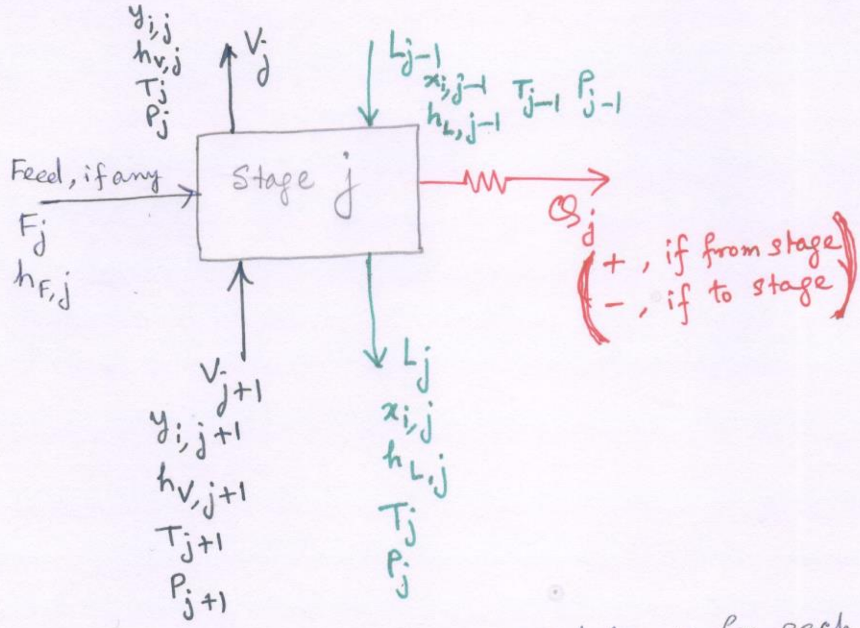
Sum-Rates method for absorption and stripping



M-type Equations – Material balance for each component (C equations for each stage)

For i^{th} component, j^{th} stage

No feed added to j^{th} stage; No reflux; No reflux

$$M_{ij} = L_{j-1} x_{i,j-1} + V_{j+1} y_{i,j+1} + \cancel{F_j} z_{i,j} - (L_j + \cancel{U_j}) x_{i,j} - (V_j + \cancel{W_j}) y_{i,j} = 0$$

E-type Equations – Phase equilibrium relation for each component (C equations for each stage)

$$E_{ij} = y_{i,j} - K_{i,j} x_{i,j} = 0$$

S-type Equation – Mole fraction summations (One for each stage, each phase

$$(S_y)_j = \sum_{i=1}^{C} y_{i,j} - 1.0 = 0$$

$$(S_x)_j = \sum_{i=1}^{C} x_{i,j} - 1.0 = 0$$

H-type Equation: Energy balance, one for each stage

$$H_j = L_{j-1} h_{L,j-1} + V_{j+1} h_{V,j+1} + \cancelto{0}{F_j h_{F_j}} - L_j h_{L,j} - V_j h_{V,j} - Q_j = 0$$

Here, h represents enthalpy per unit mole for the stream, identified in the subscript.

Solution of heat balance equation for temperature

$$H_j = L_{j-1} h_{L,j-1} + V_{j+1} h_{v_{j+1}} - L_j h_{L_j} - V_j h_{v_j} - Q_j = 0$$

Enthalpy correlations are available as (Hougen & Watson, Chemical Process Principles) for mixtures

$$h_{v,j} = \sum_{i=1}^{c} y_{i,j} (A_i + B_i T_j + C_i T_j^2)$$

$$h_{L,j} = \sum_{i=1}^{c} x_{i,j} (a_i + b_i T_j + c_i T_j^2)$$

For individual components $i = 1, 2, \cdots, C$ A, B, C, a, b, c are tabulated.

$$\Rightarrow \frac{\partial h_{v,j}}{\partial T_j} = \sum_{i=1}^{c} y_{i,j} (B_i + 2C_i T_j)$$

$$\frac{\partial h_{L,j}}{\partial T_j} = \sum_{i=1}^{c} x_{i,j} (b_i + 2c_i T_j)$$

Since $h_{v,j}$ and $h_{L,j}$ are function of T_j only, and H_j contains $h_{L,j-1}$, $h_{v,j+1}$, $h_{L,j}$, and $h_{v,j}$ terms only if L, V, and Q for each stage are known apriori.

$\Rightarrow H_j$ will be a function of T_{j-1}, T_j, and T_{j+1} only. That is H_j will depend on Temperature values of j^{th} stage, one stage below j^{th} stage, and one stage above j^{th} stage.

$$\Rightarrow H_j (T_{j-1}, T_j, T_{j+1}) = 0$$

By Taylor Series expansion, and ignoring terms beyond first order

$$H_j \left[(T_{j-1} + \Delta T_{j-1}), (T_j + \Delta T_j), (T_{j+1} + \Delta T_{j+1}) \right] = H_j (T_{j-1}, T_j, T_{j+1}) + \left(\frac{\partial H_j}{\partial T_{j-1}}\right) \Delta T_{j-1} + \left(\frac{\partial H_j}{\partial T_j}\right) \Delta T_j + \left(\frac{\partial H_j}{\partial T_{j+1}}\right) \Delta T_{j+1}$$

A Newton-Raphson scheme can be implemented by setting the left-hand side of above equation i.e., $H_j\left[(T_{j-1}+\Delta T_{j-1}), (T_j+\Delta T_j), (T_{j+1}+\Delta T_{j+1})\right]$ to zero, and look for solution of ΔT_{j-1}, ΔT_j, and ΔT_{j+1} based on initial guess of T_{j-1}, T_j, and T_{j+1}.

This is further noted that

$$\frac{\partial H_j}{\partial T_{j-1}} = L_{j-1}\frac{\partial h_{L,j-1}}{\partial T_{j-1}}$$

$$= L_{j-1}\sum_{i=1}^{c} x_{i,j-1}(b_i + 2c_i T_{j-1}) = f(T_{j-1})$$

All other terms e.g., $h_{V,j+1}$, $h_{L,j}$, $h_{V,j}$ do not depend on T_{j-1}

and similarly,

$$\frac{\partial H_j}{\partial T_j} = -L_j\frac{\partial h_{L,j}}{\partial T_j} - V_j\frac{\partial h_{V,j}}{\partial T_j} = f(T_j)$$

$$\frac{\partial H_j}{\partial T_{j+1}} = V_{j+1}\frac{\partial h_{V,j+1}}{\partial T_{j+1}} = f(T_{j+1})$$

Therefore, the iteration scheme can be set as follows

$$\left(\frac{\partial H_j}{\partial T_{j-1}}\right)^{(r)}\Delta T_{j-1}^{(r)} + \left(\frac{\partial H_j}{\partial T_j}\right)^{(r)}\Delta T_j^{(r)} + \left(\frac{\partial H_j}{\partial T_{j+1}}\right)^{(r)}\Delta T_{j+1}^{(r)} = -H_j^{(r)}$$

where $\Delta T_j^{(r)} = T_j^{(r+1)} - T_j^{(r)}$

Since every stage has one such relation, for N-stages, there will be N number of equations.

$$\begin{bmatrix} \ddots & \ddots & & & 0 \\ \ddots & \ddots & \ddots & & \\ & \frac{\partial H_j}{\partial T_{j-1}} & \frac{\partial H_j}{\partial T_j} & \frac{\partial H_j}{\partial T_j} & \\ & & \ddots & \ddots & \ddots \\ 0 & & & \ddots & \ddots \end{bmatrix}^{(r)} \begin{bmatrix} \Delta T_1 \\ \Delta T_2 \\ \vdots \\ \Delta T_{j-1} \\ \Delta T_j \\ \Delta T_{j+1} \\ \vdots \\ \Delta T_{N-1} \\ \Delta T_N \end{bmatrix}^{(r)} = \begin{bmatrix} -H_1 \\ -H_2 \\ \vdots \\ -H_j \\ \vdots \\ -H_{N-1} \\ -H_N \end{bmatrix}^{(r)}$$

Solution of tri-diagonal matrix equation by Thomas Algorithm

$$\begin{bmatrix} B_1 & C_1 & 0 & \cdots & 0 & \cdots & & \\ A_2 & B_2 & C_2 & 0 & \cdots & & & \\ 0 & A_3 & B_3 & C_3 & 0 & \cdots & & \\ 0 & 0 & A_4 & B_4 & C_4 & 0 & \cdots & \\ & & & \ddots & \ddots & \ddots & & \\ & & & & A_{N-1} & B_{N-1} & C_{N-1} \\ & & & & 0 & A_N & B_N \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ \\ x_{N-1} \\ x_N \end{bmatrix} = \begin{bmatrix} D_1 \\ D_2 \\ \vdots \\ \vdots \\ D_{N-1} \\ D_N \end{bmatrix}$$

Essentially Gaussian Elimination without operating on 'zeros'.

$$B_1 x_1 + C_1 x_2 = D_1$$

$$\Rightarrow x_1 = \frac{D_1 - C_1 x_2}{B_1} = \left(\frac{D_1}{B_1}\right) - \left(\frac{C_1}{B_1}\right) x_2 = q_1 - p_1 x_2$$

$$A_2 x_1 + B_2 x_2 + C_2 x_3 = D_2$$

$$\Rightarrow x_2 = \frac{D_2 - A_2 q_1}{B_2 - A_2 p_1} - \left(\frac{C_2}{B_2 - A_2 p_1}\right) x_3$$

$$= q_2 - p_2 x_3$$

General Form

$$x_j = q_j - p_j x_{j+1}$$

where $p_j = \dfrac{C_j}{B_j - A_j p_{j-1}}$

$$q_j = \frac{D_j - A_j q_{j-1}}{B_j - A_j p_{j-1}}$$

$$\begin{bmatrix} 1 & p_1 & 0 & 0 & \cdots & & \\ 0 & 1 & p_2 & 0 & \cdots & & \\ 0 & 0 & 1 & p_3 & 0 & \cdots & \\ \vdots & & \ddots & \ddots & & & \\ & & & \cdots & 0 & 1 & p_{N-1} \\ 0 & 0 & \cdots & & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ \vdots \\ x_{N-1} \\ x_N \end{bmatrix} = \begin{bmatrix} q_1 \\ q_2 \\ \vdots \\ \vdots \\ q_{N-1} \\ q_N \end{bmatrix}$$

$$\Rightarrow x_N = q_N$$

And recursive back substitution from the bottom-most row

$$x_j = q_j - p_j x_{j+1}$$

$$x_{j-1} = q_{j-1} - p_{j-1} x_j$$

$$\vdots$$

Thomas Algorithm can be employed to solve for $\Delta T_j^{(r)}$ based on ~~initial~~ a set of H_j values, calculated at $T_j^{(r)}$ and the Jacobian matrix evaluated at $T_j^{(r)}$.

Based on $\Delta T_j^{(r)}$, $T_j^{(r+1)}$ can be computed as equal to $T_j^{(r)} + \Delta T_j^{(r)}$, and the H_j and Jacobian matrix can be evaluated at $T_j^{(r+1)}$ to continue the iteration.

Use of ~~Scalar~~ scalar attenuation factor (α) to determine new guess of T_j

$$T_j^{(r+1)} = T_j^{(r)} + \alpha \, \Delta T_j^{(r)}$$

Scalar attenuation is useful to avoid divergence when initial guesses are not reasonably close to the true value.

α can be obtained by minimizing $\sum_{j=1}^{N} \left[H_j^{(r+1)} \right]^2$, or can be set to 1.0 when $\sum_{j=1}^{N} \left[H_j^{(r+1)} \right]^2$ is below a threshold value.

An error criteria will decide when to stop the iteration, and consider $T_j^{(r+1)}$ as the converged values.

Above analysis is based on the fact that L_j and V_j and $x_{i,j}$, $y_{i,j}$ for each ~~step~~ stage are known apriori. However, one needs to work with M-type, E-type, and S-type equations to simultaneously solve for L_j and V_j

Material balance on the lower part of the train, (i.e., over stages 1, 2, ... j-1, j), the part that appears above the j^{th} stage in the schematic drawing, referred earlier

$$L_j = V_{j+1} + \left[\sum_{m=1}^{j} (F_m) \right] - V_1 + L_0$$

This ~~new~~ expression for L_j can be substituted in M-eqn.

$$M_{ij} = \left[V_j + \sum_{m=1}^{j-1} F_m - V_1 + L_0 \right] x_{i,j-1} + V_{j+1} y_{i,j+1} + F_j z_{ij}$$
$$- \left[V_{j+1} + \sum_{m=1}^{j} F_m - V_1 + L_0 \right] x_{i,j} - V_j y_{i,j} = 0$$

Replacing $y_{i,j+1}$ with $K_{i,j+1} x_{i,j+1}$

and $y_{i,j}$ with $K_{i,j} x_{i,j}$

The M-eqn. takes the following form

$$A_j\, x_{i,j-1} + B_j\, x_{i,j} + C_j\, x_{i,j+1} = D_j$$

where

$$A_j = V_j + \sum_{m=1}^{j-1} F_m - V_1 + L_0 \qquad \text{for } 2 \leq j \leq N$$

$$B_j = -\left[V_{j+1} + \sum_{m=1}^{j} F_m - V_1 + L_0 + V_j\, K_{i,j} \right] \qquad \text{for } 1 \leq j \leq N$$

$$C_j = V_{j+1}\, K_{i,j+1} \qquad \text{for } 1 \leq j \leq N-1$$

$$D_j = -F_j\, z_{i,j} \qquad 1 \leq j \leq N$$

$$\text{with } \left. \begin{array}{l} x_{i,0} = 0 \\ V_{N+1} = 0 \end{array} \right\}$$

Therefore, the above equations forms a tridiagonal matrix equation system that can be solved to obtain $x_{i,j}$ for every component, and every stage j. of course the values of V_j and L_j have to be known apriori to solve for $x_{i,j}$

An iterative solution is required using tear variables, and Correcting based on "Sum rate" method.

The final algorithm utilizes two tear variables, T_j and V_j.

Start

Specify P_j, Q_j, N
and feed condition, if any, $K(x, y, P, T)$

Set $k = 1$

Initialize tear variables
T_j, V_j by linear interpolation

Calculate $L_j = V_{j+1} + \cancel{\sum_{m=1}^{j} F_m}^{\,0} - V_1 + L_0$

Compute $x_{i,j}$ using Thomas algorithm
with iterations after normalizing $x_{i,j}$
and updating $K_{i,j}$ until $x_{i,j}$ stops changing

Compute new L_j and then V_j

$$L_j^{(k+1)} = L_j^{(k)} \sum x_{i,j}$$

Application of sum rate method

$$V_j = L_{j-1} - L_N + \cancel{\sum_{m=j}^{N} F_m}^{\,0} + V_{N+1}$$

Normalize $x_{i,j}$ for each stage

$$x_{i,j} = \frac{x_{i,j}}{\sum_{i=1}^{c} x_{i,j}}$$

Compute $y_{i,j} = K_{i,j}\, x_{i,j}$

Normalize $y_{i,j}$

Compute T_j as per the procedure described before for solution of H-eqn.

Check whether

$$\sum_{j=1}^{N} \left[\frac{T_j^{(k)} - T_j^{(k-1)}}{T_j^{(k)}}\right]^2 + \sum_{j=1}^{N} \left[\frac{V_j^{(k)} - V_j^{(k-1)}}{V_j^{(k)}}\right]^2 \le \epsilon$$

No: $k = k + 1$

yes

Converged Solution, EXIT

Industry Criteria
Convergence based on temperature only.

$$\sum_{j=1}^{N} \left[T_j^{(k)} - T_j^{(k-1)}\right]^2 \le 0.01\,N$$

where T is in °F