Group 6

Technical report
Energy Storage and Transport project

Eindhoven, June 14, 2024

Group members (given name, surname, student ID):	
Athanasiou, Angelos, 1966790	
de Jong, Taeke, 1974653	
Mulkens, Daniel, 1937871	
Ryjoukhine, Nicolas, 2006952	
Sindic, Ryan, 1957198	
Stellinga, Jelmer, 1991450	
Date:	**Group number:**
June 14, 2024	Group 6
Coach/teacher:	
Verhoorsel, Clemens	
Tutor:	
Kooy, Lars	

Contents

list of symbols

Table 1: list of symbols

Symbol	dimension	Unit	Unit abbreviation
k	$kg \cdot m \cdot s^{-3} \cdot k^{-1}$	Watt per meter-kelvin	$W/(m \cdot K)$
Q	$kg \cdot m^2 \cdot s^{-2}$	Joule	J
T	K	Kelvin	K
p	$kg \cdot m^{-1} \cdot s^{-2}$	Pascal	Pa
a	$kg \cdot m^5 \cdot s^{-2} \cdot mol^{-2}$	Liter squared per mole squared	$L^2 \cdot bar/mol^2$
b	2	Liter per mole	L/mol
V	m^3	Cubic meter	m^3
n	-	Number of moles	-
R	$kg \cdot m^2 \cdot s^{-2}c \cdot K^{-1} \cdot mol^{-1}$	Joule per kelvin-mole	$J/(K \cdot mol)$
ρ	$kg \cdot m^3 \cdot s^{-3} \cdot A^{-2}$	Ohm meter	$\Omega \cdot m$
W	$kg \cdot s^2 \cdot s^{-1}$	Joule	J
κ	2	3	4
γ	2	3	4
η	-	Efficiency factor	-
R	$kg \cdot m^2 \cdot s^{-3} \cdot A^{-2}$	Ohm	Ω
A	m^2	Square meter	m^2

1 Energy storage and transport system

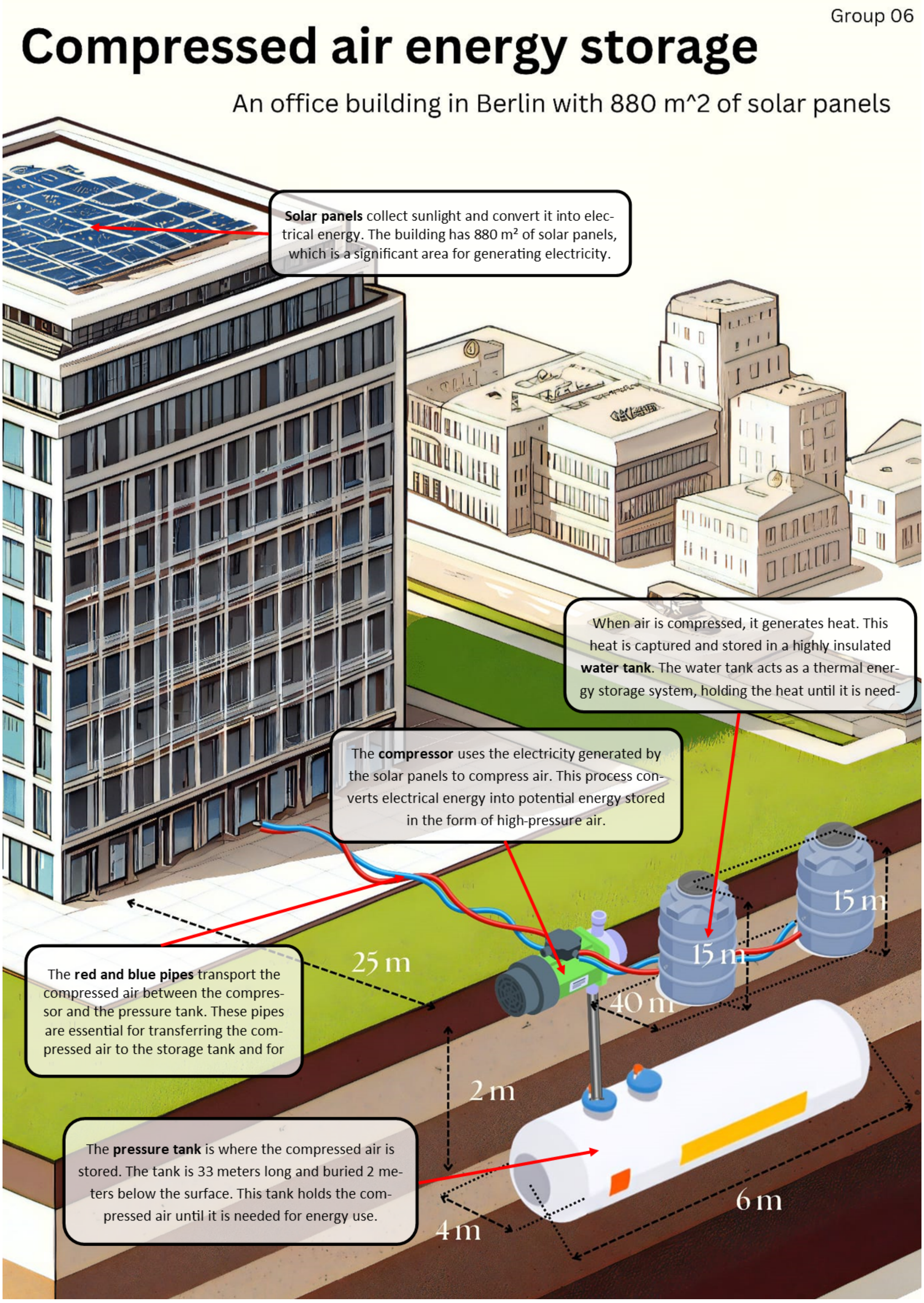

Figure 1: Annotated Infographic

2 MATLAB/Simulink model implementation

All of files for this CBL are available on this GitHub (https://github.com/NicolasRyj/4CBLA30)

2.1 The canvas (block structure)

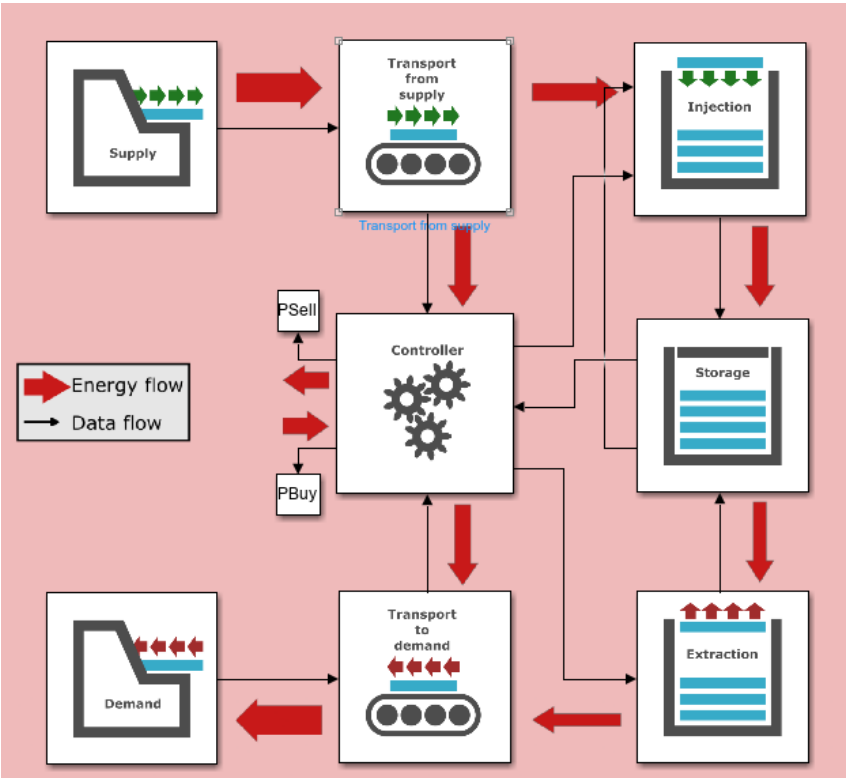

Figure 2: modified canvas

2.2 System parameters

```
%% System parameters

% Standart air constants
R = 8.31; %J/Kmol
kappa_air = 1.4; %-
density_air= 1.255; %kg/m^3
molar_mass_air= 28.96; %g/mol
atmStd = 101350; %Pa
initialP= 101350;% pa initial pressure
cvAir = 718;
%pipe information
D_pipe=0.1;%m diameter of the pipe going from the compressor to the tank
L_pipe=2.5;% m length of pipe going from the compressor to the tank
rho_pipe= 1.225; % kg/m^3 density of the fluid
F_d_pipe= 0.016; %friction factor od the pipe

% Efficiencies of subcomponents
eta_compressor = 0.75;
eta_heat_exchanger = 0.8;
E_loss=50;

% energy loss in electricity wire
rho_wire= 7.54e-5; % ohme/meter constant from data in ssa 3
lWire= 10; % meters length of the wire from solar pannels to storage
aWire= pi*0.0181^2; % area of cable data from in ssa 3
R_compressor= 12000; % ohm average resistance of compressor
Voltage= 500;% Voltage of the compressor system

% transport from supply
aSupplyTransport = 0; % Dissipation coefficient
% injection system
aInjection = 0.1; % Dissipation coefficient

% Conduction parameters;
envTemp = 293.15 ;% kelvin
k_isolation = 0.05; % wats per kelvin per meter^2 for isolation
k_steel = 50; % wats per kelvin per meter^2 for stainless steel
dx_steel= 0.01; %m thickness of the steel wall of the water tank
dx_isolation=1; %m thickness of the isolation arounf the water tank
%Van der waals constants
aVDW = 0.137258; %(Liters^2*KPa)/mol^2 - Van der Walls constant a
bVDW = 0.0000372294; % Van der Walls constant b, (liters/mol)

% Water tanks parameters (2 tanks)
H_watertank= 15; %m the height of the water tanks.
R_watertank = 20;
vWater = H_watertank*pi*R_watertank^2; %m^3 volume of the water and thus of the tank
↪ for each tank
cWater = 4186; %Heat capacity of water
```

```
densityWater = 1000;

% How much each tank is initially filled up
tankColdPercentFull = 1;
tankHotPercentFull = 1-tankColdPercentFull;

% Tank initial temperatures
coldTankTemp = 273.15+20;
hotTankTemp = 273.15+90;

% Air Tank specifications
rTank = 2; % tank radius, meters
lTank = 6; % tank length,  meters
thicknessTankWall = 0.2 ; % tank wall thickness in meters
aTank = 2*pi*rTank^2 + 2*pi*rTank*lTank ; % meter^2
vTank = lTank*pi*rTank; % meter^3
tankMaxPressure = 10*atmStd;
tankMinPressure = 1*atmStd;
tankDepth = 2 ; % how deep the tank is burried in the dirt, meters
nLostPerH = 0; % How many moles lost due to leaks per hour
```

2.3 The code for the mathemtatical-physical models

Description: Calculates the flow rate through a compressor based on the pressure ratio, using a complex polynomial relationship to determine the mass flow rate.

```
function [q, pRatio] = flowRateCompressor(p_1, p_2, rho)
    pRatio = p_2 / p_1; % Calculate pressure ratio
    x = pRatio; % Easier to read
    % Polynomial terms calculated using coefficients
    term1 = 0.166502;
    term2 = 2.93178 * 10^-7 * (684128 * sqrt((1.68491 * 10^21) * x^2 - ...
            (6.16758 * 10^21) * x + 5.64261 * 10^21) - ...
            (2.80818 * 10^16) * x + 5.13965 * 10^16)^(1/3);
    term3 = 2579.51 / (684128 * sqrt((1.68491 * 10^21) * x^2 - ...
            (6.16758 * 10^21) * x + 5.64261 * 10^21) - ...
            (2.80818 * 10^16) * x + 5.13965 * 10^16)^(1/3);
    mq = term1 + term2 + term3; % Sum of terms to get mass flow rate
    q = mq / rho; % Adjust for density to get volumetric flow rate
end
```

Equation:

$$\begin{aligned}
A &= \sqrt{1.68491 \times 10^{21} \times pRatio^2 - 6.16758 \times 10^{21} \times pRatio + 5.64261 \times 10^{21}}, \\
B &= (684128 \times A - 2.80818 \times 10^{16} \times pRatio + 5.13965 \times 10^{16})^{\frac{1}{3}}, \\
C &= 2.93178 \times 10^{-7} \times 684128 \times B, \\
D &= 2579.51/B, \\
q &= \frac{1}{\rho}\left(0.166502 + C + D\right).
\end{aligned} \tag{1}$$

Explanation: These equations determine the flow rate through a compressor by calculating the pressure ratio p_2/p_1 and applying it within a complex polynomial to estimate the flow rate. Here, p_1 and p_2 represent the inlet and outlet pressures, respectively, while ρ is the fluid density. The output includes both the pressure ratio and the volumetric flow rate q.

Description: Calculates the specific heat at constant pressure (c_p) as a function of pressure (p).

```
function c_p = pressureSpecificHeat(p)
    kp = p / 1000; % Convert pressure from Pa to kPa
    c_p = 1.00501 * exp(0.0000162361 * kp); % Compute specific heat
end
```

Equation:

$$
\begin{aligned}
kp &= \frac{p}{1000}, \\
c_p &= 1.00501 \times \exp(0.0000162361 \times kp).
\end{aligned}
\tag{2}
$$

Explanation: This function computes the specific heat at constant pressure (c_p) for a given pressure (p). The pressure p is first converted from Pascals to kilopascals (denoted as kp). The specific heat is then calculated using an exponential function (data for this equation was gathered experimentally [4]), which accounts for the effect of pressure on specific heat.

Description: Calculates the density of air in a tank (ρ_{TankAir}) using the ideal gas law.

```
function rhoTankAir = densityAirInTank(p1, R, T)
    rhoTankAir = p1 / (R * T); % Compute density using the ideal gas law
end
```

Equation:

$$
\rho_{\text{TankAir}} = \frac{p1}{R \times T}
\tag{3}
$$

Explanation: This function calculates the density of air in a tank using the ideal gas law, where $p1$ is the pressure within the tank, R is the specific gas constant for air, and T is the absolute temperature. The density, ρ_{TankAir}, is determined by dividing the pressure by the product of the gas constant and temperature, reflecting the direct relationship between these variables in the state equation of an ideal gas.

Description: Calculates the mass of air (m) given its density (ρ) and volumetric flow rate (q).

```
function m = mass_air(rho, q)
    V = q * 300; % Calculate volume using flow rate over a time period (300
    ↪  seconds)
    m = rho * V; % Calculate mass as product of density and volume
end
```

Equation:

$$
\begin{aligned}
V &= q \times 300, \\
m &= \rho \times V.
\end{aligned}
\tag{4}
$$

Explanation: This function computes the mass of air by first determining the volume of air passed through a region in a given time period, assuming the flow rate (q) is constant and the time period is 300 seconds. The mass (m) is then calculated by multiplying the air's density (ρ) by this volume (V). This approach uses the basic principle that mass is the product of density and volume.

Description: Calculates the temperature rise (ΔT) in a medium due to heating, considering the heat input rate (Q), efficiency (eff), specific heat (C_p), volumetric flow rate (q), and density (ρ).

```
function dt = temperature_rise(Q, eff, C_p, q, rho)
    P = (Q / 300) / 1000; % Convert heat input rate to kW and normalize over time
    dt = (P * (1 - eff)) / (C_p * q * rho); % Calculate temperature rise
end
```

Equation:

$$\begin{aligned} P &= \frac{Q}{300 \times 1000}, \\ \Delta T &= \frac{P \times (1 - \text{eff})}{C_p \times q \times \rho}. \end{aligned} \tag{5}$$

Explanation: This function determines the temperature rise (ΔT) in a medium when heat is added. First, the power (P) is calculated by normalizing the heat input rate (Q) over a period of 300 seconds and converting it to kilowatts. The temperature rise is then computed based on the non-utilized portion of this power (considering efficiency), distributed over the mass of the medium, which is calculated using the specific heat (C_p), the flow rate (q), and the density (ρ). The formula shows that the temperature increase is inversely proportional to the product of specific heat, flow rate, and density.

Description: Calculates the energy transferred to water (Q_{water}) based on the mass of the water (m), the temperature change (Δt), and the specific heat capacity of air (C_{air}).

```
function Q_water = energy_to_water(m, dt, C_air)
    Q_water = m * dt * C_air; % Calculate the energy based on mass, ...
    temperature change, and specific heat
end
```

Equation:

$$Q_{\text{water}} = m \times \Delta t \times C_{\text{air}} \tag{6}$$

Explanation: This function computes the amount of energy transferred to water when its temperature is increased. The energy required (Q_{water}) is calculated by multiplying the mass of the water (m) by the temperature change (Δt) and the specific heat capacity of air (C_{air}), which is used here as a proxy for the specific heat of the water. This relationship is based on the principle of energy conservation and the formula for heat transfer, $Q = mc\Delta T$, where c is the specific heat.

Description: Calculates the updated pressure (p_2) in a pressure tank and the energy transferred to water (Q_{water}), based on the initial heat and pressure energy inputs, maximum and minimum pressure limits, the gas constant, tank volume, specific heat at constant volume, and initial pressure.

```
function [p2, Qwater] = pressuretankCalculations(Q_h, Q_p, maxP, minP, R, V, Cv_air, p1)
    pAddAvailable = maxP - p1;
    pRemoveAvailable = p1 - minP;
    pChangeFromWork = (R * (Q_p - Q_h)) / (V * Cv_air - V * R);
    if pressureAddAvailable >= pChangeFromWork || pRemoveAvailable >= pChangeFromWork
        p2 = p1 + (R * (Q_pressure - Q_heat)) / (V * Cv_air - V * R);
        Qwater = Q_p - Q_h;
    else
        percentWorkCanBeAdded = pAddAvailable / pChangeFromWork;
        Q_add = Q_p * percentWorkCanBeAdded;
        p2 = p1 + (R * Q_add) / (V * Cv_air - V * R);
        Qwater = (1 - percentWorkCanBeAdded) * Q_p + Q_h;
    end
```

Equation:

$$
\begin{aligned}
\Delta p &= \frac{R \times (Q_{\text{pressure}} - Q_{\text{heat}})}{V \times (C_{v,\text{air}} - R)}, \\
p_2 &= p_1 + \Delta p, \\
Q_{\text{water}} &= \begin{cases} Q_{\text{pressure}} - Q_{\text{heat}}, & \text{if } \Delta p \text{ is within limits} \\ (1 - \text{percentWorkCanBeAdded}) \times Q_{\text{pressure}} + Q_{\text{heat}}, & \text{otherwise} \end{cases}
\end{aligned} \tag{7}
$$

Explanation: This function assesses whether the energy changes (due to heating or pressurizing) can be fully accommodated within the existing pressure limits of the tank. If the calculated pressure change from the input energies is within the allowable range, the new pressure (p_2) and the energy to the water (Q_{water}) are straightforwardly calculated. If not, the function computes what proportion of the work (energy addition) can be done within the pressure limits and adjusts the outputs accordingly. This helps in managing the safety and efficiency of the tank operations, ensuring pressures do not exceed specified bounds.

Description: Calculates the adjusted masses of two water tanks due to heat transfer between them, given their initial masses and temperatures, and the energy involved in the heat transfer.

```
function [tank1_m, tank2_m, tTankH] = heatCaptureTanks(tank1cold_m, tank2hot_m, Q_water, c, tank_1_temp, tank_2_temp)
    dMass = Q_water / c * (tank_2_temp - tank_1_temp);
```

```
    tank1_m = tank1cold_m; % Default to input value if condition is not met
    tank2_m = tank2hot_m;  % Default to input value if condition is not met
    if tank1cold_m - dMass > 0 && tank2hot_m + dMass > 0
        tank1_m = tank1cold_m - dMass;
        tank2_m = tank2hot_m + dMass;
    end
    tTankH = 273.15 + 90; % Set tank temperature to 90 degrees Celsius above absolute zero
```

Equation:

$$
\begin{aligned}
\Delta m &= \frac{Q_{\text{water}}}{c} \times (\text{tank\_2\_temp} - \text{tank\_1\_temp}), \\
\text{tank1\_m} &= \text{tank1cold\_m} - \Delta m, \\
\text{tank2\_m} &= \text{tank2hot\_m} + \Delta m.
\end{aligned}
\tag{8}
$$

Explanation: This function computes the mass transfer between two tanks due to a temperature-driven flow of heat. It ensures that mass transfer does not result in negative values, reflecting physical conservation laws. The final temperatures are assumed to reach a nominal value set by the *tTankH* variable, indicative of the heat capture process.

Description: Computes the heat dissipation losses from a hot water tank, considering tank dimensions, insulation properties, and the environment's temperature.

```
function Q_lost = dissipationLoss(tTank, envTemp, V_water, H_watertank, dx_steel, k_steel, dx_isolation, k_isolation, hotWaterMass, netWaterMass)
    % Calculates energy losses due to convection
    r_tank = sqrt(V_water / (H_watertank * pi));
    A_steel = 2 * pi * (r_tank + dx_steel) * H_watertank + 2 * pi * (r_tank + dx_steel)^2;
    A_isolation = 2 * pi * (r_tank + dx_steel + dx_isolation) * H_watertank + 2 * pi * (r_tank + dx_steel + dx_isolation)^2;
    R_steel = dx_steel / (k_steel * A_steel);
    R_isolation = dx_isolation / (k_isolation * A_isolation);
    R_total = R_steel + R_isolation;
    P_lost = (tTank - envTemp) / R_total;
    Q_lost = P_lost * (hotWaterMass / netWaterMass);
```

Equation:

$$
\begin{aligned}
R_{\text{steel}} &= \frac{dx_{\text{steel}}}{k_{\text{steel}} \times A_{\text{steel}}},\\
R_{\text{isolation}} &= \frac{dx_{\text{isolation}}}{k_{\text{isolation}} \times A_{\text{isolation}}},\\
R_{\text{total}} &= R_{\text{steel}} + R_{\text{isolation}},\\
P_{\text{lost}} &= \frac{tTank - \text{envTemp}}{R_{\text{total}}},\\
Q_{\text{lost}} &= P_{\text{lost}} \times \frac{\text{hotWaterMass}}{\text{netWaterMass}}.
\end{aligned} \tag{9}
$$

Explanation: This function estimates the heat energy lost from a hot water tank due to its surrounding environment. It incorporates the thermal resistances of the tank's steel and its insulation to compute the total thermal resistance. The heat loss is then calculated based on the temperature difference between the tank and the environment, normalized by the total resistance, and adjusted for the proportion of hot water mass to the total water mass.

3 Model application

3.1 The simulation scenario

The problem scenario given to the group at the beginning of the project consisted of a large office building in Berlin with $880m^2$ of solar panels on the roof with a yearly total demand of energy of $8.51 \cdot 10^{11} J$. The more specific demand and supply values were also given in csv data files. An important fact that is given with this information is the location of the scenario, due to the fact that the office building is located in Berlin, a densely populated area, a lot of energy systems would not be applicable due to the fact that there is a lack of space for such systems. There is also a lack of higher laying terrain like mountains or hills that can be used to store energy. That is why the group settled for a solution that would not use a relatively large area for the amount of required stored energy and is applicable anywhere. The size of our system is mainly dependent on the fluid tank to store the heat, because of that the variations are made in the fluid.

3.2 Results and interpretation

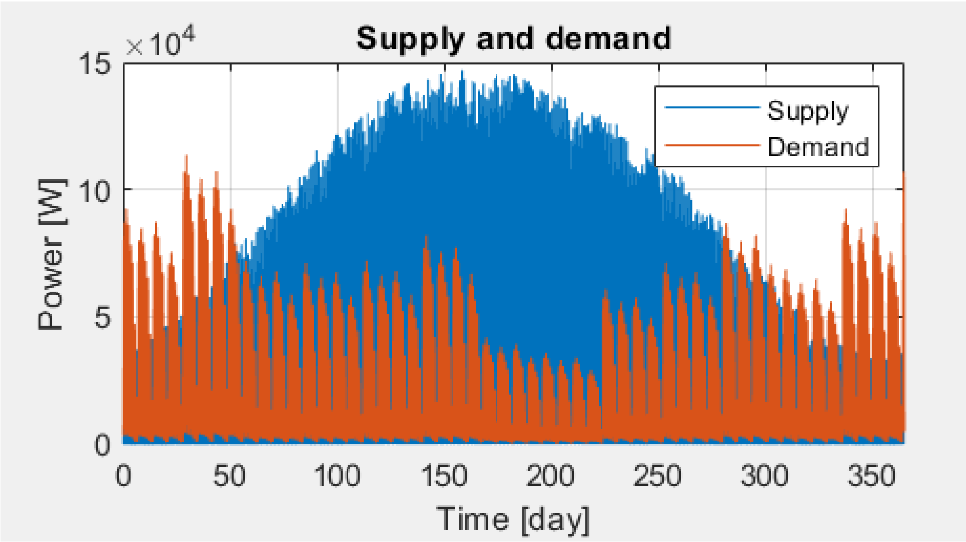

Figure 3: Supply and Demand graph

Figure 3 depicts the given total supply and demand for the system. The graph depicts a total of 365 days, one year, with day one starting at January 1st. What is clear from this graph is that the solar supply of the office building produces relatively more energy during the summer season and has barely any supply during the winter. This is where the energy storage system comes into play. As the demand is relatively higher in the winter season.

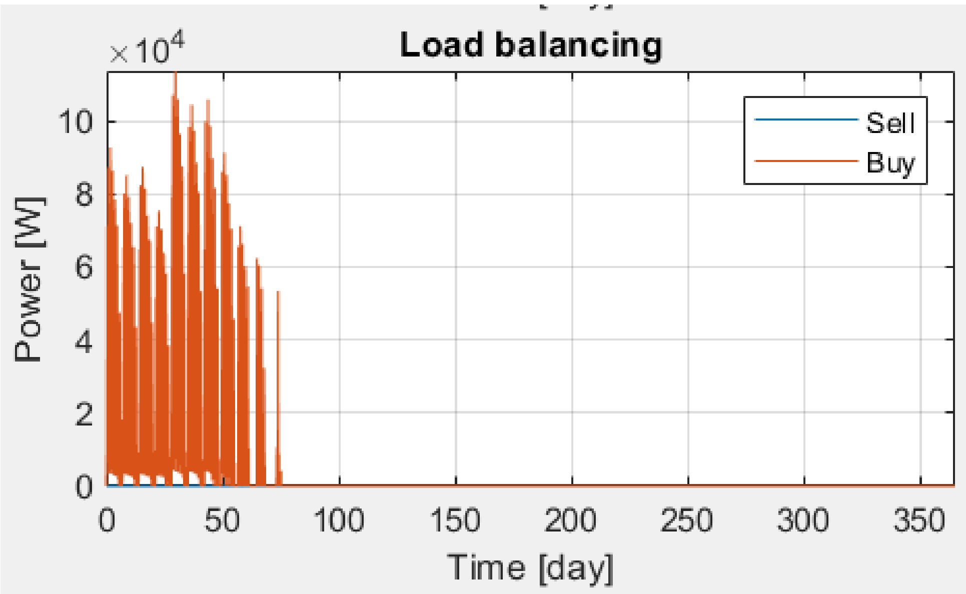

Figure 4: Load balancing graph

Figure 4, depicts the amount of purchased power in Watts relative to time in days. As the system starts running its first year it will have to purchase some power to kick start the system. The second year the system will run, barely any power will need need to be bought, an estimated 1 percent of the total power needed. The information for the second year can be found in the Appendix.

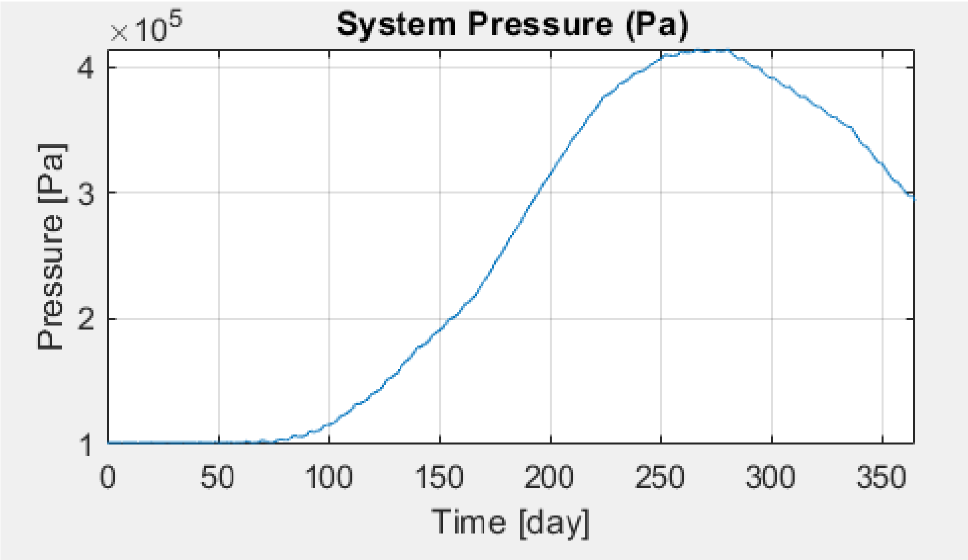

Figure 5: System pressure graph

Figure 5, gives a clear depiction of the total pressure in the system relative to the time. Pressure increases when the supply predominates the demand and vise versa. All the excess produced energy will be transferred to the system and will be stored in the pressure tank, in terms of pressure.

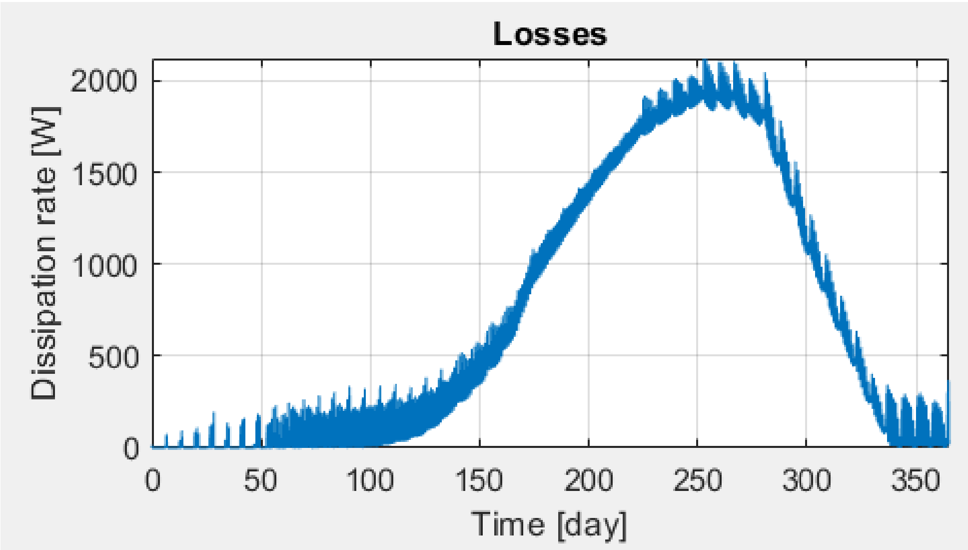

Figure 6: Losses in system graph

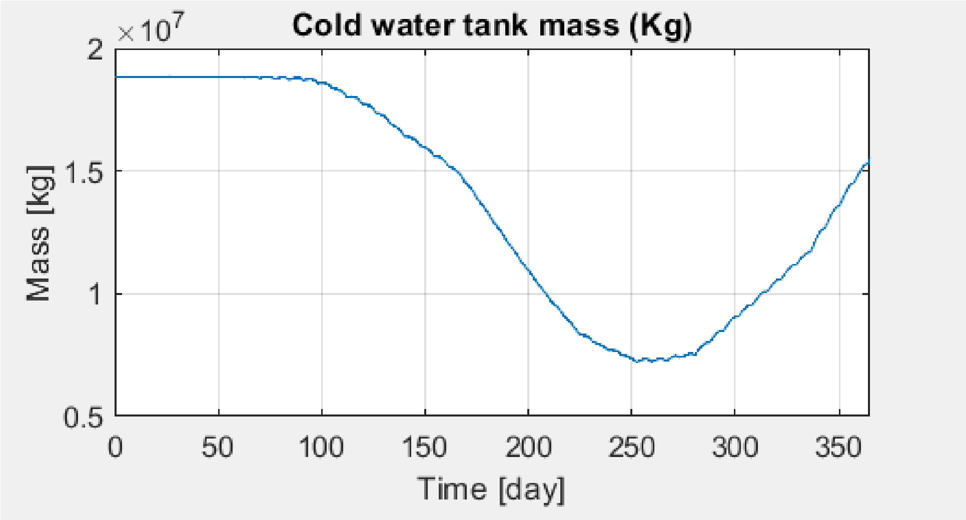

Figure 7: Cold water tank mass graph

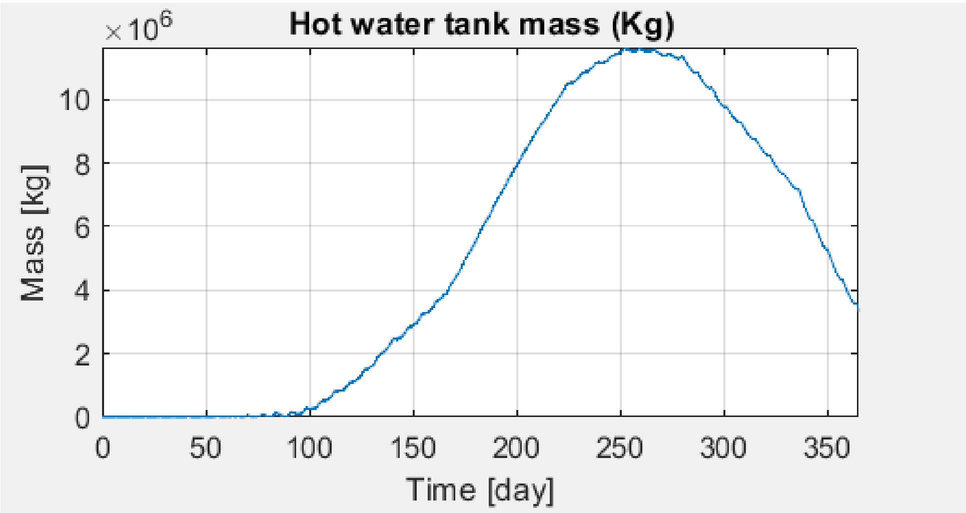

Figure 8: Hot water tank mass graph

Figure 6, this graph shows the total losses relative to time. In this graph it is clear to observe the larger amount of losses during the time when the pressure increases or decreases, as seen comparing it with Figure 5. When increasing and decreasing the pressure of the pressure tank there are various types of losses that play a role. There is a friction loss due to the piping of the pressure tank, a heat loss due to the increase in pressure and losses linked with the compressor.

Figure 7 and Figure 8, depict the cold and hot water volume in the water tank respectively. As the system will try to minimize the losses due to heat a heat exchanger is used, taking the heat from the pressure tank with cold water and storing this now hot water in another separate water tank. A clear comparison between the two graphs is that the cold water volume decreases when the hot water volume increases, due to the fact that the cold water is being transferred and converted to hot water when the pressure tank is being pressured.

4 Model development reflection

4.1 The Simulink model

What physical parameters in the Simulink model can be varied to study their effect on the performance of the EST system? For each parameter, indicate the range for which you have tested your model.
Multiple parameters can be changed in the model. The biggest ones are the volumes of both the water and air tanks. The group studied the effect of decreasing the sizes of the water tank, and as predicted, the water level in the hot tank simply rises until it caps off. Then the energy storage capacity is reached. Changing the volume/maximum allowed pressure inside the gas tank also works in either increasing or decreasing the energy storage capacity.

What is the biggest limitation of the current Simulink model? What steps can be taken to improve the Simulink model w.r.t. this limitation?
The biggest limitation is that currently the model does not have updated controller settings to check if the hot water tank is full, and if so to sell the excess energy. The system currently checks how much more energy can be added to the gas tank, and the rest is simply added to the water without checking if there is any cold water left to heat. If there isn't, the model simply loses the energy without selling. This is a drawback that only happens if there is too little water in the system, which is not the case in the uploaded model. To fix this, the controller's logic needs to be updated to check how much energy "space" is left in both tanks, and if there is not enough "space", the controller should sell.

4.2 Relation with the validation experiment

Where in the Simulink model did you make use of the physical law studied in the validation experiment?
Our model employs water (or any other liquid) to store excess heat generated when compressing gases quickly. In order to transfer the heat back from the water to the gas, as to increase it's pressure, it was required to know how effective transferring the heat would be through a heat exchanger. Our experiment replicated a primitive heat exchanger.

To what extent do the conclusions regarding the validity of the studied physical law (as discussed in your Standard Operatating Procedure) carry over to the real world EST system considered in your Simulink model? Think for example about the differences in time and length scales.
The key diffrences between the experiment we conducted and the real world application of the system come down too two main factors the scaling of size from model to sytem and the length of time the system runs compared to the experiemnt

4.3 Overall process

What is the most important lesson your group learned from the overall model development process, including both the Simulink model development and the validation experiment?
Our group's main lesson from developing the model, including creating the Simulink model and conducting validation experiments, was the importance of precisely capturing physical processes using suitable equations and functions. The challenge of selecting the right formulas, such as deciding between Van der Waals and Ideal Gas Laws for calculating moles in the pressure tank, required foresight to determine if the greater accuracy of Van der Waals justified the complexity. In Simulink, integrating various system components was complicated, necessitating careful implementation to avoid scenarios where our equations failed. Validating the experiment also presented unique challenges, especially ensuring the precision of our measuring equipment and controlling external variables across and within trials. Additionally, balancing the model's level of detail with the introduction of bugs due to increased complexity was a significant issue. Through a cyclical process of refining our model to adequately describe our system while maintaining manageable complexity and validating it with real data, we not only enhanced our technical skills but also emphasized the crucial roles of accuracy and validation in developing reliable energy storage systems.

References

[1] Michel Goossens, Frank Mittelbach, and Alexander Samarin. *The LaTeX Companion*. Addison-Wesley, Reading, Massachusetts, 1993.

[2] Albert Einstein. *Zur Elektrodynamik bewegter Körper*. (German) [*On the electrodynamics of moving bodies*]. Annalen der Physik, 322(10):891–921, 1905.

[3] Knuth: Computers and Typesetting, http://www-cs-faculty.stanford.edu/~uno/abcde.html

[4] The Engineering ToolBox (2009). Air - Specific Heat vs. Pressure at Constant Temperature. [online] Available at: https://www.engineeringtoolbox.com/air-specific-heat-various-pressures-d_1535.html [Accessed 11-06-2024]

Attachments

1. All of files for this CBL are available on this GitHub https://github.com/NicolasRyj/4CBLA30