ABE 55800
Membrane Design Problem
Nathan LeRoy
January 22nd, 2019

The System and Background

The system was defined as many groups of membrane filters connected in series. This is described in Fig. 1.

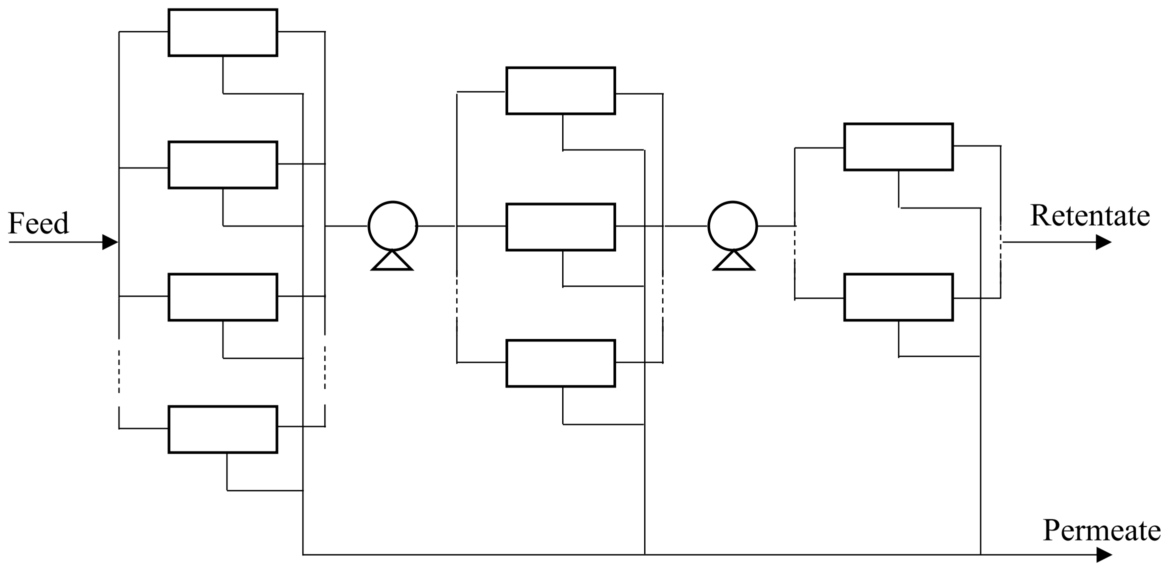


Figure 1. The membrane filter system.

The dilute sugar feed enters the system as shown on the left. The feed is split and simultaneously passed through a group of hollow tube membrane filters. The permeate is discarded and the retentate is then re-pressurized and sent through a new set of filters. This process is repeated with decreasing numbers of filters until the feed reaches the desired 10% (w/w) concentration of glucose. There are many ways to mathematically describe membrane separation. These equations will be laid out.

Permeate Flux – In our system, the permeate is going to be assumed to be water with trace amounts of sugar passing through. We can calculate the permeate flux with the following equation:

$$N_w = A_w(\Delta P - \Delta \pi)$$

Where A_w is the water permeability constant through the membrane, ΔP is the pressure drop across the membrane, and Δπ is the osmotic pressure drop across the membrane. The units of this flux is given as kg H_2O/m^2-s.

Solute Flux – We needed to consider the flux of solute (glucose) across the membrane as well. This phenomenon is not as pronounced, but it is still significant. Otherwise, our solutions will be under-concentrated, and the final solution will be diluted. We can calculate this solute flux, N_s with the following equation:

$$N_s = \frac{N_w C_2}{\rho_w}$$

With units of kg glucose/m^2-s

Pressure Drop – The pressure drop across the membrane can be calculated assuming turbulent flow and using empirically derived equations. We want pressure drop as a function of fluid velocity. This is given as:

$$\Delta P = 4f\rho\frac{\Delta L}{D}\frac{v^2}{2}$$

Where f is the friction factor. This friction factor can also be calculated with empirically relationships. Specifically, we can use the Fanning friction factor:

$$f = \frac{0.079}{Re^{0.25}}$$

Using the equation for Reynold's number, we can combine all these equations to get the final empirical relationship between pressure drop and fluid flow velocity:

$$\Delta P = 2(0.079)\mu^{0.25}\rho^{0.75}\Delta L v^{1.75}\frac{1}{D^{1.25}}$$

Flow Rates – We can use the membrane permeate flux to calculate how much water is being removed from the feed and passing through the membrane. From there, we can calculate the new concentration of the feed and new flow rate. This can be done using a simple material balance.

$$q_{out} = q_{in} - \frac{N_w A}{1000}$$

$$C_2 = \frac{C_1 q_{in}}{q_{out}}$$

We can also calculate the new fluid velocity by dividing the new outflow rate by the cross-sectional area of the tube. This defines the new pressure drop and we can repeat the process until we get to the end of our tube.

Physical Constants – Physical constants for the solutions were calculated at each step to maintain the accuracy of the solution. This includes the density of the solution, and the viscosity of the solution. The density will be calculated using the Choi-Okos equations, and the viscosity is being calculated based on empirical relations between glucose concentration and apparent viscosity (Kim, 2010).

$$\rho_{sol} = X_{gluc}\rho_{gluc} + X_w\rho_w$$

$$\eta = 0.95e^{[glucose]} - 0.006$$

The density is in units of kg/m^3 and the viscosity is reported in units of Pa-s.

Algorithm Design

The algorithm is built on step-wise calculations which are repeated until the solution reaches the theoretical end of the membrane filter.

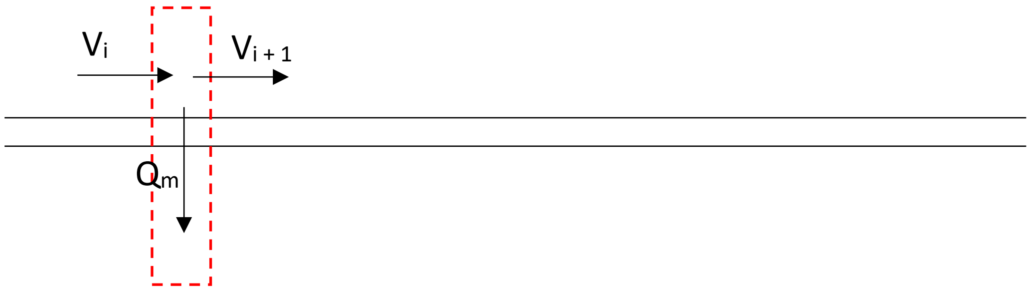


Figure 2. Numerical Integration across membrane

The overall flow of calculations can be described in the following flow chart (Fig. 3).

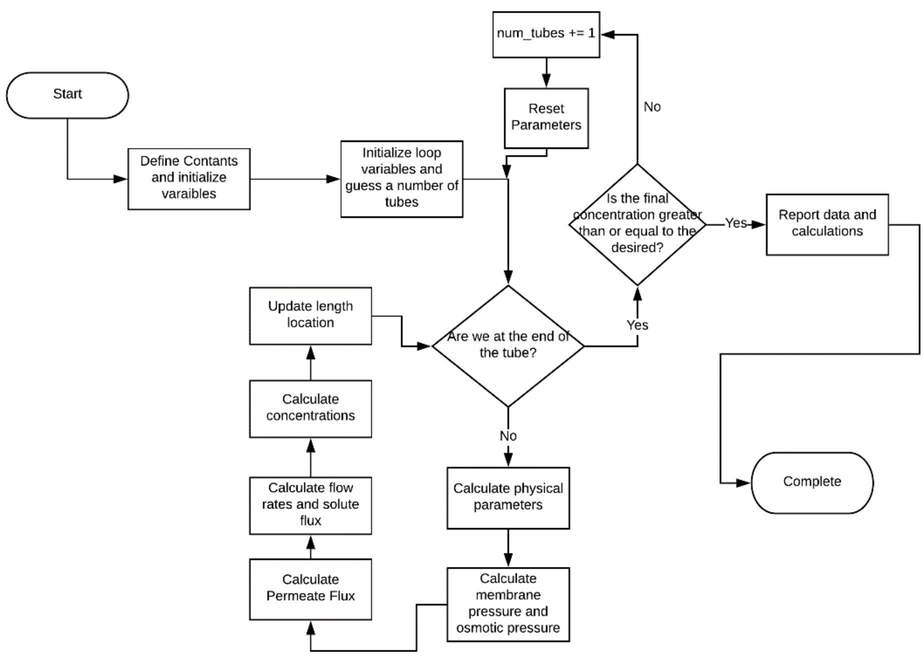


Figure 3. Flow chart for membrane design algorithm

This process was repeated for each tube grouping section. Which then gave us the final concentration of glucose in the solution.

Cost Analysis

It is no secret that the economics of engineering solutions plays a vital role in the decisions made. Part of this design was to investigate the financial investment required for all the technology being purchased and utilized. Two main areas of investment were investigated: Membrane Cost and Pump Cost/Power Requirements.

Membrane Cost – As we increase the number of required tubes and filter apparatus's, we increase the surface area of membranes required and thus, we increase the capital investment. An exponential cost correlation model was used to calculate the membrane costs:

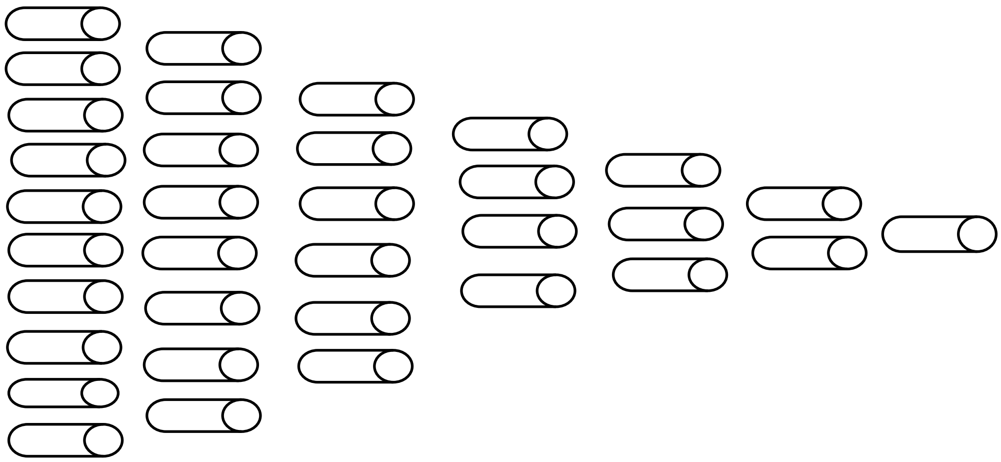
$$C = P\left(\frac{SM}{1000}\right)^{n}$$

Where C is the cost, P is the principle price, SM is the square meterage of the membranes, and n is a unitless coefficient. We were given the fact that 10 m^2 of membrane costs $400/$m^2$, thus we can calculate the principle price P to be 15,924.28. This relation can now be used to calculate the cost of any square meterage of membranes.

Pump Cost Requirements – The pumps cost capital as an initial investment, and they cost money to operate as well.

Results

The script showed that it was possible to conduct this separation process with 33 separate filter apparatuses assuming 100 tubes per apparatus. The arrangement can be done as such:



Glucose Concentration – The glucose concentration v completion percent was graphed in Fig. 4. We can see that the concentration of glucose gradually increases along the process. It appears almost linear, but is not completely linear and holds some degree of concavity to it. One interesting thing to note is that the process seems to speed up as we get closer to the end. One would expect that it slows down, especially coupled with the decreasing pressure along the membrane. This could be due to osmotic pressure differences.

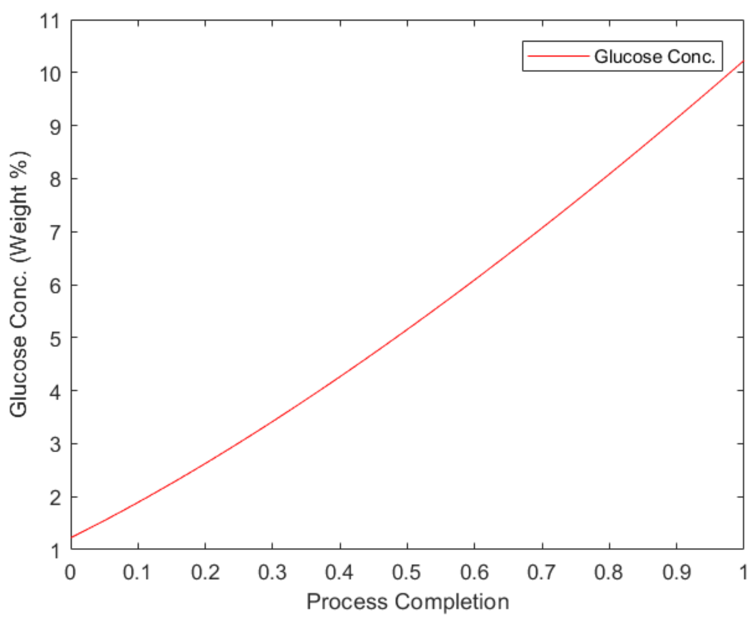


Figure 4. Glucose concentration v process time

Flow Rate – The flow rate across the process time was also graphed and analyzed. This data is shown in Fig. 5. We see that the flow rate of the retentate decreases over time. This is expected as water is passing through the membrane and being collected as permeate. This decreases the volume of the retentate and thus, decreases the overall flow rate.

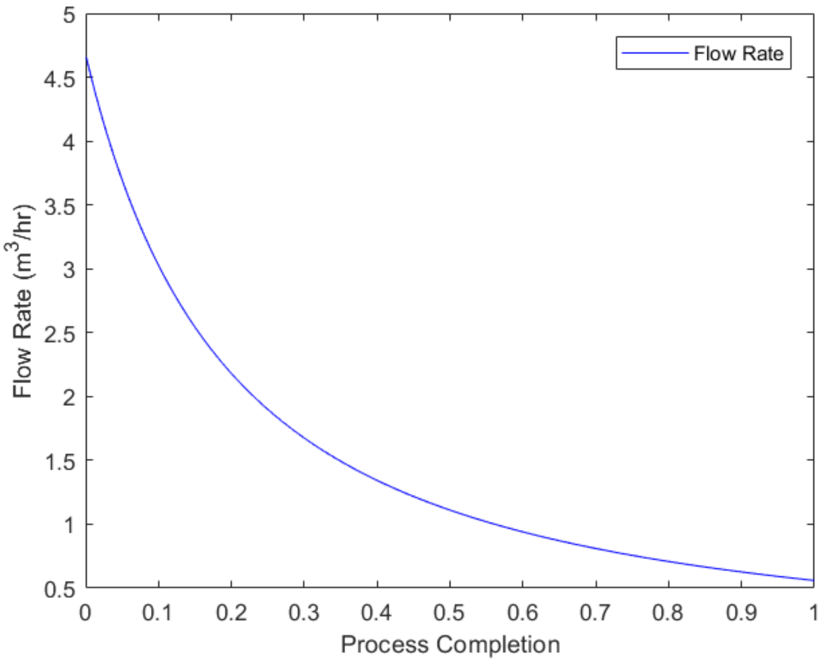


Figure 5. Flow rate changing over process completion ratio

Transmembrane Pressure – The transmembrane pressure was calculated for each section of the membrane as well. This data (Fig. 6) almost perfectly mirrors the flow rate data. This is expected as fluid velocity is the only changing parameter in the pressure differential equation, and the fluid velocity is calculated directly from the flow rate. For this exercise, the pressure remains unaltered for each section. Re-pressurizing each section of the process would most likely increase the efficiency of the process

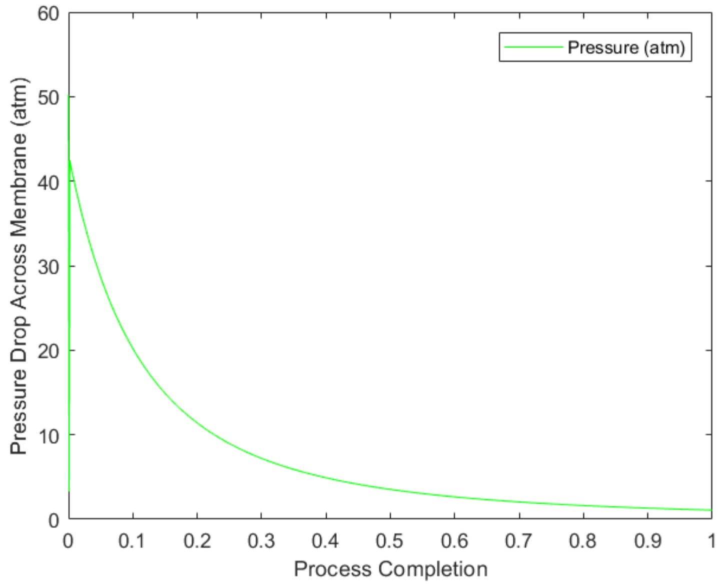


Figure 6. Transmembrane pressure along the separation process.

Cost Analysis – The cost of the membranes/filters is easiest to calculate. We can use the simple formula obtained in the cost analysis section of this report. It was found that the membranes will cost \$2.97 million. This is a very large capital investment, but it should be noted that this is a one-time cost and does not contribute to annual operating costs.

In addition to the membrane costs, we have the pump investment costs and the annual energy expenditure costs of running the pumps. The energy requirement can be calculated using the following equation:

$$Power = \frac{Q \Delta P \rho g}{\eta}$$

Assuming a pump efficiency of 55%, we can calculate the power to be:

117.52 kW

Energy costs are going to vary depending on location, but we can assume it to be \$0.12/kWhr. Assuming the pumps operate 12 hours per day, we can estimate the annual cost of a pump to be:

\$61,549

Code

```
clear all;
close all;
clc;


% Define Constants
flow_0 = 1230 * 1.0515e-6; % m^3/s
mass_frac = 0.0123;
mass_frac_out = 0.1023;
D = 15*1e-3; % diameter of tubes, meters
L = 2.46; % meter
delP_0 = 50.23; % atm
Aw = 4.01e-4; %kg H2O/s-m^2-atm
As = 1.23e-7; % m/s

% loop constants
num_tubes = 3000;
num_steps = 1000;
step_size = L/1000;



while mass_frac < mass_frac_out

    %define loops variables
    mass_frac = 0.0123;
    mem_loc = 0;
    v = 1.7; % m/s
    q_in = flow_0;
    flow_rate_store = [];
    flow_rate_store(1) = flow_0;
    mass_frac_store = [];
    mass_frac_store(1) = mass_frac;
    length_store = [];
    length_store(1) = mem_loc;
    delP_store = [];
    delP_store(1) = delP_0; %atm
    i = 2;
    area = 3.14*D*step_size*num_tubes; % m^2

    while mem_loc < L
        %calculate values
        rho = calc_density(mass_frac); % kg/m^3
        c_1 = mass_frac * rho; % kg glucose/ m^3 solution
        visc = calc_visc(mass_frac,rho); % Pa-s
        delP = calc_delP(visc,rho,step_size,v,D)/10; % atm
        delPi = calc_delPi(mass_frac,rho); % atm
        Nw = Aw*(delP - delPi); % kg H2O/m^2-s
        q_m = Nw * area * (1/1000); % m^3/s

        %calc final variables
        q_out = q_in - q_m; % m^3/s
        mass_frac_new = mass_frac*q_in/(mass_frac*q_in + (1-mass_frac)*q_in -
(Nw*area/1000)); % kg glucose/ kg solution
        mem_loc_new = mem_loc + step_size;
        v = q_out/(3.14159*(D/2)^2);
```

```matlab
            %store values
            flow_rate_store(i) = q_out;
            mass_frac_store(i) = mass_frac_new;
            length_store(i) = mem_loc_new;
            delP_store(i) = delP;

            % update parameters
            q_in = q_out; % update the q_in for the next loop
            mass_frac = mass_frac_new;
            mem_loc = mem_loc_new;
            i = i + 1;
        end

        num_tubes = num_tubes + 1;
 end

    tubes_per_filter = 100;
    num_filters = num_tubes/tubes_per_filter;

    area_tot = num_tubes*2.5*3.14159*D
    n = 0.8;
    P = 15924.28;
    cost_per = P*(area_tot/1000)^n;
    cost_tot = cost_per*area_tot

    fprintf('Final mass fraction: %0.2f %% w/w using %d tubes\n', mass_frac*100,
num_tubes);
    fprintf('Number of filter apparatas: %.0f \n', num_filters);

    figure(1)
    plot(length_store./(max(length_store)),mass_frac_store.*100, 'r-')
    xlabel('Process Completion');
    ylabel('Glucose Conc. (Weight %)');
    legend('Glucose Conc.');

    figure(2)
    plot(length_store./(max(length_store)),flow_rate_store.*3600,'b-')
    xlabel('Process Completion');
    ylabel('Flow Rate (m^3/hr)');
    legend('Flow Rate');

    figure(3)
    plot(length_store./(max(length_store)), delP_store,'g-');
    xlabel('Process Completion');
    ylabel('Pressure Drop Across Membrane (atm)');
    legend('Pressure (atm)');


function rho = calc_density(mass_frac)
    % Define constants
    temp = 25; % degrees C
    rho_w = 1000; % kg/m^3
    rho_g = 1586.2; % kg/m^3

    %calculate desnity
    rho = rho_g*mass_frac + rho_w*(1-mass_frac);

end
```

```
function visc = calc_visc(mass_frac,rho)
    %define constants
    MW_glucose = 0.180156; % kg glucose / mol

    %calculate molarity of solution
    molarity = mass_frac/rho/MW_glucose;

    %calculate viscosity of glucose solution
    visc = 0.95*exp(molarity)-0.006; % Pa-s

end

function delP = calc_delP(visc,rho,step_size,v,D);

    delP = 2*0.079*visc^0.25*rho^0.75*step_size*v^1.75*(1/D^1.25);

end

function delPi = calc_delPi(mass_frac,rho)
    % Define constants
    MW_glucose = 0.180156; % kg glucose/mol
    R = 8.205e-5; % m3-atm/mol-K
    temp = 25 + 273; % Kelvin

    %claculate the concentration of glucose
    conc_g = mass_frac/rho/MW_glucose; % mol/m^3

    %calculate the osmotic pressure of the solution
    delPi = (conc_g/MW_glucose)*R*temp; % atm

end
```