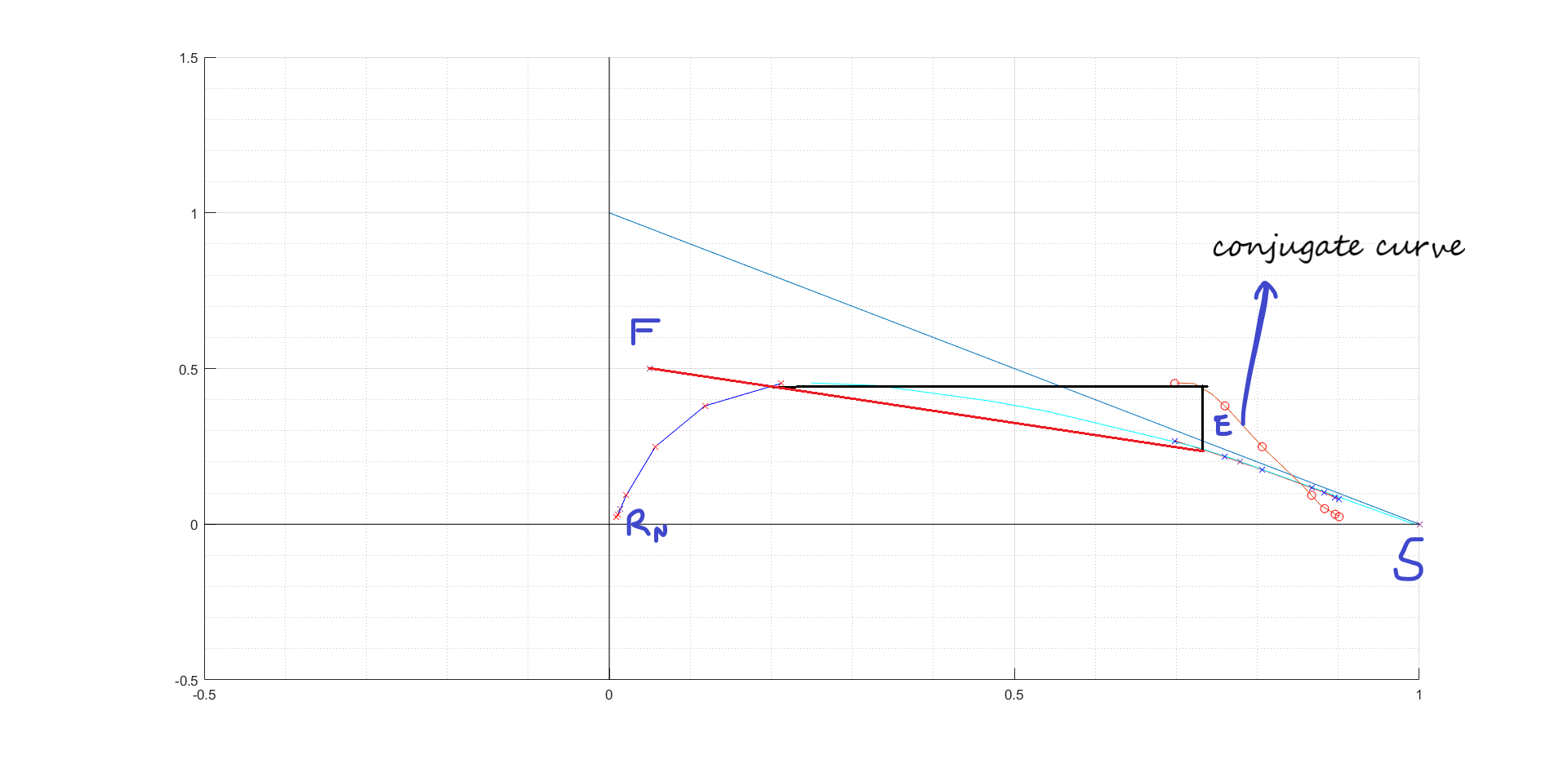
## **Drawing stages after projecting the right angled diagram to xy.**

X coordinate- composition of alcohol in raffinate

Y coordinate- composition of alcohol in extract



## **Determining the maximum extract limit**



## **Code**

% Ether-Carrier, Alcohol-A, Water-Solvent

clear; close all;

%% Equilibrium data

ar = 0.01\*[2.4 3.2 5 9.3 24.9 38 45.2];

sr = 0.01\*[0.9 1.1 1.4 2.1 5.7 11.8 21.2];

raff = spline(ar,sr);

r = spline(sr,ar);

ae = 0.01\*[8.1 8.6 10.2 11.7 17.5 21.7 26.8];

se = 0.01\*[90.1 89.6 88.3 86.7 80.6 76 69.8];

ext = spline(ae,se);

e = spline(se,ae);

%additional points

a\_add = 0.01\*[45.37 44.55 39.57 36.23 24.74 21.33 0 0];

s\_add = 0.01\*[24.93 33 47.01 54.11 72.52 76.61 99.4 0.5];

hold on;

plot(sr,ar,'b',se,ae,'r',s\_add,a\_add,'c');

%Completing the triangle

plot(linspace(0,1,5),1-linspace(0,1,5));

%Plotting the axes

plot(zeros(1,2),linspace(-0.5,1.5,2),'k');

plot(linspace(-0.5,1,2),zeros(1,2),'k');

plot(sr,ar,'rx',se,ae,'bx');

grid on; grid minor;

%% conjugate curve

curve = spline(se,ar);

plot(linspace(0.7,0.9,10),ppval(curve,linspace(0.7,0.9,10)),se,ar,'ro');

%% Given Feed Data

yF =0.5;

xF = 0.05;

yRN = 0.05;

xRN = ppval(raff,yRN);

yE1 = 0.2;

xE1 = ppval(ext,yE1);

plot([xF,xE1,xRN,1],[yF,yE1,yRN,0],'x');

%% Finding Del point

RS = polyfit([xRN,1],[yRN,0],1);

FE = polyfit([xE1,xF],[yE1,yF],1);

fun = @(x)(polyval(RS,x)-polyval(FE,x));

delx = fsolve(fun,0);

dely = polyval(RS,delx);

%% Drawing lines from the del point

%no. of lines

n = 10;

m = linspace(FE(1),RS(1),n);

line = @(y)(delx + 1./m.\*(y-dely));

raff\_int = @(y)(line(y)-ppval(raff,y));

% ext\_int = @(y)(line(y)-ppval(ext,y));

xA = fsolve(raff\_int,zeros(1,n)+1);

xB = line(xA);

line1 = @(x)(m.\*(x-delx)+dely);

% raff\_int = @(x)(line(x)-ppval(r,x));

[p,s] = polyfit(se,ae,4);

ext\_int = @(x)(line1(x)-polyval(p,x));

yB = fsolve(ext\_int,zeros(1,n)+1);

yA = line1(yB);

plot([xF,xE1,xRN,1],[yF,yE1,yRN,0],'x');

% % Resizing since only 7 points were properly obtained

% yB = yB(1:7);

% yA = yA(1:7);

% xA = xA(1:7);

% xB = xB(1:7);

figure();

hold on;

plot(sr,ar,'b',se,ae,'r',s\_add,a\_add,'c');

plot(delx,dely,'x');

%Completing the triangle

plot(linspace(0,1,5),1-linspace(0,1,5));

%Plotting the axes

plot(zeros(1,2),linspace(-0.5,1.5,2),'k');

plot(linspace(-0.5,1,2),zeros(1,2),'k');

grid on; grid minor;

plot(xB,xA,'rx',yB,yA,'bx');%,linspace(0.7,1,10),polyval(p,linspace(0.7,1,10)),'k');

%% Equilibrium projection

xeqbm = ar;

yeqbm = ae;

figure();

hold on;

plot(xeqbm,yeqbm,[0.21,0],[0.21,0],xA,yA,'.',xA,yA);

grid on; grid minor;

legend('eqbm curve','x=y','operating line','Location','northwest');

%% Stepping

i = 1;

xy = spline(yeqbm,xeqbm);

OL = spline(xA,yA);

xcoords = zeros(1,7);

ycoords = zeros(1,7);

xcoords2 = zeros(1,8);

ycoords2 = zeros(1,8);

x0 = 0.05;

y0 = spline(xA,yA,x0);

yx = spline(xeqbm,yeqbm);

LO = spline(yA,xA);

xcoords2(1)= x0;

ycoords2(1) = y0;

while y0 < 0.2

y0 = ppval(yx,x0);

xcoords(i) = x0;

ycoords(i)=y0;

x0 = ppval(LO,y0);

xcoords2(i+1) = x0;

ycoords2(i+1) = y0;

i = i + 1;

end

plot(xcoords,ycoords,'x',xcoords2,ycoords2,'o');