4CBLA30 Energy Storage and Transport

Self-Study Assignment

Group 46

SSA No.	Description
3	Poster and Mathematical Model
SSA Owner	
Pranav Joshi	

Introduction

From the previous meeting, starting on the mathematical model and the basic structure of the poster needs to be created.

Goal

- To help create a basic structure for the poster
- To start the modeling process according to the mind-map on the est website

Problems

- Understanding the Simulink model was not easy and quite time-consuming

Conclusion

- The system parameters have been set
- Simulink has been initialized to model some basics of the PHS system, however it may not be completely accurates

Recommendations

-

1 Elaboration

1.1 Creating a Basic Structure for The Poster

Looking at the winning poster and runners-ups of last year (as shown in fig. 1, fig. 2 and fig. 3). Along with this, the description of the poster on the EST website [1] was also read. The following is the common theme between them;

- Description of the Energy Supply (Solar Panels/Wind Energy along with surface area/number of wind mills)
- Concise but clear description of each sub-system and their respective function/s

- Numbers are used where neccesary (Depths/Energy/Power/Volume/Surface Area)
- The parameters that need to be tested are identified.

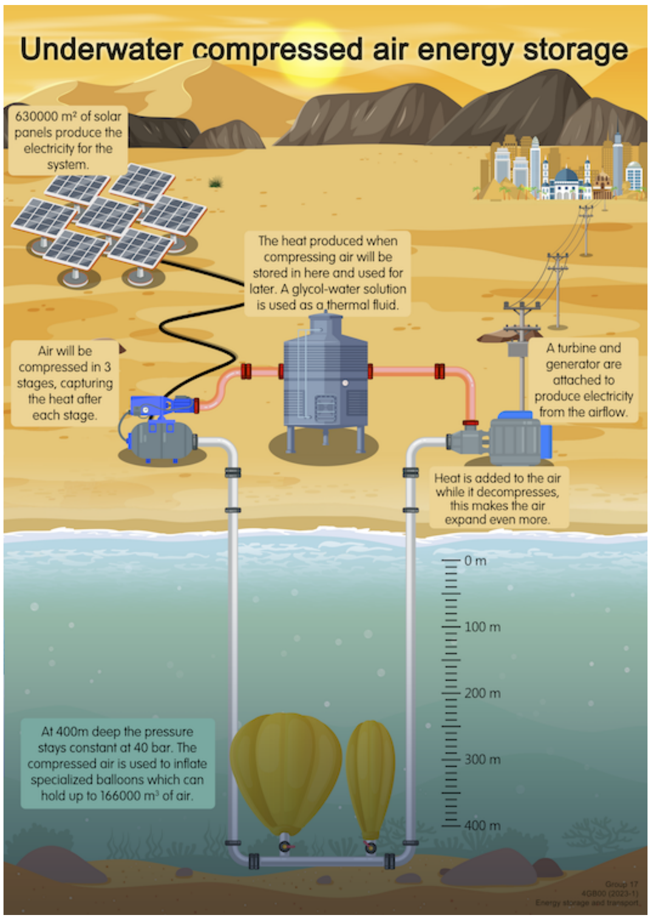

Underwater compressed air energy storage
630000 m² of solar panels produce the electricity for the system.
The heat produced when compressing air will be stored in here and used for later. A glycol-water solution is used as a thermal fluid.
Air will be compressed in 3 stages, capturing the heat after each stage.
A turbine and generator are attached to produce electricity from the airflow.
Heat is added to the air while it decompresses, this makes the air expand even more.
0 m
100 m
200 m
300 m
400 m
At 400m deep the pressure stays constant at 40 bar. The compressed air is used to inflate specialized balloons which can hold up to 166000 m³ of air.
Group 17
4GB00 (2023-1)
Energy storage and transport

Figure 1: Poster Competition WInner

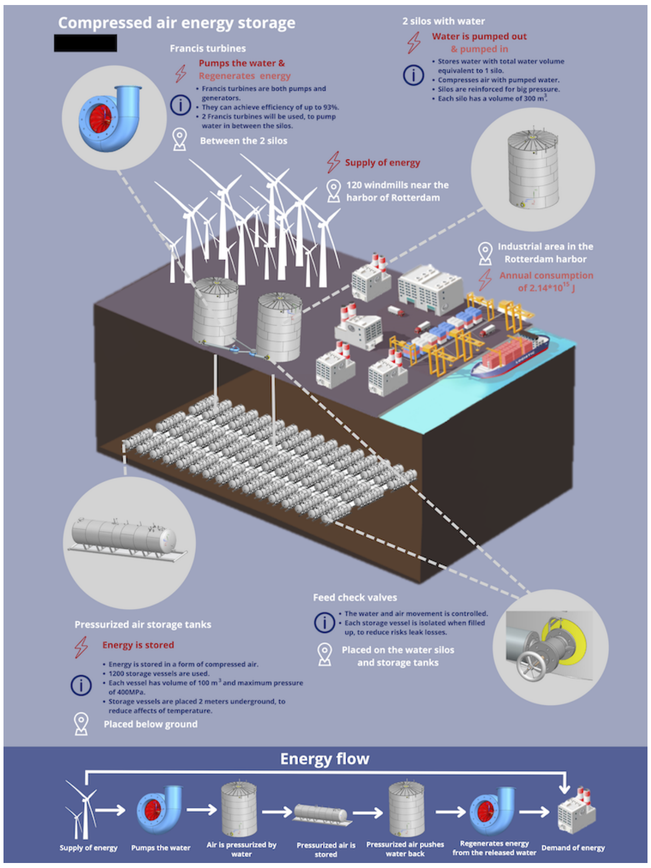

Compressed air energy storage
[redacted]
Francis turbines
Pumps the water &
Regenerates energy
• Francis turbines are both pumps and generators.
• They can achieve efficiency of up to 93%.
• 2 Francis turbines will be used, to pump water in between the silos.
Between the 2 silos
2 silos with water
Water is pumped out
& pumped in
• Stores water with total water volume equivalent to 1 silo.
• Compresses air with pumped water.
• Silos are reinforced for big pressure.
• Each silo has a volume of 300 m³.
Supply of energy
120 windmills near the harbor of Rotterdam
Industrial area in the Rotterdam harbor
Annual consumption of 2.14*10¹⁵ J
Pressurized air storage tanks
Energy is stored
• Energy is stored in a form of compressed air.
• 1200 storage vessels are used.
• Each vessel has volume of 100 m³ and maximum pressure of 400MPa.
• Storage vessels are placed 2 meters underground, to reduce affects of temperature.
Placed below ground
Feed check valves
• The water and air movement is controlled.
• Each storage vessel is isolated when filled up, to reduce risks leak losses.
Placed on the water silos and storage tanks
Energy flow
Supply of energy
Pumps the water
Air is pressurized by water
Pressurized air is stored
Pressurized air pushes water back
Regenerates energy from the released water
Demand of energy

Figure 2: Poster Competition Runner Up

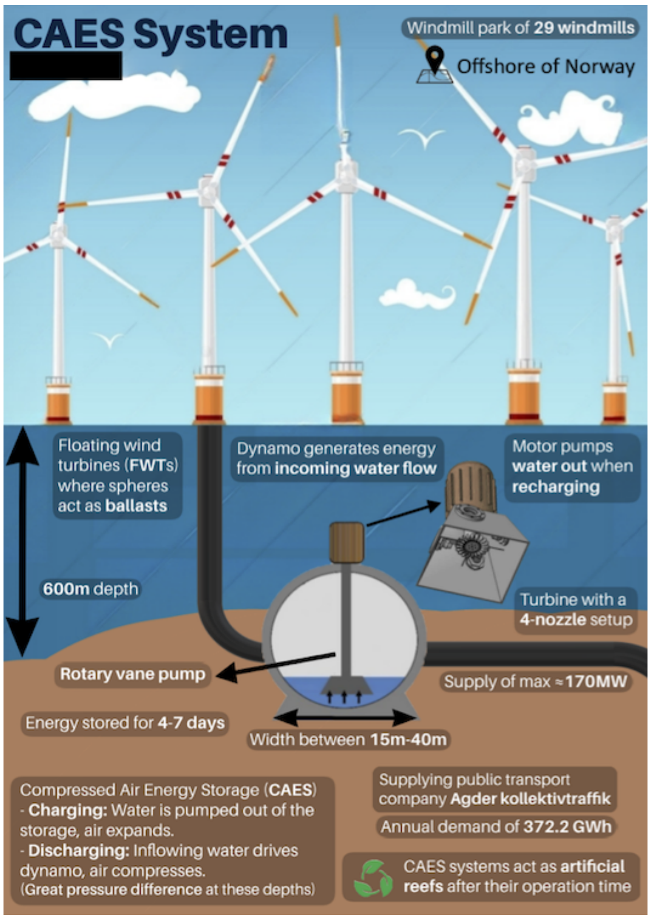
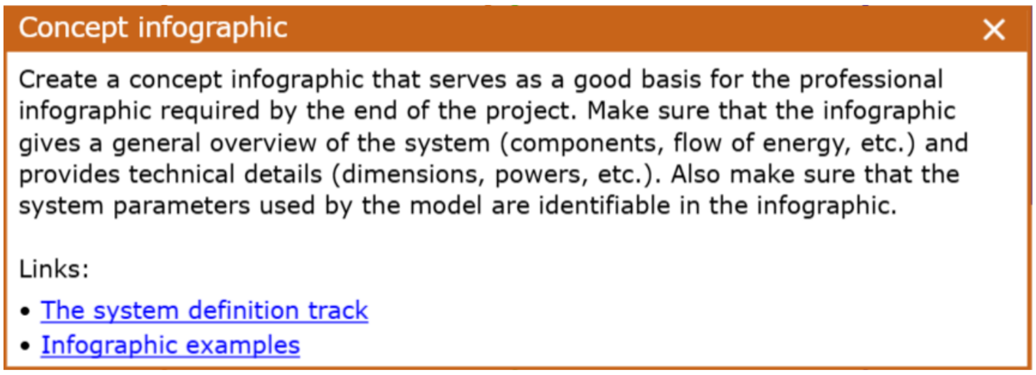

CAES System
[redacted]
Windmill park of 29 windmills
Offshore of Norway
Floating wind turbines (FWTs) where spheres act as ballasts
Dynamo generates energy from incoming water flow
Motor pumps water out when recharging
600m depth
Turbine with a 4-nozzle setup
Rotary vane pump
Supply of max ≈170MW
Energy stored for 4-7 days
Width between 15m-40m
Compressed Air Energy Storage (CAES)
- Charging: Water is pumped out of the storage, air expands.
- Discharging: Inflowing water drives dynamo, air compresses.
(Great pressure difference at these depths)
Supplying public transport company Agder kollektivtraffik
Annual demand of 372.2 GWh
CAES systems act as artificial reefs after their operation time

Figure 3: Poster Competition 2nd Runner Up


Concept infographic ×

Create a concept infographic that serves as a good basis for the professional infographic required by the end of the project. Make sure that the infographic gives a general overview of the system (components, flow of energy, etc.) and provides technical details (dimensions, powers, etc.). Also make sure that the system parameters used by the model are identifiable in the infographic.

Links:

- The system definition track
- Infographic examples

Figure 4: EST-website description

For the selected Pumped Hydro System the following needs to be highlighted/described:

- The area of solar panels that are providing power (880 m2) and their placement on top of the building
- Energy Consumption of the building in Berlin
- Depth at which the pump turbine is kept
- Volume and placement of reservoir/s
- Describing the Charging and Discharging process

Until the model is further developed and information is made more concrete, the poster cannot be progressed much further. Along with that due to lack of graphic design experience and time, currently the mathematical model work will be given higher priority.

1.2 Mathematical Model

According to the EST website mind-map (as shown in fig. 5), the components of the system have been defined, the physical phenomena have been identified, physical laws have been explored.

The mathematical elaboration needs to be done in Simulink. In parallel, the system parameters need to be determined. To start off, the system parameters are defined in a table below;

Sub-System	**Parameters**
Top Reservoir	Volume,Shape, Dimensions, Base Area, Relative Altitude (to pump turbine), Material
Pump Turbine	Efficiency
Solar Panels	Energy Output, Surface Area

Table 1: System Parameters

After further discussions and brainstorming, the parameters are defined as following:

Sub-System	**Parameter**	**Value**
Top Reservoir	Volume	28,880 m^3
	Shape	Cuboid
	Dimensions	38*38*20 m
	Base Area	1,444 m^2
	Relative Altitude	1,000 m
	Material	Reinforced Concrete [3]
Pump Turbine	Effeciency	90 %
Solar Panels	Energy Output	$9.3 * 10^4$ kWh
	Surface Area	880 m^2

Table 2: Assignment Types and Related Learning Goals

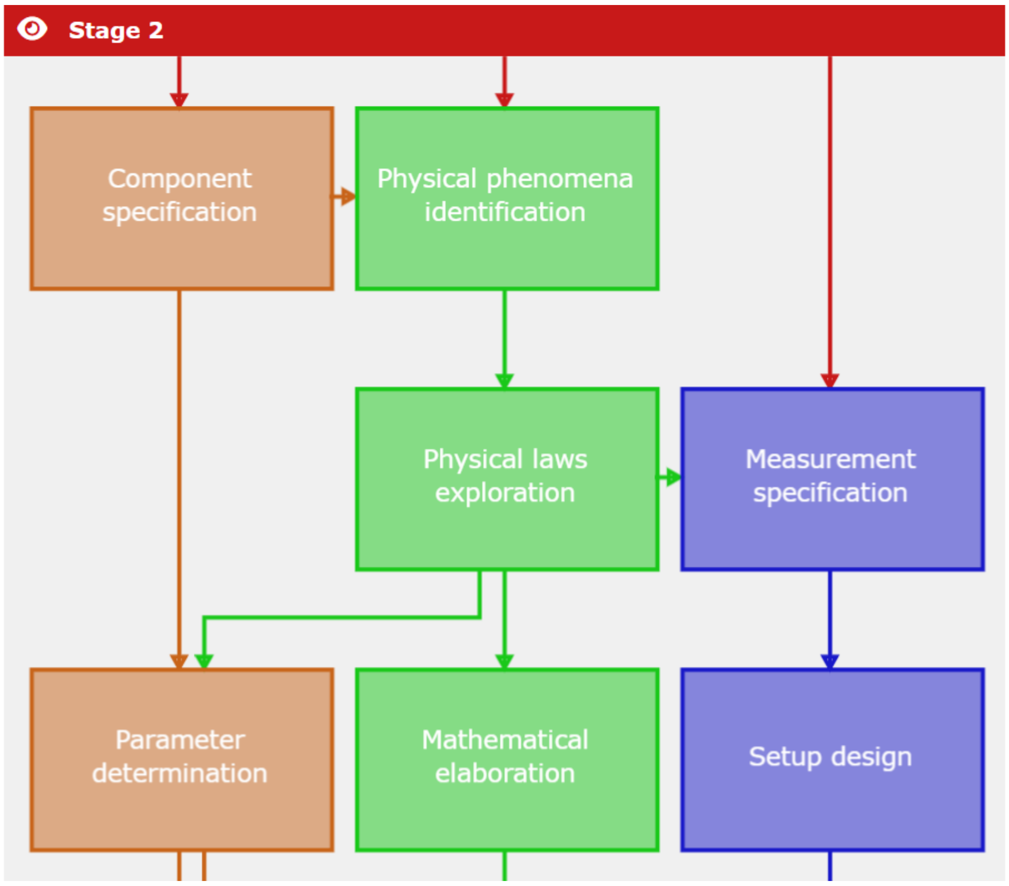


Figure 5: Mind Map [1]

1.3 Simulink Implementation

To implement parts of the system in the simulink model, the resources provided on the github page [2] were used. The simulink model was downloaded and the example given in the README file was understood.

To start off, the Dissipation coeffecient was determined for the entire PHS system. Since the system has an effeciency of 90 percent, the dissipation coeffecient (α) is given by:

$$\begin{aligned} \alpha &= 1 - \eta \\ &= 10\% \ (\eta = 90\%) \end{aligned} \tag{1}$$

(The power going into the storage system(P_{in}) is 25 kW on average, as derived in SSA 1)

After putting in the supply and demand data given by the EST website, the following graphs were obtained.

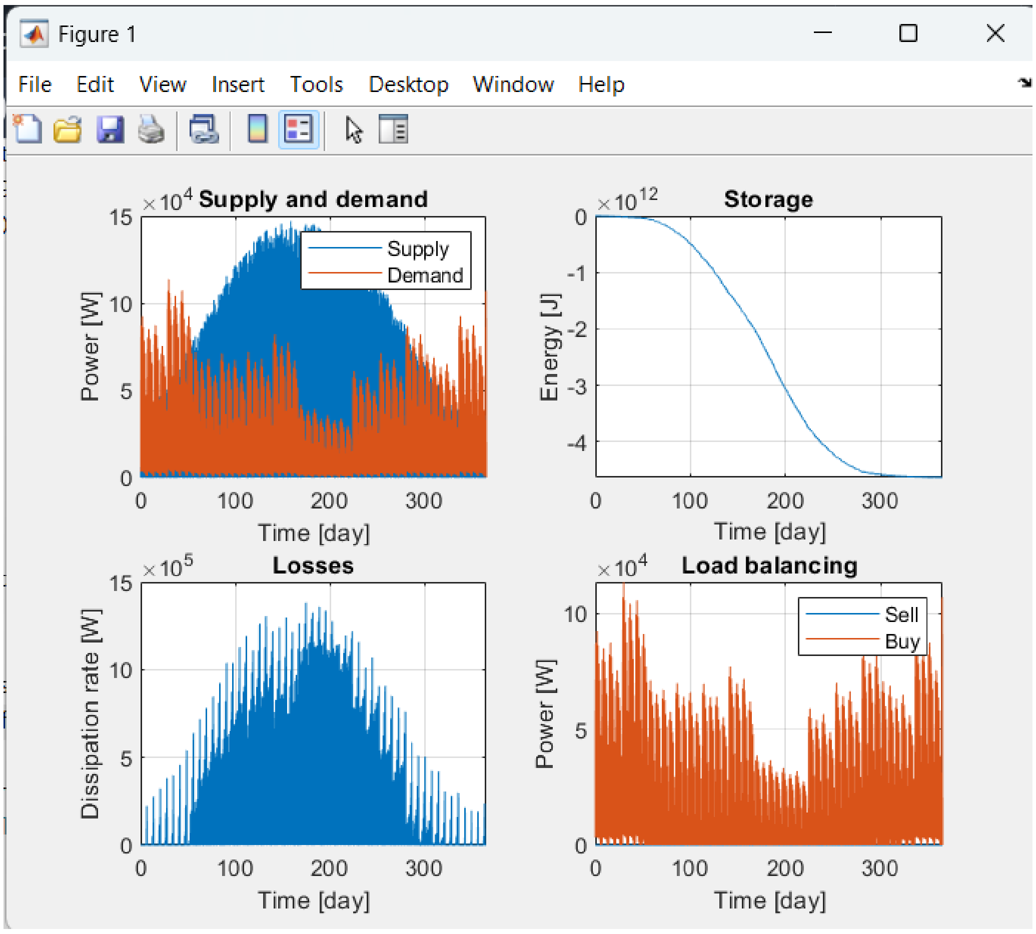


Figure 6: Graphs

However, except for the first graph, the rest of the graphs do not seem to be appropriate. The storage is negative, which is clearly inaccurate and physically wrong.

To solve this, the individual blocks of the model were investigated. Perhaps the logic was inconsistent with the required output in a certain block.

Looking at the controller block (as shown in fig. 7), the energy storage function was looked into.

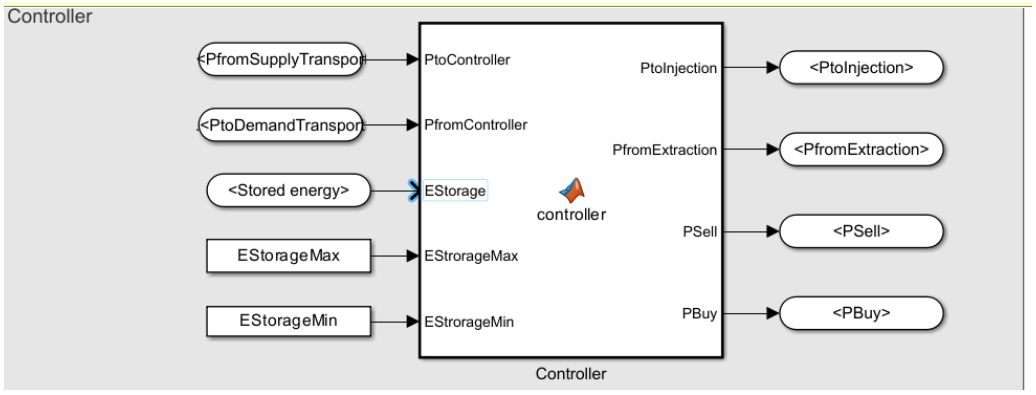


Figure 7: Controller

The given model also includes a selling and buying logic. This is inapplicable for the model that needs to be created for the PHS system. Hence, this function block needs to be changed to meet the needs of the given use-case scenario.

```
    % Storage system is full
    else
        PtoInjection = 0;
    end
```

The storage system is capable of storing all the power coming from the supply hence this code is unnecessary.

```
    %% Load-balancing
    PSell    = ΔP - PtoInjection;
    PBuy     = 0;
```

```
    %% Load-balancing

    PSell = 0;
    PBuy  = -ΔP - PfromExtraction;
```

These code snippets involve buying from and selling back to the grid, hence they were removed.

```
    function [PtoInjection, PfromExtraction, PSell, PBuy] = controller(PtoController, ...
        PfromController, EStorage, EStrorageMax, EStrorageMin)

    %% Power surplus or deficit

    ΔP = PtoController - PfromController;

    % Power surplus
    if ΔP ≥ 0

        %% Storage injection

        % Storage system is not full
        if EStorage < EStrorageMax

            PtoInjection = ΔP;

        %% Storage extraction

        PfromExtraction = 0;


    % Power deficit
    else

        %% Storage injection

        PtoInjection = 0;

        %% Storage extraction

        % Storage system is not empty
        if EStorage > EStrorageMin

            PfromExtraction = -ΔP;

        % Storage system is empty
        else
            PfromExtraction = 0;
        end
    end
```

Above is the code after making the first round of changes.

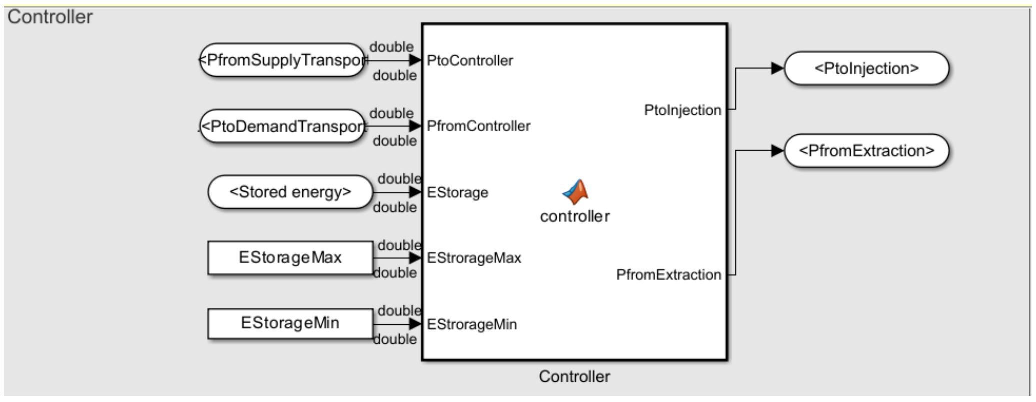


Figure 8: Enter Caption

The controller was also rectified by removing the buying and selling blocks.

As predicted, after these changes the following graphs were obtained (as shown in fig. 9):

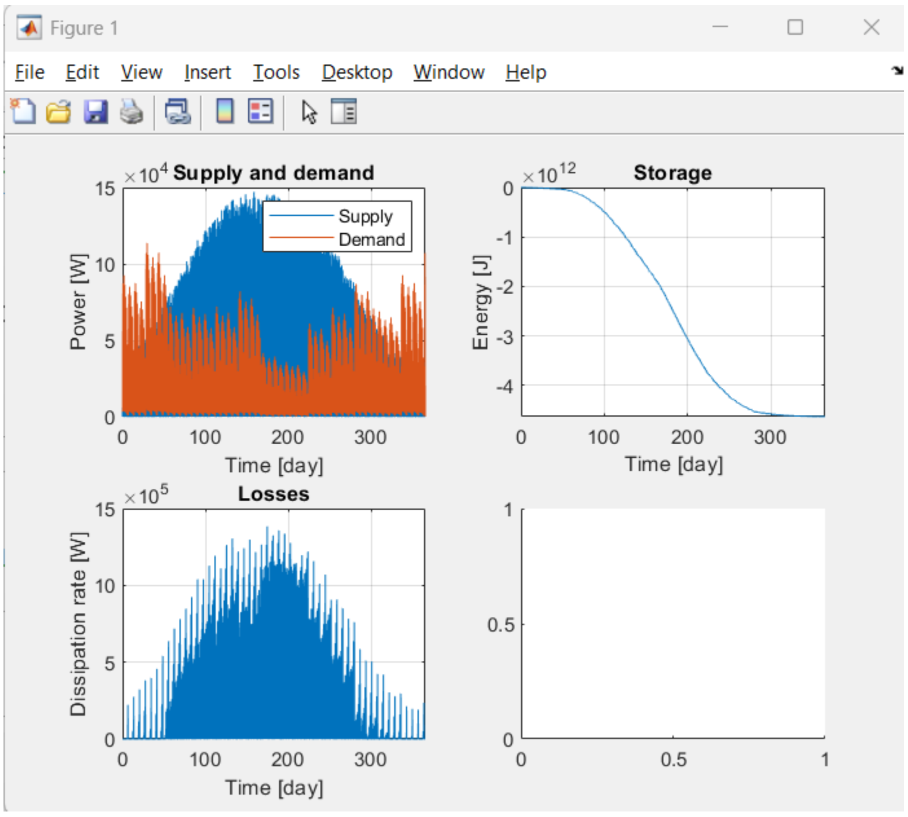


Figure 9: Enter Caption

The load handling graph was expected to disappear. Now, the storage graph needs to be fixed. While trying to understand why the storage graph goes negative, a revelation was made. In the pre-processing file, the EStorageMax variable was set to 113.678 kWh. Also, the EStorageInitial variable was set to 0 kwH.

This is wrong. The maximum Energy that needs to be stored in one day is 113.678 kwH, not the maximum energy that can be stored by the PHS system over the duration of 365 days. The maximum energy that can be stored by the PHS system is around 93,000 kwH.

Secondly, there has to be some energy initially in the battery, As seen in graph 1 of fig. 9, the time period starts with a supply deficit. Under the assumption that the battery has charged in the previous year's supply surplus, it should have some energy to counter the initial supply deficit.

```
%Extracts initial power that needs to be supplied by battery
Initial_Power_Battery=sum(Demand.Data);

% storage system
EStorageMax     = 9.3e4*unit("kWh"); % Maximum energy
EStorageMin     = 0.0*unit("kWh"); % Minimum energy
EStorageInitial = Initial_Power_Battery*0.25*unit("kWh"); % Initial energy
bStorage        = 1e-6/unit("s");  % Storage dissipation coefficient
```

After making these changes and running the model, the following graphs were obtained (as shown in fig. 10)

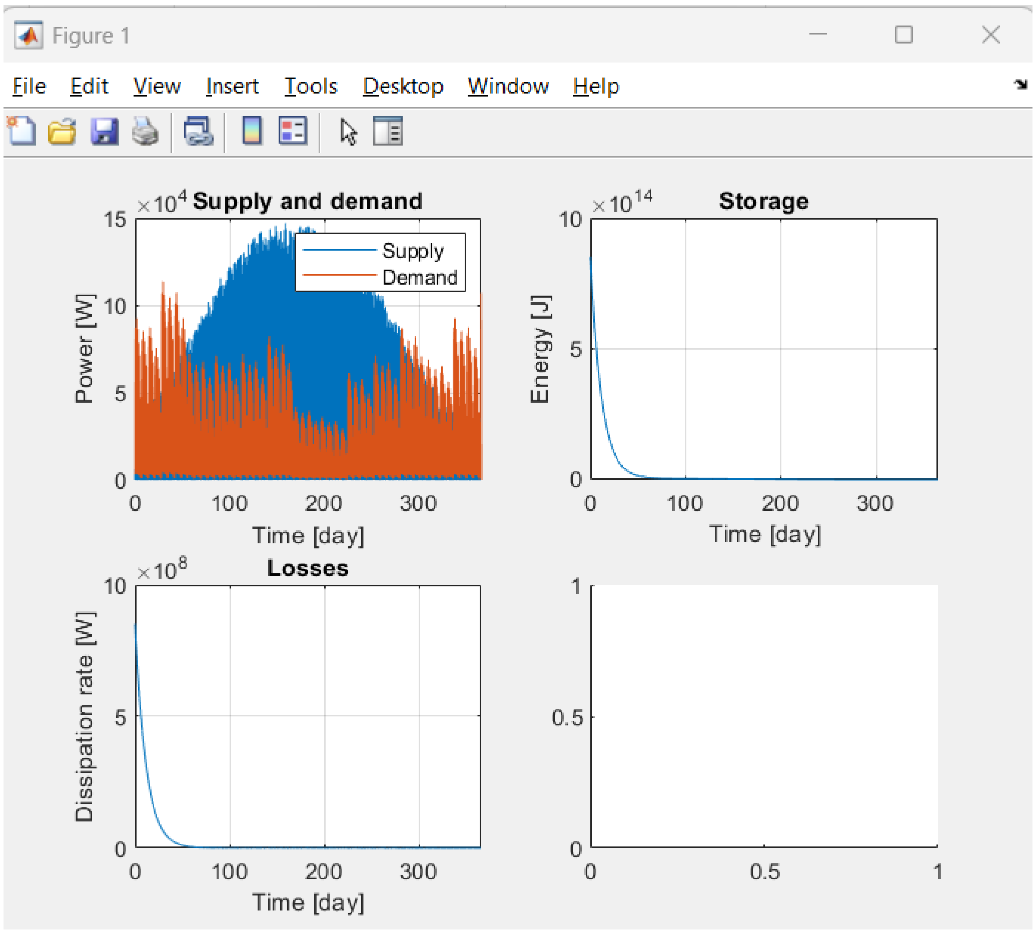


Figure 10: Graphs

The battery discharges after 50 days and then neither charged nor discharged further. This needs to be investigated. Even in the supply surplus the battery doesn't seem to be storing any energy.

In fact it appears that the initial energy put into the system dissipates in the form of losses instead of being supplied. This indicates that perhaps the loss coeffecients are inaccurate/inappropriate.

Looking into bstorage, it was initially set to 10^{-6}, however in reality the bstorage term is extremely close to 0. No water is lost in storage since it is stored underground in a closed reservoir. hence the bstorage term is made much smaller

```
EStorageMax     = 9.3e4*unit("kWh"); % Maximum energy
EStorageMin     = 0.0*unit("kWh"); % Minimum energy
EStorageInitial = Initial_Power_Battery*0.25*unit("kWh"); % Initial energy
bStorage        = 1e-30/unit("s");  % Storage dissipation coefficient
```

Now, these are the generated graphs and pie charts (as shown in fig. 11 and fig. 12):

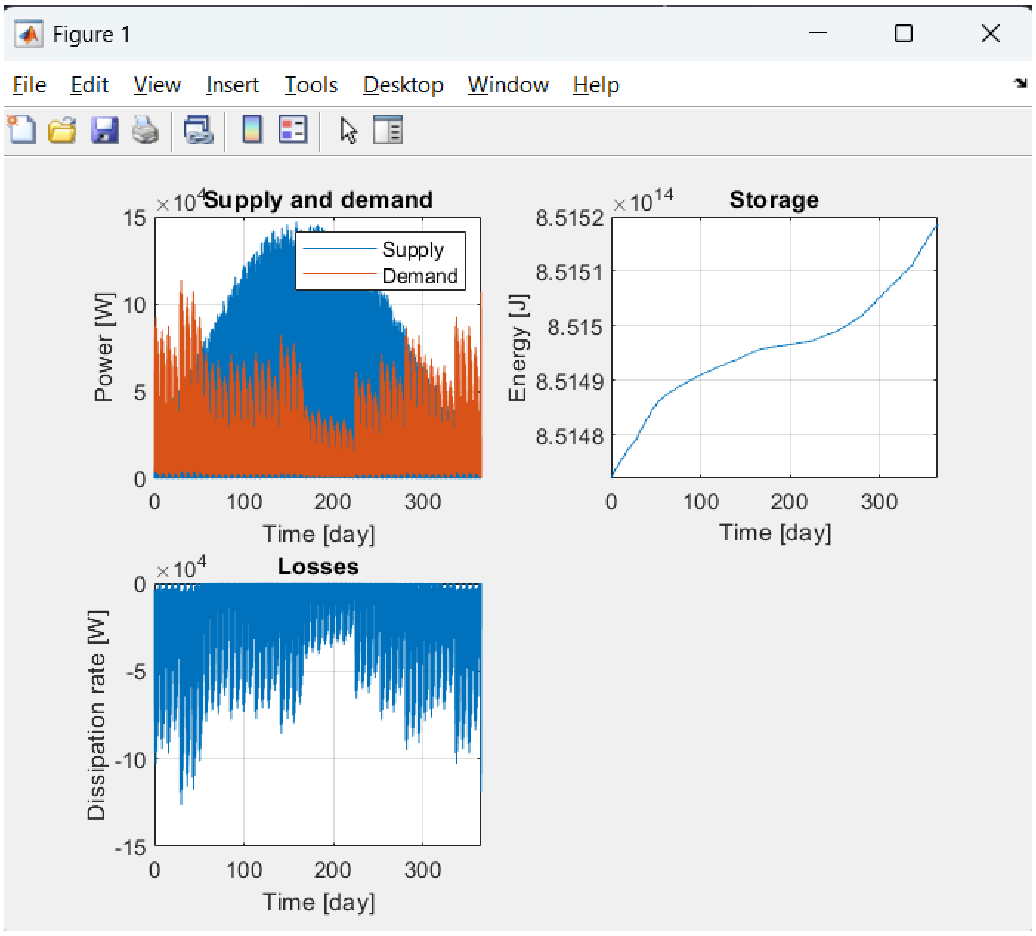


Figure 11: Graphs

To an extent these graphs make more sense now. The battery is charging in the surplus, however it appears that it never supplies this energy during the deficit period between day 250 and 365 (approximately). The losses are shown to be negative which is also highly inaccurate.

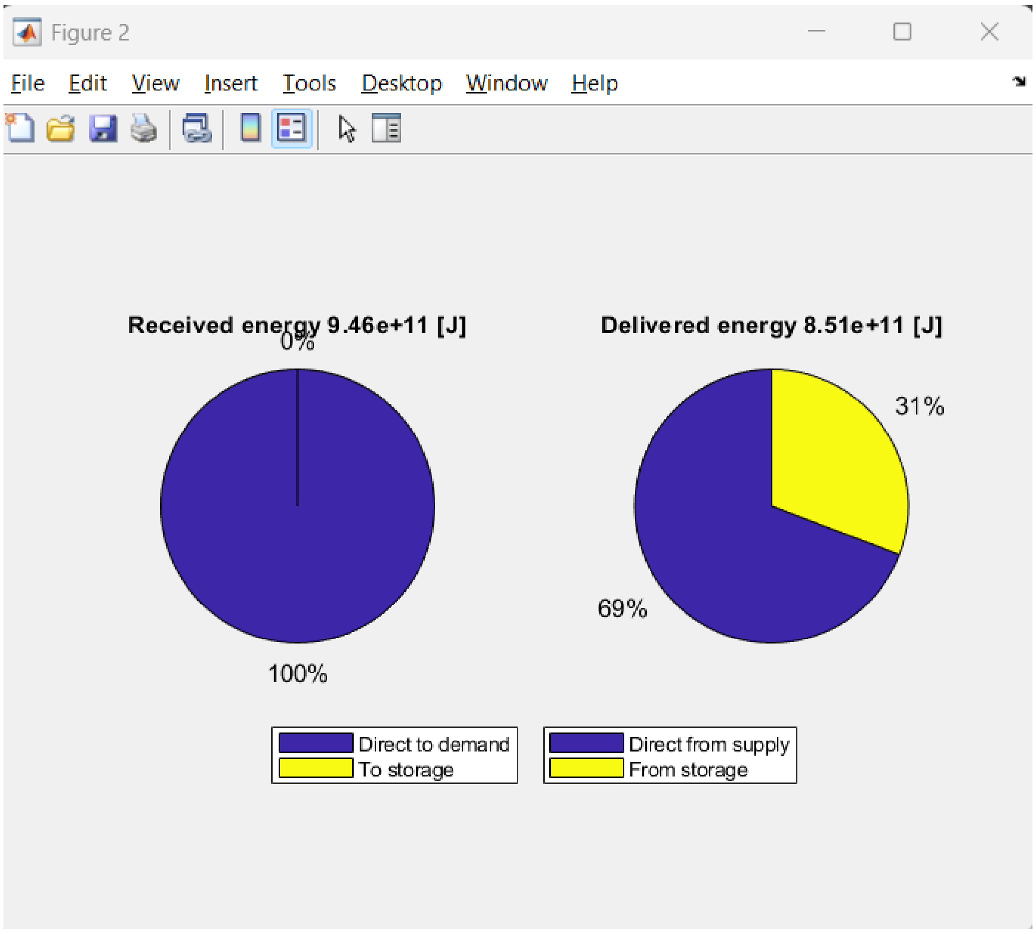


Figure 12: Pie Charts

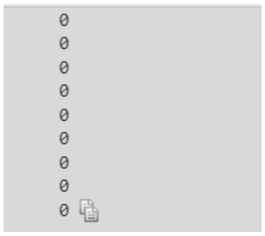
The pie chart on the left is wrong. Not all the received energy goes direct to demand, a good amount of it is stored. The pie chart on the right seems correct, majority of the delivered energy comes direct from the supply (as seen in graph 1 of fig. 11).

Firstly, to correct the discharging of the battery in the 250-365 deficit period, the PtoInjection and PfromExtraction function in the controller block needs investigation.

In an effort to debug it, PtoInjection and PfromExtraction values were displayed over the simulation

```
    function [PfromInjection, DInjection] = injection(PtoInjection, aInjection)
    DInjection = aInjection * PtoInjection;
    PfromInjection = PtoInjection - DInjection;
    disp(PtoInjection)
    disp(PfromInjection)
```

This gave the following result (as shown in fig. 13):



```
     0
     0
     0
     0
     0
     0
     0
     0
     0
```

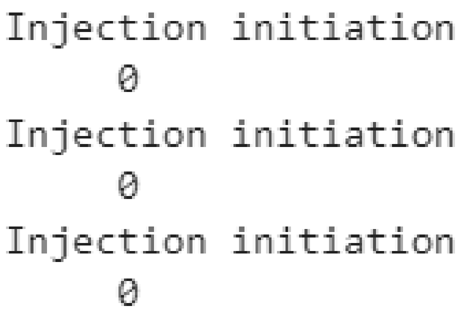
Figure 13: No Power Injected

The value of PtoInjection as well as PfromInjection was 0 consistently throughout the simulation, indicating that no power was ever injected into the storage system.

This indicates that the controller block logic for power injection is flawed. To check if PtoInjection was being properly defined and if the deltaP condition was being activated the following was done:

```
    if ∆P ≥ 0
        %% Storage injection
        disp(Injection initiation)
        % Storage system is not full
        if EStorage < EStrorageMax
            PtoInjection=∆P
            disp(PtoInjection)
        end
```

This gave the following result (as shown in fig. 14):



```
Injection initiation
     0
Injection initiation
     0
Injection initiation
     0
```

Figure 14: Controller Test Result

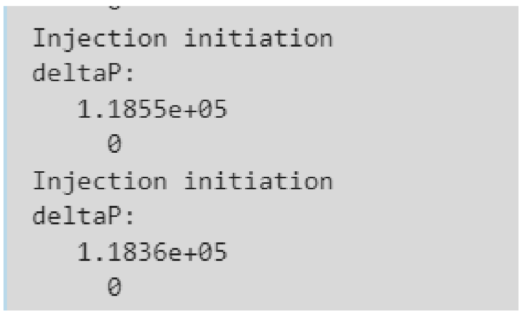
This indicates that the deltaP condition was being activated, however PtoInjection is always 0. This could mean a few things:

1. Either deltaP = 0
2. EStorage is always greater than EStrorageMax, therefore the initial energy given to the battery is too high
3. The battery doesn't have enough capacity to hold the neccesary energy

Both of these are checked. First, checking the value of deltaP and PtoInjection together:

```
    if ∆P ≥ 0
        disp("Injection initiation")
        disp("∆P:")
        disp(∆P)
        %% Storage injection
        % Storage system is not full
        if EStorage < EStrorageMax
            PtoInjection=∆P;
            disp("PtoInjection:")
            disp(PtoInjection)
        end
```

The result was the following (as shown in fig. 15):

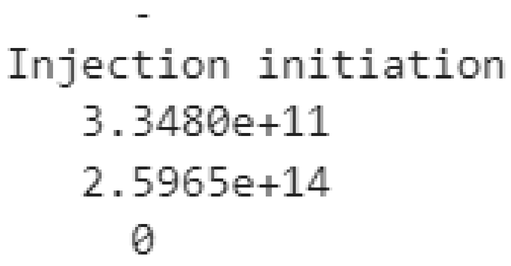


```
Injection initiation
deltaP:
   1.1855e+05
     0
Injection initiation
deltaP:
   1.1836e+05
     0
```

Figure 15: Inability to pass deltaP value to PtoInjection

This indicates that the Estorage<EStrorageMax condition never becomes true. Looking into the initial EStorage value, it was set to be too high. It was then adjusted accordingly. This still did not solve the pertaining issue.

Finally the EStorage vs. EStrorageMax values were investigated, this shows that EStorage was nearly an order of 3 (10^3) higher than EStorageMax. This was an indication that there was an issue with the units that were being used to obtain EStorage. Investigating the integrator, it was integrating power in kWs instead of kWh due to the time step in seconds. Hence this was fixed by dividing the integrated energy value by 3600.



```
Injection initiation
   3.3480e+11
   2.5965e+14
     0
```

Figure 16: Enter Caption

After making the fixes the following graphs were obtained (as shown in fig. 17 and fig. 18):

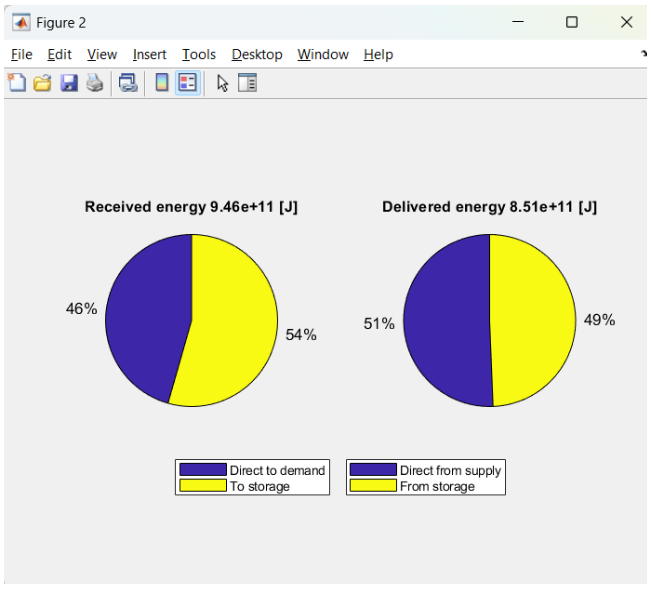


Figure 17: Pie Charts

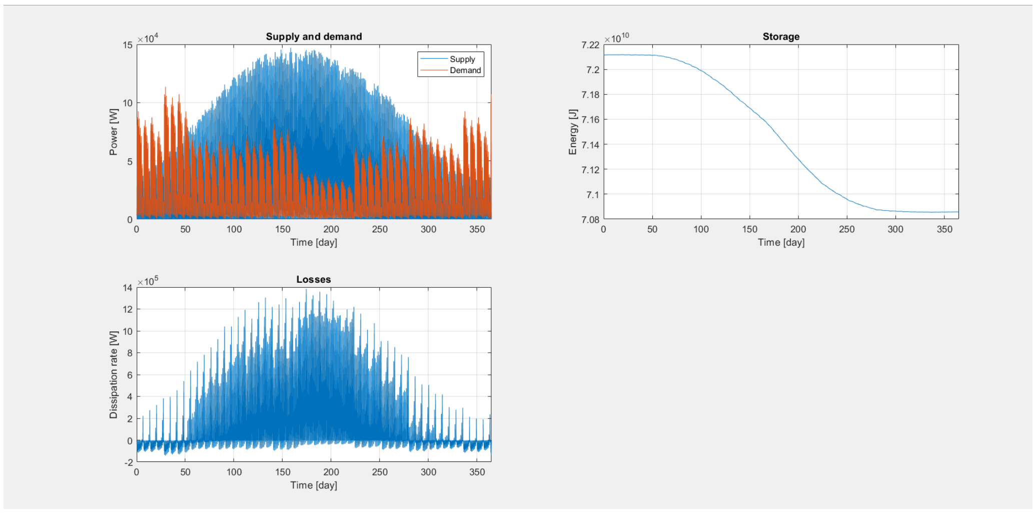


Figure 18: Graphs

While the pie charts are not in complete agreement with the top right graph in fig. 18, this can be further investigated in a future SSA.

Overleaf Link to this SSA

`https://www.overleaf.com/read/vbbwhynfkhmt#08ac9d`

References

[1] *EST - Home*. URL: `https://est.wtb.tue.nl/`. (accessed: 04.05.2025).

[2] *GitHub - Energy Storage and Transport*. URL: `https://github.com/Energy-Storage-and-Transport/EST-model/#readme`. (accessed: 04.05.2025).

[3] *Why Concrete is The Best Material for Water Tanks*. URL: `https://versatiletanks.au/why-concrete-is-the-best-material-for-water-tanks/`. (accessed: 04.05.2025).