9.9 (***Number of stages for*** Cross current ***leaching***)[3] A slurry of a seaweed is to be leached with hot water to recover a valuable protein. The slurry (48.1% solids, 49% water and 2.9% protein) enters a countercurrent leaching battery at a rate of 400 kg/h. Hot water enters at the other end at a rate of 500 kg/h. The underflow leaving the unit may have a maximum of 0.2% protein. Calculate the number of ideal contact stages required. The following 'equilibrium data' generated by laboratory tests (Chopey: *Chem. Engg. Calculations*, McGraw-Hill, 1994) may be used.

Extract (overflow) concentration, mass fraction			*Slurry (underflow) concentration, mass fraction*		
Water	*Protein*	*Solids*	*Water*	*Protein*	*Solids*
0.952	0.046	0.002	0.542	0.026	0.432
0.967	0.032	0.001	0.564	0.019	0.417
0.979	0.021	0.00	0.586	0.013	0.401
0.989	0.011	0.00	0.5954	0.0066	0.398
0.994	0.006	0.00	0.5994	0.0036	0.397
0.998	0.002	0.00	0.6028	0.0012	0.396

Perform the calculations for the cross current operation:

1. Check the conversion of the given data using the transformation as mentioned Plot the solid-liquid Equilibrium data in Ponchon Savarit diagram (plot Xc vs z and Yc vs Z).
2. Locate the **Fdash and Sdash** point. Write a MATLAB code for showing the graphical stages for crosscurrent operation as shown.
3. Calculate the concentration of the solute in mixture x_m from the material balance and locate the M point.
4. Draw the FM line and the tie lines. Calculate the overflow and underflow concentrations for a four-stage cross current operation.
5. Calculate the amount of overflow and underflow and fraction of protein separated from the feed for every stage.

Hints:

```
%SETTING UP THE UNDERFLOW AND OVERFLOW

for i = 1:length(wA)
    xc(i) = (wC_dash(i))/(wC_dash(i)+wB_dash(i));
    Z(i) = (wA(i))/(wB(i)+wC(i));
    yc(i) = (wC(i))/(wC(i)+wB(i));
    z(i) = (wA_dash(i))/(wB_dash(i)+wC_dash(i));
end
```

```
%DETERMINING THE UNDEFLOW AND OVERFLOW POINTS

    Ly = polyval(……);
    Vy = polyval(……);

    a6 = plot([…..],[…..], '--ok');
    text(xM, Ly, ['L' num2str(i) '']);

    text(xM, Vy, ['V' num2str(i) '']);

    %SOLVING FOR AMOUNT OF UNDERFLOW AND OVERFLOW

    L_dash = ……
    V_dash = ....

    %NEW F IS THE UNDERFLOW FROM THE LAST STAGE

    F_dash = L_dash;
    xCF = xM;
    zF = Ly;
```