Faculty of Sciences

Department of Astrophysics, Geophysics and Oceanography

Impact of the Presence of Surface Meltwater Lakes on the Greenland Ice Sheet Surface Mass Balance with the Help of the Regional Climate Model MAR

Thesis submitted for the title of
Master in Space Sciences, Research Focus

Academic year 2023 - 2024

Author:

Guillaume TIMMERMANS

Promoter:

Xavier FETTWEIS

Acknowledgements (in French)

Premièrement, je tiens à remercier grandement mon promoteur, Xavier Fettweis, pour son dévouement, sa disponibilité, ses conseils et son expertise dont j'ai pu bénéficier tout au long de ce travail.

Ensuite, je remercie ma famille pour son soutien inconditionnel au cours de mon parcours académique. Je ne serais jamais arrivé ici sans eux. Je suis très chanceux de vous avoir à mes côtés.

Un tout grand merci également à tante Colette pour son accueil ! C'est à Florenville que la Section 4.2.3 a vu le jour :-).

Je dois également remercier les personnes qui ont été à mes côtés ces dernières années, notamment Elise, Gabriel, Lucas, Martin, Sarah, Simon et Vincent. Les membres du "FC Spartiates" et de l'"Ajax de Cointe", Farkas, Wapiti et tous les pirates de la IVe Legia. Je n'oublie pas **L**es créat**U**res ensor**C**ellées et m**Y**stiques du B5 : Sacha, Cédric, Antoine et Adrien, ni, bien sûr, Clémence, Victor, Thomas, Lara, Robin, Hugues et toutes les personnes que j'ai eu la chance de rencontrer en géographie.

Life is what happens to you while you are busy making other plans.

John Lennon

With every mistake, we must surely be learning.

George Harrison

I feel the older I get, the more I'm learning to handle life. Being on this quest for a long time, it's all about finding yourself.

Ringo Starr

In the end, the love you take is equal to the love you make.

Paul McCartney

Executive Summary

Abstract

With global warming, supra-glacial lakes (SGLs) are increasingly widespread on the Greenland Ice Sheet (GrIS) due to enhanced meltwater runoff and ablation zone extension. However, SGLs as well as all associated hydrology are currently neglected by models simulating the evolution of the Surface Mass Balance (SMB) of the ice sheets. These models assume that meltwater runoff is directly transferred to the ocean as soon as the snowpack is liquid water-saturated, regardless of the location of this meltwater on the ice sheet or topography. Nevertheless, the presence of SGLs on the surface of the ice influences the SMB. Notably by its predisposition to retain liquid water (that could refreeze in winter) or to generate extra evaporation in summer or by its low surface albedo, locally increasing the absorption of solar energy.

To evaluate the importance of SGLs as a potentially significant SMB component, we ran MARv3.14 at high resolution (5km×5km) over South-West Greenland for the 2018-2019 hydrological year. We conducted sensitivity experiments to examine the effects of this surface water on bare ice albedo, evaporation, meltwater retention, and atmospheric feedback. Surface meltwater in the ablation zone was allowed to remain liquid or solid in bare ice areas where SGLs were detected by satellite. Additionally, we carried out a quantitative study of SGLs in Greenland using primarily LANDSAT 8 data.

We conclude that modifying the runoff parameters in the Modèle Atmosphérique Régional (MAR) model can significantly alter the modelled SMB. Also, improving the treatment of runoff in the model is essential for better quantifying the impact of supraglacial hydrology on the SMB.

Outline

In Chapter 1, we present the Greenland ice sheet and the mass balance concept, which is at the heart of this document. We also discuss other topics such as the methods allowing to estimate the GrIS mass balance and global warming, including the fact that Greenland is one of the regions where this phenomenon is most clearly observed. We also recall the supra-glacial hydrology.

After that, Chapter 2 presents our research question and gives a brief state-of-the-art of this research topic, which, has not been extensively studied to date.

In Chapter 3, we use remote sensing to learn more about hydrology and SGLs in Greenland. This part is notably relevant because it allows us to know better the processes that will be modelled. We study the location of SGLs and create a dataset that is used in the following chapter.

Finally, the major part of this report, which focuses on modelling the SMB of the GrIS, is presented in Chapter 4. We present the MAR model and study the impact of surface water treatment in it on the modelled SMB of the ice cap. To this end, we conducted a large number of numerical simulations with the MAR. We explain our modifications in the model and present and interpret the results.

Chapter 5 concludes the report.

Contents

List of Abbreviations

CART Classification and Regression Tree

CFL Courant–Friedrichs–Lewy

ECMWF European Centre for Medium-Range Weather Forecasts

GCM Global Climate Model

GEE Google Earth Engine

GFZ German Research Centre for Geosciences

GHGs Greenhouse Gases

GIA Glacial Isostatic Adjustment

GRACE Gravity Recovery and Climate Experiment

GrIS Greenland Ice Sheet

IDW Inverse Distance Weighting

IMBIE Ice Sheet Mass Balance Inter-comparison Exercise

IPCC Intergovernmental Panel on Climate Change

LASSO Least Absolute Shrinkage and Selection Operator

LHF Latent Heat Flux

MAR Modèle Atmosphérique Régional

MB Mass Balance

mmwe milimeters water equivalent

NDWI Normalized Difference Water Index

NetCDF Network Common Data Format

NCEP/NCAR National Centers for Environmental Prediction (NCEP) and the National Center for Atmospheric Research (NCAR)

NIR Near-Infrared

OLI Operational Land Imager

Polar MM Polar Mesoscale Model

QGIS Quantum Geographic Information System

RACMO Regional Atmosphere Climate Model

RANS Reynolds-averaged Navier–Stokes

RF Rainfall

RCM Regional Climate Model

RGB Red Green Blue

RU Runoff (of meltwater and rain)

SBL Surface Boundary Layer

SEB Surface Energy Budget

SF Snowfall

SGL supra-glacial lake

SHED Supra-glacial Hydrology Evolution and Drainage

SISVAT Soil Ice Snow Vegetation Atmosphere Transfer

SMB Surface Mass Balance

SROC Special Report on the Ocean and Cryosphere in a Changing Climate

SU Sublimation (from ice and snow)

SW Surface Water

SWH Surface Water Height

SWIR Short-Wave Infra-red

TSE Total Square Error

WGS 84 World Geodetic System 1984

Chapter 1

Introduction & Scientific Context of the Study

In this chapter, we present the scientific context in which our study is situated. This serves as a prerequisite for understanding the subsequent chapters, particularly Chapter 4. We present the GrIS and discuss the concept of Mass Balance (MB). We delve into this topic in depth. After presenting certain methods for estimating the MB (note that we will report the presentation of MAR to Chapter 4), we briefly address the causes and consequences of its imbalance. We conclude this chapter by addressing the subject of supra-glacial hydrology.

At the end of this introduction to the issues examined in this thesis, we will be able to present our research question, in the following chapter (Chapter 2), where we also conduct a brief state-of-the-art review of the main subject of this work: the modelling of (supra-) glacial hydrology, and its impact on the SMB.

1.1 Presentation and Climate of Greenland and its Ice Sheet

Greenland is a territory located between the Atlantic and Arctic oceans. It is nearly fully covered by an ice sheet that rests on a continental base. During winter, sea ice forms in the oceans surrounding Greenland. The GrIS exerts a strong influence on the whole Earth System. Indeed, it contributes to maintaining a relatively "cold" climate on Earth[1], due to its albedo, plus the fact that the frozen tundra stores a huge amount of CO_2. Also, it influences the oceanic circulation as well as ecosystems, as it brings relatively cold freshwater to the ocean (Cassotta et al., 2022). It should also be noted that glaciers have a high thermal inertia and that the accumulation of snow and ice causes considerable changes in the relief (François and Munhoven, 2023). It is the second most important freshwater reservoir on Earth. As mentioned later (Section 1.2.5), the melt of the GrIS could increase considerably the sea level, hence the interest in understanding and modelling the evolution of the amount of water it stores.

A considerable part of Greenland's territory lies beyond the Arctic Circle, which implies that polar nights occur there. Because of the latitude, and the altitudes exceeding 3000 meters in places, the temperatures reach particularly cold values. On average, it is almost negative everywhere, as shown in Figure 1.1 (a) (it is based on data from 2019, a relatively "warm" year, but it is still suitable for explanation, especially since we will mainly focus on that year later in the document). Let us mention that there exists a well-known relationship between the temperature and the altitude in climatology. Indeed, Figure 1.1 (b) shows that the evolution of the temperature as a function of the altitude follows roughly the adiabatic lapse rate, given by Eq. (1.1). We also notice that the temperature and the latitude are anti-correlated.

$$\frac{dT}{dz} = -\Gamma \approx -6\text{K/km} \tag{1.1}$$

This result likely did not surprise most readers. Indeed, it can be explained by physics, as detailed in the derivation found in Appendix B.

[1]Geology teaches us that the current period, the Quaternary, which is considered to have been ongoing for 2.58 million years, can be described as a "cold period" when compared to the rest of the Phanerozoic Eon (the last 541 million years).

As this thesis focuses on some aspects of the interaction between ice and meltwater, we will focus on the less cold areas of Greenland (where the melt is more important), which are located in the periphery. This is because the surface elevation increases inland.

1.2 Mass Balance of the Greenland Ice Sheet

Let us now present what the mass balance of an ice sheet is, focusing exclusively on the Greenland Ice Sheet.

1.2.1 The Concept of Mass Balance

The amount of water (mainly as ice) in the Greenland ice sheet changes with the mass balance. This budget is the net result of mass gains (mainly precipitation (P)) and losses of the ice sheet (Hanna et al., 2013) over a given time interval, usually a year. If so, one can speak of a "mass balance year". Mass losses occur through several processes as shown in Eqs. (1.2) & (1.3) (Fettweis et al., 2020). This budget can be divided into a SMB and an ice dynamics budget (Cassotta et al., 2022). The ice dynamics budget refers to ice calving while SMB is the outcome of all the surface processes (involved in Equation 1.3). Of course, the interest of the concept of mass balance is to evaluate its evolution over time to see how much the glacier gains or loses mass.

$$\mathrm{MB} = \mathrm{SMB} - \mathrm{D} \tag{1.2}$$

$$\mathrm{SMB} = \mathrm{P} - \mathrm{RU} - \mathrm{SU} - \mathrm{ER} \tag{1.3}$$

Figure 1.2 illustrates Eqs. (1.2) & (1.3), it summarises the processes controlling the MB, and the interactions between the ice sheet, the ocean and the atmosphere. Mass loss mainly occurs at the edges of the GrIS through discharge and a negative cumulative SMB. One important thing regarding ice discharge is that the ice cap does not reach the ocean on its entire contour in Greenland, an area of tundra separates it in places (see e.g. Figure 1.2 of Fettweis (2006)). The diagram also shows that the meltwater flows from where it melted to the ocean, although as we will see later (in Section 1.3), the process is not so straightforward because this water can accumulate in SGL, infiltrate, refreeze, *etc.* The simple diagram presented in Figure 1.3 is very relevant to the extent that it explains why we do not consider melting and refreezing as SMB fluxes. It would have been necessary to consider those if we defined the SMB as a snow and ice budget instead of a water (*i.e.* ice, snow and liquid water) budget.

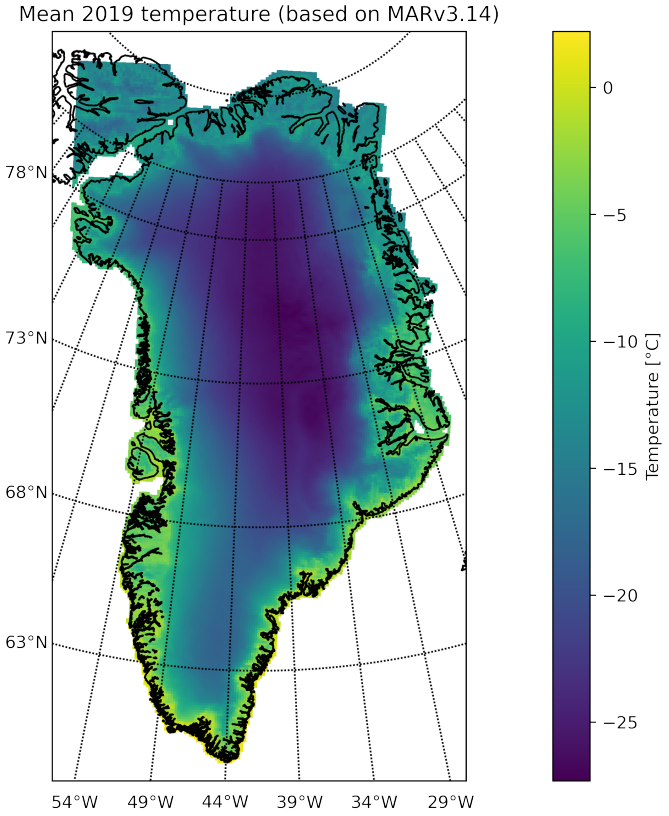
All terms appearing in Eq. (1.2) and (1.3) are explained in Table 1.1 and are expressed in units of mass per unit of time (if the budget is integrated over the whole GrIS). Therefore, if we want to evaluate the amount of mass transported by a flux over a given period, we must add the term "cumulative" (for example, the "cumulative SMB over three months"). We will sometimes omit this term if the context makes it clear.

Also, Let us note that the precipitations (P) can be either Snowfall (SF) or Rainfall (RF) (*cf.* Eq. (1.4). We refer to van den Broeke et al. (2017) for further details.

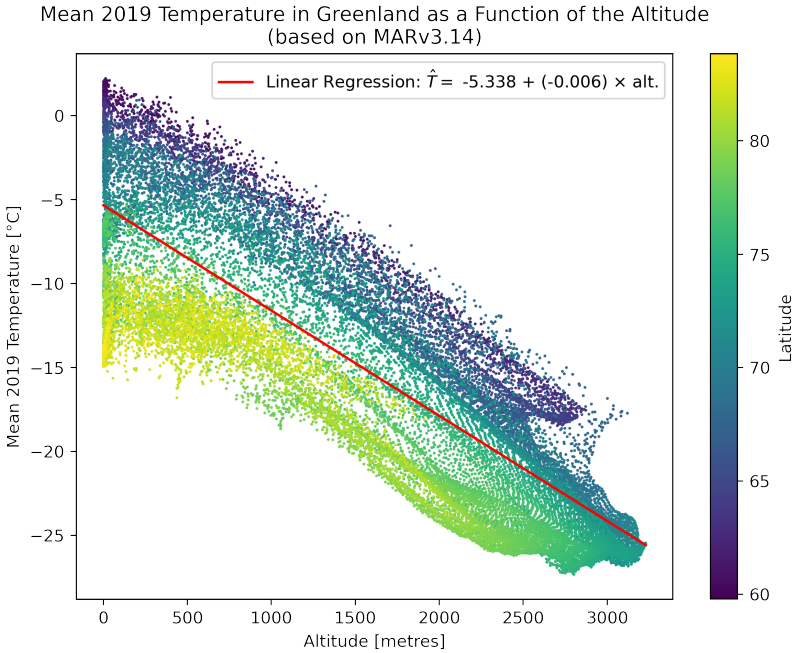
$$\mathrm{P} = \mathrm{RF} + \mathrm{SF} \tag{1.4}$$

Table 1.1: Terms involved in the mass balance

MB	Mass Balance
SMB	Surface Mass Balance
D	Ice Discharge
P	Precipitations
RU	Runoff
SU	Sublimation/Evaporation
ER	Snow Erosion by the wind



(a) Mean 2019 temperature based on MARv3.14 forced itself by ERA-5 reanalysis. (Source of the data: Fettweis (2023)).



(b) Mean 2019 temperature as a function of the altitude. The colour of the points is a function of their latitude. Each point represents a MAR pixel. It is based on the same data as in (a). The regressed (using a least-squares adjustment) intercept and slope are respectively -5.33786°C and -0.00626°C/m.

Figure 1.1: Display and analysis of the mean 2019 temperature in Greenland, as simulated by MARv3.14 forced by ERA-5 reanalysis.

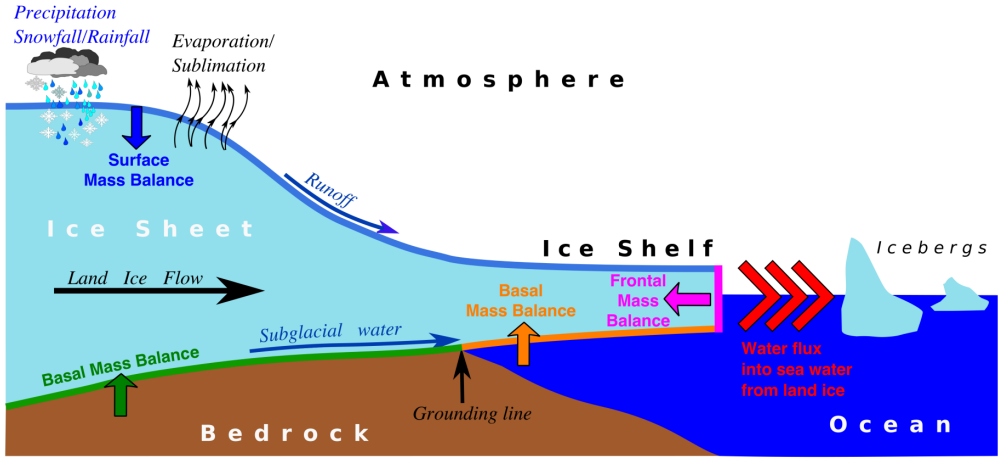


Figure 1.2: Illustration of the processes involved in the MB. Source: Nowicki et al. (2016).

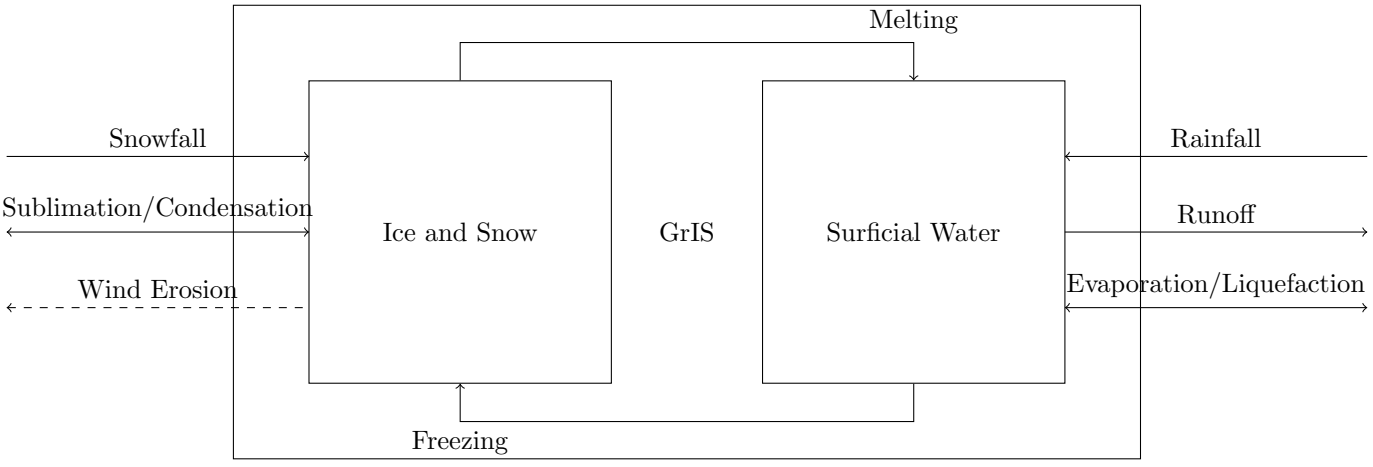


Figure 1.3: Illustration of the fluxes that influence the SMB. This diagram do not represent water in pores or under ice.

Of course, the SMB depends on the Surface Energy Budget (SEB), as it controls the melt, the Sublimation (from ice and snow) (SU), the refreezing and the condensation. This energy budget is given in Eq. (1.5), which is adapted from van den Broeke et al. (2017).

$$\mathrm{M} = \mathrm{SW}_{in} - \mathrm{SW}_{out} + \mathrm{LW}_{in} - \mathrm{LW}_{out} + \mathrm{SHF} + \mathrm{LHF} + \mathrm{G}_s = \mathrm{SW}_{in}(1-\alpha) + \mathrm{LW}_{in} - \sigma T_s^4 + \mathrm{SHF} + \mathrm{LHF} + \mathrm{G}_s \quad (1.5)$$

Where M is the surface energy budget[2], SW and LW denote the short-wave and long-wave radiation fluxes, SHF and LHF are the sensible and latent heat fluxes, G_s is the subsurface conductive heat flux (positive upwards), α is the surface albedo, σ is the Stefan-Bolzmann constant, and T_s is the surface temperature.

Relative Importance of Surface Mass Balance (SMB) Terms

Of course, all the terms involved in the SMB equation (Eq. (1.3)) do not all have the same importance, some can even be neglected. This is shown in Figure 1.4. The data come from a MAR simulation over the whole GrIS (Fettweis, 2023). We represent the mean of several SMB fluxes over all pixels of the model, so all quantities represented are expressed in units of mass per unit of time, per unit of surface. In Figure 1.4, we also represent the Surface Water (SW) flux, even if it is not involved in Eq. (1.3), as it will interest us in the following. SW corresponds to the budget of surficial water over a given period (*i.e.* the outcome of the surficial water fluxes, so melt, runoff, rain, *etc.*). We notice that loss mainly (if not only) occurs in summer (*i.e.* roughly from the beginning of June to the end of September) through the Runoff (of meltwater and rain) (RU) flux, it is also the period during which there is the fluxes of surficial water are the most important. Detailed monitoring of the GrIS SMB is available in Fettweis (2023).

1.2.2 The Mass Balance Zones

As mentioned above, the SMB is the result of all surface processes. It is governed by the interaction between the atmosphere and the surface (Fettweis, 2006). Based on whether the SMB is positive or negative, one can divide the ice sheet into two zones: the accumulation zone (where SMB > 0) and the ablation zone (where SMB < 0). The equilibrium line separates these (see e.g. Figure 1.1 of Fettweis (2006)). As an illustration, the ablation and the accumulation zones for the mean annual SMB over 30 years are presented in Figure 1.5 (a). Figure 1.5 (b) represents the (cumulative) SMB for 2019.

1.2.3 Recent Evolution of the Greenland Ice Sheet Mass Balance

The mass balance was in equilibrium until the late 1990s when Greenland contributed only negligibly to sea level rise (Hanna et al., 2013). Since then, the evolution of the mass balance has been negative, with the ice sheet thinning with the abnormal temperatures of recent decades (see Cassotta et al. (2022)). Surface processes ($\sim 50.3\%$) and dynamical processes ($\sim 49.7\%$) contribute approximately equally to mass loss (IMBIE, 2020). Between 1992 and 2018, the Ice Sheet Mass Balance Inter-comparison Exercise (IMBIE) team estimates the mass loss due to SMB variations at 1,964 $\pm$ 565 Gt, the remaining part is due to ice calving (IMBIE, 2020). The proportions due to these contributions are variable, but the proportion of the loss by ice calving is expected to decrease in the future due to glacier retreat (see e.g. Enderlin et al. (2014), Goelzer et al. (2013)).

Monitoring of the Ice Sheet

Let us now turn our attention to the methods used to (indirectly) measure the mass balance.

As explained in Otosaka et al. (2022), the launch of satellites has enabled more effective monitoring and enhanced our understanding of the Greenland (and Antarctica) ice sheets. Satellite observations are used in parallel with geophysical models of Glacial Isostatic Adjustment (GIA). The three main satellite-based techniques for monitoring the ice sheet are:

1. Observation of ice flow derived from satellite radar and optical imagery,
2. Surface elevation changes measurements,
3. Fluctuations in Earth's gravity field detection.

[2]van den Broeke et al. (2017) refers to it as the "melt energy", but M can also be linked with refreezing or simply the variation of the surface temperature.

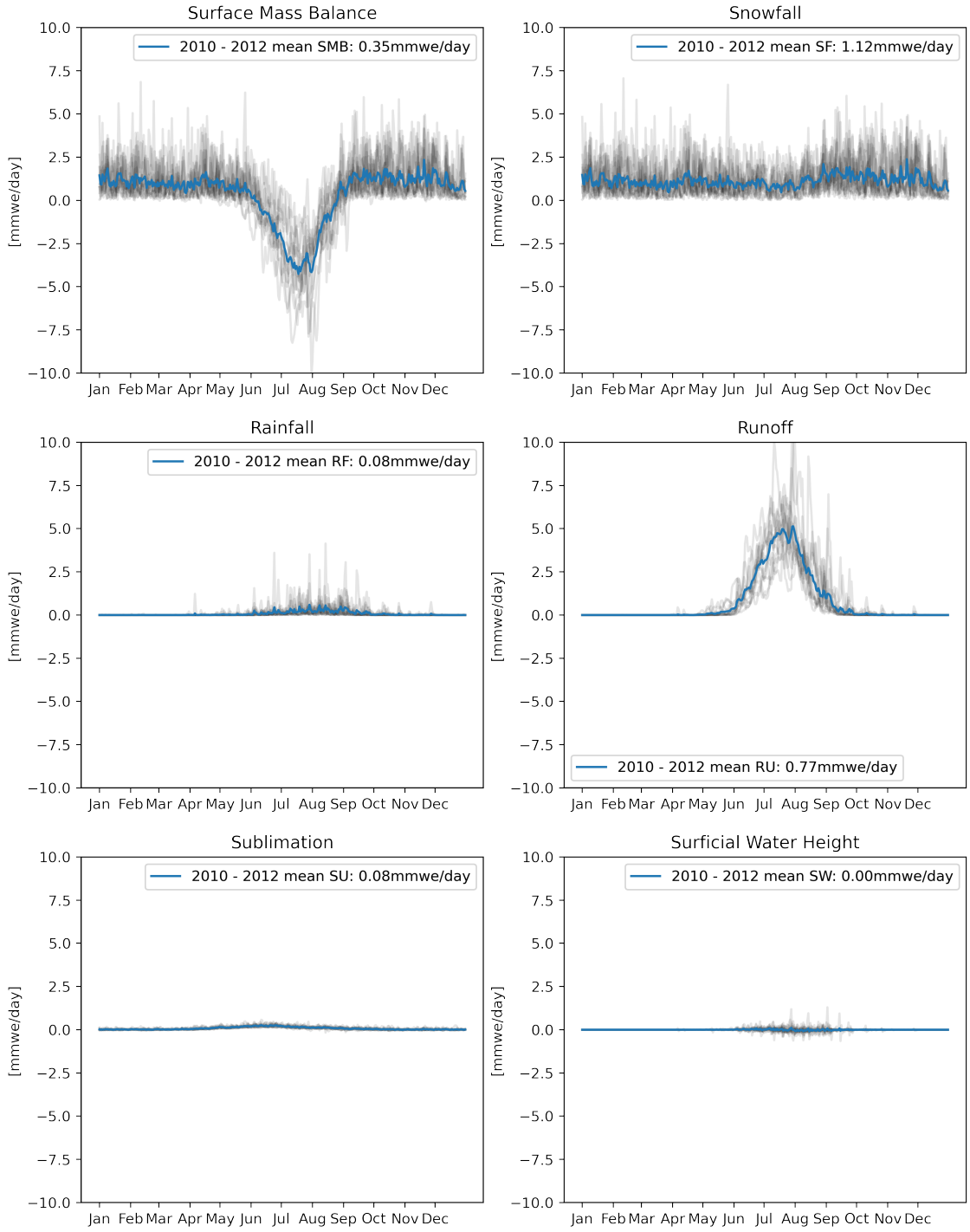
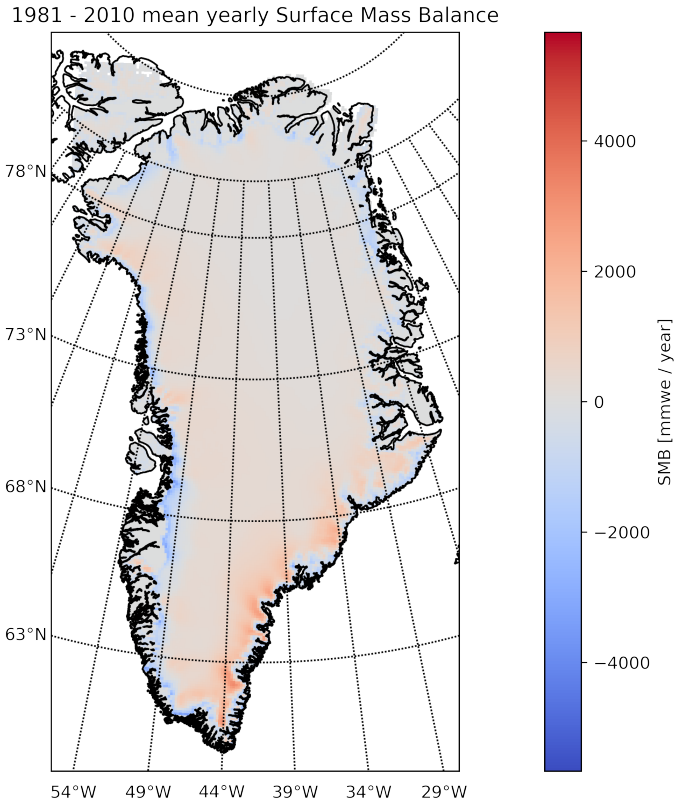
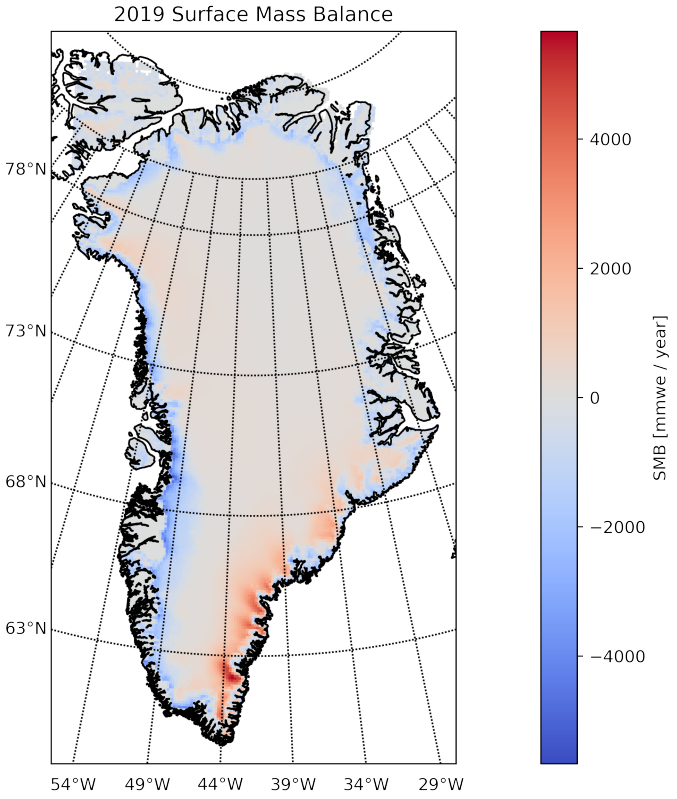


Figure 1.4: 2010 - 2022 evolution of the different terms involved in the SMB (or linked to it). Each grey line corresponds to one year between 2010 and 2022. It represents the yearly evolution of the mean (over the GrIS surface) of a given daily mass flux. Blue lines are simply the mean of the grey lines, so a mean over twelve years. In the legend of each plot, the number is the yearly mean of the blue curve, rounded to 2 decimals. All quantities are expressed in milimeters water equivalent (mmwe) per day. So multiplying it by the area of the GrIS converts the flux in terms of volume (or mass) of water per day, and integrating over time (or multiplying the daily mean by $\sim$ 365,25) leads to the mass of water that a given flux represents, on average, in terms of gain or loss for the ice sheet over one year. The represented fluxes are: SMB, SF, RF, RU, SU, and SW (which is the net flux of surficial water). These results come from a MAR simulation over the whole GrIS forced by ERA-5 reanalysis. Source: Fettweis (2023).



(a) 1981 - 2010 mean annual SMB simulated by MARv3.14, forced by ERA-5 reanalysis. Source: Fettweis (2023).



(b) 2019 cumulative SMB simulated by MARv3.14, forced by ERA-5 reanalysis. Source: Fettweis (2023).

Figure 1.5: GrIS SMB simulated over two different periods. The zone where the yearly SMB is negative corresponds to the ablation zone. The accumulation zone is where the SMB is positive. The "boundary" between these zones is the equilibrium line.

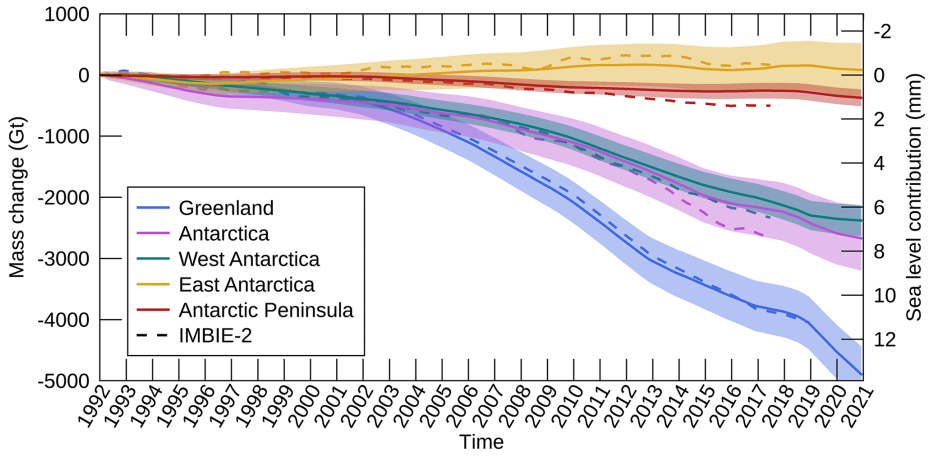


Figure 1.6: Cumulative mass changes and sea level increase contribution of the ice sheets between 1992 and 2020 obtained by combining satellite data. The shading corresponds to the 1 σ uncertainty computed as in the original publication. Dashed lines correspond to the results of IMBIE (2020) and IMBIE (2018). Source: Otosaka et al. (2022).

These three techniques are briefly presented below.

Observation of ice flow Mouginot et al. (2017) present a method to process satellite images in order to assess the velocity of the ice flow in Greenland and Antarctica. Without going into details, the technique consists in the observation of the variations of positions of features on the surface of the ice. Initially (in the 1990s), the technique was limited to the edges of the ice sheet where crevasses can be observed. Afterwards, the increase in precision of the satellites over time has made it possible to observe more subtle details and to extend the technique to the whole inlandsis. This technique (often called the input-output method) should be supplemented by the use of SMB models to account for surface processes that influence the mass balance.

Surface elevation changes measurements This technique relies on radar or laser altimetry (from satellite). It is necessary to apply a correction for GIA even if its influence is weak (Otosaka et al., 2022). Let us note that as explained in Pritchard et al. (2009), satellite measurements are spaced up to a hundred meters from one passage to another. Consequently, it is necessary to interpolate the slope of the surface of the ice sheet in order to determine the evolution of its thickness.

Measure of Earth's gravity field detection The Gravity Recovery and Climate Experiment (GRACE) and GRACE-FO missions, conducted in collaboration between NASA and the German Research Centre for Geosciences (GFZ), provide information on the Earth's gravity field that can be linked to the distribution of mass on the planet (Sasgen et al., 2020). This makes it possible to assess the evolution of the ice mass in Greenland. Like the surface elevation technique, a correction must be applied for GIA.

Observation of the Ice Sheet State

The IMBIE team published several reports where they summarise and gather the results obtained using the different satellite methods presented above (in addition to other methods, including SMB modelling). Figure 1.6 is retrieved from their latest report. It shows that the Greenland Ice Sheet has lost about 4000 Gt of water from 1992 to 2020, at an increasing rate (Otosaka et al., 2022).

Modelling of the Surface Mass Balance

Of course, since there are very few direct measurements, models play a key role in the monitoring of the GrIS state, in addition to their projections. As this is one of the key points of this report, we dedicate a section (Section 4.1) to explore this dimension further later in the document.

1.2.4 Drivers of Greenland Mass Balance Disequilibrium

Let us now examine the factors contributing to the imbalance in the mass balance of the GrIS. As noted earlier, the SEB plays a crucial role in controlling melt rates (and refreezing, *etc.*). In the context of climate change, disruptions in the SEB lead to the destabilisation of the mass balance. This section will explore this in greater detail.

As outlined in the Intergovernmental Panel on Climate Change (IPCC) Special Report on the Ocean and Cryosphere in a Changing Climate (SROC) report (Cassotta et al., 2022), the principal factors contributing to the thinning of both the Greenland and Antarctica ice sheets are of oceanic and atmospheric origin. The warming of the North Atlantic Ocean increases the calving rate, while the atmospheric properties such as precipitations, temperature and cloud cover over the ice sheet have a crucial influence on surface mass balance (SMB). Of course, there is no doubt that anthropogenic Greenhouse Gases (GHGs) emissions are responsible for the MB decrease. For example, even if it is about sea ice, it is worth mentioning that Notz and Stroeve (2016) observe a linear relationship between the loss of sea-ice area and the CO_2 emissions.

In recent decades, Greenland's climate has warmed more than the global average ("double the rate of the global mean air temperature increase", according to Richter-Menge et al. (2017)), resulting in increased melting, making the Greenland ice sheet the largest contributor to rising sea levels. As explained in Cassotta et al. (2022), several mechanisms are thought to be responsible for that "localised" temperature increase. These imply the role of surface albedo, increased water vapour content in the atmosphere, increased cloudiness, etc. Furthermore, the Arctic climate experienced several "extremes" events, notably including the summer 2019 event (Tedesco and Fettweis, 2020). With some caution and critical thinking about the data, we can quantify this with the HadCrut5 dataset (Morice et al., 2021). Figure 1.7 shows for each pixel (of the HadCrut5 grid), the scaled slope b_s of the linear trend of the temperature. b_s is computed as in Eq. (1.6).

$$b_s(\phi, \lambda) = \frac{b(\phi, \lambda) - \bar{b}}{\sigma_b} \tag{1.6}$$

Where b is the slope of the least square-adjusted regression between time and temperature (*i.e.* a model of the form, "$\hat{T} = a + b \times t$"that minimizes the Total Square Error (TSE)), $\bar{b}$ and σ_b are respectively the mean and the standard deviations of b over all pixels. ϕ and λ are respectively the latitude and the longitude. We observe that, indeed, the Arctic region, including Greenland is warming quite faster than the average[3]. As a complement, Figure 1.8 shows the corresponding slope, b for the 1980-2023[4] period. This figure is interesting because it illustrates that, more or less everywhere on Earth, the trend is towards increasing temperatures, including places where the "scaled slope" is negative.

1.2.5 Consequences of the Greenland Ice Sheet (GrIS) Surface Mass Balance (SMB) Disiquilibrium

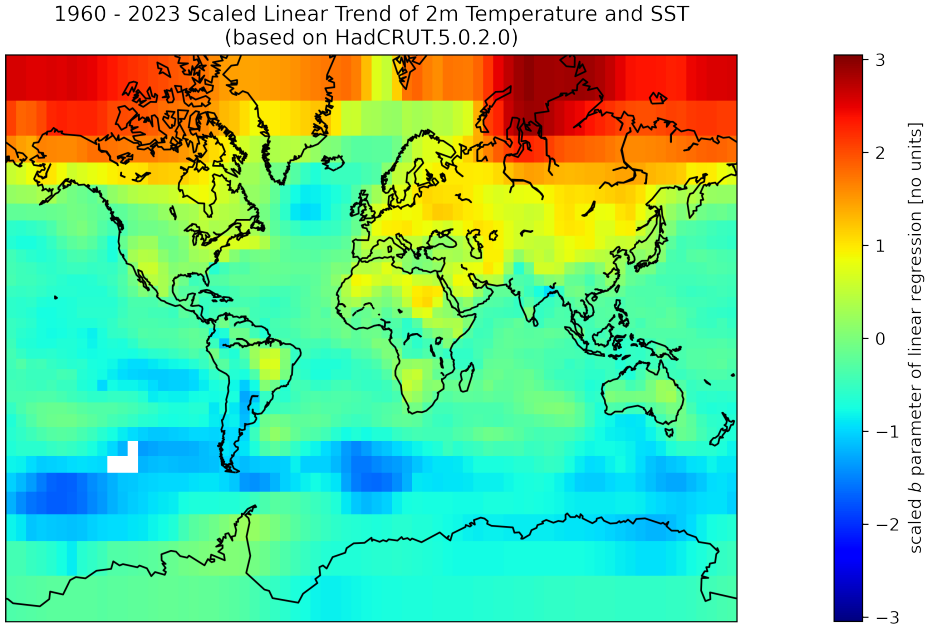
Cassotta et al. (2022) summarise the current and potential impacts of Greenland melting (as well as the melting of other components of the cryosphere). This has and will have consequences on sea level, physical oceanography, biogeochemistry and ecosystems. Of course, one of the main consequences of the melting of the Greenland Ice Sheet is the rise in the global sea level. The sea level rise resulting from the total melting of the ice sheet would be about 7.42 ± 0.05 m (Morlighem et al., 2017). This value is calculated by assessing the topography and bathymetry beneath the Greenland ice sheet, a process achieved through various techniques. Let us note that Cassotta et al. (2022) develop this subject in depth.

1.3 Overview of the Greenland Ice Sheet (GrIS) Hydrology

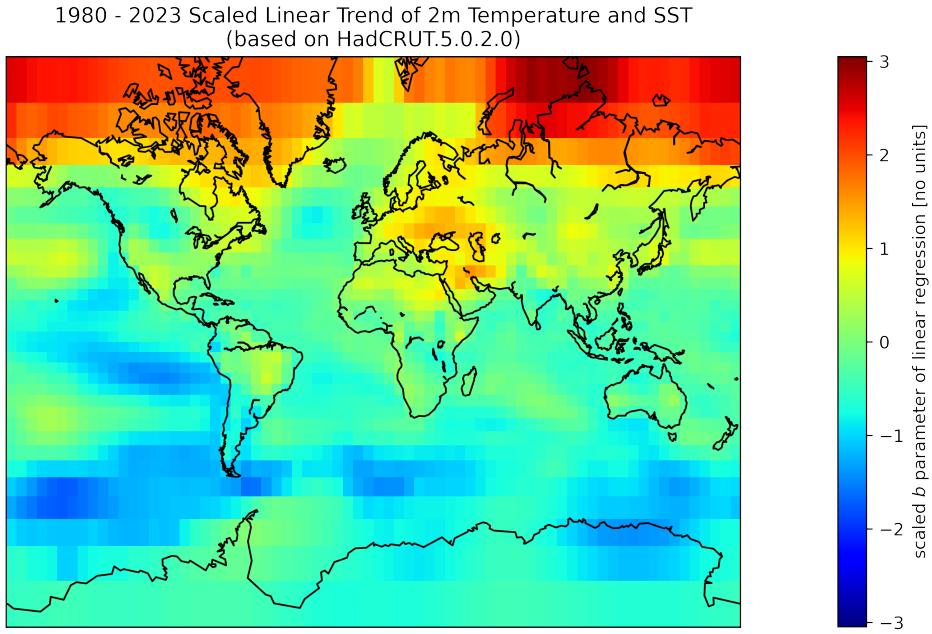
As shown in Figure 1.4, the runoff of meltwater to the ocean is among the most important terms of the SMB. Nevertheless, it is not a straightforward process. In other words, when the ice or snow melts, it can, depending on the slope, the wind, the surface (the flow is faster on ice than on snow, which has a "retention capacity"), *etc.*,

[3]Actually, as we base ourselves on the same data as Richter-Menge et al. (2017), this does not constitute independent "additional evidence". However, other sources from the literature agree with this (see Cassotta et al. (2022)).

[4]The times series presented here (Figures 1.7 & 1.8) are from January to January, so these end in January 2024, but we wrote 2023 as it does not include that much data from 2024.



(a) 1960 - 2023 Scaled Linear Trend of 2 metres Temperature and SST (based on HadCRUT.5.0.2.0). We do not represent pixels where the time series starts after 1960 (in the South Pacific Ocean).



(b) 1980 - 2023 Scaled Linear Trend of 2 metres Temperature and SST (based on HadCRUT.5.0.2.0).

Figure 1.7: Comparison of the temperature evolution trend between the world regions.

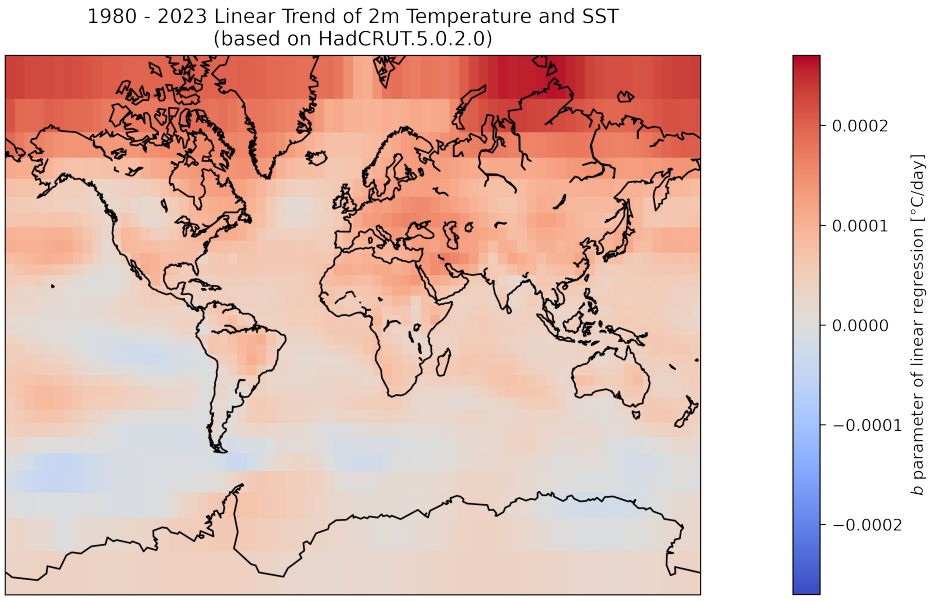


Figure 1.8: Slope of the linear regression between time and temperature for the 1980-2023 period (based on HadCRUT.5.0.2.0).

flow or stagnate it can also evaporate, refreeze or stay liquid. This water can also percolate in the snow mantle. This section aims to provide an overview of supraglacial hydrology with a focus on SGL and some elements of sub-glacial hydrology. These processes are not well known because they occur in hard-to-reach environments, and their modelling is far from satisfactory. Let us note that the book from Jansson et al. (2007) deals in depth with this topic.

In situ measurements (see Smith et al., 2017; Mankoff et al., 2020; Clerx et al., 2022) as well as remote sensing and mainly satellite data-based studies (e.g. Leeson et al., 2015; Miller et al., 2020) highlighted and quantified several hydrologic features developing in response to the GrIS melting. Those processes, that mainly occur in the ablation zone of the GrIS, are represented in Figure 1.9. Among those, the spread of SGL strongly influences the whole GrIS hydrological system. It is not clear whether it contributes to lowering or increasing the SMB as on the one hand, it triggers a decrease of the albedo, hence the temperature of the surrounding ice and the melting. Still, on the other hand, it allows meltwater to refreeze. This effect is thought to be the most important since water on the surface affects the surface albedo, regardless of its movement, but lakes increase the chance of meltwater refreezing. Furthermore, since water is denser than ice and snow, an SGL can trigger subsurface channel drainage, leading to the formation of underground water pockets, or bringing water into contact with bedrock (Dow et al., 2015), making ice movements more fluid, accelerating the dynamics of the ice cap. However, uncertainties remain as to the consequences of this process. SGL are "interconnected by surface drainage" and are stable in time (Jansson et al., 2007).

As explained in Jansson et al. (2007), in addition to surface drainage channels and lakes, water can flow through crevasses and moulins, from the surface to sometimes reach the bedrock.

The subject is currently being researched. Hydraulics can explain the movement of water, even if the hydrofracturing of the ice, as well as phase changes of water, remain difficult to model. As explained in the following, the challenge is to quantify the effect of these small-scale processes on the SMB, which is calculated at the scale of the ice cap. We can do parallelism with the use of Reynolds-averaged Navier–Stokes (RANS) in fluid mechanics applications when one is interested in the mean flow instead and not its fluctuations, and cannot afford to solve the equations at a very fine scale.

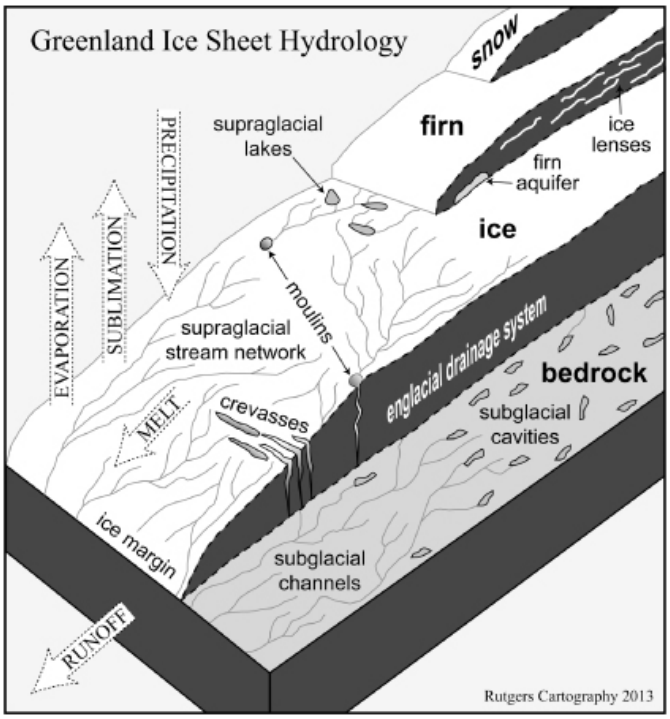


Figure 1.9: Scheme of a cross-section showing the various hydrological features developing at the edges of the ice sheet. SMB terms are represented by dashed arrows (even if we do not consider the melts as a SMB term in this thesis since it is contained in the runoff, as mentioned above). Source: Rennermalm et al. (2013).

Chapter 2

Research Question

2.1 Motivations and Objectives of this Thesis

After presenting the context in which this work takes place, let us now turn to the objectives pursued by this thesis.

We have seen that the mass balance can be decomposed into SMB and "discharge" components. Regional Climate Models (RCMs), particularly the MAR, allow for the reconstruction and projection of the SMB evolution of the GrIS, which is of interest for estimating sea level rise (based on "input" data, *i.e.*, reanalyses or results from global climate models, which are themselves forced by "socio-economic pathways" (Riahi et al., 2017)). We have also seen that several hydrological processes involving meltwater (and rainwater) take place in the ablation zone of the GrIS. Although, as we will see later (Section 4.1.1), models such as MAR simulate the physics associated with the ice sheet quite accurately, surface hydrological processes are not satisfactorily accounted for in the current model(s), they are even totally neglected.

In this context, we can raise our main research question:

What is the impact of surface hydrology, and supra-glacial lakes in particular, on the SMB of the GrIS?

Indeed, it is likely that these processes play a role in the SMB. Since we mentioned earlier that the runoff (RU) is the most significant loss term in the SMB (simulated by MAR). However, the runoff is influenced by all the hydrological processes previously discussed. Therefore, we might reasonably question whether a significant portion of meltwater is "trapped" in SGL and constrained to refreeze in winter, for example. If this refreezing process driven by the presence of lakes is very significant, future projections of SMB evolution may be affected by errors due to this factor.

In this work, we will extensively use the MAR, which is developed at the Laboratory of Climatology of the University of Liège. The main objective of this thesis is therefore to parameterise the MAR in order to address this question in terms of order of magnitude, or at least to develop methods and provide some answers, as our question is quite ambitious.

Work Plan

The primary task of this work is to quantify supra-glacial hydrological processes in Greenland, specifically within a study area where we conduct numerical simulations of the SMB using MAR. To achieve this, we rely on satellite remote sensing data, creating a dataset to account for the presence of lakes in MAR. Subsequently, we run MARv3.14 at high resolution over the South-West of the GrIS. We analyse the model's treatment of surface water. We also use the previously created dataset to adjust the model's behaviour based on the lake areas observed by satellite.

2.2 Brief State of the Art in Surface Hydrology Modelling

Now that we have introduced the basic concepts and the context in which this work takes place, as well as defined the objectives pursued, let us turn our attention to the literature and the work already done on the subject.

RCMs constitute a crucial tool for studying the evolution of the SMB of the GrIS. However, none of the existing RCMs explicitly represent surface hydrology. In these models, each pixel of the surface is an independent vertical column comprising several layers of snow or ice of varying thickness that can interact with the atmosphere without interacting with its neighbours' pixels. This prevents them from representing a multitude of hydrologic phenomena including the formation of SGLs, because water flow from one pixel to another one is impossible. This omission is mainly due to computational reasons. Nevertheless, several approaches either exist or are under development to represent the flow of liquid water on the ice sheet surface at high spatial resolution, but only at a small scale and without accounting for interactions with the atmosphere.

Among them, there is the Supra-glacial Hydrology Evolution and Drainage (SHED) model (Gantayat et al., 2023), which has shown encouraging results over GrIS in accord with observations of lake location, notably by being forced by the MAR-based runoff. This physical model of surface flow takes into account the topography with a spatial resolution much finer than the one reached by climate models. It allows to successfully represent most of the lake locations (80%) as well as their seasonal evolution (growth, drainage and refreezing) during the melt season. Also, Yang et al. (2018) developed a model for surface meltwater routing in the southwest of the GrIS, which distinguishes interfluve and open-channel flows and enables the obtaining of useful meltwater routing parameters. This model was compared to three other surface meltwater routing models (Yang et al., 2020) being forced by MAR. These models notably allowed to simulate more realistic moulin discharges than in the case where only MAR outputs were used. However, none of these are coupled to a regional climate model. Finally, Yang et al. (2022) found a positive correlation between modelled runoff (from MAR and MERA-2) and observed drainage density on the GrIS.

In summary, while there are currently several SMB models representing numerous processes, none of them account for supraglacial hydrology. There exist models for supraglacial hydrology, but they do not allow to estimate the SMB.

Chapter 3

Study of Supra-glacial Lakes in Greenland Using Remote Sensing

Before focusing on the modelling of the SMB of the GrIS and beginning to evaluate the impact of supra-glacial hydrology on it, it is relevant to study supra-glacial hydrology, and lakes in particular, from a purely observational perspective. Indeed, this work has several benefits, including the creation of a dataset based on satellite observations, which will be very useful later in the modelling of supraglacial hydrology's impact on the SMB (Chapter 4). Naturally, this chapter is also interesting because it will allow us to learn more about supra-glacial hydrology and lakes as well as the methods for processing satellite data.

Outline First, we will study the location of SGL (sections 3.1.2 & 3.1.3), this will result in the creation of a dataset that will be useful in the following. Next, we will examine the factors influencing the location of the lakes (Section 3.2). This chapter concludes with the presentation and application of a technique to assess the depth of the lakes (Section 3.4).

Throughout this chapter, we will pay particular attention to the area occupied by the lakes on the ice, as this influences the albedo. This also allows us to quantify the phenomenon and provide orders of magnitude. It is with this interest in orders of magnitude that we attempt to assess the depth of the meltwater above the ice. Providing orders of magnitude is valuable because they will be crucial for modelling the impact of hydrology on the SMB. This is important for this thesis as well as for future work.

3.1 Supra-glacial Lakes (SGL) in Greenland

First, it is appropriate to define what supraglacial lakes are, more precisely than in Section 1.3.

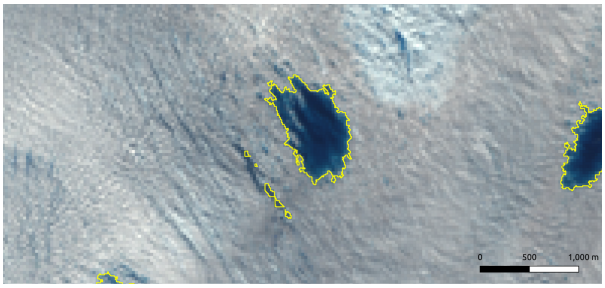
3.1.1 Definition

SGL (also called "melt ponds") are bodies of water surrounding the ice that forms in topographic depressions where melt-water accumulates. Those features are seasonal as they mainly appear in summer, but some SGL appear in the same locations from year to year. Since melt ponds are darker than the surrounding ice, they are easily identifiable which enables their study through satellite images. An example of a melt pond can be seen in Figure 3.1 (a) while the same melt pond seen "from space" can be seen in Figure 3.1 (b). Figure 3.2 shows several melt ponds that are located in the study area of this thesis.

3.1.2 Remote Sensing and Melt Ponds Location

Now that we are more familiar with the concept of SGL, it would be interesting to know where these lakes are located. We will see here that they are primarily found in the ablation zone. This is not surprising since, as mentioned earlier (Section 1.2.2), this is the area where the SMB is negative, meaning where the ice melts, and thus where water can accumulate in lakes. However, before reaching this conclusion, we present the technique we used to precisely locate the SGLs.

(a) Source: Neckel (2018)



(b) Source: GROCE (2018)

Figure 3.1: **(a)** Photograph of a melt pond in Greenland. **(b)** The same melt pond as in (a) on a Sentinel-2 image. Both images are from 29 July 2018.

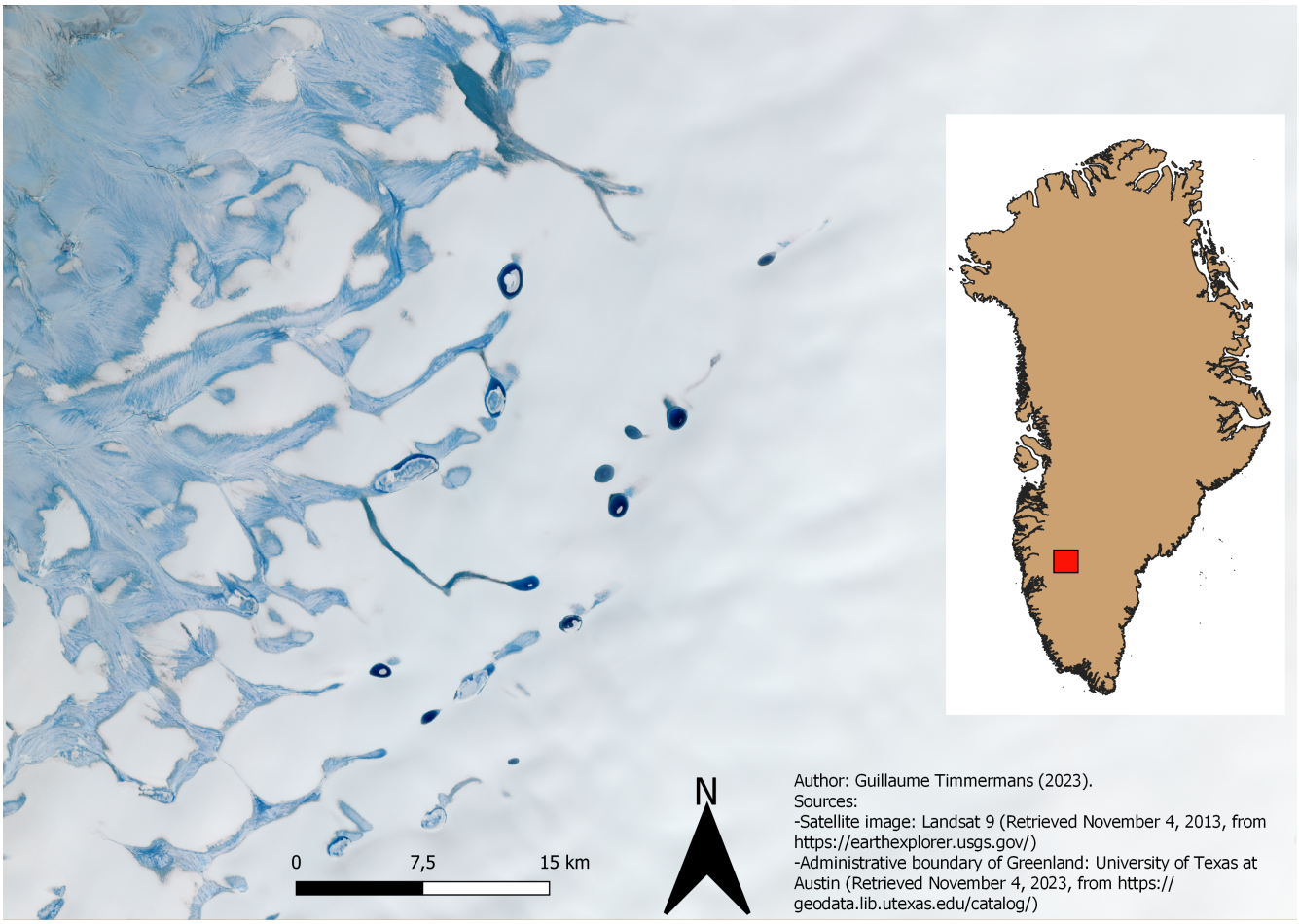


Figure 3.2: Examples of melt ponds in S.W. Greenland. Red Green Blue (RGB) composition of bands 2, 3 & 4 of Landsat 9 image from 08/25/2023 (made with Quantum Geographic Information System (QGIS)). The Red rectangle indicates the location of the image with respect to Greenland.

As a first approach, we will focus on the Southwest of Greenland. This area is of particular interest because SGLs are numerous due to relatively high temperatures. More specifically, we will concentrate on a particular area: the one we have chosen for our MAR simulations (see Chapter 4). Hereafter, we will refer to our study area as the "MAR domain". This term will be explained in Chapter 4, where we will also justify why we chose this particular zone.

Approach In order to determine the location of the supraglacial lakes as well as the period during which these features are the most widespread, it was relevant to use satellite images. We applied the Classifier package in Google Earth Engine (GEE) (Google, 2023) on Landsat 8 images to identify water bodies. This operation allowed us to download a binary raster classifying pixels according to whether they contain (melt-) water or not. It was then necessary to distinguish water bodies located on the ice sheet from others (*i.e.* the sea pixels). Indeed, the algorithm cannot distinguish seawater and melt-water.

Of course, rigour demands that we pay particular attention to the images used for the analysis. The first step was therefore to select our images. We summarise the procedure below.

Satellite Images Selection GEE allows to select the images depending on their cloud-coverage. It also enables filtering the images by date. In order to have images over the entire study area, it was sometimes necessary to deal with data taken at different dates, because of the cloudiness or the footprint of the satellite. It is important to be aware of this as it can lead to misinterpretation. Indeed, if, for example, part of the study area is covered by images taken during a warmer period and thus increased melting, it is likely that more lakes are observed there than elsewhere in the field. Since then we have always made sure to combine images that are close in time.

Once the images are selected, we can apply our algorithm to identify the pixels where water covers the surface. This latter is described below.

Supervised learning detection of melt ponds The model used was a simple Classification and Regression Tree (CART) classifier. A decision tree, more precisely. Even if ensemble methods models, such as random forests usually obtain better results for classification problems, our simple decision tree has shown excellent results in the Earth Engine map. It would have been necessary to tune the hyperparameters with cross-validation to obtain better results with a random forest model. Also, this would have been an unnecessary waste of time given the excellent results of the decision trees model, related to the fact that the classification problem consisting in identifying water-filled pixels was quite simple. The model's inputs are Landsat 8 bands 2, 3, 4, 5, 6, 7, and 10 images. It was trained with a classified dataset from Google (Google, 2023).

Raster Resolution It should be noted that satellite data can be downloaded at different resolutions. The resolution used to download the data from GEE exerts an influence on the results, as shown in Figure 3.3. We develop the subject further in the document (Figure 3.9). It should be noted that the resolution of the images is expressed in meters per pixel. This corresponds to the length of the sides of the pixels, which are (roughly) square (see Google (2024) for further details).

Finally, as already mentioned, it is necessary to differentiate between ocean pixels and lake pixels.

Ice Sheet Masking To distinguish the bodies of water on the ice sheet from that corresponding simply to the ocean, a mask was applied to the binary raster in QGIS. Before applying the raster, we tried to interpolate another raster hiding pixels that are not on the cap, using the same interpolation technique presented in Section 4.1.2. However, for high resolutions, this method proved to be extremely "costly" (in terms of computation time). Thus, we manually created a vector mask (using the graphical interface of QGIS) by visualising the satellite images (which easily distinguish between the ocean, tundra, and ice sheet).

3.1.3 Location of Supra-glacial Lakes in the Domain

Following the procedure described in Section 3.1.2, we obtained raster images of water bodies on the ice sheet. Nevertheless, as can be seen in Figure 3.4 it is difficult to interpret these raster images at a small scale. One solution would have been to represent each lake with a point on a map and adjust the size of the point based on

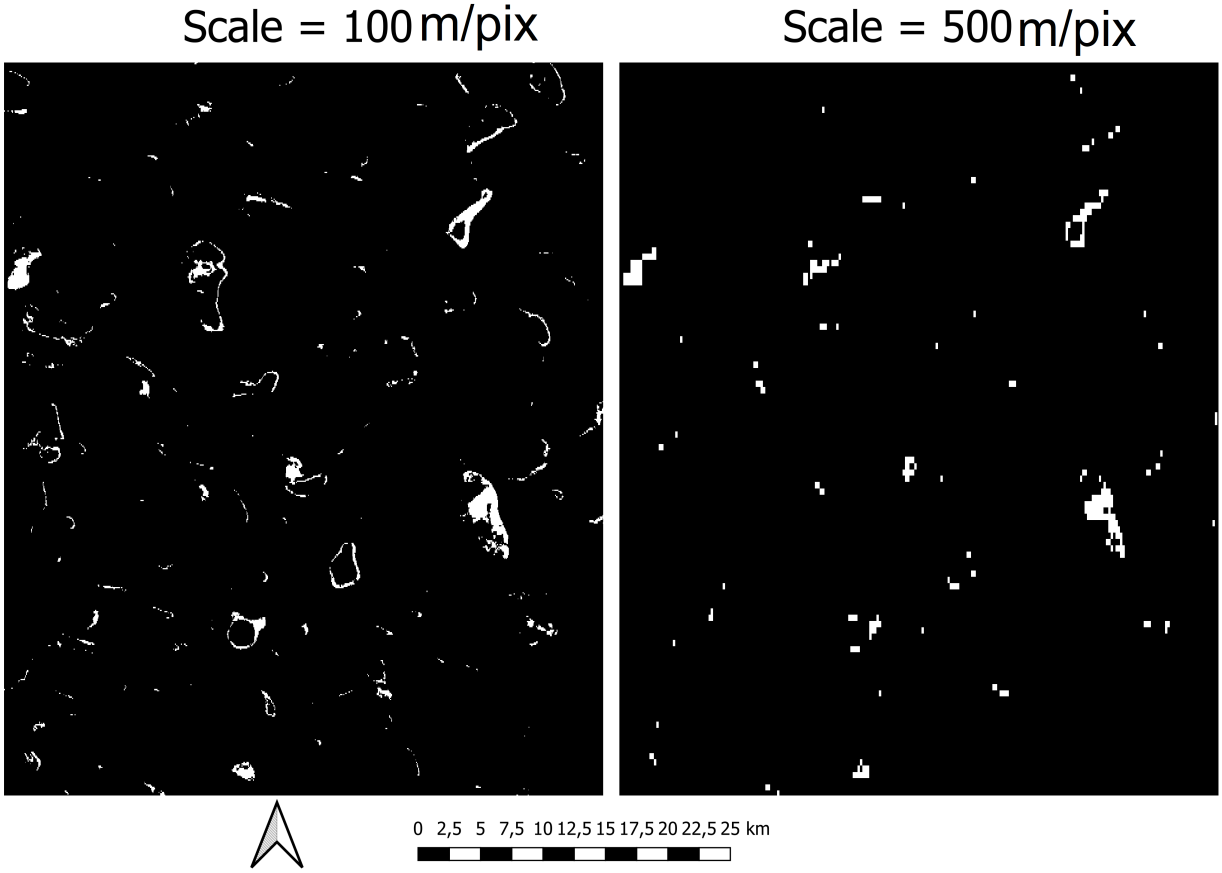


Figure 3.3: Illustration of the influence of the scale parameter when downloading data from GEE. The units of the scale are metres per pixel, which corresponds to the length of the side of a pixel.

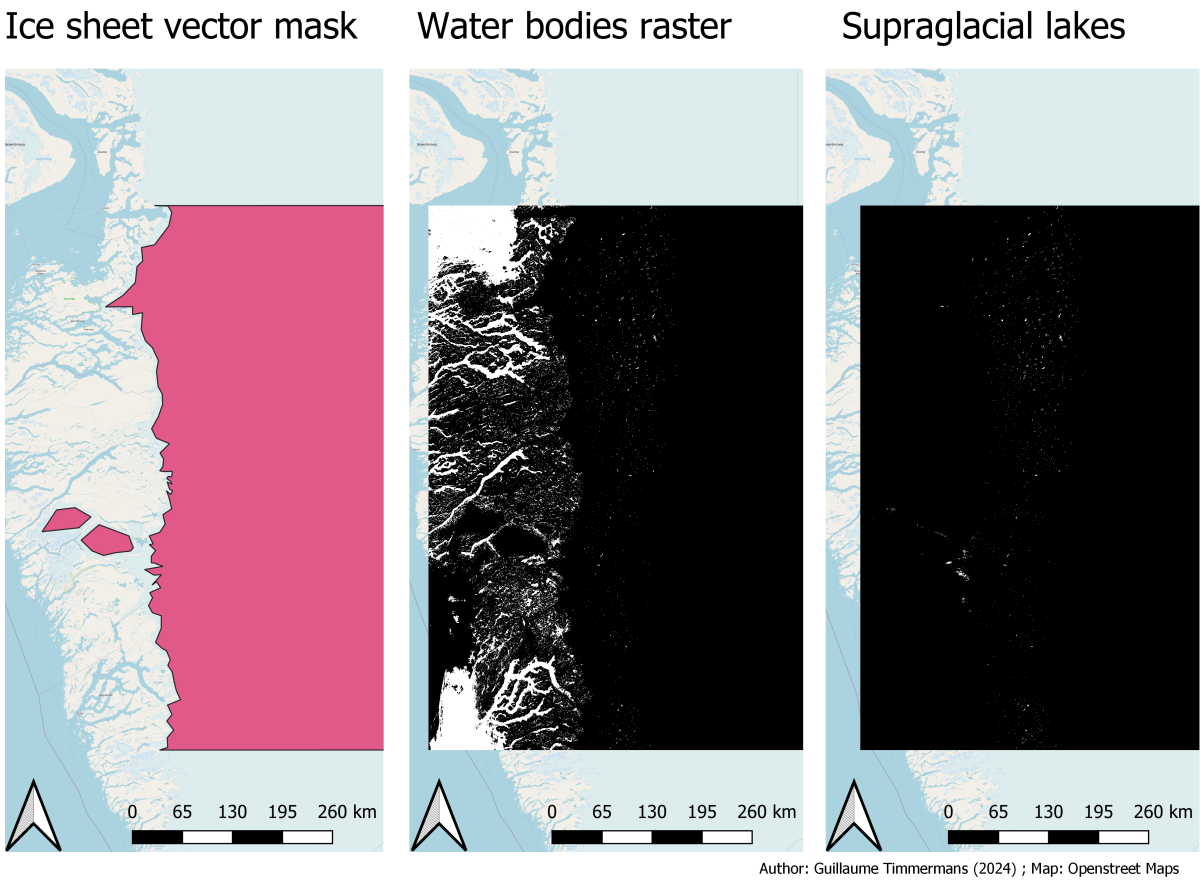


Figure 3.4: Illustration of the procedure applied in QGIS to distinguish melt ponds from other water pixels. Unfortunately, it is difficult to see SGLs on the third map with such a small scale, but you can easily see their presence by zooming in. The purpose of this figure is to show how we eliminate the water pixels that are out of the ice sheet. Therefore, we refer to Figure 3.7 for a more satisfactory illustration of the location of the lakes.

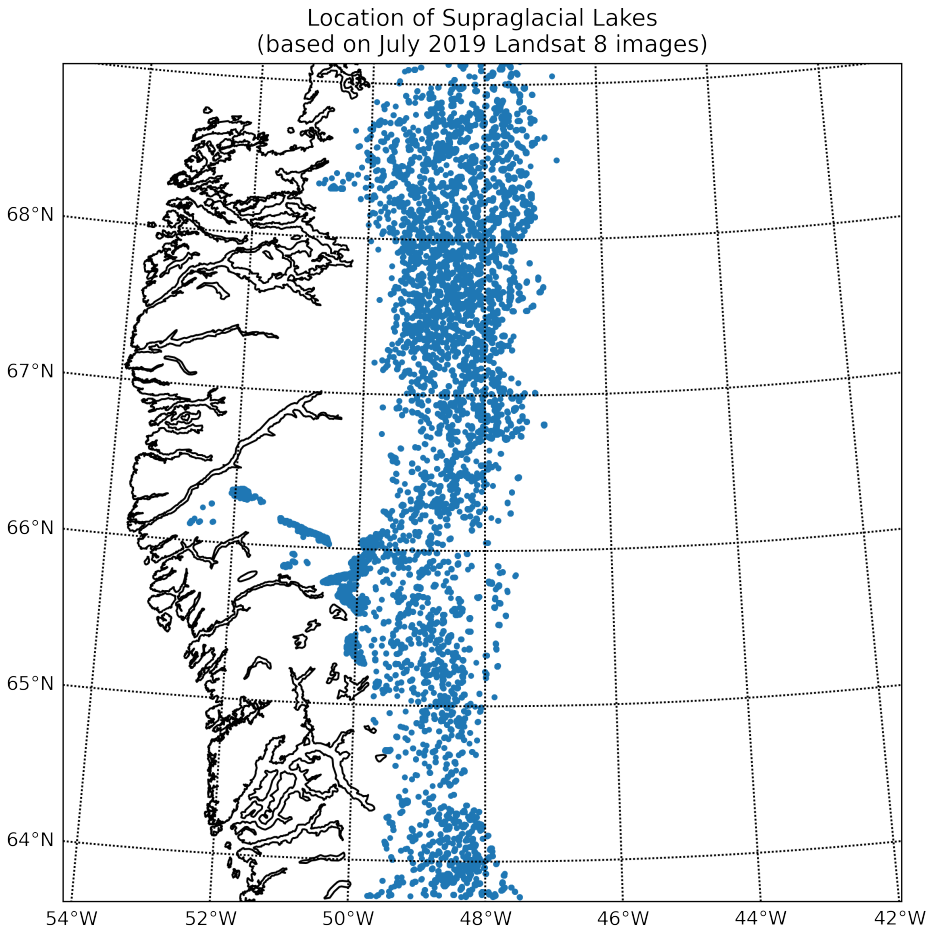


Figure 3.5: Location of the Supra-glacial Lakes at a rather "high resolution" (scale = 100m/pix) in July 2019 based on Landsat 8 images. The scatter plot was obtained following the procedure described in Section 3.1.2.

the lake's surface area. However, this requires clustering the pixels, which is not easy. Moreover, the interest is quite limited since this would only serve to graphically represent the lakes. Figure 3.5 still represents the lakes with points, associating each water pixel with a point. Since the points are too large relative to the pixel, the visual interpretation is "skewed".

Area occupied by SGL Therefore, to represent where the supra-glacial lakes were located within the domain, in a more reliable way than in Figure 3.5, we decided to compute the portion of surface occupied by SGLs in each pixel. So we wrote a program counting the number of SGLs per pixel of the MAR domain we used for our SMB simulations (*cf.* Chapter 4.1). This was done in C++ to reduce computing time, which is proportional to the number of lakes times the number of MAR pixels. The raster downloaded from GEE contains rectangular pixels whose centres are separated by a constant angular gap in longitude and also in latitude. As the Earth is (roughly) spherical, the pixels in the North of the MAR domain are smaller in area than those in the South. Indeed, if we assume that the Earth is a sphere, denote by λ the longitude and ϕ the latitude, and consider that they are given in radians, the area of a pixel is given by:

$$\int_{\lambda-\Delta\lambda/2}^{\lambda+\Delta\lambda/2}\int_{\phi-\Delta\phi/2}^{\phi+\Delta\phi/2} R^2\sin(\pi/2-\phi)\,d\lambda\,d\phi = \Delta\lambda R^2\int_{\phi-\Delta\phi/2}^{\phi+\Delta\phi/2}\cos\phi\,d\phi = \Delta\lambda R^2\left(\sin(\phi+\Delta\phi/2)-\sin(\phi-\Delta\phi/2)\right)$$

Where $\Delta\lambda$ and $\Delta\phi$ correspond to the angular distance between two pixels' centres in longitude and latitude, respectively. R is the Earth's radius.

Figure 3.6 represents the area of a pixel of GEE's raster as a function of the location in the MAR domain. Once we have computed this we can determine the percentage of the surface occupied by supra-glacial lakes for each

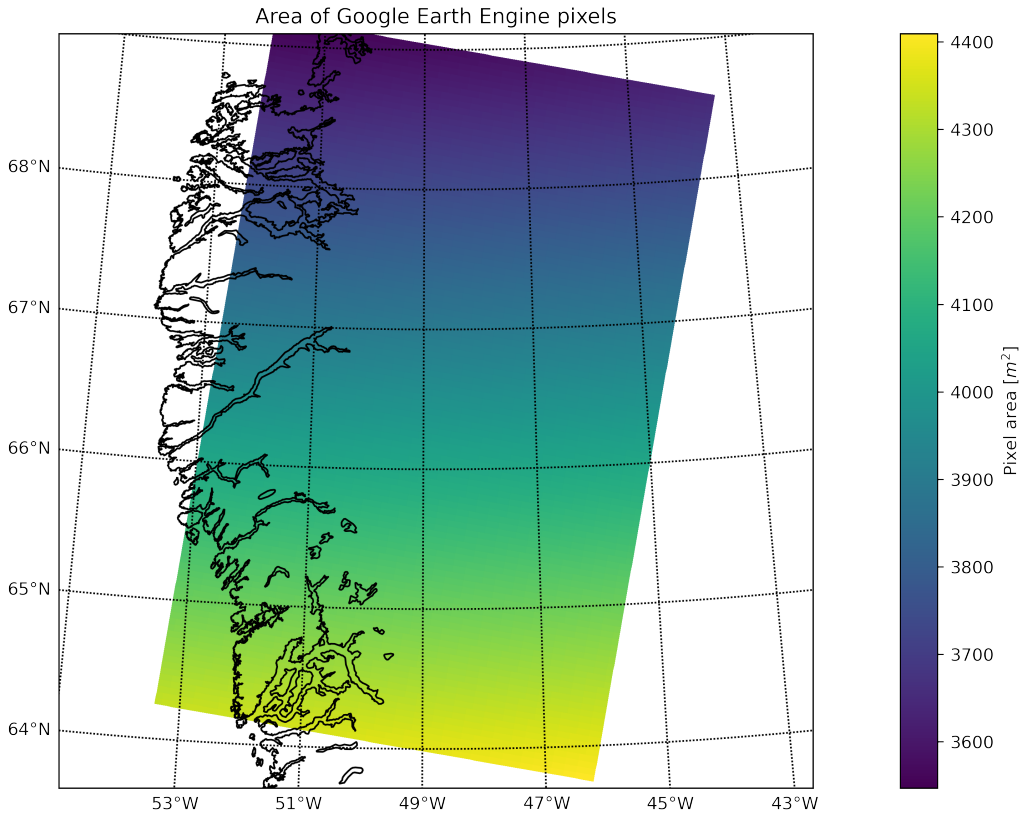


Figure 3.6: Area of GEE's pixels as a function of the location in the MAR domain. It is a function of the latitude, as explained in the text.

of the MAR pixels. It is simply the ratio of the number of supra-glacial lakes in the pixel times their area (that depends on the latitude) divided by the area of the MAR pixel ($\times 100\%$). The area occupied by supra-glacial lakes in each MAR pixel is shown in Figure 3.7 for July 2019.

In Figure 3.7, we also represent the "equilibrium line" (we wrote "Ablation Zone Boundary" in the legend) which corresponds to the boundary between the ablation zone and the accumulation zone. This line was determined based on a MAR simulation of the SMB for the year 2019, forced by the ERA-5 reanalyses. It allows us to realise that the SGLs are located exclusively in the ablation zone simulated by MAR. As already mentioned, this is explained by the fact that the ablation zone is an area where the ice melts, and consequently, where there is more water available to accumulate in lakes. It should be noted that, except for a few pixels at the ice sheet's edge, it is rare for the proportion of a pixel covered by lakes to exceed 20% (there are only five pixels that are covered by 20% or more by SGLs, the maximum being 55%). The phenomenon (*i.e.*, the occurrence of SGLs) is indeed quite spatially uniform (in the ablation zone).

To sum up, in the ablation zone, a little less than 10% of the area is occupied by SGLs, and this rate does not vary significantly within the ablation zone. There are exceptions, but very few.

Discussion of the Resolution's Influence on the Results

Nevertheless, as with all scientific results, those presented in Figure 3.7 are subject to errors. The greatest source of error in our case is certainly the resolution at which we download the images. Let us note that the algorithm for identifying water-covered pixels is also a source of error.

As mentioned above and shown in Figure 3.3, the resolution at which we download data from GEE matters and it also influences the scientific results. Indeed, if we compute the proportion of the surface occupied by SGLs in each MAR pixel, following the procedure that we describe above the results depend on the scale parameter. If the scale is

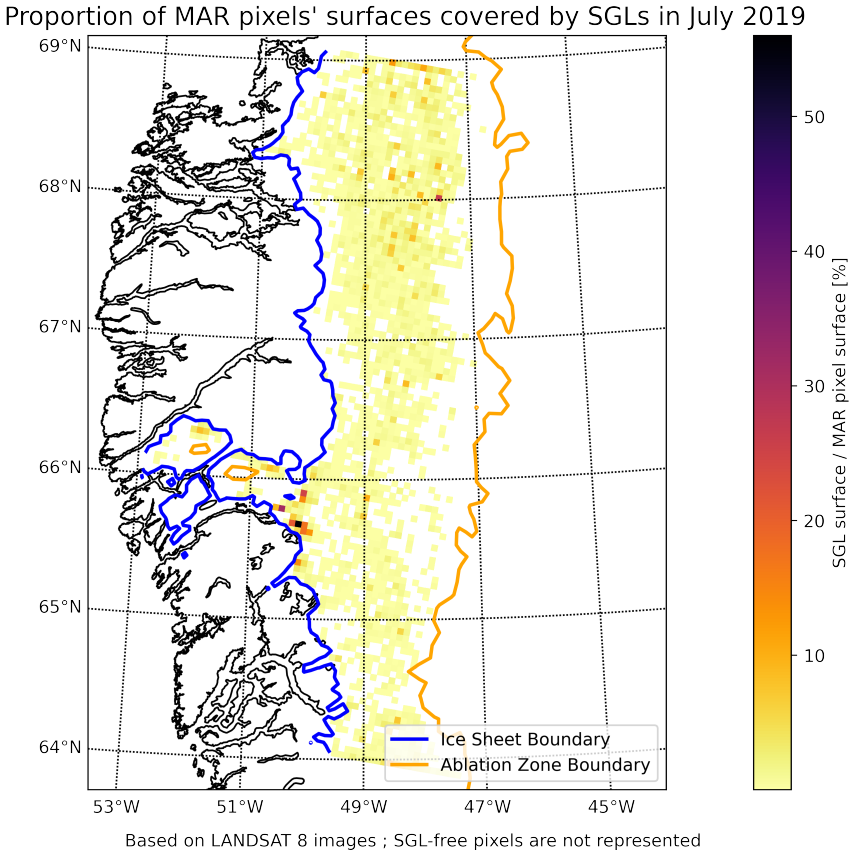


Figure 3.7: Percentage of each MAR pixel's surface occupied by supra-glacial lakes in July 2019. We refer to the text for the procedure used to obtain this result. The ablation zone boundary (*i.e.*, the equilibrium line) is the one obtained from a MARv3.14 (forced by ERA-5) simulation over the whole GrIS (it is the same data (from Fettweis (2023)) as for Figure 1.5 (a)).

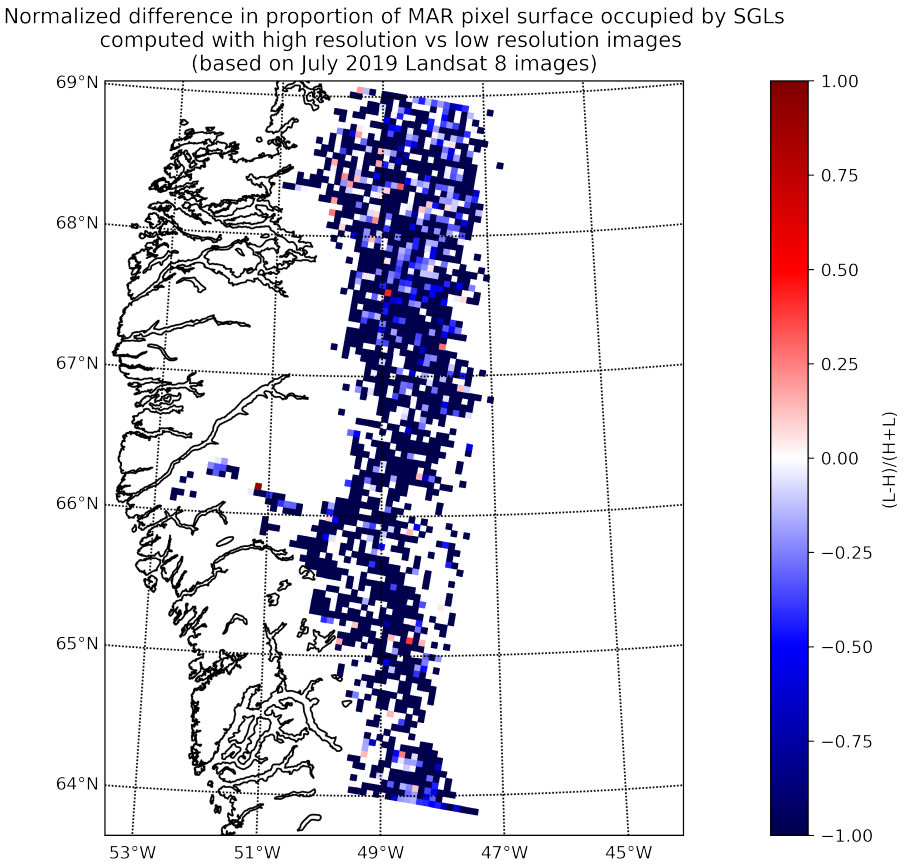
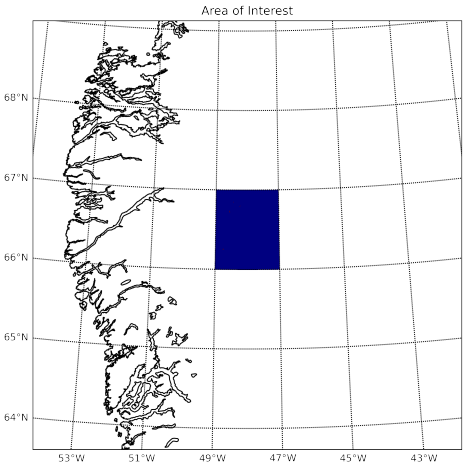


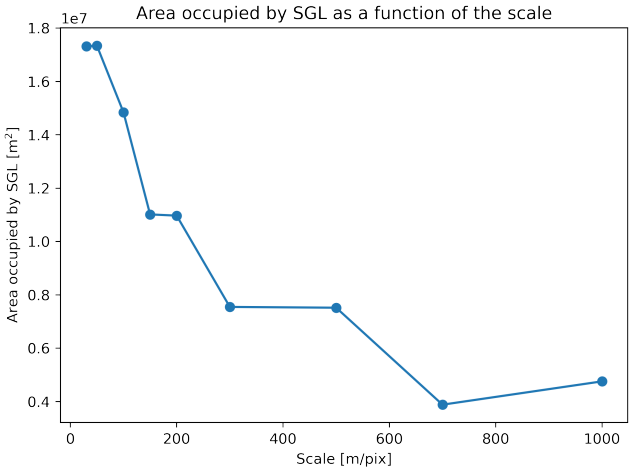
Figure 3.8: Proportion of SGLs in each pixel computed with an image of scale = 500m/pix minus the same but computed with images of scale = 100m/pix, divided by the sum of the two proportions for each pixel. If we denote the proportion of the i-th pixel's area occupied by SGL computed with scale = 500m/pix by L_i and the proportion computed with scale = 100m/pix by H_i, what is represented for each pixel i is $(L_i - H_i)/(L_i + H_i)$. Blank pixels correspond to pixels that are either out of the ice sheet or devoid of SGL (leading to a division by zero).

too large, the surface occupied by SGLs is underestimated, as shown in Figure 3.8, which represents the normalised difference between the proportion of the surface of each pixel occupied by SGL computed with two versions of the same image, downloaded with different scales. Actually, the computed total surface occupied by SGLs in the whole domain depends on the scale. If we take scale = 500m/pix, we find out that SGLs occupy ~ 156388400 m^2 whereas we obtain a value of ~ 396594340 m^2 if we take scale = 100m/pix. The second value is ~ 2.5 times greater than the first! This observation is quite annoying. Furthermore, nothing tells us that the estimate with a scale equal to 100m/pix is accurate. Nevertheless, even at bad resolution, this method allows to determine in which region and during which period the SGLs appears in the domain. An important thing is that **we should never compare results obtained with images of different scales**.

Given this, it seems relevant to focus in more detail on the effect of the scale parameter on the area occupied by SGLs. Therefore, we selected a smaller area (see Figure 3.9 (a)) to conduct an "experiment" consisting in computing the surface occupied by SGLs in this small area with the same LANDSAT 8 image downloaded at different scales. The interest of working on a small zone is, of course, that it allows one to download images with relatively small scales. Figure 3.9 (b) represents the area covered by SGLs as a function of the scale used to download the image from GEE. For accuracy, we take into account the influence of the latitude of the pixel in the computation of the total area occupied by SGLs. The zone of interest is located on the ice sheet, in the ablation zone. The smallest scale afforded for this experiment is 30m/pix because it is the resolution of LANDSAT 8, so it is not possible to be more accurate than that. Figure C.1 in Section C illustrates the impact of the scale on the area occupied by SGLs. It is an appendix because it is a bit redundant with Figure 3.3.



(a) Zone where we compute the area occupied by SGLs using images downloaded from GEE at different scales. It is located in our study zone, in the Southwest of Greenland.



(b) Area occupied by SGL in the zone of interest as a function of the scale.

Figure 3.9: **(a)** Zone of interest **(b)** Influence of scale on the area occupied by SGL. We used nine LANDSAT 8 images from July 2019 with scales of 30, 50, 100, 150, 200, 300, 500, 700, and 1000m/pix.

Figure 3.9 (b) shows a bit in the same way as Woodcock and Strahler (1987), that a scale of 100m/pix underestimates the surface occupied by SGL in comparison to a scale of 30m/pix. In contrast, a scale of 50m/pix seems to be satisfying. Nevertheless, we do not know how close the truth is to the area obtained with a scale of 30m/pix. Logically, for scales tending towards zero, the total area should converge towards a finite value, as the domain is of limited area.

Conclusion on Image Resolution At first glance, it appears that a resolution of 100m/pix is not optimal, but it is not entirely inadequate either. Given the relatively large area studied here, it would have been difficult to work with a finer resolution. Therefore, we decided to keep the data presented in Figure 3.7, obtained at a resolution of 100m/pixel. This data will be used in Chapter 4 as input for the MAR model (for some simulations). Furthermore, it is extremely unlikely that working with a finer resolution would have led to a different take-home message of this section: that lakes are primarily located in the ablation zone.

3.2 Location Factors of Supra-glacial Lakes (SGL)

After presenting a simple and effective method to locate the SGLs in Section 3.1, we explore the potential links between their location and other variables besides the SMB. The goal is to determine if a variable is strongly related to the presence of lakes. This would allow us to adjust the model's behaviour based on this variable to simulate the impact of SGLs on the SMB with MAR. After that, we will study the same thing at the scale of the GrIS. This will demonstrate that while the technique does not work well in a restricted study area, it provides better results for larger areas.

Plan In section 3.2.1, we work on our MAR domain (*i.e.* the same as the one described in Section 4.1.2), whereas in section 3.2.2, we conduct the same analysis over the whole GrIS.

3.2.1 Supra-glacial Lakes in the *Modèle Atmosphérique Régional* (MAR) domain

We can use the results obtained at a scale of 100 m/pixel to study the link between the topography and the presence of SGLs. Actually, we expect that there is a link between the slope and the presence or absence of SGL (if so, we could adapt the operation of MAR as a function of these variables), to model the effect of the SGLs on the SMB. Therefore, to evaluate whether the slope and the altitude are relevant to explain the presence of SGLs based on a methodical analysis of the data, we used Least Absolute Shrinkage and Selection Operator (LASSO), which is

a widely used technique that allows making a feature ranking (and selection). Feature ranking allows, among other things, to evaluate which variables are "strongly related" to another. Generally, this technique is used to select the "input variables" of a machine learning model, but here, we will limit ourselves to evaluating the variables that are related to the presence of SGL. Briefly, the principle of this technique is the following:

LASSO Given an output variable, y that supposedly depends on other variables x_1, x_2, x_2, ..., $x_n = X$, we build a linear model (*i.e.* a model of the form: $\hat{y} = w_0 + w_1x_1 + w_2x_2 + ... + w_nx_n$), that therefore depends on weights parameters, w_0,w_1,w_2,...w_n with the goal of minimising the following loss function[1]:

$$L(y, w_0, w_1, w_2, ..., w_n, X; \alpha) = \Sigma_{i=1}^{N} (y_i - \hat{y}_i)^2 + \alpha \Sigma_{j=0}^{n} |w_j| \tag{3.1}$$

Where n is the number of variables and N is the number of objects. We have $\hat{y}_i = w_0 + \Sigma_{k=1}^{n} w_k x_{k,i}$ where $x_{k,i}$ denotes the value of the object i for the variable k. The loss function depends on an α parameter. The reason for introducing this parameter is to identify noisy variables (and to avoid overfitting, but we will not use a model in our case). For increasing values of α, weights w of useless/noisy features will tend towards zero faster than weights of useful variables. More information on LASSO can be found in Hastie et al. (2009).

We can apply LASSO to our lake-topography problem. Indeed, we can consider that each pixel is an object, the "output variable" (y if we re-use the above notations) is the proportion of the pixel occupied by SGL, and the input variables (X) are the altitude, the slope, the latitude, and the longitude of the pixel. By doing so, we will see whether, according to LASSO, the slope or the altitude are useful in explaining lake location, compared to latitude and longitude, which, in principle, are not (for such a small domain, we can neglect the effect of latitude on temperature, which itself influences the melting, and ultimately, the presence of SGLs). As mentioned previously, a potential application would be to adapt the functioning of an SMB model, such as MAR, based on the "input variables" to account for lakes[2].

Therefore, to apply LASSO, it is first necessary to calculate the input and output variables associated with each pixel of the topography raster. We start with the output variable: the fraction of a pixel's area covered by SGLs.

Proportion of lake per pixel computation We compute the proportion of each pixel area occupied by SGLs for a topography raster of $\sim$1km^2 precision available to us thanks to Brice Noël (personal communication, 2024). Since the topography raster is not cut on a regular grid in latitude and longitude and we do not know the surface of each pixel, it is more prudent to calculate the lake pixel proportion of our raster downloaded through GEE that fits into each pixel of the raster topography. This involves computing $n_x \times n_y \times N_x \times N_y$ distances if we denote by n_x and n_y the number of rows and of columns of the first raster and by N_x, and N_y the number of rows and of columns of the second one. This operation is extremely "expensive" in computing time, and it is quite sensible to impose a condition to calculate the distance. Formally, the program used for this operation did not calculate the haversine distance if the two pixels were separated by more than 0.1° latitude or longitude. Moreover, it is useless to calculate the complete haversine distance, d which is given by

$$d = 2 \times R_E \times \arctan\left(\sqrt{\frac{a}{1-a}}\right)$$

$$a = \sin^2\left(\frac{\phi_2 - \phi_1}{2}\right) + \cos\phi_1 \times \cos\phi_2 \times \sin^2\left(\frac{\lambda_2 - \lambda_1}{2}\right)$$

Where R_E is the Earth radius, ϕ_1, and ϕ_1 are the latitude of the two pixels centres and λ_1, and λ_2 are their longitudes. Indeed, since d is a monotonously growing function of a, we can compare values of a instead of d to determine which pixel centre is the closest. With a parallelised (in openMP) C++ code, the calculation time is significantly reduced. Furthermore, in order not to have to store $n_x \times n_y + N_x \times N_y$ latitudes and longitudes in memory (which are of type long double, so ten bytes) we wrote a simple Python program to write pixel coordinates in blocks of 50,000 in separated files, then a bash script launched the C++ program iteratively to process pixels by blocks of 50,000 (the computer had then only to store $2 \times 50,000 + 2 \times n_x \times n_y$ angles in memory at each execution). Of course, we have cut the topography raster, to keep only pixels over our study area and to decrease n_x and n_y.

Ultimately, we obtain the result presented in Figure 3.10. Note that this figure is quite similar to Figure 3.7.

[1] So the model's parameters (i.e., w_0,w_1,w_2,...w_n) that will be "chosen" are those that minimises the loss function.

[2] We could, for example, modify MAR so that the "runoff speed" becomes a function of one or more input variables, provided that they effectively predict the location of the lakes.

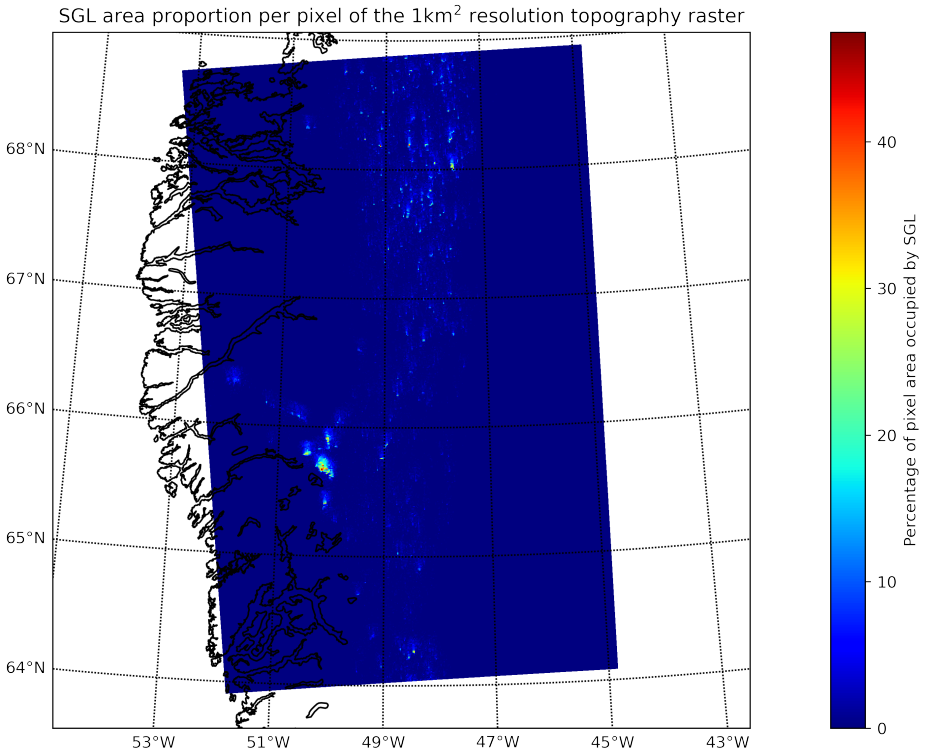


Figure 3.10: Proportion of the 1km^2 pixel area occupied by SGL.

To summarise, our objective is to see if there is a link between certain variables and the presence of SGLs. Among these variables is the slope. We think this variable could potentially be correlated with the presence of lakes, as it seems logical that water is more likely to stagnate if the slope is low. Therefore, we need to calculate the slope of each pixel from topographic data. The procedure used is detailed below.

Slope Computation The slope angle is defined as *the angle α between a path p in the horizontal plane and the sloping terrain, p is chosen such that α is maximized* (adapted from Huisman, de By, et al. (2009)). We refer to Figure 3.11 for notations. If $f(x, y)$ denotes the altitude that depends on horizontal coordinates x, and y (we can use a planar approximation as the pixel length are negligible compared to the Earth radius), the slope angle is given by (3.2) and (3.3).

$$\text{Slope} = \arctan\left(\sqrt{|\nabla f|^2}\right) \tag{3.2}$$

$$|\nabla f|^2 = \left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2 \tag{3.3}$$

To compute ∇f, we use the same technique as the one used by `numpy.gradient` (Harris et al., 2020). But we cannot use it straightforwardly (if we want to be accurate) since our altitude field $f(x, y)$ is measured relatively to a $\sim$ spheric surface of reference: the geoid. By doing so we would make a planar approximation for the whole domain. Instead, we can make a planar approximation for each pixel, which is better. The procedure is the following (Harris et al., 2020):

Gradient Computation When computing the derivative along a horizontal axis (*i.e.* $\frac{\partial f}{\partial x}$ or $\frac{\partial f}{\partial y}$), we want to minimise the difference between the true derivative and its estimate, for each pixel i: η_i. In the following, we will explain how to compute the x-derivative. The computation of the y-derivative is analogous. If we denote by (x_i,y_i) the pixel's centre coordinate, h_d and h_s forward and backward displacement along the x-direction and $f^{(n)}$ the n-th x-derivative of f, it is given by (for a constant y_i)

$$\eta_i = f_i^{(1)} - [\alpha f_i + \beta f(x_i + h_d) + \gamma f(x_i - h_s)] \tag{3.4}$$

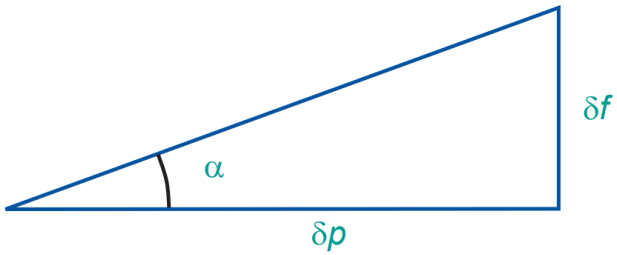


Figure 3.11: Notations for slope definition. Source: Huisman, de By, et al. (2009).

We approximate $f(x_i \pm h)$ using a second-order Taylor development:

$$f(x_i \pm h) \approx f_i \pm h f_i^{(1)} + \frac{h^2}{2!} f_i^{(1)} \tag{3.5}$$

And we get

$$\eta_i = f_i^{(1)} - [\alpha f_i + \beta f_i + \beta f_i^{(2)} h_d + \beta f_i^{(1)} h_d^2/2 + \gamma f_i - \gamma f_i^{(1)} h_s + \gamma f_i^{(2)} h_s^2/2] \tag{3.6}$$

N.B. To simplify notations, we note $f(x_i, y_i) = f(x_i) = f_i$
Equation (3.6) leads to (equaling η_i to zero)

$$\begin{cases} \alpha + \beta + \gamma & = 0 \\ \beta h_d - \gamma h_s & = 1 \\ \beta h_d^2 + \gamma h_s^2 & = 0 \end{cases}$$

So we end up with the following approximation (3.7) for $f^{(1)}$ that *ine fine* only depends on $f^{(0)}$.

$$\hat{f}_i^{(1)} = \frac{h_s^2 f(x_i + h_d) + (h_d^2 - h_s^2) f(x_i) - h_d^2 f(x_i - h_s)}{h_s h_s (h_d + h_s)} \tag{3.7}$$

Of course, we use the haversine distance to compute h_s and h_d. Let us note that Eq. (3.7) simplifies to $\hat{f}_i^{(1)} = \frac{f(x_i+h) - f(x_i-h)}{2h}$ when $h_s = h_d = h$, but using Eq. (3.7) provides a more accurate estimate without overly complicating the program. The result is shown in Figure 3.12.

LASSO Results Once we have computed each pixel's slope, and proportion of area occupied by SGLs, we can determine LASSO weights as a function of α (we mean α of Eq. (3.1)) using `scikit-learn` (Pedregosa et al., 2011). By prudence, we scaled every input variable (*i.e.* slope, latitude, longitude, and altitude). Also, we do not consider pixels that are not located on the ice sheet (a mask was applied to the data). The results are shown in Figure 3.13 for a resolution of 1km^2. We observe that the slope and the latitude are the most "relevant" variables to explain the SGLs locations. We can link Figure 3.13 to Figure 3.14. For example, we see that even if the relationship between the slope and the proportion of SGLs is not obvious and robust, there is almost no pixel with a non-zero proportion of area occupied by SGLs that has not a negative scaled slope (*i.e.* a slope below the mean, as scaling a variable means to subtract the mean and to divide the difference by the standard deviation). This translates into a negative weight, meaning that *the higher the slope, the lower the area occupied by SGLs.*

Resolution dependance As well as for the the area occupied by SGL, LASSO results are resolution-dependents. Indeed Figure 3.15 shows the results of LASSO feature ranking in the case of a 5km × 5km resolution (*i.e.* the resolution of our MAR simulations), and we observe that in this case, the slope could be interpreted as unrelated to the occurrence of SGL, which seems senseless as we do not expect to observe a lake on a non-planar surface. This can be explained by the fact that at this resolution, all pixels are almost flat. This reminds us of what we mentioned in Section 2.2: to explicitly simulate SGL as well as surface hydrology, it is necessary to work at rather fine resolutions. Let us note that the weights associated with altitude and longitude are nearly opposite, this is because they are nearly perfectly correlated (the correlation coefficient is equal to 0.85) so they somewhat cancel each other. This is illustrated in Figure 4.5.

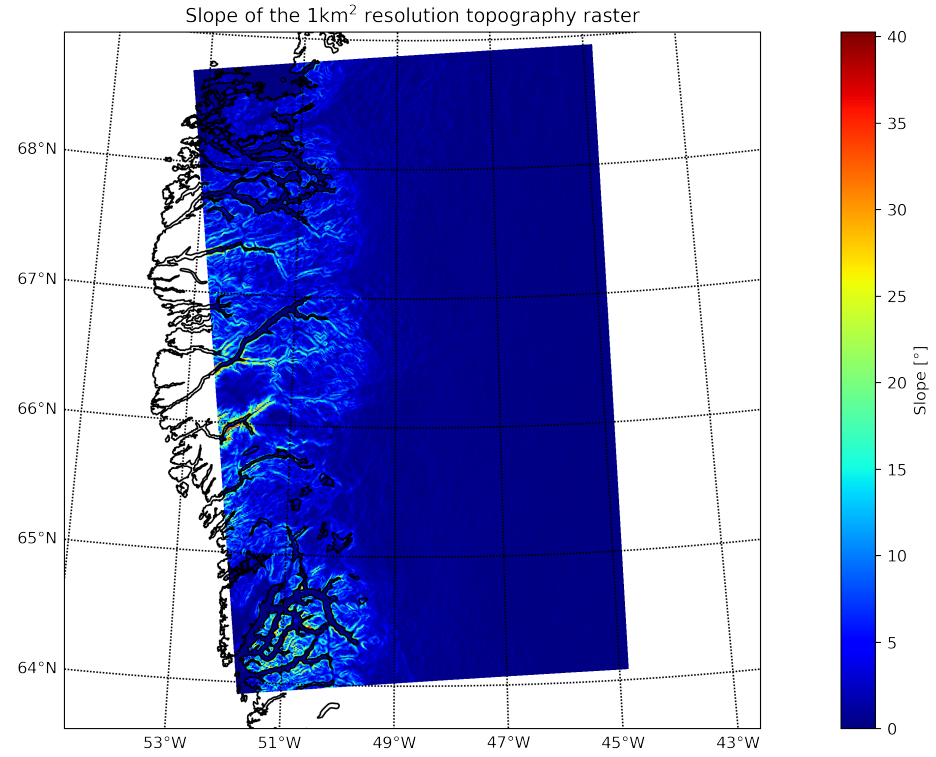


Figure 3.12: Slope derived of the 1km^2 resolution topography raster, following the procedure described in the text.

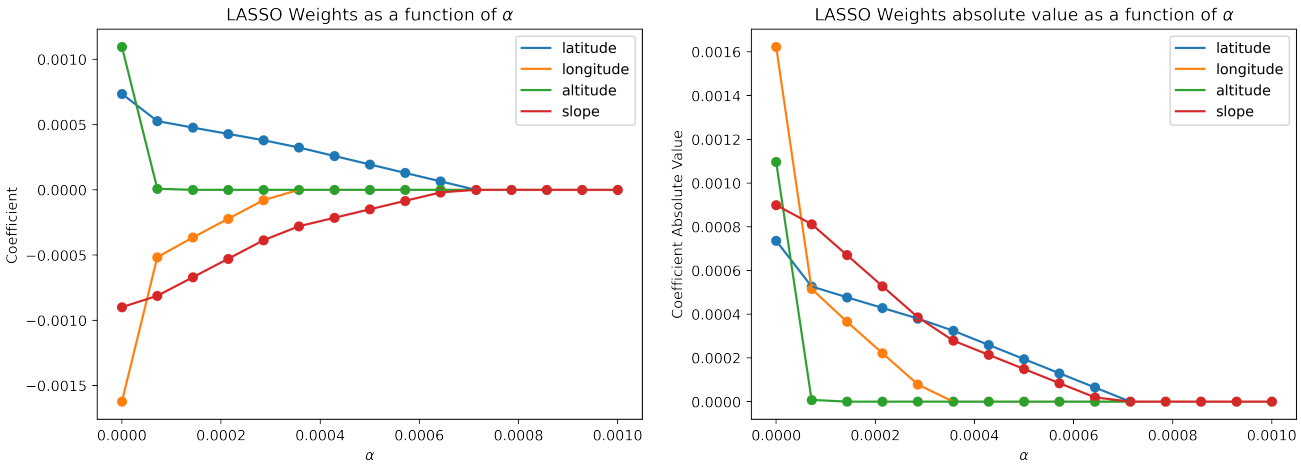


Figure 3.13: Evolution of LASSO linear model weights for increasing large weight penalty (α) coefficient. Results for the high-resolution topography raster.

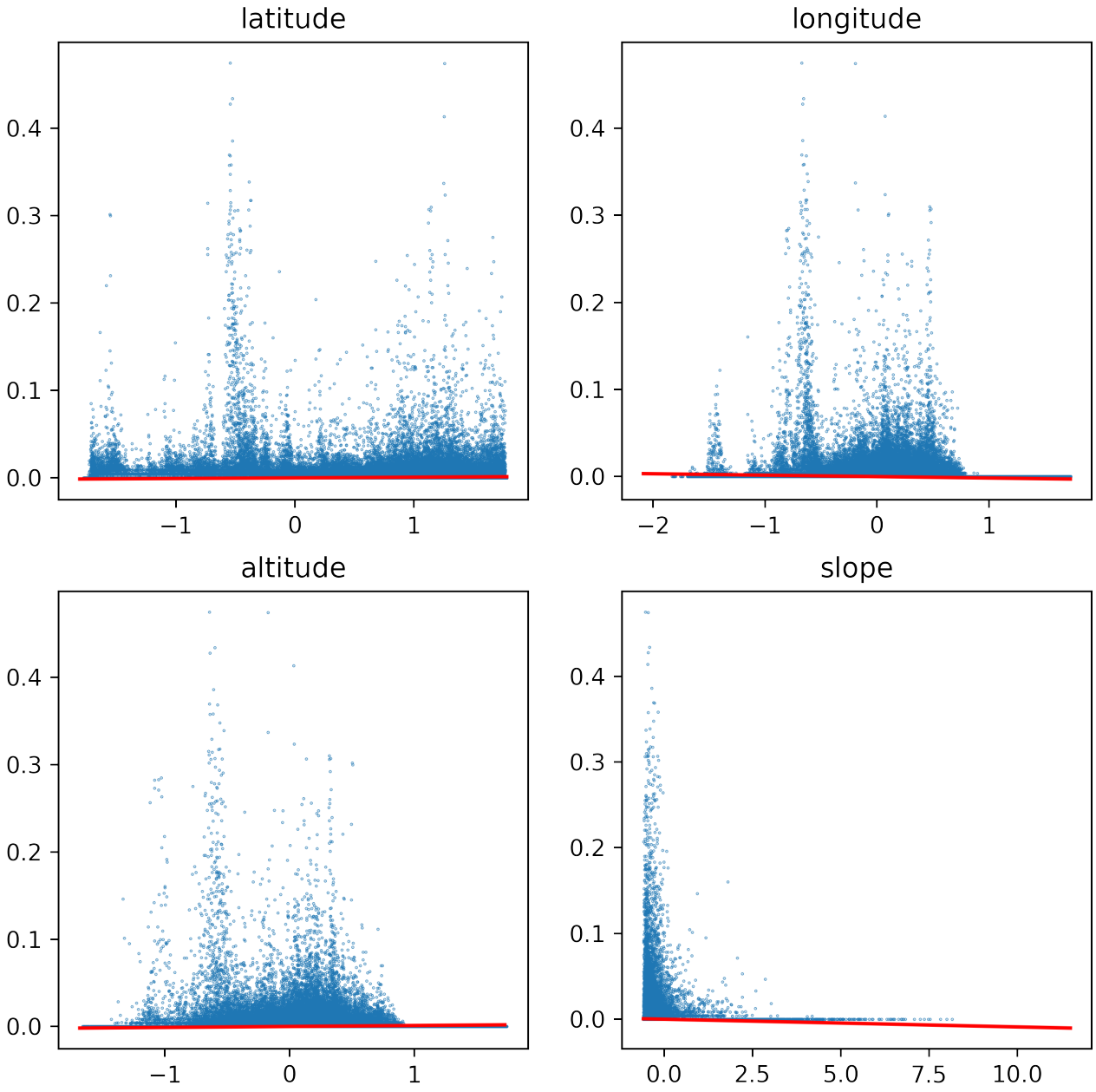


Figure 3.14: Scatter plots where each point represents a pixel. For each plot, the ordinate corresponds to the proportion of lakes occupied by SGL, and the abscissa is the scaled value of the input variable indicated in the plot title. Red lines are linear regression ($\alpha = 0$) relationships.

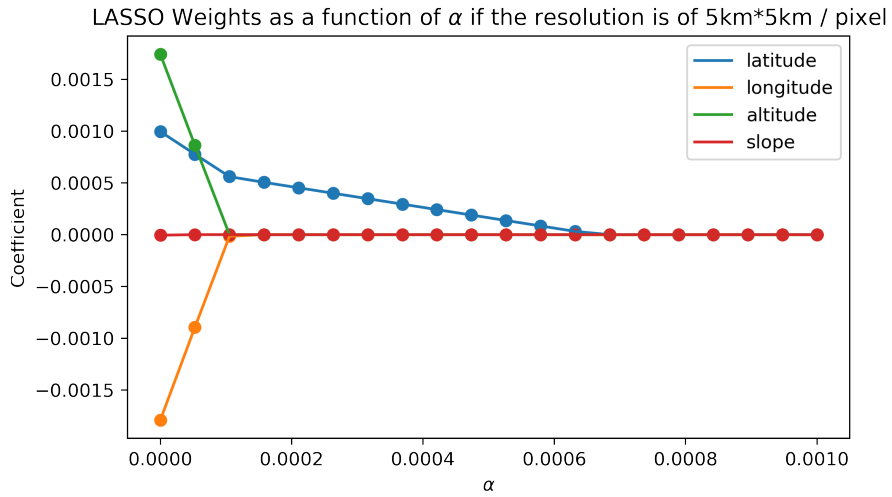


Figure 3.15: Evolution of LASSO linear model weights for increasing large weight penalty (α) coefficient. Results for the low-resolution topography raster.

There is thus no strong link between the topography (either the slope or the altitude) and the occurrence of SGLs according to LASSO, as, for example, the latitude is more "powerful" to explain it even if for such a relatively small domain it has no great influence on the temperature (we can also add that the interpretation stating that the latitude influences the temperature which controls melting cannot be applied and is wrong in this case since the weight should be negative if so, therefore it is nothing else than a coincidence related to the chosen study area). In addition, we can add that the correlation between the slope and the proportion of pixel area covered by SGLs is $\sim$0.0103 at a resolution of 1km^2.

At the end of the day, we can state that we were not able to show a strong link between topography (slope and altitude) and the occurrence of SGL in our domain (even if at much finer resolutions, there should be a link between the topography and SGL occurrence, as we do not expect to see lakes on an inclined plane). This statement complicates the parameterization of supra-glacial lakes in SMB models. Especially since the results obtained at a 5 km$\times$5 km resolution are rather poor, as no variable seems to be related to the location of SGLs (see Figure 3.15). However, if we aim to modify the MAR model to account for the presence of lakes (for instance, by adjusting the runoff process in pixels with a high concentration of lakes), it will be necessary to work at a 5 km$\times$5 km resolution, as this is the resolution at which our MAR simulations are conducted (see Chapter 4).

Furthermore, if we try to explain the occurrence of SGL with the mean temperature, we also obtain puzzling results. Figure 3.16 shows that the temperature can be interpreted as being anti-correlated with the occurrence of SGL, which is, of course, complete nonsense. Therefore, it seems obvious that in addition to the "resolution issue" evoked above, the study area is not representative of the whole GrIS (*NB* After checking, there seems not to be any outlier in the data). So there is still a hope of being able to parameterise the lakes in a GrIS scale model, but if you want to predict their location at the scale of our study area, which is much more accurate, it seems necessary to work at much higher resolution and explicitly resolve surface hydrology.

In conclusion, it is evident that it is not possible to deduce the proportion of lakes in a MAR pixel observed by satellite imagery from the variables we have analyzed. Therefore, it will not be possible to adjust the model's behaviour based on these variables to simulate the presence of lakes. We will thus use a different approach by forcing the model with observations (see Chapter 4).

Nevertheless, we would like to apply the same technique to the entire GrIS to see if our objective (namely, adapting the MAR operation for the runoff as a function of some variables to model the impact of SGLs on the SMB) would be feasible for simulations over a larger area.

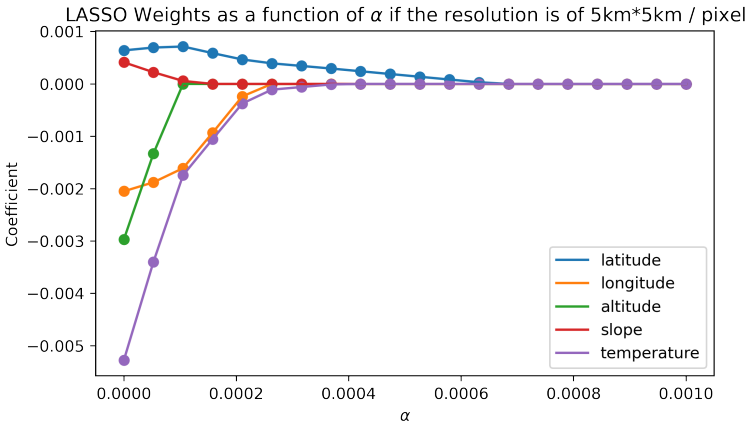


Figure 3.16: Evolution of LASSO coefficients as a function of α for the low-resolution data when we include the scaled mean temperature to the input variables.

3.2.2 Supra-glacial Lakes (SGL) Location Factors over the Whole Greenland Ice Sheet (GrIS)

After realising that temperature is anti-correlated to the presence of SGLs in our domain, it seems likely that the astonishing results obtained in Section 3.2.1 are because the analysis was conducted over a non-significant area. We, therefore, expect to obtain more consistent conclusions by working over the whole GrIS. To do so, we downloaded the same data from GEE as previously (*cf.* section 3.1.2; so classified LANDSAT 8 images from summer 2019), but with a resolution of 400m/pix[3]. This is because GEE does not allow data to be downloaded if their volume exceeds a certain threshold. In this context, we also had to cut the area of interest into several slices, because the threshold applies for each image and it enables us to avoid downloading data over the ocean. The raw data downloaded from GEE that we will use in this section is represented in Figure 3.17 (a).

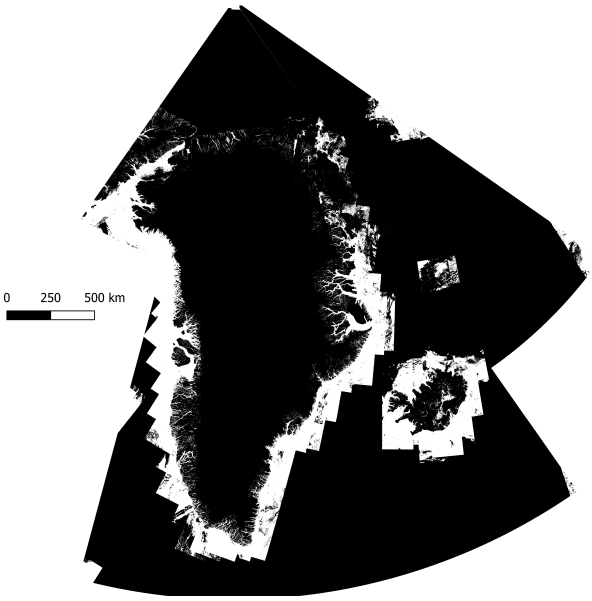
Figure 3.17 shows the data on which we will base our discussion in this section. Figure 3.17 (b) represents the topography raster that we will use. It is on the same grid as the one used for the reference MAR simulation of SMB at a 10km×10km resolution (the same data is discussed and utilised further in Section 4.2.1). The ice mask is shown in Figure 3.17 (c), it corresponds to the proportion of the MAR pixel that is covered by ice. Figure 3.17 (d) shows the proportion of each pixel of the topography raster that is covered by SGLs, pixels over the sea as well as those devoid of lakes are not represented. This was obtained by adapting the C++, Python, and Bash programs used previously. Let us note that the water location in Greenland (not only on the ice sheet) shown in Figure 3.17 (d) can be understood in the light of Figure 3.17 (b). Indeed, there exists a well-known (general) relationship between temperature and altitude (*cf.* Section1.1).

Data Masking Since we are interested in SGL, we should apply a mask to the data. It was not a big concern in the previous section (3.2.1) as we applied a vector mask in QGIS (that was designed "by hand", *cf.* Figure 3.4) in addition to the MAR mask, but here, the study area is larger and it is not worth drawing a vector layer manually. Since the results will depend on the applied mask, it seems relevant to interest ourselves in it, as we do in Appendix D. Furthermore, as many researchers work with MAR, this discussion will probably interest some readers. We explain in particular what a "99% data masking" means. This is not essential for understanding the rest of the discussion, and it is sufficient to note that it is a relatively "strict" method for delineating the ice sheet.

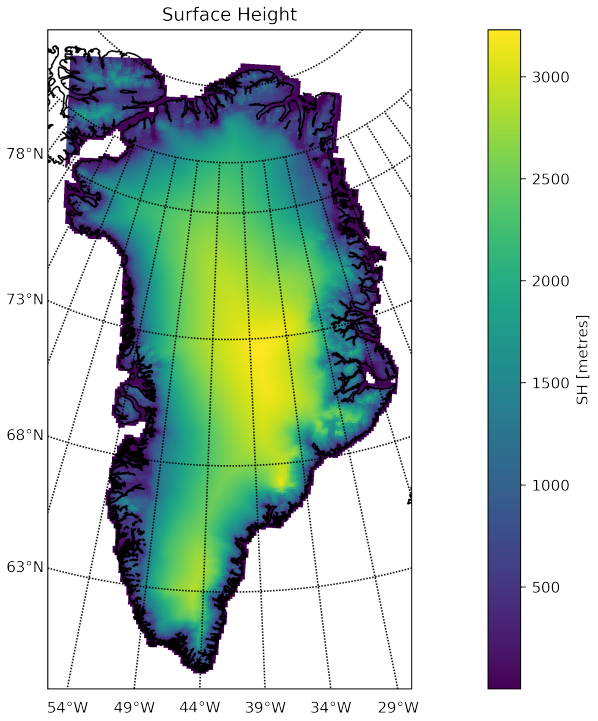
We therefore refer interested readers to Appendix D, which provides an in-depth discussion of data masking. There, we explain why it is necessary to apply a fairly strict data masking to avoid any bias[4].

[3]Meaning that we should not directly compare the percentage of area occupied by SGL with results of section 3.2.1 as we have seen that the scale introduces a bias (*cf.* Figure 3.9), but it is not really problematic.

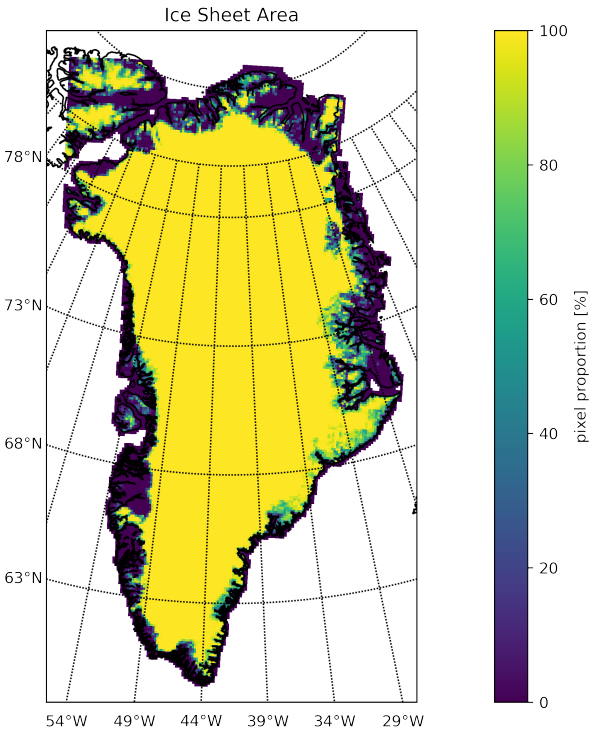
[4]So we used a threshold of 99% in the following. See Appendix D for further details.



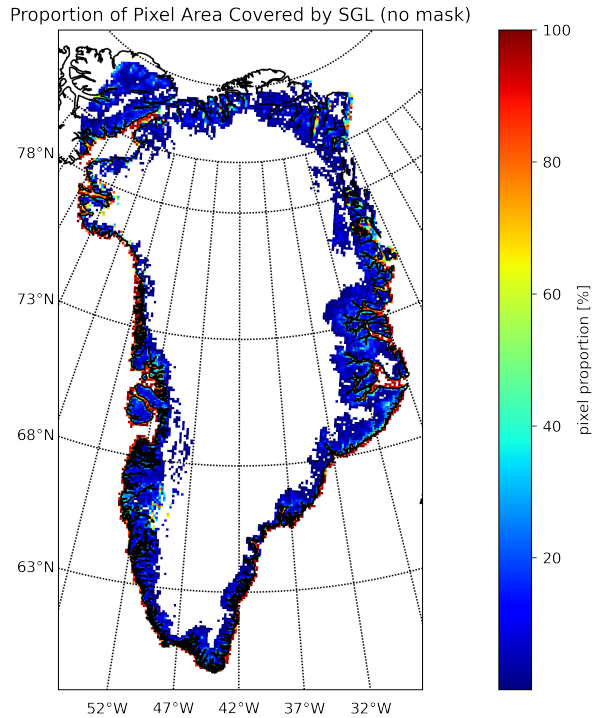
(a) Raw data from GEE. Binary raster (water pixel or not) as identified by the CART model discussed in section 3.1.2. *NB* some "no data" pixels are also shown in black (*e.g.* over the ocean).



(b) GrIS topography at $\sim$ 10km $\times$ 10km resolution.



(c) Proportion of MAR pixel area covered by ice.



(d) Proportion of topography raster's pixels covered by water. Pixels devoid of water or located over the ocean (*i.e.* where the altitude is zero) appear in white .

Figure 3.17: Data used for investigating the link between topography and SGL occurrence over the whole GrIS.

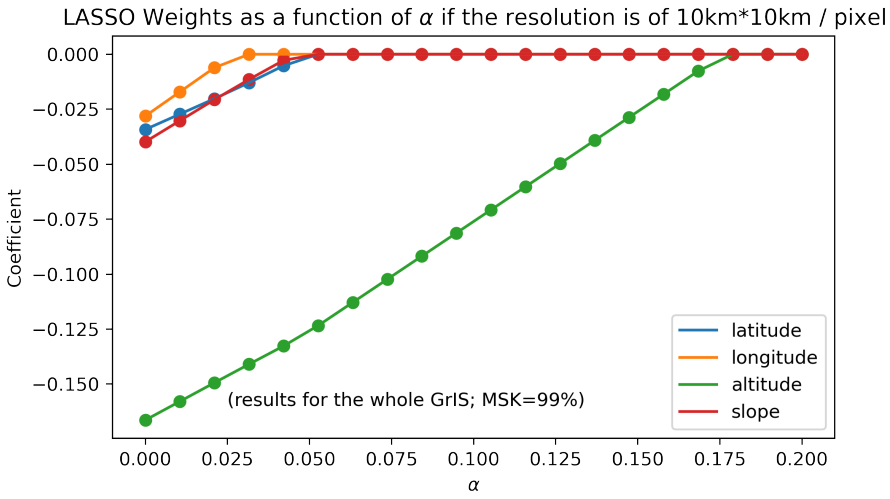


Figure 3.18: LASSO coefficients evolution as a function of α for the whole GrIS. *NB* We refer to Appendix D for further detail on the mention "MSK = 99%".

LASSO

We conducted the same LASSO analysis as in section 3.2.1 for the whole GrIS. The LASSO weights evolution as a function of the "penalization parameter", α (*cf.* Eq (3.1)), are shown in Figure 3.18. The slope derived from the altitude presented in Figure 3.17 (b) and the masked version of Figure 3.17 (c) are available in Figure 3.19.

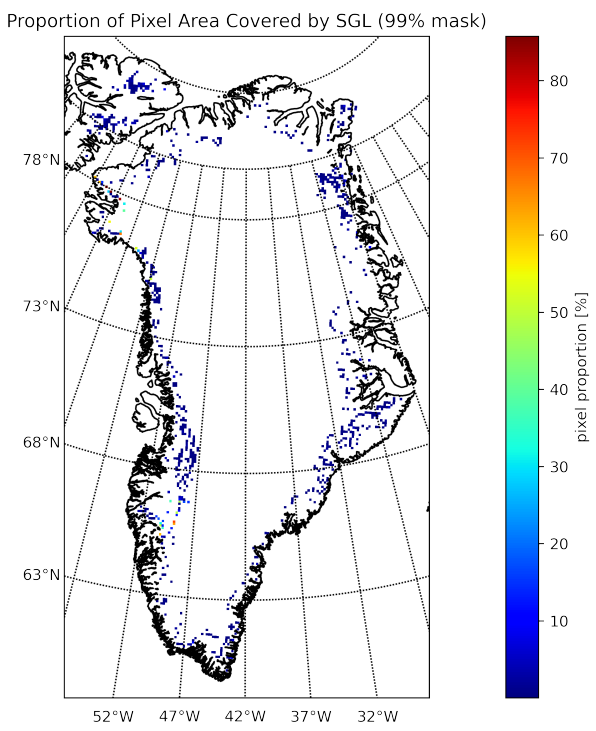
As expected, the results are much more consistent and the interpretation is straightforward: The main, if not the only, location factor of SGLs (among those we have considered) is the altitude. Of course, this is because it is almost directly linked to the temperature (this latter controls the melt) in climatology, as mentioned in Section 1.1. Therefore, we can conclude that we have demonstrated a link between altitude and the presence of SGL. This could potentially be useful for large-scale modelling of supra-glacial lakes, even though it will not be a "panacea".

3.3 Temporal Evolution of Supra-glacial Lakes (SGLs) over the Greenland Ice Sheet (GrIS)

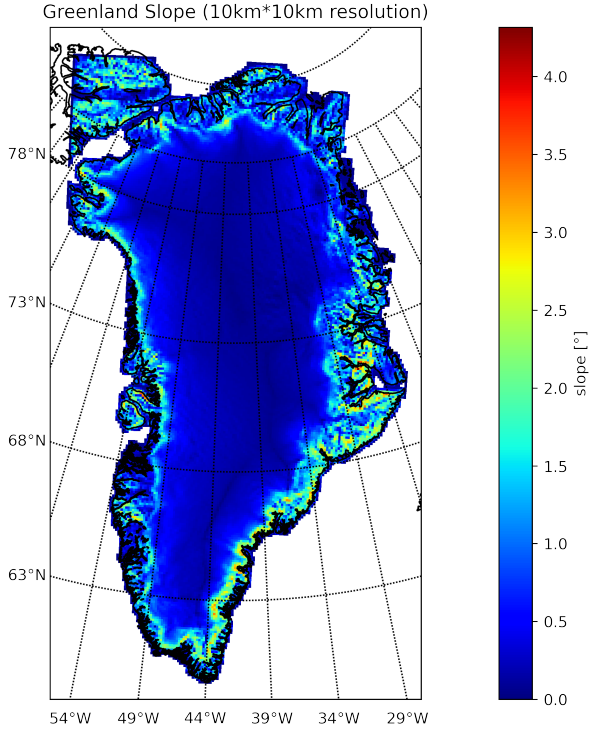
Let us now focus on the temporal evolution of supra-glacial lakes. This study presents two main interests. Firstly, it will allow us to learn more and improve our knowledge of supra-glacial lakes. Secondly, if we find that the spatial coverage of supra-glacial lakes is constant over time, *i.e.*, that it does not change from year to year, then it will mean that it is entirely feasible to force the MAR model with *constant* data on lake distribution from one year to the next. Let us note right away that this second option will probably not be fulfilled 100% as it seems evident that the occurrence of lakes is linked to temperature, and not all years are the same climatologically.

To analyse the summer-to-summer evolution of lake coverage in our study area located in southwest Greenland (see Section 3.1.2), we applied the same procedure as before, using images for the month of July with resolutions of 100m/pixel. Since our method to identify surficial water relies on Landsat 8 images, that was launched in 2013, it is impossible to go back "so far" in time (at least with our method).

Even if we expect the surface of the ice sheet covered by SGLs (in summer) to increase in time, we do not observe it in our study area over the 2014 - 2023 period. Figure 3.20 (a) shows the trend over, this period, of the area occupied by SGLs, for each MAR pixel. The colour of each pixel (in Figure 3.20 (a)) corresponds to the weight, w_1, of a linear regression model, computed for each pixel, that minimises the TSE. The input data, or argument, a, of the model is the year and its output, y, is the percentage of the area occupied by SGLs. Actually, these values were scaled in order to rule out the intercept w_0 of each pixel model. Indeed, it is necessary as the parameters of



(a) Proportion of pixel (that are covered by at least 99% of ice) occupied by SGL



(b) Slope of Greenland computed using a 10km×10km resolution topography raster.

Figure 3.19: **(a)** SGL pixel proportion when the mask is 99% **(b)** Slope of Greenland.

a linear regression (of the form $\hat{y} = w_0 + w_1 a$, where y is the percentage of SGLs in a given pixel, and a the year) are given by Eqs. (3.8) & (3.9) (Geurts and Wehenkel, 2023).

$$w_1 = \frac{\sigma_{a,y}}{\sigma_a^2} \tag{3.8}$$

$$w_0 = \bar{y} - w_1 \bar{a} \tag{3.9}$$

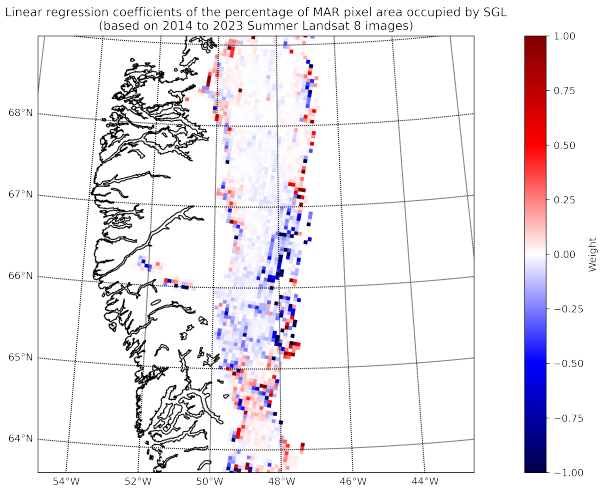
$$\tag{3.10}$$

If we denote the covariance by σ, and the variance by σ^2. We observe in Figure 3.20 (a) that there is not a clear trend to an increase or to a decrease of the area occupied by SGLs. This is due to several factors. First of all, the images are not always from the same day of the year (this would not be possible as there are clouds) which induces a bias. Second and most important, the time series is only ten years long, with 2019 included in its middle, which was a "record year" in terms of melting (Tedesco and Fettweis, 2020). This Figure therefore illustrates that the appearance of SGLs is a quite complex problem. As a complement to Figure 3.20 (a), we provide Figure 3.20 (b) that represents the denominator, σ_a^2, of Eq. (3.8), and that illustrates the variability of the area occupied by SGLs. Let us note that, as all images were downloaded with a resolution of 100m/pixel, the scale does not constitute a bias since all years are "on the same boat".

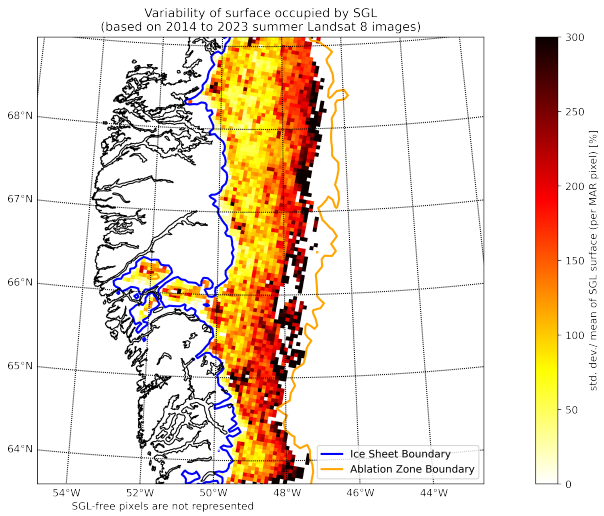
In conclusion, we have illustrated that the surface area occupied by lakes varies from year to year. Furthermore, note that this does not contradict the fact that lakes can appear in the same locations from one year to the next, as explained in Section 1.3. But this is not always the case, as the occurrence of lakes depends, of course, on the availability of melt-water, which in turn depends on climatological variables.

3.4 Meltwater Depth

Let us now focus on estimating the depth of the lakes. This section emerged from the desire to improve the model (the decision tree) used to identify pixels where the surface is covered with water (*cf.* Section 3.1.2). Indeed, in



(a) Linear regression coefficient of the area occupied by SGL for each MAR pixel. Summer means between July and September, included.



(b) Standard deviation of the area of each MAR pixel occupied by SGL, based on the same data as in (a).

Figure 3.20: 2014 - 2023 evolution of the area covered by SGL in the MAR domain.

manually creating a dataset necessary for training and evaluating a new, more efficient model, it was necessary to decide (as a human) which pixels were covered with water and which were not, based on satellite images. After some time designing the dataset, it became apparent that many pixels were in an "in-between" state, in the sense that the water coverage seemed very shallow. Naturally, the desire to assess the water depth arose. Beyond improving the pixel classification models (which could, for example, rely on a depth threshold), studying the water depth is interesting because it could quantify the total volume of surface water present on the ice sheet. This would be valuable information in itself and likely useful for modelling the SMB.

We therefore attempted to apply a technique from the literature (see below).

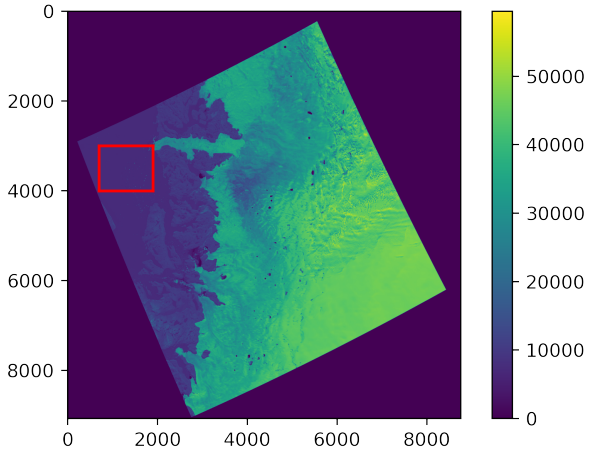
Theoretical Basis Based on the study from Pope et al. (2016), that base themselves on Philpot (1989), we know that since light rays are attenuated in water, it is possible to estimate water depth based on light measurements (*e.g.* satellite images). This can be done using Eq. (3.11) (Philpot, 1989), where z is the lake depth, R_∞ is the reflectance of deep water, R_{lake} is the lake of water pixel reflectance and A_d is the lake bottom reflectance. g depends on how the light ray is attenuated in water, it depends on the used wavelength. All terms involved in Eq. (3.11) depend on the wavelength, except z, but we will sometimes drop the subscript "$_\lambda$" for readability.

$$z = \frac{\ln(A_{d,\lambda} - R_{\infty,\lambda}) - \ln(R_{lake,\lambda} - R_{\infty,\lambda})}{g_\lambda} \tag{3.11}$$

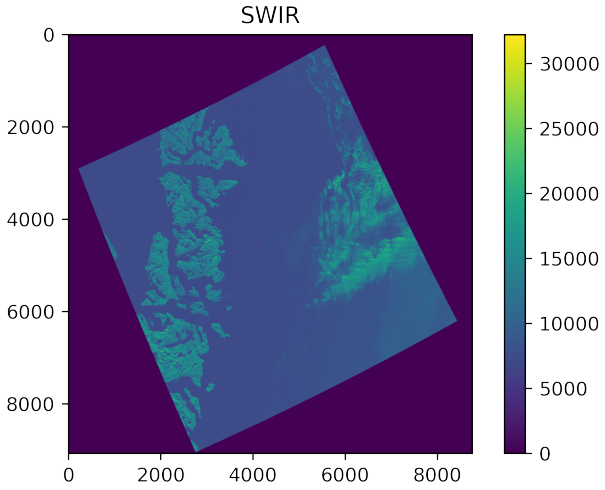
Assumptions In the first approach, we used the g coefficients provided by Pope et al. (2016) for Operational Land Imager (OLI) based on several methods (*in situ* measurements, elevation data, *etc.*). Regarding A_d, we assume that the lake bottom reflectance is the same as the surrounding ice, and we use the mean reflectance of the oceanic pixels on the same OLI image as R_∞. Also, we assume that the deviation from the vertical between the pixel and the satellite is negligible. We also neglect the potential presence of suspended matter in water as well as surface waves (Pope et al., 2016).

We therefore applied this technique to a satellite image taken in our study area.

Procedure To begin we downloaded an OLI image from 08/18/2019 over the South-West of the GrIS where loads of SGLs were observable, and with a portion of the image over the sea. We assumed that $R_\infty = 7279.3$ when using the fourth band (*i.e.* the red band, shown in Figure 3.21 (a)). This choice is based on the data shown in Figure 3.22 (a). The red band was selected because it is the most reliable according to Pope et al. (2016). It seems logical



(a) Red Band of the OLI image used for estimating lake depth. The value of R_∞ was chosen using reflectance values in the red box.



(b) SWIR Band of OLI image used for estimating lake depth.

Figure 3.21: **(a)** Red band **(b)** SWIR band.

since this band has a relatively long wavelength (which implies that it is less absorbed by water) and the sun emits a lot of it. We then made several Python scripts to build a water mask and an ice mask, using other OLI bands (Near-Infrared (NIR), Short-Wave Infra-red (SWIR), green, and blue). We notably used the Normalized Difference Water Index (NDWI), defined in Eq. (3.12), which allows identifying water-covered pixels. As shown in Figure 3.21 (b), the SWIR band enables us to identify clouds and rock. To exclude sea pixels from our water mask, we drew a polygon in QGIS and cut the raster according to it. The polygon is shown in Figure 3.23.

$$\text{NDWI} = \frac{\text{NIR} - \text{SWIR}}{\text{NIR} + \text{SWIR}} \tag{3.12}$$

We used our ice mask to create a dataset of A_d values and associated World Geodetic System 1984 (WGS 84) coordinates well-distributed over the study area. We only kept 2000 values to reduce the computation time of our algorithm (it is largely enough to cover the entire area of interest). Indeed, the computation time was directly proportional to the number of ice pixels since we made a parallelized C++ program that, for each water pixel, searched the closest ice pixel of our A_d dataset and used its radiance as A_d to compute z.

First Results With this method, we could estimate the depth of meltwater over the ice sheet. The depth over one SGL is shown in Figure 3.24. We see that our values seem relevant, even if we do not have any reliable prior information. It appears that there are `nodata` values in the middle of the SGL represented in Figure 3.24. This is because these pixels are too dark, hence too deep, which causes the difference $R_{lake,\lambda} - R_{\infty,\lambda}$ to come closer to zero, or even to be negative, and, as it is the argument of a logarithm, z cannot be computed (*cf.* Eq. 3.11). This means that the method is not suited for depth estimation above a certain threshold.

Error Propagation

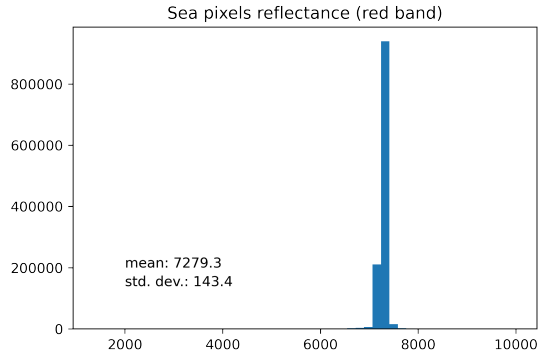
Following this, assessing the margin of error associated with our depth estimates is necessary.

In order to estimate the error associated with our meltwater depth estimation, it is necessary to derive the error propagation formula associated with the relationship in Eq. (3.11). Given a function of four variables, $f(a, b, c, d)$, the squared standard deviation around an estimated value, $\bar{f}$, is given by Eq. (3.13)

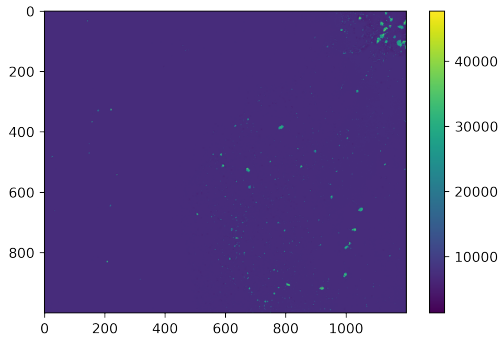
$$\sigma_f^2 = \frac{1}{N}\Sigma_{i=1}^{N}\left(f(a_i, b_i, c_i, d_i) - \bar{f}\right)^2 \tag{3.13}$$

Assuming that $\bar{f} = f(\bar{a}, \bar{b}, \bar{c}, \bar{d})$ and that (Eq.(3.14))(using a first-order Taylor approximation)

$$f(a_i, b_i, c_i, d_i) \approx f(\bar{a}, \bar{b}, \bar{c}, \bar{d}) + \Sigma_{x=\{a,b,c,d\}}\frac{\partial f(\bar{a}, \bar{b}, \bar{c}, \bar{d})}{\partial x}(x_i - \bar{x}) \tag{3.14}$$



(a) Distribution of the reflectance of the sea pixels located in the red box of the image in Figure 3.21 (a). The "outliers" (due to icebergs) were removed using a reflectance threshold of 10000.



(b) Zoom on the red box of the image in Figure 3.21 (a).

Figure 3.22: Details about the computation of R_∞

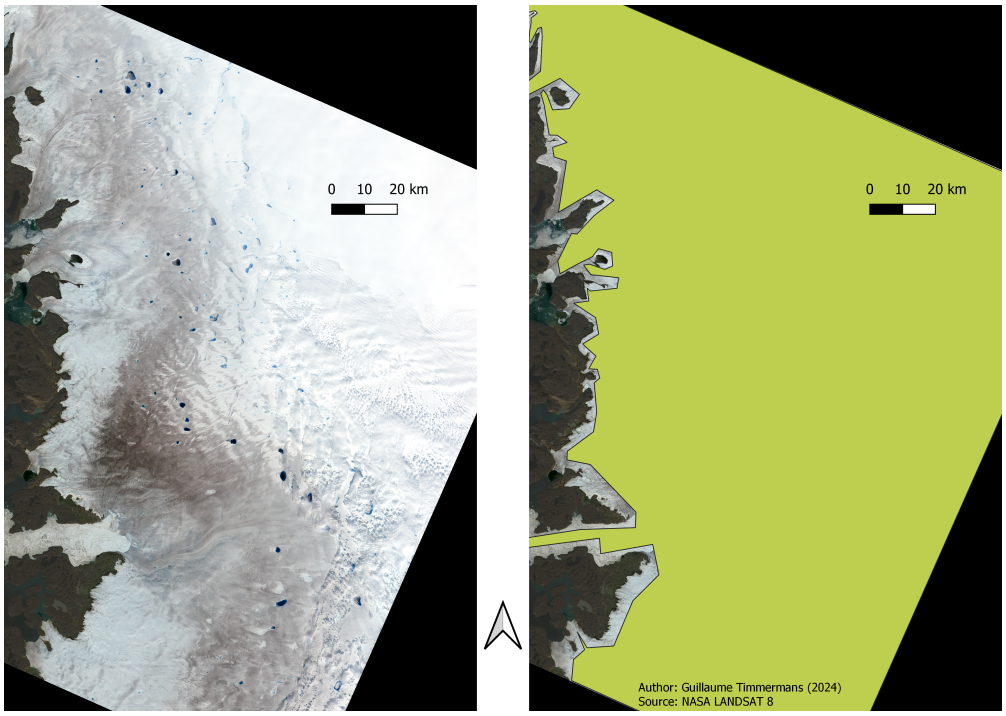


Figure 3.23: True Color image of the area of interest to evaluate SGL depth (left), and ice sheet mask (right).

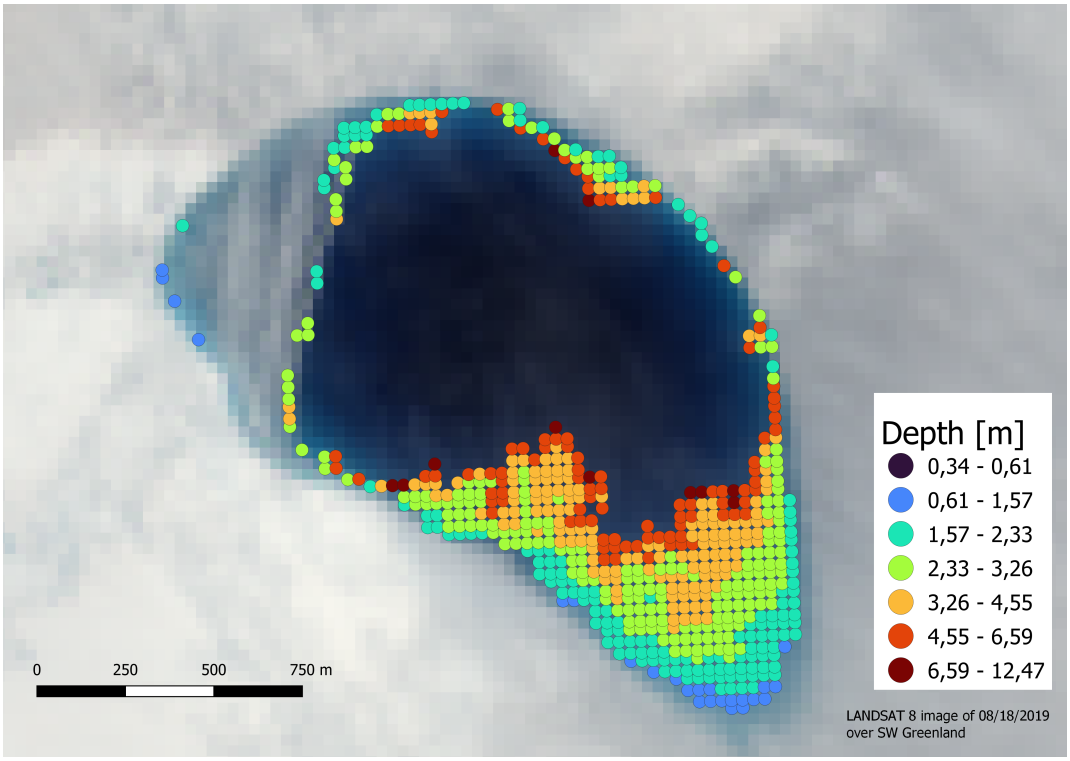


Figure 3.24: Water depth deduced using the red band of OLI.

We end up with Eq. (3.15).

$$\sigma_f^2 = \Sigma_{x=\{a,b,c,d\}} \left(\frac{\partial f(\bar{a},\bar{b},\bar{c},\bar{d})}{\partial x} \right)^2 \sigma_x^2 + \Sigma_{x \neq y} \left(\frac{\partial f(\bar{a},\bar{b},\bar{c},\bar{d})}{\partial x} \right) \left(\frac{\partial f(\bar{a},\bar{b},\bar{c},\bar{d})}{\partial y} \right) \sigma_{x,y} \tag{3.15}$$

Neglecting covariance terms, Eq. (3.15) is equivalent to Eq. (3.16), if we replace a, b, c, and d by the variables of our problem (*cf.* Eq (3.11)).

$$\sigma_z^2 = \begin{pmatrix} \sigma_{A_d}^2 & \sigma_{R_\infty}^2 & \sigma_g^2 & \sigma_{R_{lake}}^2 \end{pmatrix} \begin{pmatrix} \left(\frac{\partial z}{\partial A_d}\right)^2 \\ \left(\frac{\partial z}{\partial R_\infty}\right)^2 \\ \left(\frac{\partial z}{\partial g}\right)^2 \\ \left(\frac{\partial z}{\partial R_{lake}}\right)^2 \end{pmatrix} = \begin{pmatrix} \sigma_{A_d}^2 & \sigma_{R_\infty}^2 & \sigma_g^2 & \sigma_{R_{lake}}^2 \end{pmatrix} \frac{1}{g^2} \begin{pmatrix} \left(\frac{1}{A_d - R_\infty}\right)^2 \\ \left(\frac{1}{R_{lake}-R_\infty} - \frac{1}{A_d - R_\infty}\right)^2 \\ \left(\frac{\ln(R_{lake}-R_\infty) - \ln(A_d - R_\infty)}{g}\right)^2 \\ \left(\frac{-1}{R_{lake}-R_\infty}\right)^2 \end{pmatrix} \tag{3.16}$$

So, to estimate the error associated with our lake depth results, we must assume plausible values for the parameters $\sigma_{A_d}^2$,$\sigma_{R_\infty}^2$,σ_g^2, and $\sigma_{R_{lake}}^2$. Regarding $\sigma_{R_{lake}}^2$, we can consider that it should be close to $143.4^2 = 20534.89$, after considering what is shown in Figure 3.22, but it is less evident for the other parameters. We obtain the result presented in Figure 3.25 if we assume that $\sigma_{A_d}^2 = \sigma_{R_{lake}}^2 = \sigma_{R_\infty}^2 = 20534.89$ and that $\sigma_g^2 = 0.5^2 = 0.25\text{m}^{-2}$, as 0.5m^{-1} is the difference between the regressed, and the regressed value of g of Pope et al. (2016). We observe that, as expected, the lower the reflectance, the larger the uncertainty. This means that the method decreases in precision for increasing depth. However, for bright water pixels, the error is acceptable. For example, if we consider a pixel with $R_{lake} = 20000$ and $A_d = 35000$, which are plausible values for a trickle of water around SGL we obtain a depth of $z = 0.9737 \pm 0.0716\text{m}$. Actually, the error bar seems overly optimistic. It was obtained using assumptions for some values, and neglecting covariance terms of Eq. (3.15) plus it does not tell anything about a systematic bias due to the assumptions necessary to apply the method evoked above.

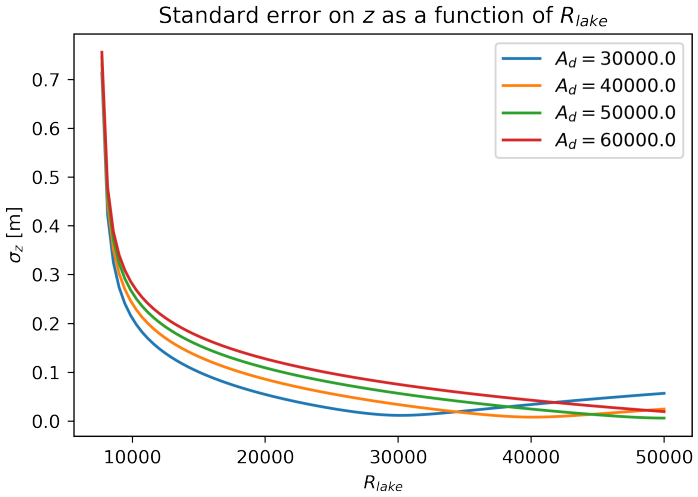


Figure 3.25: Evolution of σ_z as a function of R_{lake} for different values of A_d. The curves correspond to the square root of the expression provided in Eq. (3.11). We refer to the text for the assumptions used to obtain the plot.

Remark on the Units Let us note that since R_∞, A_d, and R_{lake} are reflectance values expressed in grey levels, they do not really have physical units. Furthermore, since the numerator of the right-hand side of Eq. (3.11) is equal to $\ln(A_{d,\lambda} - R_{\infty,\lambda}) - \ln(R_{lake,\lambda} - R_{\infty,\lambda})$, hence $\ln(\frac{A_{d,\lambda} - R_{\infty,\lambda}}{R_{lake,\lambda} - R_{\infty,\lambda}})$, the units cancel each other (which is necessary as a logarithm must take a dimensionless number as argument).

In conclusion, we attempted to apply the method of Pope et al. (2016) to estimate the depth of lakes and other water bodies on top of the ice. The results are not entirely satisfactory as the method does not work beyond a certain depth and, moreover, we are unable to provide the "error bars" associated with our results.

Further Steps Even if we are not certain about the retrieved depth using this method, all the steps used so far (data masking through NDWI, depth estimation, etc.) have been effective in classifying an image. They have successfully distinguished pixels that were covered by a relatively deep amount of water from those where the surface layer of water was very thin. This reliably classified data could help train a machine learning model capable of operating directly in GEE. Such a model would not require the determination of an R_∞, data masking, or NDWI thresholds. It could accurately identify ice pixels, water pixels, rocks, clouds, etc. Furthermore, this method could provide data to force the MAR more precisely than just providing a fraction of each pixel area covered by water (see Section 4.2.5).

Let us stress that there exist other methods to estimate meltwater depth (see for example Melling et al. (2024)), but this one was easy to implement in the first approach. Moreover, this method is the only one that can be applied at the scale of the GrIS with current data (Melling et al. (2024)). Still, the goal is not to spread too much on this subject because it is not the first object of this thesis.

Conclusion of Chapter 3 In summary, we first applied a method to locate supra-glacial lakes and estimate the area they cover. This data will be useful in the following chapter. Additionally, we confirmed that supra-glacial lakes are almost exclusively present in the ablation zone. We then attempted to link the presence of lakes to other variables, which was not entirely conclusive. We also highlighted the inter-annual variability of lake presence. Finally, we attempted to apply a technique to estimate lake depth, which provided encouraging but not entirely satisfactory results. Beyond the results, we also utilised several geomatics techniques that can be reused in the continuation of this work.

Chapter 4

Modelling of the Surface Mass Balance (SMB) of the Greenland Ice Sheet

We now come to the part of this work where we will study the modelling of the GrIS with the MAR in order to estimate its SMB. In this chapter, we start by presenting the MAR model, emphasising the parts most relevant to our study. We then explain our handling of the MAR and the numerical experiments or "sensitivity tests" we conducted. We conclude this chapter by presenting the results obtained from the corrections, additions, and modifications made to the MAR, highlighting their impact on the modelled SMB.

4.1 Methodology: Presentation of the Model & of the Reference Simulation

Regional Climate Models (RCM) Importance for Surface Mass Balance (SMB) Reconstruction and Projections

First, let us justify the interest in working with the MAR for our study (the objectives of which are presented in Section 2.1).

As evoked in Chapter 1, several approaches are developed to reconstruct and/or to project the evolution of the GrIS SMB. For example, Hanna et al. (2011) base their SMB reconstruction on statistical relationships calibrated with recent data and reanalysis that allows to go further in time. Global Climate Model (GCM) can also be used for that purpose (see e.g. Gregory and Huybrechts, 2006). In this thesis, we will mainly focus on RCM, and more precisely on the MAR, although there exists other RCM, such as Regional Atmosphere Climate Model (RACMO) (van Meijgaard et al., 2008) or Polar Mesoscale Model (Polar MM).

4.1.1 Presentation of MAR

It is because, currently, the best results in SMB modelling are obtained by RCM. They allow reconstructing the SMB evolution, based on reanalysis as well as doing projections when forced by a GCM, itself forced by "socio-economic pathways" (Riahi et al., 2017). The better performances of the RCM (with respect to GCM) are mainly due to the complex physics especially tuned for polar climates, to the full coupling with a snow model allowing to explicitly account for most of the surface-atmosphere interactions, in addition to their capacity to work at high spatial resolution (5-10km), notably allowing to explicitly take into account more local phenomena.

The *Modèle Atmosphérique Régional* (MAR) Regional Climate Model

That being said, let us describe the model.

The MAR is a RCM that was created by Gallée and Schayes (1994). It aims to downscale a simulation over a defined area to increase its spatial resolution locally. Indeed, the better spatial resolution of a RCM allows him to more precisely represent topography and to simulate local phenomena (such as katabatic winds, Foehn effect, *etc.*). Furthermore, some model parameters can be adapted as a function of the simulated domain. Actually, the MAR

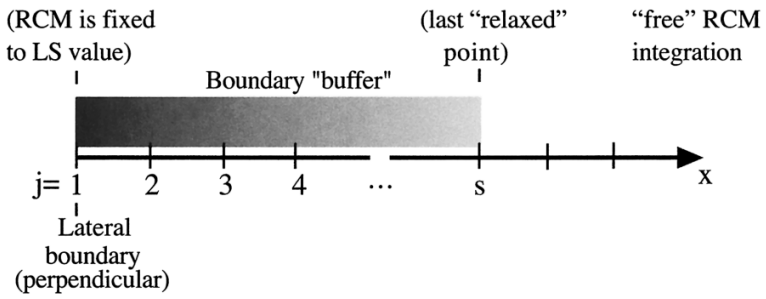


Figure 4.1: Illustration of the boundary condition integration in the MAR. LS means "Large-Scale", it refers to the large-scale forcing. Source: Marbaix et al. (2003).

has several configurations (EU for Europe, BE for Belgium, GR for Greenland, ...) some parameters vary from one version to another.

Briefly, the model (described by Gallée and Schayes (1994)) simulates the atmosphere using primitive physical equations, with the vertical coordinate being normalised pressure. Actually, we do not develop this theme in depth since the "model" that will interest us the most in this work is the MAR module that represents the interaction between the surface and the atmosphere (see below).

Because it only performs simulations on a limited region of the Earth, the MAR needs to be forced at its boundaries, either by a GCM (to downscale a climate projection), a weather forecast global model (for performing forecast) or by reanalysis (to perform hindcast).

Model Operation and Forcing From the data of forcing at the borders of its domain, the MAR simulates the state of the atmosphere and its evolution over time. The space is discretized according to an Arakawa grid. The size of the grid cells and the time step used to solve the physical equations of the atmosphere described in Gallée and Schayes (1994) can be chosen for each simulation. Let us note that spatial and temporal resolution are tied by the Courant–Friedrichs–Lewy (CFL) condition which assures that the numerical resolution is stable. For example, a time step of 30 seconds is used for a simulation with 5km×5km grid cells, 1 minute for a 10km×10km simulation, *etc.*

MAR is forced at its lateral boundaries every six hours for five variables:

1. Temperature,
2. Specific humidity,
3. Two horizontal components of the wind,
4. Surface pressure.

For large domains, the MAR is also forced at its summit for the temperature and the u & v wind components in order to prevent it from creating its own circulation (as explained in Kittel (2021)).

In practice, the boundary condition is prescribed through a "buffer zone". There are seven pixels where, for each forced atmospheric field, the model combines linearly both values (*i.e* the one computed by the MAR and the forcing value). It is done such that the transition between the MAR and the forcing model is smooth. Figure 4.1 illustrates how the boundary condition is integrated in the MAR fields.

MAR Interaction with the Surface: the Soil Ice Snow Vegetation Atmosphere Transfer (SISVAT) model

To take into account the interactions between the atmosphere and the surface, MAR is coupled to the Soil Ice Snow Vegetation Atmosphere Transfer (SISVAT) vertical one-dimensional model. This is this MAR module that will interest us the most in this thesis, as it is the centrepiece to simulate the evolution of the ice sheet state and its SMB.

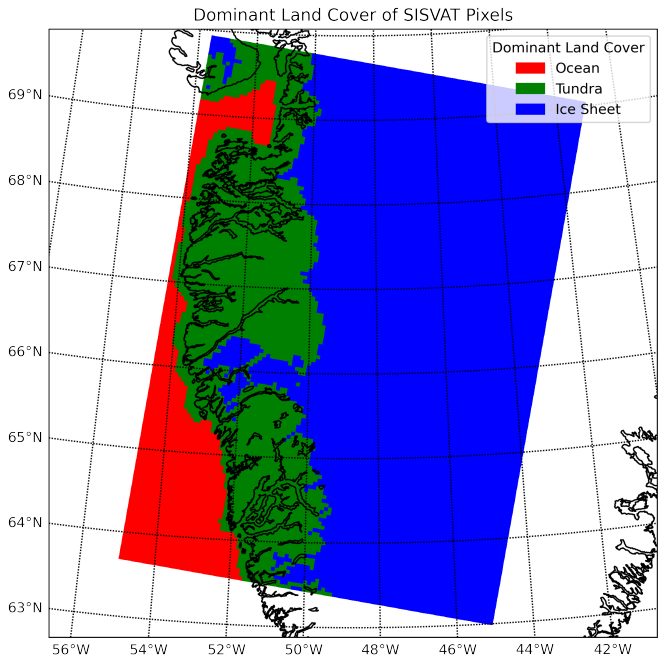


Figure 4.2: Representation of the dominant land cover for each pixel in the SISVAT model. The model's behaviour varies depending on the pixel category. As mentioned in the text, SISVAT can consider a pixel as belonging to two categories simultaneously, in given proportions.

General Operation of SISVAT At each time step, the MAR calls the SISVAT module, whose main code is written in `phy_sisvat.f90`. This code links the atmosphere and surface parts and it calls for every pixel the `sisvat.f90` program, which contains the one-dimensional surface model. `sisvat.f90` calls itself other modules such as `sisvat_tso.f90` or `sisvat_qsn.f90` that deal with specific processes (for example, `sisvat_tso.f90` is responsible for the soil/snow energy balance). They share common variables that are declared in other Fortran files. Given that pixels are treated independently, it allows SISVAT to be parallelised.

Let us remark that SISVAT can be called up to two times per time step for some pixels. Indeed, without fully explaining how the program works, (see e.g. Kittel (2021)), it is worth mentioning that MAR can consider that a pixel is covered by several surface types (*e.g.* 70 % of snow/ice and 30 % of tundra), and that this proportion can vary during the simulation (e.g. the ice sheet area can decrease a bit during summer, *etc.)*. To do so, it calls its SISVAT (*i.e.* `sisvat.f90`) module two times with different parameters (relative to the ocean and to the vegetation) (from the `phy_sisvat.f90` program) then it weights the output of the two execution and use the result as input boundary conditions for the atmosphere part.

We show in Figure 4.2 the dominant surface occupation for each pixel on July 1st of the "reference simulation" (we explain what we mean by this in more detail below, in Section 4.1.2). It is also worth noting that SISVAT can account for surface ice above an oceanic pixel (i.e., sea ice).

One-Dimensional Representation of the Ice/Snow Cover Let us now focus on how SISVAT represents the snowpack (remember, the model represents the top twenty meters of the snow/ice layer).

Fettweis (2006) describes SISVAT snow-ice model as an "*one-dimensional multi-layered energy balance model that determines the exchanges between the sea ice, the ice sheet surface, the snow-covered tundra, and the atmosphere*". The ice/snow representation is based on the CROCUS model (Brun et al., 1992). As explained in Brajkovic (2022), CROCUS, represents the snow/ice cover as the superposition of several layers of varying thickness, generally increasing downwards. The number of layers as well as their thickness is susceptible to vary at each time step.

Reijmer et al. (2012) describes the CROCUS model, which is coupled to the atmosphere through heat exchanges (see Figure 2.3 of Fettweis (2006)), notably involving the surface albedo that is computed depending on the snow

grain size, the solar zenith angle, the surficial water height, *etc.* (see Fettweis (2006)). Over snow or ice, the surface temperature is blocked to 273.16 K (so the left-hand side of Eq. (4.1) is nil), and the excess of heat contributes to melting or freezing (Reijmer et al., 2012), which means that all the interstitial water should be frozen before the temperature can decrease. The surface energy budget is given by Eq. (4.1) (Reijmer et al., 2012):

$$\rho c_i \frac{\partial T_{sn}}{\partial t} = \frac{\partial}{\partial z}\left(K \frac{\partial T_{sn}}{\partial z}\right) + L_f F + L_f M + \frac{\partial Q}{\partial z} \tag{4.1}$$

Where

- T_{sn} is the snow/ice temperature;
- c_i is the ice heat capacity;
- $K = 2.22\left(\frac{\rho}{\rho_w}\right)^{1.88}$ (ρ_w is the density of liquid water) (Yen, 1981);
- $L_f F$ & $L_f M$ are heat fluxes related to freezing and melting, respectively;
- Q is the effect of radiation (only short-wave penetrates beyond the first layer).

Each layer is characterised by a proportion of liquid water, whose evolution is described by Eq. (4.2) (Reijmer et al., 2012):

$$\rho \frac{\partial W_r}{\partial t} = \frac{\partial}{\partial z}(-U_w) + F + M \tag{4.2}$$

Where W_r is the liquid water proportion, U_w is the vertical water flux, F, and M are the freezing/melting rates of water in the layer.

Also, the model represents the snow metamorphism, *i.e.*, the densification of snow due to the pressure exerted by the upper layers.

Simulation of Meltwater Runoff in SISVAT Let us now turn to the simulation of the meltwater flow present on the surface of the ice layer in the model.

As explained in Lefebre et al. (2003), the SISVAT model evacuates the melt water through runoff. The runoff rate is based on Zuo and Oerlemans (1996) and is computed as follows:

$$\frac{dW_r}{dt}_{runoff} = -\frac{W_r}{t^*} \tag{4.3}$$

$$t^* = c_1 + c_2 \exp\left(-c_3 S\right) \tag{4.4}$$

Where W_r is the surface water height or the internal liquid water content in excess of the maximum water saturation, S is the surface slope, c_1, c_2, and c_3 are constants respectively set to 0.33 days, 25 days and 140 in the study of Lefebre et al. (2003). Equation (4.3) leads to $W_r(t) = A \exp\left(-t/t^*\right)$ with A equal to the surface water height at time $t = 0$: $A = W_r(0)$. t^* is the "turnover time" of the surface water with respect to the runoff flux, it is the time necessary to decrease by a factor e the surficial water through runoff, as shown in Figure 4.3.

Let us note that Equation (4.3) does not consider other fluxes of liquid water (than runoff) that influence the surface water height. These fluxes do not depend on W_r and make the system nonlinear. Those fluxes are (*cf.* Figure 1.3):

1. Rain;
2. Snow and ice melt;
3. Refreezing[1];

[1] An error in the code prevented this process from occurring, which we have resolved (see further details below).

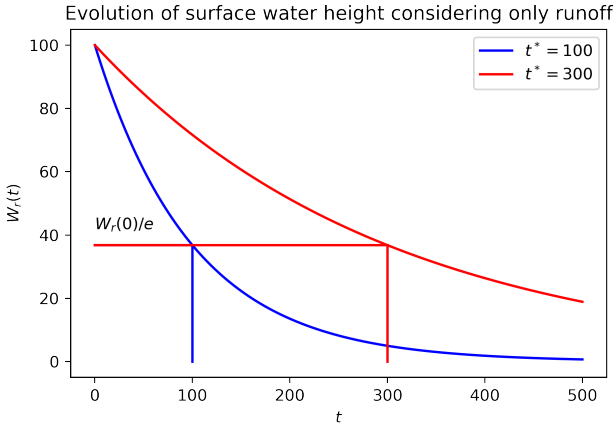


Figure 4.3: Influence of t^* on the runoff flux in the model from Zuo and Oerlemans (1996).

4. Evaporation[2];

5. Condensation.

Several simulations with different values of c_1 and c_2 (*cf.* Equation(4.4)) were run to assess these parameters' influence over the system. All simulations started on May 1, 2019, with the same initial conditions (that of the reference simulation). The results are presented in Section 4.2.2.

Without calling into question the flow model described by Equations (4.3) and (4.4), one of the defects of SISVAT is that this runoff water disappears from the model once it has flowed. In other words, the model subtracts a runoff term from surface water at each time step for each pixel, without adding it anywhere else. So the SISVAT model calculates surface water (being the budget of the fluxes enumerated above (except that it neglects evaporation)) and then subtracts a runoff term calculated following Equations 4.3 and 4.4. But it does not add that trickle water to the amount of surficial water of a pixel that is located lower in altitude as it happens in reality (as well as infiltration), leading to the formation of melt ponds. This is because, as mentioned earlier, SISVAT models each pixel as an independent entity, (so the discretization of the snowpack varies from one pixel to another, for example), allowing the model to be parallelised, which is of great interest since most applications of the model are computationally expansive (e.g., long-term projection).

Furthermore, it is obvious that the "runoff law", described by Equations (4.3) and (4.4) is not always a perfect representation of reality, since lakes, where one can assume that water stagnates, are formed in places.

4.1.2 Reference simulation

Motivations

This thesis aims to quantify the impact of surficial water on the SMB through parametrisation of the MAR. To achieve this goal, running a "reference simulation" is necessary. It will provide initial conditions and serve as a basis for comparison with other simulations (which will be sensitivity tests, in which case we can refer to them as "sensitivity simulations" (we will also refer to them as "simulations" when the context is clear), "numerical experiments", or model corrections). Given the focus on hydrological processes occurring at the edges of the GrIS, the simulation does not need to cover the entire ice sheet. The intention to limit the calculation time also justifies our need for a simulation on a limited area. Indeed, given that our work involves doing a lot of MAR runs, it would be incensed to work on the entire GrIS.

[2]Actually, SISVAT do not take into account the evaporation of surface meltwater over the ice, nevertheless, we experiment with adding it in this thesis.

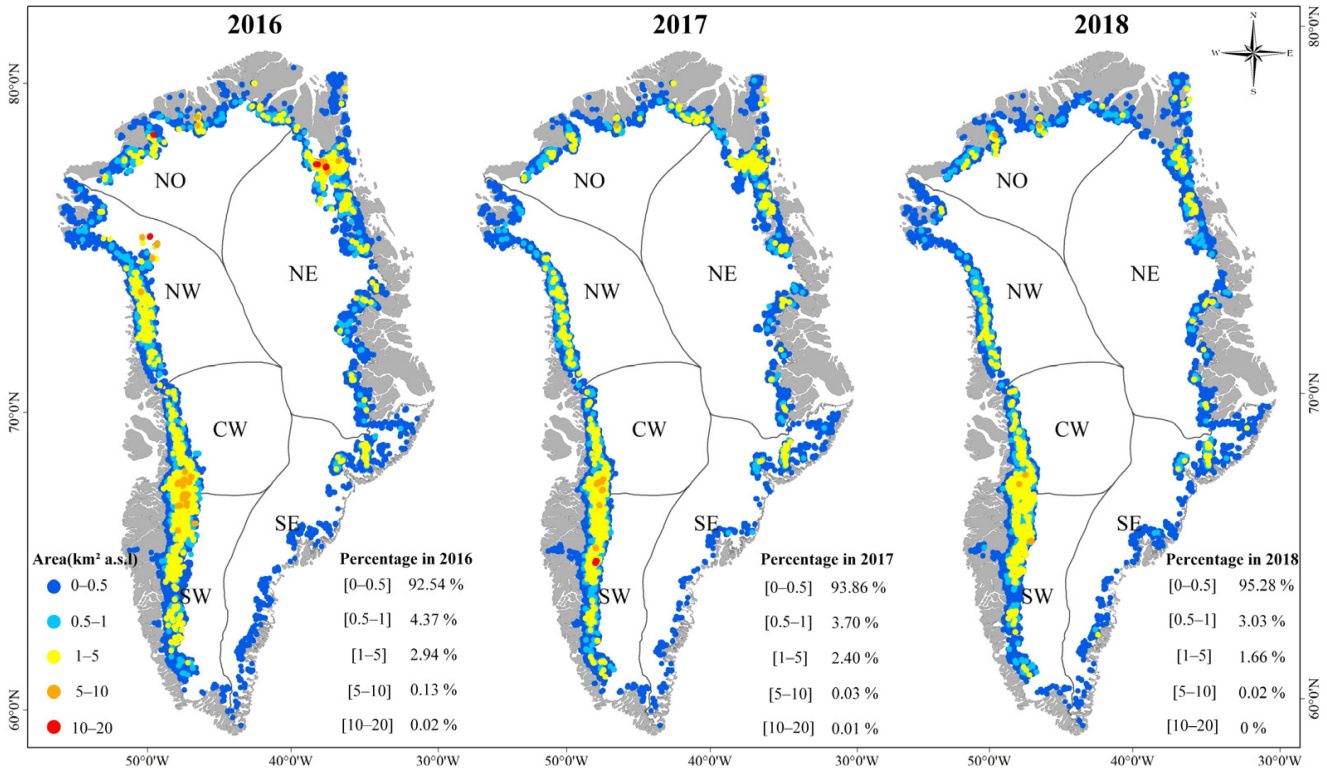


Figure 4.4: Distributions of the supraglacial lakes during the 2016, 2017 and 2018 melt seasons. These results come from the analysis of satellite images. Source: Hu et al. (2021).

MAR Domain

Let us now justify why we have chosen the study area already used in Chapter 3.

Figure 4.4 shows that there are many melt ponds in South West Greenland, and it is consistent with Figure 3.19. So we chose this zone of Greenland to perform our simulations. Indeed, since our goal is to study the impact of SGLs on the SMB, this area appears to be highly relevant. Due to the modest size and irregular spatial distribution of melt ponds (*cf.* Figure 3.2), working at a relatively high resolution is advisable. Therefore, we opted for pixels of size 5km × 5km. To adhere to the CFL condition, the associated time step is set to 30 seconds. Since MAR employs a "buffer zone" where atmospheric fields from the forcing model (ERA-5 in our case) and MAR are averaged, it is essential to avoid strong outward winds or wind divergence in this zone. Consequently, to prevent issues with katabatic winds, we tilted the domain (with respect to local meridians) so that its left boundary corresponds to the ridgeline of the ice sheet.

The domain we opted for and its topography can be seen in Figure 4.5. Fifteen pixels of "buffer zone" have been removed on each side of the domain. Moreover, we previously showed Figure 4.2, which also depicted our MAR domain, but without removing the pixels from the "buffer zone".

Forcings and initial conditions

As mentioned in Section 4.1.1, it is necessary to force the model with data either coming from reanalysis or from a GCM. We opted for ERA-5 reanalysis (from European Centre for Medium-Range Weather Forecasts (ECMWF)), which is more accurate than National Centers for Environmental Prediction (NCEP) and the National Center for Atmospheric Research (NCAR) (NCEP/NCAR) reanalysis for atmospheric fields.

Concerning the ice cap, which is simulated by the SISVAT module that is fully coupled to the MAR model, it is essential to provide initial conditions. Indeed, SISVAT, which simulates the state of the top 20 meters of snow or ice (and thus not the entire thickness of the ice sheet, which can reach several kilometres in some places), requires initial

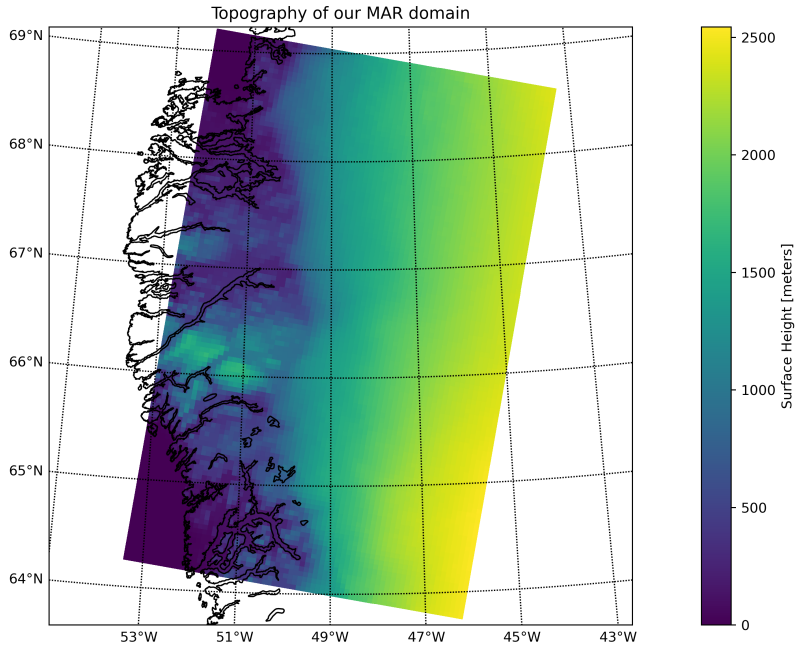


Figure 4.5: Surface height of our MAR domain, without the "buffer zone".

conditions. Without these, it is estimated that ten years of simulation are needed for the model to independently create a snow cover close to reality, thereby mitigating the influence of the initial conditions. Therefore, to limit the computation time and to start our simulation with a (modelled) snow mantle close to reality, we had to use the results of a previous MAR simulation over the whole Greenland Ice sheet at a lower resolution (10km $\times$ 10km) as initial conditions for the ice sheet. This "forcing" simulation is itself forced by reanalysis (ERA-5). These reference results are available in Fettweis (2023).

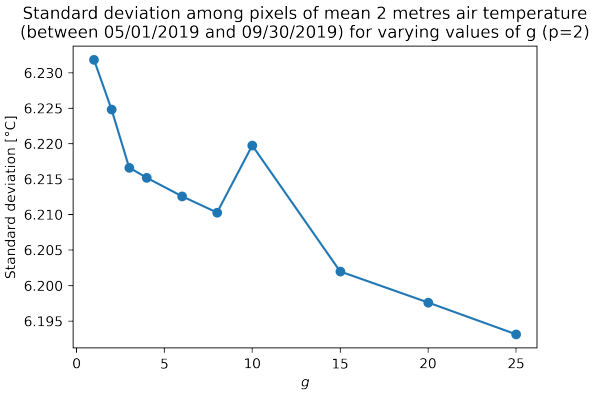
Period of Reference

The reference simulation starts in September 2018 and ends in September 2019. To avoid having results dependent on initial conditions (since we do not have initial conditions for the atmosphere), we base our analyses on the period from May 2019 to September 2019, which roughly corresponds to the melting season. 2019 was a record year in terms of melting (Tedesco and Fettweis (2020); Fettweis (2023)), which implies that the hydrological processes we study were especially important. Note that working over only a few months allows us to limit the calculation time, which is interesting because, as already mentioned, we will conduct several numerical simulations at relatively high resolutions.

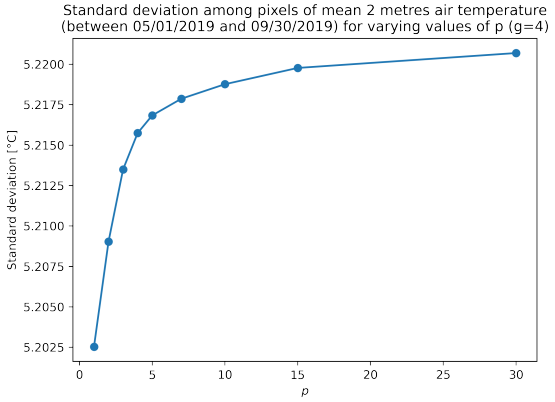
Data Processing to Assess the Reference Simulation

In order to verify whether our reference simulation at a 5km $\times$5km resolution provides acceptable results, we compare it to the results of the simulation which was used as initial conditions for the state of the ice sheet. MAR outputs are spatial data, provided for each pixel of coordinate (`i`, `j`) of a grid and for each time t (also noted `l`). If the variable also depends on the height the associated coordinate is denoted `k`. All results are stored in Network Common Data Format (NetCDF) files. Some pre-processing steps are necessary to compare both simulations' results. We briefly present them in the following.

Interpolation To be able to compare both simulations with our simulations, we need to use a common reference frame. In other words, it is necessary to interpolate a gridded dataset onto another grid. Indeed, it allows computing (and plotting) differences per pixel, correlations, *etc.* To do so, a common method is Inverse Distance Weighting



(a) Standard deviation among pixel temperature value if we interpolate the high-resolution simulation's results onto the "low-resolution" (10km×10km) simulation's grid with varying values of parameter g.



(b) Standard deviation among pixel temperature value if we interpolate the high-resolution simulation's results onto the "low-resolution" (10km×10km) simulation's grid with varying values of parameter p.

Figure 4.6: Effect of parameters g and p on the interpolation. We interpolate the mean 2 metres temperature during the period of reference of the high-resolution simulation onto the forcing simulation's grid.

(IDW) interpolation. For each pixel of the reference grid, the value is computed as follows

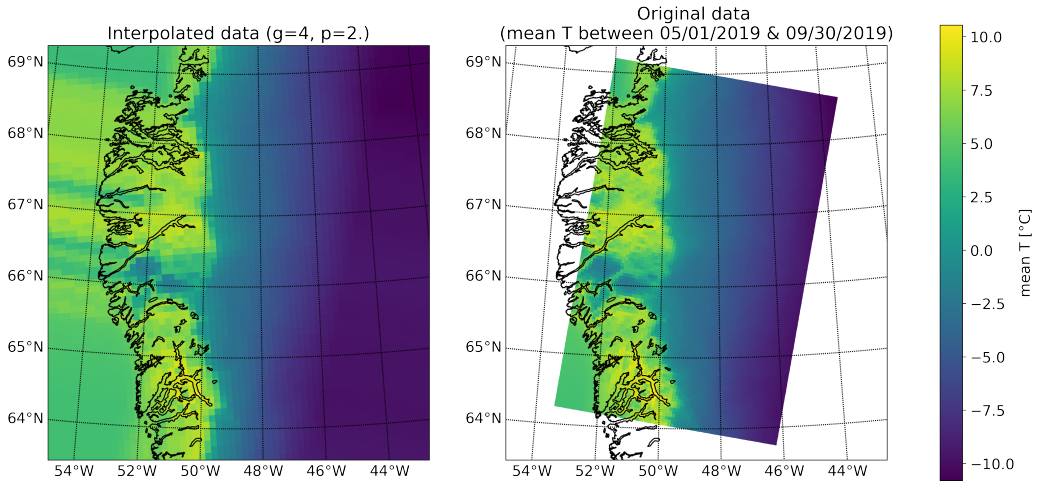
$$\mathrm{val}_{destination\ grid}(i,j) = \frac{\Sigma_{k=1}^{g} \mathrm{val}_{origin\ grid,\ k}(1/d_k^p)}{\Sigma_{k=1}^{g}(1/d_k^p)} \tag{4.5}$$

Where $\mathrm{val}_{destination\ grid}(i,j)$ is the interpolated value of the first dataset on the reference grid, for the pixel of coordinates (i,j), whereas $\mathrm{val}_{origin\ grid,\ k}$ and d_k $(k = \{1,2,...,g\})$ are respectively the values of the g closest pixels (with respect to the centre of the pixel (i,j) of the reference grid) of the origin grid, and the associated distances. Parameter p is the ponderation of the interpolation. Of course, the larger p, the stronger the influence of the closest pixel on the interpolated value ($\mathrm{val}_{destination\ grid}(i,j)$), compared to more distant pixels, among the g nearest pixels.

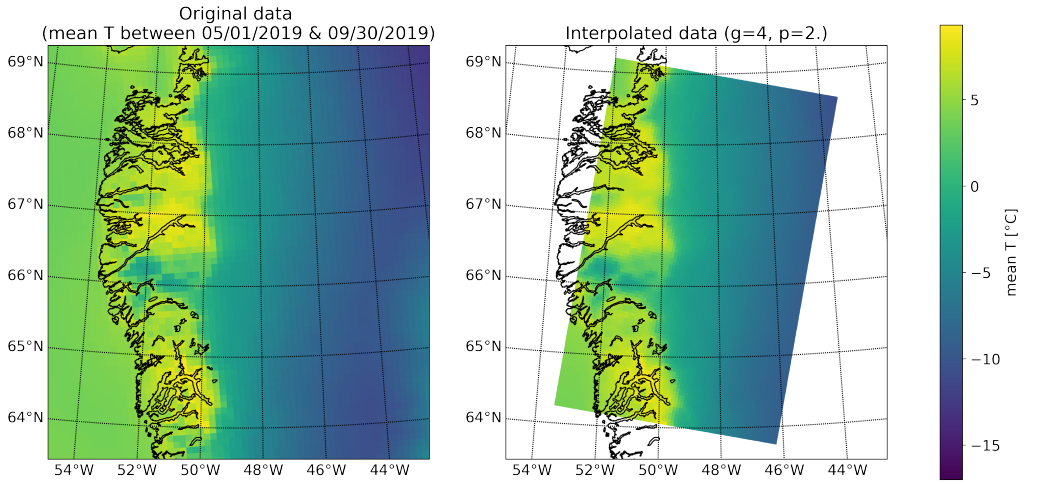
A Fortran program was written to perform the IDW interpolation calculation and write the results in a NetCDF file. Before comparing our reference simulation to the forcing simulation, we can interest ourselves in the interpolation itself and in particular the effect of p and g parameters (*cf.* Eq. (4.5)) on the result. Figure 4.6 shows the influence of these parameters on the standard deviation of the mean temperature interpolated from the 5km×5km resolution simulation grid to the 10km×10km resolution simulation grid. As expected, the interpolation is "smoother" when p decreases or when g increases (for a fixed value of p). The effect of g is shown in Figure 4.6 (a) and the effect of p is shown in Figure 4.6 (b).

Let us note that we can choose to interpolate the results of our simulation over the southwest of the GrIS onto the grid of the "low-resolution" (10km×10km) MAR simulation (that provided the initial conditions for the other), and vice versa, this is represented by the two pairs of maps in Figure 4.7. Figure 4.7 (a) represents the interpolation of the simulation over the southwest Greenland data onto the reference grid, while Figure 4.7 (b) represents what is obtained by the reverse operation. However, since the resolution of our simulation (*i.e.* the one over the southwest of Greenland) is higher than the "low-resolution" (10km×10km) simulation's one, it is better to interpolate our results onto the 10km×10km-resolution grid. Performing the opposite leads to "problems" in regions where some variables vary greatly from one pixel to another. This is particularly the case on the edge of the ice sheet where the altitude varies greatly over a small distance when moving in the direction perpendicular to the coast.

Indeed, as shown in Figure 4.8, it is better to work on the "low-resolution"(10km×10km) simulation's grid. This is because since the other MAR simulation has a higher resolution, it better represents spatial variations of a variable. The IDW interpolation does not improve the spatial resolution of data, so there will always be a precision deviation where gradients are high. This is exactly what is represented in Figure 4.8. At the edge of the inlandsis, where the altitude, like the temperature, varies a lot over small distances, these two variables being connected, we



(a) Interpolation of our simulation's data (right) onto reference grid (left). The interpolated variable is the average temperature between May 1 and September 30, 2019.



(b) Interpolation of reference data (left) onto our simulation's grid (right). The interpolated variable is the average temperature between May 1 and September 30, 2019.

Figure 4.7: The two maps at the top (Figure 4.7 (a)) represent the interpolation of the high-resolution (5km×5km) simulation results to the "low-resolution"(10km×10km) simulation's grid while the two maps below (Figure 4.7 (b)) represent the interpolation in the "opposite direction". We refer to Eq. (4.5) for the meaning of p and g.

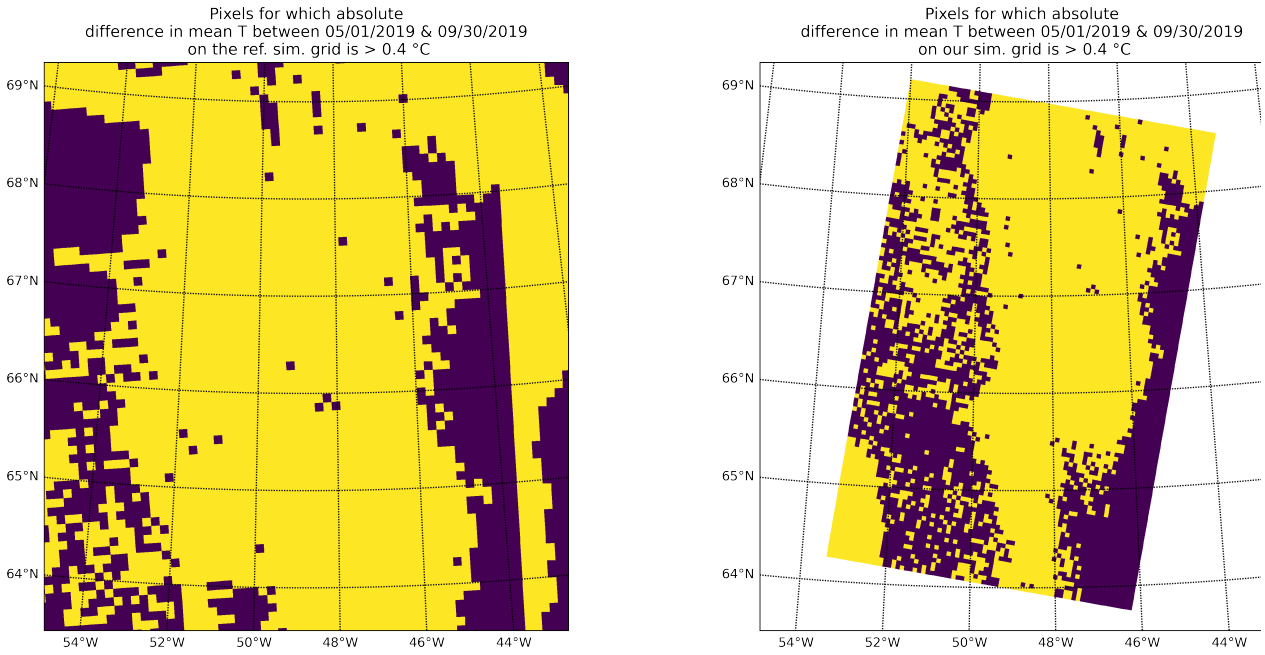


Figure 4.8: Representation of pixels for which the difference in mean temperature between May 1 and September 30, 2019, is higher than 0.4°C (in purple (if the difference is lower, the pixel is in yellow); therefore, *the purple pixels are those where the error (, the anomaly) is significant, greater than 0.4°C.*). We chose the threshold of 0.4°C because it represents the standard deviation of the absolute differences between the simulation at a 5km×5km resolution and the one at a 10km×10km resolution when we interpolate the data from the 10km×10km resolution simulation to "our" simulation over the southwest of the GrIS. The left map represents what is obtained if we interpolate our simulation's results onto the 10km×10km resolution grid (that we also denote the "reference grid"). The right map corresponds to the other case. The coastline is not represented to improve visibility. We chose to represent whether the difference is higher or lower than a threshold instead of representing the absolute difference using a continuous colour map because the large errors outside the MAR domain when working on the 10km×10km grid (up to 20°C) make it impossible to see errors inside the 5km×5km MAR domain with a continuous colour scale.

observe more difference between the two datasets, if the one with the lowest spatial resolution is interpolated to the most accurate.

Of course, if we interpolate our simulation's data on the reference grid, the results are disastrous outside our domain, as can be seen in Figure 4.8. This is obviously because there is no data outside the domain (we must therefore talk about extrapolation rather than interpolation). These poor results are not problematic since we are only interested in what happens inside the MAR domain over the southwest GrIS.

Data Masking Performing a naive interpolation leads to surprising results near the tundra, on the West side of the chosen domain. Indeed, it is necessary to exclude certain pixels via a mask because otherwise, one would risk for example taking into account values modelled on the tundra or the ocean, in the calculation of the interpolation, for the pixels at the edge of the ice sheet. If it's not too bad for some variables like temperature, it is a serious bias for others, such as SMB for example. Therefore, we used a SISVAT variable as a mask: the percentage of each pixel area that is occupied by ice, which we discussed in detail in Appendix D. The associated threshold was 50%. It should also be noted that for the calculations and figures that follow, we have also excluded pixels outside the domain, based on a distance criterion, in order to remain within the framework of interpolation, and not extrapolation.

4.2 Results

We now arrive at the part of this work where we will present the main results obtained from our "experiments" and sensitivity tests using the MAR model. Before that, we begin with an evaluation of our reference simulation reliability, building on the discussion from the previous section.

4.2.1 Assessment of the reference simulation

Comparison to a MAR simulation over the whole Greenland Ice Sheet (GrIS)

Now that we have presented the interpolation and justified the choice to interpolate the data from our simulation to the grid of the reference simulation (which provided the initial conditions of our simulation), we must check if the results of our simulation are consistent. In order to avoid any confusion, note that in this section, we consider that the "reference simulation" is the forcing simulation, at 10km × 10km resolution, but subsequently, as we will no longer use it, the "reference simulation" will refer to the simulation on the southwest of Greenland, without modification to the MAR code (*i.e.* the one we interpolate), hence the title of this section (Section 4.2.1).

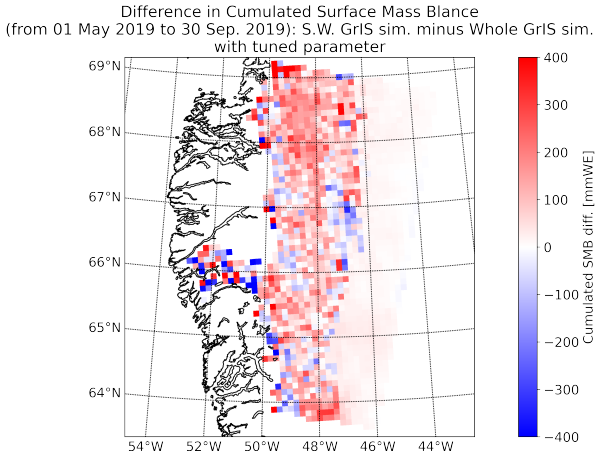
As some of the variables linked to the SMB (including the SMB itself) are essential for our study, we will focus on these for the assessment of our first MAR simulation. But before that, let us note that, without going into too much detail, it is possible to adjust certain parameters of the MAR to improve its performance. These are often referred to as "magic parameters". After analysing the results of our simulation over the southwest of the GrIS, we reran a simulation over the same domain (with the same initial conditions, *etc.*) by modifying a "magic parameter": we artificially increased the humidity by 2.5% at the domain boundaries. After comparing the base simulation with the new one, we concluded that the simulation with increased boundary humidity led to better results, so after this section, we will refer to this "tuned parameter simulation" as the "reference". The comparison between these two simulations is presented in Appendix G. Below, we compare the reference simulation (with a 10km×10km resolution) and the second simulation we conducted (*i.e.* with the "tuned magic parameter").

In Figure 4.9, we compare the reference simulation (at a resolution of 10km×10km) to our "tuned" humidity parameter simulation. Specifically, we show, for each pixel, the absolute difference between our simulation and the "reference" for the following four variables: SMB, SF, RF, and the melt-water production that are all cumulative from May to September 2019, including. We remark (in Figure 4.9 (a)) that our simulation overestimates the SMB (*i.e.* it underestimates the loss of mass, as the SMB is negative in the domain during summer). As shown in Figures 4.9 (b), there is a slight overestimation of snow precipitation in our simulation. Figure 4.10 puts these discrepancies into perspective by comparing the cumulative SF from the two simulations. It appears that the spatial variability of snowfall is the same for both simulations, but the discrepancies noted earlier are not entirely insignificant. Furthermore, it is reassuring to see that there is not an overestimation across the entire domain; instead, we observe the opposite in a small area at the edge of the ice sheet. This highlights the importance of examining differences (or anomalies) in terms of *relative* differences, as we do below (in the discussion of Figure 4.12). Before proceeding, note that Figure 4.9 (c) indicates that there do not seem to be significant issues related to precipitation (in terms of rain, not snow), while Figure 4.9 (d) aligns with our earlier observation that our simulation overestimates the SMB, as it shows that melt is slightly underestimated in the 5km×5km-resolution simulation.

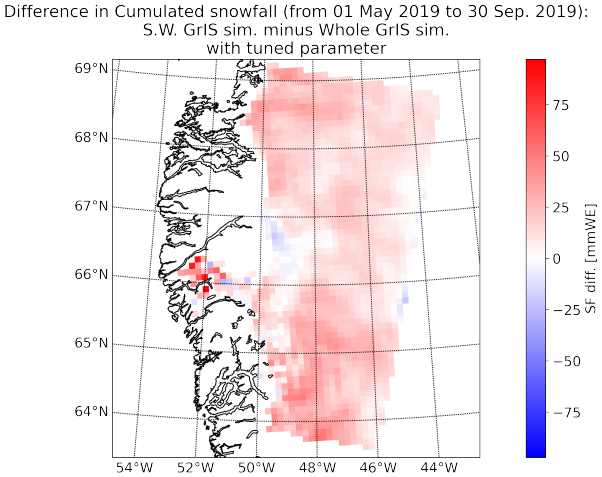
To get an idea of the orders of magnitude, Figure 4.11 represents the cumulative SMB for the two simulations that we compare (and not their difference). It is shown that the cumulative SMB is of the order of 1000 mmwe during the period of interest. The Appendix E contains similar Figures for the other variables.

For the reasons mentioned above, we can now turn our attention to relative anomalies, as this allows us to put the errors into perspective and assess whether they are significant or problematic. To avoid any confusion, let us clarify what we mean by the term "relative anomaly" (or "relative difference") especially since we will be using this indicator multiple times. Consider a given quantity X (such as SMB or temperature) simulated by a reference simulation A (leading to a value of X_A) and another simulation B (providing a value X_B). The relative difference of the quantity X between simulations A and B is given by (Eq. (4.6)):

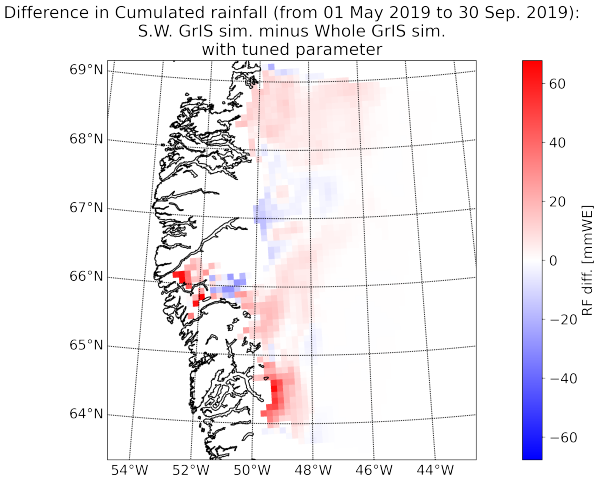
$$\text{Relative Difference } [\%] = 100\% \times \frac{X_B - X_A}{X_A} \tag{4.6}$$



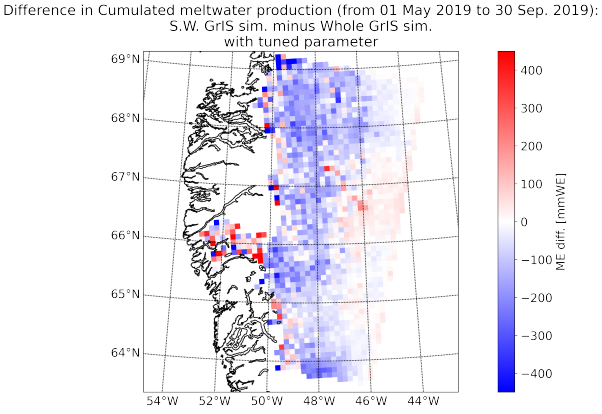
(a) Difference in the cumulative SMB over five months: our tuned simulation minus the reference simulation.



(b) Difference in the cumulative SF over five months: our tuned simulation minus the reference simulation.



(c) Difference in the cumulative RF over five months: our tuned simulation minus the reference simulation.



(d) Difference in cumulative meltwater production over five months: our tuned simulation minus the reference simulation.

Figure 4.9: Difference between our simulation with an increased humidity at the boundary of the domain and the reference one for four variables linked to the SMB. All differences are expressed in mmwe.

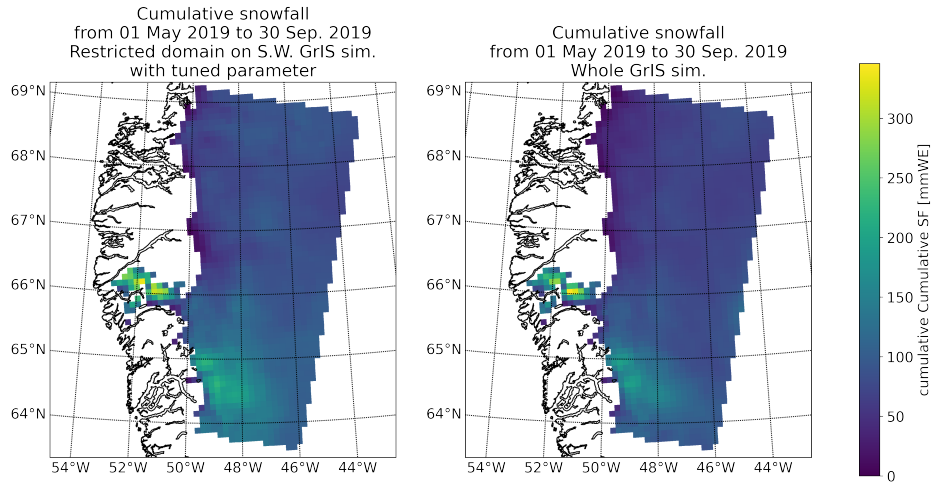


Figure 4.10: Cumulative snowfall from May to September 2019 for the two MAR simulations (both forced by ERA-5 reanalyses) compared here: the simulation on the smaller domain in the southwest of the GrIS with a resolution of 5km×5km (left) and the simulation on the entire ice sheet with a resolution of 5km×5km (right).

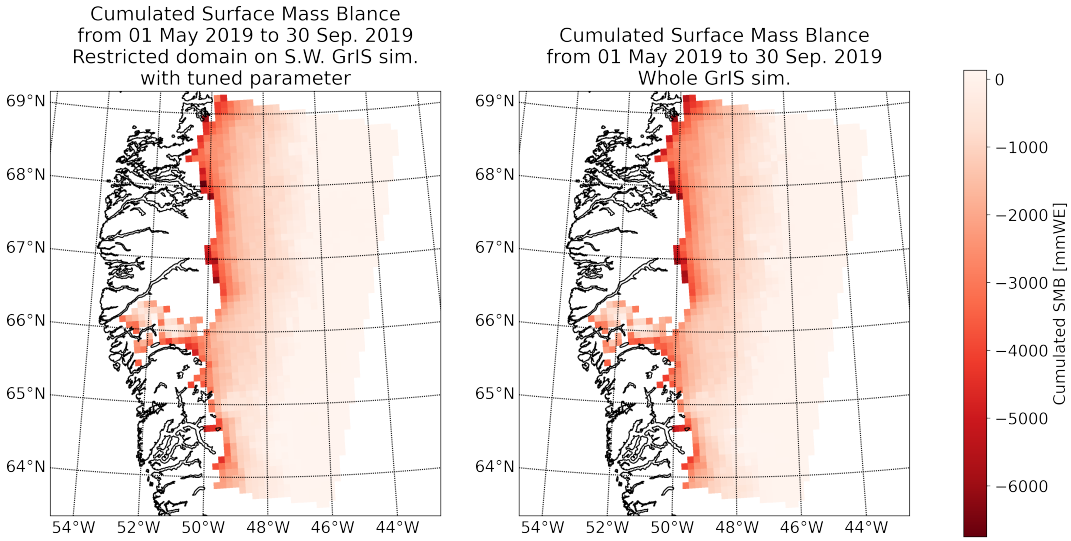
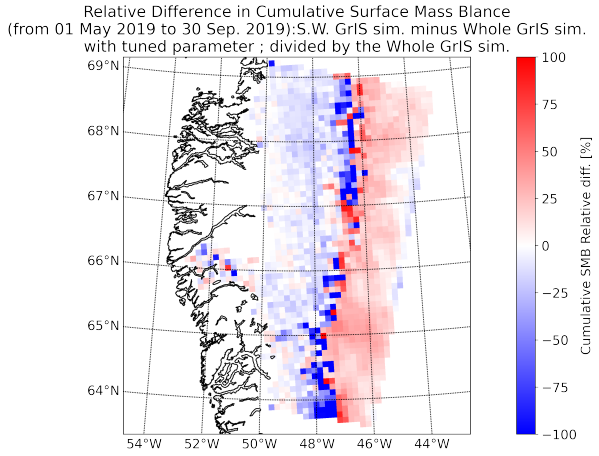
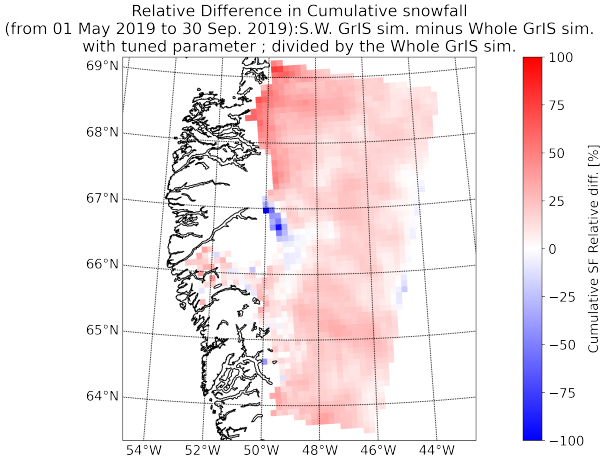


Figure 4.11: Cumulative SMB between May and September for our simulation with a "tuned" parameter (at 5km×5km of resolution) (left), and the "reference" simulation (at 10km×10km of resolution) (right).

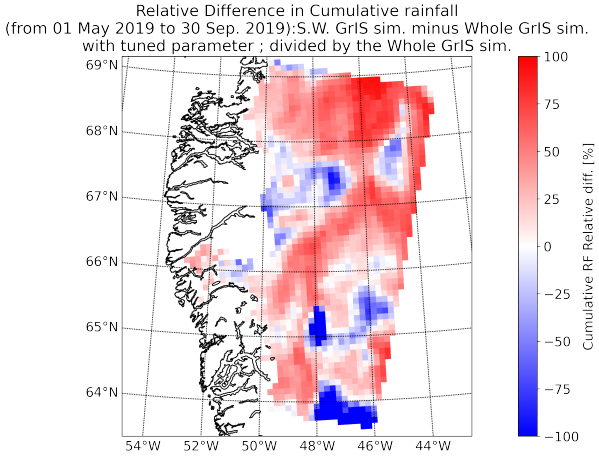
Figure 4.12 shows the relative anomalies of the four variables used to compare the two studied simulations. We observe that the relative anomalies of SMB are very small (i.e., between -10% and +10%) in the ablation zone. Some pixels exhibit extreme values, but this is solely due to low values of the denominator (close to the "equilibrium line", i.e. where SMB = 0). We note that the relative differences in cumulative snowfall are around 25% to 50% in the worst cases, although most pixels show acceptable values. The relative differences in rainfall are more significant. Finally, the results in terms of melting are acceptable.



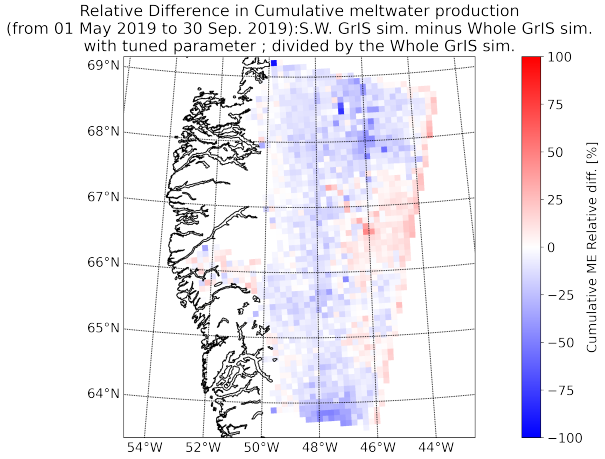
(a) Relative difference in the cumulative SMB over five months: our tuned simulation minus the reference simulation, divided by the reference simulation.



(b) Relative difference in the cumulative SF over five months: our tuned simulation minus the reference simulation, divided by the reference simulation.



(c) Relative difference in the cumulative RF over five months: our tuned simulation minus the reference simulation, divided by the reference simulation.



(d) Relative difference in cumulative meltwater production over five months: our tuned simulation minus the reference simulation, divided by the reference simulation.

Figure 4.12: Relative Difference between our simulation with an increased humidity at the boundary of the domain and the reference one for four variables linked to the SMB. We show relative anomalies, that we multiply by 100, so the units are %. By "reference", we mean the 10km×10km MAR simulation over the whole GrIS. The colour scale ranges from -100% to 100%. Some pixels may contain more extreme values, but this is generally due to low denominator values (e.g., an SMB anomaly divided by an SMB close to zero).

Regarding the relatively high differences for cumulative rainfall, we can find some reassurance by looking at Figure 4.13, where we observe that the amount of rainfall is less important than those for snowfall (*cf.* Figure 4.10), which means it less critical to have an error in this term.

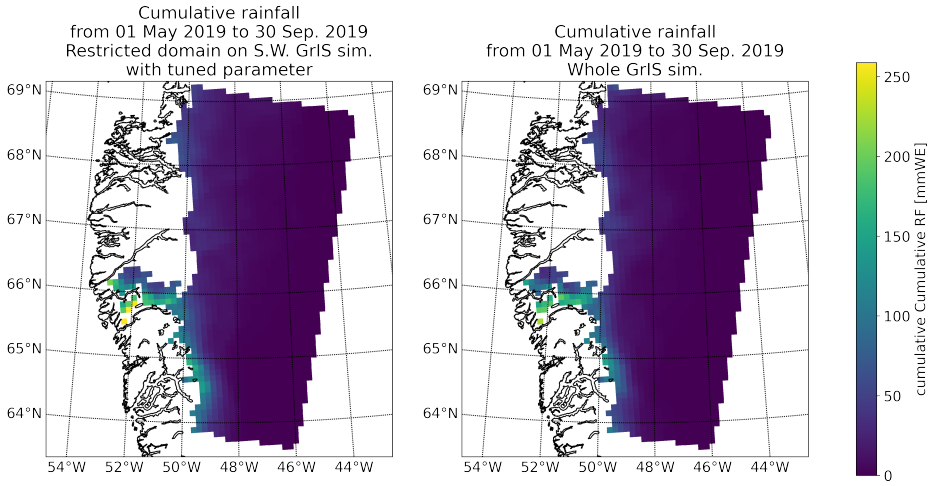


Figure 4.13: Cumulative rainfall from May to September 2019 for the two MAR simulations (both forced by ERA-5 reanalyses) compared here: the simulation on the smaller domain in the southwest of the GrIS with a resolution of 5km×5km (left) and the simulation on the entire ice sheet with a resolution of 5km×5km (right).

Comparison with ERA-5

As explained above, we prefer assessing our reference simulation by comparing its results with a MAR simulation over the whole GrIS since it is a widely used reference in SMB reconstruction, and we are mainly interested in the variables of the SMB. Nevertheless, we could have compared our simulation to other data sources. Hence we can compare our results to ERA-5 data. Briefly, ERA-5 are reanalysis, so they "*combines model data with observations from across the world into a globally complete and consistent dataset using the laws of physics*" (ECMWF, 2024). Let us note that ERA-5 data serves as forcing for out MAR reference simulation, so it is not a completely independent data source. As MAR is known for "downscaling simulations coming from GCM or reanalysis", ERA-5 data is provided at a lower resolution than MAR outputs. Also, since they do not share a common grid, we had to interpolate the MAR results onto the ERA-5 grid. As the computer used for the previous interpolation was not available, it was the opportunity to program a new algorithm in Python, using several threads. The code is available in Appendix F.

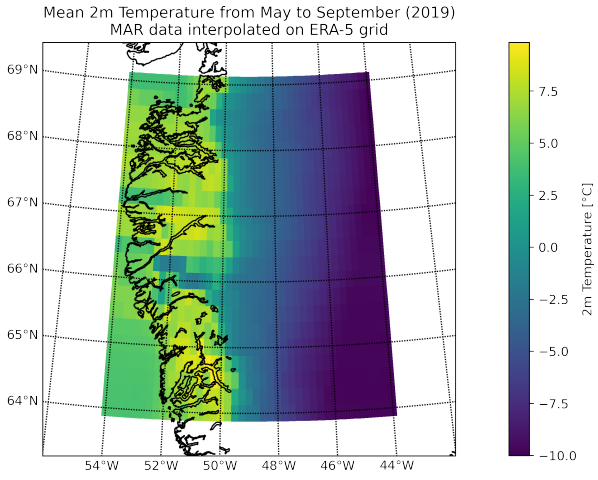
Figure 4.14 shows that the temperature from our MAR simulation are consistent with ERA-5 data, on average. As the resolution of ERA-5 is significantly lower, relatively large differences are observed at the ice sheet's edges, where there is an important temperature gradient. Nevertheless, these results show we can continue our study by being satisfied with our reference simulation. However, let us remark that Figure 4.14 (c) shows that MAR seems to overestimate the temperatures, which could lead to an underestimation of the SMB.

Conclusion on our Simulation Assessment

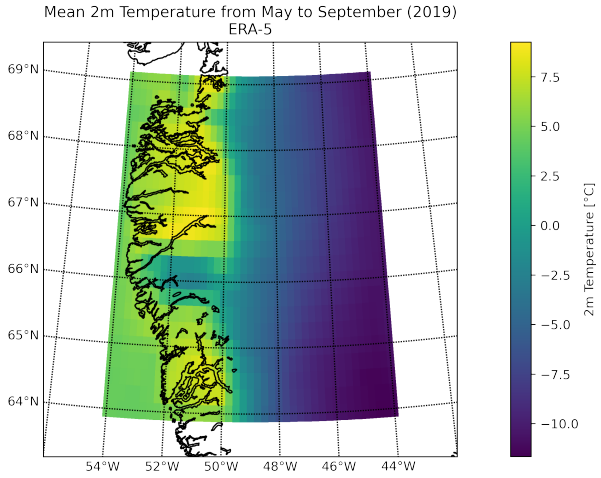
In conclusion, we can affirm that our MAR simulation conducted at a 5km×5km resolution over the southwest of the GrIS provides satisfactory results, allowing us to continue our study. However, it is worth noting that our simulation slightly overestimates snowfall and does not represent rainfall as accurately as the other variables. Note, finally, that the differences compared to the simulation at a 10km×10km resolution are minor, and this latter simulation is also likely to be a little biased, or at least, it is not a "panacea".

4.2.2 Sensibility Tests in MAR & Influence of the Refreezing on the Surface Mass Balance (SMB)

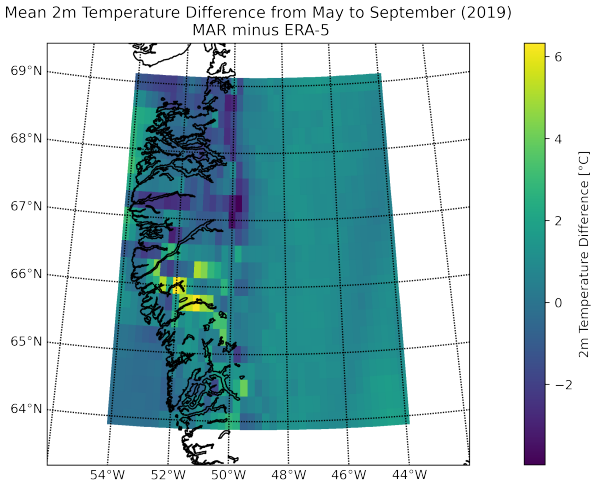
Now that we have run our first MAR simulation and verified that its results are acceptable, we can turn our attention to the results obtained by modifying certain parameters and mechanisms of the MAR. We will evaluate these modifications against the validated simulation we discussed above.



(a) Interpolated MAR data.



(b) ERA-5 data. Source: ECMWF (2024)



(c) Difference between (a) and (b).

Figure 4.14: Comparison of the Mean 2 metres temperature from our MAR simulation to ERA-5 data.

Table 4.1: Parameters of t^* of the different test simulations.

simulation	c_1 [hours]	c_2 [hours]	c_3 [unitless]
01	3	15	140
02	360	360	140
03	720	720	140
04	24	48	140
05	120	120	140
06	240	240	140
07	720	1440	140

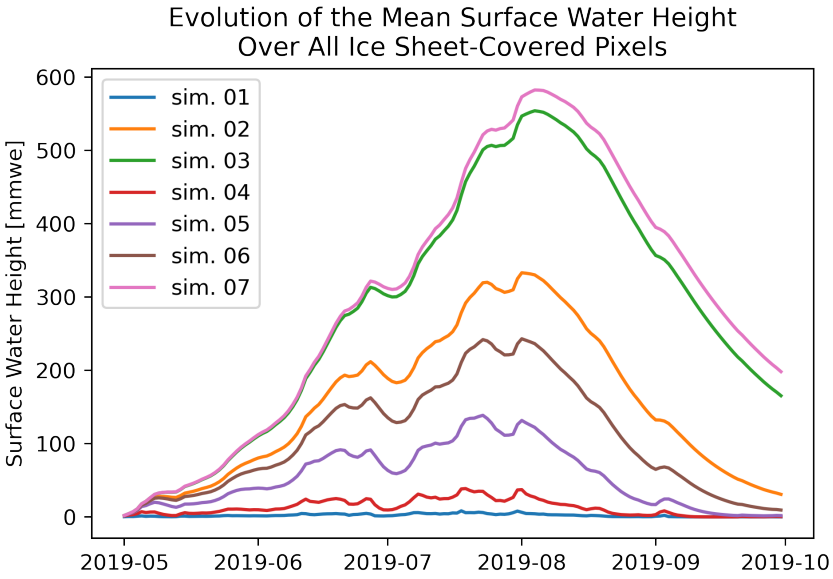


Figure 4.15: Summer evolution of the mean (over all pixels) surface water height for the different simulations. Pixels out of the ice sheet were ignored in the mean. We refer to Table 4.1 for the different simulations' c_1 and c_2 parameters.

Modification of the "Turnover" Time (t^*)

As explained in Section 4.1.1, the runoff rate follows a decreasing exponential law with a characteristic time (*i.e.* necessary to decrease by a factor e the initial surface water height, $W_r(0)$ in the absence of other surface water flux), t^*, that depends on three constants (*cf.* Equation (4.4)). To assess the influence of this parameter on the whole system, several MAR simulations were launched with different values of c_1 and c_2. The values of c_1 and c_2 corresponding to each simulation are listed in Table 4.1.

Obviously, modifying the value of t^* has a huge impact on the surficial water height, as the other fluxes do not straightforwardly depend on it. The original values are: $c_1 = 10800$s (3 hours), $c_2 = 54000$s (15 hours), and $c_3 = 140$. Figure 4.15 shows the summer evolution of the mean (over ice sheet pixel in the domain) surficial water height (namely "SWSN" in MAR outputs) for each simulation listed in Table 4.1. We observe that c_1 has a higher impact than c_2 on the mean surficial water height. For example, the difference between simulations 07 and 03 is not that high even if c_2 of simulation 07 is two times higher than c_2 of simulation 03.

Spatial variability Let us stress that even when we modify t^*, the surficial water height varies greatly over the ice sheet (in the domain). This is illustrated in Figure 4.16, where each line corresponds to an ice sheet-pixel. Similarly Figures 4.17, 4.18, and 4.19 illustrate at three different dates the spatial variability of the surficial water height, for simulations 01, 02, and 07 respectively. It appears clearly that the surficial water height is higher at relatively low altitudes (*cf.* Figure 4.5). Of course, this can be linked to the concept of adiabatic lapse rate we discussed earlier (in Section 1.1) since it states that the lower the altitude, the higher the temperature.

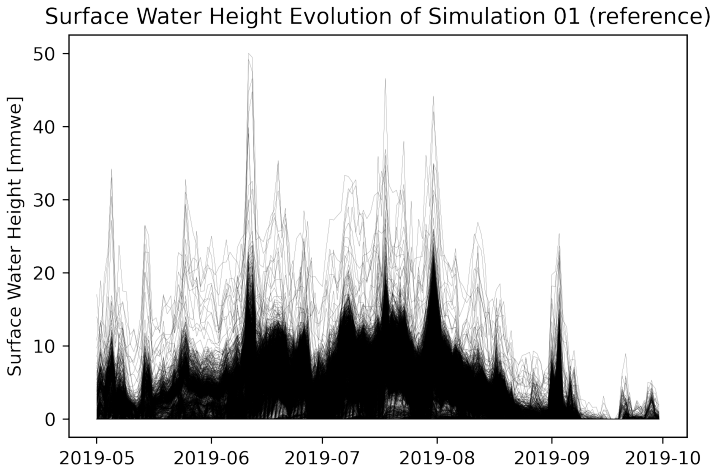


Figure 4.16: Evolution of surficial water height during summer for simulation 01 (*cf.* Table 4.1). Each line corresponds to the surficial water height evolution of a pixel. Pixel out of the ice sheet are not represented.

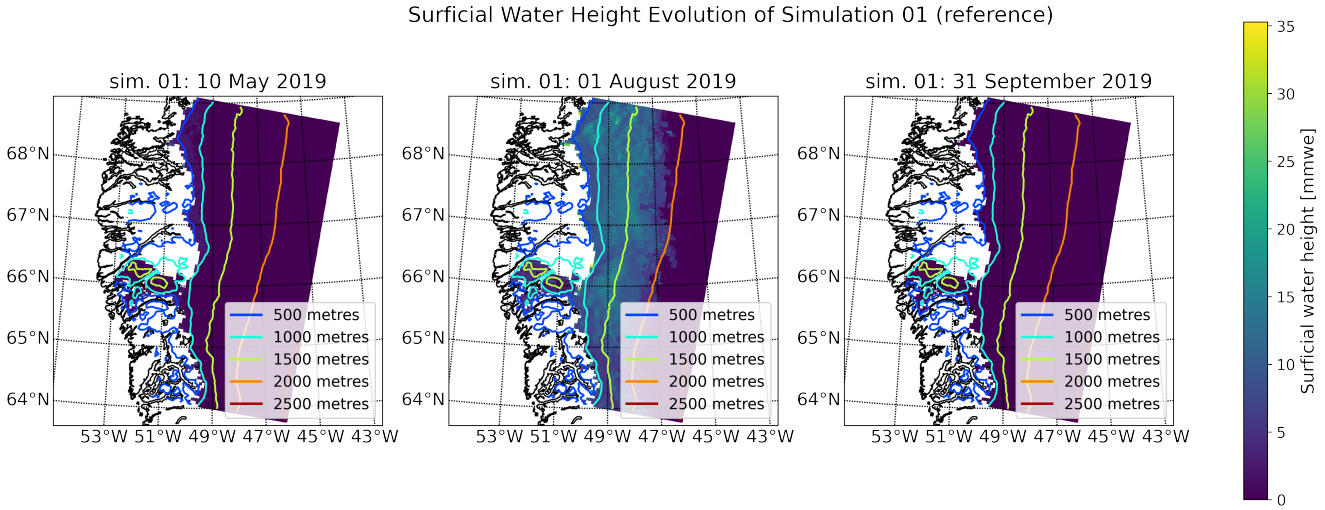


Figure 4.17: Surficial water height in the domain for simulation 01 (reference) at three different dates. We also present the topographic contours.

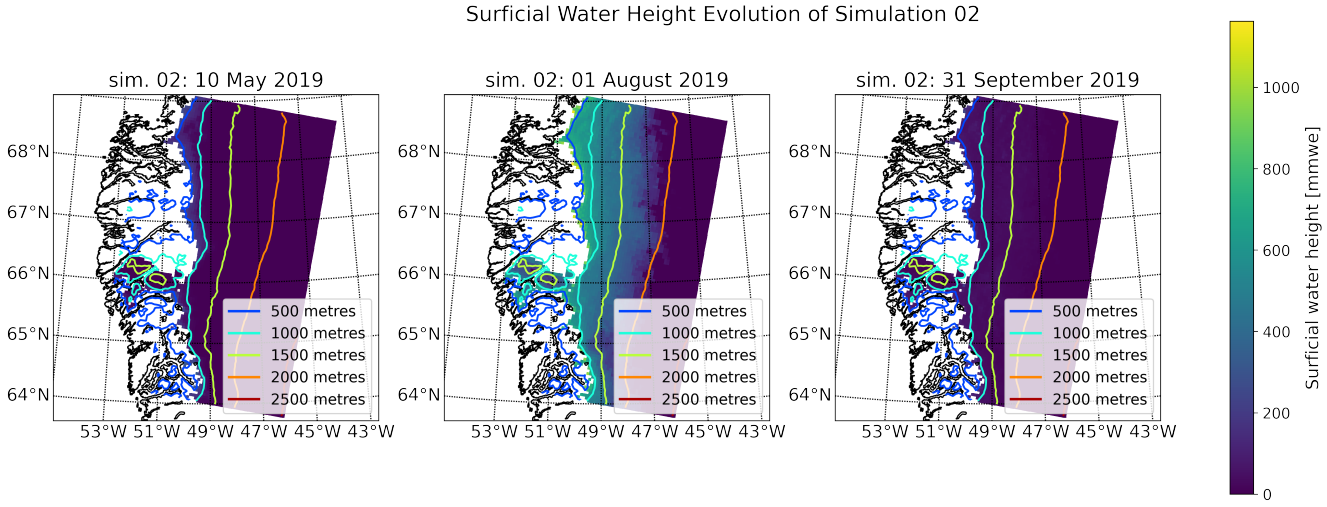


Figure 4.18: Surficial water height in the domain for simulation 02 (*cf.* Table 4.1) at three different dates. We also present the topographic contours.

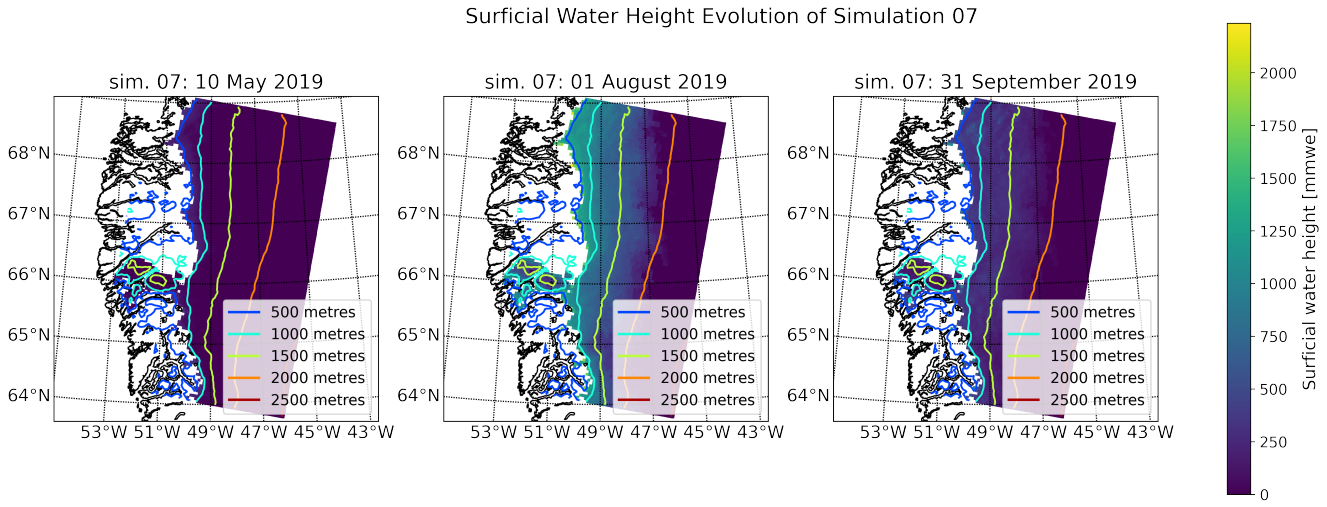


Figure 4.19: Surficial water height in the domain for simulation 07 (*cf.* Table 4.1) at three different dates. We also present the topographic contours.

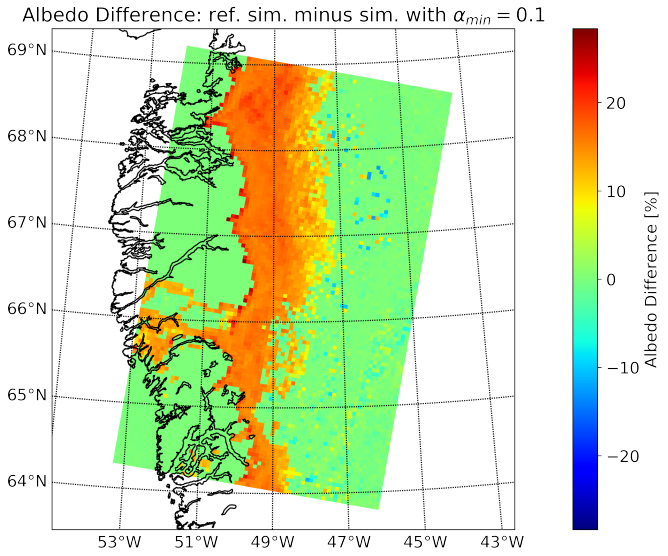


Figure 4.20: Illustration of the Influence of α_{min} on albedo: Difference in mean albedo between the simulation with $\alpha_{min} = 0.1$ and the reference simulation on August 1, 2019.

Let us now discuss how the surficial water height slightly influences the albedo in SISVAT.

Influence on the albedo Note that the MAR (or more specifically SISVAT) accounts for the influence of surface water height on albedo. Indeed, in the case where the first layer (in the discretized representation of the snowpack by SISVAT) is pure ice (which generally means snow that has melted and gradually converted into ice during the summer), the surface albedo is incremented by a term proportional to a decreasing exponential function of the surface water height. Its expression takes the following form (Eq (4.7)):

$$\alpha = \alpha_{min} - (\alpha_{min} - \alpha_{max})e^{-\sqrt{\mathrm{SWH}/C}} \tag{4.7}$$

Where α is the surface albedo, α_{min} is the minimal bare ice albedo, α_{max} is the maximal bare ice albedo, SWH is the surface water height, and C is a positive constant expressed in the units of SWH (set to 50 mmwe in MARv3.14).

The Eq. (4.7) is such that the albedo has a maximum value of $\alpha_{\max}$ when the surface water height is zero and tends towards a minimum value of $\alpha_{\min}$ as the surface water height increases.

Since this expression of albedo only applies when the surface is covered with bare ice (alternatively, the model assumes that the water infiltrates into the snowpack, and thus it is expected that this water does not influence the surface albedo, so this latter depends on the snow properties) and since the difference between α_{min} and α_{max} is small (0.05), plus the fact that differences in surface water height have a diminishing impact as they increase, overall, this has a limited effect on melting, and the SMB.

Actually, the values of α_{min} and α_{max} in MARv3.14 are 0.5 and 0.55, respectively. For the sake of illustration, we conducted a simulation starting from August 1st with $\alpha_{min} = 0.1$. The difference in albedo between this simulation and the reference simulation on August 1, 2019, is illustrated in Figure 4.20. We observe that, indeed, this modification only impacts the ablation zone where the surface water height is non-zero.

After setting the parameter $\alpha_{\min}$ to 0.1, we reran the simulation, this time imposing that meltwater leaves the ice sheet directly (which corresponds to $t^* \to 0$), to illustrate the influence of water height on albedo via the relation of Eq. (4.7). The results are presented in Figure 4.21. Note that if the difference between $\alpha_{\max}$ and $\alpha_{\min}$ is not exaggerated as we have done, this effect is not observed.

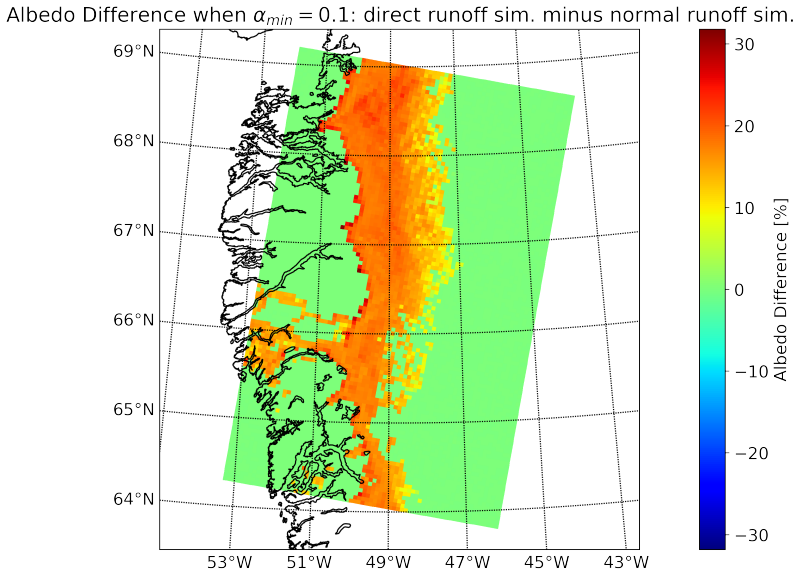


Figure 4.21: Illustration of the influence of the surficial water height on albedo when setting $\alpha_{\min}$ to 0.1: Difference in mean albedo between the simulation with direct runoff (so no surficial water) and the "normal runoff simulation" (i.e., c_1=3h & c_2=15h) on August 1, 2019.

Table 4.2: Parameters of t^* and ice albedo of the different test "simulations".

simulation	c_1 [hours]	c_2 [hours]	c_3 [unitless]	ice/snow albedo [unitless]
09	3	15	140	0.55
10	720	1440	140	0.55
11	3	15	140	0.745
12	720	1440	140	0.745

Constant Albedo

Taking advantage of our discussion on albedo, we present the results of another sensitivity test we conducted. We ran the MAR four times, imposing specific albedo values regardless of the surface water height or snow-pack properties. It is akin to adding dye to the ice/snow. These "simulations" naturally provide extreme results, their sole purpose being to explicitly illustrate how albedo can influence melt. The albedo values are listed in Table 4.2.

Figure 4.22 shows the evolution of surficial water height for these illustrative MAR runs (denoted hereafter simulations 09, 10, 11, and 12 (*cf.* Table 4.2)). We observe that, as already mentioned, c_1 and c_2 strongly influence the surface water height, but also that imposing a constant albedo for ice and snow has an impact on surface water height. Indeed, there is up to twice as much surficial water in simulation 10 than in simulation 12 because the albedo is lower. This is because our modifications significantly impact the energy budget of the top layer, which can greatly affect the amount of melting water. Of course, these results are solely intended to illustrate this effect and do not represent any physical reality.

4.2.3 Influence of t^* on the Surface Mass Balance (SMB) & Modification of the Surface Water Refreezing Condition

Now that we have conducted our first sensitivity tests with the MAR and concluded that the surface water height has little influence on the albedo, it is evident to verify whether the surface water height or rather the rate at which it flows, influences the SMB. This influence would occur not through albedo, but rather by enhancing refreezing.

Since the runoff is among the most important terms of the SMB (*cf.* Figure 1.4, the SMB is defined in Eq. (1.3)), modifying its operation in the MAR should influence the simulated SMB. Indeed, we can expect that changing the

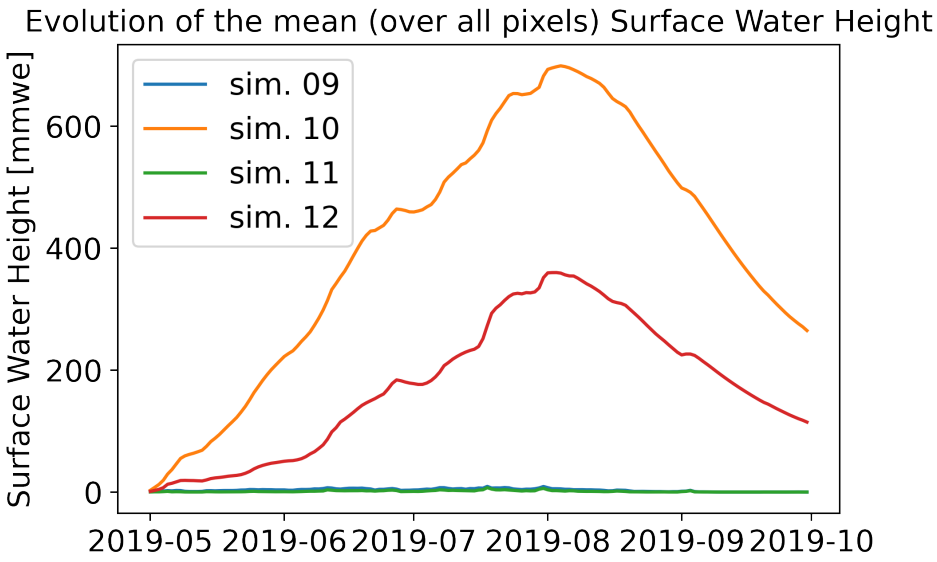


Figure 4.22: Summer evolution of the mean (over all pixels) surface water height for the different simulations. Pixels out of the ice sheet were ignored in the mean. We refer to Table 4.2 for the different simulations' c_1 and c_2 parameters and constant snow/ice albedo.

"turnover time", t^*, will lead to a situation where water remains on the surface of the ice sheet at the beginning of winter, allowing it to refreeze and thus contribute to the increase of the SMB during the period of interest.

Nevertheless, we did not observe this phenomenon by comparing the cumulative SMB of the simulations presented above. So, we extended these simulations by a few months to allow the surface water to freeze or drain, but we still did not obtain the expected result. We then ran a simulation where runoff was nonexistent ($t^* \to \infty$), and we noticed that the amount of surface water never actually decreased. This led us to realise something was wrong, and indeed, upon reviewing the code, we detected an error.

In fact, SISVAT had a mechanism to allow the surface water to refreeze, but it was never activated due to an error in specifying the condition necessary for its launch. We detail this error and the reasoning that led us to identify it in Appendix H.

Before pursuing, it is worth noting that the error we identified does not completely invalidate the results of all previous MAR simulations before its discovery (this will be confirmed later). Indeed, surface water refreezing does occur and should be taken into account in SMB modelling, but the majority of meltwater still either runs off over the ice or infiltrates into the snowpack.

A process to correct the error was thus undertaken. During this process, an important observation emerged: refreezing should only occur when surface water is on bare ice[3]. Indeed, if the snowpack is not composed of pure ice, the water infiltrates into it. This had to be taken into account in the correction. This is particularly important because if not considered, it allows water on top of snow, which would normally infiltrate, to refreeze. This results in the creation of ice on the surface of the snowpack, significantly reducing the albedo, which can substantially increase melt rates, sometimes leading to cases where the addition of refreezing increases the amount of surficial water. That is why we decided to consider that surface water was directly removed from the model if it was not located above bare ice.

[3]The bare ice is, according to Fettweis (2023), the area where the surface density is higher than 900kg/m^3. It is thus where ice is observable at the surface, generally due to snow melting. Its albedo is generally lower than snow, and we mainly observe it in summer in the ablation zone.

During the correction process, it was observed that in cases where snow fell on top of a pixel covered with bare ice already overlaid with water, a new layer of snow was created on top of the ice mantle, thereby increasing the albedo. This constitutes an error in the same way as the previous one. However, as with the first error, it is not particularly alarming since this situation is relatively rare. This is because, first, the water generally drains, and second, in reality, if liquid water is present on the surface, temperatures are generally positive, and precipitation is mostly rain. Nonetheless, it was necessary to correct this because, in reality, it does occasionally snow over a body of water. Supra-glacial lakes are typical locations where this might occur.

Therefore, if the aforementioned case occurred, we ensured that the snow was converted into rainwater (when reaching the ground) at the beginning of the code executed at each time step, taking care to account for the latent heat associated with the melting of this snow, assuming its temperature to be zero.

Impact of our Corrections on the Modelled Surface Mass Balance (SMB) & on Surface Properties

We then relaunched our simulations with the corrections described above, which can be summarized as follows:

- The surface water above bare ice (so the water that does not infiltrate in the snow mantle) can refreeze.
- The surficial water above snow infiltrates and thus leaves the model (that, as a reminder only models the first 20m of snow/ice) through direct runoff.
- The snow that falls into water melts, and the latent heat is taken into account in the energy budget.

Let us now discuss their influence on the modelled SMB. Firstly, if we change nothing in the code except for the modifications mentioned above, the impact on the SMB is negligible. Indeed, Figure 4.23 shows the relative differences between the two simulations. One might think, from looking at the map, that the differences are significant in the west of the domain, but this is solely because the cumulative SMB in this area is close to zero (near the "equilibrium line"), as mentioned when we compared our reference simulation to a simulation covering the entire GrIS (at a resolution of 10km×10km), so the denominator is close to zero (*cf.* Eq. (4.6)).

Next, if we compare simulations where runoff is absent (i.e., $t^* \to \infty$), the differences of SMB are very pronounced outside the bare ice zone, as shown in Figure 4.24. This is expected because our corrections impose surface water runoff as soon as it is not above bare ice (we consider that it infiltrates into the snowpack and leaves - through direct runoff - the SISVAT model domain, which only covers the top 20 meters of snow, for reference). In the bare ice zone, we do not observe differences, since as a reminder, we consider water to be lost as soon as it has flowed away. Therefore, whether the water refreezes or remains on top of the ice does not affect the SMB according to the definition we have adopted here.

This (i.e. Figure 4.24) confirms that our modifications lead to consistent results. To further ensure this, we can also show Figure 4.25, which demonstrates that the Surface Water Height (SWH) begins to decrease at the end of summer in our corrected runoff simulation, whereas it previously (*i.e.*, before our intervention) remained on a "plateau". Let us note that, as with simulations where we force the albedo (see the illustration of dye on snow), these simulations poorly represent physical reality. However, they are quite interesting in the sense that they allow us to verify that our code is functioning correctly. We observe that refreezing occurs when the temperature become negative, which was expected.

Regarding the correction that ensures snow melts when it falls on surface water, we can confirm that the issue is properly resolved by examining Figure 4.26. It shows that in our uncorrected "reference" simulation, a snow precipitation event over a pixel containing bare ice leads SISVAT to create a new snow mantle. This significantly decreases the density at the top of the modelled snow/ice layer, as fresh snow is much less dense than ice. This also affects the surface albedo, since, as mentioned earlier, snow has a higher albedo than ice.

However, we have just seen that the impact of our corrections on the SMB over a summer period (or rather from May to September) is roughly negligible. Indeed, the scenario described, where our modification causes the surface properties (and the albedo in particular) to differ between the two simulations, occurs several times during the simulated period but is generally quite short. Thus, it does not lead to significant differences in the accumulated SMB.

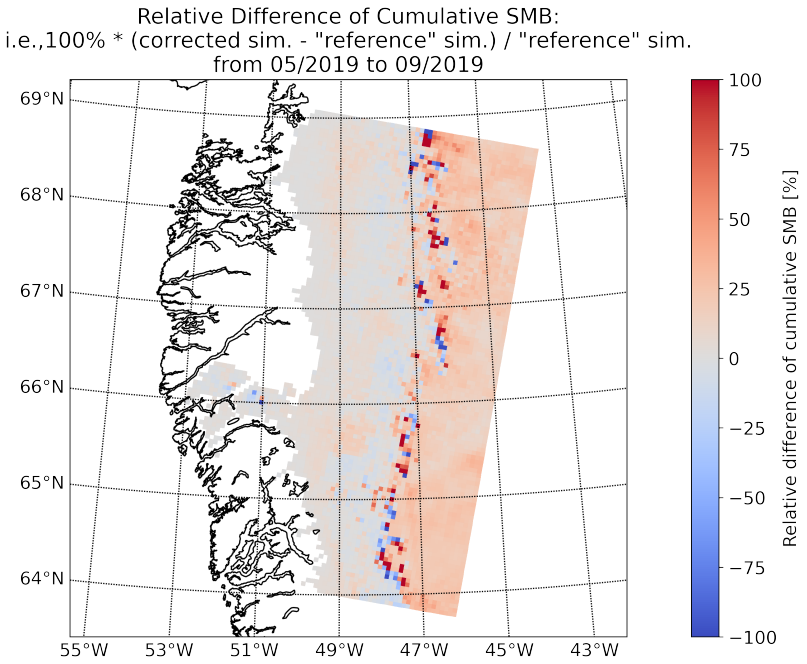


Figure 4.23: Relative difference of cumulative SMB between our reference MAR simulation and its version corrected for the errors in the code we mention in the text.

Moreover, this situation (where a new snow layer is created, decreasing the albedo) typically occurs at the end of summer (early winter). Indeed, precipitation is mostly rain in the bare ice zone during summer. At this time (in winter), solar radiation is less significant due to the high latitudes, and thus, albedo differences ultimately have a lesser impact on the energy balance (certainly less than in summer), therefore this does even less impact the SMB.

Also, it is evident from Figure 4.26 that the SWH remains zero for our corrected simulation as long as there is no bare ice at the surface.

Influence of t^* on the Surface Mass Balance (SMB)

After this "interlude" explaining the corrections we brought to the code, we can finally conclude our discussion about our sensitivity tests in MAR and their influence on SMB. Without any surprise, we observe in Figure 4.27 that if we increase c_1 & c_2 (*cf.* Eq. (4.4)), then the SMB increases, because we allow the surficial water hat melts in summer) to refreeze in winter. Indeed, for low values of c_1 and c_2, if the ice melts, it flows away quickly and does not have time to refreeze. However, if very large values are imposed for these parameters, the SWH remains very high at the end of summer, and the water ultimately freezes instead of flowing away.

This underscores the significant impact that parameters c_1 and c_2 have on the modelled SMB. The longer it takes for the water to flow, the greater the refreezing. It is also important to remember that, ideally, water should flow from one pixel to another, which means that the time between melting and reaching the tundra or ocean can be considerable.

The Figure 4.28 displays the same data as Figure 4.27, but in terms of relative differences. It appears that the increase in SMB when setting $c_1 = c_2 = 720$h can reach up to 50% in the ablation zone.

We launched another simulation where we multiplied by two the value of c_1 (*cf.* Eq. (4.4)), setting it to 6 hours (we did not change the value of c_2). The results are shown in Figure 4.29. We observe that the SMB increases for most pixels, but it is less clear than when setting $c_1 = c_2 = 720$h because we observe (in Figure 4.29) that for some pixels, the difference is negative (*i.e.*, the SMB decreases a bit). Actually, the SMB differences are much less significant, as shown in Figure 4.29 (b).

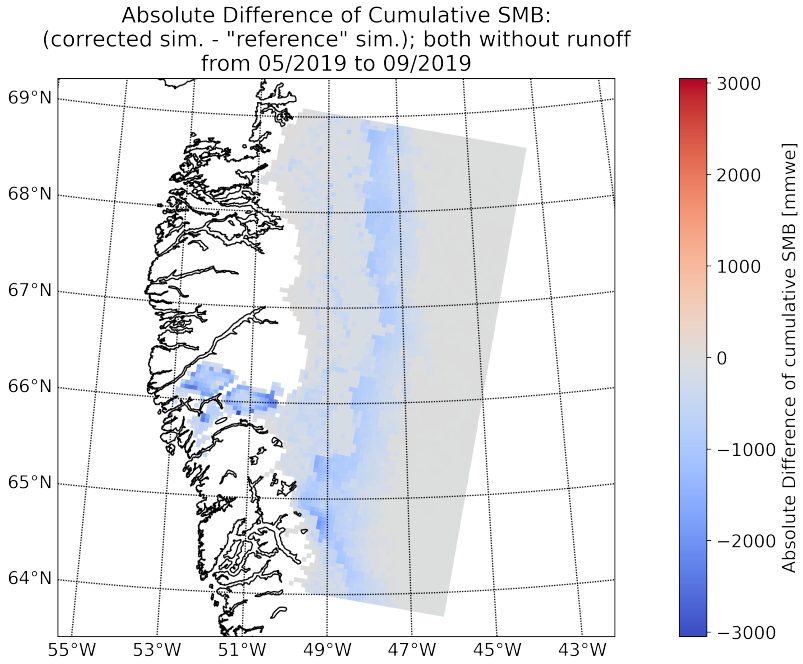


Figure 4.24: Absolute difference of cumulative SMB between our reference MAR simulation without runoff and its version corrected for the errors in the code we mention in the text, also without runoff.

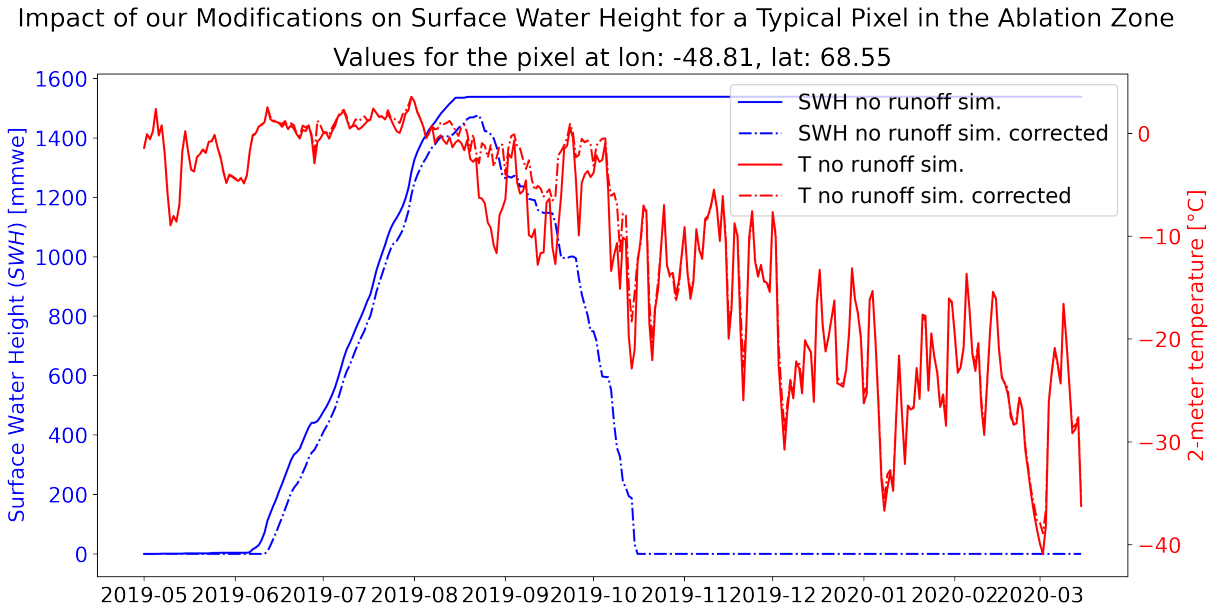


Figure 4.25: Effect of our code corrections on the "experimental" simulations where runoff is assumed to be non-existent, for a representative pixel in the ablation zone. We also present the temperatures of the two simulations to show that refreezing occurs when temperatures drop below zero. Note that the legend slightly obscures the end of the SWH evolution curve for the uncorrected no-runoff simulation. As mentioned repeatedly in the text, this curve does not change in winter (when it should decrease due to refreezing). Since it is clear that it remains stagnant at around 1600mmwe, this was the best place to position the legend without obscuring other, more important information.

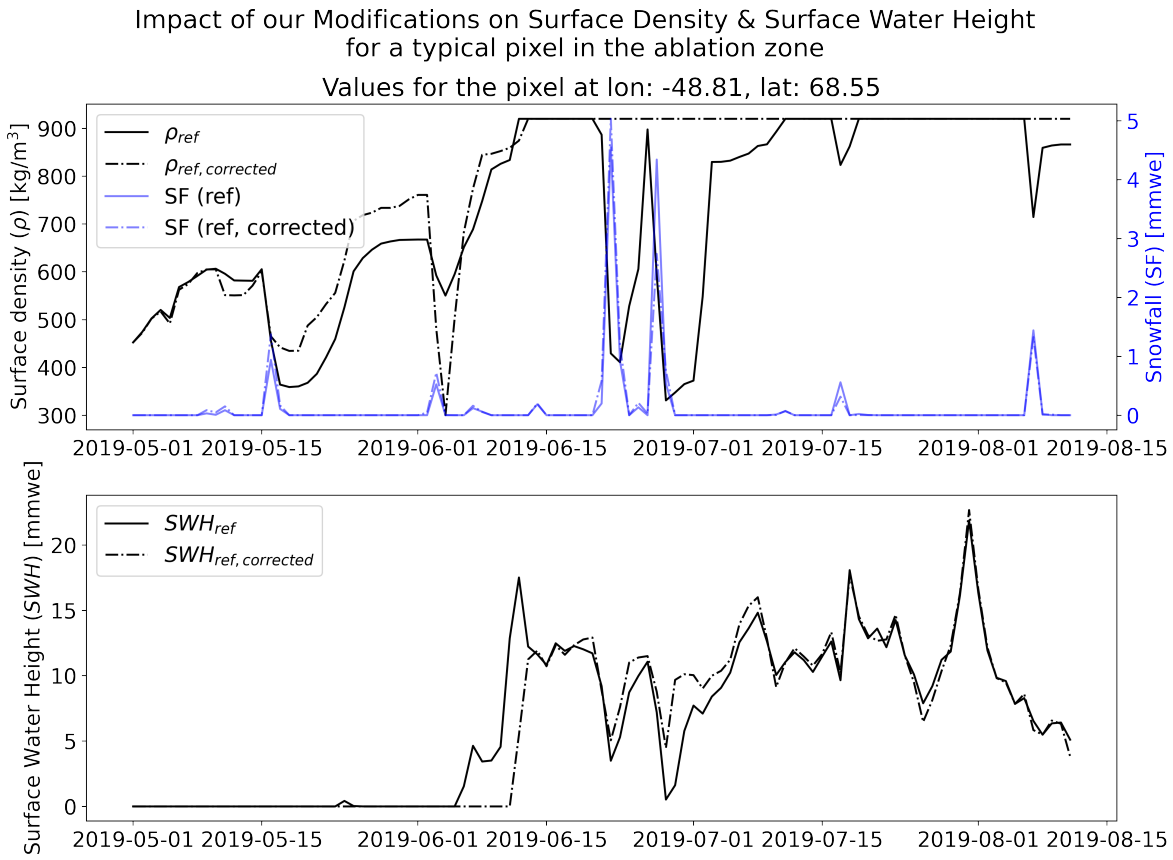


Figure 4.26: Impact of our modifications on the surface density (ρ) for a typical pixel in the ablation zone, as well as on its SWH. We compare our reference simulation (i.e., without any modifications to the MARv3.14 code) and a simulation where we only apply the corrections mentioned in the text. The subscript $_{ref}$ means that we keep the parameters c_1 and c_2 (*cf.* Eq. (4.4)) from the reference simulation (so 3h and 15h respectively), while $_{corrected}$ means that we do not modify anything in the code except the corrections mentioned in the text.

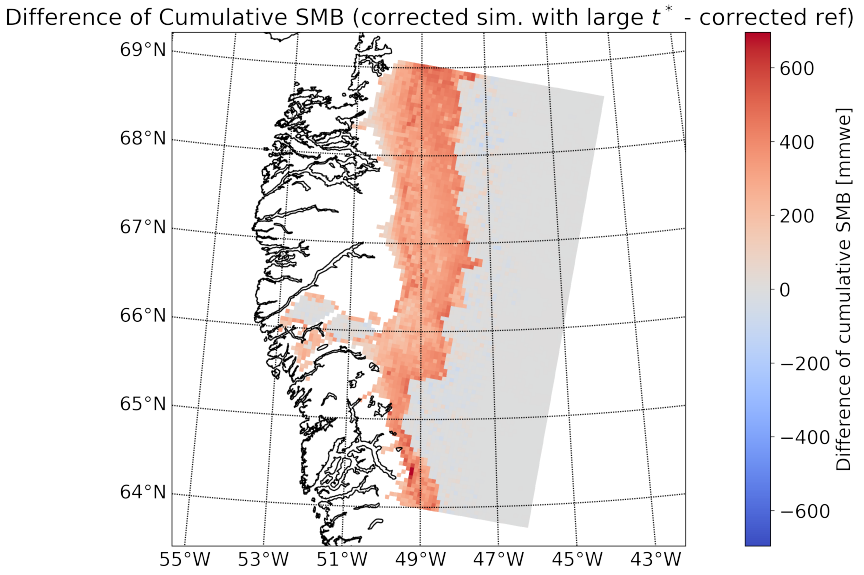


Figure 4.27: Cumulative SMB difference (from May 1, 2019, to December 15, 2019) between the corrected reference simulation (where c_1 and c_2 are the default values in MARv3.14, i.e., 3 and 15 hours, respectively) and the simulation where c_1 and c_2 are set to 720 hours (for both), and where the corrections mentioned in the text are also applied. Note that for the simulation with $c_1 = c_2 = 720$ hours, the SWH is restricted to 1 meter by enforcing direct runoff of the surplus, which only occurs for the "warmest" pixels located at the ice sheet margin (we also do it for the other simulation, but it never occurs).

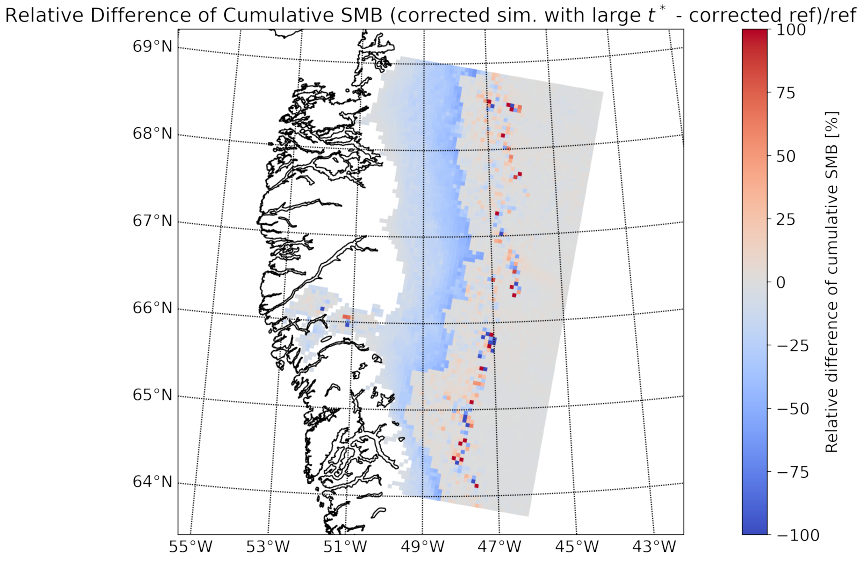
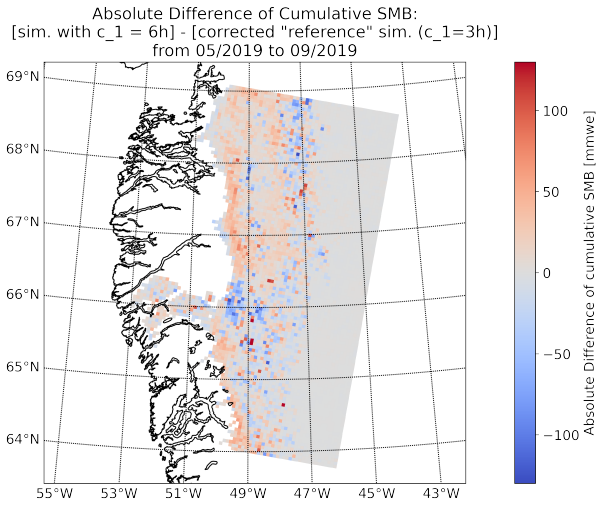
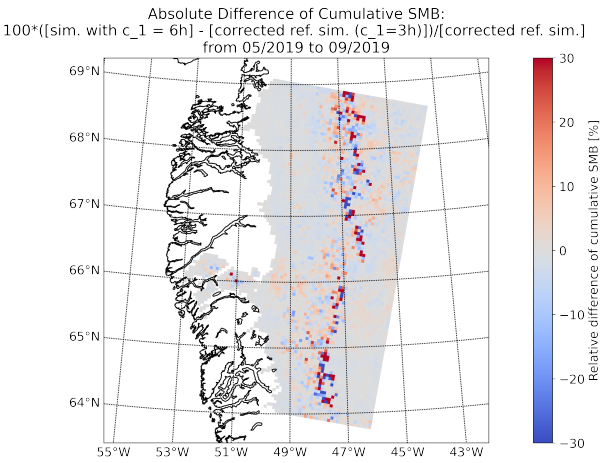


Figure 4.28: Cumulative SMB relative difference (from May 1, 2019, to December 15, 2019) between the corrected reference simulation (where c_1 and c_2 are the default values in MARv3.14, i.e., 3 and 15 hours, respectively) and the simulation where c_1 and c_2 are set to 720 hours (for both), and where the corrections mentioned in the text are also applied. Note that for the simulation with $c_1 = c_2 = 720$ hours, the SWH is restricted to 1 meter by enforcing direct runoff of the surplus, which only occurs for the "warmest" pixels located at the ice sheet margin (we also do it for the other simulation, but it never occurs). The apparent sign difference with respect to Figure 4.27 is because the SMB is negative in the ablation zone (*cf.* Eq. (4.6)).



(a) Cumulative SMB difference (from May 1, 2019, to December 15, 2019) between the corrected reference simulation (where c_1 and c_2 are the default values in MARv3.14, i.e., 3 and 15 hours, respectively) and the simulation where c_1 is set to 6 hours (c_2 is not modified), and where the corrections mentioned in the text are also applied.



(b) Relative difference of cumulative SMB (from May 1, 2019, to December 15, 2019) between the corrected reference simulation (where c_1 and c_2 are the default values in MARv3.14, i.e., 3 and 15 hours, respectively) and the simulation where c_1 is set to 6 hours (c_2 is not modified), and where the corrections mentioned in the text are also applied.

Figure 4.29: Impact of setting $c_1 = 6$h on the SMB.

4.2.4 Simulation of a Gigantic Lake

In line with the few experiments made with SISVAT, simulation 16 was launched with the following modification in the code: The runoff is fully redirected towards a fixed area of 266 pixels in the domain from where it runoffs towards the sea. Also, c_1 and c_2 were set to 72 hours (ten times less than for simulation 03). Figure 4.30 (a) shows this zone, and Figure 4.30 (b) represents the temporal evolution of surface water height in and out of this zone, as well as water fluxes.

Of course, this simulation does not represent any physical reality, just like the simulations where we deliberately force albedo values or the ones where we suppress water runoff. Their interest also lies in gaining experience with the model... However, in reality, water flows from one location to another. Therefore, adding the capability for water to flow from one pixel to another in the model would be pertinent. But, this should be done thoughtfully, considering that parallelisation of the code might pose a problem[4].

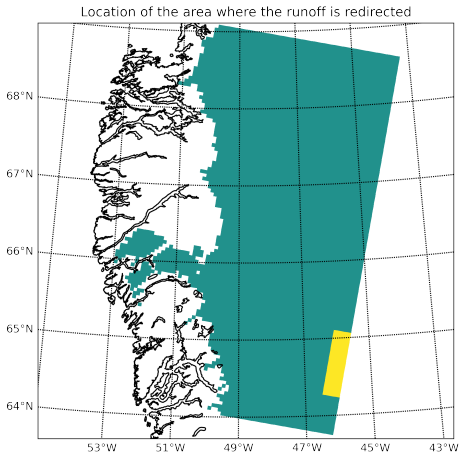
4.2.5 Consideration of Surface Water Evaporation

Now that we have thoroughly discussed water runoff in the MAR, its refreezing, and its connection with albedo, we can address the fact that the MAR neglects its evaporation.

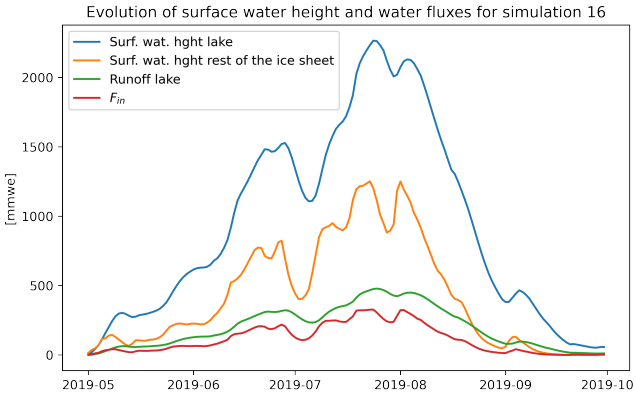
Motivation

One of the main tasks of this thesis is to add the possibility for surficial water to evaporate into MAR. This should enable us to evaluate the impact of evaporation on the SMB. Indeed, MAR neglects the surface water (over ice) evaporation and only considers its runoff and refreezing. Even if, paradoxically, it has a variable that corresponds to the amount (in mmwe) of water over the ice, which we have quite a bit discussed so far. Also, MAR considers the effect of the ice sublimation on the SMB, even when there is surficial water over the ice.

[4]The simulation discussed here, where we create a gigantic lake, was, of course, a bit slower than the other simulations presented in this report. Moreover, if we wanted to allow water to flow to the downstream pixel, we should account for the topography, etc. So it would be more complicated to implement than redirecting it all to specific pixels, as we have done.



(a) Location of the area where the runoff is redirected (in yellow).



(b) Evolution of surface water height and of water fluxes for simulation 16. F_{in} corresponds to the flux of water (per pixel) towards the fictional lake, so the runoff from pixel out of the yellow rectangle in Figure 4.30 **(a)** received by the pixels in this rectangle. F_{in} and the runoff are expressed in mmwe/day.

Figure 4.30: Graphical summary of simulation 16's hydrology.

So the initial MAR version (used so far in this thesis) considered that even with a meter of surficial water over it, an ice pixel loses mass due to sublimation. However, in such a configuration, the surficial water evaporates, and the ice does not sublimate. This also happens in reality, as there are SGL over the ice sheet. To estimate the importance of that on the SMB, one can include it in the SISVAT module.

Water Vapour Transfer Between the Surface and the Atmosphere in MAR

Appendix I explains how the part of the code that computes the latent heat flux associated with the evaporation. The flux is the product of a coefficient (that depends on the Surface Boundary Layer (SBL) air properties) times the difference between the SBL top specific humidity and the saturation specific humidity, multiplied by the latent heat of vaporisation of the uppermost snow/ice layer, that might contain a certain amount of liquid water, $L_{x,\mathrm{H_2O}}$. The amount of sublimated ice/snow is deduced from the calculated latent heat flux.

The latent heat of the surface[5] $L_{x,\mathrm{H_2O}}$ is computed as follows (Eq. (4.8)):

$$L_{x,\mathrm{H_2O}} = \begin{cases} 2.5 \times 10^6 \mathrm{J/kg} & \text{if not on the ice sheet} \\ (1-f)2.8345 \times 10^6 + (f)2.5 \times 10^6 \mathrm{J/kg} & \text{otherwise} \end{cases} \tag{4.8}$$

Where f is the water content of the uppermost snow or ice layer in SISVAT, in $\mathrm{m^3/m^3}$. The model considers (for the condition of Eq. (4.8)) that a pixel is on the ice sheet if the number of ice and snow layers, is not zero.

The saturation ice and water pressure, $p_{sat,ice}$ & $p_{sat,water}$, are computed as follows (Eq. (4.9)):

$$q_{sat} = \begin{cases} \frac{0.622\ p_{sat,ice}}{p - 0.378\ p_{sat,ice}} & \text{if } T_{surface} \leq 273.15\mathrm{K} \\ \frac{0.622\ p_{sat,water}}{p - 0.378\ p_{sat,water}} & \text{otherwise} \end{cases} \tag{4.9}$$

These parameters depend on the surface temperature through the relations of Eq. (4.10) & (4.11) (where $T_{surface}$ is the surface temperature and p is the surface pressure).

$$p_{sat,ice} = 6.1070\mathrm{hPa} \times \exp\left(6150\left(\frac{1}{273.16} - \frac{1}{T_{surface}}\right)\right) \tag{4.10}$$

[5]It works if the surface is covered by snow, ice or water (over the ocean for example). We do not present how the latent heat is computed over the tundra.

$$p_{sat,water} = 6.1078\text{hPa} \times \exp\left(6827\left(\frac{1}{273.16} - \frac{1}{T_{surface}}\right)\right) \times \exp\left(5.138\ln\left(\frac{273.16}{T_{surface}}\right)\right) \tag{4.11}$$

Actually, modifying the surface temperature has more impact on the Latent Heat Flux (LHF) than setting $L_{x,\mathrm{H_2O}} = 2.5 \times 10^6$J/kg (*i.e.* the latent heat of water at 0°C), as it makes the LHF always so that there is evaporation instead of condensation, whereas the second action only contributes to increasing the LHF, no matter its sign. Therefore, if we only modify $L_{x,\mathrm{H_2O}}$, there is almost no net effect on the SMB. However, it seems likely that the surficial water temperature is not zero. This could be the result of its lower albedo. Hereafter, we discuss the results of simulations where we changed the value of $L_{x,\mathrm{H_2O}}$ (we set its value to that of water for the pixels where we want to replace sublimation with evaporation), but also the surface temperature.

Of course it would be necessary to compute the surface water temperature, for future developments.

Changes Made in the Code and Firsts Results

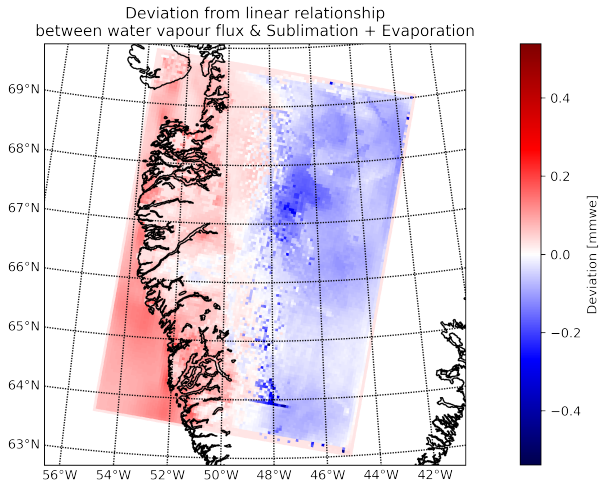
First, we created a variable corresponding to the fraction of each pixel area covered by water, re-using the data that is presented in Figure 3.7. Then we created a double for several variables linked to the evaporation/sublimation fluxes. Actually, we computed the LHF flux as the weighted (by our new lake fraction variable) average of the flux it would have been if the pixel was fully water-covered and the one it would have been if it was only covered by ice. Also, we prevented the runoff, to facilitate the interpretation of our results, because the quantities of water that evaporates in our simulations are very small compared to those that runoff. Removing runoff allows us to observe differences in water height between simulations that are only due to the three remaining surficial water fluxes: melting, refreezing and evaporation.

So we computed two LHF fluxes for each pixel, corresponding respectively to pure ice and water. Then we had to decrease the surficial water height, in agreement with the LHF flux corresponding to pure water, and multiply the amount of evaporated water by the fraction of water over the pixel. We do a similar operation for the sublimation (the flux is multiplied by one minus the fraction of lakes in the pixel). For consistency, adding evaporated/sublimated water to the atmosphere was necessary. We present below some results that were obtained by imposing that the temperature of the surficial water was 4°C, and by taking the minimum between 1 and 80 times the real satellite-based lakes fraction, as the fraction of the pixel covered by surficial water. This is obviously not realistic, even though the albedo could justify that the SGL temperature is not 0°C, the values mentioned here are way too large. We opted for these to observe better the modification's impact in the MAR. Note that we set the lake fraction to zero for the pixels of the buffer zone.

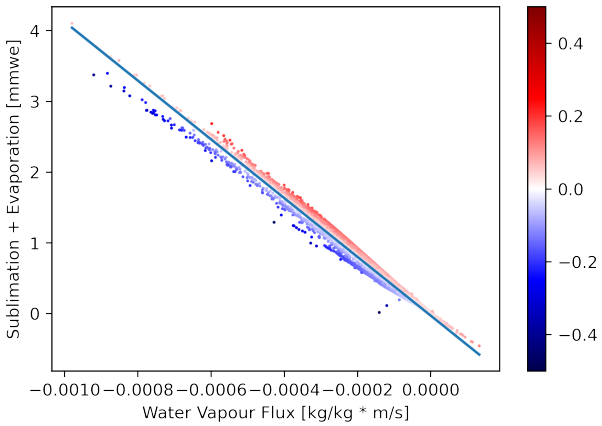
Figure 4.31 presents consistent data that somehow validates that the transfer of evaporated water to the atmosphere occurs correctly. The deviations from the linear relationship are quite small, and due in particular to the anisotropy of the air density or relative humidity, as well as to the quantities of evaporated water. Figure 4.32 shows the impact of our modifications on the sublimation and evaporation flux during one summer. As mentioned above, the fluxes are overestimated because of the assumed lake temperature of 4°C.

Increasing by 4°C the surficial water temperature triggered feedback in the model, leading to completely absurd results. Indeed, the air above the ice sheet is always water vapour-saturated during the "simulation", as shown in Figure 4.33 (it compares the specific humidity of simulations if we increase by 4°C the SW temperature, and the reference). It directly condensed, and converted into surficial water again, such that there was still a significant amount of surficial water at the end of summer. Rather than interpreting these results in detail, it seems more important to emphasise that increasing the surficial water temperature in the model as we did here (which is only done for evaporation but not for the refreezing) does not enable to model the surface of the GrIS correctly.

Next, we reran several simulations in which we artificially increased the surface water temperature, assuming that surficial water spanned over the whole pixel area. We present in Figure 4.35 the evolution of the amount of evaporated water from July 1 to August 15 for a pixel in the ablation zone. It appears that if we do not impose excessively high temperatures, the evaporation flux does not change significantly compared to the sublimation flux.



(a) Map of the deviation from the linear relationship between the water vapour flux between the atmosphere and the surface (negative downwards) and the amount of water sublimated or evaporated after our modifications. We compute the fluxes as averages over 24 hours of simulation on 07/01/2019.



(b) Representation of each pixel of the left map in a scatter plot, and comparison to the linear relationship. Each point corresponds to a pixel, the colormap is the same as in (a).

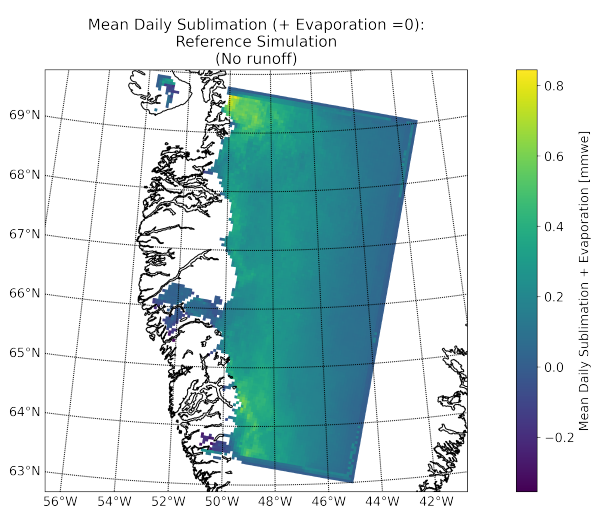
Figure 4.31: Assessment of the water vapour transfer from SISVAT to the atmospheric part of the MAR after our modifications.

It should be noted that it is not entirely unreasonable to assume that the temperature of the water above the ice could be approximately 4 degrees higher than that of the ice. However, it seems unlikely that this temperature difference would remain constant (as we assumed in our simulations) over time. This, along with the fact that we significantly exaggerated the area covered by lakes, is why the results presented in Figure 4.33 are extreme and far removed from reality (indeed, it is as if there were a constant layer of mist above the ablation zone).

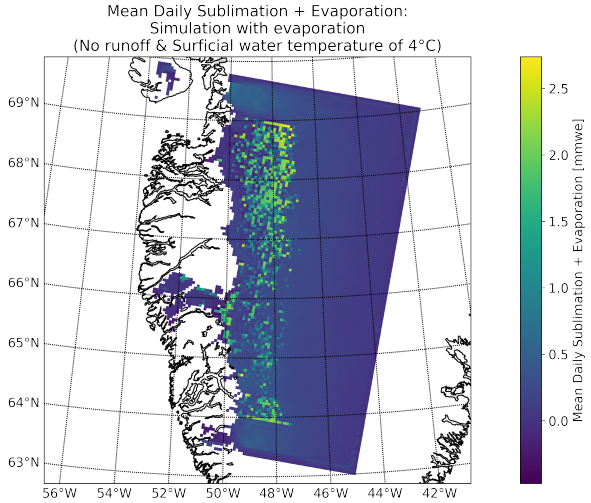
We present in Figure 4.34 the difference (absolute in (a) and relative in (b)) in the sum of cumulative sublimation and evaporation between July 1st and August 15th, 2019, comparing the simulation where evaporation replaces sublimation whenever the SWH is positive, assuming that the water temperature is 1°C higher than that of the surface ice, and the reference simulation (where only sublimation occurs). It appears that this does not consistently lead to an increase in flux, and overall, there does not seem to be a clear net effect. In fact, at the ice sheet margins, where sublimation was negative—meaning solid condensation dominates sublimation—our modification somewhat contributes to an increase in SMB, as we have reduced the value of $L_{x,\mathrm{H_2O}}$ (*cf.* Eq. (4.8)).

Let us also remind that, in addition to the assumption about water temperature, only a small portion of the pixel is actually affected by evaporation in reality, while the rest experiences only sublimation since lakes do not cover 100% of the area. We present results for only part of the summer because it should be noted that these lakes are not present year-round.

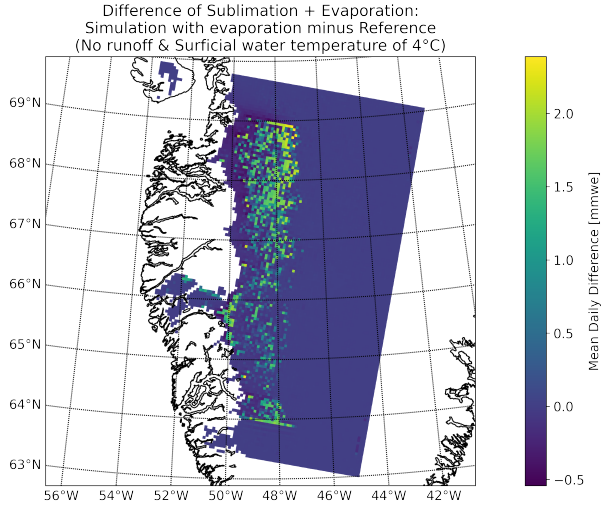
Conclusion on SW evaporation After extensive work to incorporate evaporation into the model, we concluded that it strongly depends on the temperature of the surface water, which is completely normal. Without modifying the water temperature, we do not observe significant changes in the SMB. Therefore, since it is quite likely that the surface water is slightly warmer than the ice in reality, it would be necessary to model its temperature accurately to represent its evaporation in the MAR credibly. We cannot go further without more information about the water temperature. All we can assert is that the impact on the SMB is weak when the temperature is kept at zero degrees Celsius.



(a) Mean daily sublimation between May 2019 and September 2019. Simulation without our modifications to the code.

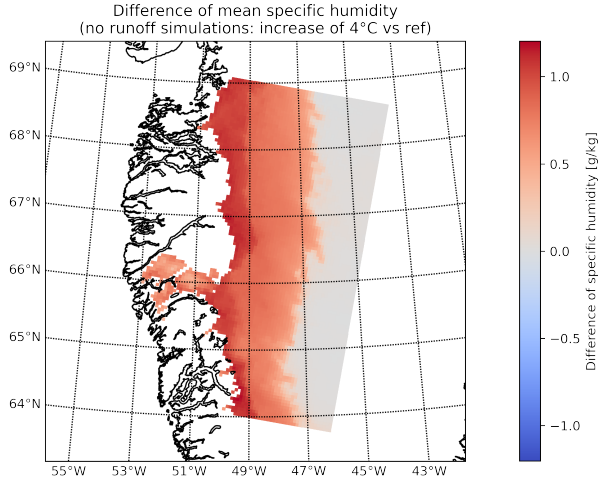


(b) Mean daily sublimation and evaporation between May 2019 and September 2019. Simulation with our modifications to the code.

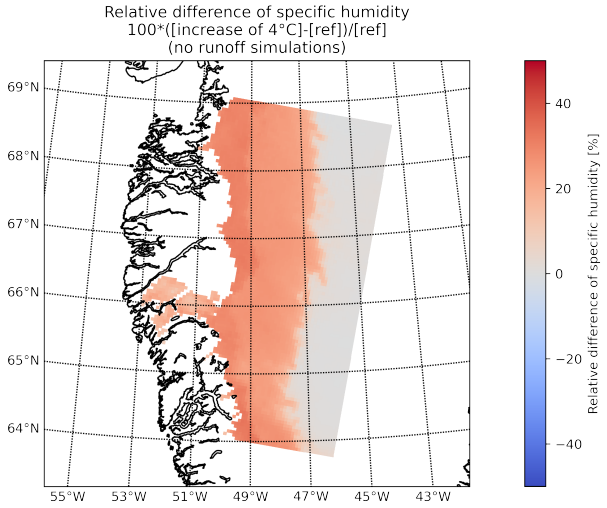


(c) Difference between (4.32b) and (4.32a).

Figure 4.32: Impact of the modifications (adding evaporation over the lakes, assuming that the surface water temperature is 4°C higher than the ice temperature and multiplying by 80 the area occupied by SGLs, *cf.* text) on the amount of sublimated and evaporated water on the ice sheet.

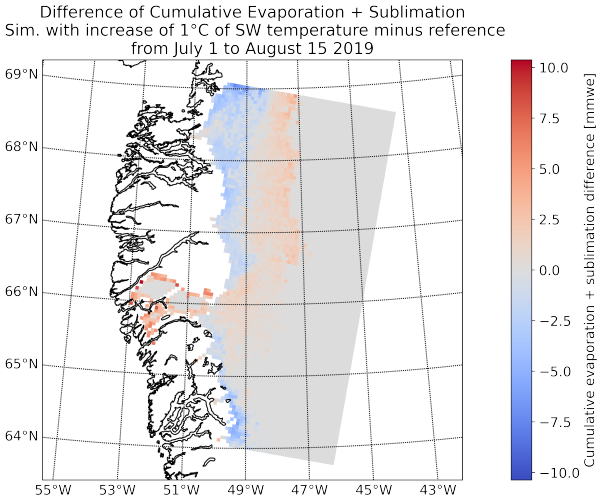


(a) Difference of mean specific humidity (at 2m above the surface) between a simulation where we increase the surficial water temperature by 4°C and the reference (we subtract the specific humidity of the second simulation to the one of the first) from May 2019 to September 2019.

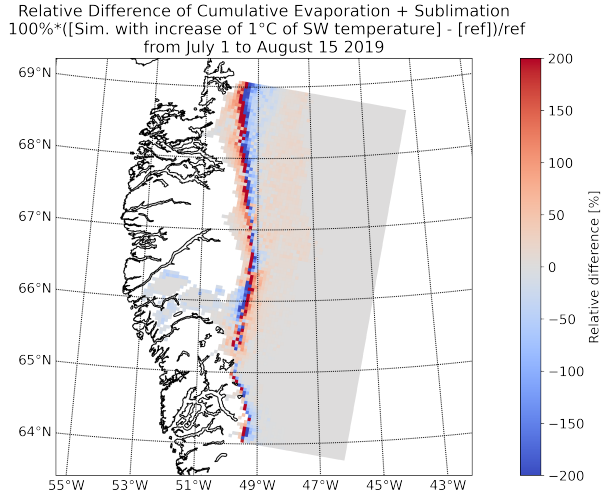


(b) Relative difference of mean specific humidity (at 2m above the surface) between a simulation where we increase the surficial water temperature by 4°C and the reference (we subtract the specific humidity of the second simulation to the one of the first) from May 2019 to September 2019. We normalise by 2-meters specific humidity of the reference simulation.

Figure 4.33: Influence of the surface water temperature on the 2 meters specific humidity.



(a) Difference in cumulative evaporation and sublimation between July 1st and August 15th, between the simulation where evaporation is included and the surface water temperature is assumed to be 1°C higher than the underlying ice, and the reference simulation (without evaporation).



(b) Relative difference in cumulative evaporation and sublimation between July 1st and August 15th, between the simulation where evaporation is included and the surface water temperature is assumed to be 1°C higher than the underlying ice, and the reference simulation (without evaporation).

Figure 4.34: Comparison of the simulation where we replace the sublimation by evaporation whenever the SWH is positive and assuming the water temperature is 1°C larger than the ice temperature to the reference.

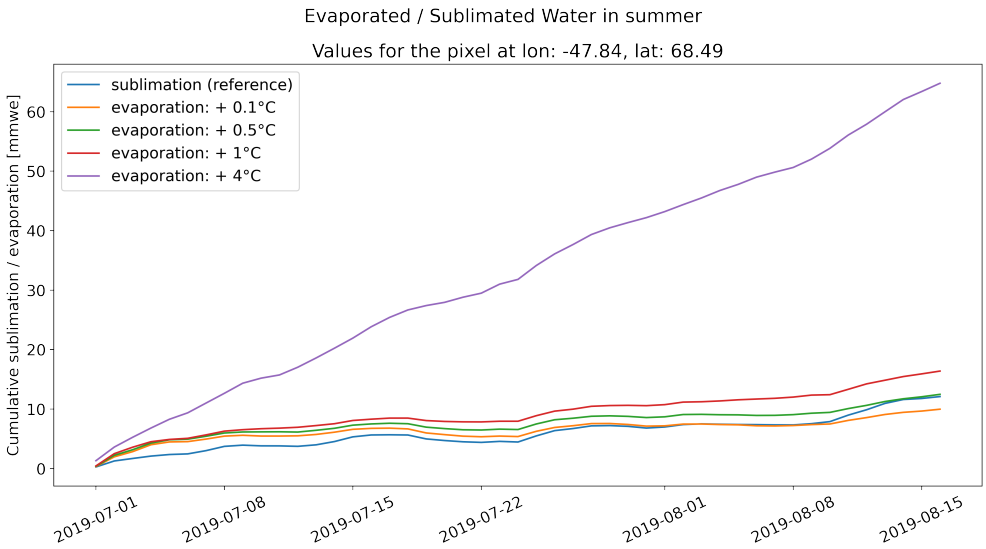


Figure 4.35: Cumulative evaporation (and sublimation for the reference) between July 1 and August 15 if we artificially increase the water temperature (i.e., we assume that the water temperature is 0.1, 0.5, 1 and 4 °C higher than the temperature of the uppermost ice layer). We present here the values for a pixel in the ablation zone.

4.2.6 Attempt to Estimate the Impact of Supra-glacial Lakes (SGLs) on Surface Mass Balance (SMB) with MAR

We can now move on to discussing the latest simulations we have run. To attempt to model the impact of lakes, we have adapted the parameter t^* according to the observations made in Chapter 3. Specifically, we conducted several simulations where c_1 and c_2 (*cf.* Eq. (4.4)) were functions of the fraction of each pixel's surface area occupied by lakes. The goal was to consider that water had a longer residence time for pixels where more lakes are observed in satellite images. This is a way to account for small-scale hydrological processes in the large-scale surface mass balance.

The simulations we ran for this purpose included the following modification, with different values for the constants t_1, t_2, t_3, and t_4 that all have the dimension of time (in addition to the corrections allowing for refreezing, etc., mentioned earlier) (Eqs. (4.12) & (4.13)):

$$c_1 = t_1 + t_2 \times f \tag{4.12}$$
$$c_2 = t_3 + t_4 \times f \tag{4.13}$$

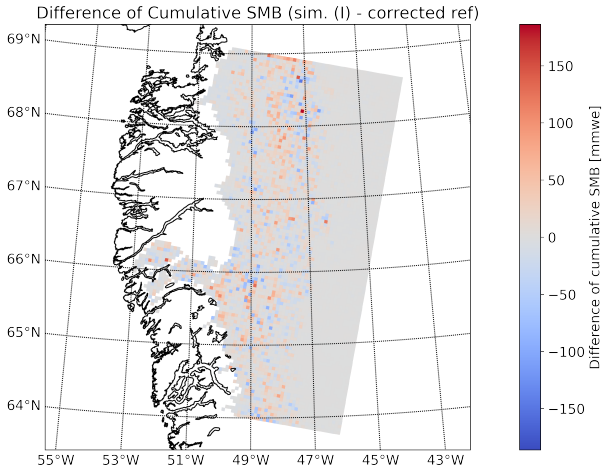
Where $f \in [0, 1]$ is the fraction of a pixel that is covered by SGLs, according to our satellite-observation-based dataset, this is such that the higher the observed area occupied by SGLs, the slower the runoff. These (Eqs. (4.12) & (4.13)) are, of course, linear relationships designed such that c_1 & c_2 evolves linearly from a given value (t_1 & t_3) to another ($t_1 + t_2$ & $t_3 + t_4$) for increasing values of f. This constitutes the first approach to accounting for small-scale hydrology's impact on the larger-scale state of the ice sheet, and ultimately on its surface mass balance. Nevertheless, it is important to note that this approach is probably not optimal, but rather intuitive. The hypotheses of our small model are reasonable, as we consider that the characteristic time associated with runoff is proportional to observations, but its expression (which does not take into account the properties of the snowpack, etc.) and its constants are likely simplistic.

Important note: We also imposed in all subsequent simulations that when SWH exceeds 1000mmwe, the surplus automatically runs off. This is justified as we consider it unrealistic to have such large water depths over the area of one pixel (i.e., $\sim 25\text{km}^2$).

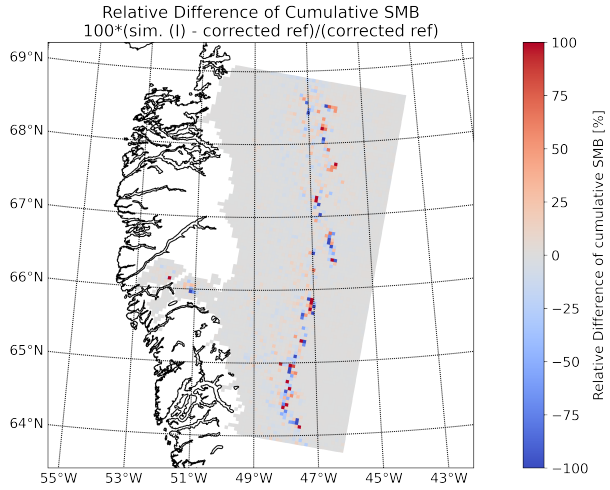
The values of t_1, t_2, t_3, and t_4 (as well as other specificities), for the simulations we launched for this section are listed in Table 4.3. To avoid confusion with the simulations launched previously (without our corrections to MAR), we will use Roman numerals. The first simulation launched had the following values: $t_1 = 3\text{h}$, $t_2 = 7\text{days} - 3\text{h}$,

Table 4.3: Parameters of t_1, t_2, t_3, and t_4 and of the different test "simulations" launched to attempt to estimate the effect of SGLs on the summer 2019 SMB in our southwest GrIS domain.

Simulation	t_1	t_2	t_3	t_4	Comment
I	3h	7 days - 3h	15h	21 days - 15h	/
II	3h	14 days - 3h	15h	42 days - 15h	/
III	3h	140 days - 3h	15h	420 days - 15h	$f = \max(1, 10 \times$ fraction covered by SGL$)$



(a) Difference of cumulative SMB (from May 2019 to December 2019) between simulation I and the "corrected" reference (i.e., the simulation where we do not modify anything in the MARv3.14 code except our correction mentioned in the text).



(b) Relative difference of cumulative SMB (from May 2019 to December 2019) between simulation I and the "corrected" reference (i.e., the same as in (a)). As usual, the large values close to the equilibrium lines are due to the small denominator (*cf.* Eq. (4.6)).

Figure 4.36: Comparison of simulation I (*cf.* Table 4.3) to the corrected reference.

$t_3 = 15\text{h}$, and $t_4 = 21\text{days} - 15\text{h}$. This did not lead to significant variations in the SMB, as shown in Figure 4.36. Indeed, the values of t_1, t_2, t_3, and t_4 are not so large and f is generally very small (it is 0.3 % on average over the pixels covered by 90% or more by ice/snow).

We then launched a second simulation (namely simulation II *cf.* Table 4.3), changing only the maximum "characteristic time" (t^*) achievable for each pixel. As shown in Figure 4.37, this did not significantly alter the results. This figure also demonstrates that for simulations I and II, it is even difficult to establish a clear link between the lake fraction and the difference in cumulative SMB from May to December 2019.

To verify under which assumptions our model could significantly impact the SMB, we essentially "boosted" the parameters for simulation III (*cf.* Table 4.3). We notably multiply f (*cf.* Eqs. (4.12) & (4.13)) by ten. This can be justified by the fact that our data might be biased by the scale at which we downloaded them (*cf.* Section 3.1.3), potentially leading to an underestimation of f (although probably not by a factor of 10 ...). Beyond this, we can justify this choice with a desire to experiment with what would happen with ten times more lakes. Note also that the surface area of a lake is likely smaller than that of its watershed. The results are presented in Figure 4.38.

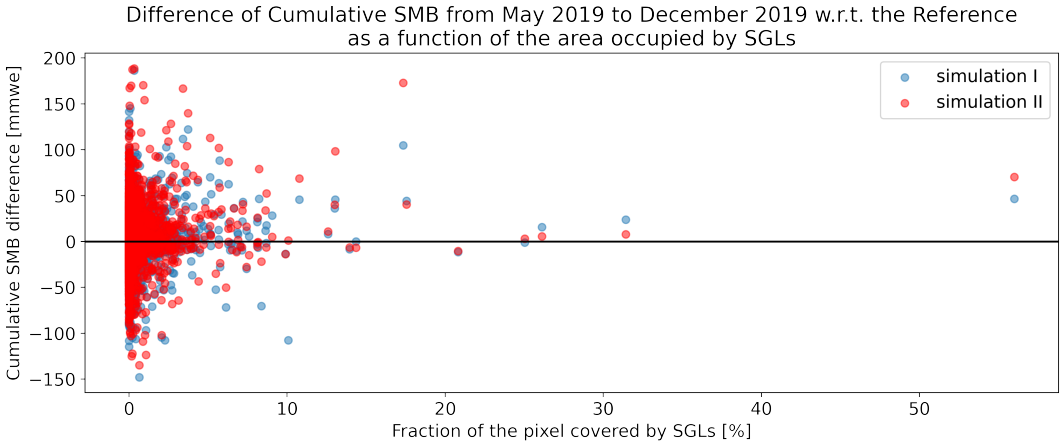
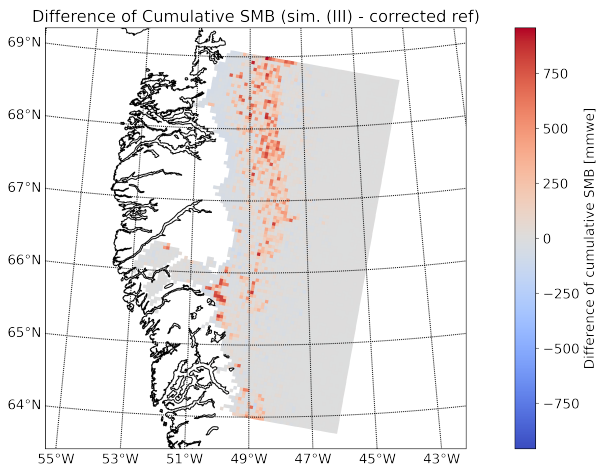
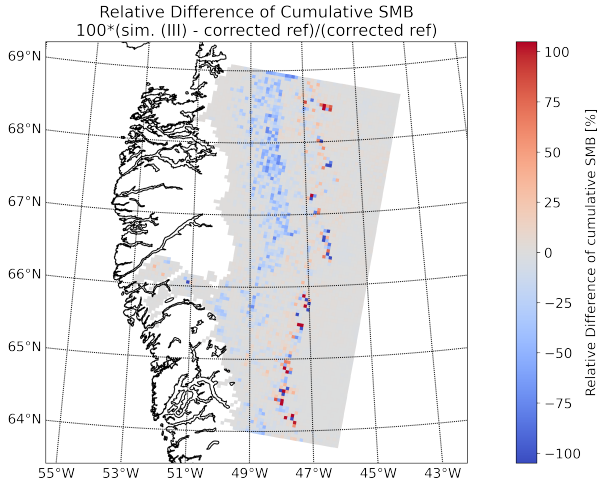


Figure 4.37: Difference of cumulative SMB with respect to the corrected reference as a function of the fraction of each pixel covered by SGLs, for simulations I and II (*cf.* Table 4.3). Each point on the scatter plot corresponds to a pixel in our MAR domain.



(a) Difference of cumulative SMB (from May 2019 to December 2019) between simulation III and the "corrected" reference (i.e., the simulation where we do not modify anything in the MARv3.14 code except our correction mentioned in the text).



(b) Relative difference of cumulative SMB (from May 2019 to December 2019) between simulation III and the "corrected" reference (i.e., the same as in (a)). Let us note that the cumulative SMB is negative, hence the apparent "sign difference" between (a) and (b) (*cf* Eq. (4.6)).

Figure 4.38: Comparison of simulation III (*cf.* Table 4.3) to the corrected reference. Note that we prevent SWH from exceeding 1000mmwe by forcing the direct runoff of the surplus.

4.3 Discussion

Let us now discuss the results obtained so far. We focus on the results of the part of the work that deals with SMB modelling, as it is the most relevant to answer our research question.

To summarise our results, we first verified that it was, acceptable to lead numerical experiments on our MAR domain on the Southwest of the GrIS at a resolution of 5km×5km, after comparing it to a reference MAR simulation.

Through several simulations, we then assessed the sensitivity of the modelled SMB to certain parameters. We have shown that, as expected, modifying the runoff characteristic time (t^*) parameters (c_1 and c_2) of the model from Zuo and Oerlemans (1996) affects the SWH. We then underlined that the effect of SWH on the albedo, and thus on the melt, was very weak, in the current version of MAR. The influence of the albedo on the SWH was then further explored, it was shown that if we assume a low surface albedo, the melt increases, and conversely, in agreement with the SEB.

We fixed an error in the code allowing us to evaluate the impact of t^* on the SMB through refreezing. We concluded that setting large values for c_1, and c_2 allows water that melts in summer to freeze in winter, which can increase the SMB.

A "test example" (in Section 4.2.4) allowed to remind that water should flow from one pixel to another and that it is a "weakness" of MAR.

We evaluated the impact that surface water evaporation could have on the SMB. We concluded that it was necessary to account for surface water temperature to satisfactorily assess this process.

Finally, we adapted the model from Zuo and Oerlemans (1996) to the satellite observations of SGLs. We have shown that with our initial hypotheses (on the t_1 t_2, t_3, and t_4 parameters), the effect on the SMB was weak. However, increasing the values of those parameters and of the area covered by SGLs has shown that the refreezing of surface water flowing slowly could influence the SMB.

It would be overly optimistic to draw conclusions about the influence of lakes on the SMB at this stage. Indeed, we should not forget that our model for the parameters c_1, and c_2 might be too simple. Also, it is important to keep in mind that the Zuo and Oerlemans (1996) model may not be perfect. Yet, it is this model that governs the runoff of meltwater, which is the most significant loss term in the SMB (we have shown that t^* can influence the SMB). And it is certain that allowing water to flow from one pixel to another would allow a more realistic representation of the reality.

If we cannot definitively conclude about evaporation, it does seem that if lakes have a significant impact on the SMB, it is primarily through their influence on runoff, which facilitates refreezing. The effect of albedo should also be evaluated more thoroughly, as this parameter can influence melting.

Regarding the limitations of our study, it is evident that, as we have just mentioned, the representation of runoff in MAR is likely too far from reality. Furthermore, it should be noted that we only worked on a restricted area of the GrIS and on a single year.

Chapter 5

Conclusion and Perspectives

5.1 Conclusion

Regional climate models are currently the best tools available to scientists for estimating the surface mass balance of ice sheets. However, these models, and the MAR in particular, often have a simplistic or non-existent representation of meltwater runoff and its interaction with the surface, despite runoff being the most significant term of mass loss in the SMB. Satellite imagery clearly identifies supra-glacial lakes in the ablation zone of the Greenland ice sheet and other stretches of water. Observations thus demonstrate that water does not directly flow from its melting location to the ocean, as assumed by the models. In this context, we raised the question of the impact of supra-glacial hydrology on the GrIS SMB.

We have therefore conducted an in-depth study of the treatment of runoff and surface water in MAR. Among the parameters evaluated, the characteristic runoff time can, when fixed to large values, significantly impact the SMB by causing summer meltwater to freeze. Other parameters, such as albedo, can also influence the SMB.

At the end of our numerical simulations analysis, it appears that the calibration of certain evaluated parameters, namely albedo and runoff characteristic time, can influence the modelled SMB over a year. Therefore, it is important to emphasize that calibrating these values is crucial. Certainly, it seems that it would likely be beneficial to revise the way water flows in the MAR, to more accurately represent reality, and allow to answer the research question.

5.2 Perspectives

Throughout this report, we have highlighted that the representation of runoff within the MAR model lacks the necessary sophistication to adequately account for the impact of surface hydrology. Revisiting and refining the model's treatment of runoff would likely be beneficial to determine whether the omission of surface hydrological processes might skew our estimations and projections of the surface mass balance.

One of the primary challenges lies in adhering to computational resource limitations, which necessitates balancing the model's sophistication with its computational demands. Running a detailed water flow model at the pixel scale and linking the results to other forcing variables could allow us to evaluate the effects of small-scale processes. From these results, we could deduce a simplified model to couple with MAR, although success is not guaranteed.

To reduce computational time, we could also implement the elevation class methodology, dividing each MAR pixel into several sub-pixels with their own topography, allowing the atmospheric part of MAR to run at a lower resolution than its snow-ice module. A hydrologic model could be integrated into MAR to explicitly simulate SGL formation in summer and perform future projections as the ablation zone is expected to expand in warmer climates. Our developments could be based on existing hydrological models, such as SHED (Gantayat et al., 2023), mentioned in Section 2.2.

Bibliography

Brajkovic, J. (2022). Estimation à long terme (2300) de la capacité de rétention de l'eau de fonte de la calotte du groenland [Available at http://hdl.handle.net/2268.2/15912].

Brun, E., David, P., Sudul, M., & Brunot, G. (1992). A numerical model to simulate snow-cover stratigraphy for operational avalanche forecasting. Journal of Glaciology, 38(128), 13–22. https://doi.org/doi:10.3189/S0022143000009552

Cassotta, S., Derksen, C., Ekaykin, A., Hollowed, A., Kofinas, G., Mackintosh, A., Melbourne-Thomas, J., Muelbert, M., Ottersen, G., Pritchard, H., et al. (2022). In Special report on ocean and cryosphere in a changing climate intergovernmental panel on climate change (ipcc). Cambridge University Press.

Clerx, N., Machguth, H., Tedstone, A., Jullien, N., Wever, N., Weingartner, R., & Roessler, O. (2022). In situ measurements of meltwater flow through snow and firn in the accumulation zone of the sw greenland ice sheet. EGUsphere, 2022, 1–31. https://doi.org/10.5194/egusphere-2022-71

Cushman-Roisin, B., & Beckers, J.-M. (2011). Introduction to geophysical fluid dynamics: Physical and numerical aspects. Academic press.

Dow, C. F., Kulessa, B., Rutt, I., Tsai, V., Pimentel, S., Doyle, S., Van As, D., Lindbäck, K., Pettersson, R., Jones, G., et al. (2015). Modeling of subglacial hydrological development following rapid supraglacial lake drainage. Journal of Geophysical Research: Earth Surface, 120(6), 1127–1147.

ECMWF. (2024). Era5 hourly data on single levels from 1940 to present. Retrieved July 21, 2024, from https://cds.climate.copernicus.eu/cdsapp#!/dataset/reanalysis-era5-single-levels?tab=overview

Enderlin, E. M., Howat, I. M., Jeong, S., Noh, M.-J., Van Angelen, J. H., & Van Den Broeke, M. R. (2014). An improved mass budget for the greenland ice sheet. Geophysical Research Letters, 41(3), 866–872. https://doi.org/https://doi.org/10.1002/2013GL059010

ESA. (2024). Eo browser. Retrieved March 22, 2024, from https://www.sentinel-hub.com/explore/eobrowser/

Fettweis, X. (2006). Reconstruction of the 1979-2005 greenland ice sheet surface mass balance using satellite data and the regional climate model mar [Available at https://dial.uclouvain.be/pr/boreal/en/object/boreal%3A5384/datastreams]. Université Catholique de Louvain Belgique.

Fettweis, X. (2023). Real-time state of the 2022-2023 greenland ice sheet surface mass balance as simulated by the regional climate model mar. Retrieved May 10, 2024, from https://www.climato.uliege.be/cms/c_5652668/fr/climato-greenland?id=c_5652668&id=c_5652668

Fettweis, X., Hofer, S., Krebs-Kanzow, U., Amory, C., Aoki, T., Berends, C. J., Born, A., Box, J. E., Delhasse, A., Fujita, K., et al. (2020). Grsmbmip: Intercomparison of the modelled 1980–2012 surface mass balance over the greenland ice sheet. The Cryosphere, 14(11), 3935–3958.

François, L., & Munhoven, G. (2023). Climate change and impacts. Retrieved August 13, 2024, from https://www.programmes.uliege.be/cocoon/20242025/cours/SPAT0027-3.html

Gaba, E. (2013). La projection de mercator. Retrieved March 20, 2024, from https://fr.wikipedia.org/wiki/Indicatrice_de_Tissot#/media/Fichier:Tissot_indicatrix_world_map_Mercator_proj.svg

Gallée, H., & Schayes, G. (1994). Development of a three-dimensional meso-γ primitive equation model: Katabatic winds simulation in the area of terra nova bay, antarctica. Monthly Weather Review, 122(4), 671–685.

Gantayat, P., Banwell, A. F., Leeson, A. A., Lea, J. M., Petersen, D., Gourmelen, N., & Fettweis, X. (2023). A new model for supraglacial hydrology evolution and drainage for the greenland ice sheet (shed v1. 0). Geoscientific Model Development, 16(20), 5803–5823. https://doi.org/10.5194/gmd-16-5803-2023

Geurts, P., & Wehenkel, L. (2023). Introduction to machine learning [Lecture at University of Liège].

Goelzer, H., Huybrechts, P., Fürst, J., Nick, F., Andersen, M., Edwards, T., Fettweis, X., Payne, A., & Shannon, S. (2013). Sensitivity of greenland ice sheet projections to model formulations. Journal of Glaciology, 59(216), 733–749. https://doi.org/10.3189/2013JoG12J182

Google. (2023). Supervised classification. Retrieved January 25, 2024, from https://developers.google.com/earth-engine/guides/classification

Google. (2024). Scale. Retrieved August 1, 2024, from https://developers.google.com/earth-engine/guides/scale

Gregory, J., & Huybrechts, P. (2006). Ice-sheet contributions to future sea-level change. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, 364(1844), 1709–1732.

GROCE. (2018). Satellite image (sentinel-2 true colour image, © esa 2018) of the same supraglacial lake. Retrieved November 4, 2023, from https://groce.de/en/our-projects/7-supraglacial-lakes/

Grodent, D. (2023). Atmosphere of the earth [Lecture at University of Liège]. Retrieved August 10, 2024, from http://www.umccb.ulg.ac.be/Edu/cours_f.php

Hanna, E., Huybrechts, P., Cappelen, J., Steffen, K., Bales, R. C., Burgess, E., McConnell, J. R., Peder Steffensen, J., Van den Broeke, M., Wake, L., et al. (2011). Greenland ice sheet surface mass balance 1870 to 2010 based on twentieth century reanalysis, and links with global climate forcing. Journal of Geophysical Research: Atmospheres, 116(D24). https://doi.org/https://doi.org/10.1029/2011JD016387

Hanna, E., Navarro, F. J., Pattyn, F., Domingues, C. M., Fettweis, X., Ivins, E. R., Nicholls, R. J., Ritz, C., Smith, B., Tulaczyk, S., et al. (2013). Ice-sheet mass balance and climate change. Nature, 498(7452), 51–59. https://doi.org/https://doi.org/10.1038/nature12238

Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., Kern, R., Picus, M., Hoyer, S., van Kerkwijk, M. H., Brett, M., Haldane, A., del Río, J. F., Wiebe, M., Peterson, P., … Oliphant, T. E. (2020). Array programming with NumPy. Nature, 585(7825), 357–362. https://doi.org/10.1038/s41586-020-2649-2

Hastie, T., Tibshirani, R., Friedman, J. H., & Friedman, J. H. (2009). The elements of statistical learning: Data mining, inference, and prediction (2). Springer.

Hu, J., Huang, H., Chi, Z., Cheng, X., Wei, Z., Chen, P., Xu, X., Qi, S., Xu, Y., & Zheng, Y. (2021). Distribution and evolution of supraglacial lakes in greenland during the 2016–2018 melt seasons. Remote Sensing, 14(1), 55. https://doi.org/https://doi.org/10.3390/rs14010055

Huisman, O., de By, R. A., et al. (2009). Principles of geographic information systems. ITC Educational Textbook Series, 1, 17.

Hunter, J. D. (2007). Matplotlib: A 2d graphics environment. Computing in Science & Engineering, 9(3), 90–95. https://doi.org/10.1109/MCSE.2007.55

IMBIE. (2018). Mass balance of the antarctic ice sheet from 1992 to 2017. Nature, 558(7709), 219–222. https://doi.org/DOI:10.1038/s41586-018-0179-y

IMBIE. (2020). Mass balance of the greenland ice sheet from 1992 to 2018. Nature, 579(7798), 233–239. https://doi.org/DOI:10.1038/s41586-019-1855-2

Jansson, P., Näslund, J.-O., & Rodhe, L. (2007). Ice sheet hydrology-a review.

Kittel, C. (2021). Present and future sensitivity of the antarctic surface mass balance to oceanic and atmospheric forcings: Insights with the regional climate model mar [Available at https://hdl.handle.net/2268/258491]. ULiège-Université de Liège.

Kunimune, J. (2018). The lambert azimuthal equal-area projection with tissot's indicatrix of deformation. Retrieved March 20, 2024, from https://en.wikipedia.org/wiki/Lambert_azimuthal_equal-area_projection#/media/File:Azimuthal_Equal-Area_with_Tissot's_Indicatrices_of_Distortion.svg

Lawhead, J. (2022). Pyshp. Retrieved March 20, 2024, from https://pypi.org/project/pyshp/

Leeson, A., Shepherd, A., Briggs, K., Howat, I., Fettweis, X., Morlighem, M., & Rignot, E. (2015). Supraglacial lakes on the greenland ice sheet advance inland under warming climate. Nature Climate Change, 5(1), 51–55. https://doi.org/10.1038/nclimate2463

Lefebre, F., Gallée, H., van Ypersele, J.-P., & Greuell, W. (2003). Modeling of snow and ice melt at eth camp (west greenland): A study of surface albedo. Journal of Geophysical Research: Atmospheres, 108(D8). https://doi.org/https://doi.org/10.1029/2001JD001160

Mankoff, K. D., Noël, B., Fettweis, X., Ahlstrøm, A. P., Colgan, W., Kondo, K., Langley, K., Sugiyama, S., Van As, D., & Fausto, R. S. (2020). Greenland liquid water discharge from 1958 through 2019. Earth System Science Data, 12(4), 2811–2841. https://doi.org/https://doi.org/10.5194/essd-12-2811-2020

Marbaix, P., Gallée, H., Brasseur, O., & van Ypersele, J.-P. (2003). Lateral boundary conditions in regional climate models: A detailed study of the relaxation procedure. Monthly weather review, 131(3), 461–479. https://doi.org/https://doi.org/10.1175/1520-0493(2003)131⟨0461:LBCIRC⟩2.0.CO;2

Melling, L., Leeson, A., McMillan, M., Maddalena, J., Bowling, J., Glen, E., Sandberg Sørensen, L., Winstrup, M., & Lørup Arildsen, R. (2024). Evaluation of satellite methods for estimating supraglacial lake depth in southwest greenland. The Cryosphere, 18(2), 543–558. https://doi.org/10.5194/tc-18-543-2024

Miller, J. Z., Long, D. G., Jezek, K. C., Johnson, J. T., Brodzik, M. J., Shuman, C. A., Koenig, L. S., & Scambos, T. A. (2020). Brief communication: Mapping greenland's perennial firn aquifers using enhanced-resolution l-band brightness temperature image time series. The Cryosphere, 14(9), 2809–2817. https://doi.org/https://doi.org/10.5194/tc-14-2809-2020

Morice, C. P., Kennedy, J. J., Rayner, N. A., Winn, J. P., Hogan, E., Killick, R. E., Dunn, R. J. H., Osborn, T. J., Jones, P. D., & Simpson, I. R. (2021). An updated assessment of near-surface temperature change from 1850: The hadcrut5 data set. Journal of Geophysical Research: Atmospheres, 126, Article e2019JD032361. https://doi.org/10.1029/2019JD032361

Morlighem, M., Williams, C. N., Rignot, E., An, L., Arndt, J. E., Bamber, J. L., Catania, G., Chauché, N., Dowdeswell, J. A., Dorschel, B., et al. (2017). Bedmachine v3: Complete bed topography and ocean bathymetry mapping of greenland from multibeam echo sounding combined with mass conservation. Geophysical research letters, 44(21), 11–051. https://doi.org/10.1002/2017GL074954

Mouginot, J., Rignot, E., Scheuchl, B., & Millan, R. (2017). Comprehensive annual ice sheet velocity mapping using landsat-8, sentinel-1, and radarsat-2 data. Remote Sensing, 9(4). https://doi.org/10.3390/rs9040364

Neckel, N. (2018). Photograph of a meltwater lake on the 79°n glacier. Retrieved November 4, 2023, from https://groce.de/en/our-projects/7-supraglacial-lakes/

Notz, D., & Stroeve, J. (2016). Observed arctic sea-ice loss directly follows anthropogenic co2 emission. Science, 354(6313), 747–750. https://doi.org/DOI:10.1126/science.aag2345

Nowicki, S. M. J., Payne, A., Larour, E., Seroussi, H., Goelzer, H., Lipscomb, W., Gregory, J., Abe-Ouchi, A., & Shepherd, A. (2016). Ice sheet model intercomparison project (ismip6) contribution to cmip6. Geoscientific Model Development, 9(12), 4521–4545. https://doi.org/10.5194/gmd-9-4521-2016

Otosaka, I. N., Shepherd, A., Ivins, E. R., Schlegel, N.-J., Amory, C., van den Broeke, M., Horwath, M., Joughin, I., King, M., Krinner, G., et al. (2022). Mass balance of the greenland and antarctic ice sheets from 1992 to 2020. Earth System Science Data Discussions, 2022, 1–33. https://doi.org/https://doi.org/10.5194/essd-15-1597-2023

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12, 2825–2830.

Philpot, W. D. (1989). Bathymetric mapping with passive multispectral imagery. Applied optics, 28(8), 1569–1578.

Pope, A., Scambos, T. A., Moussavi, M., Tedesco, M., Willis, M., Shean, D., & Grigsby, S. (2016). Estimating supraglacial lake depth in west greenland using landsat 8 and comparison with other multispectral methods. The Cryosphere, 10(1), 15–27. https://doi.org/https://doi.org/10.5194/tc-10-15-2016

Pritchard, H. D., Arthern, R. J., Vaughan, D. G., & Edwards, L. A. (2009). Extensive dynamic thinning on the margins of the greenland and antarctic ice sheets. Nature, 461(7266), 971–975. https://doi.org/DOI:10.1038/nature08471

Reijmer, C. H., van den Broeke, M. R., Fettweis, X., Ettema, J., & Stap, L. B. (2012). Refreezing on the greenland ice sheet: A comparison of parameterizations. The Cryosphere, 6(4), 743–762. https://doi.org/10.5194/tc-6-743-2012

Rennermalm, A. K., Moustafa, S. E., Mioduszewski, J., Chu, V., Forster, R., Hagedorn, B., Harper, J. T., Mote, T., Robinson, D. A., Shuman, C., et al. (2013). Understanding greenland ice sheet hydrology using an integrated multi-scale approach. Environmental Research Letters, 8(1), 015017. https://doi.org/DOI10.1088/1748-9326/8/1/015017

Riahi, K., van Vuuren, D. P., Kriegler, E., Edmonds, J., O'Neill, B. C., Fujimori, S., Bauer, N., Calvin, K., Dellink, R., Fricko, O., Lutz, W., Popp, A., Cuaresma, J. C., KC, S., Leimbach, M., Jiang, L., Kram, T., Rao, S., Emmerling, J., ... Tavoni, M. (2017). The shared socioeconomic pathways and their energy, land use, and greenhouse gas emissions implications: An overview. Global Environmental Change, 42, 153–168. https://doi.org/https://doi.org/10.1016/j.gloenvcha.2016.05.009

Richter-Menge, J., Overland, J., Mathis, J., Osborne, E., et al. (2017). Arctic report card 2017: Arctic shows no sign of returning to reliably frozen region of recent past decades. https://doi.org/https://doi.org/10.25923/8ntk-7817

Sasgen, I., Wouters, B., Gardner, A. S., King, M. D., Tedesco, M., Landerer, F. W., Dahle, C., Save, H., & Fettweis, X. (2020). Return to rapid ice loss in greenland and record loss in 2019 detected by the grace-fo satellites. Communications Earth & Environment, 1(1), 1–8. https://doi.org/https://doi.org/10.1038/s43247-020-0010-1

Smith, L. C., Yang, K., Pitcher, L. H., Overstreet, B. T., Chu, V. W., Rennermalm, Å. K., Ryan, J. C., Cooper, M. G., Gleason, C. J., Tedesco, M., et al. (2017). Direct measurements of meltwater runoff on the greenland

ice sheet surface. Proceedings of the National Academy of Sciences, 114(50), E10622–E10631. https://doi.org/https://doi.org/10.1073/pnas.1707743114

Tedesco, M., & Fettweis, X. (2020). Unprecedented atmospheric conditions (1948–2019) drive the 2019 exceptional melting season over the greenland ice sheet. The Cryosphere, 14. https://doi.org/10.5194/tc-14-1209-2020

van den Broeke, M., Box, J., Fettweis, X., Hanna, E., Noël, B., Tedesco, M., van As, D., van de Berg, W. J., & van Kampenhout, L. (2017). Greenland ice sheet surface mass loss: Recent developments in observation and modeling. Current Climate Change Reports, 3, 345–356. https://doi.org/https://doi.org/10.1007/s40641-017-0084-8

van Meijgaard, E., Ulft, L., Berg, W., Bosvelt, F., Hurk, B., Lenderink, G., & Siebesma, A. (2008). The knmi regional atmospheric model racmo version 2.1 [Available at https://cdn.knmi.nl/knmi/pdf/bibliotheek/knmipubTR/TR302.pdf]. Tech. Rep. 302, KNMI.

Woodcock, C. E., & Strahler, A. H. (1987). The factor of scale in remote sensing. Remote sensing of Environment, 21(3), 311–332. https://doi.org/https://doi.org/10.1016/0034-4257(87)90015-0

Yang, K., Sommers, A., Andrews, L. C., Smith, L. C., Lu, X., Fettweis, X., & Li, M. (2020). Intercomparison of surface meltwater routing models for the greenland ice sheet and influence on subglacial effective pressures. The Cryosphere, 14(10), 3349–3365. https://doi.org/10.5194/tc-14-3349-2020

Yang, K., Smith, L. C., Andrews, L. C., Fettweis, X., & Li, M. (2022). Supraglacial drainage efficiency of the greenland ice sheet estimated from remote sensing and climate models. Journal of Geophysical Research: Earth Surface, 127(2). https://doi.org/DOI:10.1029/2021JF006269

Yang, K., Smith, L. C., Karlstrom, L., Cooper, M. G., Tedesco, M., van As, D., Cheng, X., Chen, Z., & Li, M. (2018). A new surface meltwater routing model for use on the greenland ice sheet surface. The Cryosphere, 12(12), 3791–3811. https://doi.org/10.5194/tc-12-3791-2018

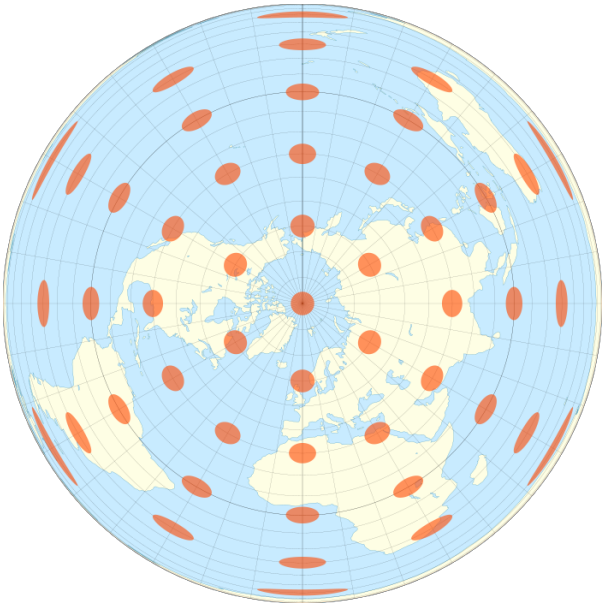
Yen, Y.-C. (1981). Review of thermal properties of snow, ice, and sea ice (81). US Army, Corps of Engineers, Cold Regions Research; Engineering Laboratory.

Zuo, Z., & Oerlemans, J. (1996). Modelling albedo and specific balance of the greenland ice sheet: Calculations for the søndre strømfjord transect. Journal of Glaciology, 42(141), 305–317. https://doi.org/doi:10.3189/S0022143000004160

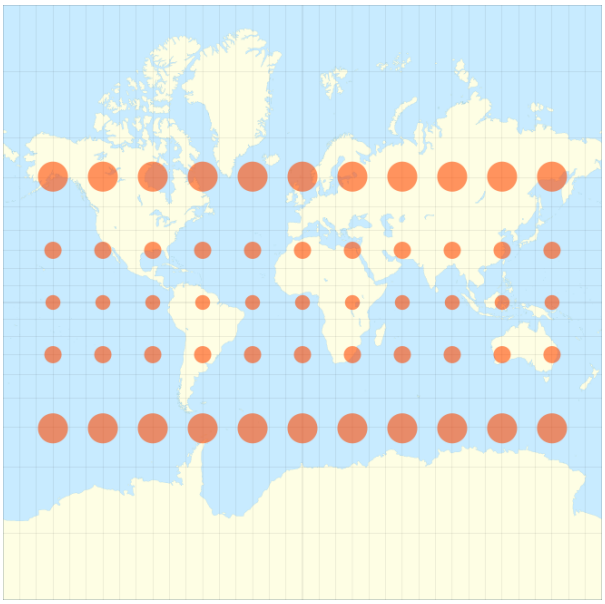
Appendix A

Comment on Maps & Figures

Most of the maps shown in this document were created using the mpl_toolkits.basemap and matplotlib python libraries (Hunter, 2007). The cartographic projection is the Lambert azimuthal equal area projection. This choice was made because it allows to visually interpret maps without bias. For example, if we map circles of equal radius (*i.e.* Tissot indicatrixes) with this projection, they will be strongly deformed, but all of them will have the same area. This property is interesting in our case because, for example, when working with a non area-conservative projection, if we have an area where a quantity (*e.g.* the SMB) is strongly above or under the mean, located on the map in a zone where the area are expanded, human eye will interpret this phenomenon as more important than if the same area is located somewhere else (and conversely). Especially when the reader must interpret a lot of maps as it is the case in this document. To illustrate this phenomenon, Figure A.1 (a) & (b) shows Tissot's indicatrices for our projection and for another (the Mercator projection, that is notably used for Figure 1.7).



(a) Lambert Equal Area Projection. Source: Kunimune (2018).



(b) Mercator Projection. Source: Gaba (2013).

Figure A.1: Tissot's indicatrixes for two cartographic projections

An example of code to map a numpy array over the GrIS is available in Listing A.1.

Listing A.1: Generation of a Map Over Greenland

```
import numpy as np
import matplotlib.pyplot as plt
from netCDF4 import Dataset
```

```
import cartopy.crs as ccrs
import cartopy.feature as cfeature
from mpl_toolkits.basemap import Basemap

def make_map(lat_array, lon_array, array, masque=None, titre='titre', label='label',
                    nomfichier='plot_diff.png', colormap='jet', symmetric = False):
    f, ax = plt.subplots(1,1, figsize = (18,8))

    m1 = Basemap(width=1600000,height=2900000,
                resolution='h',projection='laea',
                lat_ts=(np.mean(lat_array)+1),lat_0=(np.mean(lat_array)+1),
                lon_0=np.mean(lon_array),
                ax = ax)
    lons = lon_array
    lats = lat_array
    x, y = m1(lons, lats)

    if masque is not None:
        plotarray = np.ma.masked_where(masque == 0, array)
    else:
        plotarray = array
    if symmetric:
        maximum = np.max(np.array([np.abs(np.max(plotarray)),
        np.abs(np.min(plotarray))]))
        img = m1.pcolormesh(x, y, plotarray, cmap=colormap, vmin =
        -maximum, vmax = maximum)
    else:
        maximum = np.max(plotarray)
        minimum = np.min(plotarray)
        img = m1.pcolormesh(x, y, plotarray, cmap=colormap,
        vmin = minimum, vmax = maximum)

    m1.drawcoastlines()
    m1.drawcountries()
    m1.drawparallels(np.arange(int(np.min(lat_array)-10), int(np.max(lat_array)+30), 5),
    labels = [1, 0, 0, 0])
    m1.drawmeridians(np.arange(int(np.min(lon_array)-30), int(np.max(lon_array)+30), 5),
    labels = [0, 0, 0, 1])
    ax.set_title(titre)


    cbar = f.colorbar(img, ax=ax)
    cbar.set_label(label)
    f.savefig(nomfichier, dpi = 350)


make_map(lat_array, lon_array, np.ma.masked_where(topo ==0,topo),
    titre='Surface Height', label="SH [metres]",
    nomfichier="shwholegris.png", colormap='viridis')
```

Appendix B

Physics Behind the Adiabatic Lapse Rate

Let us explain the physics behind Equation (1.1).

The atmosphere, composed primarily of N_2 and O_2 within its first 100 kilometers (referred to as the Homosphere[1]), behaves as a fluid governed by the Navier-Stokes equations, that describe the fluid motions. Without delving into the details (see Cushman-Roisin and Beckers (2011) for instance), the two dominant terms in the equation describing the vertical motion of an air parcel are typically the pressure gradient and the weight of the air parcel. This reflects the idea that if we imagine a "bubble of atmosphere", i.e. an air parcel, the pressure at its base will be slightly lower than at its top. The force that drives fluids toward lower pressure, acting on our air volume, is counterbalanced by the weight of the air parcel, resulting in a nil acceleration. This is mathematically expressed as (Eq. (B.1)):

$$\frac{\partial p}{\partial z} = -\rho g \tag{B.1}$$

Where p is the pressure, z is the vertical coordinate, g is the gravity acceleration, and ρ is the air density.

Source Acknowledgement The following discussion is based on the course from Grodent (2023).

We therefore have a relationship that links pressure and altitude. To connect temperature to altitude, it is important to know that the troposphere, which is the lowest part of the atmosphere (when defining atmospheric layers based on the temperature profile), is generally in convective equilibrium. This implies that if a temperature imbalance occurs, convection is the mechanism that drives the temperature profile toward its equilibrium state. During their vertical motion, air parcels are characterised by energy, which adheres to the first law of thermodynamics (Eq. (B.2)):

$$\delta Q = \delta W + \delta U \tag{B.2}$$

Where δQ is the heat exchange with the environment, δW is the mechanical work associated with the air contraction or expansion, and δU is the variation of the internal energy.

If the motion is rapid, the exchange with other parcels can be neglected, and it can be assumed that $\delta Q = 0$, so that the air parcel is subject to an adiabatic transformation. We have, $\delta W = PdV$ (P is the pressure, and dV the variation of the air mass volume), and $\delta U = C_v dT$ (C_v is the heat capacity of the air, and dT the variation of the temperature). We will develop further the expression of δU.

[1] In fact, the Homosphere can be defined as the part of the atmosphere where the relative concentrations of gases change very little with altitude. It is bounded by the homopause at an altitude of approximately 100 km. Thus, it is the chemical composition, rather than altitude, that defines this layer of the atmosphere.

Using the the ideal gas relationship: $PV = nRT$ (P is the pressure, V the volume, n the number of moles, R the ideal gas constant, and T the temperature (in Kelvins)), one can re-express PdV, and using the Mayer Relationship ($C_p - C_v = R$, where C_p and C_v are respectively the specific heat capacities at constant pressure and at constant volumes), we get (fixing $n = 1$):

$$C_p dT = \frac{RT}{P} dP \tag{B.3}$$

Re-using Eq. (B.1), we get:

$$\frac{dT}{dz} = -\frac{Mg}{C_p} \tag{B.4}$$

Where M is the molar mass (we somehow replaced ρ).

This result corresponds to the dry adiabatic lapse rate. If the air is humid, then as it cools, the water vapour condenses and releases heat (modifying the term δU). This has the effect of reducing the variation in temperature with altitude.

Appendix C

Area Occupied by Supra-glacial Lakes (SGL) as a function of the image resolution

See Figure C.1.

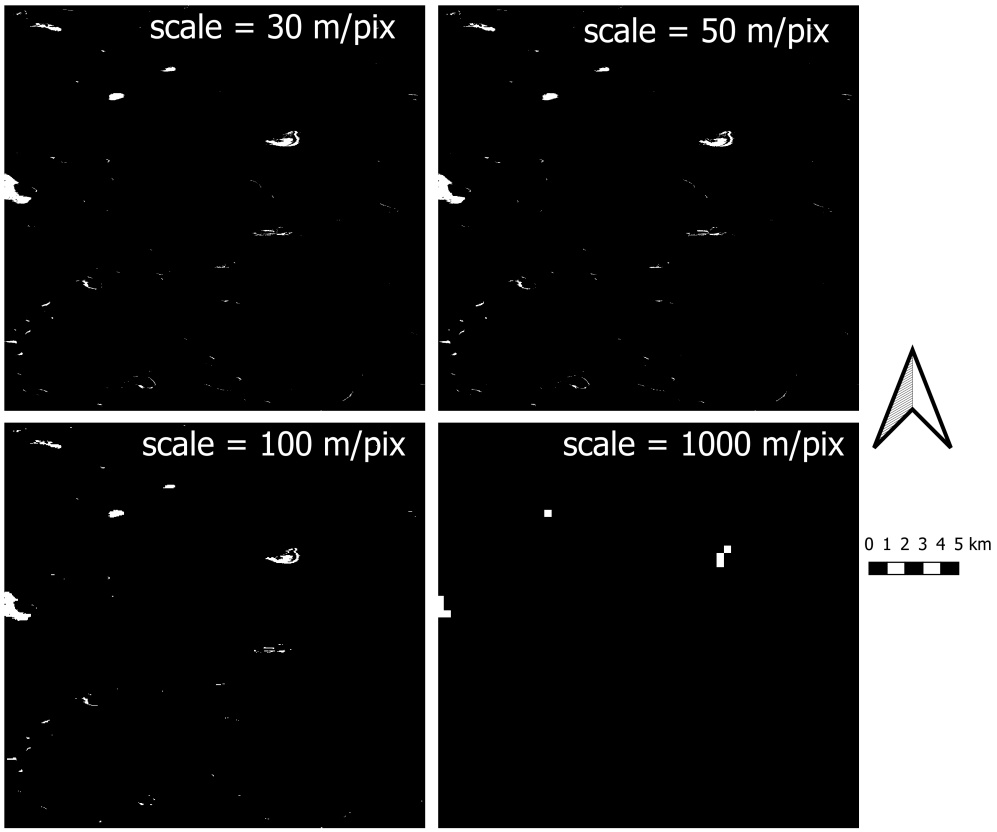


Figure C.1: SGL pixels appears in white.

Appendix D

Modèle Atmosphérique Régional (MAR) Ice Sheet Mask Assessment

MAR provides in its NetCDF output a `MSK` field (standing for mask) (see Figure 3.17 (c)). It is not a function of time. Briefly, it corresponds to the proportion of the pixel area that is considered by the model as covered by ice. In this document, when we write that "a mask is applied to the data", we generally mean that we set the threshold to 50% of the area occupied by ice to distinguish ice sheet pixels from others. Here we compare this mask to satellite data from summer 2019, to assess where the threshold should be put if we want to accurately identify the SGL. This step is crucial in our approach to identifying SGL localisation factors because the slightest error can introduce very large biases. Indeed, if one does not mask an ocean pixel, it will constitute a huge outlier.

As shown in Figure 3.17 (d) there is a lot of inland water east of Greenland (the region highlighted in Figure D.1), so it is a "critical region" because the `MSK` varies "slowly" (the contour lines are well spaced), meaning that a small difference in the threshold translates into a huge horizontal distance.

To compare `MSK` threshold to satellite data, we converted the contour lines of the `MSK` field to shapefiles. This was done in Python following this procedure: First, we used scikit-image (`skimage.measure`) (Lawhead, 2022) to obtain the indexes, that we will denote (i, j), of the pixels, were `MSK` = threshold (`MSK` was converted into a `numpy` array). Since we also have two arrays containing the longitudes and the latitudes of pixel centres. All three arrays are of the same shape. Generally, i and j are real numbers. Second, to obtain the corresponding latitudes (ϕ) and longitudes (λ), we used a kind of IDW interpolation. If we use the same notations as in Figure D.2, the latitude, ϕ, of the (i, j) pixel is obtained through[1] Eq. (D.1).

$$\phi = \frac{\Sigma_{x,y}\phi_{xy}\left((i-i_x)^2+(j-j_y)^2\right)^{-1}}{\Sigma_{x,y}\left((i-i_x)^2+(j-j_y)^2\right)^{-1}} \tag{D.1}$$

Where x and y vary in $\{0, 1\}$. The relationship for the longitude is analogous. This step's result is shown in Figure D.1. Then, using PyShp (Lawhead, 2022), we could convert these contours to ESRI shapefiles assuming that geographic coordinates were in WGS 84. Finally, we can upload this data into QGIS or in GEE.

Let us note that this method for "drawing" the contour lines does not exactly correspond to what happens when we apply a mask to a raster. Actually, the mask is delimited by the boundaries of the edges pixels. Both contour lines are compared in Figure D.3. As the first method is easier to implement, and both do not differ that much, we will keep the method explained above (*i.e.* described with Figure D.2).

The QGIS OpenLayers Plugin enables us to compare our MAR mask contours to Bing Maps Aerial data, as shown in Figure D.4. This method is very efficient as it is not necessary to download high-resolution satellite images. Nevertheless, a serious drawback of this method is that we do not know the date of aerial images! To address this issue, we can download Sentinel-2 images from Sentinelhub (ESA, 2024). As a result, we obtain Figure

[1] *NB* Another solution is to simply convert the indexes to integers in Python, but it is quite arbitrary.

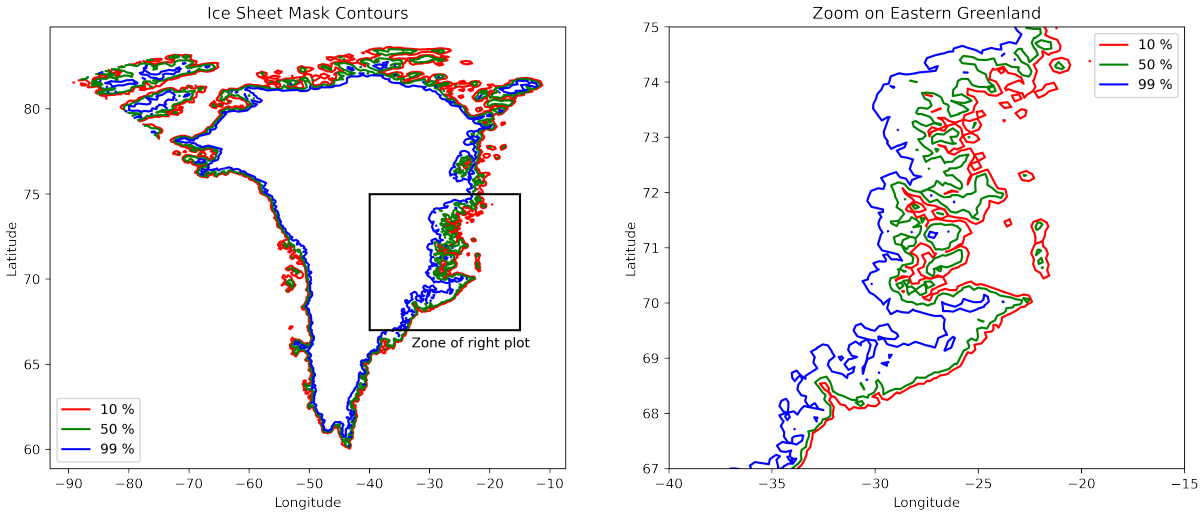


Figure D.1: Contour lines of the mask presented in Figure 3.17 (c). Both maps use a *plate carrée* projection (*i.e.* $x = \lambda$, and $y = \phi$).

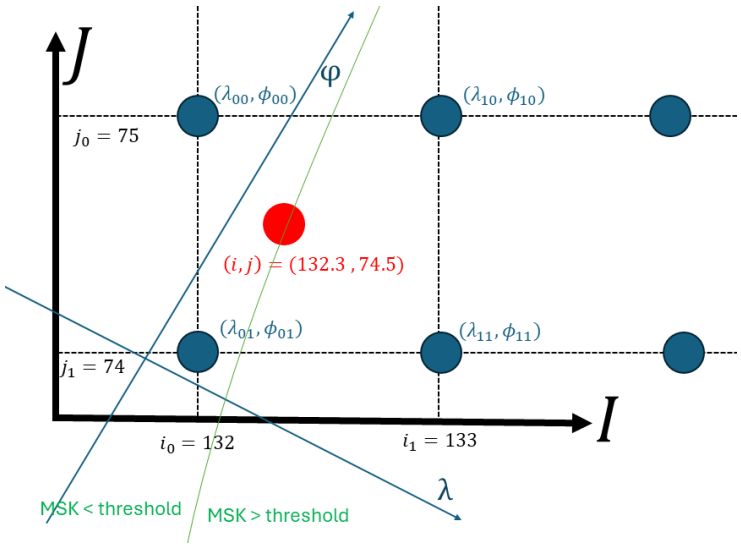


Figure D.2: Notations for the algorithm allowing to obtain latitudes and longitudes of mask contour lines points. Blue circles represent pixel centres. i_0, i_1, j_0, and j_1 are integers whereas i and j are not, hence the arbitrary values written as example in the illustration.

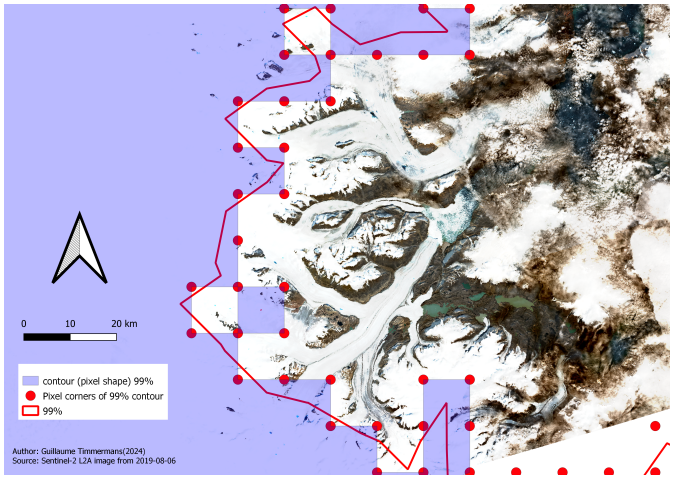


Figure D.3: Comparison of the two methods for "drawing" the 99% contour lines.

D.5. It is the same as Figure D.4 but with Sentinel-2 data. We focus on this area because we know from Figure D.1 that it should be the most critical.

Figure D.5 shows that there are a few dark features (that seem not to be SGL) on the image included in the polygon delimited by the `MSK`=99% contour line. Therefore, we should conclude that the MAR mask is not totally up to date (at least in this area). Nevertheless, let us directly dampen this statement, it is totally normal since MAR aims at modelling the SMB at the GrIS scale. For this purpose, it is not worth representing the ice-sheet boundary at the meter scale. But it introduces a bias in our identification of SGL as rock is more likely to be confused with lakes by the regression tree used to classify LANDSAT 8 data.

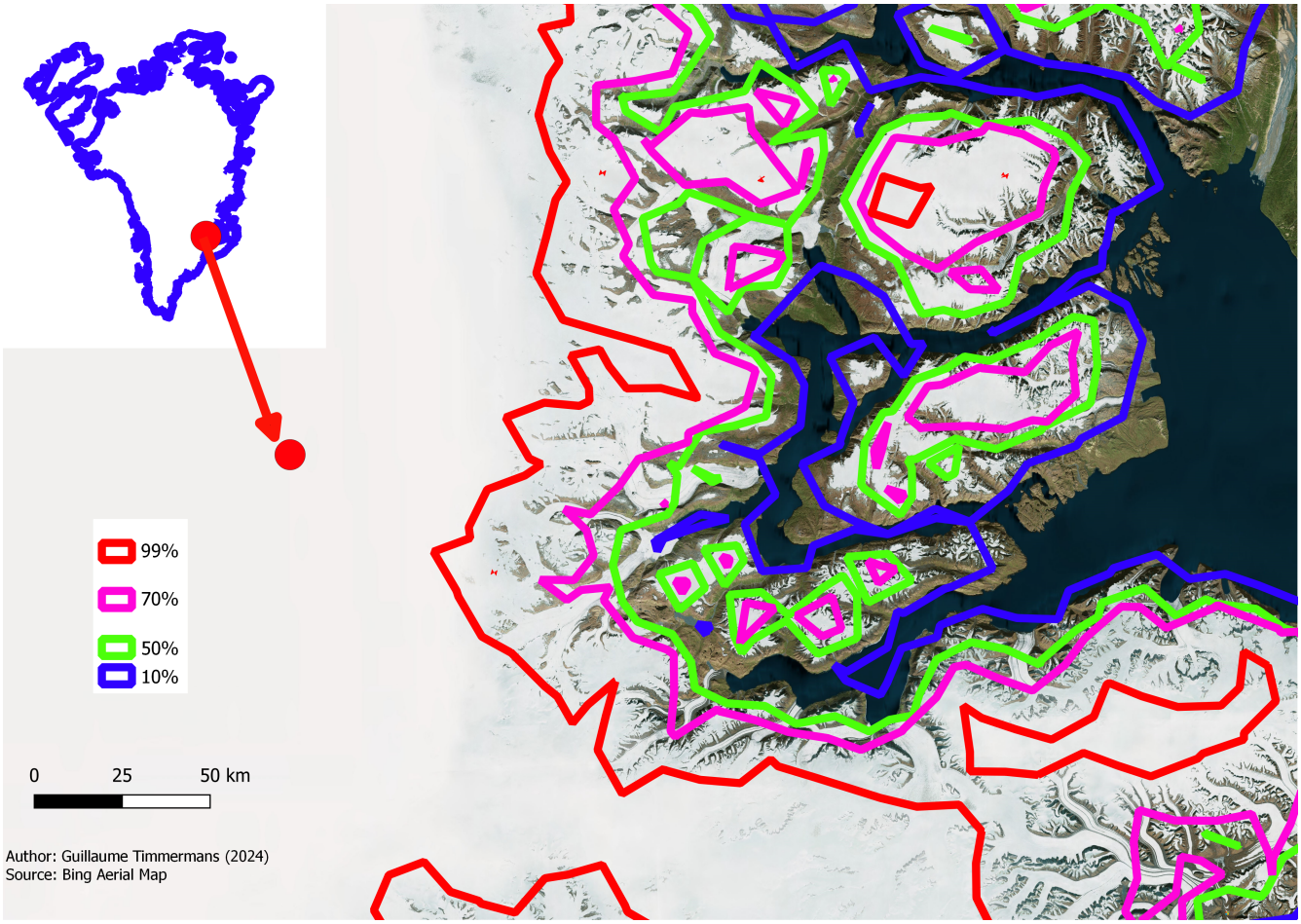


Figure D.4: Comparison of MAR mask contour lines to Bing Aerial Maps in Eastern Greenland with QGIS. The red dot enables identifying the location of the area of interest.

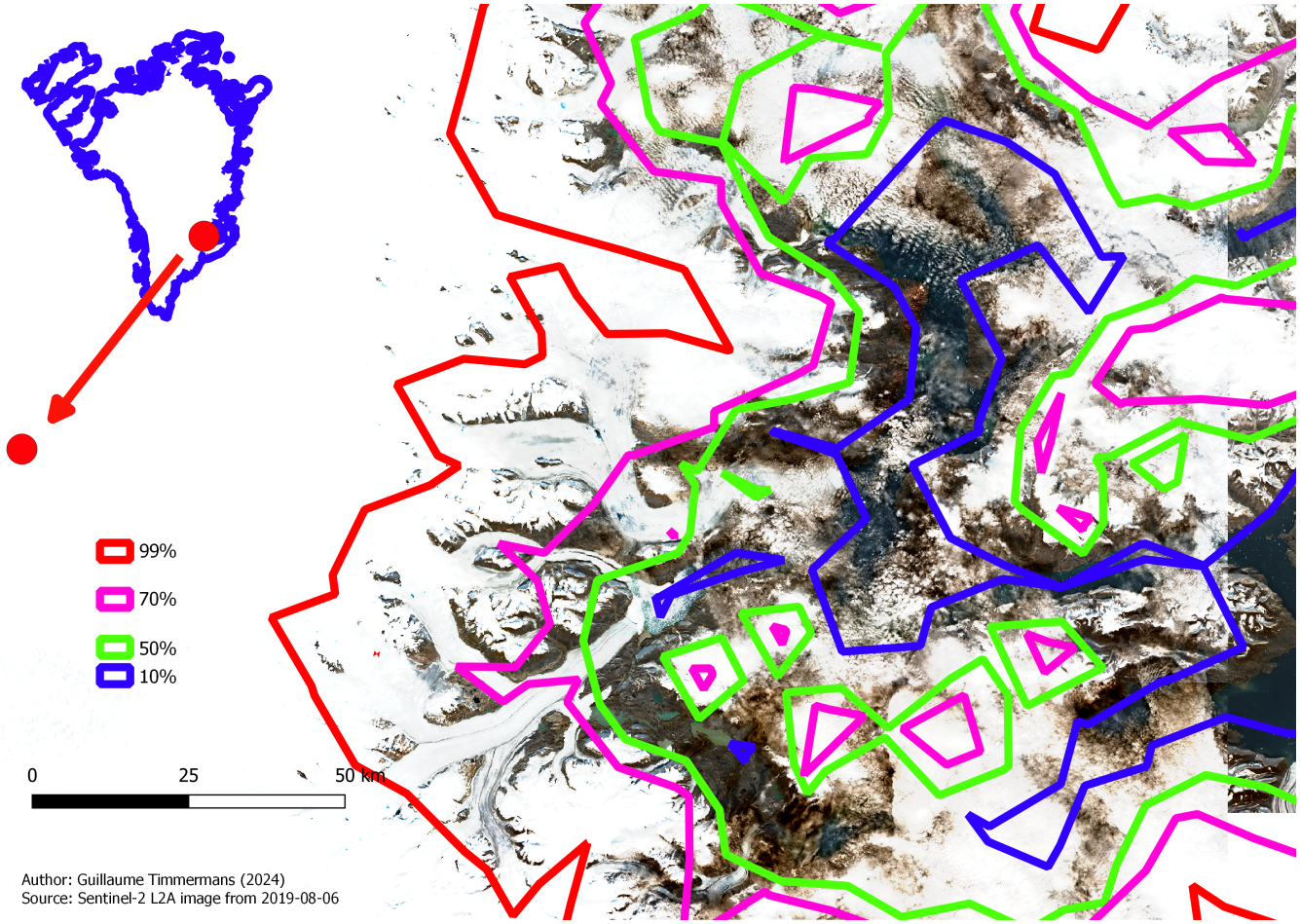


Figure D.5: Comparison of MAR mask contour lines to Sentinel-2 image from August 6, 2019 in Eastern Greenland with QGIS. The red dot enables identifying the location of the area of interest.

Appendix E

Additional Information for the Reference Simulation Assessment

Figures E.1 and E.2 compare the "reference" simulation (at a resolution of 10km×10km) over the whole GrIS and our simulation over the South-West of Greenland (with a "tuned" parameter).

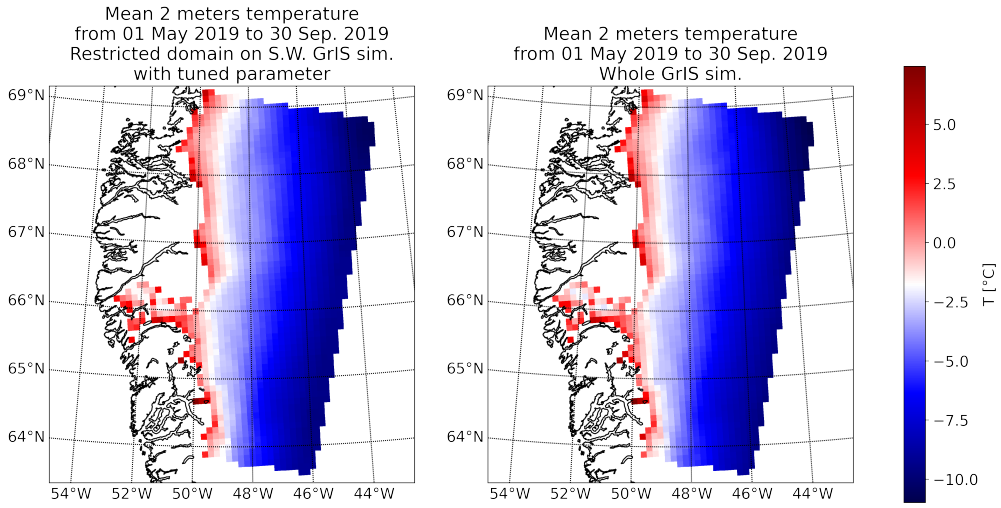


Figure E.1: Mean 2 metres temperature between May and September for our simulation (right), and the reference simulation (left).

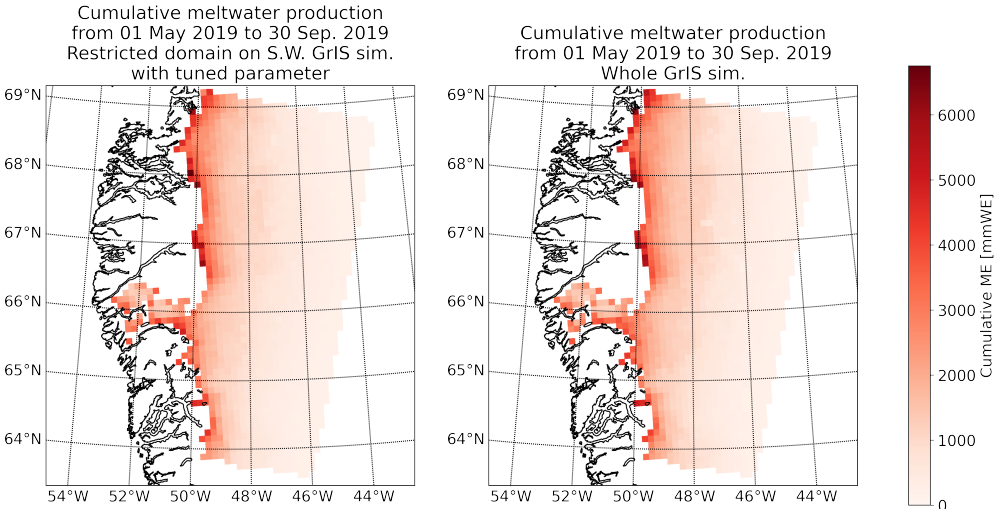


Figure E.2: Cumulated meltwater production between May and September for our simulation (right), and the reference simulation (left).

Appendix F

Multi-Threading Interpolation in Python

Listing F.1: Efficient Python code to interpolate ERA-5 data

```
import logging
import threading
import time
import numpy as np
import matplotlib.pyplot as plt
from netCDF4 import Dataset
def distance(lat,lon,lat1,lon1):
    lat,lon,lat1,lon1 = np.pi/180.*lat,np.pi/180.*lon,np.pi/180.*lat1,np.pi/180.*lon1
    a = np.sin((lat1-lat)*0.5)**2 + np.cos(lat)*np.cos(lat1)*(np.sin(0.5*(lon1-lon))**2)
    d = 2.*6378.*np.arctan(np.sqrt(a/(1-a)))
    return d
def lookforclosest(inf,sup, interpolated_array):
    data01 = Dataset('../influenceparametrezuo/assembled_c01.nc')
    lon_mar, lat_mar, tmar = [np.array(data01.variables[x]) for x in ['LON', 'LAT', 'TTZ']]
    tmar=np.mean(tmar[:,0,:,:],axis=0)
    data = Dataset('./file.nc')
    lon_era = np.array(data.variables['longitude'])
    lat_era = np.array(data.variables['latitude'])
    lon_era, lat_era = np.meshgrid(lon_era, lat_era)
    tera = np.array(data.variables['t2m'])
    miniref = distance(0.,0.,50.,50.)
    for x in range(inf,sup):
        print("X--->",x)
        for y in range(np.shape(lon_era)[1]):
            #print("Y--->",y)
            usedi, usedj = np.ones(4, dtype=int)*-999,np.ones(4, dtype=int)*-999
            valeurs,distances= np.ones(4, dtype=float)*-999,np.ones(4, dtype=float)*-999
            for u in range(4):
                #print(u)
                mini = miniref
                for i in range(np.shape(lon_mar)[0]):
                    for j in range(np.shape(lon_mar)[1]):
                        dd = distance(lat_mar[i,j],lon_mar[i,j],lat_era[x,y],lon_era[x,y])
                        if (dd<mini):
                            for g,v in enumerate(usedi):
                                used = False
                                if (i==v)&(j==usedj[g]):
```

```
                                    used = True
                                    break
                            if (used==False):
                                usedi[u]=i
                                usedj[u]=j
                                valeurs[u]=tmar[i,j]
                                mini = dd
                                distances[u] = dd
                    interpolated_array[x,y]=0.
                    somme = 0.
                    for u in range(4):
                        somme += 1./(distances[u]**2)
                        interpolated_array[x,y]+=valeurs[u]*1./(distances[u]**2)
                    interpolated_array[x,y]/=somme
if __name__ == "__main__":
    data = Dataset('./file.nc')
    interpolated_array = np.zeros_like(np.array(data.variables['t2m'])[0,:,:])
    print(np.shape(interpolated_array))
    xx = threading.Thread(target=lookforclosest,args=(0,7,interpolated_array))
    yy = threading.Thread(target=lookforclosest,args=(7,14,interpolated_array))
    zz = threading.Thread(target=lookforclosest,args=(14,21,interpolated_array))
    #aa = threading.Thread(target=lookforclosest,args=(39,41,interpolated_array))
    xx.start()
    yy.start()
    zz.start()
    #aa.start()
    xx.join()
    yy.join()
    zz.join()
    #aa.join()
    np.savetxt("./testinterpolpython.dat",interpolated_array)
    plt.imshow(interpolated_array)
    plt.colorbar()
    plt.show()
    test = np.loadtxt("./testinterpolpython.dat")
    plt.imshow(test)
    plt.colorbar()
    plt.show()
```

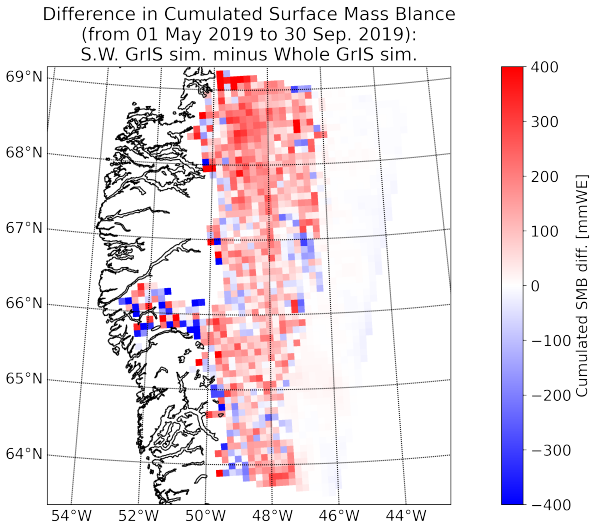
Appendix G

Assessment of the Reference Simulation without the "Tuned Parameter" and Comparison with the Other

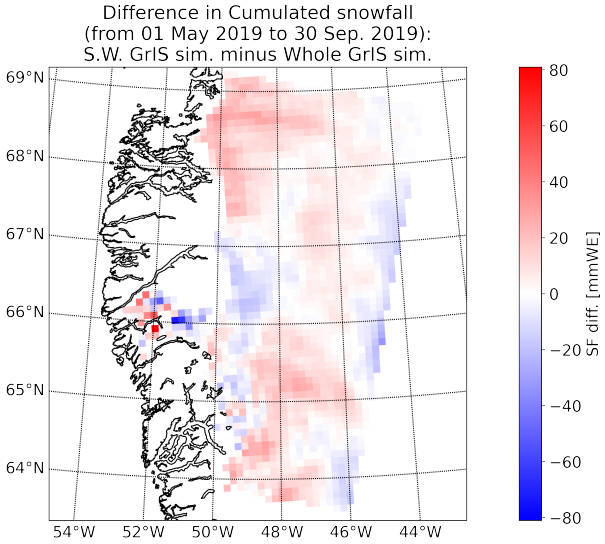
Figure G.1 shows the anomalies associated with the simulation where we did not modify the "magic" humidity parameter at the domain boundaries.

The results do not change much with the correction, but they are a little better, as shown in Figure G.2. Indeed, on average[1], the absolute anomaly difference is positive, which means that the absolute SMB anomaly of the normal simulation is greater than the absolute SMB anomaly of the simulation with a tuned humidity parameter. A Figure similar to Figure G.1 for this latter simulation can be found in Appendix G.

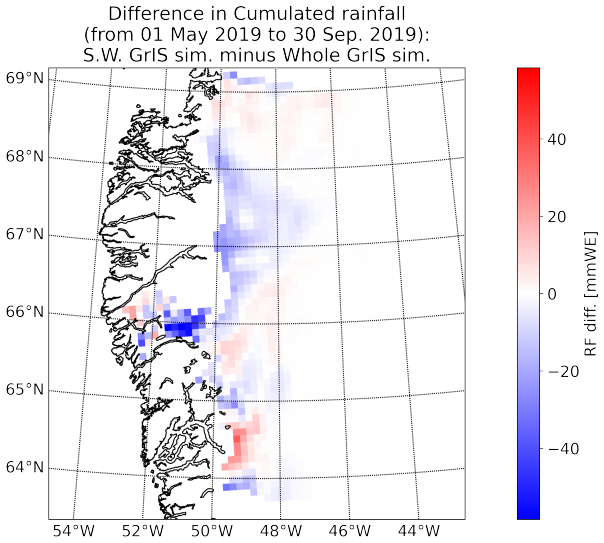
[1] As shown in Figure G.2 (b), the median is close to zero, and only some pixels make the difference.



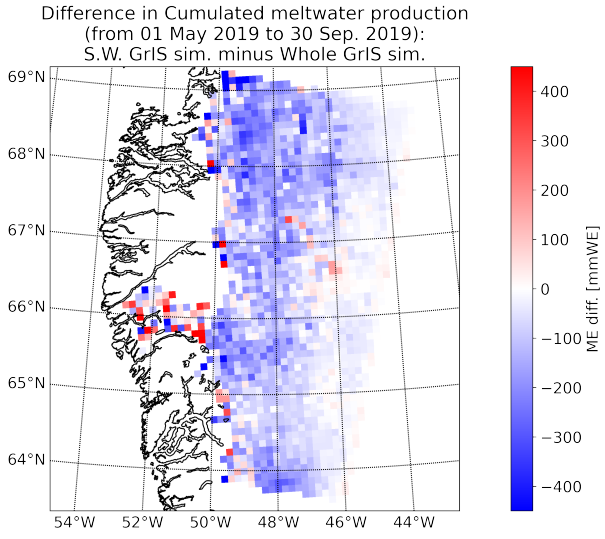
(a) Difference in the cumulated SMB over five months: our simulation minus the reference simulation.



(b) Difference in the cumulated SF over five months: our simulation minus the reference simulation.

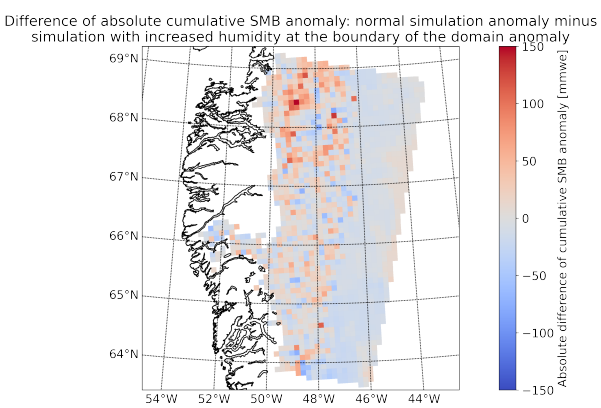


(c) Difference in the cumulated RF over five months: our simulation minus the reference simulation.

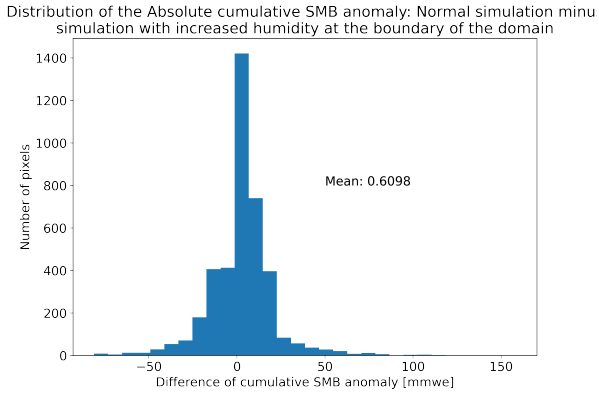


(d) Difference in meltwater production over five months: our simulation minus the reference simulation.

Figure G.1: Difference between our simulation and the reference one for four variables linked to the SMB. All differences are expressed in mmwe.



(a) Map of the absolute anomaly difference



(b) Distibution of the absolute anomaly difference.

Figure G.2: Difference in SMB anomaly between the two simulations on the South-West Greenland domain.

Appendix H

Detection of a Small Problem in MAR

Let us compare the simulations 01 and 03 (*cf.* Table 4.1). As shown in Figure 4.15, the third simulation models a significant SWH over ice sheet pixels at the end of summer. Therefore, we had to extend the simulations to the start of the next summer, so that the residual water could freeze or flow. This is represented in Figure H.2 and in Figure H.1, which represents the same data, with a different point of view. If we had stopped the simulation in September, the reference simulation (*i.e.* simulation 01) would have had a smaller SMB than the third, meaning a larger mass loss. But by extending it in time, the opposite happens, even if it was expected that surface water would freeze during the winter (instead of leaving the ice sheet through runoff). Note, however, that the difference between the two is quite small. Moreover, if the domain corresponded to the entire GrIS, it is not certain that the reference simulation has a smaller SMB than the third simulation. Indeed, we observe in Figure H.3 that the difference is positive (which means that the increase of t^* decreases the SMB) near the edges of the ice sheet, whereas it is negative towards the inland.

"Mistake" Detection in the Code Nevertheless, the refreezing effect should be more important, and the SMB should be greater for a simulation with a t^* so large that there is water over the ice at the end of summer that can freeze.

To detect the small problem in the MAR that we will discuss in the following, we analysed the spatiotemporal differences between the two simulations (*i.e.* the one with a large t^* (simulation 03), and the reference). This was done using graphical representations like the one in Figure H.4, where we notably notice that the SMB fluxes are smoother for the third simulation, because of the larger t^*. We see no great variations between the two simulations of RF nor SF in Figure H.4 (b) & (c). Figures H.5 & H.9 also contributed to help us understand that something was wrong with the model. We thought the surface water was running before the temperatures became cold enough to freeze it, even with a very large t^* which was why the increase in t^* did not increase the SMB. Figure H.6 could suggest this because we see that indeed, in the transition 2019-2020, when temperatures pass significantly below zero °C, for pixels with unexpected results (i.e., those where the increase of t^* decreases the SMB), there is little surficial water left to freeze. To find out, we launched a simulation where the runoff was zero (*i.e.*, $t^* \to \infty$).

Figure H.7 shows that for the simulation where we prevent the runoff, the surficial water height never decreases. This taught us that there was a problem in the MAR's SISVAT module, which notably deals with surface hydrology. So we analyzed the conditions necessary to freeze surface water in the code, over a typical day where we would expect it to happen. As explained in Section 4.1.1, SISVAT uses a vertically discretized snow/ice representation (*i.e.* a layered model). For all layers, when there is energy available, it is used to freeze or to melt (depending on the sign) the interstitial water (*i.e.*, stored in the pores of the snow). What is not said in Eq. (4.1) is that in principle, when there is a negative amount of energy, and when the uppermost snow or ice layer is devoid of interstitial water, the surficial water (if there is surficial water) should freeze. This case was planned in the model, but there were inconsistencies in the code.

It was written (in the Fortran source code of the MAR) that the negative heat should contribute to freezing the surficial water when the following conditions were met:

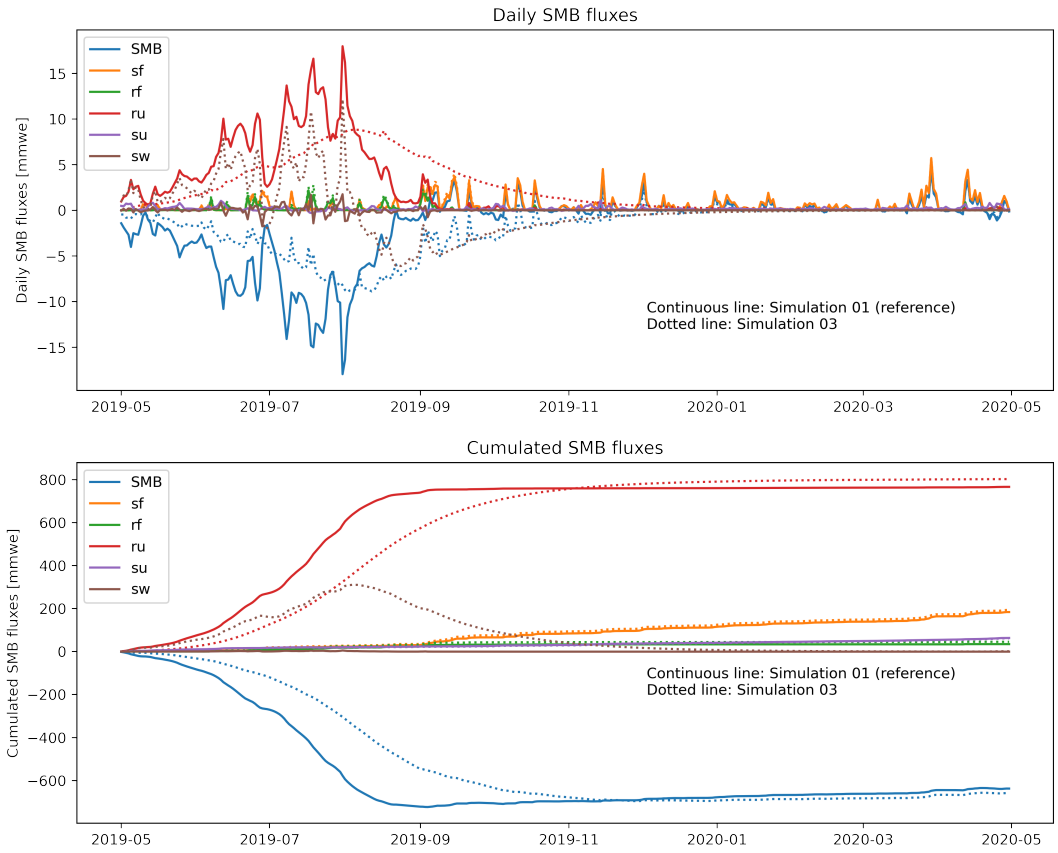
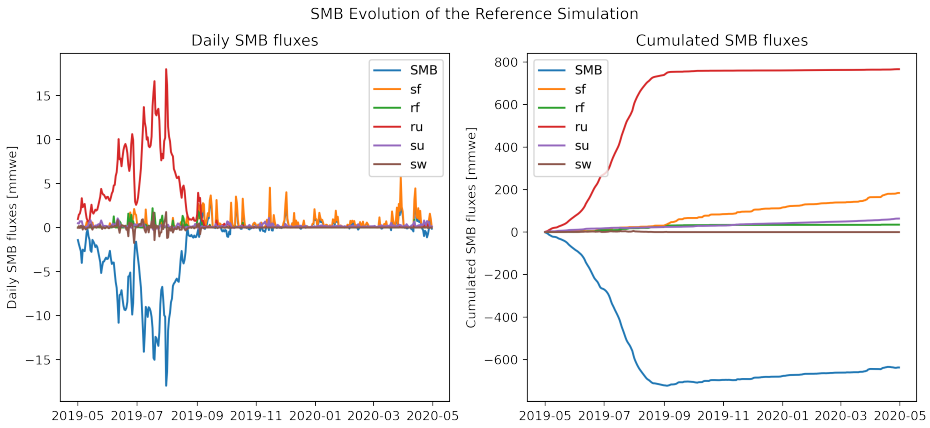
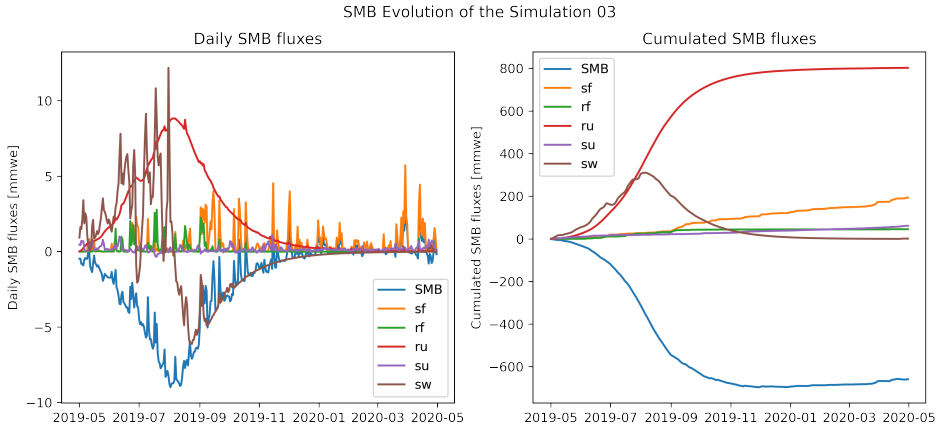


Figure H.1: Evolution of the SMB fluxes for two simulations with a different t^* parameter.



(a) SMB evolution of the reference simulation.



(b) SMB evolution of the simulation 03.

Figure H.2: Illustration of the influence of t^* on the SMB. Note that it was necessary to extend the simulations in time to clear the surficial water in the third simulation. The left plots in (a) & (b) represent the time derivatives of the right plots.

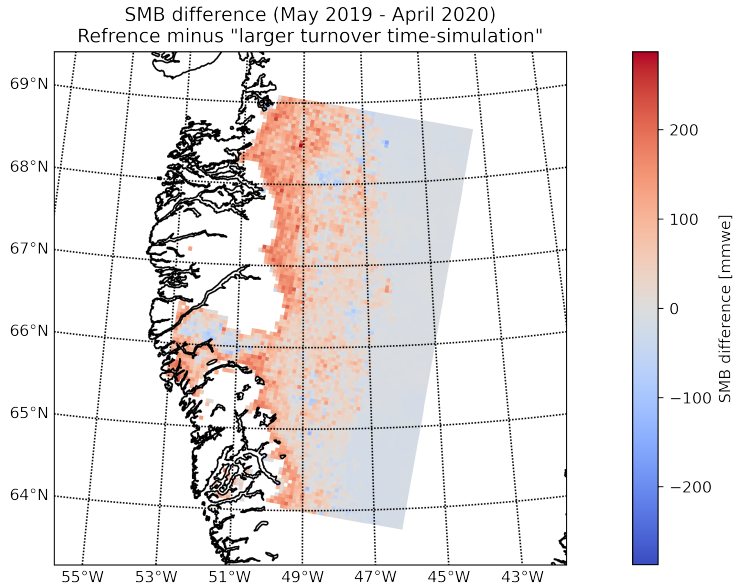
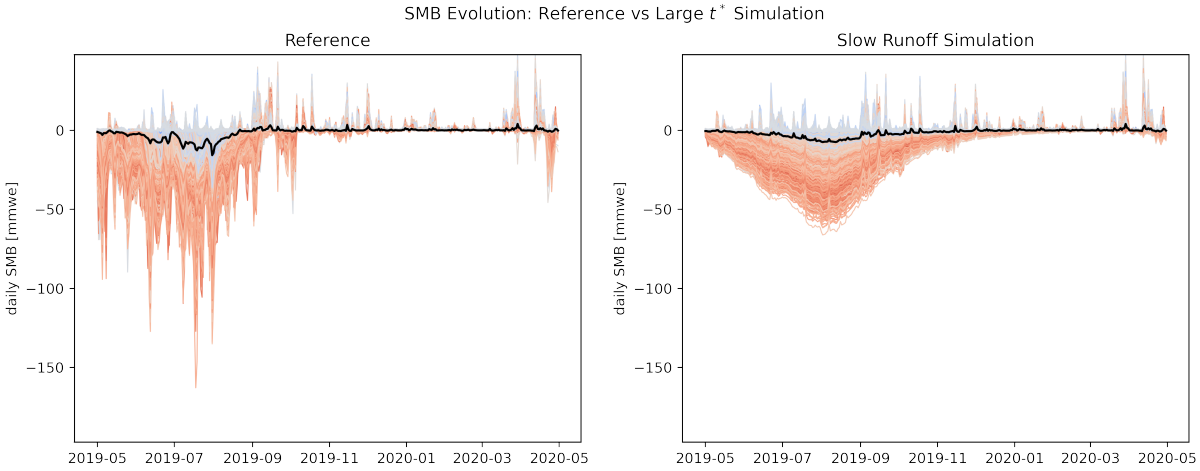
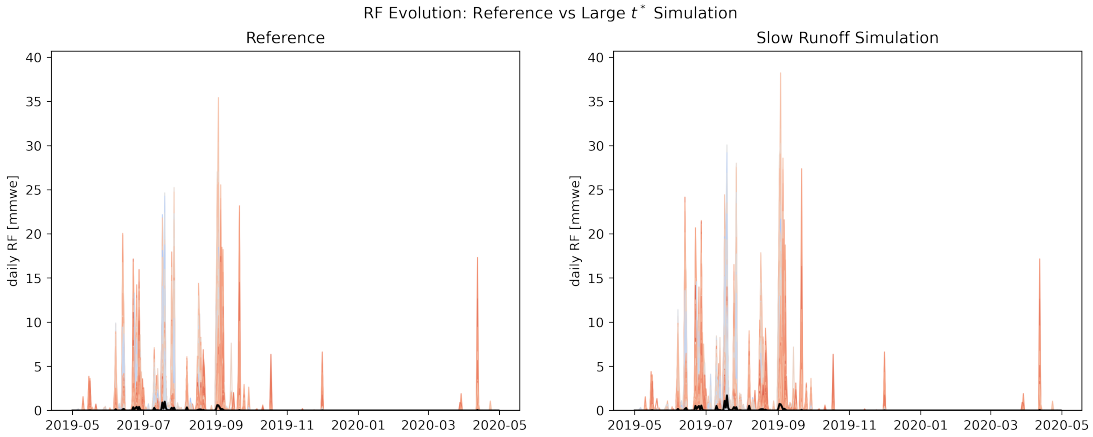


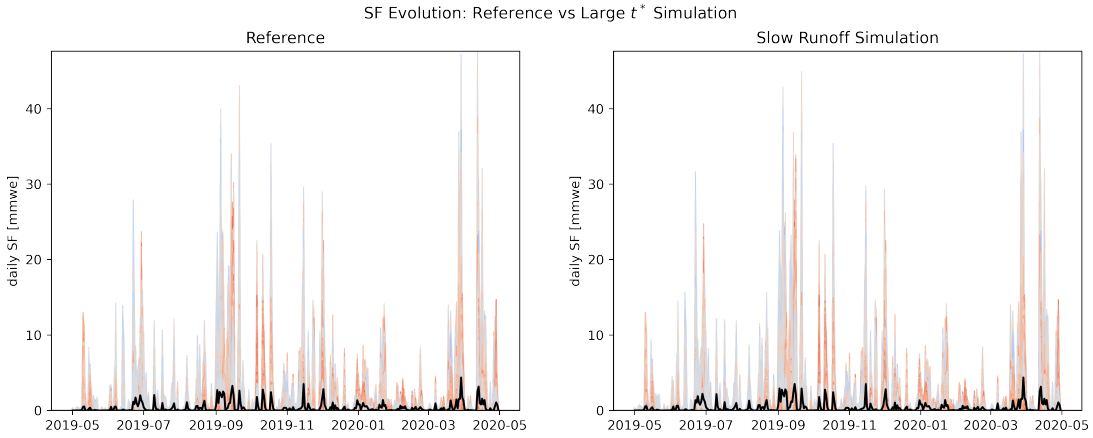
Figure H.3: Difference in SMB: reference minus simulation 03 (*i.e.* the one with a larger t^*).



(a) Comparison of the simulation 03 (with a large t^*) to the reference in terms of SMB.

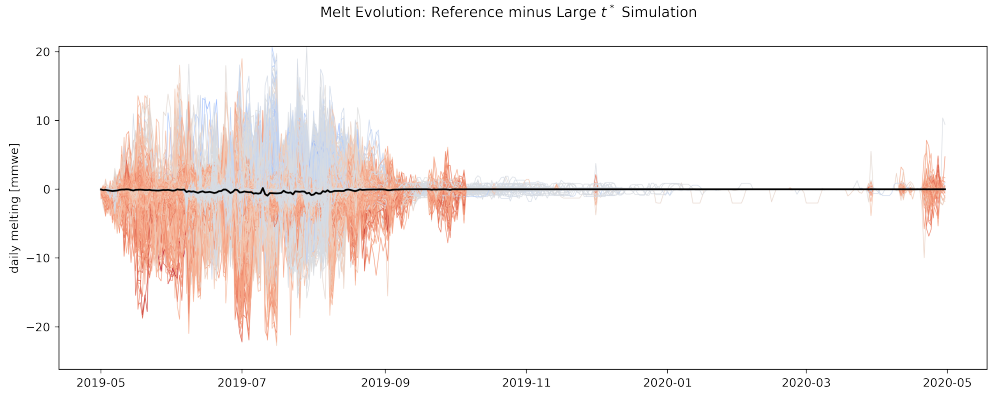


(b) Comparison of the simulation 03 (with a large t^*) to the reference in terms of rainfall.

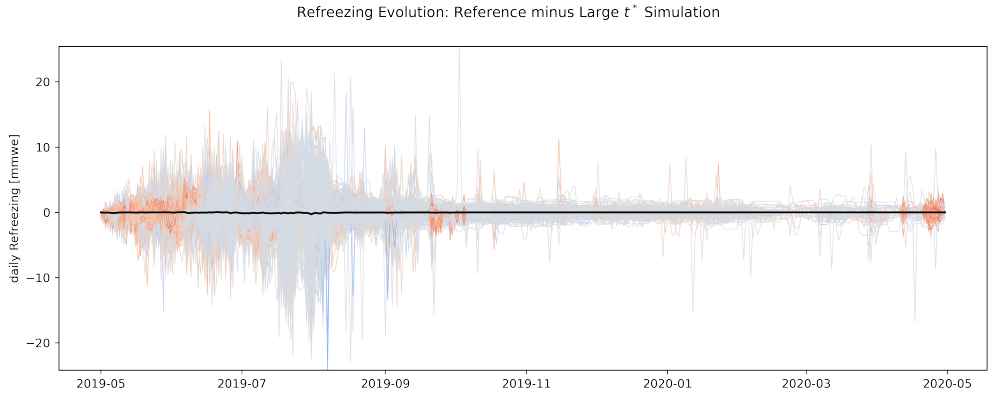


(c) Comparison of the simulation 03 (with a large t^*) to the reference in terms of snowfall.

Figure H.4: Comparison of the daily SMB (and fluxes) for the reference simulation and the one with a larger t^*. Each line corresponds to the evolution of a pixel. The colour of the lines corresponds to the colour of the pixel in Figure H.3, hence the difference of SMB between the two simulations (reference - large t^*).



(a) Difference of melting between the two simulations



(b) Difference of refreezing between the two simulations

Figure H.5: Difference between the reference simulation and the one with a larger t^* for the daily melting/refreezing. Each line corresponds to the evolution of a pixel. The colour of the lines corresponds to the colour of the pixel in Figure H.3, hence the difference of SMB between the two simulations (reference - large t^*).

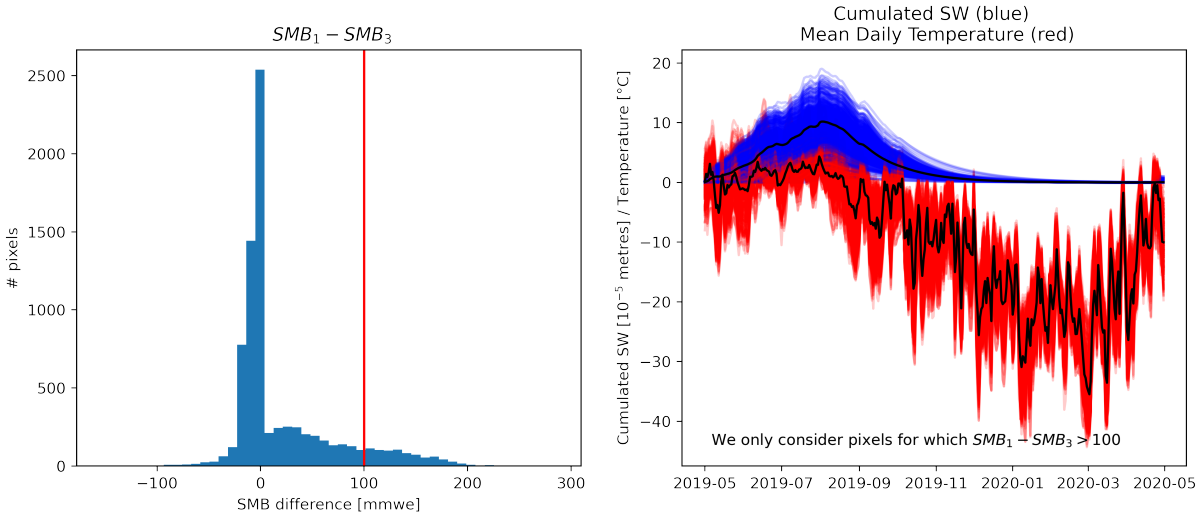
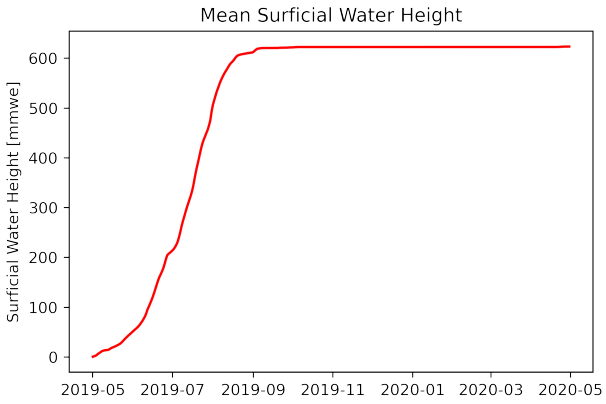
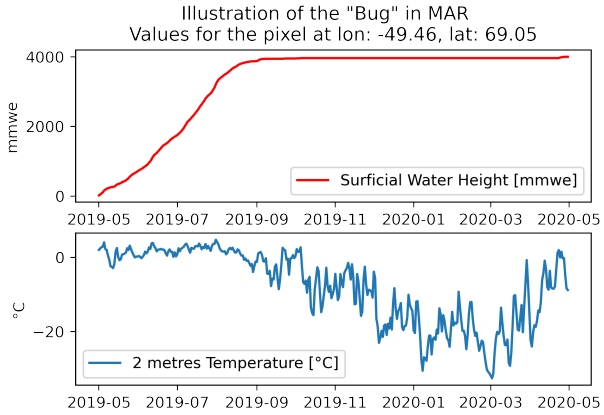


Figure H.6: Comparison of the cumulative SW flux (*i.e.*, the surficial water height), and the temperature of the third simulation (with a large t^*) for the pixels where the difference of SMB between the reference simulation and the simulation with a larger t^* is greater than 100 mmwe.



(a) Mean (over all pixels) surface water height for the no-runoff simulation.

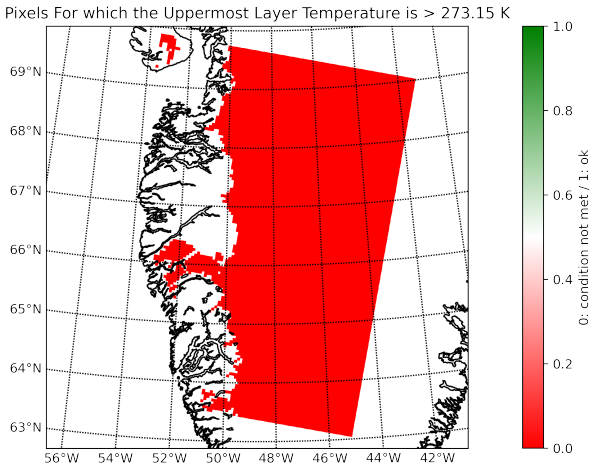


(b) Evolution of the surface water height, and the 2 metres temperature over one year for a "representative" pixel.

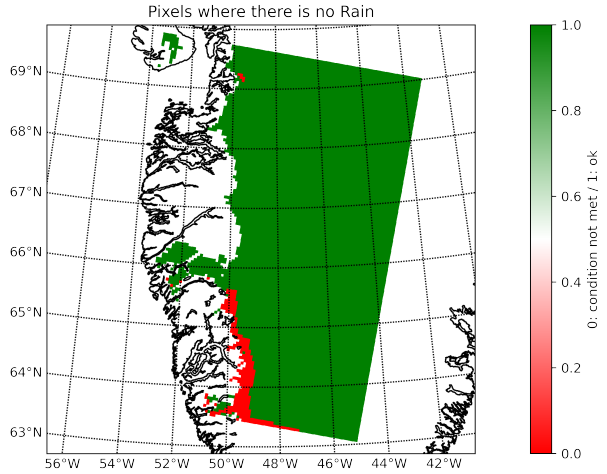
Figure H.7: Illustration of the bug in the MAR: the surficial water never freezes.

1. The uppermost layer temperature should be greater than the snow melting temperature (in addition to a sign error, it is not possible, since the layer temperature is blocked at 273.15 K until the whole layer has frozen or melted)
2. There should be no rain (otherwise it is the rain that freezes)
3. The layer density must be greater than 830 kg/m^3 (so that it only works over the ice, *i.e.* in the ablation zone).
4. There must be surficial water.

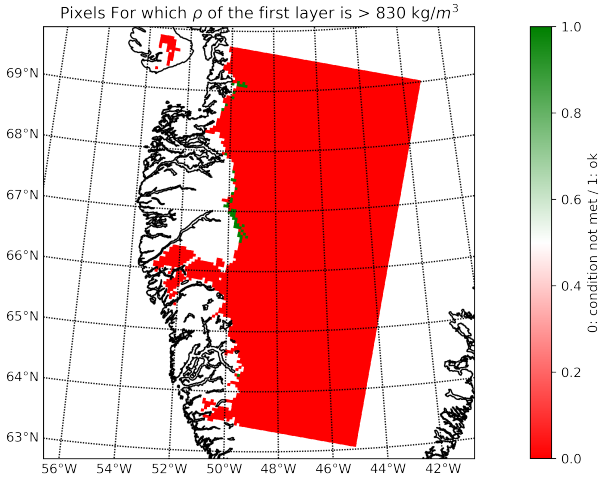
We re-launched the "no runoff simulation" on the first of December 2019 (because in winter, the surficial water should freeze, of course), to evaluate where and when the conditions were met. The results are presented in Figure H.8. The four conditions were never met during the whole simulation.



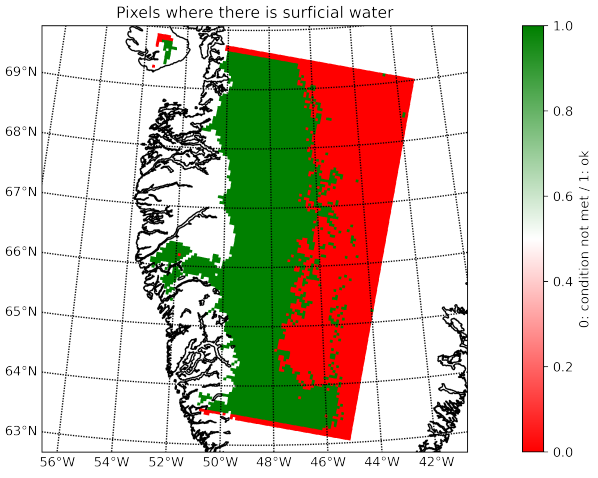
(a) Illustration of the first condition on 12/01/2019.



(b) Illustration of the second condition on 12/01/2019.



(c) Illustration of the third condition on 12/01/2019.



(d) Illustration of the fourth condition on 12/01/2019.

Figure H.8: Illustration of the four conditions to be met to freeze the surficial water, if there is an excess of negative available in the first snow layer. Simulation on the first of December 2019 at midnight. The temperature is negative in almost the whole MAR domain.

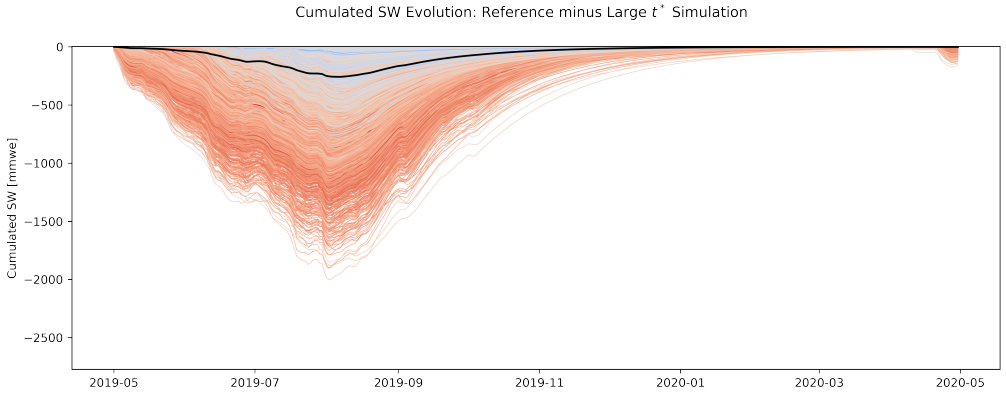


Figure H.9: Difference between the reference simulation and the one with a larger t^* for the surficial water height (*i.e.* the accumulated SW flux). Each line corresponds to the evolution of a pixel. The colour of the lines corresponds to the colour of the pixel in Figure H.3, hence the difference of SMB between the two simulations (reference - large t^*).

Appendix I

Latent Heat Flux (LHF) Computation in SISVAT: Details of the Code

In SISVAT, for each pixel, the LHF is computed as follows (Listing I.1):

Listing I.1: Computation of the Latent Heat Flux in SISVAT

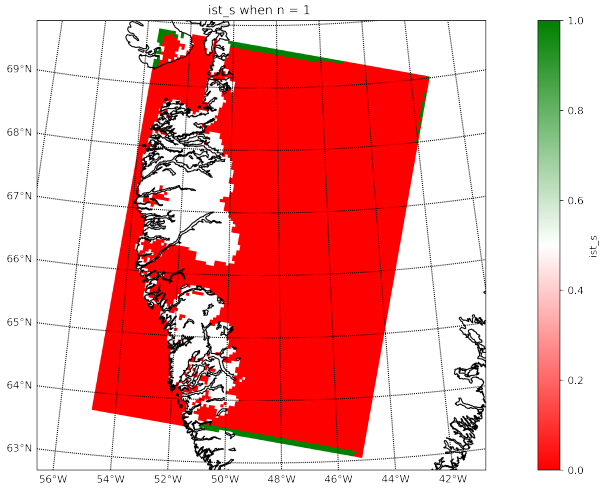
```
HL___D(ikl) =  &
(ist__s * ro_Wat * dz_dSV(0) * (etaNEW(ikl) - etaBAK(ikl)) / dt__SV &
+ ist__w * f_HSHL(ikl) * (QaT_SV(ikl) - qsatsg(ikl))) * Lx_H2O(ikl)
```

The variables of Listing I.1 are explained in the following:

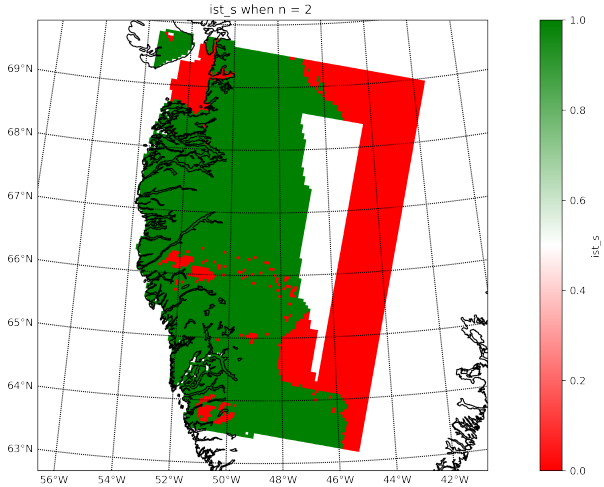
1. `ist__s`: Soil (tundra) marker
2. `ro_Wat * dz_dSV(0) * (etaNEW(ikl) - etaBAK(ikl)) / dt__SV`: only used over the tundra (*cf.* text)
3. `ist__w`: Water and ice marker
4. `f_HSHL`: Factor common to the latent and to the sensible heat fluxes
5. `QaT_SV`: SBL Top Specific Humidity
6. `qsatsg`: Soil Saturation Specific Humidity
7. `Lx_H2O(ikl)`: Latent Heat of Vaporization/Sublimation (depends on the proportions of ice and of water)

The method explained in Appendix J allows to understand that `ist__s` is zero for ice and water pixels, whereas `ist__w` is equal to one for these. These parameters are mapped in Figure I.1.

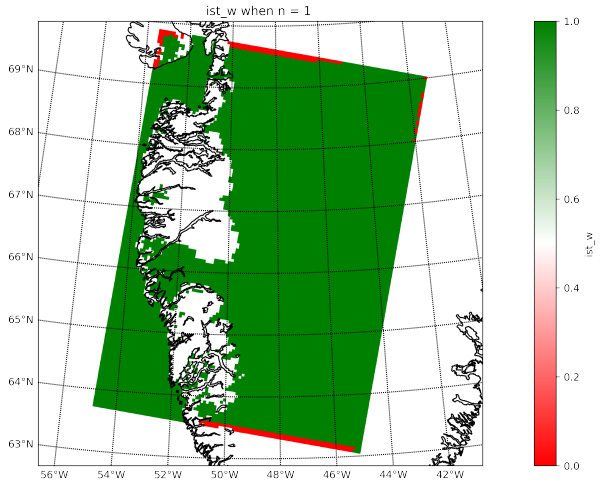
Knowing that `ist__w` is an ice & water marker, we should only focus on the corresponding part of the code presented in Listing I.1. It tells us that the model considers that the LHF is proportional to the difference between the actual specific humidity, and the saturation one, denoted q_{sat} and computed as in Eq. (4.9). Also, it is proportional to the latent heat of vaporisation, $L_{x,\mathrm{H_2O}}$, computed as in Eq. (4.8). These two parameters are influenced by the presence of water. Indeed, if there is a SGL, the computation of $L_{x,\mathrm{H_2O}}$ must be the same as for "oceanic pixels". But also, if there is water, over the ice sheet, the model automatically considers that its temperature is 273.15 K. Nevertheless, even if very few measurements are available, it is very likely that the temperature of the SGL water is greater than zero (°C).



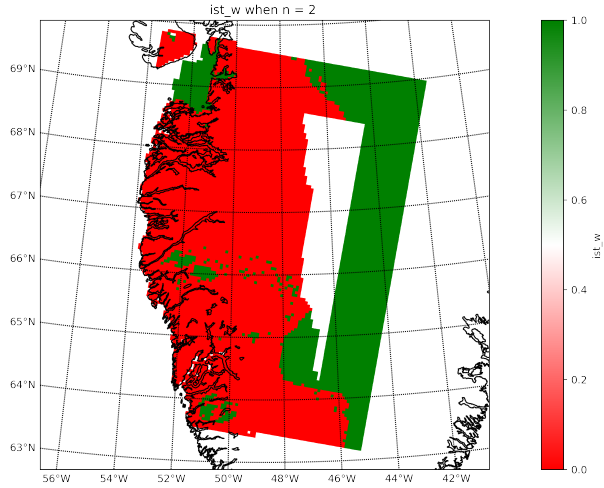
(a) ist s for n = 1



(b) ist s for n = 2



(c) ist w for n = 1



(d) ist w for n = 2

Figure I.1: ist s & ist w for different calls of SISVAT .

Appendix J

Programs Written to Help Understand the MAR Code

In order to see at a given time step which value is taken by a given variable for each MAR pixel, it is necessary to write it in a file. This can be done as follows (Listing J.1):

Listing J.1: Writing the values of a variable in a file

```
open(unit=668, file='/path.../guiview21apr_1_.txt',status='old', access = 'append')
write(668,'(f16.11)') guillaumeview(:,:,1)
write(668,*), '++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++'
close(668)
```

In order to see the evolution of a variable further in the simulation, a condition depending on the number of time steps done since the beginning of the run can be included, allowing to reduce the memory consumption by the operation of Listing J.1. This is written in Listing J.2. Using 60 as a modulo time step leads to 48 outputs per 24 hours if the time step is 30 seconds.

Listing J.2: Condition to reduce the memory consumption

```
if (mod(iterun, 60) .eq. 0) then
 *** previous code ***
 endif
```

The array `guillaumeview(:,:,1)` must be (declared and) filled with the variable that we want to observe, each element corresponds to a pixel, (*i.e.*, a call of `sisvat.f90`). This can be done as follows (Listing J.3), using an intermediate variable, `guillaume`, that must be declared as common into a module used by each code relevant for our work into MAR.

Listing J.3: Filling the array

```
(...) * code of phy_sisvat.f90 *
call SISVAT(1)
if (n .eq. 1) then!first level of sub-pixel
guillaumeview(i,j,1) = guillaume
endif
(...) * code of phy_sisvat.f90 *
```

Once the code is compiled and the MAR launched for several time steps, it can be stopped (using simply `ctrl + C`), then we can launch the following bash script (Listing J.4).

Listing J.4: Data managing to map the variables

```
#!/bin/bash
```

```
cd results_sim/
rm -f *txt
rm -f *gif
cd ..

python3 traitement_sortie.py
cd results_sim
for i in *".txt" ; do
echo ${i}


ferret <<EOF
use ICE.20190701.c03.nc
define grid/x=lat/y=lat gg
file/var=swf/col=1/grid=gg ${i}
shade swf
fill/title = ${i}
frame/file = ${i}.gif
quit
EOF
#ignore0(swf)
done
```

The script of Listing J.4 depends on the following Python program (Listing J.5) :

Listing J.5: Python script organizing what was written by the fortran program (phy_sisvat)

```
import numpy as np
numberoffiles = 0
for g in [1, 2, 3]:
    with open('./guiview21apr_'+str(g)+'_.txt') as file:
        lignes = file.readlines()
        number = 0
        values = []
        noflines = 0
        for i in lignes:
            mot = i.split()
            if mot[0][0] != '+':
                if mot[0][0]!='*':
                    values.append(float(mot[0]))
                else:
                    values.append(-999.)
                noflines+=1
            #print(mot, mot[0], mot[0][0])
            else:
                #print(noflines)
                noflines=0
                number +=1
                valeurs = np.array(values)
                values = []
                if (g==1) & (numberoffiles%25 ==0):
                    np.savetxt('./results_sim/'+str(g)+'_'+str(number)+'_21apr.txt',
                               valeurs)
                    print('un fichier')
                numberoffiles +=1
```

An example of a result that was obtained using the bash script of Listing J.4 is given in Figure J.1. For readers accustomed to the MAR code, it represents the variable `f_HSHL` of `sisvat.f90`, after 26 time steps on 07/01/2019.

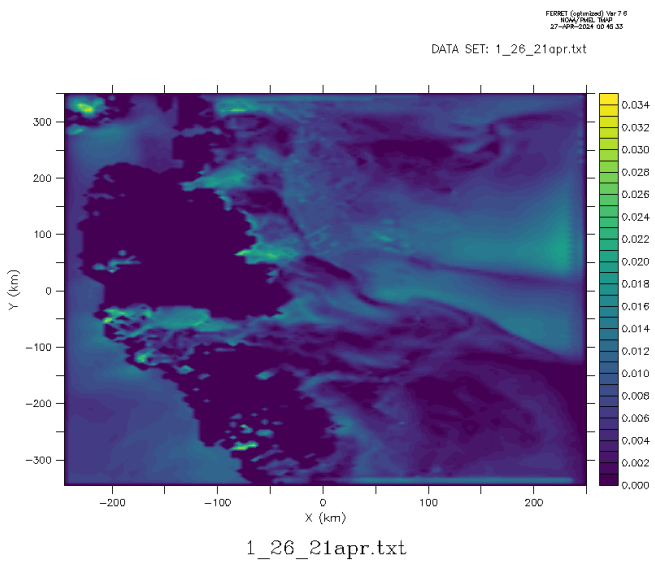


Figure J.1: Example of a variable mapped thanks to the script of Listing J.4.

Layered Output If one wants to see what are the values taken by a variable that has a vertical dimension, some additions allow to do this. In `phy_sisvat.f90`, one should add what is shown in Listing J.6. The processing of Listing J.5 remains valid, and to represent graphically the variable, the code of Listing J.7 can be used. It enables to obtain a plot such as the one presented in Figure J.2.

Listing J.6: Filling the array on several vertical layers.

```
(...) * code of phy_sisvat.f90 *
call SISVAT(1)
     if (n .eq. 1) then
         guillaume_it2 = 1
         do guillaume_it =-nsol,isnosv(1),+1
              guillaume_clayer(i,j,guillaume_it2) = eta_sv(ikl,guillaume_it)
              guillaume_it2 = guillaume_it2+1
        enddo
    endif
(...) * code of phy_sisvat.f90 *
```

Listing J.7: Python script plotting value of a variable as a function of the layer index.

```
import numpy as np
import matplotlib.pyplot as plt
f, ax = plt.subplots(figsize = (5,8))
cmap = plt.cm.viridis
for i in range(1,25):
    datao = np.loadtxt('path/gui_clayer_clean_'+str(i)+'.txt')
    datao = np.reshape(datao, (21,np.shape(lon_mar)[0], np.shape(lon_mar)[1]) )
    color = cmap(i / 24)
    plt.plot(np.mean(np.mean(np.ma.masked_where(sh21<0.1,
                np.ma.masked_where(firstcall21==0,datao[:,:,:])),
                axis = 1), axis =1), label = str(i), color = color)

plt.title("Reference")
plt.legend(title = '#-hours')
```

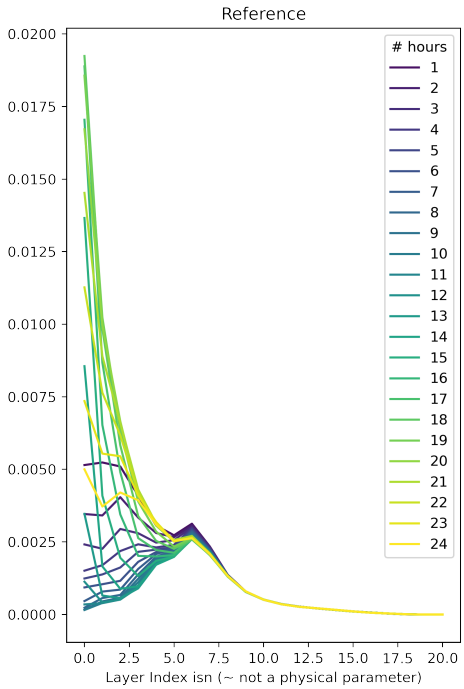


Figure J.2: Plot made via the script of Listing J.7.

```
plt.xlabel("Layer Index isn (~ not a physical parameter)")
plt.ylabel("Mean eta_sv exluding pixels out of the ice sheet")
plt.savefig("eta_sv_layer_index_ref.png", dpi = 350)
```

Remark on the method utility Of course, we could have used `outice.f90`, but this method is easier to use, faster, and more reliable, as we can directly access the variable that we want in the code, without the need to use a variable common to the module on which we work (*e.g.* `sisvat.f90`) and `outice.f90`. From a personal point of view, as I never used Fortran before this thesis, it was easier to rely on Pyhton than on Fortran, because it is a programming language that is more adapted to simple text management, as done in Listing J.5.

J.1 Examples of Application

As an example, our programs have identified the pixels for which the SISVAT model is called at a specific time step, and for which land cover. Indeed, we explain in Section 4.1.1 that the surface of MAR pixels can be modelled as a "mix" of two types of land cover. Thus, a pixel can, for instance, consist of 20% tundra and 80% ice sheet, *etc.* This is something we already knew, but it has been illustrated precisely and satisfactorily for a user beginning to work with the MAR model. Thus enabling the user to strengthen his understanding of the code.

Another example of our program's application is that it has allowed us to verify if what we explain in Section 4.2.2 is correct. Indeed, the expression of the Equation 4.7 in Fortran is given by (Listing J.8):

Listing J.8: Expression of Eq. (4.7) in Fortran 90.

```
! (...)
albWIc = aI1dSV - (aI1dSV - aI2dSV) &
              * exp(-(rusnSV(ikl) &
                      * (1.-SWS_SV(ikl) &
                         * (1 - min(1, iabs(isn - isnoSV(ikl))))) &
                      / ru_dSV)**0.50)
! (...)
```

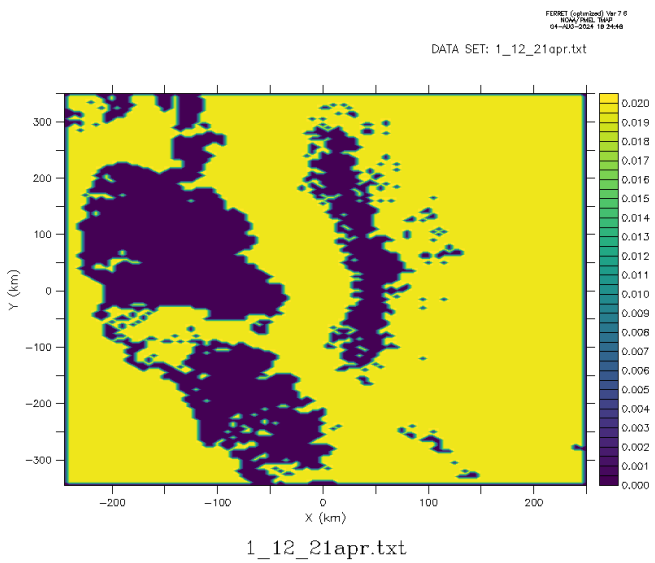


Figure J.3: Output of the program application to verify the value of an expression.

, and involves many variables, making it complicated to understand. However, we can illustrate that this piece of code:

```
(1.-SWS_SV(ikl) * (1 - min(1, iabs(isn - isnoSV(ikl))))/ ru_dSV
```

is almost always equalled 1 (if we multiply it by `ru_dSV` that equals 50) for the pixels and the period of interest. It is shown in Figure J.3.