MECC System

Calculations

Ryan San Juan

The variables

|  |  |
| --- | --- |
|  | Buoy and piston 1 position |
|  | Buoy and piston 1 velocity |
|  | Buoy and piston 1 acceleration |
|  | Piston 2 position |
|  | Piston 2 velocity |
|  | Piston 2 acceleration |
|  | Lever arm angle |
|  | Lever arm angular velocity |
|  | Lever arm angular acceleration |
|  | Force between buoy and piston 1 |
|  | Force between piston 1 and lever |
|  | Force between piston 2 and lever |
|  | Pressure at piston 1 face |
|  | Pressure at piston 2 face |
|  | Pressure at membrane 1 inlet |
|  | Pressure at membrane 2 inlet |
|  | Water flux through membrane 1 |
|  | Water flux through membrane 2 |
|  | Salt flux through membrane 1 |
|  | Salt flux through membrane 1 |
|  | Feed concentration in membrane 1 |
|  | Feed concentration in membrane 2 |
|  | Permeate concentration in membrane 1 |
|  | Permeate concentration in membrane 2 |
|  | Concentration in reservoir |
|  | Height of reservoir |
|  | High flushing concentration in membrane 1 |
|  | High flushing concentration in membrane 2 |

The governing equations

1. Buoy force balance

2. Piston 1 force balance (piston 1 refers to the piston directly under the buoy)

3. Lever arm balance

4. Piston 2 force balance (piston 2 refers to the piston directly under the buoy)

5. Membrane 1 pressure drop (membrane 1 refers to the first pass membrane)

Upward buoy motion

Downward buoy motion J1=0 and Vm1 = 0

6. Membrane 2 pressure drop (membrane 2 refers to the second pass membrane)

Upward buoy motion J2=0 and Vm2 = 0

Downward buoy motion

7. Pressure at the face of piston 1

Upward buoy motion

Downward buoy motion

8. Pressure at the face of piston 2

Upward buoy motion

Downward buoy motion

9. Membrane 1 water flux

Upward buoy motion

Downward buoy motion

10. Membrane 2 water flux

Upward buoy motion

Downward buoy motion

11. Membrane 1 salt flux

Upward buoy motion

Downward buoy motion

(unsure about salt flux during inactivity)

12. Membrane 2 salt flux

Upward buoy motion

Downward buoy motion

13. Membrane 1 feed side concentration

Upward buoy motion

Downward buoy motion

14. Membrane 2 feed side concentration

Upward buoy motion

Downward buoy motion

15. Membrane 1 exit concentration

Upward buoy motion

Downward buoy motion

16. Membrane 2 exit concentration

Upward buoy motion

Downward buoy motion

17. Lever arm piston 1 side geometric

18. Lever arm piston 2 side geometric

19. Reservoir height equation

Upward buoy motion

Downward buoy motion

(adding some brine (Y% of brine) from pass 2 to maintain long term height)

20. Reservoir concentration equation

Upward buoy motion

Downward buoy motion

21. Membrane 1 high concentration flushing brine

Upward buoy motion

Downward buoy motion

22. Membrane 2 high concentration flushing brine

Upward buoy motion

Downward buoy motion

Differencing equations

Second order accuracy backward differencing formulas for positional variables y, yp2, and θ. (derived from taylor series expansion)

First order accuracy backward differencing formulas for membrane concentrations

First order accuracy backward differencing formulas for reservoir height and concentration

The algorithm

* At each time step, solve equations 1 – 20
  + During the beginning, evaluate y\_dot to see if the buoy is moving upward or downward
* Evaluate conditions and see if they have changed
* Repeat the previous two steps based on the conditions
* Move on to the next time step
* Use successive over/underrelaxation to help with stability

First solve for all the geometry because that’s where some of the constraints are and they determine the constraints on the rest of the system

Solve for y from buoy

Becomes

**STEP 1: make an initial y guess and ydot guess**

Solve the buoy equation using old information for y

Solve the ydot equation using y information

Bound ydot based on each state

**STEP 2: Solve for the rest of the geometry**

Ydot is known, now solve for everything else in terms of what is known

Solve for y from theta

Solve for y from y\_dot

Solve for y derivatives

Solve for theta using Newton-Raphson root finding method

From y

From yp2

Solve for derivatives of theta

Solve for φ1 and φ2

Solving for yp2

From theta

Solving for yp2 from yp2\_dot

Solving for yp2 derivatives

**STEP 3: Solve for forces**

After getting all the positional variables, solving for forces is next

Solve for F2 from piston 2 equation

Solve F1 from lever arm equation

**STEP 4: Fluxes**

Next, calculate fluxes based on velocity

If pp then

If flushing then

If pp then

If flushing then

If pp then

If flushing then

If pp then

If flushing then

**STEP 5: Pressures**

Next, solve for the pressures

Membrane 1 entrance pressure

Membrane 2 entrance pressure

Upward buoy motion J2=0 and Vm2 = 0

Downward buoy motion

Pressure at the face of piston 1

Upward buoy motion

Downward buoy motion

Pressure at the face of piston 2

Upward buoy motion

Downward buoy motion

**STEP 6: Concentrations**

Next, solve for all the concentrations

Membrane 1 exit concentration Ce1

If upward buoy and pp then

Otherwise

If downward buoy and pp then

Otherwise

Membrane 1 high flushing concentration

If upward buoy and flushing

Otherwise

Membrane 2 high flushing concentration

If downward buoy and flushing

Otherwise

Membrane 1 feed side concentration

If upward buoy and pp then

If upward buoy and flushing then

Otherwise downward buoy

Membrane 2 feed side concentration

If downward buoy and pp then

If downward buoy and flushing then

Otherwise upward buoy

**STEP 7: Reservoir terms**

Next calculate reservoir stuff

Reservoir height

Upward buoy motion

Downward buoy motion

(adding some brine (Y% of brine) from pass 2 to maintain long term height)

Reservoir concentration

Upward buoy motion

Downward buoy motion

Next calculate buoy force

Solving for Fb from piston 1 equation

**STEP 8: Check the logic**

Finally, check the logic

If unrestricted,

If buoy up and and -yp2dot is too high, switch to restricted

If buoy down and -ydot is too high, switch to restricted

If restricted

Recalculate y with buoy equation

Recalculate geometries for ydot and yp2dot

If ydot < 0, switch to buoy down

If ydot > 0, switch to buoy up

If buoy down and -ydot is not high enough, switch to unrestricted

If buoy up and -yp2dot is not high enough, switch to unrestricted

Thoughts: if it goes bananas during the loop and switches after, the old values will be bananas still

Things to do:

* Implement Ali’s method
* Restrict velocity to bounds

Archive this stuff is just ideas

IF RESTRICTED FLOW

Solve Fb from buoy equation

Bounding

Ali’s model

Inputs:

* Inlet concentration
* Flow rate in
* Feed concentration array Cfn-1

Outputs

* Concentration array at next time step
* Pressure array (first element is inlet pressure)
* Flow rate array
* Salt flux array
* Water flux array
* Salinity near active layer array
* Permeate salinity array
* Mass boundary layer array

Implementing Ali’s model:

Instead of the equations for membrane flux, membrane concentration, and membrane inlet pressure, calculate using the PFRO\_WEC function that Ali developed

Things to do

* Implement the model
* Calculate piston face pressures

Getting parameters from PFRO\_WEC

Pressures are a little different because they are measured at the wall, meaning there is one extra element

Modification of steps:

**STEP 4: Use PFRO\_WEC on each membrane**

Membrane 1:

Note to self: don’t use the new Cf1n array during the nth time step. Keep referring to the old concentration and return the newest once all iterations are done

Membrane 2:

**STEP 5: transform arrays into usable quantities**

**STEP 6: Calculate the pressures at the piston faces**

Pressure at the face of piston 1

Upward buoy motion

Downward buoy motion

Pressure at the face of piston 2

Upward buoy motion

Downward buoy motion

Continue from step 7

z

e

ffggh

sdsd

mine works ali’s does not

Permeate production works for me

Flushing does not

* Newton Raphson converges to the wrong root for theta calc
* The initial guess for y is too far off.
* Calculate yeq for each state

Experimental bounding procedure

1. Calculate y from buoy equation as an initial guess
2. Calculate ydot
3. Bound ydot based on membrane state
   1. The upper bounds for ydot are based on the yp2dot bounds for the other membrane
   2. Give an approximation

Calculating the equilibrium height

1. Set wave profile to zero
2. Calculate all values for first 5 time steps,
3. If y is greater than zero, add y to yeq
4. Repeat until yeq does not change

For the future (+ things I would do if I started over)

* To make it easier for the next time, linearize terms
* Try combining equations to a few ODEs
* Figure out why the solver in Ali’s code does not converge
* Fix the upward buoy motion for flushing
* Right now, the system is unstable without restrictions. In the future, possibly use a different approach to solve the system. Try euler implicit method for time stepping
* USE A CYLINDRICAL BUOY ITS SO MUCH EASIER
* Certain equations (like lever arm inertia and pressure drop across pipe segments) have a very small effect on the system, so they may be able to be neglected
* The pressure equations use the modified Bernoulli’s for pipe flow. There may be a better way to model this that’s a lot more accurate, since its unsteady.
* Put in the time to calculate a good equilibrium height for the buoy

Totals

Fresh water production

Brine production

Upward buoy

Downward buoy

Seawater used