Design Final Report

Serena Takada

7/25/14

Problem Definition

Introduction

This summer was my first time working with AguaClara. With that said, my goal this summer was to learn how to use the Automated Design Tools. After I felt more comfortable working with MathCad and AutoCad, I coded the recent updates from Honduras into the Sedimentation Tank and Stacked Rapid Sand Filter codes. I worked on updating the code for the Chemical Dose Controller used in India and applying the materials list to other hydraulic components but was interrupted by other challenges. In the future, I would like to work on updating the code for the Chemical Dose Controller and materials list.

Design Details

This summer was dedicated to updating the plant design based on the updates made in Honduras. First, I relocated the distribution weir in the inlet box of the Stacked Rapid Sand Filter that so the dump water from the inlet channel no longer overflowed and contaminated the clean water on its way to distribution. Second, I sloped the floor of the inlet channel in the Sedimentation Tank to establish a constant flow rate within the inlet channel. Afterward, I moved the Hopper Viewers in the outlet channel of the Sedimentation Tank into the inlet channel of the Sedimentation Tank in order to make it more accessible for the workers. Next, I moved the Floc Skimmers that were already in the same inlet channel further from the manifold pipes in order to move it out of the way of the Hopper Viewers. Finally, I placed extra hose clamps and feet in the Sedimentation Tank in order to support the horizontal inlet manifold pipes.

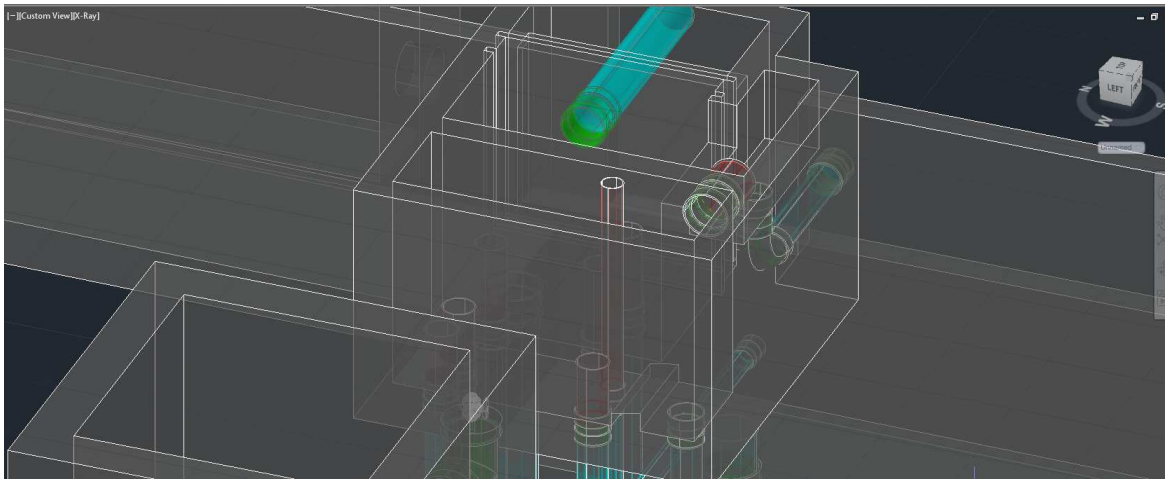


Figure 1: Distribution Barrier Drawing

Documented Progress

Distribution Barrier

During backwash of the Stacked Rapid Sand Filter, dump water from the Stacked Rapid Sand Filter was moved to the outlet box through filter outlet pipes. The dump water that came out of the outlet pipe stubbed around a large radius and spilled past a short weir that separated the outlet box from the distribution box next to it. The constraints in this problem were the number and size of outlets that needed to fit in the outlet and distribution box. In order to solve this issue, I moved the distribution barrier further from the outlet pipes. This distance was determined by the space fitting function. This distance would be the furthest possible distance the distribution barrier could be from the outlet pipes without obstructing the distribution pipe. In case this distance was not enough, I also angled the top of the distribution barrier that so the weir would be shorter on the side closer to the outlet pipes. Removing a wedge from the short weir would force the water that hits the weir to bounce off and return back to the outlet box where it belongs. Please see Figure 1 below. This process was difficult because it tested my ability to use the built in shape functions in MathCad. Finally, we determined the code that will mirror the distribution barrier every time the concrete in the Stacked Rapid Sand Filter was mirrored. Please see Figure 2 below.

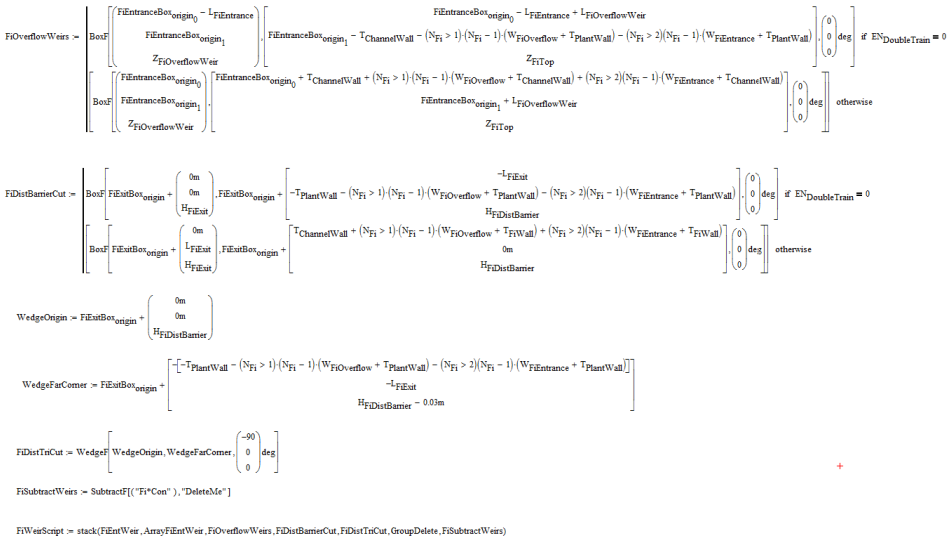


Figure 2: Distribution Barrier MathCad

Inlet Channel Floor

Water from the Flocculation Tank was not being distributed evenly amongst all of the Sedimentation Tanks. To solve this problem, I inclined the floor of the inlet channel that so the inlet channel of the Sedimentation Tank furthest form the Flocculation Tanks would be elevated the most. The constraint was that the concrete where the coupling of the Inlet Pipe will be needed to be flat. I made the highest part of the entire channel one free board height below the height of the Inlet Wall. I also elevated the coupling and lengthened the Vertical Inlet Manifold and cut down the Removable Inlet Pipe according to the change in height of the new inlet channel floor. The only remaining issue is that the width of the wedge of concrete that is removed from the floor of the Inlet Channel seems to have the wrong length. So, there is a really small slice of concrete that is not being removed. Please see Figures 3, 4, 5, 6, and 7below.

Hopper Viewer

The Hopper Viewers used to be in the outlet channel of the Sedimentation Tank. This position was inconvenient for the plant operator who had to look over the inlet channel to see into the Hopper Viewers located in the outlet channel. I moved the Hopper Viewers into the inlet channel that so the operator can have easy access to it. The constraints were that the pipes and couplings needed to be elevated depending on the height of the inlet channel floor at that point, the inlet channel wall will obstruct the Hopper Viewer closest to the Plant Wall,

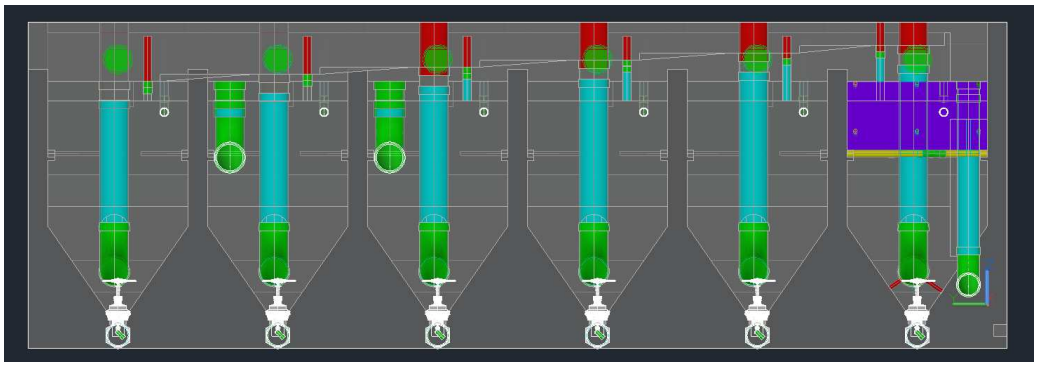


Figure 3: Inlet Channel Floor Drawing

Figure 4: Inlet Channel Floor MathCad 1

```
AngleInletFloor := | local ← SelectSedCon
                   | dZ ← InletFloorHeight
                   | HeightVector ← InletHeight
                   | BlockOrigin ← InletOrigin
                   | BlockFarCorner ← [ Plant_Origin_1 + (-W_SedDiffuserOutlet)/2 + outerradius(ND_SedJetReverser) + W_SedBay/2 - T_SedDiffuserWall - S_FromSedWall + ConRadius(ND_SedManifold) + T_SedWallDivider + SedManifoldSpace + SedHopperViewerSpace + 3S_Fitting + 2ConRadius(ND_Se
                   |                    BlockOrigin_0 - W_SedInletChannelPreWeir
                   |                    (BlockOrigin_2 + HeightVector_1) ]
                   | Block ← Box[ BlockOrigin, BlockFarCorner, (0 0 0) deg ]
                   | local ← stack(local, Block)
                   | NumTanks ← 2
                   | while NumTanks ≤ (N_SedTanks - 1)
                   |     BlockOrigin ← BlockFarCorner + [ W_SedInletChannelPreWeir ; 0m ; -(HeightVector)_(NumTanks-1) ]
                   |     BlockFarCorner ← BlockOrigin + ( 4S_Fitting + 2·ConRadius(ND_SedManifold) + T_SedWallDivider + SedManifoldSpace + SedHopperViewerSpace + 2ConRadius(ND_SedHopperViewer) ; -W_SedInletChannelPreWeir ; HeightVector_NumTanks )
                   |     Block ← Box[ BlockOrigin, BlockFarCorner, (0 0 0) deg ]
                   |     local ← stack(local, Block)
                   |     NumTanks ← NumTanks + 1
                   | local ← stack(local, FreezeSedTank)
                   | WedgeOrigin ← [ Plant_Origin_1 + (-W_SedDiffuserOutlet)/2 + outerradius(ND_SedJetReverser) + W_SedBay/2 - T_SedDiffuserWall - S_FromSedWall + ConRadius(ND_SedManifold) + T_SedWallDivider + SedManifoldSpace + SedHopperViewerSpace + 3S_Fitting + 2ConRadius(ND_Sed
                   |                  InletOrigin_0
                   |                  InletOrigin_2 + HeightVector_1 ]
                   | WedgeFarCorner ← WedgeOrigin + ( -T_SedWallDivider - SedManifoldSpace - SedHopperViewerSpace ; -W_SedInletChannelPreWeir ; dZ )
```

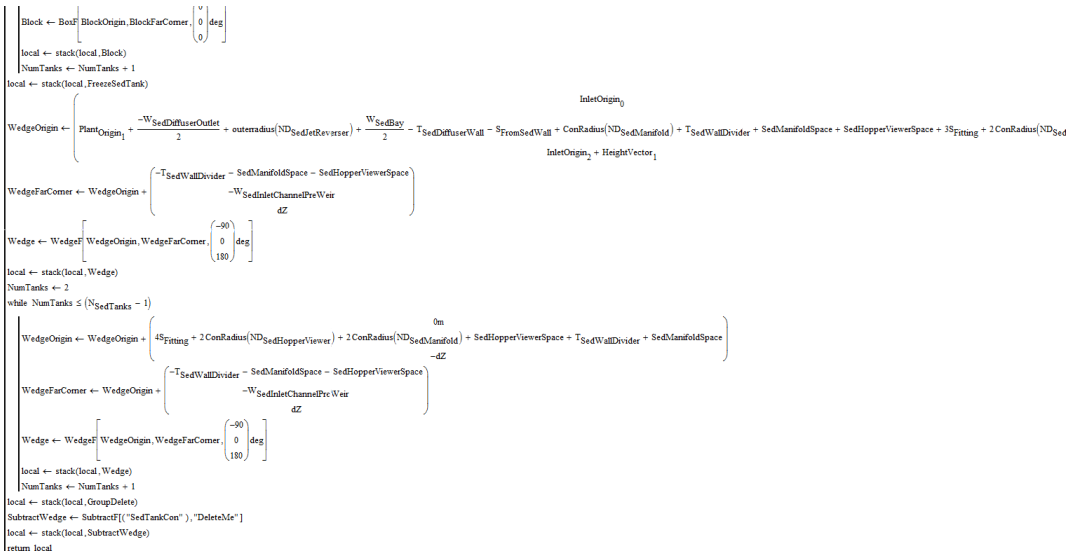


Figure 5: Inlet Channel Floor MathCad 2

Figure 6: Inlet Pipes MathCad

```
SedRemovableLayer := layer_new("SedRemovablePlu" ,red)
SelectSedRemovables := SelectLayer("SedRemovablePlu")
SedRemovableFrz := FreezeLayer("SedRemovablePlu")

ShorterRemovablePipes :=
    local ← SedRemovableLayer
    dZ ← InletFloorHeight
    HeightVector ← InletHeight
    PipeOrigin ← SedManVertCoupling_Origin + (0m, 0m, HeightVector_1)
    PipeLength ← H_SedInletChannel + SocketDepth(ND_SedManifold) − HeightVector_1
    Pipe1 ← PipeF[PipeOrigin, PipeLength, (0, 90) deg, ND_SedManifold, PS_Default]
    local ← stack(local, Pipe1)
    NumPipes ← 2
    while NumPipes ≤ N_SedTanks
        PipeOrigin ← PipeOrigin + (0m, 4·S_Fitting + 2·ConRadius(ND_SedHopperViewer) + 2·ConRadius(ND_SedManifold) + T_SedWallDivider + SedHopperViewerSpace + SedManifoldSpace, −dZ)
        PipeLength ← PipeLength + dZ
        Pipe ← PipeSUBF[PipeOrigin, PipeLength, (0, 90) deg, ND_SedManifold, PS_Default, ("SedTankCon")]
        local ← stack(local, Pipe)
        NumPipes ← NumPipes + 1
    local ← stack(local, SedRemovableFrz)
    return local
```

Figure 7: Inlet Removable Pipes

and the coupling of the Hopper Viewer needs to be on a flat surface. The Hopper Viewer in the Sedimentation Tank furthest from the Flocculation Tank was placed to the side of the Inlet Manifold Pipe that was on the opposite side of the Plant Wall. This will prevent the Inlet Wall from obstructing the Hopper Viewer. For the other Sedimentation Tanks the Hopper Viewers were placed on the side of the Inlet Manifold Pipe further from the Plant Wall. The flat area of the inlet channel also needed to be extended in order to have flat concrete to put the Hopper Viewer Coupling in. All of the Hopper Viewer Removable Pipes and Couplings were elevated depending on the height of the floor of the inlet channel and the length was also extended accordingly. Please see Figure 8 below. The code uses a while loop instead of an array, like the rest of the code, because the height of the inlet channel floor increases. Please see Figures 9, 10, and 11 below.

Floc Skimmer

After relocating the Hopper Viewers, the Floc Skimmers were actually obstructing the view through the Hopper Viewer. In order to avoid this, I moved the Floc Skimmers further form the Launder Pipes by 3 pipe fitting spaces. This made sure that the Floc Skimmers were no longer in the way of the Hopper Viewers. An exception was made for the Sedimentation Tank closest to the Plant Wall because the Hopper Viewer for this Sedimentation Tank is on the opposite side of the Inlet Manifold Pipe as compared to the other Sedimentation Tanks. So, the Floc Skimmer was not blocking the view from the Hopper Viewer to start with. Please see Figures 12, 13, 14, 15, 16, and 17 below.

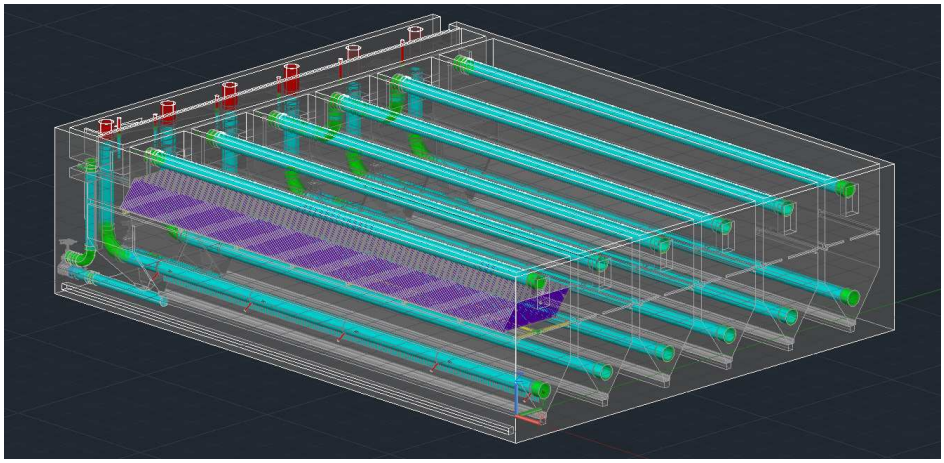


Figure 8: Hopper Viewer Drawing

Figure 9: Hopper Viewer Coupling MathCad

```
SedHopperViewerCoupling :=
    local ← SelectSedInletFittings
    dZ ← InletFloorHeight
    HeightVector ← InletHeight
    CouplingOrigin ← [ SedManVertCoupling_Origin_0 ; SedManVertCoupling_Origin_1 + ConRadius(ND_SedHopperViewer) + ConRadius(ND_SedManifold) + 2·S_Fitting ; Z_SedChannelBottom − SocketDepth(ND_SedHopperViewer) + HeightVector_1 ]
    Coupling1 ← CouplingSUBF[ CouplingOrigin, (0 ; 270) deg, ND_SedHopperViewer, PS_Default, ("SedTankCon") ]
    local ← stack(local, Coupling1)
    CouplingOrigin ← CouplingOrigin + [ 0m ; T_SedWallDivider + 2·SedHopperViewerSpace + 2·ConRadius(ND_SedHopperViewer) + 2·S_Fitting ; −dZ ]
    Coupling2 ← CouplingSUBF[ CouplingOrigin, (0 ; 270) deg, ND_SedHopperViewer, PS_Default, ("SedTankCon") ]
    local ← stack(local, Coupling2)
    NumPipes ← 3
    while NumPipes ≤ N_SedTanks
        CouplingOrigin ← CouplingOrigin + [ 0m ; W_Sed + T_SedWallDivider ; −dZ ]
        Coupling ← CouplingSUBF[ CouplingOrigin, (0 ; 270) deg, ND_SedHopperViewer, PS_Default, ("SedTankCon") ]
        local ← stack(local, Coupling)
        NumPipes ← NumPipes + 1
    local ← stack(local, SedInletFittingFrz)
    return local
```

```
HopperViewerShorterPipes :=
local ← SelectSedRemovables
dZ ← InletFloorHeight
HeightVector ← InletHeight
PipeOrigin ← ( SedManVertCoupling_Origin_0 ; SedManVertCoupling_Origin_1 + ConRadius(ND_SedHopperViewer) + ConRadius(ND_SedManifold) + 2·S_Fitting ; Z_SedChannelBottom − SocketDepth(ND_SedHopperViewer) + HeightVector_1 )
PipeLength ← H_SedInletChannel + SocketDepth(ND_SedHopperViewer) − HeightVector_1
Pipe1 ← PipeF[PipeOrigin, PipeLength, (0 ; 90) deg, ND_SedHopperViewer, PS_Default]
local ← stack(local, Pipe1)
PipeOrigin ← PipeOrigin + ( 0m ; T_SedWallDivider + 2·SedHopperViewerSpace + 2·ConRadius(ND_SedHopperViewer) + 2·S_Fitting ; −dZ )
PipeLength ← PipeLength − dZ
Pipe2 ← PipeF[PipeOrigin, PipeLength, (0 ; 90) deg, ND_SedHopperViewer, PS_Default]
local ← stack(local, Pipe2)
NumPipes ← 3
while NumPipes ≤ N_SedTanks
    PipeOrigin ← PipeOrigin + ( 0m ; W_Sed + T_SedWallDivider ; −dZ )
    PipeLength ← PipeLength + dZ
    Pipe ← PipeSUBF[PipeOrigin, PipeLength, (0 ; 90) deg, ND_SedHopperViewer, PS_Default, ("SedTankCon")]
    local ← stack(local, Pipe)
    NumPipes ← NumPipes + 1
local ← stack(local, SedRemovableFrz)
return local
```

Figure 10: Hopper Viewer Removable Pipes

Figure 11: Hopper Viewer Pipes

```
HopperViewerLowerPipes :=
local ← SelectSedInletPipes
dZ ← InletFloorHeight
HeightVector ← InletHeight
PipeOrigin ← ( SedManVertCoupling_Origin_0 ; SedManVertCoupling_Origin_1 + ConRadius(ND_SedHopperViewer) + ConRadius(ND_SedManifold) + 2·S_Fitting ; Z_SedChannelBottom − SocketDepth(ND_SedHopperViewer) + HeightVector_1 − CouplingLength(ND_SedHopperViewer) + 2·SocketDepth(ND_SedHopperViewer) )
PipeLength ← T_ChannelWall − CouplingLength(ND_SedHopperViewer) + SocketDepth(ND_SedHopperViewer) + HeightVector_1
Pipe1 ← PipeF[PipeOrigin, PipeLength, (0 ; 270) deg, ND_SedHopperViewer, PS_Default]
local ← stack(local, Pipe1)
PipeOrigin ← PipeOrigin + ( 0m ; T_SedWallDivider + 2·SedHopperViewerSpace + 2·ConRadius(ND_SedHopperViewer) + 2·S_Fitting ; −dZ )
PipeLength ← PipeLength − dZ
Pipe2 ← PipeF[PipeOrigin, PipeLength, (0 ; 270) deg, ND_SedHopperViewer, PS_Default]
local ← stack(local, Pipe2)
NumPipes ← 3
while NumPipes ≤ N_SedTanks
    PipeOrigin ← PipeOrigin + ( 0m ; W_Sed + T_SedWallDivider ; −dZ )
    PipeLength ← PipeLength − dZ
    Pipe ← PipeSUBF[PipeOrigin, PipeLength, (0 ; 270) deg, ND_SedHopperViewer, PS_Default, ("SedTankCon")]
    local ← stack(local, Pipe)
    NumPipes ← NumPipes + 1
local ← stack(local, SedInletPipeFrz)
return local

SedInletChannelScript := stack(InletCalculations, SedTankLayer, AngleInletFloor, ElevateCoupling, LongerPipes, ShorterRemovablePipes, SedHopperViewerCoupling, HopperViewerShorterPipes, HopperViewerLowerPipes)
```

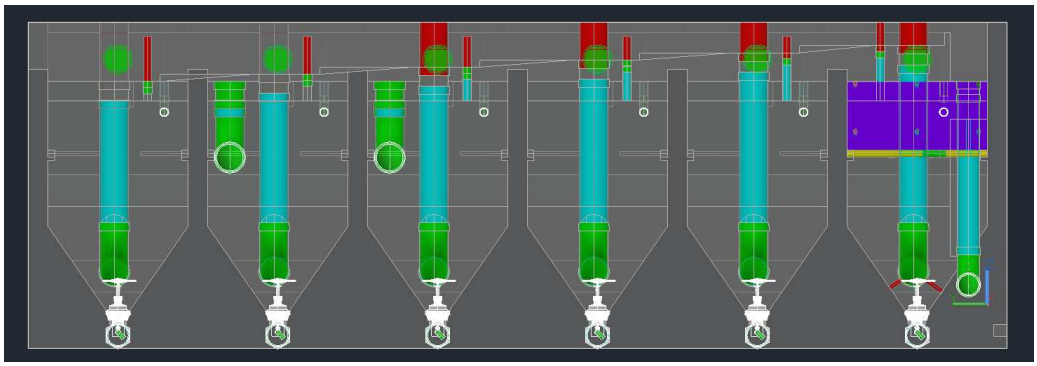


Figure 12: Floc Skimmer Drawing

Floc Skimmer Origins and Dimentions

$$SedSkimmerCoupling_{Origin} := \begin{bmatrix} Plant_{Origin_0} - L_{Sed} - T_{PlantWall} + SocketDepth(ND_{SedSkimmer}) \\ Plant_{Origin_1} + \frac{W_{SedBay}}{2} - ConRadius(ND_{SedLaunder}) - 4S_{Fitting} - ConRadius(ND_{SedSkimmer}) + (W_{Sed} + T_{SedWallDivider}) \\ Z_{SedChannelBottom} - T_{ChannelWall} - S_{Fitting} - ConRadius(ND_{SedSkimmer}) \end{bmatrix}$$

$$SedSkimmerCap_{Origin} := \begin{pmatrix} Plant_{Origin_0} - L_{Sed} + W_{SedExitChannel} + W_{SedInletChannel} + T_{SedWeir} + T_{ChannelWall} + \frac{CapDiameter(ND_{SedSkimmer})}{2} \\ SedSkimmerCoupling_{Origin_1} \\ Plant_{Origin_2} + HW_{Sed} - H_{SkimmerToSurface} \end{pmatrix}$$

$$SedSkimmerHorizPipe_{Origin} := \begin{pmatrix} SedSkimmerCap_{Origin_0} - ElbowRadius(ND_{SedSkimmer}) \\ SedSkimmerCoupling_{Origin_1} \\ SedSkimmerCoupling_{Origin_2} \end{pmatrix} - \blacksquare$$

$$SedSkimmerElbow_{Origin} := SedSkimmerHorizPipe_{Origin} + \begin{pmatrix} ElbowRadius(ND_{SedSkimmer}) \\ 0 \\ ElbowRadius(ND_{SedSkimmer}) \end{pmatrix}$$

$$SedSkimmerVertPipe_{Origin} := SedSkimmerCap_{Origin}$$

Floc Skimmer Fittings

$SedSkimmerFittingScript :=$

local ← SelectSedRemovables
NumTanks ← 3
Origin ← $SedSkimmerCap_{Origin}$
$SedSkimmerCap \leftarrow CapF\left[Origin, \begin{pmatrix} 0 \\ -90 \end{pmatrix} deg, ND_{SedSkimmer}\right]$
while NumTanks ≤ $N_{SedTanks}$
  $Origin \leftarrow Origin + \begin{pmatrix} 0m \\ W_{Sed} + T_{SedWallDivider} \\ 0m \end{pmatrix}$

Figure 13: Floc Skimmer MathCad 1

Figure 14: Floc Skimmer MathCad 2

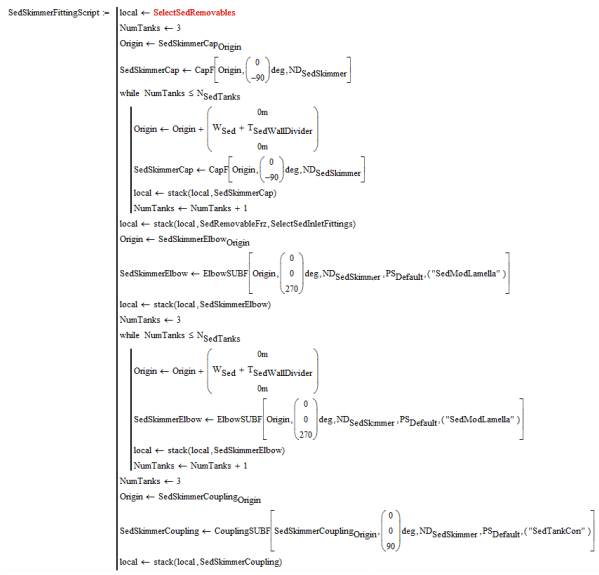


Figure 15: Floc Skimmer MathCad 3

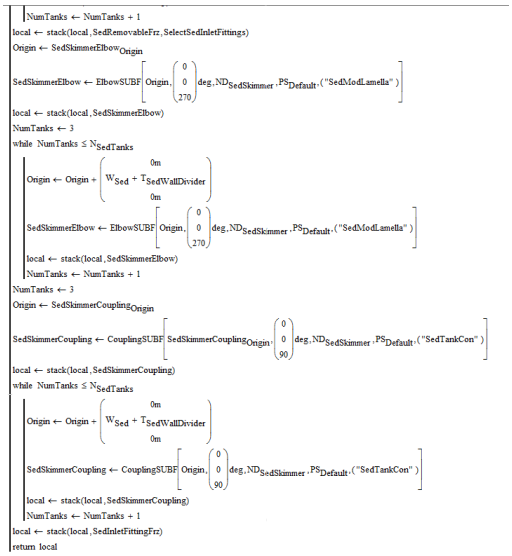


Figure 16: Floc Skimmer MathCad 4

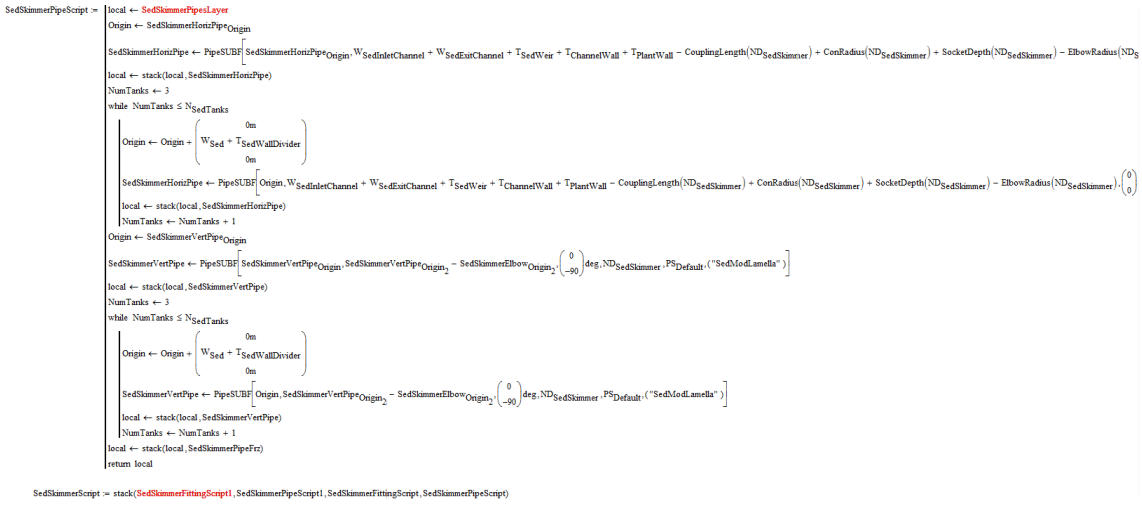
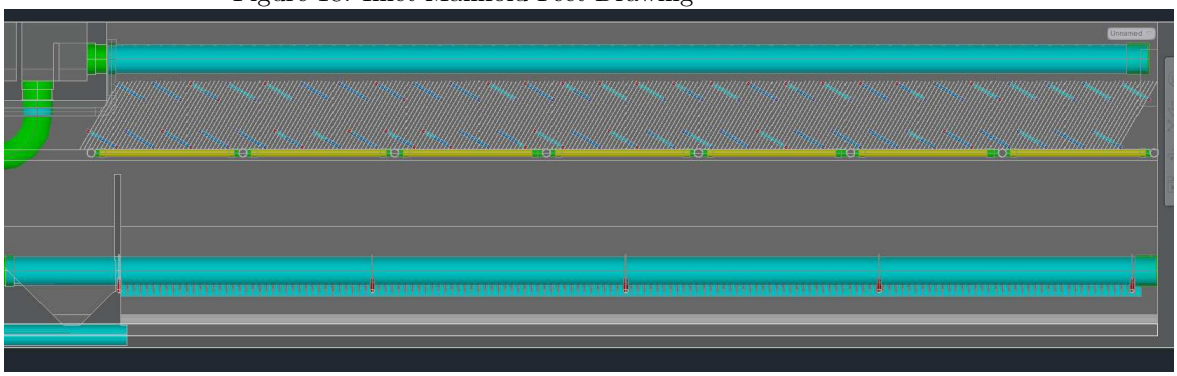


Figure 17: Floc Skimmer Pipe MathCad

Figure 18: Inlet Manifold Feet Drawing



Inlet Manifold Feet

Depending on the flow rate of the plant, the length of the Sedimentation Tank and Horizontal Inlet Manifold Pipe will change. In order to support the Horizontal Inlet Manifold Pipe when they are longer, I added horse clamps and feet o the Horizontal Inlet Manifold Pipe. I created a while loop that will distribute an even number of horse clamps and feet o support the Horizontal Inlet Manifold Pipe. The longer the pipe, the more support it would receive.

Future Work

The design of the plant is constantly being updated. It is important for the Design Team to keep up with these improvements. In the future, we should work on cleaning up the concrete arrays in the Sedimentation Tanks, and updating the code for the Chemical Dose Controller and materials list. We would also like to reorganize the code for the Sedimentation Tank in order to make it easier to understand.

Figure 19: Inlet Manifold Feet MathCad 1

```
SedFeetScript := | NumFeet ← N_ManifoldFeetSets
                 | local ← ManFeetLayer
                 | dX ← |Loc_Feet| / (NumFeet − 1)
                 | OriginLong ← ManFeetLong_OriginPoint
                 | OriginShort ← ManFeetShort_OriginPoint
                 | OriginLong ← OriginLong + (dX, 0m, 0m)
                 | OriginShort ← OriginShort + (dX, 0m, 0m)
                 | ManFeetLong ← PipeF[OriginLong, L_LongFoot·(−AN_actual, 90deg), ND_ManFeet, PS_Default]
                 | ManFeetShort ← PipeF[OriginShort, L_ShortFoot·(180deg − AN_actual, −90deg), ND_ManFeet, PS_Default]
                 | local ← stack(local, ManFeetLong, ManFeetShort)
                 | while NumFeet > 3
                 |   | OriginLong ← OriginLong + (dX, 0m, 0m)
                 |   | OriginShort ← OriginShort + (dX, 0m, 0m)
                 |   | ManFeetLong ← PipeF[OriginLong, L_LongFoot·(−AN_actual, 90deg), ND_ManFeet, PS_Default]
                 |   | ManFeetShort ← PipeF[OriginShort, L_ShortFoot·(180deg − AN_actual, −90deg), ND_ManFeet, PS_Default]
                 |   | local ← stack(local, ManFeetLong, ManFeetShort)
                 |   | ManFeetLong ← PipeF[OriginLong, L_LongFoot·(−AN_actual, 90deg), ND_ManFeet, PS_Default]
                 |   | ManFeetShort ← PipeF[OriginShort, L_ShortFoot·(180deg − AN_actual, −90deg), ND_ManFeet, PS_Default]
                 |   | local ← stack(local, ManFeetLong, ManFeetShort)
                 |   | NumFeet ← NumFeet − 1
                 | HoseOrigin ← HoseClamp_Origin + (dX, 0m, 0m)
                 | HoseClampOuter ← CylinderF[HoseOrigin, R_Clamp + T_Clamp, L_Clamp·(0, 0)deg]
                 | HoseClampInner ← CylinderF[HoseOrigin, R_Clamp, L_Clamp·(0, 0)deg]
                 | DrawHoseClamps ← stack(SelectHoseClamps, HoseClampOuter, FrzLayerHoseClamp, HoseClampInner, GroupDelete, SubtractInner, DrawHole1, DrawHole2, GroupDelete, SubtractHole, FrzLayerHoseClamp)
                 | local ← stack(local, ManFeetLayerFrz, DrawHoseClamps)
                 | NumFeet ← N_ManifoldFeetSets
                 | while NumFeet > 3
                 |   | HoseOrigin ← HoseOrigin + (dX, 0m, 0m)
                 |   | HoseClampOuter ← CylinderF[HoseOrigin, R_Clamp + T_Clamp, L_Clamp·(0, 0)deg]
                 |   | HoseClampInner ← CylinderF[HoseOrigin, R_Clamp, L_Clamp·(0, 0)deg]
                 |   | DrawHoseClamps ← stack(SelectHoseClamps, HoseClampOuter, FrzLayerHoseClamp, HoseClampInner, GroupDelete, SubtractInner, DrawHole1, DrawHole2, GroupDelete, SubtractHole, FrzLayerHoseClamp)
                 |   | local ← stack(local, DrawHoseClamps)
                 |   | NumFeet ← NumFeet − 1
                 | return local

ManFeetScript := stack(ManFeetLayer, ManFeetLong, ManFeetShort, ManFeetLayerFrz, DrawHoseClamp1, SelectHoseClamps, SelectFeet, ArrayFeet, FrzLayerHoseClamp, SedFeetScript)
```

Figure 20: Inlet Manifold Feet MathCad 2

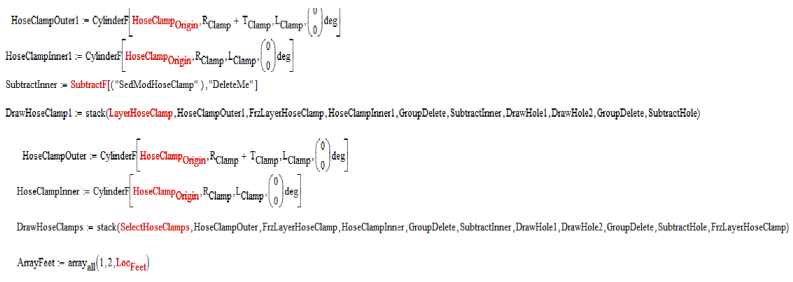


Figure 21: Inlet Manifold Hose Clamps MathCad