Master's thesis

2022

Gonzalo Sanfeliu Moreno

NTNU
Norwegian University of Science and Technology
Faculty of Engineering
Department of Civil and Environmental Engineering

Master's thesis

NTNU
Norwegian University of
Science and Technology

Gonzalo Sanfeliu Moreno

Feasibility study of floating solar panels on a hydropower reservoir with winter ice

June 2022

NTNU
Norwegian University of
Science and Technology

Norwegian University of
Science and Technology

Feasibility study of floating solar panels on a hydropower reservoir with winter ice

Gonzalo Sanfeliu Moreno

Nordic Master's Programme in Cold Climate
Submission date: June 2022
Supervisor: Knut Vilhelm Høyland
Co-supervisor: Jukka Tuhkuri

Norwegian University of Science and Technology
Department of Civil and Environmental Engineering

Abstract

This project has been done because of the need of increasing the production of renewable energy in the Nordics. As solar energy is one of the easy options to go with, company are searching for a way of installing solar farms in areas that they already own. As hydropower reservoirs offer a large area, it has become an interesting option for these companies. Furthermore, floating solar panels technology has been greatly developed during the last years, making this option even more appealing. On the other hand, the fact that the reservoir freezes in winter creates some extra challenges that will need to be dealt with. Some of these challenges come from depth change in the reservoir, the forces that can appear towards the reservoir dam because of the wind drag on the solar panels and the extra difficulty of designing systems with unknown parameters due to the lack of proper ice data. Some of the forces that can be produced, such as the ones caused by the thermal stress or the increased drag because of the wind hitting the solar panels, have been estimated by using a self-developed tool, and some recommendations for the systems, as well as future research that would need to be done before engaging in the system construction have been explained.

The main conclusion of this study is that the installation of a floating solar panel system seems technically feasible. Both, the ice forces from the wind drag and from thermal expansions seem to be acceptable, even though some of the environmental data is uncertain. Part of this uncertainty comes from the unknown way in which the floating panels themselves covering most of the water surface will alter the ice conditions. Together with some of the systems such as mooring, and joints will need to be further studied to prove the technical feasibility of the whole project. Furthermore, the economic feasibility of the system, that has not been checked, could completely change the decision of the interested company on moving forward with the project, as some of the components may not be commercially available due to the special requirements because of the cold.

Dette prosjektet er gjort grunnet behovet for å øke produksjonen av fornybar energi i Norden. Siden solenergi er et av de enklere alternativene å gå for, søker selskapene etter en måte å installere solenergifarmer i områder som de allerede eier. Ettersom vannkraftreservoarene tilbyr et stort område, har det blitt et interessant alternativ for disse selskapene. Videre har flytende solcellepanelteknologi blitt kraftig utviklet de siste årene, noe som gjør dette alternativet enda mer attraktivt. På den andre siden gjør det slik at reservoaret fryser om vinteren som skaper ekstra utfordringer som må håndteres. Noen av disse utfordringene kommer fra dybdeforandring i reservoaret, kreftene som kan oppstå mot reservoardammen på grunn av vindmotstanden på solcellepanelene, og den ekstra vanskeligheten med å designe systemer med ukjente parametere på grunn av mangelen på riktige isdata. Noen av kreftene som kan produseres, slik som de som forårsakes av termisk stress eller økt luftmotstand på grunn av vinden som treffer solcellepanelene, er estimert ved å bruke et egenutviklet verktøy, og noen anbefalinger for systemene, samt fremtidig forskning som må gjøres før man engasjerer seg i systemkonstruksjonen, har blitt forklart.

Hovedkonklusjonen i denne studien er at installasjon av et flytende solcellepanelsystem virker teknisk gjennomførbart. Både iskreftene fra vinddraget og termiske ekspansjoner ser ut til å være akseptable, selv om noen av miljødataene er usikre. En del av denne usikkerheten kommer fra den ukjente måten selve flytepanelene som dekker meste av vannoverflaten vil endre isforholdene. Sammen med noen av systemene som fortøyning og skjøter vil det måtte studeres videre for å bevise den tekniske gjennomførbarheten av hele prosjektet. Videre kan den økonomiske gjennomførbarheten til systemet, som ikke er kontrollert, fullstendig endre beslutningen til det interesserte selskapet om å gå videre med prosjektet, da noen av komponentene kanskje ikke er kommersielt tilgjengelige på grunn av de spesielle kravene grunnet kulden.

Acknowledgements

I would like to sincerely thank a lot of people that has helped me during this long journey, but in order not to make the reading of the thesis longer I will have to reduce it to some special people. My deepest gratitude to my supervisor, Knut Vilhelm Høyland, for all the help given when working on the thesis and all the useful comments given on the writing stage, I wouldn't have been able to finish without it.

Thank you to TrønderEnergi for proposing the topic and assisting by providing as much data as they had. A special mention to Maxime Ramstein from Romande Energie for all the questions answered to a student asking for help.

To all my friends that have had to hear my complaints because of not being able to find solutions or because of the code not working countless times and always cheered me up, thank you very much. And, to the two of you that made me stop working so I could rest by making me laugh and doing some outdoors activities, thank you, this wouldn't have been done without those breaks.

Finally, the biggest thanks to my parents that always helped me by encouraging me to continue even in the worst moments.

Contents

Figures

Equations

Tables

Terms

Photovoltaic panel: Device that uses sunlight as a source of energy to generate direct current electricity.

Block unit: The biggest unit in the grid consisting of solar panels units and the necessary electrical components to transform and send the electricity.

Grid unit: Each unit that forms the Block unit, it can contain several solar panels or electric equipment.

Photovoltaic unit: Grid unit formed by solar panels.

Transformer unit: Grid unit containing the necessary electric equipment to transform and send the electricity produced in the photovoltaic units.

1. Introduction

Society is getting more and more interested on clean and sustainable energy. Some countries in Europe having even more than half of the gross energy consumption coming from renewable sources and having a European average of nearly a quarter of the energy consumption being renewable [Figure 1]. Due to the current increase on the renewable energy demands to meet the EU2030 goal and the EU2050 climate neutrality, some companies are studying the possibility to increase the amount of renewable energy production they have. With it, a lot of interest on solar and wind power has arisen, but the way these two technologies are used nowadays has several downsides. The first downside of this technologies is the land occupancy and the environment destruction.

On the other side, there are also some pros on using solar energy against the use of wind energy. First of all, and most importantly, it is cheaper long-term because of the very little maintenance required due to the solar panels not having any moving parts. It is also noticeable that, due to the size of the panels compared with the wind turbines, more panels can be installed in the same area, allowing for a more customizable farm shape. Solar panels are also less conspicuous than turbines, which is an important factor when installing them in the nature, as well as being silent, unlike turbines. Furthermore, the expected solar electricity production is easier to predict than the wind one. Some of the cons include that the wind turbines produce significantly more power than the solar panels, and the upfront cost for them is lower than the one for solar installation (when comparing price per MW installed).

When it comes to solar power production inland, very large areas need to be cleared to be able to install the farms, and this, on top of having an economic cost for the companies, has a big impact on the environment, including local flora and fauna. In order to solve this problem, a new technology consisting of floating solar panels is being developed. This technology is supposed to allow for easier to make systems and increased efficiency, while reducing the extra space needed due to the use of reservoirs that are already being used for hydropower production.

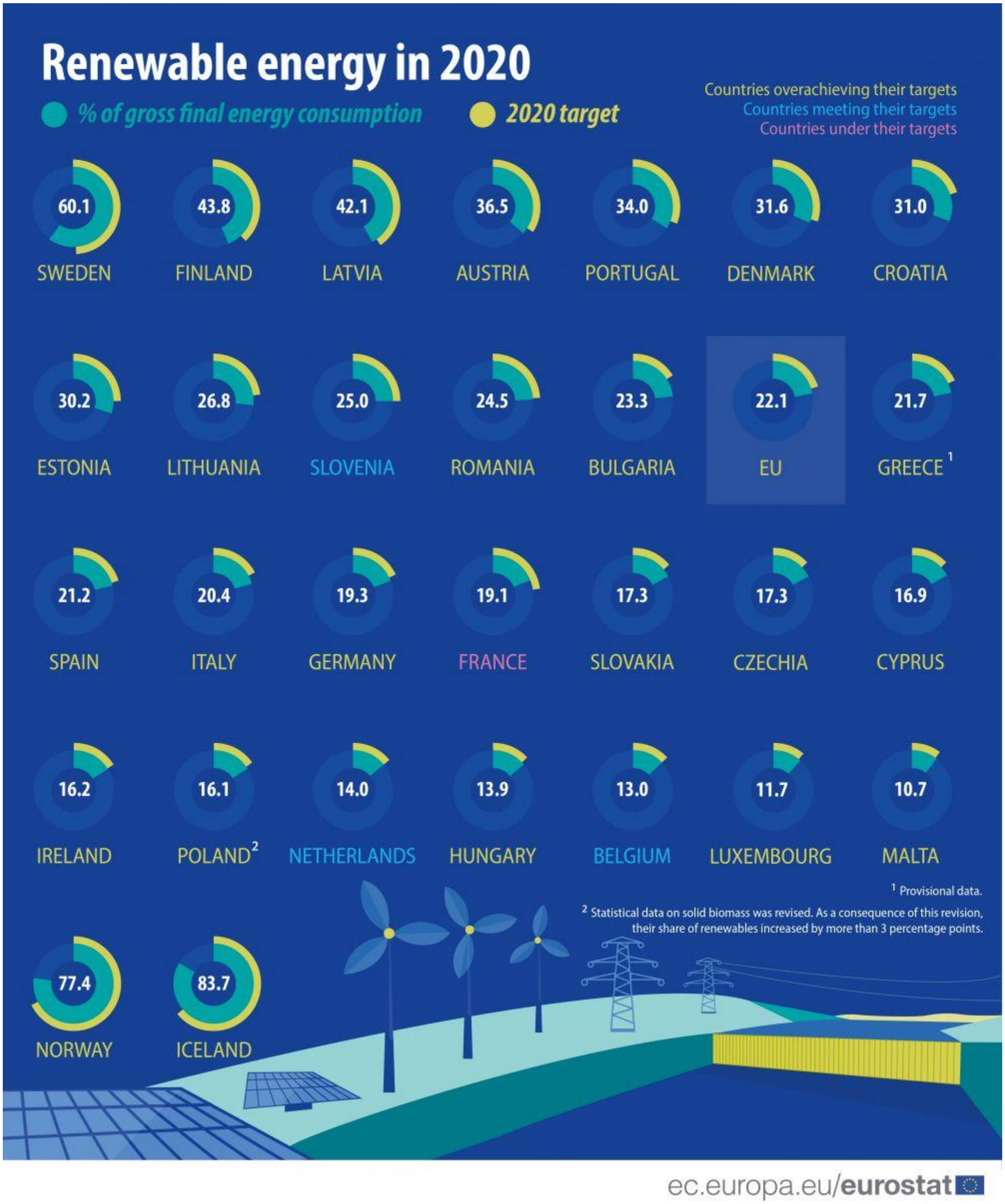


Figure 1. European renewable energy consumption 2020.

On the other hand, due to the low temperatures in the north of Europe, ice appears in the reservoirs in winter, what creates a new challenge to be faced, as it can change a lot of the designs used for floating solar panels in other parts of the world.

1.1. Floating solar panels

Floating photovoltaics (FPV) is the name given to a solar panels system attached on a structure that floats on a body of water, usually a reservoir or a lake. Since 2016, the market for this renewable energy technology has rapidly grown. Between 2007 and 2013, the first 20 plants with capacity of a few tens of kWp were developed. In 2020, the installed electricity reached 3 GW, with 10 GW expected by 2025.

Some of the most interesting benefits that this kind of systems provide are the following. First of all, the main advantage is that there is no need to occupy land, and even if the water area needs to be paid, usually, electricity producers already own it in the form of reservoirs to produce hydropower. The second beneficial point is that, because of having no fixed structure and be more compact than the land-based farms, the installation and decommissioning is easier, faster, cheaper, and more straightforward, making it a completely reversible installation. The third feature of this technology is that it saves water at the same time than increasing its quality. This happens due to the reduction on the evaporation cause by the partial coverage of the reservoirs. The fourth feature is that, because of the system being floating on water, its cooling is very easy, which, due to the reduction on thermal drift, can increase the power production by up to a 10%. Another feature is that it can help to control the algal blooms due to the reduction of the amount of sun reaching the surface.

On the other hand, this technology has also some disadvantages and challenges to be solved. The first one is that the cost to install one of these systems is higher than the one for normal land-based solar farms (being able to be up to a 25% higher). Another one is that due to being on water over its entire life, corrosion resistance must be higher than in a land-based farm, on top of having to be more careful with the electrical safety because of the close presence of water. The complexity on the maintenance of them is also a big challenge, as they cannot be accessed by walking as in a land farm or, if they are, the water around must be always considered as a risk. But, even with this challenges, there are already a couple of power stations with nominal powers over 150 MW, including one with 320 MW.

1.1.1. Solar panel installation components.

Solar panels are the main component (also called PV panels or PV modules), they are the ones that convert the sun energy into DC electricity.

The floaters are a series of plastic rafts that make the whole system stay floating.

The mooring system is the one in charge of keeping the whole PV grid in place, and it's formed by the mooring line and the anchor. The mooring line is the line that connects the floating grid with the anchor. Its design depends on different conditions such as the temperature to which it will be exposed, the weight of the floating grid and the forces that are expected on the grid. The anchor, as seen in [Figure 2], is the piece that makes sure that the grid doesn't move by providing support to the solar panels from the waterbed. This part is important because it must withstand all the forces transferred by the mooring line to keep the whole floating system on place, as the floating system moving could cause big problems, including damages to the system if it crashed against the dam or gets stuck on the shore and safety issues if it moves during the solar panels system servicing.

Some of the most common anchor types are the vertical load (VLA), the drag anchor, and the suction pile [Figure 2].

The electrical components include a combine box, used to merge the energy provided by all the solar panels to fed the inverter, the inverter, used to transform the DC electricity provided by the combine box into AC electricity, the transformer, to step up the voltage to avoid energy losses when transferring the electricity, and the cabling, that includes all the cables needed to transport the electricity between the previously mentioned systems as well as to transport it to land.

The last component that is needed in a solar panel grid is a walkway or a space to drive a boat (depending on the type of PV grid). It is very important, especially for servicing the floating systems.

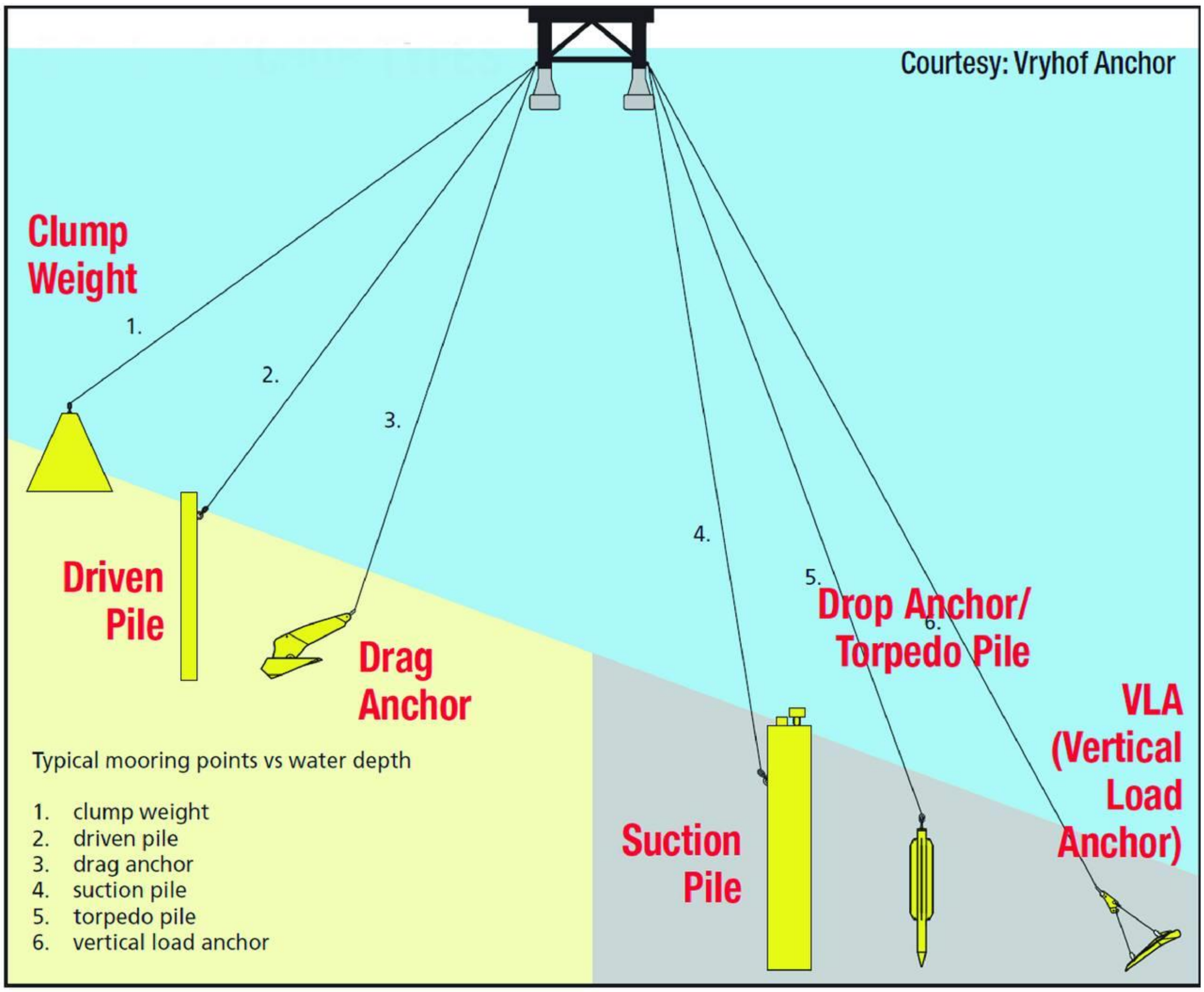


Figure 2. Anchor types.

1.1.2. Two main types of floating PV grids

The flat PV grid consists of a surface created by the floaters covering the water, on top of which the solar panels are laid, either flat or with a bit of inclination [Figure 3].

Figure 3. Flat PV grid. (nrgisland.com)

This layout is very good when there are no external actions that can affect the system. I.e., this system works well in lakes with no wind and no waves in which the solar panels can get sun energy without the need of tilting them, and there's no expectancy of snow or dirt to fall on top of them.

The system itself is cheaper than the other type of PV grid because no structure needs to be added, making the installation faster and cheaper (although not the floaters, that will be a more important percentage of the whole system). Furthermore, it is easier to let walkways in here, as well as work on servicing the solar panels.

Structure-based PV grids, on the other hand, are systems in which, in between the solar panels and the floaters there is a structure that separates the first ones from the water surface [Figure 4].

Figure 4. Structure-based PV grid. (houseofswitzerland.org)

This kind of structures have several advantages when it comes to dealing with external actions, as can be waves and ice, as they would be crashing against the floaters with no options to arrive to the sensible electrical systems, and snow or dirt as it has space to fall under the panels instead of piling up on top of them.

Another notable advantage of this system is the ability to instal tilting systems for the PV panels in an easier way, which can serve not only to follow the sun and maximize the efficiency of the solar panels all day long, but also to get rid of snow or dirt that accumulate on top of them by heating the surface while tilting the solar panel to an optimal angle (Yan, Qu, Chen, & Feng, August 2020) or by spraying water, that in a floating installation could be directly got from underneath, on top of the solar panels (Abdolzadeh & Ameri, January 2009).

1.2. Structure of the report

In this report, the feasibility of installing a PV grid covering 2 km^2 of a reservoir will be studied. For it, firstly, all the main challenges that will need to be faced are explained with details on [Section 4], and afterwards, based on some meteorological databases obtained from TrønderEnergi, some parameters including expected forces, stresses and displacements will be calculated in [Section 5], where it will also be explained how the data has been processed (code can be found in Appendix B – Code). To be able to understand where the calculations come from, and why some of the problems appear, in [Section 2] all the used theory has been explained. On [Section 3], a technical explanation of all the setup has been given, including both, information about the reservoir, the grid layout and its conditions, and information about some of the used systems, including mooring system, solar panels, and joints. All of it chosen to be able to meet the company's requirements. All the results are presented and their implications on the project discussed in [Section 6]. In [Section 7], all the assumptions used to propose the solutions for each of the challenges, which are also explained in that section, are exposed. Furthermore, some of the research that will be needed in order to get a final design for this PV grid has also been stated in the [Section 7]. Finally, a conclusion of the work done and the technical feasibility installing a floating solar panel grid in the reservoir in Falningsjøen is given in [Section 8].

2. Theory

In this part, firstly, the main concepts will be explained, followed by the equations that have been used to calculate the needed parameters.

2.1. Technical concepts

2.1.1. Ice growth

The first thing to understand why this analysis is necessary is that ice can form and alter the conditions of the reservoir where the system is intended to be installed. As found in (Høyland, 2021), the surface layer of water must reach the freezing point before ice can form and begin to expand, but since water has a larger heat capacity and density than air, it takes longer to heat and cool. In contrast to the winter, the water is generally cooler than the air in the summer. [Figure 5] depicts the cooling and freezing process in a typical late fall circumstance where the air temperature is far below freezing but the water temperature is still warmer. When surface water particles get heavier as they become colder, they sink and begin a convective mixing process. This will continue as long as the surface water particles cool down and grow heavier. Because the maximum density in fresh water is reached at around 4 ◦C, when the whole water column reaches this temperature, convective mixing ceases and ice forms on the surface. One major consequence of this process is that the water depth is a crucial parameter for ice development, and hence for the ice thickness. Ice will not develop if the water is deep enough, and the winter is short enough not to reach the aforementioned 4 degrees.

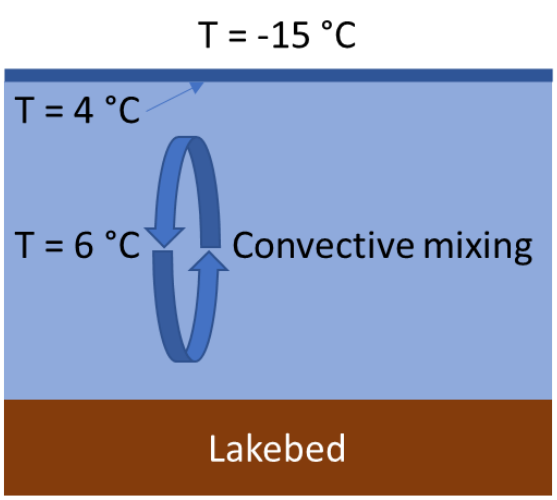


Figure 5. Typical situation of ice formation in fresh water.

When a thin ice cover forms, most of the ice development occurs at the bottom of the ice, and the latent heat generated during this process is transferred up through the ice and into the cold air. This is the primary ice development process unless there is significant precipitation or coastal impacts that cause significant flooding. Stefan's law, with or without correction factors, certain purely empirical formulae, or computational heat transfer equations can all be used to estimate ice formation and ice thickness. Another method that increases ice formation or ice thickness is rafting. It occurs more often when the ice is thin, and it explains recorded discontinuities in the ice cover across short horizontal distances, as described later in this section.

Ice, on the other hand, can be flooded, resulting in the formation of new ice on the previous ice cover's surface. Snowfall may collapse the ice, allowing water to rise on top of it. When rain or high air temperatures and radiation cause the snow surface to melt and trickle down through the snow cover, superimposed ice can also form. The water freezes when it enters a freezing zone. Fresh ice, which is often porous, is formed as a result of this process. It may cause ice or an icy coating to form on top of the snow.

2.1.2. Wind driving

The main concept that needs to be understood in order to understand this thesis is that the wind, due to its friction with the ice sheets, can cause a significant force, and push the ice strongly enough to make it move. This can cause the ice sheet to break, to crush against a vertical wall, that can be a dam, or another sheet of ice, to start climbing the shores, to raft, or even to ridge.

2.1.3. Rafting

As described in (Tuhkuri, 2014), rafting is the process of overriding a sea ice sheet with another. A break dividing a sea ice cover into two sheets and a compressive force moving these two sheets towards each other are required for rafting to occur. Rafting is divided into two types: simple rafting and finger rafting. As demonstrated in [Figure 6], simple rafting is a two-dimensional phenomenon in which one ice sheet overlays another, doubling the ice thickness. [Figure 6] also shows finger rafting, a type of rafting in which the two ice sheets overlap to form fingers. Finger rafting is a three-dimensional procedure that involves not only moving one ice sheet over another but also breaking the sheets to form the fingers.

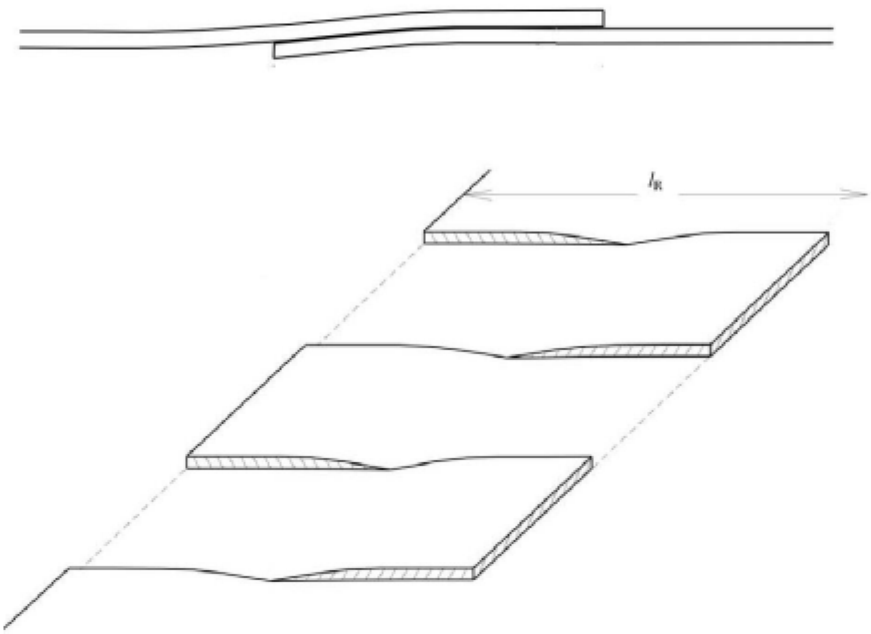


Figure 6. Finger rafting (above) and simple ice rafting (below). (Tuhkuri, 2014)

Rafting starts when one ice sheet slides over the other and continues for as long as there is enough pushing power to keep the action going, or until one of the ice sheets buckles or bends. The force that stops this motion is the friction force between the two ice sheets.

2.1.4. Ridging

(Tuhkuri, 2014) also explains that ice ridges are extended ice rubble piles that cross the ice pack in the same way that mountain ridges do in a hilly terrain (as in [Figure 7]). Ridges may be isolated, straight, and separated from each other by smooth level ice, or alternatively, they can be curved and form a field of ridges, making it difficult to determine where one ridge ends, and another begins.

Figure 7. Image of a ridge in the Moosehead Lake, USA. (iceboat.me)

A ridge's cross-sectional profile is seen in [Figure 8]. A ridge is usually divided into three sections. The big underwater component is the ridge keel, while the visible and above-water part is the ridge sail. A triangular keel and sail are the most common. The consolidated layer at the ridge's waterline is the ridge's third major component. The consolidated layer is mostly a solid block of ice, unlike the keel and sail, which are made up of separate ice chunks that are either frozen together or free.

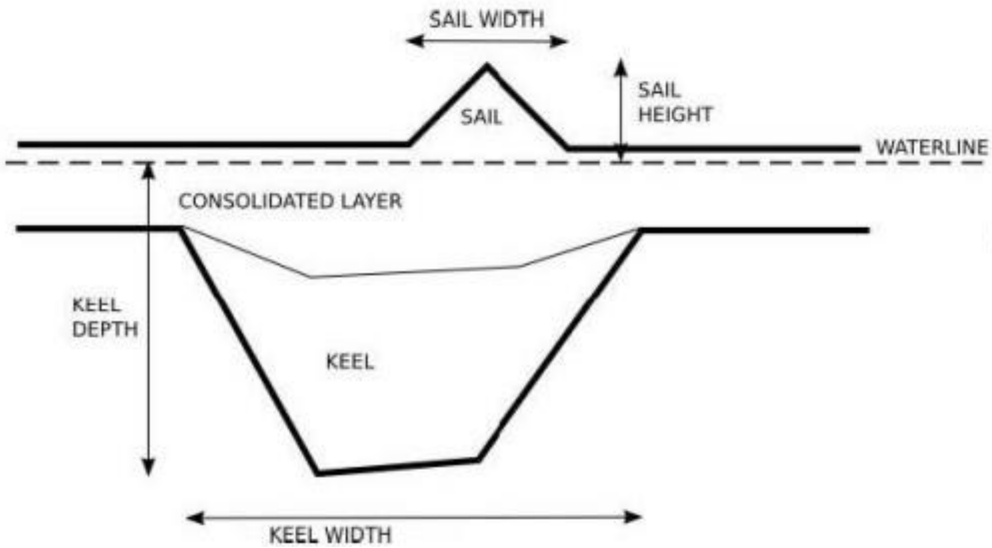


Figure 8. Simplified presentation of a cross section of an ice ridge. (Tuhkuri, 2014)

2.1.5. Thermal expansion

Thermal expansion happens when the temperature of a material changes, causing it to expand and grow larger. To comprehend how this occurs, what temperature is must be understood first. Temperature is defined as the average kinetic (or movement) energy of the molecules in a material. When the temperature rises, the molecules move, on average, quicker. When a substance is, then, heated, the molecules move quicker and take up more space as a result, tending to migrate into previously unoccupied places. As a result of this, the object's size increases.

This mechanism becomes especially important in the case of having dams, which act as a wall stopping the free expansion of the ice, and because of that, transforming all the expansion in a stress on them.

2.2. Equations

2.2.1. Force needed to create a ridge

To calculate the force that needs to be applied to the ice to make it break and start ridging, the following equation has been used, as recommended by the (ISO19906, 2010b):

$$F_{RidgeBuilding} = A \cdot h_i^{1.25} \cdot l_r^{0.46} \qquad \textbf{[1]}$$

To get this force (in MN), the width of the expected ridge, l_r, is given in metres, the thickness of the level ice, h_i, also given in metres, and the parameter *A*, that is an empirical factor that can be from 2 to 10, although from past data, it is usually closer to 2 than 10. In this case and because of the impossibility to get this empirical factor, it is assumed to be 4.

2.2.2. Driving force

To calculate the force that the ice can apply because of drifting, the general formula is taken and modified based on some assumptions.

$$F_{driving} = F_{air} + F_{current} \qquad \textbf{[2]}$$

As there is no registered data on current, some assumptions are done. Because of it being a reservoir, very low current is expected, so it will be assumed to be 0, thus, only considering the wind force.

$$F_{driving} = F_{air} = \rho_a \cdot C_a \cdot A_i \cdot u_a^2 \qquad \textbf{[3]}$$

In this equation, ρ_a is the density of the air, A_i is the area covered by ice that the air has contact with, u_a is the velocity of the wind and C_a is the drag coefficient between air and ice, which is also an empirical value.

2.2.3. Ice thickness for the ice to ridge

By combining Equation [1] and Equation [3], an expression to get the thickness that the ice should have for the wind to be able to break it and make it ridge is obtained.

$$
\begin{aligned}
&F_{RidgeBuilding} = F_{air} \\
&A \cdot h_i^{1.25} \cdot l_r^{0.46} = \rho_a \cdot C_a \cdot A_i \cdot u_a^2 \\
&h_i^{1.25} = \frac{\rho_a \cdot C_a \cdot A_i \cdot u_a^2}{A \cdot l_r^{0.46}} \\
&h_i = \sqrt[1.25]{\frac{\rho_a \cdot C_a \cdot A_i \cdot u_a^2}{A \cdot l_r^{0.46}}}
\end{aligned}
\qquad [4]
$$

2.2.4. Thermal expansion

The thermal expansion of the ice is used to calculate the maximum displacement of the joints when the ice expands and the maximum stresses on the same joints when the ice contracts. To reach these values, the first equation to consider is the definition of thermal expansion:

$$\varepsilon^T = \alpha * \Delta T \qquad [5]$$

Where ε^T is the strain, ΔT is the increase of temperature in degrees Celsius and α is the coefficient of thermal expansion of ice in 1/◦C.

The thermal expansion coefficient for pure ice can be expressed by (Michel & Drouin, 1971) as:

$$\alpha = (54 + 0.18T) \cdot 10^{-6} \qquad [6]$$

After knowing the strain, and to get the stress, it is assumed that the walls are fixed, what means that there cannot be any expansion. By applying this boundary conditions to the Equation [5], by adding the elasticity modulus, it is got that the strain becomes stress as seen in the following equation:

$$\sigma = \alpha \cdot \Delta T \cdot E \qquad \textbf{[7]}$$

Continuing from Equation [5] and assuming this time that there is free expansion, or in other words, that the ice can expand and nothing will stop it, the increase in length can be obtained. With this boundary condition present, and by multiplying the strain by the length of the ice sheet, and assuming that the ice will behave as if it was a beam, the increase in length can be obtained:

$$\Delta L = \alpha \cdot \Delta T \cdot L \qquad \textbf{[8]}$$

3. Technical information

3.1. Reservoir

The reservoir that has been used for this study is the one located in the lake Falningsjøen, located in the region of Innlandet, in Norway, which approximate coordinates are 62°36’29.1” N, 10°24’51.0” E ([Figure 9]).

The lake has an average area of 2.43km2 and a water level variation over the year that can reach 50m.

Due to the location of this lake, it completely freezes every winter, reaching ice thicknesses of 100 cm (maximum value recorded), and that represent a problem for floating structures that would need to operate all year long and will not be able to be removed.

Out of those 2.43 km^2, a bit more than 2 km^2 are wanted to be covered with solar panels by the interested company ([Figure 10]), what theoretically, considering a density of solar panels of 1MW per ha and a yield of 1000 kWh/kWp would generate 200 GWh annually, or the equivalent of the energy consumed by approximately 28.082 people in one year (assuming the average net electricity consumption per capita in the residential sector in Norway in the year 2020 (Statista, 2022)).

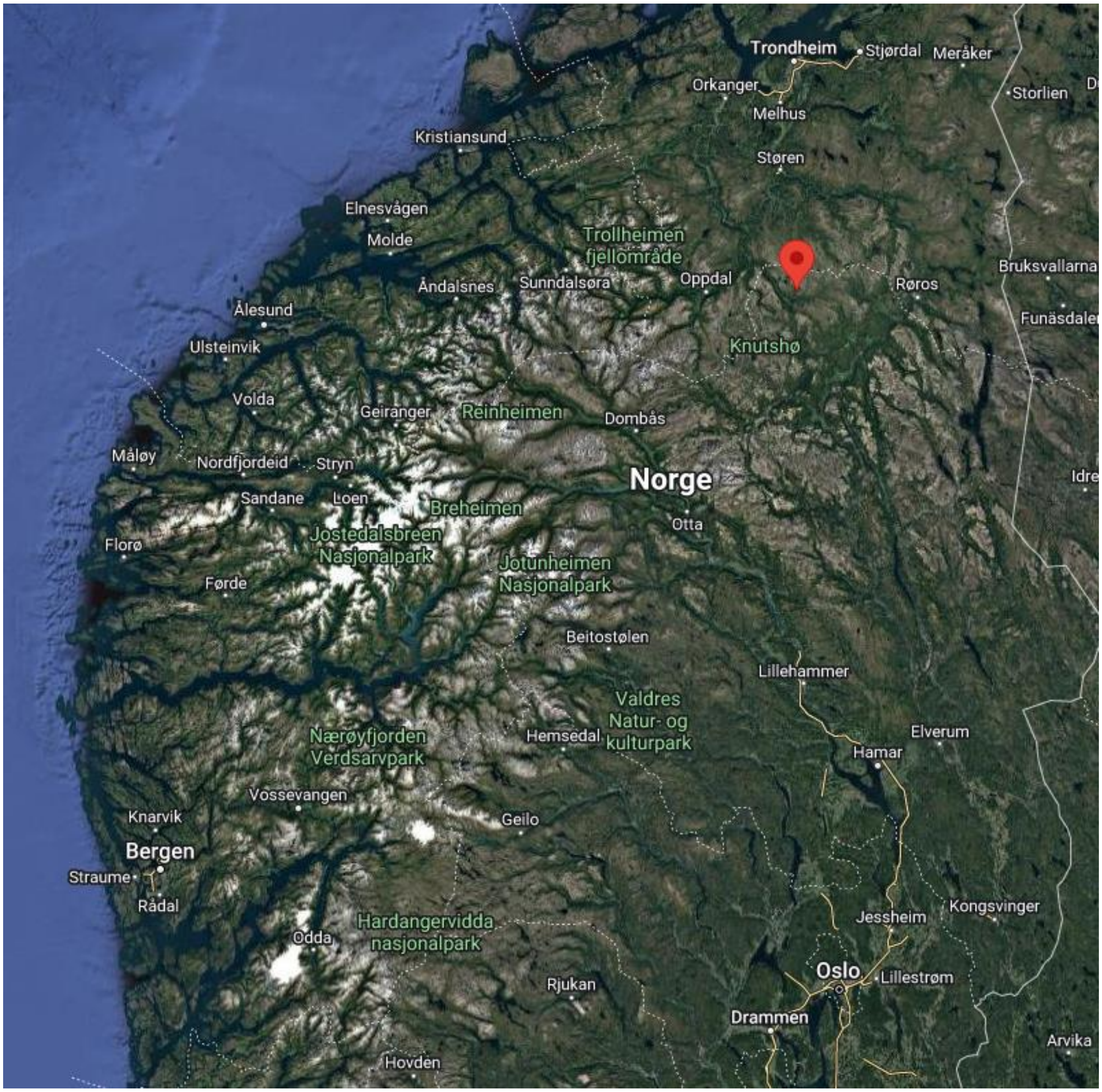


Figure 9. Position of the Falningsjøen reservoir inside of Norway marked by a red point.

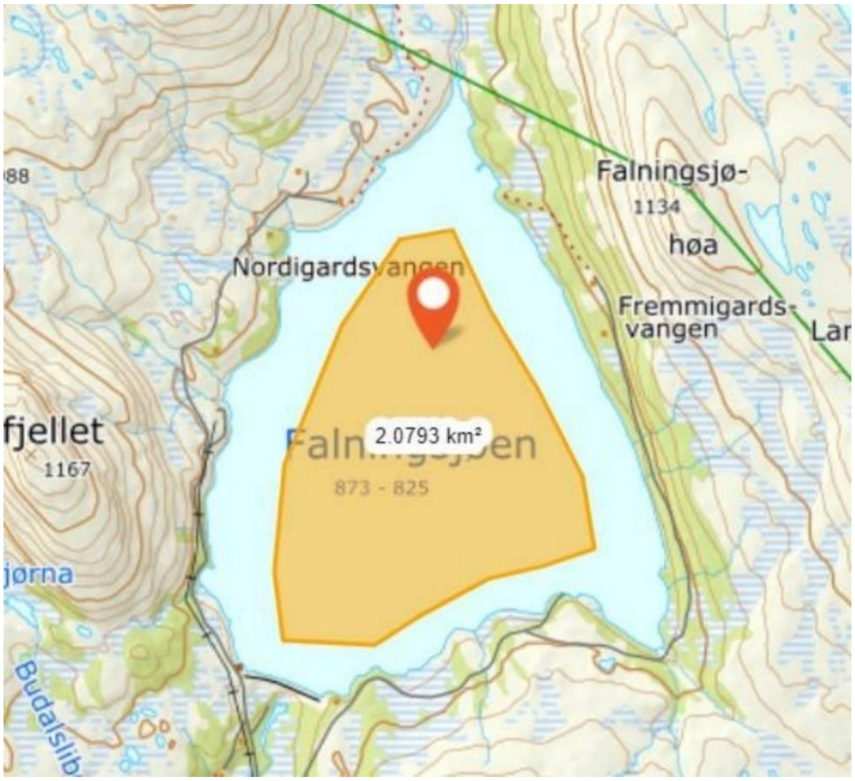


Figure 10. Shape of the reservoir and coverage requested by TrønderEnergi.

3.2. Block and grid units layout

To fill as much area as possible, a series of space optimization iterations have been done. After all several iterations, the result of it has been the layout shown on [Figure 11]. With a total of 284 blocks, covering an area of about 2.1 km2, obtaining the coverage asked by the company.

Each block consisting of 36 grid units, of which 35 are photovoltaic units and 1 of them is a transformer unit that receives the electricity produced by the 35 photovoltaic units and sends it to the main electric transformer on the hydropower plant in the dam to send it to the national electric grid. Block configuration shown in [Figure 12].

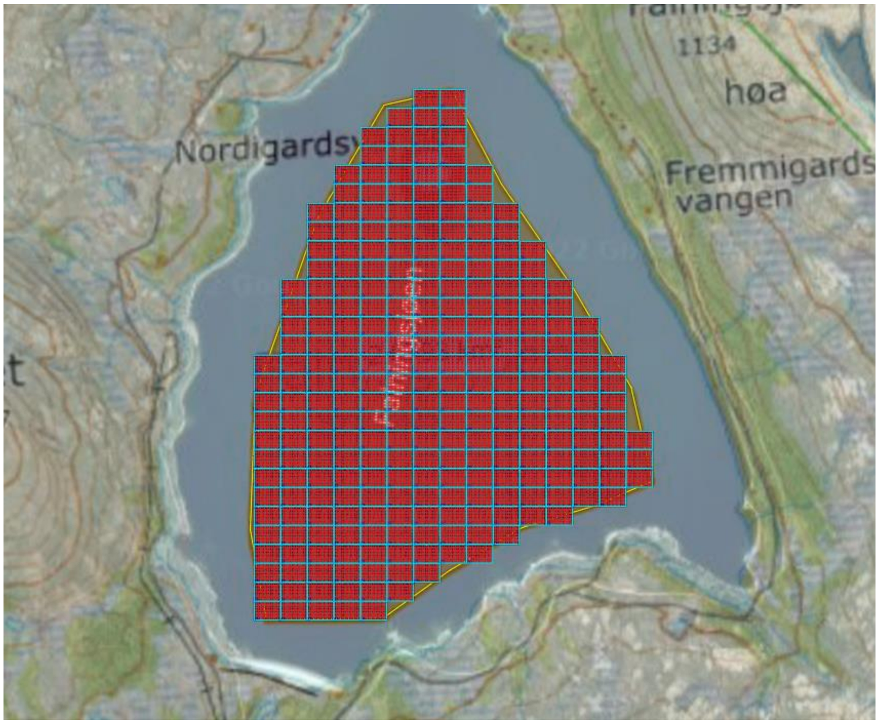


Figure 11. Final layout of solar panel blocks.

The blocks are composed by 6 rows with 6 units each, with a separation between each of them of 2.5 m as can be observed on the [Figure 12]. These grid units are attached to the next one by 2 joints with 2 full degrees of freedom and one partially restricted degree of freedom. Each photovoltaic unit consists of 40 solar panels arranged in 5 rows with 8 solar panels in each row configuration as can be seen in [Figure 13]. This layout allows to have a total of 9940 photovoltaic units with 284 transformer units, with a total of 397.600 solar panels.

It is also important to notice that there is no mechanical difference between blocks and units, and block is only used as a term to describe all the units that are connected to the same electrical unit, where the electricity produced in the photovoltaic units is converted and transferred.

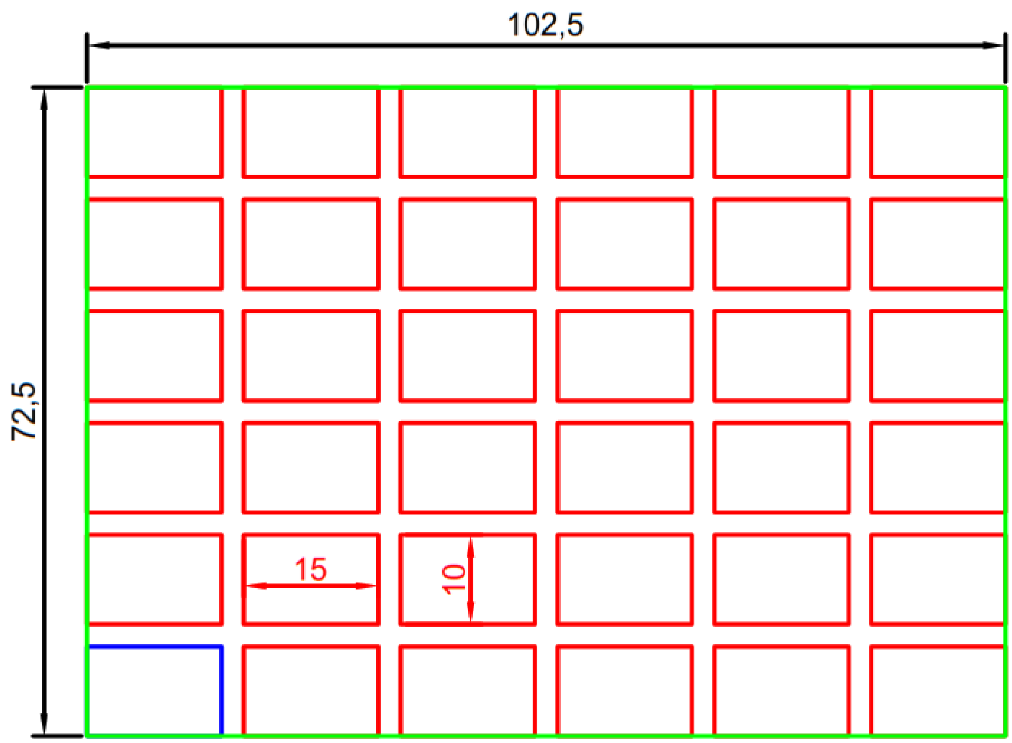


Figure 12. Block layout. In red photovoltaic units, in blue transformer unit. Measures in metres.

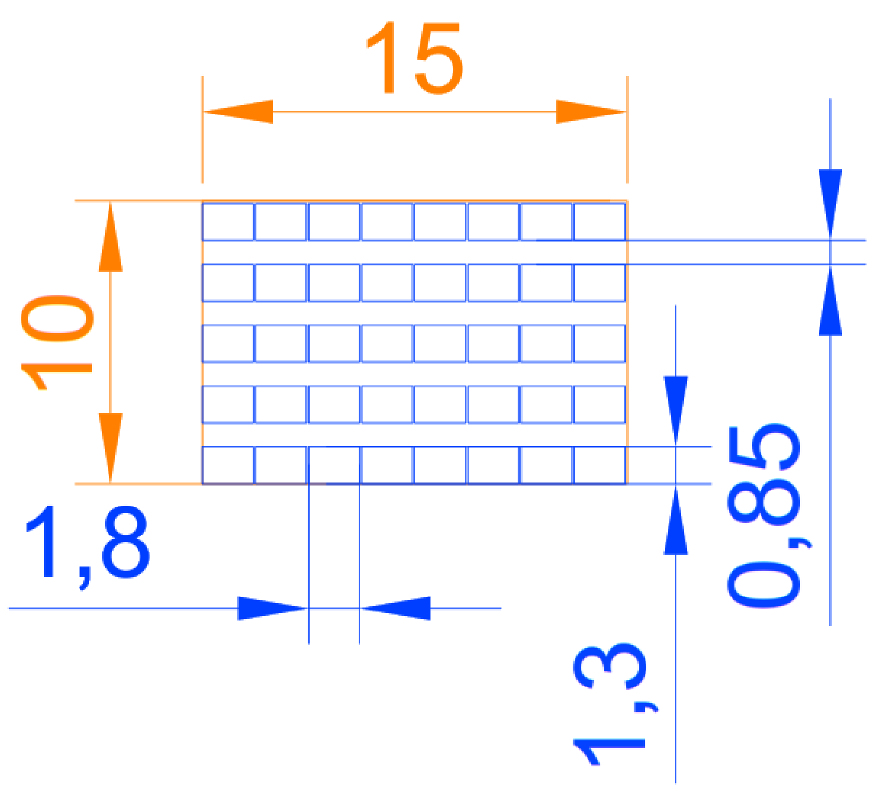


Figure 13. Photovoltaic unit layout. Each blue bloc represents one solar panel. Measures in metres.

3.3. Photovoltaic panels arrangement.

The measures of the solar panels as well as the geometrical arrangement are shown in the [Figure 14]. At the same time, each row is separated to the next one by 0.85 m to allow for enough space when servicing them.

The solar panels are mounted on top of a metal structure that raises them 1 metre above the floaters as can be seen in [Figure 15], and this metal structure is at the same time mounted on top of the floaters, which have a special geometry, such as the one found on the hulls of the ships that navigate in the Arctic designed specifically to support these lateral pressures and move the ship on top of the ice to break it with its weight when falling down. An example of this is the Arktika seen in [Figure 16]. This geometry will allow the floaters to stay on top of the ice thanks to the lateral pressure applied by it (further explanation in Section 7.2).

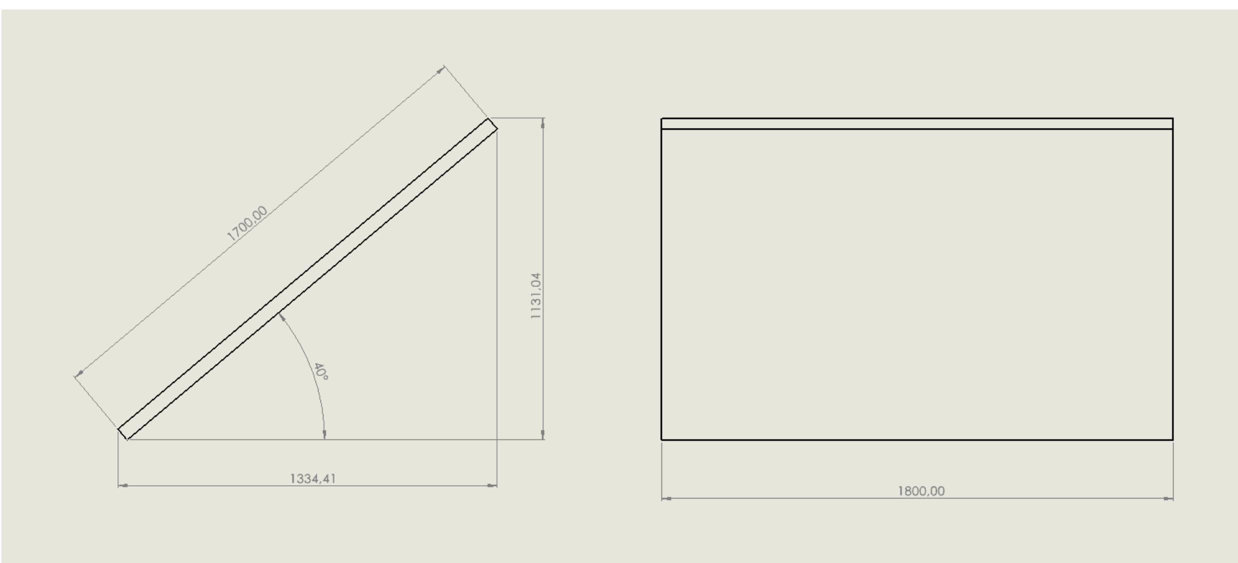


Figure 14. Geometry and measures of the solar panels (in millimetres).

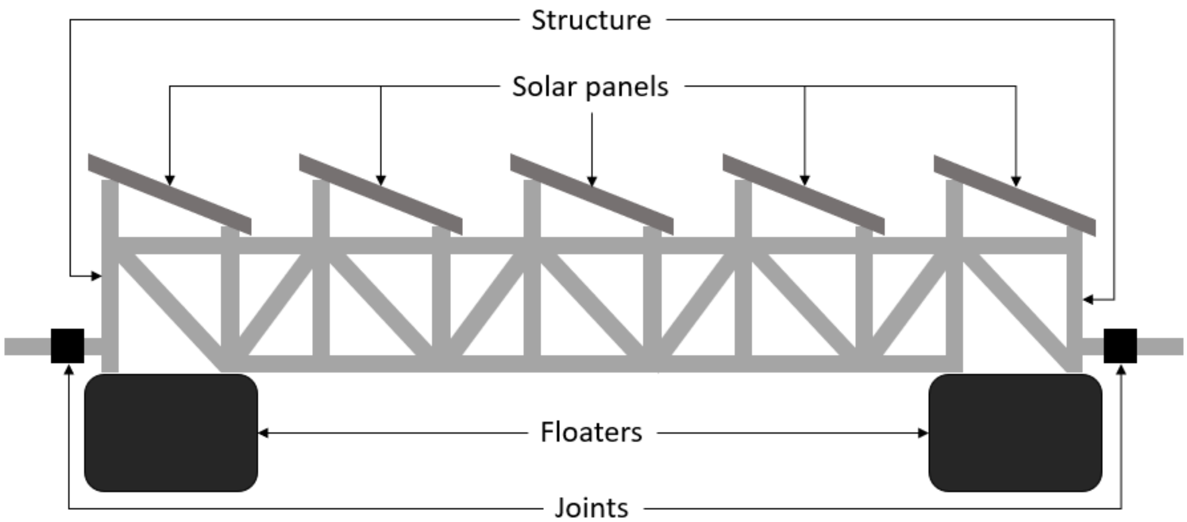


Figure 15. Side view of the initial idea of one PV unit.

Figure 16. Arktika icebreaker. (Sitdikov, 2016)

These solar panels are double sided, what means they can produce electricity on both sides, helping to solve more problems that appear in winter (see Section 7.3 for further explanation). With an assumed power production of 0.2 kW per solar panel, the estimated yearly production comes to 232.2 GWh, as shown in [

Table 1], reaching the company expectations.

Solar panels	397600	units
Power per solar panel	0,2	kW
System power	79520	kW
Average sun hours per day	8	h
Average daily production	636,2	MWh
Yearly production	232,2	GWh

Table 1. Electricity production estimation.

3.4. Mooring system

To attach the whole floating system to the bottom of the reservoir, a mooring system consisting of several mooring chain lines is used. As seen in [Figure 19], each block side is attached either to another block or to the lakebed. As each grid unit ([Figure 13]) has 2 attachment points per side, as can be seen in [Figure 17], they are mounted on the structure that will stay on top of the ice, and the free sides (the ones without another unit's side next to them) will have a retractable mooring system (as in the block unit shown in [

Figure 18]).

Each mooring point consist of 2 mooring lines attached to the anchors, and each block ([Figure 12]) has 6 mooring points per side (one per grid unit, as the two attachment points merge into one mooring point [Figure 18]). As can be seen in [Figure 19], the total block sides that need mooring is 30 in the vertical direction and 56 in the horizontal direction, what adds up a total of 86 sides, 516 mooring points and 1032 mooring lines.

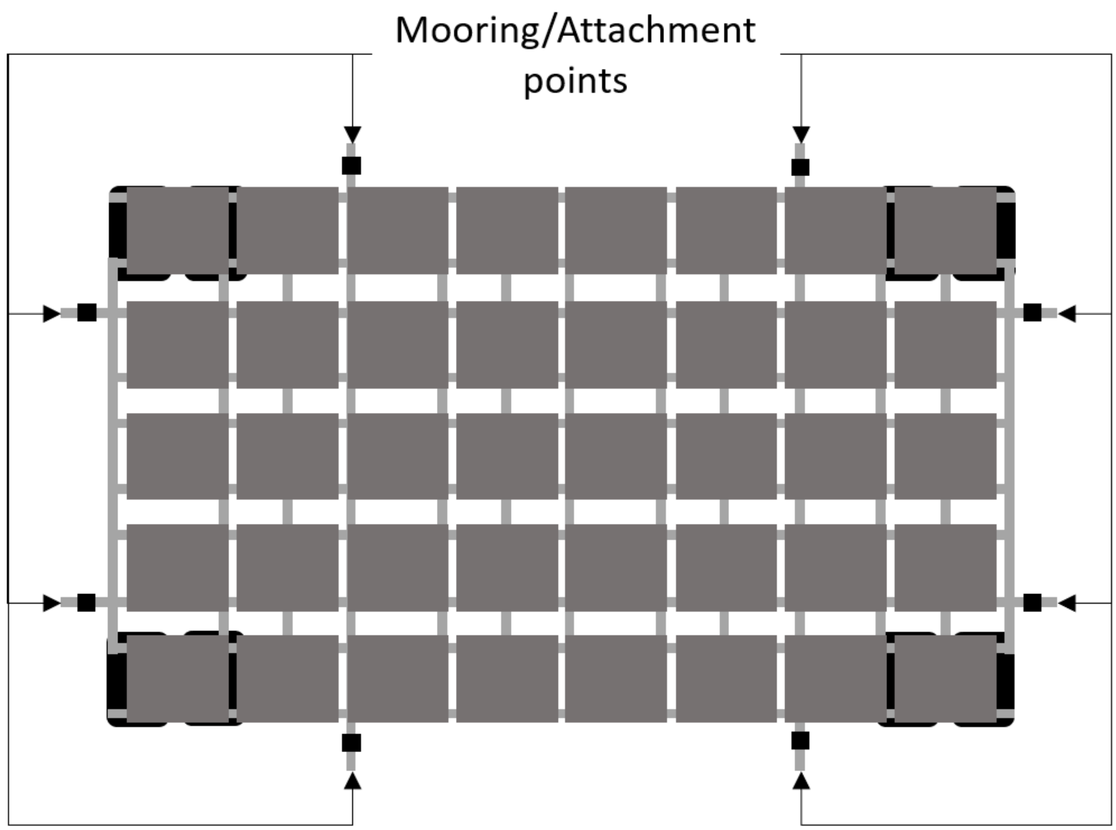


Figure 17. PV unit and location of the attachment/mooring points.

Figure 18. Example of existing block unit with mooring lines. (Romande Energie, 2021)

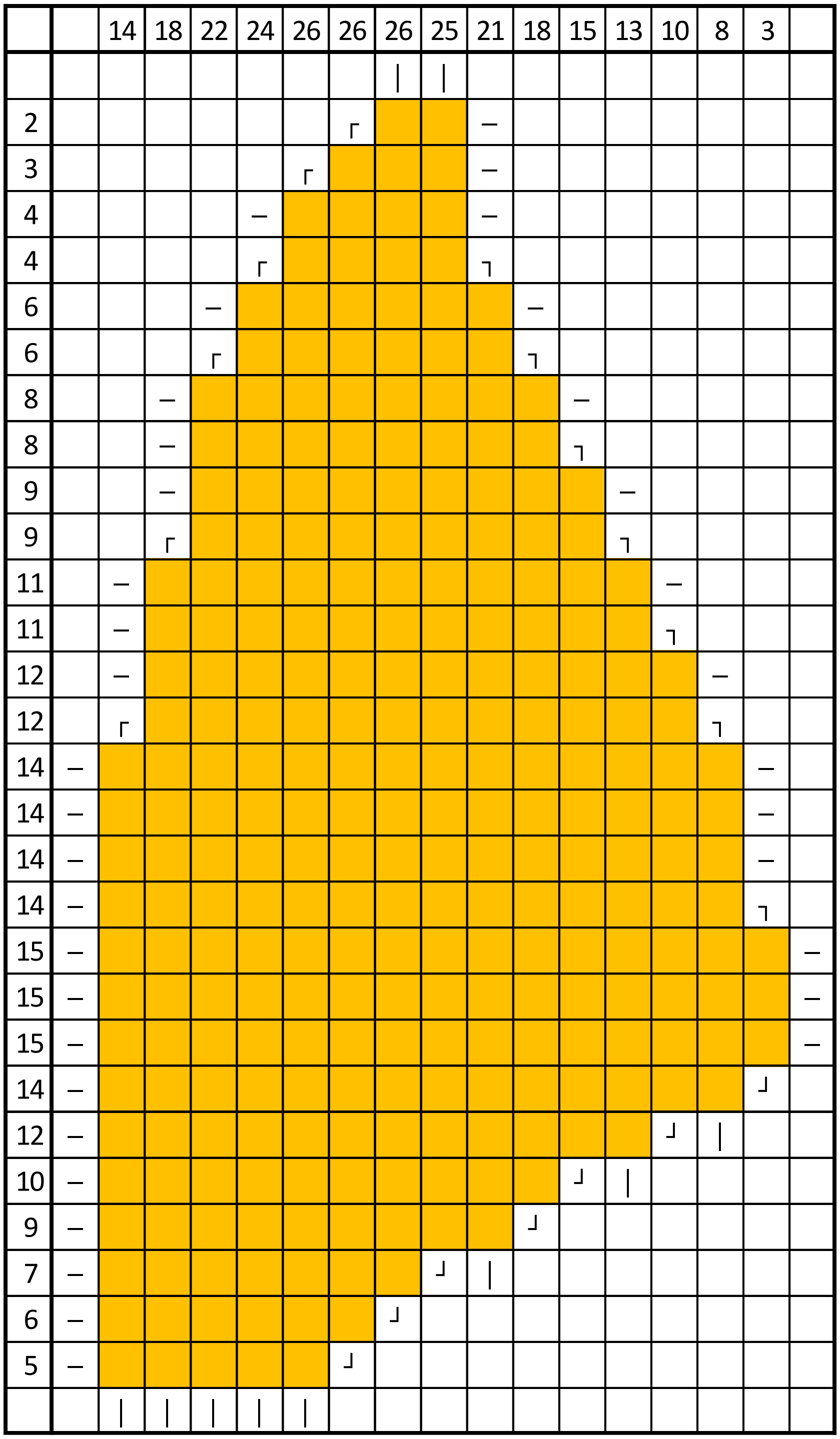


Figure 19. Mooring scheme. Yellow blocks represent blocks. Lines represent a blook mooring, which is 6 mooring points and 12 mooring lines. Numbers account for the total of blocks in each row and column.

4. Main challenges

4.1. Change in the ice conditions

Due to the solar panel system covering a big part of the reservoir, the ice conditions will be modified. It is very challenging to understand how this mechanism will work, as at the same time that it blocks the cold wind (resulting in warmer water), it also blocks the sun arriving to the water surface (resulting in colder water), effects that counteract. This is one of the fields in which more in-depth research needs to be done, as the way in which this will affect the ice is not known. At the same time, the snow conditions could also affect the ice in a different way.

4.2. Ice drag force leading to ridge formation

Due to the nature of the freezing situation accruing in winter in the reservoir, it is very important to check if the air drag on the ice can break the solid sheet of it and create a ridge, as it would greatly increase the forces that the floating system could potentially face.

4.3. Thermal expansion of the ice

The thermal expansion is very important, as it has a great effect on the displacement that the joints must withstand, as well on how much the mooring system will be pulled to the sides. It depends on the maximum distance between the extremes of the system and the number of joints in each direction.

4.4. Joints system

To allow for enough displacement to withstand the thermal expansion, proper joints need to be chosen. They must allow enough movement to keep the integrity of the whole structure both when floating on the water and when resting on top of the ice.

4.5. Mooring system

The mooring system is one of the main structural elements of the system. It must allow enough movement not to create too much stress on the joints while restraining the movement of the system.

This system must work differently when the structure is floating than when it is resting on top of ice.

4.6. Increased ice drag

As the mooring will allow free movement while the system is resting on top of the ice, all the drag loads will be transferred to the ice, thus, greatly increasing the loads that the ice will apply on the existing dam.

5. Calculations

In this section, 4 extreme cases have been calculated. The first case is when the wind blows with the maximum recorded speed. This wind will create a drift force that will push the ice into the solar panel system, and if it is strong enough, it can break and start applying forces to the metal structure. The second calculation done is regarding the thermal expansion. Here it is calculated the maximum expansion that could appear in each joint and the maximum stress (following the free expansion and limited expansion scenarios respectively). The third case is the drag force that can appear due to the wind on the solar panels, that is the force that will be transferred to the mooring system, and finally, the last case is the joint drag force of the solar panel system and the ice surrounding it that can be pushed against the dam.

5.1. Data

To be able to perform all the previously mentioned calculations, some data is required. As seen in [Table 2] the water temperature and the current are not available, what limits the precision of thermal expansion and drag calculations. The snow thickness data, even if it exists, has a very poor quality, what makes it unusable for thermal calculations. On the other hand, the air temperature and wind speed are in the databases with hourly values and a very high quality of data, what allows to get some calculations done after following some assumptions. The ice thickness, even if important for the calculations, is monthly registered and the quality is not good. Due to this, the existing dataset has needed some modifications and estimations to be used.

	Have it?	Obtention	Periodicity	Quality
Ice thickness	Yes	Manual	Monthly	Poor
Snow thickness	Yes	Manual	Monthly	Very Poor
Air temperature	Yes	Automatic	Hourly	Good
Wind speed	Yes	Automatic	Hourly	Good
Water temperature	No	-	-	-
Current	No	-	-	-
Water level	Yes	Automatic	Hourly	Good

Table 2. Input data, obtention method, periodicity, and quality of it.

The first step to take to be able to proceed to the calculations is to process the data. In this case, databases with hourly information for different parameters (found on Appendix A – Databases parameters) of the conditions in the reservoir were given by the company. One database was given for each year from 1999 to 2021.

On the other hand, some other parameters about the ice, snow, slush, etc… were obtained from iskart.no. This data was registered on the website by the company after doing in-situ measurements, and it was put on databases individual databases for each parameter for the whole period of time.

5.2. Program

In order to be able to process all the data in an organized way, a small python program that launches all the different scripts was created. As can be seen in [Figure 20], this program will show some options to run all the different scripts and go back to the menu after finishing running them. It is important to know that the scripts are made to be run in the given order to avoid possible errors, although it is possible to change the order of the calculation scripts. The compiler must always be run first in case it has never been run and can be omitted if it has been run on a previous occasion.

5.2.1. Compiling script

The first script works as a compiler and allows the user to choose which period to study, although in this case the whole available period (1999 to 2021) was checked. In other words, it puts all the data from the selected period in one file that can be used by the rest of the scripts. As can be seen in [Figure 22], to be able to use the script, the databases must be stored in a subfolder called “Data” in the same location as the python scripts. The script will ask then for the starting and ending year to compile the databases and check if the answers make sense (both being numbers, the initial year being smaller than the ending year, and both being in the year range given, being the minimum number of years compiled 2). If the input data is correct, the script will open the first database and copy the data from it. To be able to have a usable database for the next steps, the introduction information, including location and information about how the data is obtained, is deleted, leaving only the column names and the data. The next step is opening the following year databases one by one, copying only the data (without the parameters names) and add it to the end of the previously copied data in a new file called “temporary.txt” that will be the one utilized by the other scripts.

5.2.2. Ridge calculation script

The Ridge script is the first one to calculate forces. It is important to note that all of the calculation scripts (Ridge.py, Thermal.py and Drag.py) contain a function to export the DataFrame into csv files that can be read in other software or imported to other scripts in order to further process them without the need to run all the scripts again. In order to be able to operate with databases, the “Pandas” module needs to be loaded. Built on top of the Python programming language, pandas is a quick, powerful, versatile, and simple open source data analysis and manipulation tool. This module will be used on every script that involves manipulating data.

As can be seen in the [Figure 23], the script starts by opening the file previously compiled, from which it will extract the wind speed at 10 metres, the closest to the surface available, and store it in a DataFrame. As the wind speed data comes on an hourly basis, but the ice thickness data available only comes monthly, the next step is to check for the maximum monthly value for wind speed, as it will be the one giving the worst-case scenario. The periodicity has been chosen based on the longest periodicity of the data that needs to be used, in this case the ice thickness. Then, the ice database is loaded and joined to the wind speed in a new DataFrame called *month*, but as there is a lot of data missing for the ice thickness values, the script will estimate it. To do that, the first step is to remove the months that are known to have no ice on the lake thanks to in-site observations, which are from June to November. Then, as the worst-case scenario is always the most limiting one, out of all the data for each month, the maximum thickness is checked and used to fill the months without data.

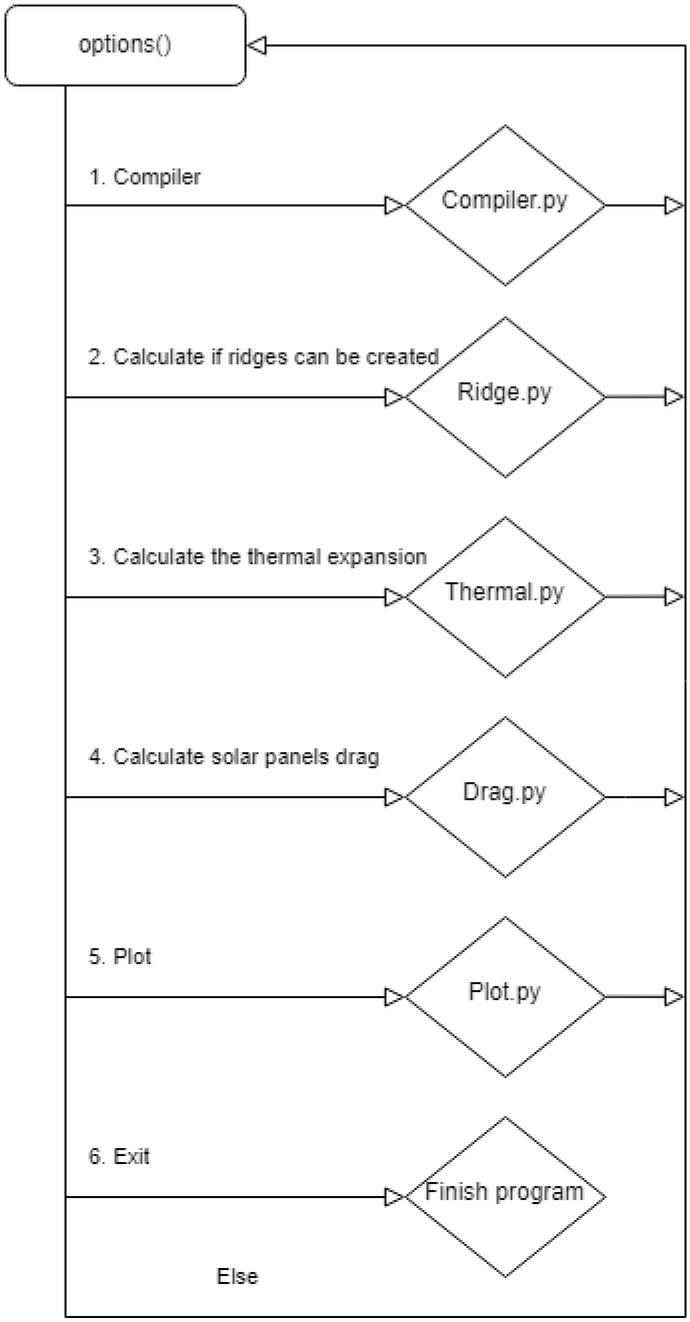


Figure 20. Flowchart of the main program. Showing selectable options and the script they open.

There is one exception in May, as it is known to have ice due to in-site observations, but the thickness is unknown, thus not having data for it. The thickness of it has been estimated with cubic regression using the data of the previous months and assuming 0 thickness in June (observations), getting a cubic polynomial trendline ([Figure 21]) that fits the given data and gives an estimate for May thickness of 53cm.

Once the ice thickness column in the DataFrame is complete, the script precedes to calculate the maximum force that the friction with the air can produce. For it, Equation [3] is used with the following parameters:

The used air density is 1.367 kg/m^3, which is the density at 1 atmosphere and -15 °C, the area is set to 1500000 m^2, which is the maximum area of ice that is open and can be pushed against the solar panel system. The drag coefficient has been chosen as 0.002, which will give the worst-case scenario result while staying in a relatively smooth ice surface scenario as is the case for a reservoir. The wind speed is taken from the DataFrame to have the maximum possible force for each month.

This calculation part also includes measures to prevent adding a force when there is no ice. After having the air driving force, it is included in the main DataFrame as a new column called *Fa*. The next step is to calculate the required force to build a ridge, which gives an estimate of the force needed to break the ice and make it start applying forces on the solar panels structures that is resting on top of the ice. This calculation part also includes measures to prevent adding a force when there is no ice. After having the air driving force, it is included in the main DataFrame as a new column called *Fa*.

The next step is to calculate the required force to build a ridge, which gives an estimate of the force needed to break the ice and make it start applying forces on the solar panels structures that is resting on top of the ice. To be able to calculate this force, Equation [1] is used with the following parameters:

The width of the expected ridge is 1000 m, parameter that comes restricted by the layout of the solar panel system ([Figure 11]). A, which is a completely empirical factor that varies between 2 and 10 depending on the situation, but is usually closer to 2 than to 10, is set to 4, although there are no reasons behind this assumption. The ice thickness is taken from the DataFrame, and the result is stored in a new column named *Fr* in the same DataFrame.

After having both forces calculated, the script runs a check test to compare the columns *Fa* and *Fr*, and creates a new column called *Ridge* that contains either "YES" or "NO" depending on if the force the air can apply is greater than the one needed to form a ridge or not respectively. Based on that new column, the maximum force that can appear is decided.

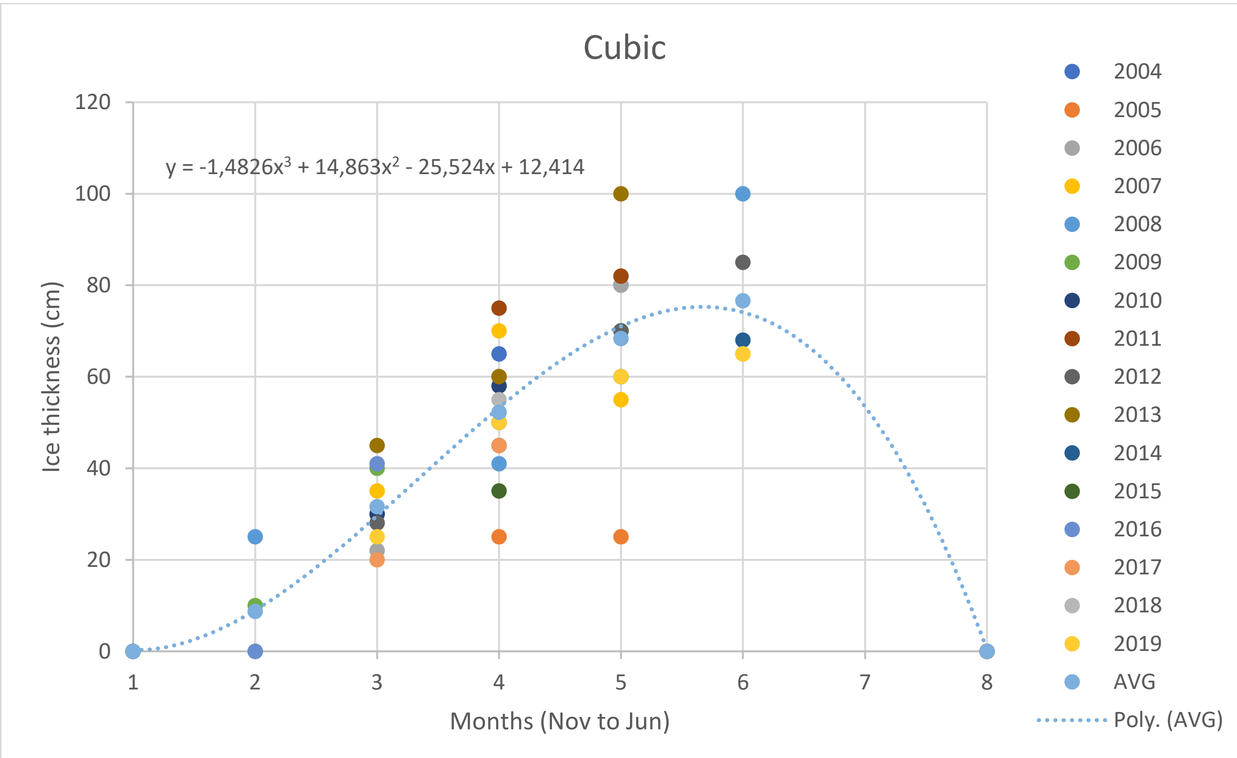


Figure 21. Ice thickness and cubic trendline.

Finally, the script calculates the thickness that the ice should have for the wind to be able to break it and make it ridge by using Equation [4] using the data from the *Fr* and *Fa* columns.

5.2.3. Thermal expansion script

The ice thermal expansion calculations are done in the Thermal script, that follows the [Figure 24]. The first step on this script is to obtain the hourly temperatures from the temporary databased created by the compiler script. After having them saved in a DataFrame called *temp*, the daily maximums and minimums are extracted into the DataFrame *daymaxt*, that contains the date, the maximum temperature of that day and the minimum temperature. This data is then compared with the ice data, and the months in which there is no ice are removed from the list. The script proceeds then to calculate the increase in length in both directions, x and y, as the maximum displacement in each joint is different in each direction, and store that in a new column. To get this value, Equation [8] is used, and with the total expansion divided by the number of joints, a new column for each direction is filled. Due to the system setup used, each unit is attached to the next one using 2 silent blocks, thus having 310 joints (26 blocks, with 5 units each, which means 10 joints per block plus 2 joints in between each block) in the length direction y and 178 joints (15 blocks, with 5 units each, which means 10 joints per block plus 2 joints in between each block) in the width direction x. From here, the displacements are converted into equivalent stresses, creating another two columns. Finally, the maximum expansion of the whole system, the maximum expansion of each joint and the equivalent maximum stress in both directions, length and width, are printed and stored in variables to be checked afterwards.

Some assumptions are made in this process. The first one, and the most important one, is that the change of air temperature is the same as the change in the ice temperature, what it known to be incorrect, but will give a high maximum that will never be reached, thus making the estimate safer. Another assumption is that the maximum stress that will appear in the joints, will appear because there is no elasticity at all, thus the silent block not being there. A third assumption is that the expansion will be free, and the joints will show no resistance to it. Both of them known to be incorrect due to the way the silent blocks that are proposed in the setup for each joint work.

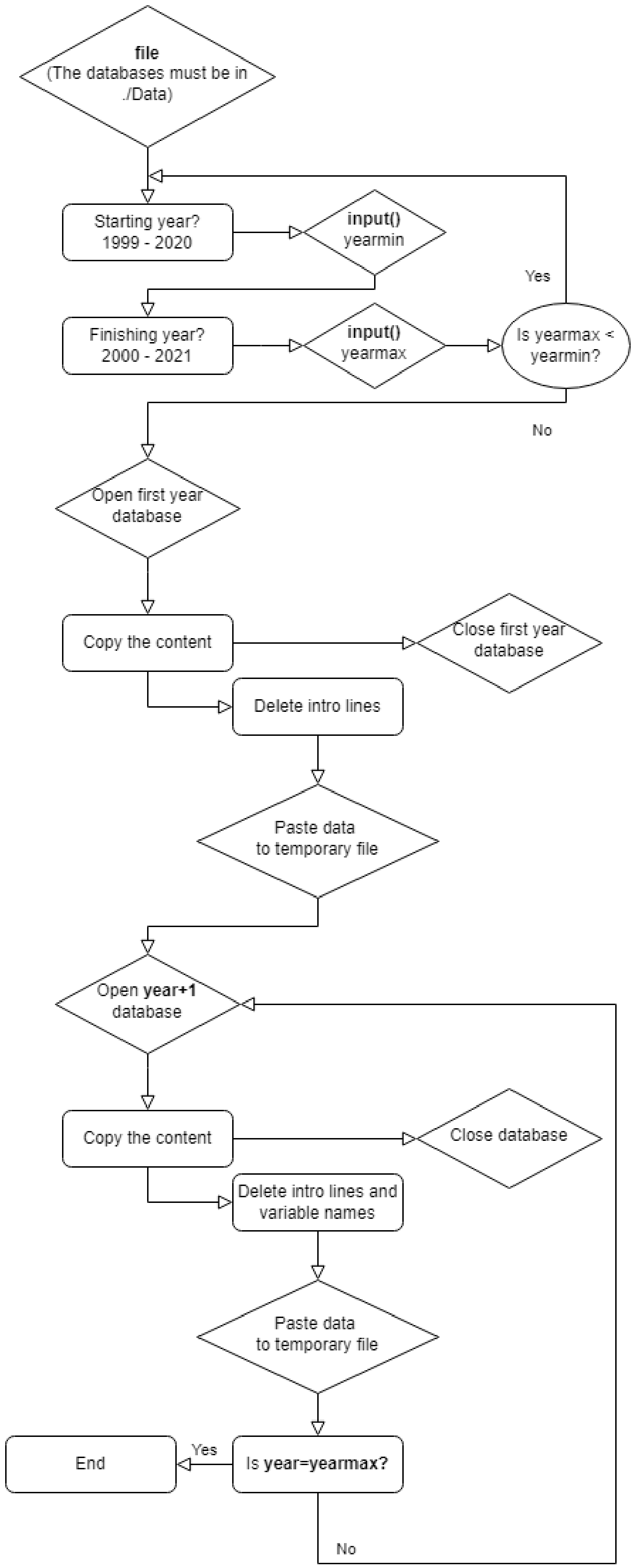


Figure 22. Flowchart of the compiler script. It allows to make a common database of the years to be studied.

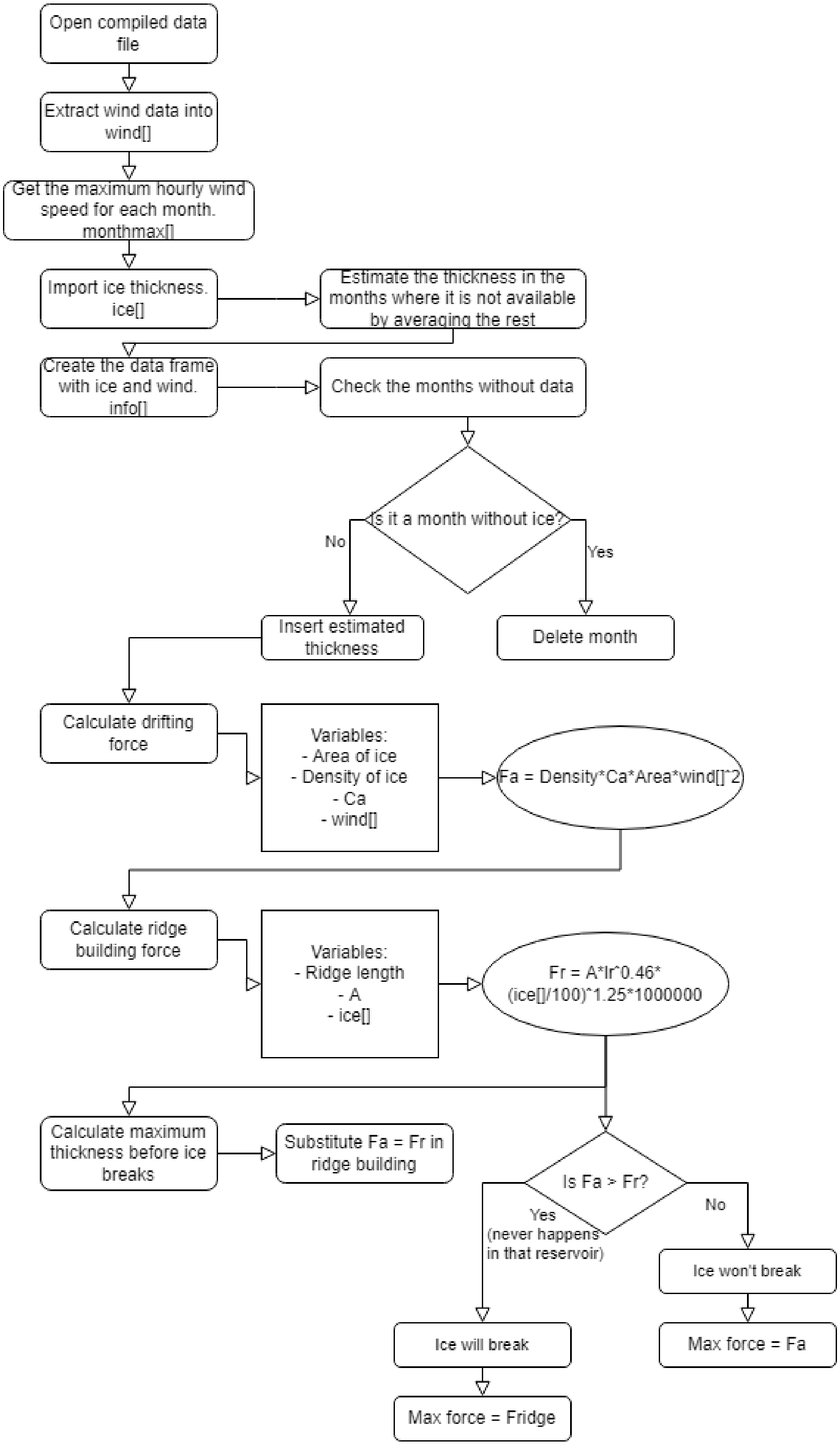


Figure 23. Flowchart of the script to check for ridges. It checks the wind force and the possibility of ridging.

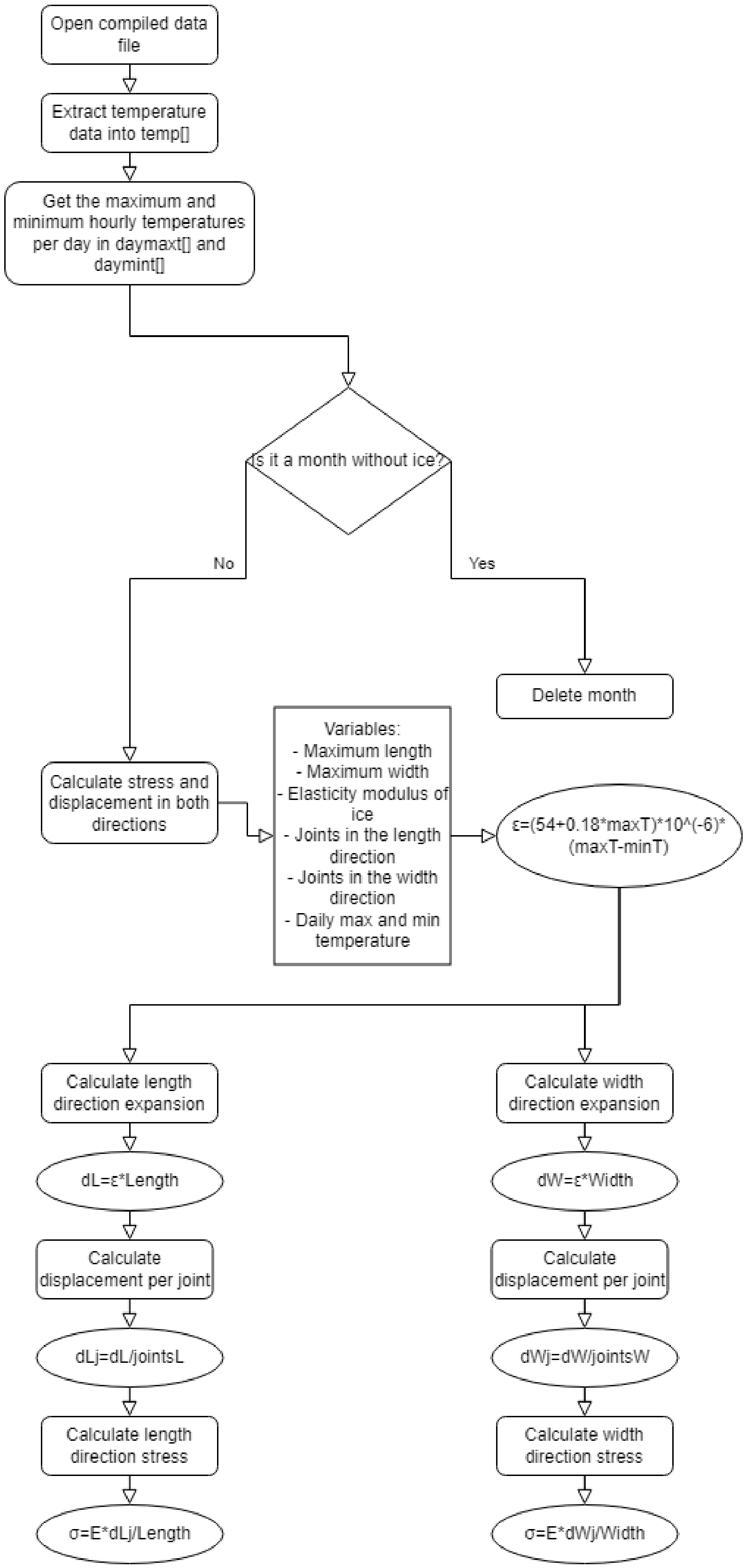


Figure 24. Flowchart of the thermal expansion script. It checks the maximum displacement and stress that can appear.

5.2.4. Drag calculation script

The final calculations script is the Drag one, as found on [Figure 25]. This script calculates two different forces. On one side, it calculates the maximum force that the ice sheet, together with the solar panels on it, can apply to the dam, and on the other side, it calculates the force that the mooring system will have to support when there is no ice due to the wind pushing the solar panels. The first step is to extract the hourly data for wind from the temporary database into a DataFrame called *wind* to extract the daily maximum to another DataFrame called *drag*. From here, the first step is to calculate the drifting force due to the wind with just the solar panels. For this purpose, Equation [3] is used. The solar panels have been considered as big pieces of ice in order to get a useful drag coefficient, which has been assumed to be 0.0085, the equivalent of very rough ice surface with over 1-metre-high features (Leppäranta, 2005). Here the area used directly depends on the blocks used (284 in this setup) and their dimensions (102.5 x 72.5 metres), and the density of the air is kept as 1.367 kg/m^3 as if it was always at -15 °C. These forces are stored in a new column of *daymax* called *Fawoi* (Force of Air WithOut Ice) that show not only the force that the mooring system will see when there's no ice, but also the extra force that the solar panels will cause on the dam when there's ice in the lake. Once this is completed, the script runs a similar function with the same Equation [3] but setting the area as 4000000 m^2 (4 km^2 that is the area of the whole lake) minus the area covered by the solar panels (calculated as in the previous function) and the drag coefficient as 0.002 (meaning a relatively smooth surface). The particularity of this function is that it checks if there is ice or not at that time. If there is, the result of the force will be the addition of the previously calculated force plus the newly calculated one, if there is none, the result will be 0. This way, the force that the dam will receive due to the wind on the ice can be estimated.

5.2.5. Plotting script

The final script is the Plot script, that follows [Figure 26]. The first line in the code will set a style for the plot so all of them can be used afterwards without visual differences. Afterwards, the script starts by showing an option menu asking which data wants to be plotted. There are 5 options, and all of them plot the requested data against the time during the whole available period of time. The first option, Air force on the ice in front of the system, will start by importing the DataFrame obtained from the Ridge force calculation script. The DataFrame will be processed to change indexes and finally the plot showing the force that the air can apply to the sheet of ice that can be found in between the shore and the solar panel system will be done. This window will be open and allow plot manipulation until it is manually closed, or any key is pressed in the script, going back to the menu again. The second option, Force on the dam due to the air, will do the same operations than the first option, but importing the database made in the Drag script. In this case, the addition of the forces of the air in the ice and the force produced on the solar panels due to the air drag will be plotted. The third option will perform the same operations as the second but showing only the force produced by the air drag, that will be the force that the mooring system will need to withstand. It is important that every option imports the database even if it has been imported before because of the modifications that are done in it to plot each parameter.

The fourth option, Stresses caused by the thermal expansion, will load the DataFrame produced by the Thermal script. In this case, as more than one parameter needs to be plotted, a big modification needs to be done in the DataFrame, putting the data from both stress columns in one and adding a new column that tells if the stress is in the Length or in the Width direction. After this modification is done, the data is plotted showing both stresses in the same graph. Just as in the previous options, an action will need to be taken in order to close the graph and go back to the menu. The last plotting option, Maximum extension of each joint, also works with the Thermal database, but in this case, the two columns that are put together are the length increase per joint and the width increase per joint. Both parameters are then plotted in one single graph. Finally, the option number 6, Exit, just closes the loop so the whole script closes and the program exits to the main script.

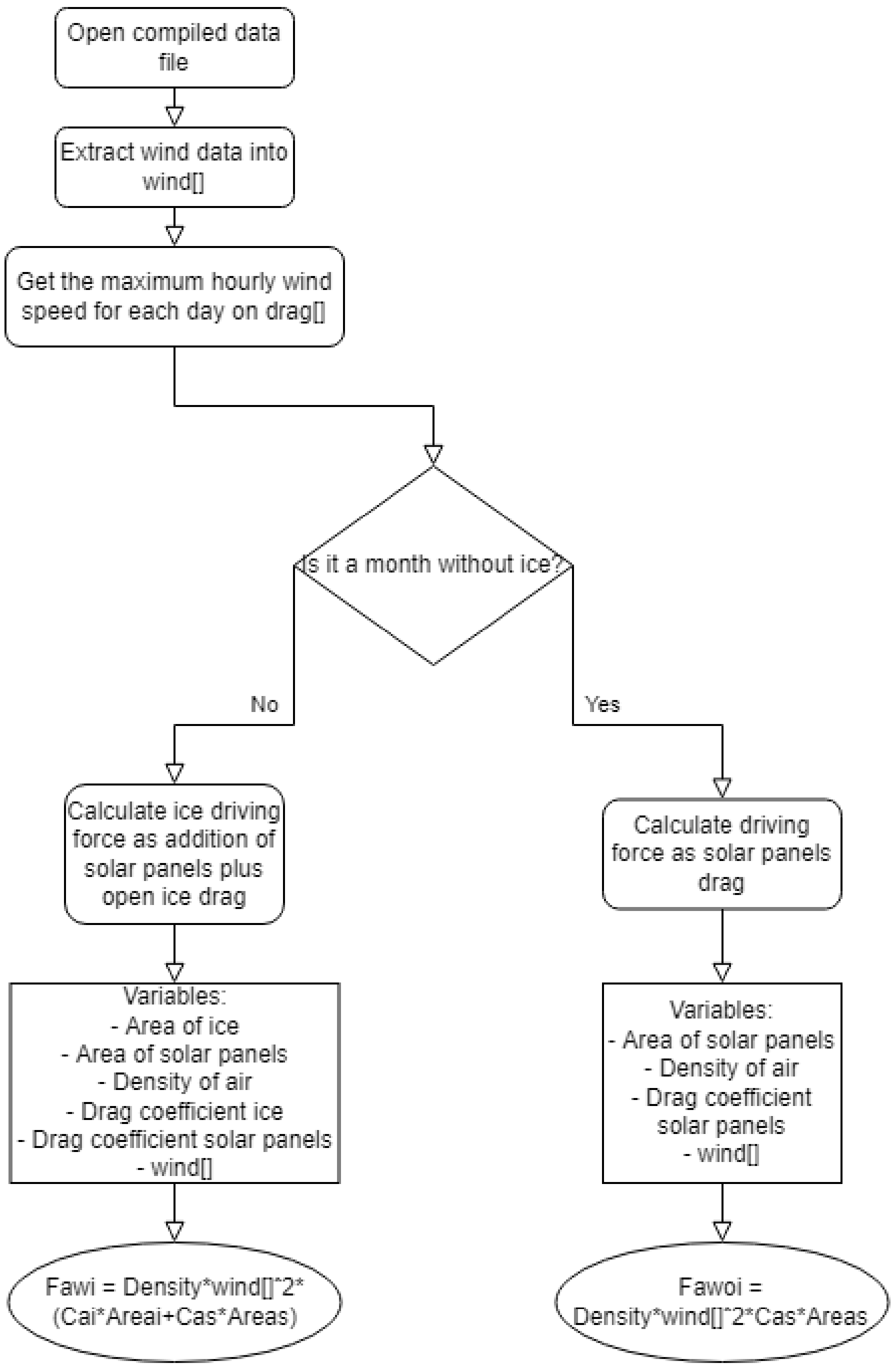


Figure 25. Flowchart of the drag script. It calculates the maximum drag that the wind can cause with and without ice.

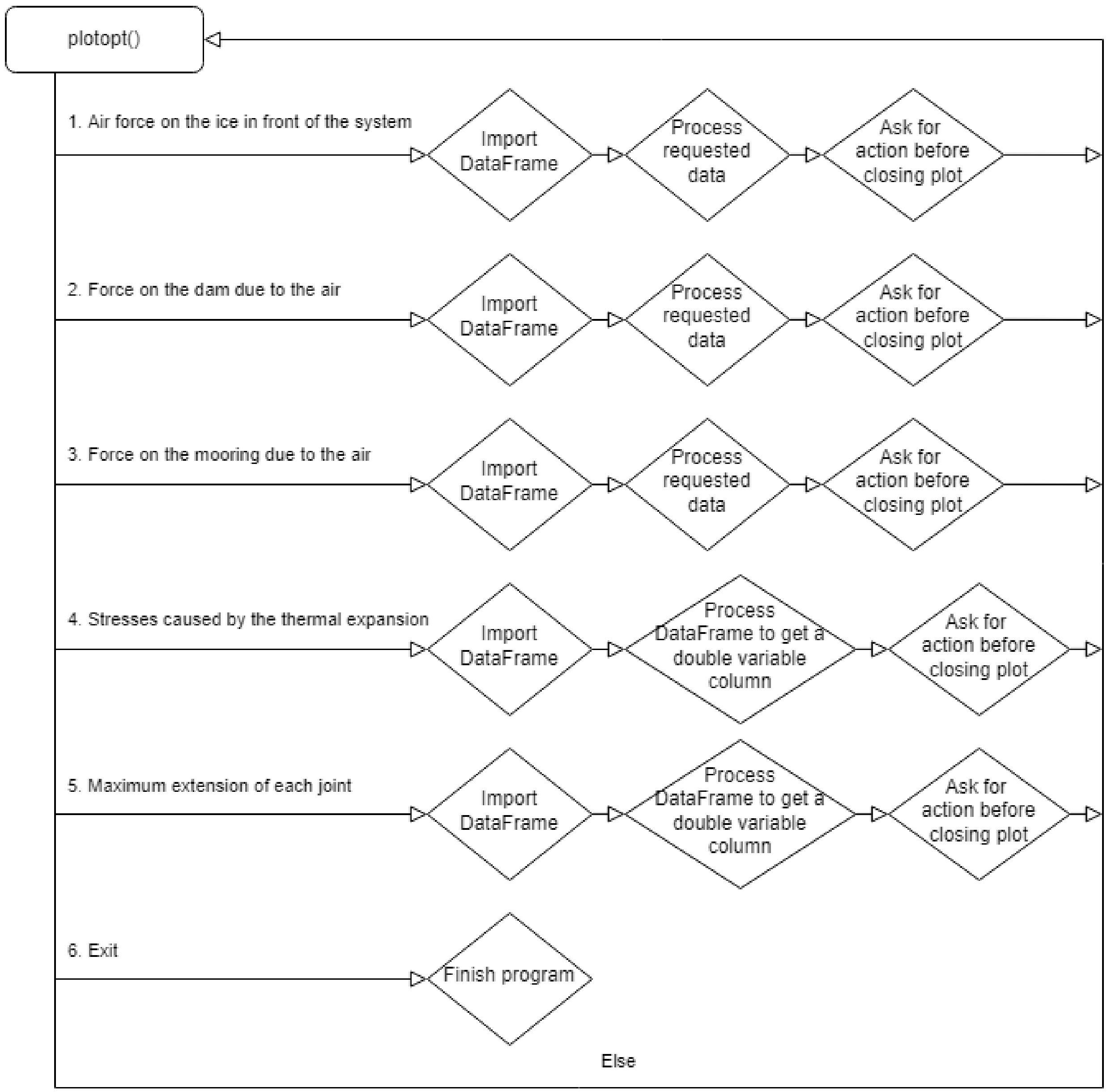


Figure 26. Flowchart of the plotting script. It shows the selectable options and the actions carried on after choosing each.

6. Results and discussions

To understand the results of each script, each section has a table with the parameters used and a table with the final results of it.

6.1. Ridge script

Area	1500000	m2
Air Density	1,367	kg/m3
Ca	0,002	-
Ridge length	1000	m
A	4	-

Table 3. Parameters used for ridge and free ice force calculations.

In the Ridge formation script, these parameters are shown in the [Table 3]. Using these parameters, plus the data obtained from the databases, both the force that the wind is able to create, and the needed force to create a ridge have been calculated for every month of the whole checked period.

In the [

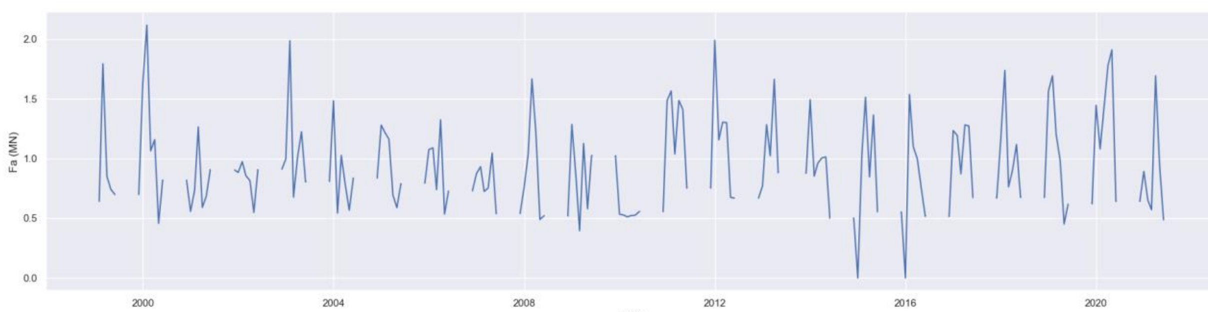


Figure 27], the force that the air can provide is shown. This graph is useful to see the peaks of force, that as can be seen as not following any trend over time. The biggest peaks appear in seemingly random years.

The result of these calculation has been that a ridge can never be created because either the wind is not strong enough or the surface of ice is not big enough. As this last one cannot change, the wind speed can be checked.

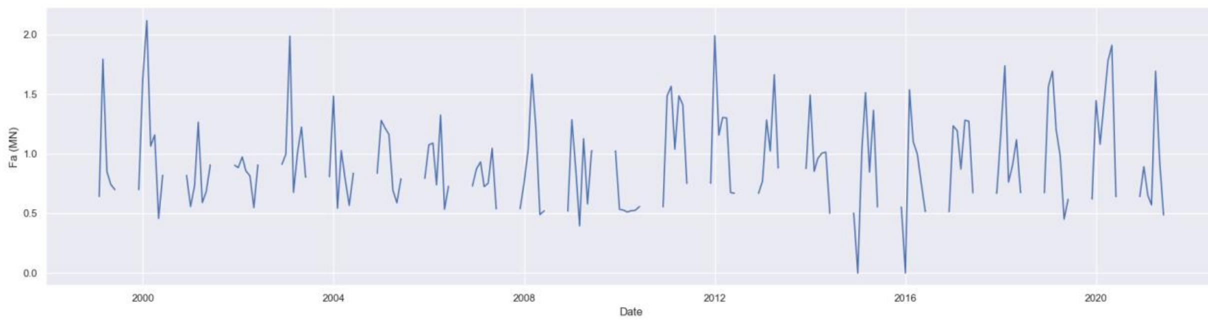


Figure 27. Force that the ice is able to produce on the free ice vs time.

With a visual inspection to the [

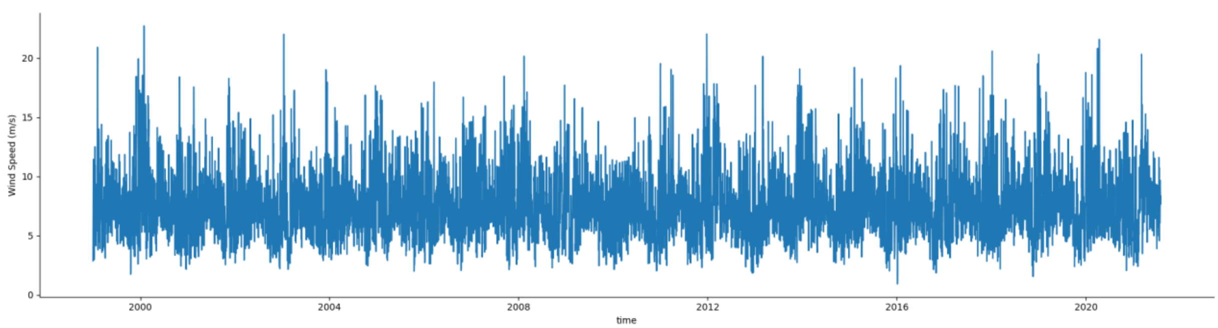


Figure 28], it can be seen that there is no trend indication an increase in wind speeds, seeing a pattern that gets repeated every year, thus, being relatively safe when assuming that the force that the ice can create on the ice due to the drag is not going to be greatly increased, and it will not reach the force needed to break the ice and create a ridge, or in this case, start piling up ice against the structure.

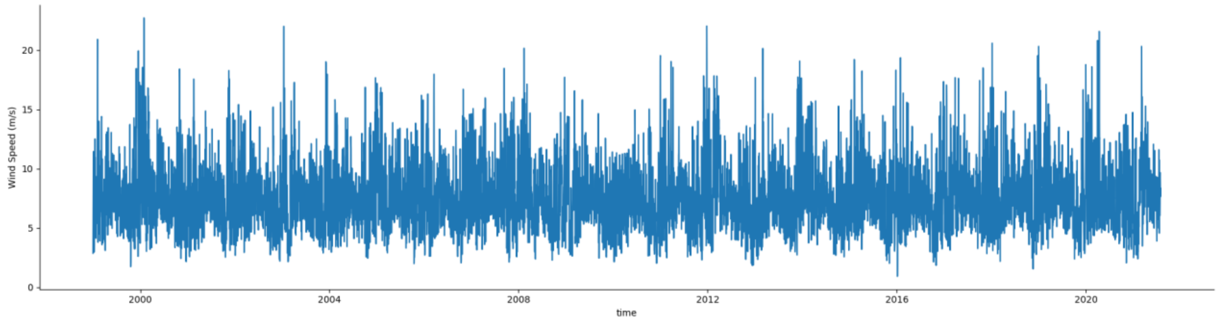


Figure 28. Wind speed against time.

By using the ridge checker script, it is possible to know that there is no moment in which a ridge will be formed. With further calculations, the force to break the ice and create a ridge is found to be on average 62 times bigger than the one the air can produce, and to be, in the worst-case scenario over 3 times bigger.

With all this information, it is safe to assume that the maximum force that will be created by the wind on the ice sheet that can potentially separate and produce extra forces on the structure [Figure 29], as stated in [Table 4], 2.1 MN, will be transferred to the rest of the ice without breaking it and without affecting the structure. These loads will be further checked on in the Drag script part.

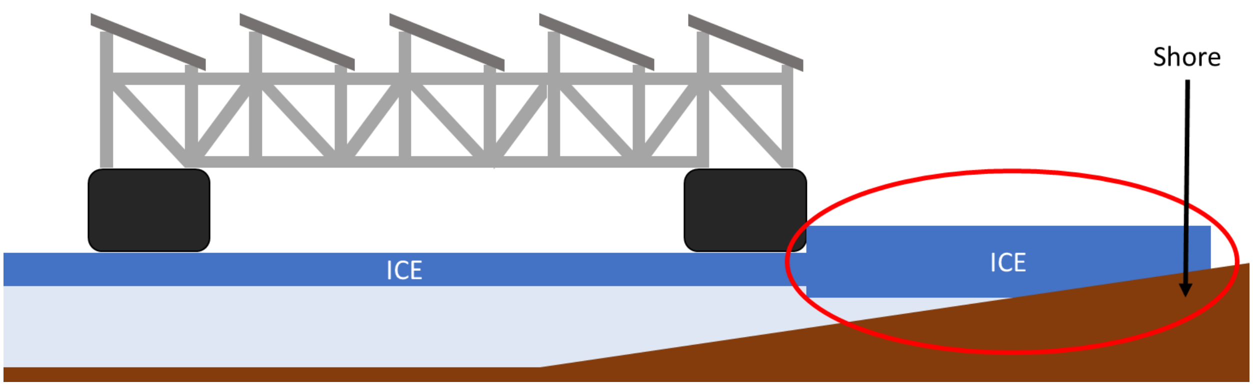


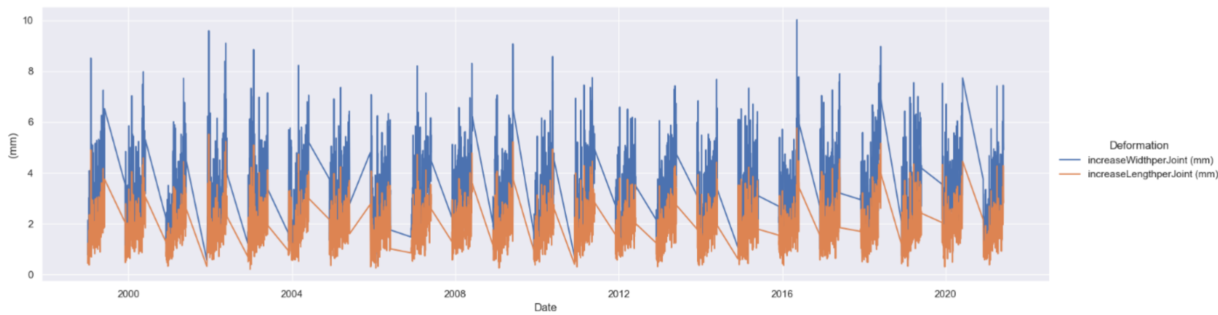
Figure 29. Ice that can break and cause forces on the structure.

Maximum force of wind in free ice	2.116.929,6	N
Average ridging force respect to drag force	6229.30	%
Minimum ridging force respect to drag force	319.50	%

Table 4. Ridge script results.

6.2. Thermal expansion script

By using the parameters in [Table 5



], the thermal script results were obtained. These results show in [Table 6] also make clear that the expansion and stress will be higher on the width direction due to the geometry of the units ([Figure 13]).

In this table can be seen that the elastic modulus of the ice is assumed as 9 GPa, that corresponds to pure ice, which is not the case, so it can be already assumed that the real stress in each joint will be smaller than the one shown in the results.

From the [Figure 30] and [Figure 31], it can be observed that, both, the stress and the deformation in the joints are bigger in the width direction, as expected because of the units being wider than they are longer. At the same time, it can be seen that the peaks on stress and deformation usually happen on the beginning or on the end of the winter season because of the bigger temperature change. To check that behaviour, the delta on temperature has been plotted, as seen in [

Figure 32], proving that there are changes in the temperature of up to 15°C.

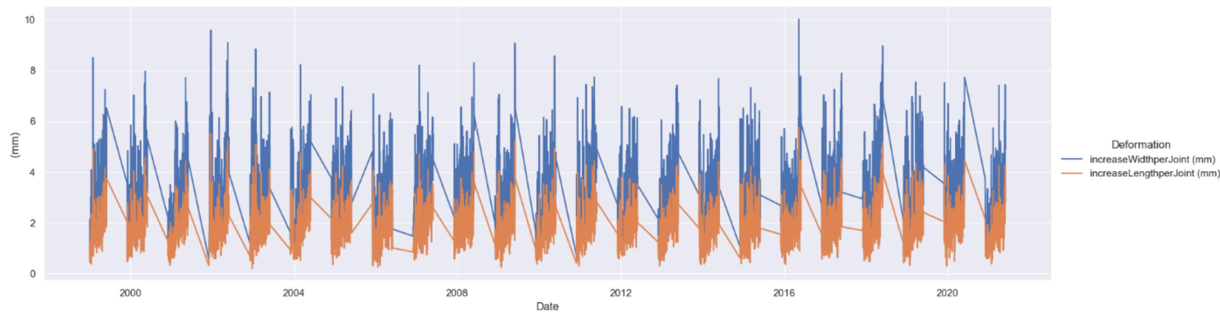


Figure 30. Maximum displacement needed in the joints in both directions.

Figure 31. Equivalent stresses in the joints in both directions.

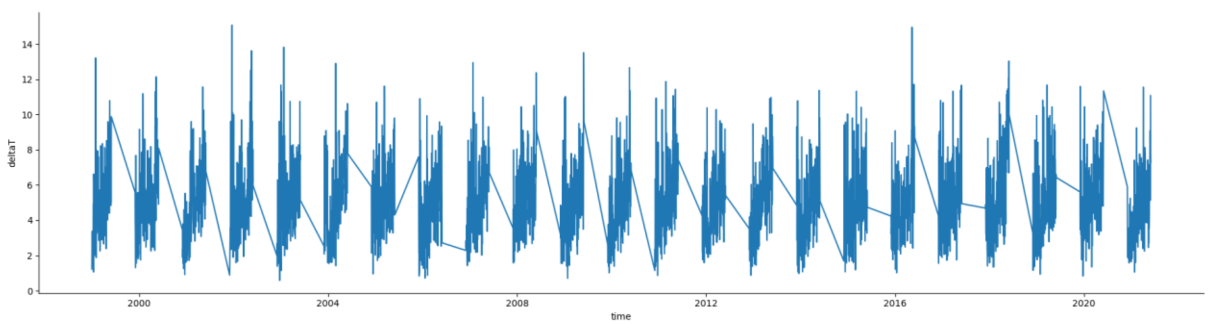


Figure 32. Daily temperature difference against time.

The rest of the parameters are just specific for this layout, the number of blocks in each direction and the position of them (as seen in [Figure 19]) defines the total length, 1947.5 metres, and width, 1572.5 metres, as well as the number of joints, that will follow the formula $Joints = 12 * Blocks - 2$, due to having 10 joints in each block and 2 joints in between them. This script is fully dependant on the precision when building the whole setup, as just a small change on the dimensions of the units, the space in between them or differences in the joints dimensions will add up, because of the big number of units, resulting on a total length or width increase that can reach the metres range.

Length	1947,5	m
Width	1572,5	m
BlocksL	26	units
BlocksW	15	units
JointsL	310	units
JointsW	178	units
E	9	GPa

Table 5. Parameters used for thermal expansion and stresses calculations.

Max increase in width	1327,07	mm
Max increase in length	1643,54	mm
Max increase in width per joint	10,02	mm
Max increase in length per joint	5,77	mm
Max stress in width	57,36	kPa
Max stress in length	26,65	kPa

Table 6. Thermal expansion script results.

This is the script with the most assumptions, what means that these results are the least accurate, but due to these same assumptions, the real expected results would be smaller, thus being on the safe side if designing for these values.

As mentioned before, the main assumptions are the fact that the change in air temperature is the equal to the change in ice temperature, which is known to be wrong, but it will offer a high maximum that will never be achieved, making the estimate more secure, the fact that the maximum stress calculation is assuming no elasticity in the joints to achieve that stress, omitting the silent blocks, and the fact that to calculate the maximum displacement in each joint. It is also assumed that the fixed points (to the mooring system) are just at the end of the blocks, something that won't be true either because there are some mechanisms needed for the mooring system that will need space to be installed. It is also not considered that the floaters can slide on top of the ice thanks to their geometry, making all of these calculation less useful to define the behaviour of the system (more on this will be discussed in Section 7 when talking about the floaters and the mooring system).

6.3. Drag force script

Blocks	284	units
Block width	102,5	m
Block length	72,5	m
Air Density	1,367	kg/m3
Ca of solar panels	0,0085	-
Ca of ice	0,002	-
Area of solar panels	2 110 475	m2
Area of ice	1 889 525	m2

Table 7. Parameters used for the air drag on the solar panels and the whole ice sheet calculations.

The calculations for the drag are based on the parameters shown in [Table 7]. The most important parameter for these calculations, the Area, directly depends on the number of blocks, but the distance between them is not considered. The air density is kept as it was in the ridging calculations and the drag coefficient for the solar panels considered as 0,0085, equivalent to very rough ice.

As can be observed in [Figure 33], the wind is stronger during the winter period of each year, and because of that, the force that the mooring system would need to withstand, that is the drag that the air will cause on the solar panels when there is no ice in the lake, would be bigger in winter ([Figure 33]). It is known, though, that those big forces will not happen because of the existence of ice, but they need to be considered in case of an abnormally warm year that would prevent the ice to be created due to the reservoir water being warmer than usual or the air temperature being higher than usual. This maximum force on the mooring is, as seen in the [Table 8], 12.66 MN, which divided among all the mooring points (516) would give a need for each mooring point to support 24.53 kN of horizontal force.

On the other hand, as can be seen in [Figure 34], the maximum force that the air can create on the ice while having the solar panels in the middle, or what is the same, the maximum force that the dam could face because of the air pushing everything that is floating towards it, is, as seen in [Table 8], 15.33 MN which is a 171.48% higher than when the solar panels were not installed (Maximum possible force without solar panels is 5.65 MN).

This difference in forces during time can be seen in [Figure 35], and, as it can be appreciated, the force increase due to the solar panels is significant enough to require further study with more precise calculations.

To reach these results there are some assumptions considered. The first, and most important one when calculating the forces that the dam can face, is that the drag force on the solar panels will be completely transferred to the ice. This is not as straightforward as it may seem, because, due to the geometry of the floaters, they may be able to slide a bit on top of the ice, being this movement restricted by the friction between the floaters and the ice and the overwater mooring system. Another assumption done when calculating the force on the dam is that all the force produced by the drag will end up there. This is also known not to be real, as the shore on the sides of the reservoir will create a reaction force (due to the friction) opposing the movement of the ice, furthermore, not all the contact on the dam side will be the dam itself, as part of the ice will try to climb the shore next to the dam, absorbing part of that force too.

Maximum force on the mooring	12,66	MN
Maximum force on the dam	15,33	MN

Table 8. Wind drag script results.

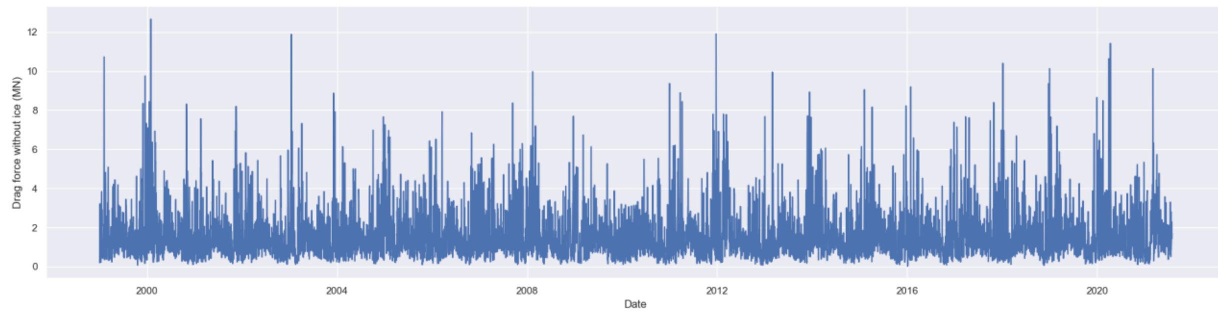


Figure 33. Drag force due to the wind on the solar panels.

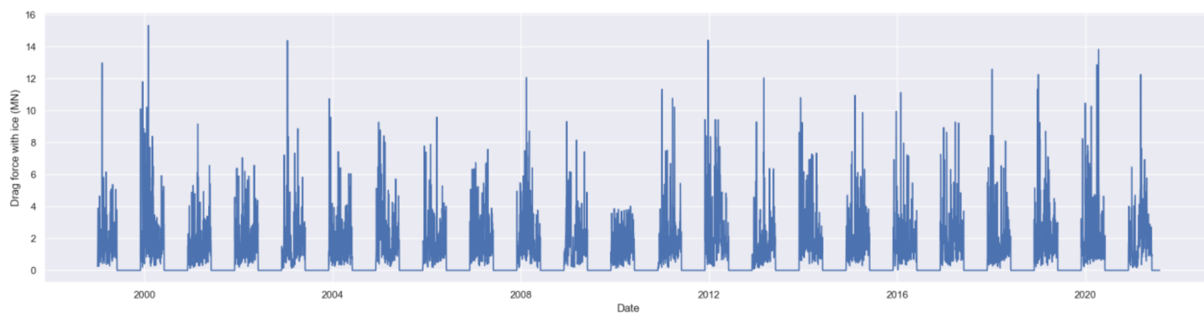


Figure 34. Drag force due to the air on the lake ice with the photovoltaic system on it.

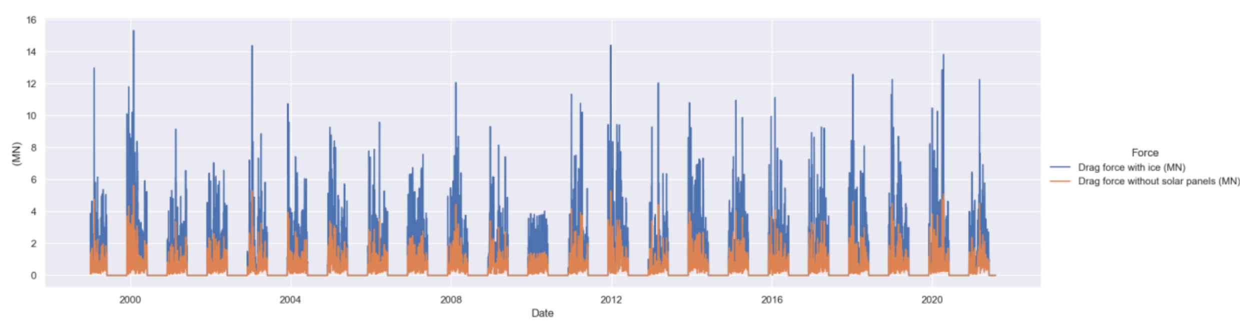


Figure 35. Comparison of the maximum drag force on the dam with and without the solar panels.

7. Assumptions, proposed solutions and further research

7.1. Solar panels mounting system

The metal structure on top of which the solar panels are mounted and that raises them 1 metre above the water is assumed to be strong enough to withstand the momentum created by the air when hitting the solar panels in such a high angle, and, at the same time be light enough not to overly increase the size of the floaters, which would increase the overall cost of the project greatly.

A further study focusing only on this structure would be necessary to ensure the overall integrity of the system once assembled.

7.2. Floaters

One of the base assumptions of this work is that the floaters will at the same time than being able to lift the whole weight of the system when floating, be able to lift the whole system when the ice is being formed and create enough friction with the ice when it is on top to partly but not fully limit the movement on top of it. In order to achieve all of this, the floaters must have a special geometry that will need to be studied and simulated. For this matter, the problems coming from the solar panels and the metal structure assembly need to be solved.

The way these floaters work when floating and with over the ice is very straightforward, but some extra parameters need to be considered when it comes to the process to move from the first to the latter, as the ice trying to compress the floater will create a deformation. For it, some elasticity in the material that will need to be allowed for the floater not to crack will be required, but, at the same time it needs to be rigid enough to be able to move up when pressed.

Some initial ideas are declined planes and bottom-cut cylinders as shown in [Figure 36], because of them being good geometries to tackle the required loads shown in [Figure 37]. It is also important to understand that the vertical component is the one needed to lift the whole system, while the horizontal ones are the ones that need to be controlled for the floaters not to break, so a floater with a large wall angle will probably work better.

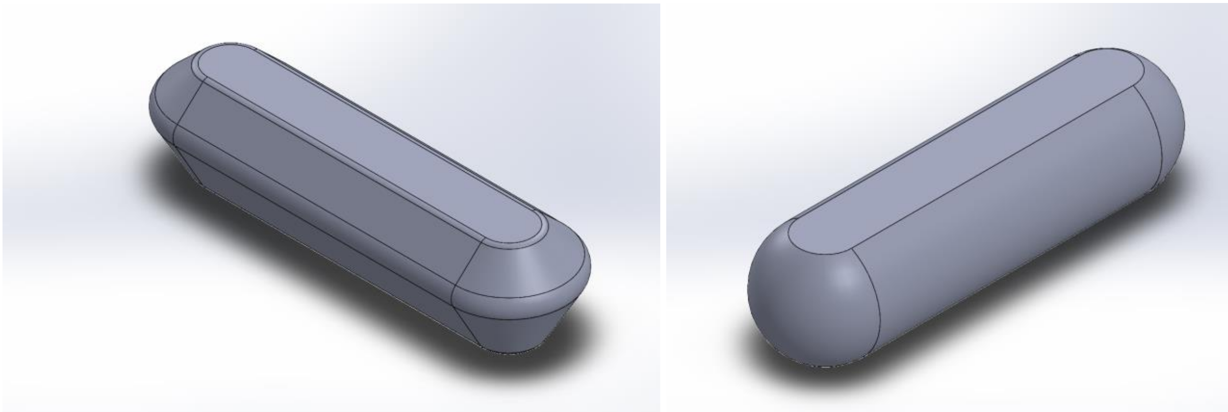


Figure 36. Geometry proposals for the floaters.

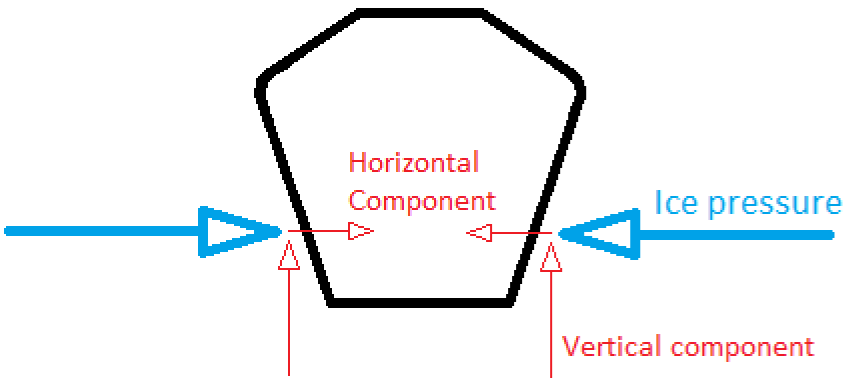


Figure 37. Ice pressure on the floater. (Arrows do not show magnitudes)

7.3. Solar panels

The solar panels need to be bi-facial. What this means is that they can generate power on both sides, increasing the efficiency of them up to a 35% (Rodríguez-Gallegos, et al., 2020), being one of the most important features to reach the needed electricity production. Furthermore, the fact that this solar panels can produce electricity on both sides means that there will be some electricity production when the solar panels are covered by snow in winter. Following the previously mentioned article numbers, the solar panels could produce up to a 26% of the total electricity production just with the back face. This could be increased by the fact that thanks to the structure on which the solar panels are mounted on, the snow that would have piled up on the bottom ice will have enough space to reflect the sunlight, thus making more likely to achieve an even higher electricity production ([Figure 38] shows the system working with snow). This electricity production would, furthermore, increase the temperature of the solar panels enough to get to positive degrees and melt a layer of the snow they have on top, causing the snow on top of the panels to fall. This system would solve several of the problems that appear, but it has not been tested yet, so it would need to be further study when planning this project.

Another system that would be able to be included but has not been considered in this project is a solar panel tilting mechanism, as it would further increase the efficiency of the whole system.

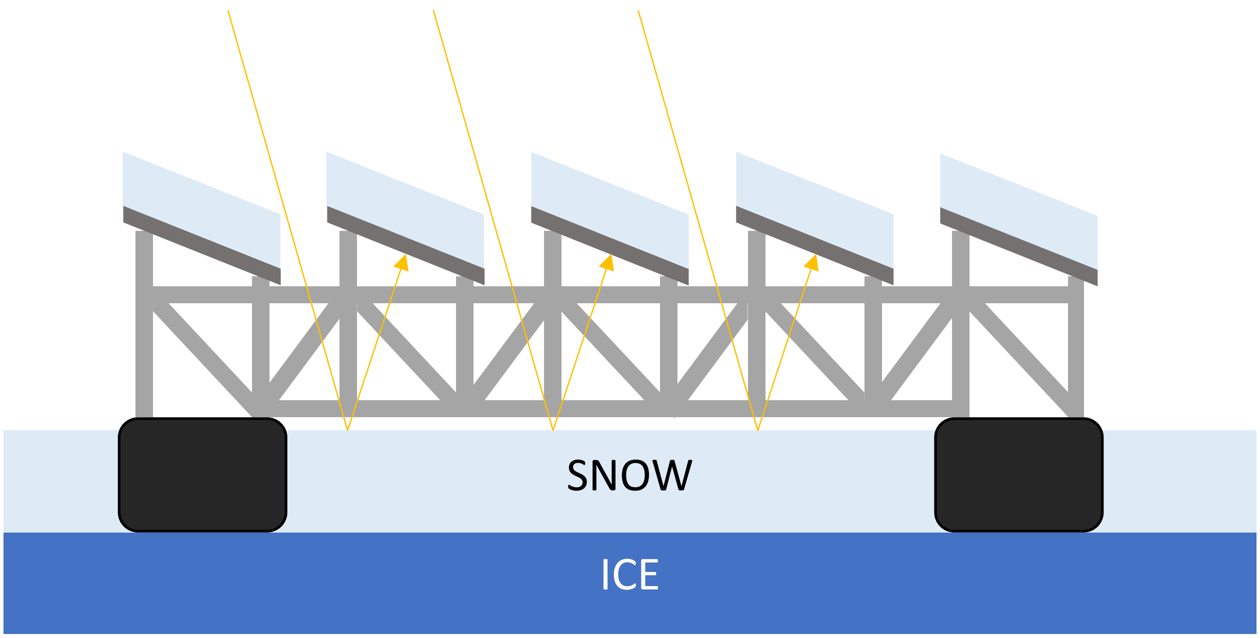


Figure 38. System working with snow. Yellow arrows being sunlight.

7.4. Ice drag force leading to ridge formation

Even though it has been checked that there should be no ridging occurring with the current wind and ice thickness, due to the solar panels blocking the wind and the sun on the surface, the ice growth is probably not the same as it is when there's nothing on top of it. This could cause that, if the thickness of the ice under the solar panels is thinner than expected (and thinner than the surroundings), and the wind manages to break it, the ice layers could start crushing, ridging, or rafting (as shown in [Figure 39]). This should be researched into, as it could mean that the floaters would face some unexpected forces that could influence their design. Furthermore, not counting with current information, should also be fixed, as it can increase the forces produced.

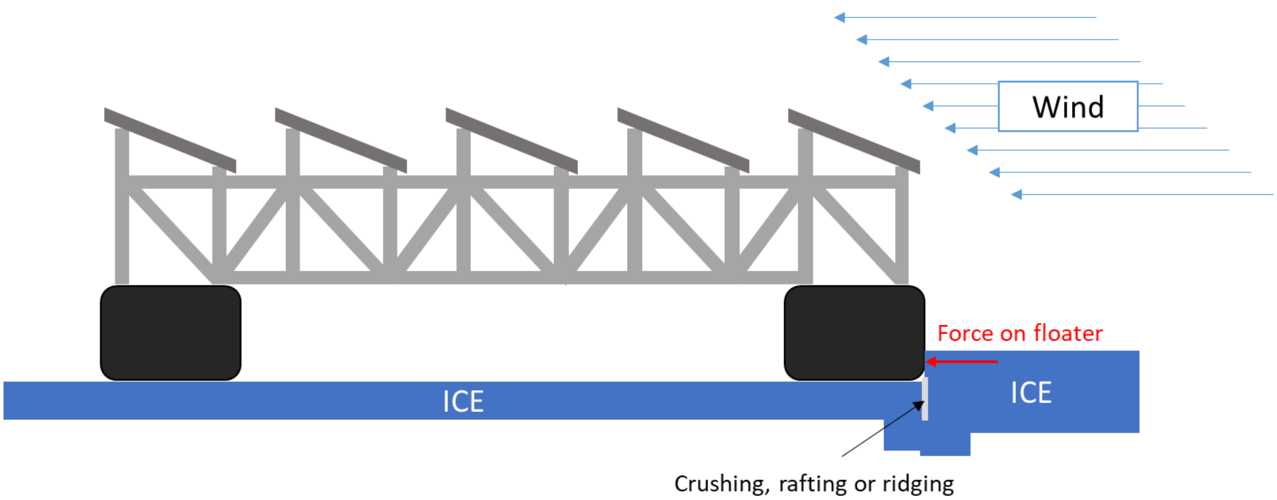


Figure 39. Ice force on floaters due to crushing, rafting, or ridging.

Another thing that should be researched into is the effect of the parameter *A*, that is the empirical factor that can be from 2 to 10, as in this case, and because of the impossibility to get the empirical factor, it has been assumed to be 4. Thus, the effect of it on the whole calculations has not been checked.

7.5. Thermal expansion of the ice

One big assumption taken regarding the thermal expansion of the ice is that all the systems will not be affected because of them being on top of the ice, but that is also something that cannot be said with big certainty, as the water may start freezing around the floaters and get them fixed in place, as shown in [Figure 40], which would either increase the loads in the joints or make them deform as exposed in Section

6. It is also important to note than for a correct calculation of the thermal expansion, the water temperature should be also considered as the change on the ice temperature will never be as big as the change on the air temperature.

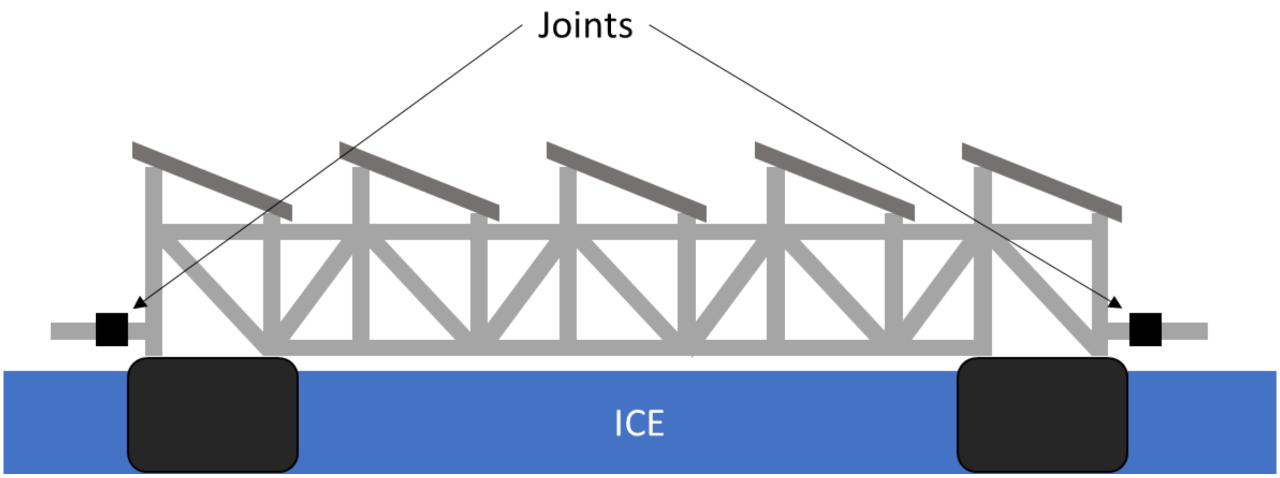


Figure 40. Photovoltaic unit with joints location.

7.6. Joints system

On top of the loads and displacement that the joints need to allow with sub-zero temperatures, they also need to withstand the wind loads when there is no ice and the possible bending in the whole system due to the weight of the mooring system. This could be fixed adding different joints in the units that would hold the mooring system to avoid all the rest of the system to be forced to bend as seen in [Figure 41].

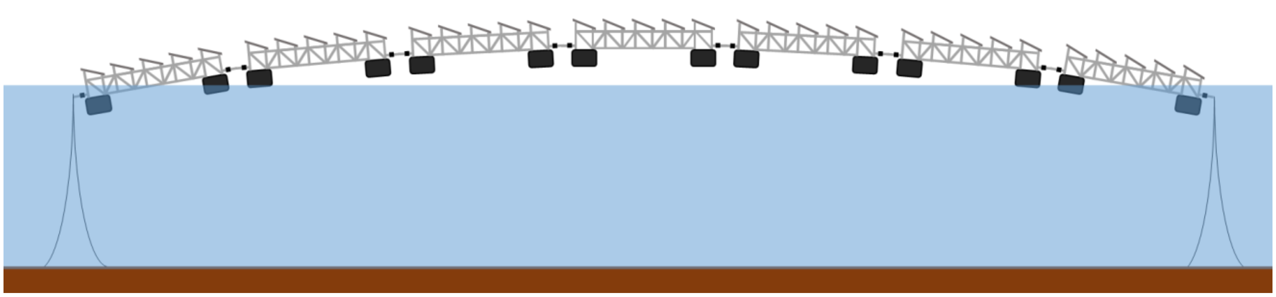


Figure 41. Exaggerated representation of the system bending.

After a small research, suitable joints to withstand the requirements of loads and displacement shown in Section 6 have been found in the market, in the form of silent blocks that on top of the aforementioned stress and deformation, can also deal with some bending in the y and z axis.

This part could lead to another research, or to be part of the research that needs to be done for the mooring system.

7.7. Mooring system

The mooring system is one of the main structural elements of the system. It must allow enough movement not to create too much stress on the joints while restraining the movement of the system on the lake.

This system must work differently when the structure is floating than when it is resting on top of ice. In the floating case, the mooring system must be designed not to be overweighted, as having 1032 mooring lines when the reservoir is on the top limit (meaning that the depth of the water will be over 50 metres as seen in [Table 9]) can produce a big force on the attachment points, making the floaters need to resist more force not to sink on creating the need to get an extra system to attach the mooring to. Furthermore, because of the big changes in the water level, there must be any system to be able to pick up some of the mooring chain length not to allow free movement on the system. This system would need to be installed either on top, which would require an extra floating structure to mount it on, or on the reservoir bed, what would require it to be waterproof [Figure 42]. In both cases a control system would be required to check the water level and allow for more or less mooring line to be released.

Minimum level	825,50	msl
Maximum level	872,79	msl
Difference	47,29	m

Table 9. Water level in the reservoir.

In the winter case, when the whole system is resting on top of ice, due to the forces seen in Section 6, the recommended solution is to release enough mooring line so that the system can have completely free movement, and is the ice the one that controls that movement by hitting the shore and trying to climb it. This way, the mooring system will not have to withstand all the loads of the system plus the ice, including not only wind and thermal expansion but also current, which has not been checked in this thesis. In order to achieve this, the system to take or give mooring line would need to be installed on the reservoir bed, as the mooring line will get stuck on the ice when it freezes. The problem that appears due to this configuration is that the thermal expansion will affect the joints on top because it will pull the mooring lines apart on the points it is frozen. This will create a need of making sure that the joints can withstand that or to add a second system on top of the ice, as the one in [Figure 42], to allow for the needed displacement of the system on top of the ice.

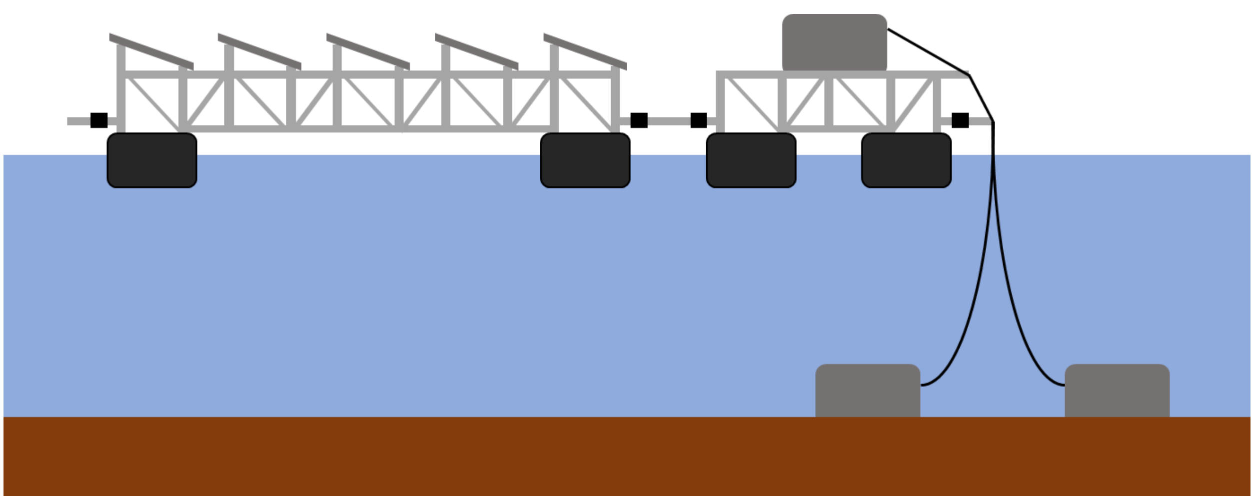


Figure 42. Representation of both options mooring line control systems proposed.

This part is the one that will presumably require for the biggest research, as all the photovoltaic grid depends on it. Specially when it comes to the system to allow for more than 50 metres of difference from the highest to the lowest point on the water level and when it comes to the weight of the mooring system, that can make the edge units to sink.

7.8. Increased ice drag

As the mooring should allow free movement while the system is resting on top of the ice, all the drag loads will be transferred to the ice, thus, greatly increasing the loads that the ice will apply on the existing dam. This makes very important to research on the force that can actually be transferred from the photovoltaic grid to the ice on which it is resting, and if there is a need for mooring it so it does not freely slide until it gets to the shore. The thickness of the ice underneath the system is also an important parameter, as it can break because of the movement of the system on top, making it falling to the water and transferring only part of the load. On top of the statement previously given, the current is also not considered because of the lack of that data, what means that the result could variate even more. This part is completely based on assumptions, and if any of the assumptions taken is incorrect, it would mean that a whole part of the system would need to be rethought.

8. Conclusion

To sum up, this thesis compiles the work, research and thinking done to check the viability of installing a big solar panel system in the Falningsjøen reservoir that TrønderEnergi uses to produce hydropower electricity. The benefits, as the inexistent need for land acquisition, the regulation of the water temperature, etc… that make this project interesting have been presented. The theory needed to understand the thesis have also been explained, as well as technical information about the reservoir and the proposed system setup and layout. The main challenges, as the change on the ice conditions because of the solar panels covering the surface, the possibility of the wind producing forces that can damage the structure by making ice crush on it, the thermal expansion producing stresses or deformations on the solar panels structure, the design of the mooring and joints systems to withstand the extra forces that can appear because of the freezing conditions or the increased drag that will increase the force on the dam, are explained and an initial solution is given for each of them. The calculations done to solve the 3 major challenges are explained together with an explanation of how the tool created to solve them works. This tool, divided in 3 parts, calculates the forces that can be applied to the structure and damage it, the maximum displacement and maximum stress that the thermal expansion can produce, and the force that will arrive to the mooring system (when there is no ice) and the dam (when the reservoir is frozen) because of the drag produced by the wind on the solar panels system. Finally, the results of the calculations are presented and discussed. These results indicate that, there will be no ice force on the structure, due to the inability of the air to make the ice ridge, the maximum expansion in a joint is about 10 mm, the maximum stress that can appear in a joint is 57.5 kPa, the maximum force that can be received in the mooring system is about 13 MN, and the maximum force that the dam could have to support is about 15.5 MN. The expected research to be carried on to be able to correctly design every part of the system is also presented to be able to move forward with the project.

After all this work, this whole project seems technically viable. However, a lot of the assumptions taken to be able to give answers to the problems and calculate important information will need to be checked. Some of them, such as the ones affecting the mooring system, are of major importance, meaning that being one of those assumptions wrong would need the whole project to be restudied. Besides all the assumptions, some of the proposed solutions need to be in-dept studied, as they are just ideas that seem technically doable, but which economically viability has not been checked.

9. Bibliography

Abdolzadeh, M., & Ameri, M. (January 2009). Improving the effectiveness of a photovoltaic water pumping system by spraying water over the front of photovoltaic cells. *Renewable Energy, Volume 34, Issue 1*, 91-96.

Dörenkämper, M., Arifeen, W., Abhishek, K., de Jong, M., Jan, K., & Thomas, R. (15 January 2021). The cooling effect of floating PV in two different climate zones: A comparison of field test data from the Netherlands and Singapore. *Solar Energy, Volume 214*, 239-247.

Høyland, K. V. (2021). Compendium on Ice actions from TBA4260 (NTNU).

ISO19906. (2010b). *Petroleum and natural gas industries - Arctic offshore structures. Tech.* Geneva, Switzerland: International Standard, International Standardization organization.

Leppäranta, M. (2005). The drift of sea ice. ISBN 3-540-40881-9: Springer-Praxis, Springer-Verlag Berlin Heidelberg New York.

Michel, B., & Drouin, M. (1971). *Les poussées d'origin thermique exercée par les couvert de galce sur les structures hydraulique. Tech. Rep. Rapport S-23.* Québew, Canada: Faculté des Science, Génie Civil, Université Lavel.

Rodríguez-Gallegos, C., Liu, H., Gandhi, O., Singh, J., Krishnamurthy, V., Kumar, A., . . . Peters, I. (2020, July 15). Global Techno-Economic Performance of Bifacial and Tracking Photovoltaic Systems. *Joule, 4*(7), 1514-1541. doi:https://doi.org/10.1016/j.joule.2020.05.005

Romande Energie. (2021, 06 07). *TU*. Retrieved 05 22, 2022, from Flytende solcellepaneler i høyden leverer mye mer energi: https://www.tu.no/artikler/flytende-solcellepaneler-i-hoyden-leverer-mye-mer-energi/510833

Sitdikov, R. N. (2016, 06 27). *the Arctic*. Retrieved 05 25, 2022, from New Rosatom engine for the Arktika, the world's most powerful nuclear icebreaker: https://arctic.ru/infrastructure/20160627/380363.html

Statista. (2022, 03 23). Retrieved 05 22, 2022, from Net electricity consumption per capita in the residential sector in Norway from 2008 to 2020: https://www.statista.com/statistics/1025221/net-electricity-consumption-per-capita-in-households-in-norway/

Tuhkuri, J. (2014). Ice ridge formation. *Tech. rep., Encyclopedia of Life Support System*, 18p.

Yan, C., Qu, M., Chen, Y., & Feng, M. (August 2020). Snow removal method for self-heating of photovoltaic panels and its feasibility study. *Solar Energy, Volume 206*, 374-380.

Appendix A – Databases parameters

Parameter	Unit	Description	Type
time		UTC time stamp (YYYY-MM-DD HH:MM)	
wSpeed.x	m/s	Wind speeds at different heights above ground (x).	Instantaneous
wDir.x	deg	Wind directions at different heights above ground (x).	Instantaneous
sqrtTKE.x	m/s	Wind speed given as standard deviation in m/s. Derived from the turbulent kinetic energy (TKE) at different heights above ground (x).	Instantaneous
press.x	Pa	Pressure at different heights above ground (x).	Instantaneous
temp.x	Celsius	Temperature at different heights above ground (x).	Instantaneous
rh.x	%	Relative humidity at different heights above ground (x).	Instantaneous
cloudWater.x	mg/kg	Cloud water content at different heights above ground (x).	Instantaneous
cloudIce.x	mg/kg	Cloud icing content at different heights above ground (x).	Instantaneous
psfc	Pa	Pressure at site	Instantaneous
msl	Pa	Pressure at mean sea level	Instantaneous
wSpeed.850hpa	m/s	Wind speeds at pressure level 850hPa.	Instantaneous
wDir.850hpa	deg	Wind speeds at pressure levels 850hPa.	Instantaneous
temperature.2	Celsius	Temperatures at 2m	Instantaneous
waterTemp	Celsius	Water temperature	Instantaneous
soilTemp.0-10cm	Celsius	The temperature in the upper 10 cm of the soil	Instantaneous
relHumidity.2	%	Relative humidity in height 2m above ground level	Instantaneous
snowDepth	m	Snow depth (if present)	Instantaneous
vis.s	m	Visibility at surface	Instantaneous
sensHeatFlux.s	w/m2	Sensible Heat Flux at surface	Instantaneous
totPrecip.s	kg/m^2	Total Precipitation at surface	1h Accumulated
downShortWaveFlux.s	w/m2	Downward shortwave irradiance at surface	1h Average
swdDir.s	w/m2	Direct shortwave irradiance at surface	1h Average
swdDif.s	w/m2	Diffuse shortwave irradiance at surface	1h Average
cloudBottom	m	Height of cloud bottom	Instantaneous
cloudTop	m	Height of cloud top	Instantaneous
totalCloudCover.a	%	Total cloud cover in atmosphere	1h Average
convCloudCover.a	%	Convective cloud cover in atmosphere	1h Average
rmol	1/m	Inverse Monin-Obukhov-Length	
znt	m	Roughness length	Instantaneous
u*	m/s	U-start (friction velocity)	Instantaneous
pblh	m	Height of the PBL boundary layer	Instantaneous

Appendix B – Code

Main.py

```
def options():
    print('What do you want to do?')
    print('''1. Compile Data files
2. Calculate forces and check if ridging will occur
3. Calculate stress and displacements due to Thermal stresses
4. Calculate the drag force due to the wind
5. Plot
6. EXIT''')

i = 0
while i < 10:
    options()
    if int(input()) == 1:
        import Compiler
        options()
    if int(input()) == 2:
        import Ridge
        options()
    if int(input()) == 3:
        import Thermal
        options()
    if int(input()) == 4:
        import Drag
        options()
    if int(input()) == 5:
        import Plot
        options()
    else:
        print('Exiting')
```

Compiler.py

```
file = './Data/EmdEuropeEra5_N62.605072_E010.410248_%s.txt'

print('Which year do you want to start compiling at? (1999-2020)?')
yearmin = input()
yearmin = int(yearmin)
print('Which year do you want to finish compiling at? (2000-2021)?')
yearmax = input()
yearmax= int(yearmax)

year = yearmin

while year <= yearmax:
    f = open(file % year, 'r')
    lines = f.readlines()
    f.close()
    # delete useless information
```

```
    if year == yearmin:
        del lines[0:10]
    else:
        del lines[0:11]
    if year == yearmin:
        tempf = open('./Data/temporary.txt', 'w+')
        for line in lines:
            tempf.write(line)
        tempf.close()
    else:
        tempf = open('./Data/temporary.txt', 'a+')
        for line in lines:
            tempf.write(line)
        tempf.close()
    year += 1
else:
    print('Compilation finished')
```

Ridge.py

```
import pandas as pd
from sympy.solvers import solve
from sympy import Symbol

df = pd.read_csv('./Data/temporary.txt', sep="\t", index_col=0)
wind = df[['wSpeed.10']]
wind.index = pd.to_datetime(wind.index)

# Get the maximum hourly wind speed per month
monthmax = pd.DataFrame()
monthmax = wind.resample('m').max()
monthmax.columns = ['wSpeed']

# Import ice csv
ice = pd.read_csv('./Data/Ice thickness.csv', sep="|", index_col=0)
ice.columns = ['iThickness']

# Import snow csv
snow = pd.read_csv('./Data/Snow thickness.csv', sep="|", index_col=0)
snow.columns = ['sThickness']

# Create a DataFrame with wind speed, ice thickness and snow thickness
info = monthmax
month = []
for i in range(1999, 2021, 1):
    for j in range(1, 13, 1):
        month.append('%i-%i' % (i, j))
for i in range(2021, 2022, 1):
    for j in range(1, 8, 1):
        month.append('%i-%i' % (i, j))
info.index = month
info.index = pd.to_datetime(info.index)
info.reset_index(level=0, inplace=True)
info = info.join(ice).join(snow)
```

```
info = info.set_index('index')

# Remove from june to november (no ice)
info = info.drop(info.index[info.index.month.isin([6, 7, 8, 9, 10, 11])])

# Get maximum for each month
monthmaxice = info['iThickness'].groupby(info.index.month).max()
monthmaxsnow = info['sThickness'].groupby(info.index.month).max()

# Add estimations of max ice and snow in May (no data)
monthmaxice[5] = 53
monthmaxsnow[5] = 0

# Check months and substitute NaN for maximum (worst case scenario)
info.reset_index(level=0, inplace=True)
for i in info[info.iThickness.isnull()].index:
    info.loc[i, 'iThickness'] = monthmaxice[info.loc[i, 'index'].month]
for i in info[info.sThickness.isnull()].index:
    info.loc[i, 'sThickness'] = monthmaxsnow[info.loc[i, 'index'].month]
info = info.set_index('index')

# Calculate drifting force
Area = 1500000  # 1.5 km2 (maximum area that the air can push against the
dam), max lake area=4 km2
aDensity = 1.367    # kg/m3 at -15°C
Ca = 0.002
Fa = []
info.reset_index(level=0, inplace=True)
for i in range(0, len(info), 1):
    if info.at[i, 'iThickness'] != 0:
        Fa.insert(i, aDensity * Ca * Area * info.at[i, 'wSpeed'] ** 2)
    else:
        Fa.insert(i, 0)
info = info.assign(Fa=Fa)
info = info.set_index('index')

# Calculate force required to build a ridge
lr = 1000  # m, width of the ridge
A = 4    # Empirical factor (2-10, but closer to 2)
Fr = []
info.reset_index(level=0, inplace=True)
for i in range(0, len(info), 1):
    Fr.insert(i, A * lr ** 0.46 * (info.at[i, 'iThickness'] / 100) ** 1.25 *
1000000)   # CHECK WITH KNUT
info = info.assign(Fr=Fr)
info = info.set_index('index')


# Check if a ridge can be created
Ridge = []
info.reset_index(level=0, inplace=True)
for i in range(0, len(info), 1):
    if ((info.at[i, 'Fa'] >= info.at[i, 'Fr']) and (info.at[i, 'Fr'] != 0)):
        Ridge.insert(i, 'YES')
    else:
        Ridge.insert(i, 'NO')
info = info.assign(Ridge=Ridge)
```

```
info = info.set_index('index')
Ridges = info[info['Ridge'].str.contains("YES")]

#Check the maximum force that the dam will support
if len(Ridges.index) == 0:
    Max_force = info['Fa'].max()
else:
    for i in range(0, len(info), 1):
        if ((info.at[i, 'Ridge'] == "YES") and (info.at[i, 'Fr'] >=
Max_force)):
            Max_force = info['Fr'].max()
print ('Max force %sN' % Max_force)

# Properly name columns
info.columns = ['wSpeed (m/s)','iThickness (cm)','sThickness (cm)','Fa
(N)','Fr (N)','Ridge']

# Export info DataFrame
info.to_csv(r'./Data/DataFrames/info.csv', decimal=',')

#Check ice thickness required for the ridge to create
Area = 2000000  # 1.5 km2 (maximum area that the air can push against the
dam), max lake area=4 km2
aDensity = 1.367    # kg/m3 at -15°C
Ca = 0.002
lr = 400  # m, width of the ridge
A = 4    # Empirical factor (2-10, but closer to 2)
Icethick = []
info.reset_index(level=0, inplace=True)
for i in range(0, len(info), 1):
    x = Symbol('x')
    Icethick.insert(i, print(solve(A*(lr**0.46)*((x/100)**1.25)*1000000-
info.at[i, 'Fa (N)'])))
info = info.assign(Icethick=Icethick)
info = info.set_index('index')
```

Thermal.py

```
import pandas as pd

df = pd.read_csv('./Data/temporary.txt', sep="\t", index_col=0)
temp = df[['temp.10']]
temp.index = pd.to_datetime(temp.index)

# Get the maximum and minimum hourly temperatures per day
daymaxt = pd.DataFrame()
daymaxt = temp.resample('d').min()
daymaxt.columns = ['mintemp']
daymint =pd.DataFrame()
daymint = temp.resample('d').max()
daymint.columns = ['maxtemp']
daymaxt = daymaxt.join(daymint)

# Delete the months without ice
daymaxt = daymaxt.drop(daymaxt.index[daymaxt.index.month.isin([6, 7, 8, 9,
```

```
10, 11])])

# Calculate stress and displacement total and per joint
Length = 1947.5     # Length of the whole grid (m)
Width = 1572.5      # Width of the whole grid (m)
BlocksL = 26
BlocksW = 15
JointsL = 12 * BlocksL - 2
JointsW = 12 * BlocksW - 2
E = 9   # GPa 4-9, 9 is the wcs
# Variables
Lengthincrease = []
Widthincrease = []
Lengthincreasepj = []
Widthincreasepj = []
EqStressL = []   # E = stress / strain
EqStressW = []
daymaxt.reset_index(level=0, inplace=True)
for i in range(0, len(daymaxt), 1):
    Lengthincrease.insert(i, (54 + 0.18 * daymaxt.at[i, 'maxtemp']) * 10 **
(-6) * Length * (daymaxt.at[i, 'maxtemp']-daymaxt.at[i, 'mintemp']) * 1000)
    Widthincrease.insert(i, (54 + 0.18 * daymaxt.at[i, 'maxtemp']) * 10 ** (-
6) * Width * (daymaxt.at[i, 'maxtemp']-daymaxt.at[i, 'mintemp']) * 1000)
    Lengthincreasepj.insert(i, Lengthincrease[i] / JointsL)
    Widthincreasepj.insert(i, Lengthincrease[i] / JointsW)
    EqStressW.insert(i, E * (Widthincreasepj[i] / (Width * 1000)))
    EqStressL.insert(i, E * (Lengthincreasepj[i] / (Length * 1000)))
daymaxt =
daymaxt.assign(Lengthincrease=Lengthincrease).assign(Widthincrease=Widthincre
ase).assign(Widthincreasepj = Widthincreasepj).assign(Lengthincreasepj =
Lengthincreasepj).assign(EqStressW=EqStressW).assign(EqStressL=EqStressL)
daymaxt = daymaxt.set_index('time')

#Show end results
maxincwidth = daymaxt.Widthincrease.max()
print ('Max increase in Width %s mm' % daymaxt.Widthincrease.max())
maxinclength = daymaxt.Lengthincrease.max()
print ('Max increase in Length %s mm' % daymaxt.Lengthincrease.max())
maxincwidthjoint = daymaxt.Widthincreasepj.max()
print ('Max increase per Width joint %s mm' % daymaxt.Widthincreasepj.max())
maxinclengthjoint = daymaxt.Lengthincreasepj.max()
print ('Max increase per Length joint %s mm' %
daymaxt.Lengthincreasepj.max())
maxstresslengthjoint = daymaxt.EqStressL[daymaxt.Lengthincrease.idxmax()]
print ('Max stress per joint %s GPa length' %
daymaxt.EqStressL[daymaxt.Lengthincrease.idxmax()])
maxstresswidthjoint = daymaxt.EqStressW[daymaxt.Widthincrease.idxmax()]
print ('Max stress per joint %s GPa width' %
daymaxt.EqStressW[daymaxt.Widthincrease.idxmax()])

#Properly name columns
daymaxt.columns = ['mintemp (C)','maxtemp (C)','increaseLength
(mm)','increaseWidth (mm)','increaseWidthperJoint
(mm)','increaseLengthperJoint (mm)','EquivalentStressW
(GPa)','EquivalentStressL (GPa)']
```

```
# Export daymaxt DataFrame
daymaxt.to_csv(r'./Data/DataFrames/daymaxt.csv', decimal=',')
)
```

Drag.py

```
import pandas as pd

df = pd.read_csv('./Data/temporary.txt', sep="\t", index_col=0)
drag = df[['wSpeed.10']]
drag.index = pd.to_datetime(drag.index)

# Get the maximum hourly wind speed per day
drag = drag.resample('d').max()
drag.columns = ['wSpeed']

# Calculate drifting force with just solar panels
blocks = 284
blockw = 102.5 # width of one block in metres
blockl = 72.5 # length of one block in metres
Area = blocks*blockw*blockl  # depends on the installed blocks. max lake
area=4 km2
aDensity = 1.367    # kg/m3 at -15°C
Ca = 0.0085
Fawoi = []
drag.reset_index(level=0, inplace=True)
for i in range(0, len(drag), 1):
    Fawoi.insert(i, aDensity * Ca * Area * drag.at[i, 'wSpeed'] ** 2)
drag = drag.assign(Fawoi=Fawoi)
drag = drag.set_index('time')

# Calculate drifting force with ice
blocks = 284
blockw = 102.5 # width of one block in metres
blockl = 72.5 # length of one block in metres
Areawi = 4000000 - blocks*blockw*blockl  # depends on the installed blocks.
max lake area=4 km2
Cai = 0.002
Fawi = []
Fawosp = []
drag.reset_index(level=0, inplace=True)
for i in range(0, len(drag), 1):
    if 6 <= drag.at[i,'time'].month <=11 :
        Fawi.insert(i, 0)
        Fawosp.insert(i, 0)
    else:
        Fawi.insert(i, (aDensity * Cai * Areawi * drag.at[i, 'wSpeed'] ** 2)
+ (aDensity * Ca * Area * drag.at[i, 'wSpeed'] ** 2))
        Fawosp.insert(i, aDensity * Cai * 4000000 * drag.at[i, 'wSpeed'] **
2)
drag = drag.assign(Fawi=Fawi).assign(Fawosp=Fawosp)
drag = drag.set_index('time')

# Check and print the maximum forces with and without ice
Maxforcewi = drag['Fawi'].max()
Maxforcewoi = drag['Fawoi'].max()
```

```
Maxforcewosp = drag['Fawosp'].max()
Forceincrease = (Maxforcewi/Maxforcewosp-1)*100
print ('Max force with ice %s N' % Maxforcewi)
print ('Max force without ice %s N' % Maxforcewoi)
print ('Max force without solar panels %s N' % Maxforcewosp)
print ('Solar panels force increase %.2f%%' % Forceincrease)

# Properly name columns
drag.columns = ['wSpeed (m/s)','Drag force without ice (N)','Drag force with
ice (N)','Drag force without solar panels (N)']

# Export Drag DataFrame
drag.to_csv(r'./Data/DataFrames/drag.csv', decimal=',')
```

Plot.py

```
import seaborn as sns
import matplotlib.pyplot as plt
import pandas as pd

# Apply the default theme
sns.set_theme()

def plotopt():
    print('What do you want to plot?')
    print('''1. Air force on the ice in front of the system
2. Force on the dam due to the air
3. Force on the mooring due to the air
4. Stresses caused by the thermal expansion
5. Maximum extension of each joint
6. EXIT''')

i = 0
while i < 10:
    plotopt()
    if int(input()) == 1: # Air force on ice in front of system
        # Import data
        info = pd.read_csv(r'./Data/DataFrames/info.csv', index_col=0)
        info.index.names = ["Date"]
        info.index = pd.to_datetime(info.index, format='%Y-%m-%d')
        # Create a visualization
        sns.relplot(data=info["wSpeed (m/s)"], kind="line")
        plt.show()
        print('Press any key to go back to the plot menu')
        input()
        plotopt()
    if int(input()) == 2: # Force on the dam due to the air
        # Import data
        drag = pd.read_csv(r'./Data/DataFrames/drag.csv', index_col=0)
        drag.index.names = ["Date"]
        drag.index = pd.to_datetime(drag.index, format='%Y-%m-%d')
        # Create a visualization
        sns.relplot(data=drag["Drag force with ice (N)"], kind="line")
        plt.show()
        print('Press any key to go back to the plot menu')
```

```
        input()
        plotopt()
    if int(input()) == 3: # Force on the mooring due to the air
        # Import data
        drag = pd.read_csv(r'./Data/DataFrames/drag.csv', index_col=0)
        drag.index.names = ["Date"]
        drag.index = pd.to_datetime(drag.index, format='%Y-%m-%d')
        # Create a visualization
        sns.relplot(data=drag["Drag force without ice (N)"], kind="line")
        plt.show()
        print('Press any key to go back to the plot menu')
        input()
        plotopt()
    if int(input()) == 4: # Stresses caused by the thermal expansion
        # Import data
        daymaxt = pd.read_csv(r'./Data/DataFrames/daymaxt.csv', index_col=0)
        daymaxt.index.names = ["Date"]
        daymaxt.index = pd.to_datetime(daymaxt.index, format='%Y-%m-%d')
        daymaxt.reset_index(level=0, inplace=True)
        daymaxt=daymaxt.melt(id_vars=['Date','mintemp (C)','maxtemp
(C)','increaseLength (mm)','increaseWidth (mm)','increaseLengthperJoint
(mm)','increaseWidthperJoint (mm)' ], var_name='Stress' ,value_name='(GPa)')
        # Create a visualization
        sns.relplot(x = 'Date', y = '(GPa)', hue='Stress', data = daymaxt,
kind="line")
        plt.show()
        print('Press any key to go back to the plot menu')
        input()
        plotopt()
    if int(input()) == 5: # Maximum extension of each joint
        # Import data
        daymaxt = pd.read_csv(r'./Data/DataFrames/daymaxt.csv', index_col=0)
        daymaxt.index.names = ["Date"]
        daymaxt.index = pd.to_datetime(daymaxt.index, format='%Y-%m-%d')
        daymaxt.reset_index(level=0, inplace=True)
        daymaxt = daymaxt.melt(id_vars=['Date', 'mintemp (C)', 'maxtemp (C)',
'increaseLength (mm)', 'increaseWidth (mm)','EquivalentStressL (GPa)',
'EquivalentStressW (GPa)'], var_name='Deformation',value_name='(mm)')
        # Create a visualization
        sns.relplot(x = 'Date', y = '(mm)', hue='Deformation', data =
daymaxt, kind="line")
        plt.show()
        print('Press any key to go back to the plot menu')
        input()
        plotopt()
    else:
        print('Exiting')

''' # Code for drag comparison
drag = pd.read_csv(r'./Data/DataFrames/drag.csv', index_col=0, decimal=',')
drag.index.names = ["Date"]
drag.index = pd.to_datetime(drag.index, format='%Y-%m-%d')
drag.reset_index(level=0, inplace=True)
drag = drag.melt(id_vars=['Date', 'wSpeed (m/s)', 'Drag force without ice
(N)'], var_name='Forces', value_name='(N)')
sns.relplot(x='Date', y='(N)', hue='Forces', data=drag, kind="line")'''
```