Computation Module for number of stages and minimum reflux using MATLAB

BY : AMIT KUMAR (19JE0115)

****This problem statement has been taken****

Ponchot Savarit Method: Distillation column, binary mixture

1000 kg/hr of a mixture containing 42 mole percent heptane and 58 mole percent ethyl benzene is to be fractionated to a distillate containing 97 mole percent heptane and a residue containing 99 mole percent ethyl benzene using a total condenser and feed at its saturated liquid condition. The enthalpy-concentration data for the heptane-ethyl benzene at 1 atm pressure are as follows:

xheptane	0	0.08	0.18	0.25	0.49	0.65	0.79	0.91	1.0
yheptane	0	0.28	0.43	0.51	0.73	0.83	0.90	0.96	1.0
Hl (kJ/kmol) x 10-3	24.3	24.1	23.2	22.8	22.05	21.75	21.7	21.6	21.4
Hv (kJ/kmol) x 10-3	61.2	59.6	58.5	58.1	56.5	55.2	54.4	53.8	53.3

GOVERNING EQUATIONS.

- **Variables:**

- Liquid flow rate across plates : L_n
- Vapor flow rate across plates :V_{n+1}
- Mole ratio in vapor and liquid phase of each tray :y_{n+1} , x_n
- Molar enthalpy of vapor and liquid phase across each tray : H_{Vn+1} , H_{Ln} .

Constants:

- Reflux ratio (R) , Top and Lower product flow rate (D & W) and their respective mole ratios(x_d , x_w) , Heat Duties (Qb,Qd), Feed Flow rate (F, z_f) .

RECTIFYING SECTION:

$V_{n+1} = L_n + D$

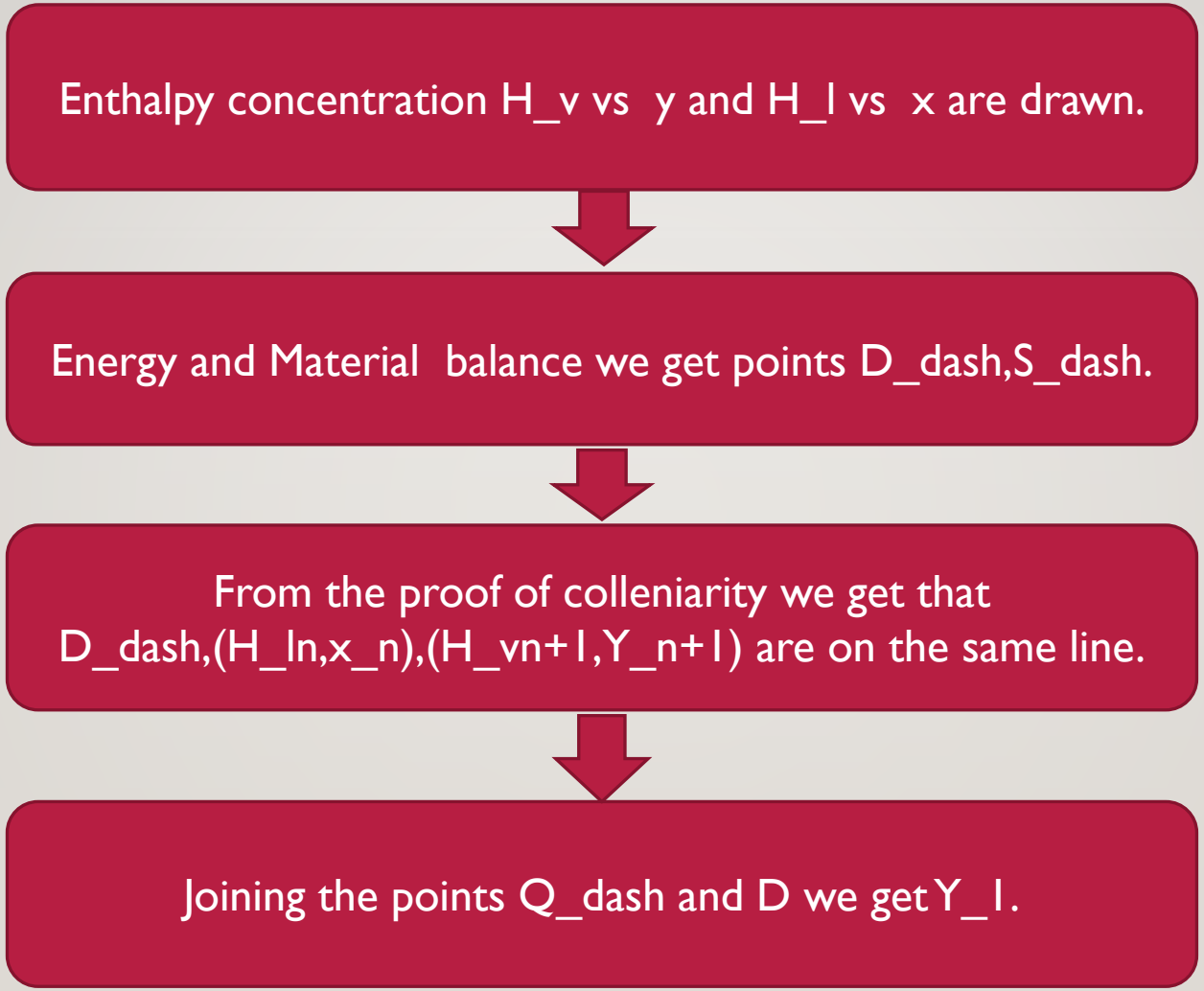
- $V_{n+1} * y_{n+1} = L_n * x_n + D * x_d$
- $V_{n+1} * H_{Vn+1} = L_n * H_{Ln} + Qd'$
- $y = interp1(xe, ye, x_n)$
- $H_{Ln} = interp1(xe, Hl, x_n)$
- $H_{Vn+1} = interp1(ye, Hv, y_{n+1})$

STRIPPING SECTION:

$V_{m+1} = L_m - W$

- $V_{m+1} * y_{m+1} = L_m * x_m - W * x_W$
- $V_{m+1} * H_{Vm+1} = L_m * H_{Lm} - Qc'$
- $y = interp1(xe, ye, x_m)$
- $H_{Lm} = interp1(xe, Hl, x_m)$
- $H_{Vm+1} = interp1(ye, Hv, y_{m+1})$

Flowchart for graphical Ponchon savarit:



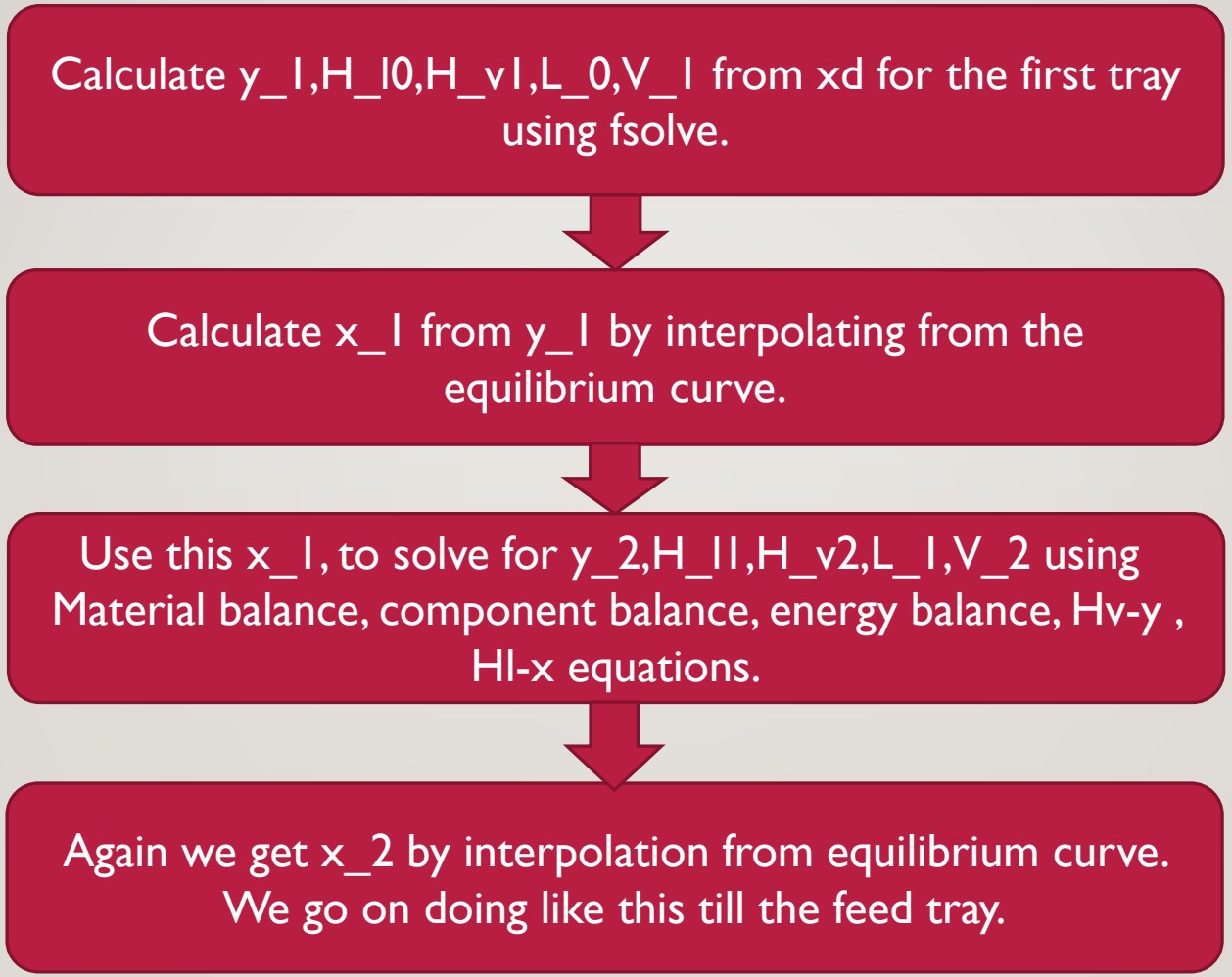
From the Y_1 using euillibrium data we get x_1.

From collinearity principle joining point L_1 and D_dash we get Y_2.

Repeat till it crosses the feed line and rectifying section is done.

Joining the W and S_dash we get y_n and using equillibrium data we get x_n , same repeated as rectifying section.

Flowchart for Numerical ponchon savarit:



When we go past the feed tray, the governing equations will change but the procedure will remain the same.

The main problem we faced is when we tried to write the equations for a particular tray. the material balance, component balance, energy balance equations required y_n+1 and x_n whereas the equilibrium relation is valid between y_n and x_n. We tackled the problem by using a global variable i.e. we are sending x_n from the previous iteration to be used in the current iteration in equilibrium relation

Steps Involved -

1. Using the classic Ponchon-Savarit method we created the heat map .
2. Finding out the number of the stages and feed plate tray from the classic Ponchon-Savarit.
3. Based on the number of stages and the feed plate entry we write the equations for each stage. For each stage we know 'x' value (which we get from previous stage or for 1st stage it will be xd) and then there are 5 variables (unknowns) y, Hl, Hv, L, V which will be solved by fsolve. Then we can find x for next stage from equilibrium curve using y from current stage.
4. Writing the material balance, component balance, energy balance, HL-x curve, Hv-y curve and x-y curve we get the required equations to solve the variables we have.
5. At each stage using fsolve to solve the set of the equations we get 5 variables solution using equilibrium curve function we get x using y.

6. Same process is repeated till we reach the feed plate, till this step we use rectifying section equations.

7. After crossing the feed plate now the same process is repeated for stripping section using stripping section equations.

8. After collecting values of x, y, H_l, H_v, V, L at each stage we draw a heat map and compare it with the heat map that we got using simple Ponchon-Savarit.

Attaching MATLAB code below :

```
clear;
close;
clc;

global X

% information given in the question
F = 1000/103.48; % Kmol/hr
zf = 0.42;  % feed conc.
xf = 0.42; % feed conc.
xd = 0.97; % distillate composition
xw = 0.01; % bottoms composition
R = 2.5; % reflux ratio
D = 4.2681; % distillate in kmol/hr, calculated from overall material balance
W = 5.7256; % bottoms in kmol/hr, calculated from overall material balance

% no. of stages under observation in trial and error method
n=11;

% feed tray at no. 6
feed_tray=6;

% equilibrium data
xe = [0 0.08 0.18 0.25 0.49 0.65 0.79 0.91 1];
ye = [0 0.28 0.43 0.51 0.73 0.83 0.9 0.96 1];

E_curve_y = polyfit(xe,ye,8);
```

```
xe_new = 0:0.001:1;
for k = 1:length(xe_new)
    ye_new(k) = polyval(E_curve_y,xe_new(k));
end

% recifying section functions
fun1 = @Rectifying_equations;

% calculation of V1 and other varibles for first tray from given xd
X = xd;
x0 = [0.5002,10.82,15.09,21.63,53.85];
y = fsolve(fun1,x0);

% variables to store the data at each tray
x_tray(1)=X;
y_tray(1)=y(1);
L_tray(1)=y(2);
V_tray(1)=y(3);
Hl_tray(1)=y(4);
Hv_tray(1)=y(5);

% for all the tray in rectifying section we use the
% same strategy, calculate x from previous iteration
% from equilibrium data, and then in the current iteration
% use this x value to calculate variables namely
% y,L,V,Hl,Hv
```

```
for i= 1:1:feed_tray
    X = interp1(ye_new,xe_new,y(1));
    x_tray(i)=X;
    fun1 = @Rectifying_equations;
    x0 = [0.5002,10.82,15.09,21.63,53.85];
    y = fsolve(fun1,x0);
    y_tray(i)=y(1);
    L_tray(i)=y(2);
    V_tray(i)=y(3);
    Hl_tray(i)=y(4);
    Hv_tray(i)=y(5);
end

% for all the tray in stripping section we use the
% same strategy yet again, calculate x from previous iteration
% from equilibrium data, and then in the current iteration
% use this x value to calculate variables namely
% y,L,V,Hl,Hv

for i= (feed_tray+1):1:n
    X = interp1(ye_new,xe_new,y(1));
    x_tray(i)=X;
    fun2 = @Stripping_equations;
    x0 = [0.5002,10.82,15.09,21.63,53.85];
    y = fsolve(fun2,x0);
    y_tray(i)=y(1);
    L_tray(i)=y(2);
```

```
    L_tray(i)=y(2);
    V_tray(i)=y(3);
    Hl_tray(i)=y(4);
    Hv_tray(i)=y(5);
end


x_tray = x_tray';
y_tray = y_tray';
L_tray = L_tray';
V_tray = V_tray';
Hl_tray = Hl_tray';
Hv_tray = Hv_tray';

figure(1)
x_names = {'x (equi)', 'y (equi)','L','V','HL','HV'};
y_names = 1:1:11;
h = heatmap(x_names,y_names,[x_tray,y_tray,L_tray,V_tray,Hl_tray,Hv_tray]);

h.Title = 'By using fsolve to solve equations at every stage';
h.YLabel = 'Composition of components in each stage';
h.XLabel = 'Components in liquid phase and vapour phase';
```

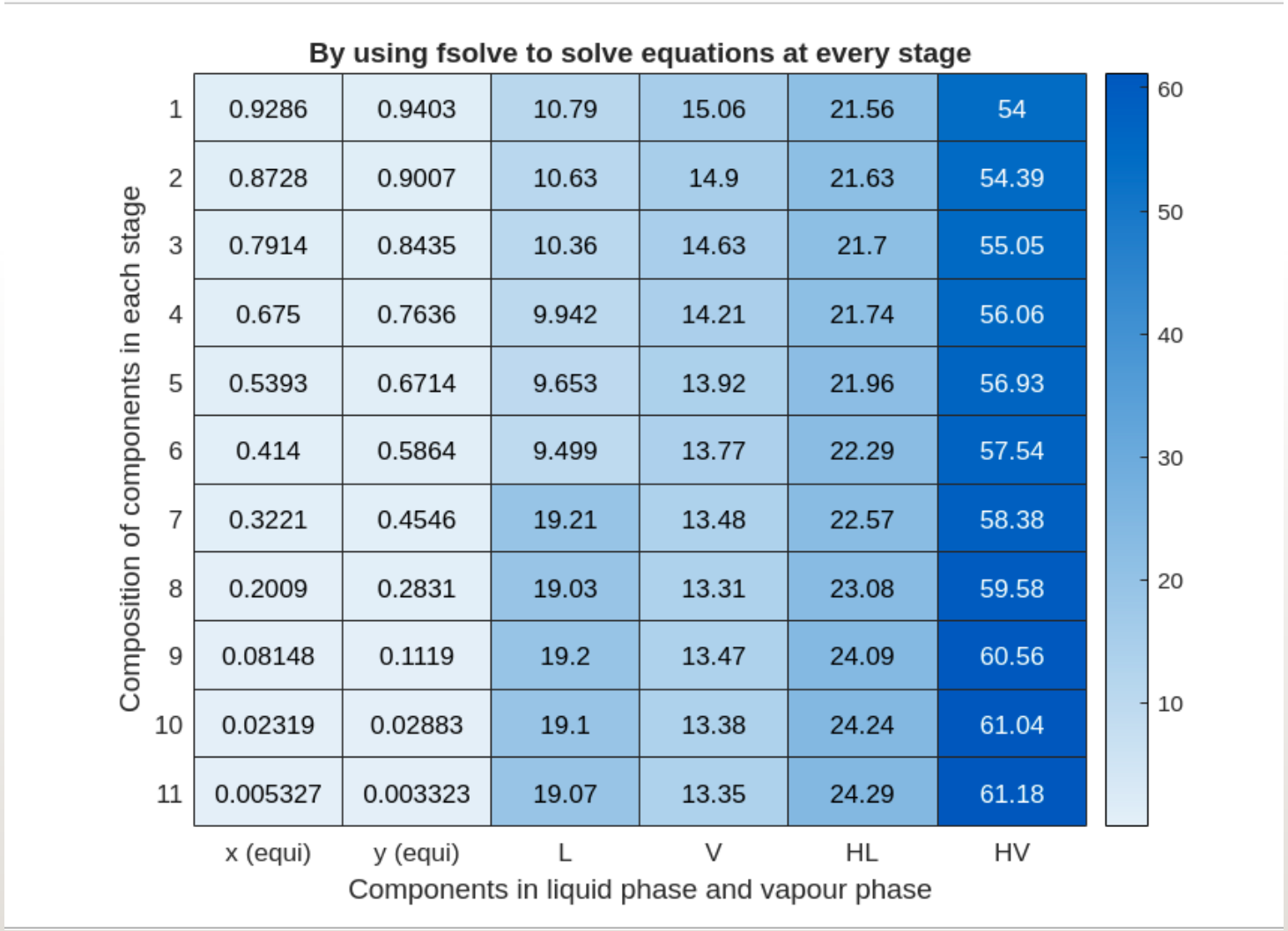
```
% rectifying section equations
function F = Rectifying_equations(x)

    global X

    % information given in the question
    zf = 0.42;
    xf = 0.42;
    xd = 0.97;
    xw = 0.01;
    R = 2.5;
    D = 4.2681;
    W = 5.7256;

    % data given in question
    xe = [0 0.08 0.18 0.25 0.49 0.65 0.79 0.91 1];
    ye = [0 0.28 0.43 0.51 0.73 0.83 0.9 0.96 1];
    Hl = [24.3 24.1 23.2 22.8 22.05 21.75 21.7 21.6 21.4];
    Hv = [61.2 59.6 58.5 58.1 56.5 55.2 54.4 53.8 53.3];


    % fitting a polynomial function for equilibrium curve
    E_curve_y = polyfit(xe,ye,8);

    xe_new = 0:0.001:1;
    for k = 1:length(xe_new)
        ye_new(k) = polyval(E_curve_y,xe_new(k));
```

```
    end

    Hl_p = polyfit(xe,Hl,1);
    Hv_p = polyfit(ye,Hv,1);
    Hw = polyval(Hl_p,xw);
    Hv2 = polyval(Hv_p,xw);
    Hd = polyval(Hl_p,xd);
    Hv1 = polyval(Hv_p,xd);
    Hf = polyval(Hl_p,xf);

    qd_dash = R*(Hv1-Hd) + Hv1;
    slope_F = (qd_dash - Hf)/(xd-xf);
    intercept_F = -slope_F*xd + qd_dash;
    qw_dash = slope_F*xw + intercept_F;

    %x(1) = y
    %x(2) = Ln
    %x(3) = Vn
    %x(4) = HLn
    %x(5) = HVn

    % material balance equation
    F(1) = x(3) - x(2) - D;

    % component balance equation
    F(2) = x(3)*x(1) - x(2)*X - D*xd;

```

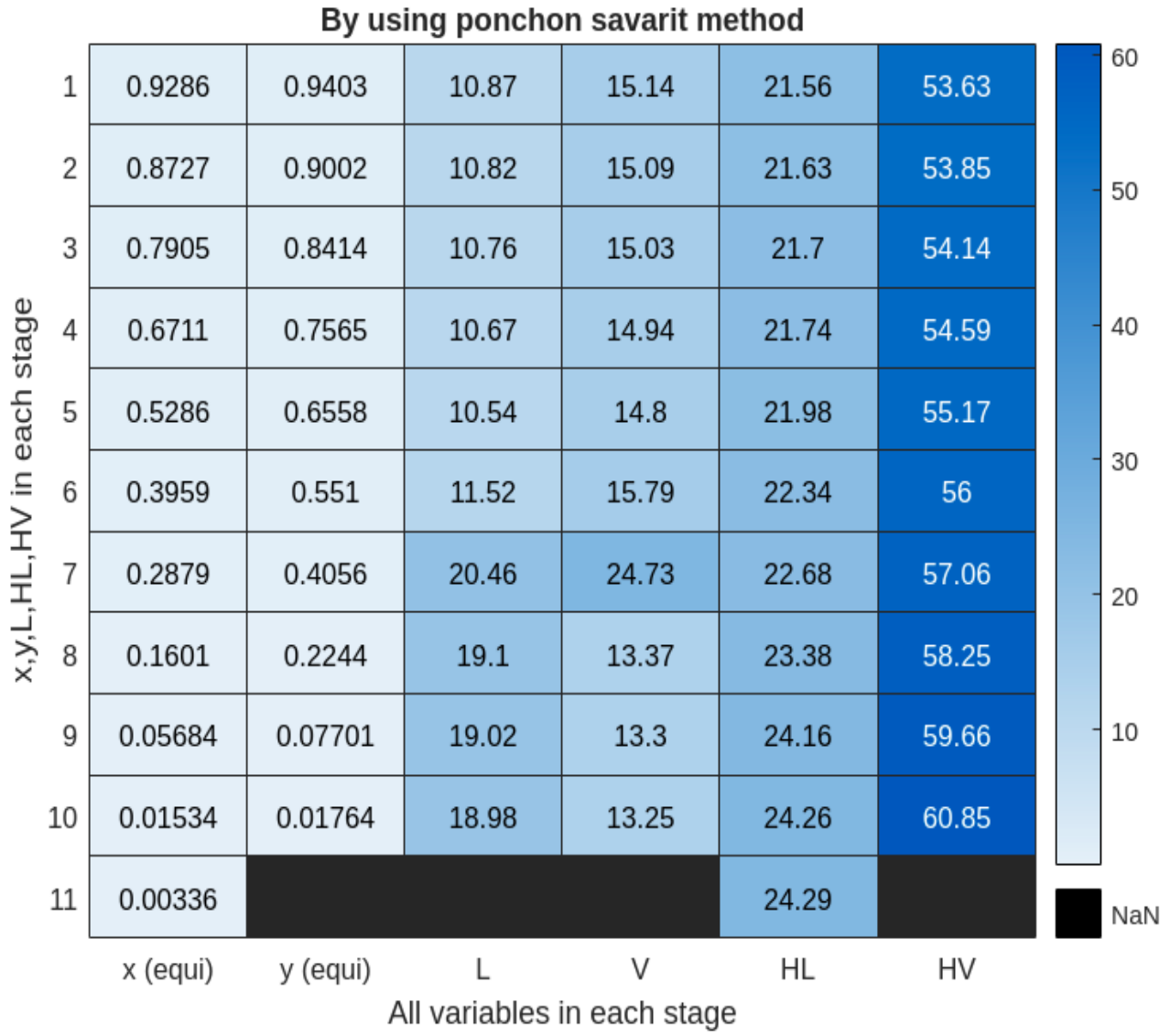
```
    % interpolation of Hl
    F(4) = x(4) - interp1(xe,Hl,X);

    % interpolation of Hv
    F(5) = x(5) - interp1(ye,Hv,x(1));
end


% stripping section equation
function F = Stripping_equations(x)

    global X

    % information given in the question
    zf = 0.42;
    xf = 0.42;
    xd = 0.97;
    xw = 0.01;
    R = 2.5;
    D = 4.2681;
    W = 5.7256;

    % data given in question
    xe = [0 0.08 0.18 0.25 0.49 0.65 0.79 0.91 1];
    ye = [0 0.28 0.43 0.51 0.73 0.83 0.9 0.96 1];
    Hl = [24.3 24.1 23.2 22.8 22.05 21.75 21.7 21.6 21.4];
    Hv = [61.2 59.6 58.5 58.1 56.5 55.2 54.4 53.8 53.3];
```

```
Hl = [24.3 24.1 23.2 22.8 22.05 21.75 21.7 21.6 21.4];
Hv = [61.2 59.6 58.5 58.1 56.5 55.2 54.4 53.8 53.3];


% polynomial fit for equilibrium
E_curve_y = polyfit(xe,ye,8);

xe_new = 0:0.001:1;
for k = 1:length(xe_new)
    ye_new(k) = polyval(E_curve_y,xe_new(k));
end

Hl_p = polyfit(xe,Hl,1);
Hv_p = polyfit(ye,Hv,1);
Hw = polyval(Hl_p,xw);
Hv2 = polyval(Hv_p,xw);
Hd = polyval(Hl_p,xd);
Hv1 = polyval(Hv_p,xd);
Hf = polyval(Hl_p,xf);

qd_dash = R*(Hv1-Hd) + Hv1;
slope_F = (qd_dash - Hf)/(xd-xf);
intercept_F = -slope_F*xd + qd_dash;
qw_dash = slope_F*xw + intercept_F;

%x(1) = y
%x(2) = Ln
```

```
209
210 -     qd_dash = R*(Hv1-Hd) + Hv1;
211 -     slope_F = (qd_dash - Hf)/(xd-xf);
212 -     intercept_F = -slope_F*xd + qd_dash;
213 -     qw_dash = slope_F*xw + intercept_F;
214
215       %x(1) = y
216       %x(2) = Ln
217       %x(3) = Vn
218       %x(4) = HLn
219       %x(5) = HVn
220
221       % material balance equation
222 -     F(1) = x(2) - x(3) - W;
223
224       % component balance equation
225 -     F(2) = x(2)*X - x(3)*x(1) - W*xw;
226
227       % energy balance equation
228 -     F(3) = x(2)*x(4) - x(3)*x(5) - qw_dash*W;
229
230       % interpolation of Hl from x
231 -     F(4) = x(4) - interp1(xe,Hl,X);
232
233       % interpolation of Hv from y
234 -     F(5) = x(5) - interp1(ye,Hv,x(1));
235 - end
```



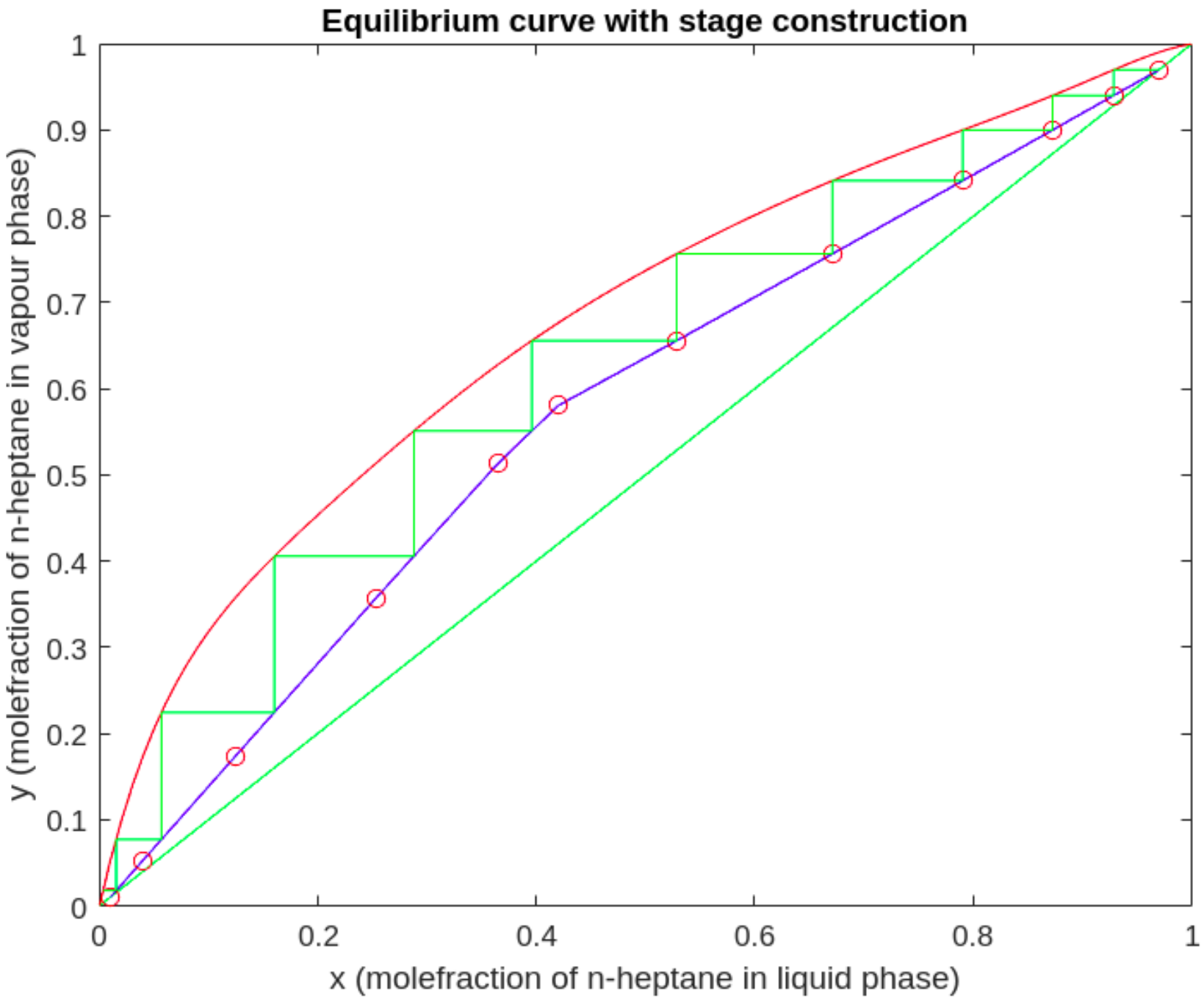
Heat map Using our code

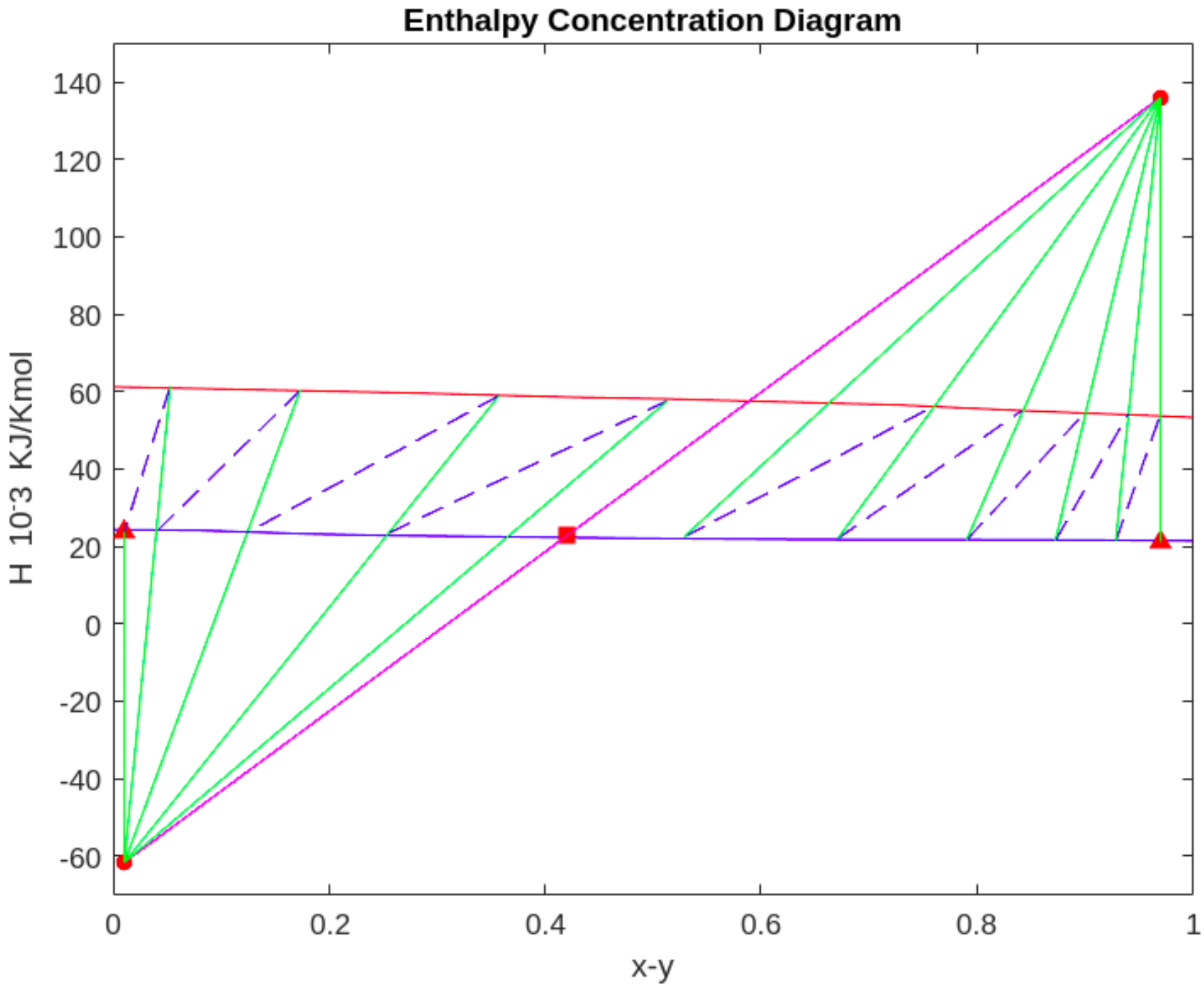


Heat map using classical ponchon savarit

CONCLUSION

- By comparing the heat maps from numerical and graphical ponchon savarit method (above two pictures) we conclude that our approach produce the results close to the ponchon


Equilibrium curve with stage construction
y (molefraction of n-heptane in vapour phase)
x (molefraction of n-heptane in liquid phase)
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
0
0.2
0.4
0.6
0.8
1


Enthalpy Concentration Diagram
140
120
100
80
60
40
20
0
-20
-40
-60
H 10^3 KJ/Kmol
0
0.2
0.4
0.6
0.8
1
x-y