Dynamical network models for melt ponds on sea ice

Michael Coughlan
St. Edmund Hall
University of Oxford

A thesis submitted for the degree of
Doctor of Philosophy

Trinity Term 2021

Tá an tráchtas seo tiomanta do mo thuismitheoirí, Ann agus John,
agus do mo sheanthiusmitheorí.

"On the surface it seems that the work of engineers, designers and research mathematicians consists only of polishing and refining, easing this joint and balancing that wing until there is no longer a wing joined visible to a fuselage, but a perfectly developed form freed at last from its matrix, a spontaneous and mysterious whole with unified quality of a poem. It seems that perfection is attained not when there is nothing more to add, but when there is nothing more to take away. At the climax of its evolution, the machine conceals itself entirely."

- Antoine de Saint-Exupéry, Wind, Sand and Stars

Acknowledgements

I would like to thank my three supervisors, Ian Hewitt, Sam Howison and Andrew Wells. Credit must be given to UKRI and NERC for generously funding this work, and to the Clarendon fund, St. Edmund Hall and Geoff Chatas for generously providing my stipend. I am grateful to my college, St. Edmund Hall, and the Mathematical Institute, and all there who made both places wonderful environments for work and study. I would also like to thank Martin Doble from Polar Scientific Ltd, Jeremy Wilkinson at BAS, the science team on the SODA project and all of the crew aboard the USCGC Healy in autumn 2019, for giving me several months of reality checks on my assumptions about sea ice. Finally, an unending debt of gratitude is due to my family, for supporting me to get as far as even starting this DPhil, and for supporting me during the early stages of the Covid-19 pandemic, and to my partner Marjolein Poortvliet.

Abstract

We present a model for the formation and evolution of melt ponds on sea ice, written as a dynamical system on a network. These ponds have long been suggested as a contributing factor to the discrepancy between observed and predicted sea ice extent; ponds have a lower albedo than bare ice, so they contribute to the ice-albedo feedback. Of particular interest here is the geometry and topology of the ponds as they grow and connect. It has been observed that there is a change in the fractal dimension of the pond system. Further, it has been suggested that pond systems go through a percolation process with ponds connecting across a whole floe at a phase transition, and that a percolation threshold plays a key role in pond dynamics.

We present a physically based network model for systems of ponds, which can be used to examine both their individual and collective behaviour. Each pond initially occupies a distinct catchment basin, modelled as a node; a differential equation represents the melting dynamics, and mass conservation governs the evolution of the pond volume and water level. Ponds can connect together through fluxes of water between catchment basins, represented by edges. Ponds can also drain into the sea, through another form of edge.

We use the model to explore how the evolution of pond area and hence melting depends on model parameters, and find that it depends most sensitively on surface roughness and the timing of onset of drainage from the ponds. We qualitatively reproduce the observed transition in fractal dimension as the ponds grow, and the typical life-cycle of the pond system, with the areal coverage growing, then shrinking as the ponds drain to sea level, and then slowly increasing again. We examine the importance of percolation thresholds, and in some cases, find evidence of anomalous percolation. We also use model results to predict the extent of pond coverage in the future, warming Arctic.

Contents

Chapter 1

Introduction

1.1 Sea ice and melt ponds

Much of the waters of the Arctic basin are covered by a layer of sea ice, of order metres thick. In some places, this ice persists for many years, while in others it melts completely during the summer, opening up passages through the ocean [Weeks and Hibler, 2014]. As the Arctic warms, these passages will become more extensive, allowing shipping to operate from the Atlantic to Pacific through the Arctic (the aim of the famed search for the Northwest Passage). For this reason and others, an understanding of how the Arctic will change in the coming years is of great importance to us.

The reflectivity of this ice helps to regulate the climate of the northern hemisphere, in a process known as the ice-albedo feedback [Deser et al., 2000]. This ice-albedo feedback is one of the predominant ways in which the ice helps to regulate global temperature and climate. Solar radiation incident on the surface of the ice is largely reflected back into space, while radiation incident on open ocean water is almost completely absorbed, leading to heating. Ice has an albedo (reflectivity) of approximately 0.7 (though this can vary throughout the year, especially as the coverage of melt-ponds varies) [Polashenski et al., 2012], and therefore reflects most of the sun's radiation incident on the Arctic. The ocean has an albedo of less than 0.1, and so absorbs most of the incident radiation [Hanesiak et al., 2001]. Therefore, the wider and thicker the ice cover in the Arctic, the cooler it remains, due to reflection of solar radiation. The converse is also true. The lower the extent of the ice cover, the more heat is absorbed in the system, which, at a high enough air and ocean temperatures can lead to ice loss, accelerating the absorption of energy into the system and accelerating the loss even further. The decline in ice extent can be stalled with an adequate

drop in air temperature, such as at the end of summer, which causes a crust of ice to re-form on the ocean surface, and is an inherent part of the Arctic's annual cycle.

The last two decades have seen significant increases in ice loss, due to higher atmospheric temperatures in the Arctic and other factors [Kwok and Untersteiner, 2011].[1] Melting now begins earlier in the spring than in previous decades, more ice is lost during the summer, and freeze-up occurs later in the autumn. As the ice cover shrinks, the decay rate accelerates and the Arctic loses ice more quickly. Recent ice loss has been correlated with extreme weather events in the rest of the northern hemisphere and a causal link has been suggested between them [Notz, 2012]. Several changes to climates at lower latitudes have been attributed to the shrinking extent of Arctic ice [Vihma, 2014]. These changes include: higher summer temperatures in Europe, with drier soils; changes in the East Asian monsoon; and anomalous cyclone formation in the Pacific. For these reasons, a clear understanding of the state of the Arctic ice, and how it is changing is of the utmost importance to our understanding of our current climate, and is essential to predicting and anticipating how global climate and weather patterns may change in the coming decades.

The ice-albedo feedback is complicated by the pooling of water from the melting of the top surface of the ice during the summer, which creates features known as melt ponds. Melt ponds have an albedo between that of bare ice and open water [Morassutti and Ledrew, 1996], and absorb energy into the ice surface, accelerating melting and the formation of open water in regions with a high density of ponds, such as those seen in Figure 1.1(b). Ponds form in early summer, as the snow and ice on the surface begins to melt, filling the hollows in the topography (see Fetterer and Untersteiner [1998] for example). Even though the ice is porous, as fresh melt-water trickles into the bulk ice, this water refreezes, blocking the pores and rendering the ice impermeable. This allows the ponds to form [Polashenski et al., 2017], and as the melt season progresses, they interconnect and join, and can even grow to such an extent that they flood entire floes. As the ice warms, and becomes more permeable (reversing the process of pore-blocking at the onset of melting) ponds eventually begin to drain through the ice such that their water levels settle at a height just above sea level. The pond area tends to shrink during this drainage process [Polashenski et al., 2012]. After the water levels in ponds relax to a hydrostatic equilibrium, they sometimes regain some area, and persist until the end of summer. Throughout this period, their extent is controlled by the topography of the bare ice that sits above sea level, changing due

[1] The Arctic experienced a particularly high temperature anomaly in the early part of 2018, when temperatures were 20°C higher than the same period in previous winters.

to further volume loss and hydrostatic balance. Ponds refreeze in autumn, unless the parent ice floe completely melts or breaks up [Arntsen et al., 2015]. During drainage, ponds in some areas may disappear completely and not return. In other areas, ponds might not form at all.

Hohenegger et al. [2012] looked at the area-perimeter relationships of ponds from a large number of satellite images throughout the melt season, and found that systems of ponds have very particular geometric properties. They quantified the roughness of perimeters of ponds, using the fractal dimension D, defined by

$$p \propto \sqrt{a}^{D}, \tag{1.1}$$

where p and a are the perimeters and areas of individual ponds, and found changes between a disconnected phase, where $D \approx 1$, to a highly connected phase where $D \approx 2$. They found that the transition happens at a pond area of somewhere between 10 and 10^3 m^2. They argued that this suggests the existence of a phase transition between the two states, an associated percolation process during pond evolution, and a threshold between the two states, which we refer to as the percolation transition. [Popović et al., 2020a] used a toy model of the drainage phase of pond evolution to argue that the percolation transition gives an upper bound for pond coverage during the period post drainage. This constraint alone can be used to better parametrise pond coverage than current representations of ponds in climate models [Flocco and Feltham, 2007, Flocco et al., 2010, Polashenski et al., 2012].

The work of Popović et al. [2020a] demonstrates that connectivity is an important factor that controls the extent of ponding on floes. However, they restricted their analysis to one stage in the formation of ponds. In fact, several percolation processes exist in the evolution of ponds, with different physics dominating the processes of initial flooding, drainage, and refilling in the late summer.

In order to further investigate the percolation processes involved in pond evolution, and the role of connectivity in pond behaviour, in this work we model the formation and evolution of pond systems using methods from network science, thermodynamics and fluid mechanics, building on earlier modelling approaches [Bowen et al., 2018, Popović et al., 2018, Scott and Feltham, 2010]. We do this by modelling pond formation and evolution as a dynamical system on a network structure. A network model is a natural way of writing a system in such a way that we can examine percolation processes.

Pond models of three main types exist. One type focuses on the physics of individual ponds. These include models that focus on the thermodynamics of ponding

[Skyllingstad and Paulson, 2007, Taylor and Feltham, 2004], and models that examine particular process such as refilling post-drainage [Popović and Abbot, 2017], or refreezing [Flocco et al., 2015]. Another main type aims to model systems of ponds (many ponds and their interactions) using cell-based approaches. Some of these include very detailed physics, but overestimate pond extent when compared with observations, especially later in the season [Scott and Feltham, 2010, Skyllingstad et al., 2015]. A third type uses probabilistic methods and methods from statistical mechanics to model systems of ponds during particular, short stages in pond evolution [Ma et al., 2019, Popović et al., 2018, 2020a,b]. While narrow in scope, these have provided valuable insights into the behaviour of ponding, and have illustrated novel ways to represent ponding mathematically. There is therefore an opportunity to improve our models of ponding, using appropriate methods and results from models that focus on modelling individual ponds while also considering new mathematical structures for use in models for systems of ponds, and to build models that would represent ponding throughout the whole melt season. Some existing models have focussed on discretizing a floe into rectangular grid-cells. However, we know that ponding is controlled by the topography, and the ponds form within natural catchment areas for surface water [Petrich et al., 2012]. We will use these irregular catchments, instead of cells, as the discrete elements of ice topography.

Furthermore, a catchment-based model can be more computationally economical than a grid cell-based model, as the physics of melting and pond formation can be parametrised for each catchment. A network model can also easily be built in a way that is modular, so that the effects of individual processes of aspects of the physics can be included or not, with minimal modification to the model. This computational efficiency will allow us to carry out a range of experiments on the processes involved in ponding, on larger floes than a cell-based model would allow. A deeper knowledge of the behaviour of these processes can better inform how we parametrise ponds in climate models, and further our understanding of ponds in general.

(a)

(b)

Figure 1.1: (a) Sea ice floes in the Beaufort Sea, taken in September 2019, aboard the USCGC Healy. Open water leads shown have a width of the order of several metres. (b) Systems of interconnected melt ponds on sea ice, taken in summer 2014, courtesy of M. Doble. The darker ponds have melted through to the sea. The lighter ones have not. For reference, the RV Araon icebreaker (bottom left) is 110 m long.

1.2 Overview of the modelling approach

In this thesis, we describe a model of intermediate complexity for the evolution of melt ponds on sea ice, using methods and models from network science appropriately coupled with physical modelling of melt pond thermodynamics and mass transport of surface water. As a starting point, we know from Petrich et al. [2012] and Scott and Feltham [2010] that the topography of the ice at the start of the season is a controlling factor on the geometry of pond formation. Then, it is sensible to first take an ice topography and divide it into catchment basins, each with a local minimum and a boundary defined along the ridges between neighbouring minima. These catchment basins will be taken as the nodes in our network and will be chosen such that the set of basins completely covers the domain.

For a given node, we can draw an edge to each of the neighbours on its boundary. This process is outlined in Figure 1.2. This planar graph of potential edges defines the network at time zero, the start of the melt season. In accordance with observations from Webster et al. [2015], as the melt season begins, the ice layer starts to melt, quickly blocking the pores in the ice through the refreezing of melt water, halting drainage [Polashenski et al., 2017] and causing melt-water to settle on the surface. We assume that the melt-water of each catchment basin collects at its minimum height, and that the time taken for water to flow into the pond is negligible. Thus each node has an associated volume of water in it and an associated pond area, controlled by the water level filling up along the contours of the topography. Eventually, once drainage begins, nodes will connect through the ice to the sea, allowing water to flow out of the pond system. We will model the behaviour up until the floors of the ponds melt all the way through the base of the floe.

The thermodynamics and fluid mechanics, and hence the growth of a single pond are complex, and care must be taken in deriving an evolution equation for a pond node. This involves making some simplifications to the heat transport in the pond. We require an equation for the evolution of water level in each catchment basin and a corresponding equation for the area of each pond.

As a catchment area fills, eventually the water level will exceed the lowest saddle-point on the boundary and any excess water spills into the neighbouring pond over this point. At this stage in the model, the flux of water between the two ponds causes the edge between them to become active and directed, in the direction of the water flux. We must then derive an equation for this flux, and a new equation for changes in water level, subject to the constraint that the water level will be fixed at the height

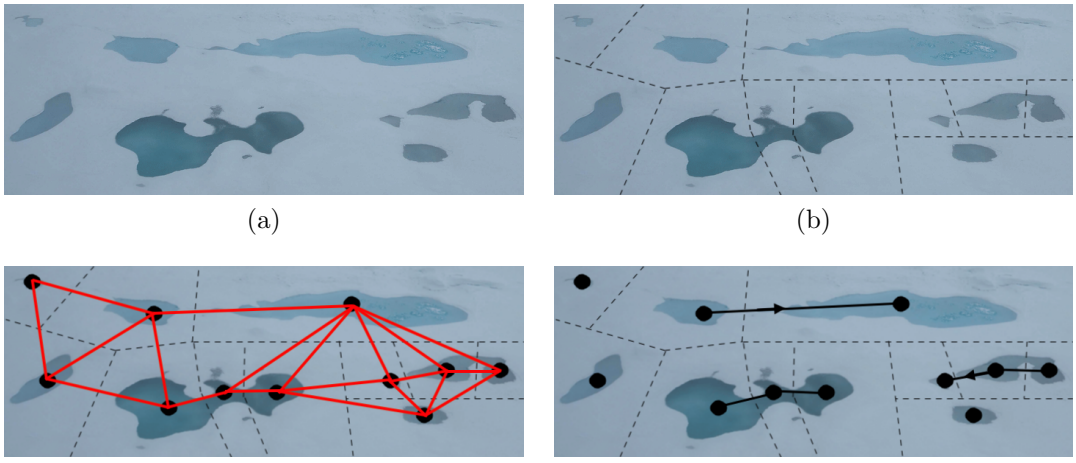


Figure 1.2: Schematic visualisation of building a network model from an ice topography. Panel (a) shows the domain, with ponds, whereas the model will start from a pond-free state. In panel (b), we divide the domain into catchment basins. In panel (c), we identify the local minima in the domain and the network of potential edges. In panel (d), active edges are added where ponds join or overflow into one another. Arrows denote directed flow from a higher pond to a lower pond. Background image reproduced from [Lu et al., 2018].

of the saddle-point in the overflowing pond. Further, as the melt season progresses, neighbouring ponds will flood over the boundaries between them and join, forming the intricate interconnections of ponds described by Hohenegger et al. [2012]. The joining of ponds is best represented by undirected active edges, with flow rates subject to the constraint that the water levels in the joined ponds are equal. This constitutes another equation for the fluxes between ponds and the evolution of the water level in each joined pond.

To model drainage once the ice becomes porous later in the summer, we connect each catchment node to a 'sea' node. A catchment will begin to drain once the ice thickness beneath the pond reaches a predefined critical value. This critical thickness is used as a proxy for ice temperature, which controls permeability. We choose height as a proxy because a relationship exists between ice thickness and internal temperature [Taylor and Feltham, 2004]. The drainage flux is modelled as Darcy flow, and is dependent on the head of water between the pond surface and sea level.

To summarise, our network will be defined by an initial state and arrangement of nodes, and three different states of edges between catchments: inactive potential edges, active overspilling edges, and active joined edges. A flux equation must be derived for each type of edge (with potential inactive edges having the trivial equation

of zero flux). The type that each edge becomes will be controlled by the height of the water levels in the ponds relative to the heights of the saddle points. Each catchment node will also be connected to the sea node, defining another type of edge, with separate conditions for whether these are active or inactive, and with associated water fluxes, controlled by the height of water levels in the ponds relative to sea level. All of these fluxes will then be incorporated into an 'activity' equation for the water level of each node.

The model presented in this thesis is supported by fieldwork undertaken in Autumn of 2019 in the Beaufort Sea. As part of the SODA project I sailed from Dutch Harbor, Alaska on the 29$^{\text{th}}$ of August, aboard the USCGC Healy, north to the Beaufort Sea, to a maximum latitude of 81°N, and disembarked at Nome, Alaska on the 12$^{\text{th}}$ of October. The aim of the cruise was deployment and retrieval of underwater instruments tethered to the sea floor, and of floating buoys initially drilled into the ice and allowed to drift with on the floes and eventually in the ocean. We also deployed drifting buoys on the ice and deployed a number of underwater autonomous vehicles. Instruments deployed on the ice were arranged on two specifically chosen ice stations on two different floes.

During the cruise, I made observations of the Arctic Ocean, the Arctic environment and the state of the ice. I took transect measurements of the ice thickness on the floes on which we located the ice stations, and made inspections of the ice topography, especially of areas with newly re-frozen ponds. During these inspections I took measurements which I used to estimate scale lengths for the lateral extent of pond catchments, and the average deviation of topographic heights. These are used in Chapter 3. I also made some inspections of the shorelines of refrozen ponds, especially the overhanging cliffs, and carried out drilling transects of ponds to determine the shape and slope of pond floors. These transects are discussed in Appendix B.

1.3 Mathematical and geophysical questions

Whilst separate models have focussed on individual aspects, the life-cycle of ponds during the summer, including initial growth, drainage, and re-growth post-drainage is still poorly reproduced by any one model. We aim to develop a model that will resolve the entire summer-time evolution of melt ponds, as described in Section 1.2, and spend much of this thesis doing so. While developing the model, we are motivated by a wish to understand how the behaviour of ponds depends on parameters that include the melt rates of the ice, the initial surface topography of the floe, and properties of the

ice. We also wish to understand the processes involved in pond evolution, and the mathematical structures that have been observed in models for pond geometry and behaviour. Given that several authors have focussed on the geometry of connected pond systems, and speculated that the connectivity could reflect an important aspect of pond dynamics, we will use the model to investigate what causes and controls the observed evolution of pond areas, perimeters, and connectivity.

In this thesis we have several questions to investigate. We must determine how to accurately represent the essential physics involved in pond evolution throughout the summer melt period as a dynamical system on a network, using a concise set of activity equations with a set of edge rules on an underlying network structure. This presents several mathematical modelling questions and issues of structural and algorithmic design, which we will focus on for several chapters. These questions include how to parametrize the dynamics of melt within catchments as node behaviour and how to write equations and constraints for the different fluxes between catchments, which define the edge behaviour, as well as conditions for when fluxes can occur between catchments. They also include how to model drainage as a distinct form of water flux, with different constraints and conditions. These processes will require particular mathematical structures to represent the connectivity and activity of such a network, which will have several different types of edge.

We will use the modular structure of the model, and take advantage of its computational simplicity to attempt to determine what are the more important processes and parameters that control pond extent and loss of ice volume due to ponding. In particular, we investigate how the progression of events in ponding (filling, drainage and refilling) depend on the properties of ice floes, and what processes need to be included in the model to recreate observations. With the network model, we aim to use mean-field approximations to estimate the area coverage of ponds and the volume loss of ice through the summer as functions of the melt rates and characteristics of the ice topography. We also aim to derive approximate time scales for the events that occur during pond evolution, including the time at which a floe could flood, the time of the onset of drainage, and the time at which pond water levels relax to sea level post-drainage. The model will also be used to investigate why it is possible to observe floes unexpectedly absent of ponds (such as on floes close to other floes that do not have ponds).

We check that our model recreates the area-perimeter scaling and fractal dimension D, initially investigated by Hohenegger et al. [2012], and later verified from independent observations by Huang et al. [2016]. Given that two regimes of fractal

dimension are observed in pond systems, and that there is a transition between them during the melt season, questions that naturally arise include: what is the underlying reason for this, and does the transition between the two regimes reflect any fundamental change in the dynamics of the system.

We aim to determine more clearly the role of connectivity in pond systems, and ask whether it accounts for behaviours that have not yet been explored, particularly when the finite size of floes is accounted for, something that our model addresses, but which previous models have not. Given that pond systems display evidence of a percolation process, we aim to determine a simple description of this, determine the nature of the percolation threshold, and investigate whether the percolation process differs substantially from classical random percolation. The question also arises as to whether the percolation process has any effect on the evolution of pond coverage, its growth rate, and the maximum extent of pond area. Indeed, what controls pond coverage will be a recurring question throughout the thesis, and one of the main questions we attempt to answer.

The obvious question that arises in studying the Arctic today is how will it change as the planet warms. At the end of the thesis, we will use the understanding gained from our model to discuss how changes to the ice pack and the meteorological conditions in the Arctic may affect pond extent in the future.

1.4 Thesis structure

The thesis starts with a review of the literature in Chapter 2. In Section 2.1, we discuss observations of, and models for, sea ice, focussing particularly on ponds. We discuss the thermodynamics of pond formation and evolution in Section 2.2, and provide a rationale for how we parametrize the melt rates of bare and ponded ice. In Section 2.3, we provide an introduction to topics in network science and dynamical systems on networks, providing enough material such that the reader can understand the aspects of the thesis that rely on network theory. We also provide an introduction to percolation theory, and discuss some recent advances in the field that include alternative models for percolation that display novel behaviour. A brief introduction to fractal dimension and how this relates to pond geometry is given at the end of Section 2.3. A deeper discussion on aspects of the theory of fractal dimension is given in Appendix C.

In Chapter 3, we model and describe node behaviour, namely the growth of a pond within a catchment subject to melting outside of the pond and within the pond. The

basic model for pond growth is developed in Section 3.2. We treat the problem both in one dimension (a pond in a long, thin channel) in Section 3.3, which helps to build intuition for the system and the model, and then in two dimensions (a pond in an axisymmetric catchment) in Section 3.4.

In Chapter 4, we model the fluxes between ponds in individual catchments as, and after, they begin to join. We define the types of edges that exist in our model, the conditions for when these edges form and break and quantify the fluxes along these edges, first in one dimension, in Section 4.2, then in two dimensions, in Chapter 4.3. In Section 4.4 we then use the equations for fluxes to determine an interaction term for our model, at which point we are able to describe mathematically the formation and evolution of systems of ponds on impermeable ice. An example numerical simulation of this model in one dimension is presented in Section 4.5.

In Chapter 5 we present simulations of the model and describe how we represent the ice topography numerically. Section 5.1 covers how the topography must be built, in such a way that we can easily define the architecture for our network model. This involves defining individual nodes and catchments and the boundaries between them, determining where edges can form, and determining point heights for peaks, troughs and saddles on the ice. We compare our synthetic topographies to data for the surface of floes at the start of the melt season in Section 5.2. We then discuss numerical simulations of the model without porous drainage in Section 5.5, and a version of the model with drainage over the side of floes in Section 5.6, asking how this particular process affects pond coverage.

In Chapter 6 we introduce porous drainage. We describe it mathematically in Section 6.1, and incorporate it into the network infrastructure in 6.2. In particular we outline how we determine when and how the transition in porosity occurs. With the inclusion of drainage, the model then accounts for behaviour from the onset of ponding to the time when ponds begin to melt completely through the floe. We provide an analysis of the model that includes drainage in Section 6.3, paying particular attention to determining what parameters have the most profound effects on drainage, and refilling of ponds post-drainage. For completeness, in Section 6.4, we present a version of the model that includes both drainage over the sides of floes and porous drainage.

Chapter 7 presents analysis of the model. From the output of several numerical experiments of the model, in Section 7.2, we derive time scales for all of the individual events in pond formation and evolution, and asymptotic behaviour of pond coverage during the melt season. The dependence of the model output on the parameters is explored, and we discuss the conditions under which ice does not form ponds, even if

melting occurs. The model is compared with data and observations in Section 7.4. We describe how to find the signatures of the area-perimeter scaling for ponds, and how to partition the dataset into the two regimes of scaling, and then test our model against the results of Hohenegger et al. [2012] in Section 7.5. We look for percolation processes in pond evolution, and examine the behaviour of these processes in Section 7.6. We also discuss what may be good, but simple models for the percolation behaviour we observe, and determine how these percolation processes depend on the topography, especially on the density of edges that pond networks can have.

Finally, in Chapter 8, we conclude, and discuss what our model and findings may mean in a warming Arctic in the near future. The appendices include a discussion of some considerations for numerical implementation of the model in Appendix A, a discussion of how the pond floor changes through time in Appendix B, and how the fractal dimension of modelled ponds changes through time in Appendix C.

Chapter 2

Relevant literature and theory

2.1 Mathematical models for sea ice

2.1.1 Sea ice

Arctic sea ice forms as the atmosphere in the region around the pole cools well below the freezing temperature of the ocean water in autumn, and the ocean loses latent heat to form a layer of ice. The solidification process is controlled by heat conduction through the ice, from the ice-sea interface vertically to the (cold) atmosphere above, leading to a slow down of growth as the ice layer thickens [Weeks and Hibler, 2014]. The solidification process involves expelling the salt present in seawater, which moves through channels of brine in the bulk of the ice [Zhu et al., 2006]. The presence of these channels makes the ice porous.

As a membrane between the Arctic ocean and the atmosphere, sea ice regulates the thermodynamics of both systems. Importantly, through the ice-albedo feedback, it regulates the amount of solar radiation absorbed by the polar oceans [Weeks and Hibler, 2014]. During the summer, the ice melts slightly from underneath as warm water migrates into the Arctic basin, and the upper ocean warms as it absorbs radiation from the sun. The ice melts predominantly on its upper surface due to increased air temperatures and solar radiation [Weeks and Hibler, 2014]. The water from surface melting then either flows over the edge of an ice floe or pools on the surface, forming melt ponds, that initially flood (or partially flood) the floe. Once the ice warms, the surface water begins to drain through brine-channels, and leaves melt ponds with intricate geometries, as seen in Figure 1.1 [Polashenski et al., 2012].

The ice is of maximum extent just prior to the onset of melting in late spring and has a minimum extent at the end of the melt season in September [Fetterer and Untersteiner, 1998]. Variations in the seasonal cycle are large; some areas which begin

with more than a metre of ice thickness in April often lose all of this by September. From September onwards freezing begins again, and remaining ice grows over the winter, forming multi-year ice (MYI). This is distinct from ice in its first year after forming on open ocean, referred to as first-year ice (FYI). First year ice tends to be flat, and relatively thin (around 1 to 2 metres thick), whereas multi-year ice tends to be rougher, due to the effects ponding, and also compressive forces and deformation as floes collide. MYI also tends to be thicker, due to refreezing in its second and subsequent winters, and mechanical deformation throughout this period. Due to this cycle, and the differences between sea and atmospheric conditions in different regions of the Arctic (such as the effect of land around the archipelagos, or currents in the Bering Straight), sea ice exists in a range of forms. These forms include large ice packs of consolidated ice floes spanning hundreds of kilometres wide in the centre of the Arctic, smaller floes of ice tens to hundreds of metres in width in the periphery (the marginal ice zone or MIZ) and small disks of ice less than a metre across in open water (called 'pancake ice') [Weeks and Hibler, 2014].

2.1.2 Sea ice models

Sea ice has also long been of interest to mathematicians, due to the variety of modelling problems it has presented. Stefan [1891] was motivated by the experiences of maritime expeditions searching for the Northwest passage.[1] These expeditions occasionally became ice-bound, causing much damage to ships as the ice mass started to expand after the September minimum. Sailors and scientists alike at the time were astonished at the rate at which the ice front (the moving boundary of the ice) would grow to enclose ships. A large dataset was amassed by Strachan [1885], from which Stefan was able in the 1890's to derive an expression for the progress of the freezing front of an ice layer in water, dependent on the heat fluxes in the ice and in the water [Stefan, 1891]. The mathematical ideas developed by Stefan formed the basis

[1]The earliest written account of sea ice seems to be from the Greek adventurer Pytheas [Hawkes, 1977], who, in an expedition dated to around 330 B.C., sailed from what is now France to the British Isles, and is said to have given the first written account of a circumnavigation of Britain. He went from there north to a place referred to as 'Thule', and sailed further north again until he encountered pancake ice and could make no further progress. Around here he also encountered the winter sun at the summer solstice. Pytheas' original account has been lost, and so our knowledge of his expeditions is completely from second-hand sources who quote him, some of whom disbelieved his accounts. His accounts have since been verified as plausible (including by the famous Fridtjof Nansen, no less). The location of Thule is disputed, but candidates include Iceland, the Faroe Islands, and the west coast of Norway [Cunliffe, 2001].

of 'Stefan problems', or analysis of a moving boundary in a PDE problem. These are discussed in Section 2.2.

The Northwest passage was eventually navigated by Roald Amundsen during an expedition lasting from 1903 to 1905, and progress in Arctic science accelerated through the early 20$^{\text{th}}$ century up to the Second World War, especially due to advances in aviation and aeronautics [McCannon, 2012]. Important contributions to the field of sea ice research were made by Russian scientists, who began studying the ice pack in earnest in the early decades of the Soviet Union, with the establishment of floating ice stations in 1935. An early and comprehensive account of this was published by Zubov [1943], and is probably the first extensive description of melt ponds in the literature. Zhubov described 'melt lakes', which may have been larger and deeper than modern ponds, if his observations can be taken to be accurate.

In the second half of the 20$^{\text{th}}$ century, as the Cold War intensified, American scientific attention turned increasingly polewards, especially towards the evolution of ice thickness. Maykut and Untersteiner [1971] derived a detailed model for sea ice thickness, incorporating a surface energy balance for heat fluxes at the top surface of the ice, a heat equation in the ice domain, subject to a source term for incoming solar radiation, decaying in intensity by a Beer-Lambert law, and Stefan's condition at the bottom surface.[2]

Quantifying and predicting the amount of ice in the Arctic remains an open question to modellers [Kwok and Untersteiner, 2011]. The main focus has been on quantifying ice extent, ice thickness, ice albedo, the energy budget for the ocean-ice-atmosphere system in the Arctic, and how all of these things change over time. Thorndike et al. [1975] published a different model for ice thickness to Maykut and Untersteiner, using a probabilistic approach rather than uniform thickness, as in the case of the Maykut et al. [1969] model, to derive a partial differential equation for the ice thickness distribution. This model was better able to describe the Arctic ice as a whole, as the ice is made up of many different types of ice with varying thickness. The model was able to describe all extremes of ice, from fragile first-year ice to large pressure ridges, ice formations that occur when two floes collide, piling up ice at the interface between them.

An example of the annual ice thickness distribution can be seen in Figure 2.1. The tail at large thickness, representing pressure ridges, is clearly evident. The peak

[2]The pair first published the model for the Rand corporation [Maykut et al., 1969] who were interested in conditions that would alter the thickness of the ice mass. The paper ominously noted that the heating may or may not be deliberately anthropogenic, and even goes into a six-page discussion on the feasibility of different artificial methods to melt the entire ice cover.

at two metres can also be seen, representing first and second-year ice, and a shoulder for multi-year ice.

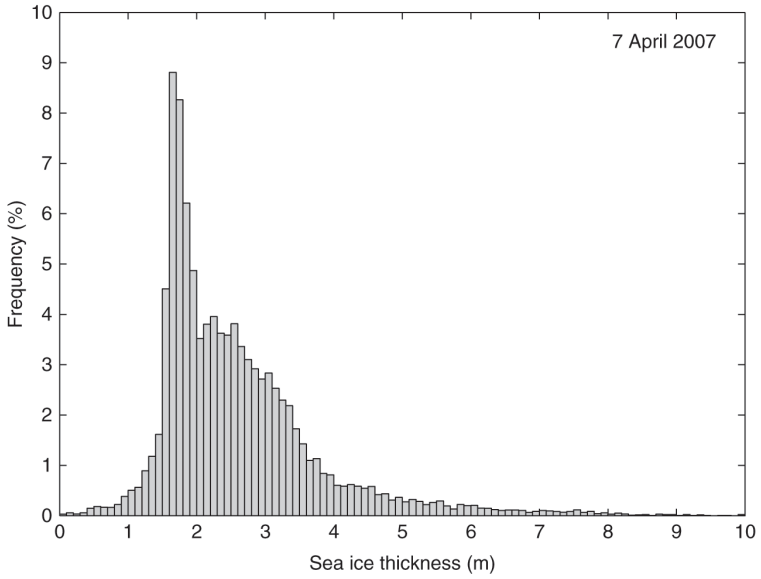


Figure 2.1: Example distribution of sea ice thickness, taken from a transect in the Beaufort Sea in April 2007, at 50 m intervals, reproduced from [Lemieux et al., 2017].

Throughout the rest of the 20$^{\text{th}}$ century, Thorndike made improvements to the model as datasets improved [Thorndike, 2000, 1992]. Toppaladoddi and Wettlaufer [2015] recently examined and modified Thorndike's initial PDE and found a representation as a Fokker-Planck equation. For realistic parameters they found very good agreement with some of the ice thickness distributions from the NASA ICEsat data. Many sea ice schemes used in current climate models are based on the models originally developed by Maykut and Untersteiner [1971] and Thorndike et al. [1975].

The improvements to Arctic data in recent decades include the vast amounts of data taken on the Soviet ice stations that became available to Western science after the fall of the Berlin Wall. Radionov et al. [1997], for example, describe trends in snow cover, atmospheric conditions and stages of the melt season from 1954 to 1991, taken from 29 of the 31 Soviet ice stations. Datasets on thickness below sea level recorded by US naval submarines have been declassified steadily since the 1960's [Kwok and Untersteiner, 2011]. The richest source of data of the Arctic in recent decades are Satellite datasets, which include the ICESat, ICESat-2, SHEBA [Uttal et al., 2002], Cryo-Sat, MODIS and MEDEA datasets. Higher resolution data also exists from extended field campaigns such as expeditions that contributed to the SHEBA and HOTRAX datasets. These measure both colour and temperature, from

which scientists can infer ice coverage, open water coverage, and pond coverage. Thickness and roughness however, are harder to measure. Further data will become available in the very near future from the MOSAIC project, in which the Polarstern was allowed to drift freely with the ice from September 2019 to the spring of 2020.

2.1.3 Contemporary ice loss

In the years after the boom in available data for the Arctic, it became apparent that the ice cover is quickly diminishing, in line with increasing annual average temperatures in the region. Kwok et al. [2009] give estimates for thickness and volume of Arctic ice from 2003 to 2008 using ice draft data provided by submarines and ICE-Sat satellite data. They reported a 42% decrease in multi-year ice coverage and an average thinning of 0.6 m for multi-year ice thickness over this period. Seasonal ice thickness, representing ice with a lifetime of a year or less, was found to be nearly constant, and its extent was found to increase annually due to spreading of first-year ice into areas previously covered by multi-year ice. A detailed overview for a general audience of the extent of the retreat of the ice cover, and its causes and effects was given by Kwok and Untersteiner [2011]. They mention in particular that climate models have vastly under-predicted sea-ice loss, although recent model developments allow for faster ice decline.

2.1.4 Melt ponds

The ice-albedo feedback is complicated by the existence of melt ponds. Water from the melting of the snow and ice surface during the summer collects in the hollows of the ice topography to form ponds. The ponds have a variable albedo, which depends mostly on depth, but importantly is much lower than that of ice [Morassutti and Ledrew, 1996]. The greater the coverage of melt ponds in a given area, the lower the effective albedo for that area, and the more energy the ice in that area absorbs during the summer. Melt ponds were proposed as a reason for the inaccuracies of sea ice and climate models as far back as the Maykut et al. [1969] RAND white paper. Quantifying the albedo of melt ponds, and their areal coverage throughout melt seasons has dominated scientific investigation into ponds since then. Grenfell and Maykut [1977] attempted to estimate an albedo for ponds from available data and some mechanistic models. The work was limited by the technology and data available at that time, and little data existed for the fraction of pond coverage on the ice cover, which is particularly important for estimating an average albedo for the Arctic. Later,

Morassutti and Ledrew [1996] took *in situ* measurements of albedo and depth of ponds in the Canadian Arctic Archipelago and found that albedo varied with cloud cover, pond depth and ice type, and that the morphology of the ice controls the depth of ponds. They also measured marked differences in ice-pond depth distributions of first-year, multi-year and land-fast ice.

Over the past few decades, more evidence of the role of ponds in ice loss has been collected and more has been learned about the complicated thermodynamics of these ponds. Schröder et al. [2014] correlated the September ice minimum, predicted using a global circulation model, with the modelled melt-pond fraction in early summer. Arntsen et al. [2015] examined the floe-size distributions for the 2014 melt season and found that they were heavily influenced in September by the distribution of melt ponds earlier in the season, whereby ponds cause weaknesses in floes. The authors go on to show that once put under stress from wind or currents later in the melt season, floes can then fracture along the long axis of ribbon-like ponds. Agarwal et al. [2011] provide histogram data for sea ice albedo in the Arctic basin over the course of the melt season for several years from Advanced Very High Resolution Radiometer (AVHRR) data. Melt ponds have been associated with changes in the histograms of albedo, with histograms for July and August, when the ponds represent a large part of the surface, showing markedly lower average albedo than other parts of the year.

Datasets for pond coverage improved from the 1990's onwards as more field expeditions were taken, and more satellite and aerial photography data became available, as mentioned above. Fetterer and Untersteiner [1998] give early observations of pond coverage and life-cycle and mention the uncertainties and unknowns, particularly in measuring pond albedo, and in quantifying the ponds' contribution to the ice-albedo feedback. Analysis of aerial photography and satellite data relating to ponds from circa 2000 to circa 2010 can be found in [Hanesiak et al., 2001], [Tschudi et al., 2008] and [Rösel and Kaleschke, 2012]. Sankelo et al. [2010] discuss data taken during a 2007 voyage in the Arctic using photographs taken from a camera mounted on the mast of the research vessel. The authors calculated melt pond fractions on ice from the data set and reported average pond fraction in late summer of circa 0.14 and a maximum of 0.4, in line with other measurements around the Arctic. Observations of ponding are also reported by Landy and Ehn [2014], Perovich et al. [2009] and Diaz et al. [2018].

Ehn et al. [2011] measured pond features along transects on land-fast ice to examine topographic changes in the ponds and ice surface, and transmission rates of

solar radiation for the ponds into the ice underneath. They found that to make transmission estimates, one needs information on the size and shape of melt ponds, and information on ice thickness under the pond, whereas for albedo these are unnecessary. Webster et al. [2015] observed the seasonal evolution of ponds on first-year sea ice for the first time and found larger pond fractions than on multi-year ice and that the melt season on first-year ice started 3 weeks earlier than for multi-year ice.

In recent years methods from machine learning have been used to improve estimates of pond coverage from satellite and aerial photography, taking advantage of algorithm development and increased computational capacity, such as the work of Han et al. [2016], and Fors et al. [2017]. In one of the more recent developments of tools for processing and analysing pond data, Wright and Polashenski [2018] developed a machine learning algorithm for sea ice that classifies satellite and aerial photography into bare ice, melt ponds and open ocean. We use this tool in Chapter 7 as part of our analysis.

Some attempts to analyse the areal coverage of ponds and how it changes over time have focused on the geometry and topology of the pond system. Hohenegger et al. [2012] looked at the area-perimeter relationships of ponds from many satellite images towards the end of the melt season. They found two distinct types of pond geometry, each with a different fractal dimension. One, which they called 'simple ponds', had perimeters that were close to circular, and fractal dimensions $D \approx 1$. These ponds were observed to have areas up to $a \approx 10$ m^2. The other, called 'self-similar ponds' had ramified perimeters that resembled space-filling curves, with $D \approx 2$. These were observed to have areas greater than $a \approx 1000$ m^2. Ponds with areas between these two values were called 'transitional ponds', and suggested that these represented ponds in a transition regime from simple to self-similar, and that the existence of the two separate regimes, and a transition regime suggested a percolation process, where pond systems move quickly from one regime to another. This would entail pond systems initially forming as a collection of simple ponds, which, as they flood, begin to connect to one another, forming transitional ponds. With a small change to the system, such as an increase in water level, these transitional ponds would then join together, and the system would very quickly form the large, ramified, self-similar ponds. The paper also suggests that the existence of the two different scaling regimes for ponds, and that their distinctive fractal dimensions, and the existence of a transition between the regimes can be used as a benchmark against which numerical models for ponds (or synthesised ponds) can be compared.

Huang et al. [2016] took data from a Chinese Arctic expedition in 2010 and analysed pond fractions, areal coverage and area perimeter ratios to determine power-law relations between them, similar to that of Hohenegger et al. [2012]. The authors found a sharp change in slope of the plot of the logarithm of perimeter against the logarithm of area, at a length scale of 100 m^2, in agreement with the transition to fractal behaviour found in Hohenegger et al. [2012].

2.1.5 Drainage of ponds

The life cycle of ponds is complicated greatly by drainage through the bottom surface once ponds fill and grow to a depth at which the ice warms so much that freshwater can no longer refreeze. This induces large-scale vertical percolation (as opposed to network percolation, discussed later) through the porous ice underneath. The form that ponds take on sea ice is greatly controlled by the way in which surface water drains through the ice after floes initial flood in early summer [Diaz et al., 2018]. Polashenski et al. [2012] offer a detailed description of the formation and evolution of melt ponds, and informed by experiment and satellite data, present a model for the drainage mechanism. This model explains some of the observed trends in pond coverage in the melt season. These trends involve an initial growth in pond coverage, followed by a sudden decrease in pond extent once the ice becomes very porous and drainage begins, at which point water levels in ponds fall to near sea level, and remain constrained at this level until the ponds either refreeze or the floes break up. After drainage there is also a small recovery in pond area over the course of the rest of the summer.

As ponds start to fill, the hydrostatic pressure at a pond bottom can grow large enough that the water begins percolating through the brine channels in the ice matrix. Ponds can also drain through macroscopic flaws. Polashenski et al. [2012] claimed that macroscopic flaws can form as the brine channels widen during the percolation process. The understanding of percolation drainage became more nuanced still when Polashenski et al. [2017] presented the results of several experiments validating the theory of pore blocking.

'Pore blocking' describes the process whereby the freshwater melt of early stage ponds re-freezes in the brine channels of the ice, making the bulk ice impermeable. This stops the percolation drainage of ponds early in the season. As the ice warms, the freshwater ice in the channels thaws again, causing a sudden increase in permeability, due to the inability of freshwater to refreeze as it flows deeper into the ice. The experiments demonstrated that drainage in ponds instead is controlled both by the

hydrostatic head and the temperature of the ice. Effectively, the authors show, the vertical permeability of sea ice has a transition from a low permeability regime to a high permeability regime. This behaviour is consistent with observations [Diaz et al., 2018] that in early summer, ponds grow for several weeks and almost flood floes, then over the course of around two days, ponds drain to near sea level.

The conditions at which the permeability changes are still only partially known. [Polashenski et al., 2017], from experiments, found that the permeability of the ice increases to around $10^{-10} - 10^{-12}$ m^2 once the bulk ice underneath a pond warms to -1.6 C, and becomes considerably larger once the temperature increases to -0.9 C. However, the temperature within the ice underneath a pond can be difficult to model, and is very difficult to measure in the field [Kim et al., 2018]. Quantifying when this major change in the permeability of sea ice occurs is still an open question, and will be discussed in Chapter 6.

Previous work attempting to determine the porosity and permeability include both experimental work, and modelling based on percolation theory. Eicken [2002] used chemical tracers to measure permeability of the ice and to estimate drainage fractions for ponds and found a relationship between melt volume and ice permeability that predicted retention fractions in ponds, the fraction of melt water that remains in the pond once drainage begins. The authors observed retention rates of 0.4 late in the season. Golden et al. [1998] developed an empirical 'rule of fives' to approximate a change in percolation behaviour in sea ice, whereby small changes in brine volume fraction around a threshold value of about 5%, corresponding to variations in temperature around a critical ice temperature of about $-5°$C, control an important transition between regimes of negligible permeability and permeability sufficient for significant flow. For more detail, Golden [2003] gives a review of the use of percolation theory in determining bulk properties and behaviour of sea ice.

Drainage can also occur over the sides of finite floes. This has not been considered in previous models for ponds. The modular nature of our model allows us to easily include side drainage in this thesis, and it is covered in Chapters 5 and 6.

2.1.6 Models of pond physics

A number of models for ponds have been developed, with a broad range of complexity, and as mentioned in Chapter 1, these can either focus on the physics of individual ponds to make conclusions that apply to ponds generally, or of systems of many ponds and their interactions. In order to examine the thermodynamic effects and albedo of individual ponds, Taylor and Feltham [2004] used a two-stream radiative model in

one dimension to model the heat transfer in sea ice with the inclusion of a snow layer and a melt pond layer. Lüthje et al. [2006] developed a partial differential equation model for systems of melt ponds on sea ice using an analogy to a two-phase system, much like earlier sea ice and brine inclusion models on a micro-scale. They also developed a cell model for ponding, where the ice was divided into square cells in the horizontal plane, with rates of melting of ice and pooling of water calculated in each cell. Water transport was also allowed between cells. Later, Scott and Feltham [2010] used the cell model of Lüthje et al. [2006] and the theromdynamic and radiation model from Taylor and Feltham [2004] to develop a more sophisticated cellular model for systems of ponds. Importantly, the model shows that the extent and geometry of late stage ponding is strongly controlled by the initial topography. Popović and Abbot [2017] modelled the growth of individual melt ponds after drainage by examining the hydrostatic balance of ice floes in the sea. The model accounts for an important factor in the changes in pond coverage as the season develops, with pond fraction decreasing once the ponds drain through macroscopic flaws.

Skyllingstad and Paulson [2007] used large eddy simulation methods to examine the lateral melting of individual ponds and the dependence of melting rate on pond shape. The authors found that there was some dependence of melt on shape, but not on the complexity of pond shapes. Skyllingstad et al. [2009] developed a model representing a number of melt ponds on a floe as the pooling of water on an uneven ice surface. Skyllingstad et al. [2015] used a cell model for to investigate the development of systems of poonds, similar to that of Scott and Feltham [2010], with melting behaviour and thermodynamics similar to that described in [Skyllingstad et al., 2009]. However, the authors included refreezing due to changes in ice temperature with pond depth, recreating the permeability transition suggested by Polashenski et al. [2012]. Having included this process, they found that pond coverage is still dominated by initial ice topography and snow variability and coverage, in line with the findings of Scott and Feltham [2010].

Climate modellers have used some of these models to improve the predictions of climate models, as part of the natural process of innovation and improvement in the field. Flocco et al. [2010] used the model for melt-pond evolution derived by Scott and Feltham [2010], to parametrize ponding in the Los Alamos CICE sea ice model. The use of this parametrization led to an improvement in the agreement of the CICE climate model with observed ice thickness and extent. Flocco et al. [2012] further tested the model detailed in Flocco et al. [2010] and found that predicted September ice volumes with the pond scheme were much closer to observations than models

without melt ponds. The melt pond scheme was most sensitive to the parameters of atmospheric and oceanic forcing, variations of optical properties and the amount of snowfall. For a flavour of how ponds are incorporated into climate models, a detailed overview of the current incorporation of melt ponds into the CICE climate model is given by Holland et al. [2012]. The approach taken by climate modellers to modelling melt ponds differs from the approaches mentioned earlier for modelling individual ponds or pond systems. Due to the scope and complexity of climate models, particularly general circulation models, it is not possible to resolve ponding in climate models, except through simple and computationally cheap parametrizations. Cheap and accurate parametrization of ponds are very valuable to these models, and therefore the polar science and climate modelling communities look to continually improve them.

2.1.7 Connectivity and network models

Several authors have used ideas from statistical physics and complex systems theory to model systems of melt ponds. Ma et al. [2019] used an Ising model to model the evolution of melt pond area and geometry in the horizontal plane. The model is motivated heavily by the relations found in Hohenegger et al. [2012], and reproduces the statistics of observed pond geometries. Bowen et al. [2018] modelled the evolution of melt ponds using a uniform water level on a simulated topography based on a two-term random Fourier series in two dimensions. Pond extent was determined by the level-set intersection of the water level with the random topography, and the model was able to replicate observed pond extent and geometry, with little physics. Popović et al. [2020b] used a cell model similar to that of Scott and Feltham [2010] to model the drainage of ponds. They started with flooded ice and allowed drainage through cells, in a random process parametrized by the thermodynamics of ice-melt, in an aim to simulate the random opening of macroscopic flaws in the ice. They found that this causes a percolation process, similar to a forest fire model [Turcotte, 1997], and suggested that this process displayed universality across different topographies, and used these findings to derive a universal function for pond coverage during the drainage phase. In this case, 'universality' is taken to mean that the behaviour of the percolation process, dependent on the fraction of open drainage holes, is the same for all topographies, under an appropriate scaling. The authors also found that the percolation threshold for this process was a strong control on the behaviour of the drainage process and argue that the pond fraction at the percolation threshold can serve as an upper bound for pond fraction post-drainage.

Popović et al. [2018] modelled pond coverage using a 'void' model. This consisted of randomly placing circles on a plane, and then classifying the points not covered by any circle as water covered, thereby defining ponds as connected regions not covered by the laid circles. The void model could be thought of as the dual of a tiling of overlapping circles placed at random in a two dimensional domain. They showed that the model replicates the geometry of observed ponds and then concluded that according to the model, the bare ice that exists later in the melt season is determined by the initial undulations on the ice surface, in line with other observations that topography dominates ponding. From observations, the existence of snow dunes or rougher sea ice topography have been seen to greatly influence melt pond formation and coverage [Petrich et al., 2012]. Popović et al. [2020a] expanded on the results of [Popović et al., 2018] to model snow coverage on an ice floe as a randomly distributed set of mounds, shaped as Gaussian curves in three dimensions. The model was tested against Lidar data of snow and ice topographies from [Polashenski et al., 2012] and found that the modelled snow topography was almost identical to the underlying ice topography. The authors used the statistics of simulated synthetic topographies and a hydrostatic balance model to model the initial stage of pond development while the ice is still impermeable, and proposed criteria for the conditions under which ice would become pond-free later in the summer. They found that this depends primarily on the surface roughness of the ice, with smoother ice more prone to becoming pond-free.

Barjatia et al. [2016] analysed aerial photography of ponds, and used an image segmentation method to identify connected regions of ponding. They then built a conductance network from the segmented images. So far this has been the main attempt to represent ponds as a more complicated network than a simple grid-cell representation.

2.1.8 Modelling opportunities

Several questions about melt ponds, pond systems, and the modelling of both present themselves. Many are driven by the need to quantify pond extent throughout the summer, pond volume, geometry, depth and the rate of ice loss on floes with ponds. As will always be the case, an opportunity to improve on parametrizations of melt ponds in climate models exists. Given that this is the case, does a catchment-based model rather than a cell model lead to improvements in predictions of pond extent in a model and thereby a more accurate parametrization of ponding for use in climate models? In such a model of intermediate complexity, we must determine what are the most important physical processes to include. Some work has been done on

examining how the topography affects pond coverage. Little work has focussed on how the topography affects the topology of pond systems, and how the topology of pond systems early in the melt season affect coverage later in the season. This is of interest, given that there is evidence of a percolation process in pond systems, which will depend on the topology. Further, this percolation process has not been analysed in detail, especially during the filling of ponds.

2.2 Thermodynamics of ponding on ice

For the model we build and describe in this thesis, it is preferable for the behaviour of a single pond to be described simply and concisely so that the evolution of ponds in a network can be modelled without overly costly computation. The thermodynamics of a single melt pond, and the calculation of melt rates for ice in ponds are complicated, with heating by absorption of the short-wave radiation in the bulk of the pond, conduction and melting at the ice-water interface at the pond bottom and side walls, and a surface energy balance at the interface with the atmosphere. The thermodynamics of the melting of bare sea ice is somewhat simpler, but some detailed modelling is still required to quantify bare ice melt rates.

However, it is possible to parametrize the thermodynamics of both processes into simple melt rates for bare and ponded ice $v_b(t)$ and $v_w(t)$ respectively. A detailed model for the thermodynamics of the melting of bare and ponded ice, is described by Taylor and Feltham [2004]. A brief sketch of the important physics involved is presented here. Because ice floes and ponds are much thinner than they are long or wide, variations in temperature along an ice floe or pond are small. Therefore, we consider a one-dimensional slab model for bare ice - an ice region of finite thickness sitting on top of an infinite ocean layer and below an infinite atmosphere layer. For a pond, we consider a finite water layer above a finite ice layer, with an infinite atmosphere above the water and an infinite ocean below the ice.

The melting of bare ice is dominated by the balance of heat fluxes on the top surface of the ice, the surface energy balance (SEB). We write the SEB for bare ice (which is also a Stefan condition, and upper boundary condition for the slab model) as

$$(1-\alpha_i)I(0) + B - \sigma T_{mf}^4 + \hat{H}_i(T_a - T_{mf}) - \rho_i L_i v_b = k_i \frac{\partial T_i}{\partial z} \quad \text{at } z = h, \tag{2.1}$$

where we have assumed that the surface is melting. The height of the ice surface is $z = h$. The term B is the down-welling long-wavelength (long-wave) radiation

caused by cloud cover, σT_{mf}^4 is the black-body radiation from the ice (with σ being the Stefan-Boltzmann constant), which we assume is at the melting temperature of fresh water T_{mf}. The term $\hat{H}_i(T_a - T_{mf})$ is a parametrization of the sensible heat fluxes due to the heat and motion of the atmosphere just above the ice. This term is dependent on the air temperature T_a, and the parameter $\hat{H}$ includes contributions from the average wind speed, the specific heat capacity of the atmosphere, the density of the overlying air and a heat transfer coefficient. We assume that an average value of this parametrization and air temperature can be used during the Arctic summer. In this illustrative example, we neglect a small, humidity-dependent latent heat term on the left-hand side of the SEB.

The melting term $\rho_i L_i v_b$ includes the density of ice ρ_i and the latent heat of melting for ice L_i. The short-wavelength (short-wave) radiation term $(1 - \alpha_i)I(0)$ accounts for the high-energy radiation absorbed in the very top centimetres of the ice, which has albedo (reflected fraction) α_i, and contributes greatly to melting. We write the irradiance here as $I(0)$ as we prescribe incoming short-wave radiation at the upper surface of the system, assuming no dissipation so far. The heat flux from the ice $k_i \frac{\partial T_i}{\partial z}$, where k_i is the thermal conductivity of ice, can be determined by the heat equation in the ice layer,

$$\rho_i c_i \frac{\partial T_i}{\partial t} = k_i \frac{\partial^2 T_i}{\partial z^2} - \frac{\partial I_i}{\partial z} \tag{2.2}$$

with the addition of a radiative heating term due to the irradiance in the ice I_i. The parameter c_i, is the heat capacity of ice. The model of Taylor and Feltham [2004] assumes a constant heat flux from the ocean as a boundary condition at the base of the floe.

For a pond, a similar SEB applies at the top surface,

$$B - \sigma T_w^4 + \hat{H}_w(T_a - T_w) = k_w \frac{\partial T_w}{\partial z} \quad \text{at } z = h + d, \tag{2.3}$$

where we have denoted the depth of the pond d, where σT_w^4 is the black-body radiation due to the pond temperature T_w, and $\hat{H}_w(T_a - T_w)$ is a parametrization of the sensible heat fluxes between the atmosphere and the pond. The parameter $\hat{H}_w$ differs from $\hat{H}_i$ only by the heat transfer coefficient, which is for air-water rather than air-ice. Again, for the pond, we neglect a small, humidity-dependent latent heat term on the left-hand side of the SEB. Temperature in the pond is given by a similar heat equation to that which describes the ice layer,

$$\rho_w c_w \frac{\partial T_w}{\partial t} = k_w \frac{\partial^2 T_w}{\partial z^2} - \frac{\partial I_w}{\partial z}, \tag{2.4}$$

where c_w is the heat capacity of the water. There is no short-wave radiation term in (2.3) as this is accounted for in the heat equation for the pond, due to the longer length scale over which this radiation is absorbed in water.

Another Stefan condition applies at the pond floor,

$$\rho_i L_i v_w + k_i \frac{\partial T_i}{\partial z} = +k_w \frac{\partial T_w}{\partial z} + (1 - \alpha_w) I(d) \quad \text{at } z = h. \tag{2.5}$$

Here k_w is the thermal conductivity of freshwater. The first term on the left-hand side (LHS) is the latent heat, the second is the heat flux into the ice, and the term on the right-hand side (RHS) is the heat flux from the pond water. For the melting equation on the pond floor (2.5), the heat flux into the ice is determined in the same way as for the bare ice case. The second term on RHS is the short-wave radiation term, with albedo of the pond-ice interface α_w and reduced irradiance $I(d)$ due to losses having passed through the pond layer.

The depth of ponds only extends to tens of centimetres [Morassutti and Ledrew, 1996], which is less than the extinction length scale for short-wavelength radiation in water, which at 500 nm wavelength is of the order of tens of metres [Hale and Querry, 1973] (given an extinction coefficient of 0.025 m^{-1}). Therefore much of the incident radiation will be either absorbed in the ice or reflected by the bottom of the pond. Due to wind-driven and convective turbulence in the ponds, as a first order approximation, we assume that the pond is well mixed and that the bulk water temperature remains slightly above the melting temperature [Skyllingstad and Paulson, 2007]. For this reason, we assume that there is a thin, diffusive boundary layer in the water layer at the pond floor, describing a transition between the bulk pond temperature and the melting temperature. Therefore, the heat flux from the pond, the term on the RHS of equation (2.5), can be parametrized by a Newton cooling expression, $\frac{\partial T}{\partial z}|_d = J(\bar{T}_w - T_{mf})$, where J is a heat transfer coefficient (of order of the depth of the boundary layer), and $\bar{T}_w$ is the depth-averaged bulk pond temperature. Water temperatures in ponds have been measured to be just above freezing [Roeckner et al., 2012], indicating that accumulation of heat in the pond can be considered to be negligible. We then assume that, during midsummer, there is a steady balance of heat in the pond, and that any excess heat flux from the SEB contributes to melting. Therefore, we assume that a simple approximation of the melt rate for the ice within the pond can be determined, based primarily on the net balance of incoming and outgoing fluxes at the pond surface, accounting for the solar radiation reflected and absorbed at the pond floor. We also assume that a simple melt rate for bare ice can be determined based primarily on the net balance of incoming and outgoing fluxes at the ice surface.

In previous models for melt ponds, such as that of Scott and Feltham [2010] it has been approximated that $v_w \approx 2v_b$, measured by Untersteiner [1961], and motivated by the model of Taylor and Feltham [2004]. From field measurements, the melt rate in ponds varies between 1 and 3 cm/day [Untersteiner, 1961], [Polashenski et al., 2012]. The primary factors accounting for this disparity in the melt rates include the difference between the surface temperature and pond water temperature, which impact outgoing energy fluxes, and the differences between the albedos of bare ice in contact with the atmosphere and water covered ice. In this thesis we will assume the melt rates $v_w(t)$ and $v_b(t)$ are constant, but will build the model in such a way that it can accommodate time dependent melt rates, or melt rates that depend on the average depth of ponds.

Given that the ponds are shallow and exist in catchment basins with shallow curvature, we neglect lateral melting in the ponds. We also neglect wind-driven focussed melting at the sides of ponds, whereby ponds would favourably melt at one end as warmer pond water is pushed by the prevailing winds towards this end [Skyllingstad and Paulson, 2007]. Early stage ponds consist primarily of fresh water from surface snow melt, and are assumed to be fresh. Where salinity has been measured in late stage ponds previously, these tend to be significantly fresher than seawater. However, recent observations have found evidence of ponds with more significant salinity, sometimes as high as 29 g/kg [Kim et al., 2018]. The inclusion of a moderate-to-high salt concentration would shift the melting temperature of the ice and the thermodynamics in the pond, altering the melt rates. In our model, we will neglect the effects of salinity in the thermodynamics of ponds. However, we acknowledge that this is an uncertainty in parametrizing melt rates.

2.3 Networks, percolation and fractal dimension

In this section we present an introduction into topics in network science. This is not intended to be an in-depth tutorial on the field, but a basic description of a network and those definitions and results that are relevant to the work presented in this thesis, especially in Chapters 4, 5, 6 and 7. We also present an introduction to percolation theory, anomalous percolation, and fractals, all of which will prove useful in Chapter 7. This material on networks is taken from several well known books and review papers, to which we direct the reader for more information, if needed. Newman [2018] and Barabási and Pósfai [2016] both provide good introductions to the theory of networks, including basic mathematical models and methods of analysis, as well as key

results in the field. Porter and Gleeson [2016] give in-depth discussion of dynamical systems on networks. All of these seminal books on network theory are from the last decade or so. Percolation theory is older, with popular textbooks dating back to the 1990's and further. Stauffer and Amnon [2003] and Bollobás and Riordan [2006] both cover the important elements of percolation theory, the development of which dates back as far as 1957 [Broadbent and Hammersley, 1957]. Percolation on networks is covered by Newman [2018] and Barabási and Pósfai [2016]. Anomalous percolation is a more recent sub-field, having arisen in 2009 [Achlioptas et al., 2009]. Review papers by D'Souza et al. [2019] and D'Souza and Nagler [2015] give a good overview of the field as it is at present. Fractals are a mature field, and an excellent book on the mathematics of the topic exists by Falconer [1990]. Geophysical applications of fractals are covered by Turcotte [1997] and the seminal overview is by Mandelbrot [1983].

2.3.1 Network science

2.3.1.1 Definitions

When studying complex systems, we are presented with something that is made up of many individual units, connected together by a set of relationships. One example is an ecosystem, or food web, where the system is made up of units, individual species, and relationships exist between them, in which one species tends to eat another. To model a system of this kind, we represent it as a network. A network consists of a set V of individual units, called nodes, and a set E of relationships called edges. These define a mathematical structure called a graph

$$\mathcal{G} = (V, E). \tag{2.6}$$

Each edge, e is a defined as a pair of nodes v and v', whereby $e = (v, v') \in E$. We denote the number of nodes in a graph as n, and the number of edges as m.

These structures were first developed by Euler, in order to solve the famous Königsberg bridge problem in 1735 (now Kaliningrad, Russia). The wealthy market town is located at a branch in the river Pregel with part of the town on an island in the middle of the river. This divided the town into four landmasses, which were connected by seven bridges. A question arose as to whether someone on a walk could cross all of the bridges, but only cross each bridge once. By representing each landmass as a node, and each bridge as an edge, Euler was able to prove that this was

impossible, having derived the formula

$$n - m + f = 2 \tag{2.7}$$

where m is the number of edges, n is the number of nodes and f the number of faces. Here a face is an area bounded by connected edges of the graph. A graph is planar if it can be projected onto a plane without any edges crossing, and the graph of the Königsberg bridge is a very clear example of a planar graph.

An edge is said to be undirected if it represents a reciprocal relationship, whereby $(v, v') \in E$ and $(v', v) \in E$. An edge is said to be directed if this is not the case. A directed edge runs 'from' the first node in the edge pair v, 'to' the other v'. The nodes in the edge (v, v') can also be thought of as 'negative' and 'positive' respectively, in that something flowing in the network would flow out of v and into v'. The nodes v and v' can also be thought of as the 'left' and 'right' nodes in the edge respectively. In an ecosystem, for example, there is an directed edge between the fox and the rabbit, as the fox will eat the rabbit, but not vice versa. In fact, in a food web, the vast majority of edges are usually directed. In a social network, however, edges are typically undirected. If Alice is a contact of Bob's, then Bob in turn must be a contact of Alice's. A network is undirected if it has no directed edges. Otherwise it is directed. We define a neighbour of a node as another node that is joined to it by an edge.

An edge can also carry a weight, a function that denotes how important it is in comparison to other edges, and gives this importance a numerical value. A network has a multi-edge if more than one edge joins the same two nodes. An edge is a self-edge if it joins a node to itself. Both multi-edges and self-edges can cause difficulties in numerical algorithms that make use of networks. Given that we will not absolutely need them in our model, we will not discuss them further.

2.3.1.2 Data structures

The simplest data structure to record a network is an edge list, which is simply a list of the members of the set of edges. The edge list is written as a set of pairs of nodes,

$$\{(u_1, v_1); (u_2, v_2); \ldots; (u_m, v_m)\}. \tag{2.8}$$

This can be written as an $m \times 2$ matrix, or a pair of column vectors of length m, one for the u's and another for the v's, which we will do later in this thesis. We must be careful in using an edge list to distinguish between a directed and undirected network. For an undirected network, it is implied that equation (2.8) includes (v_1, u_1) as well

as (u_1, v_1) and so on. For a directed network, this is not the case, and if an edge goes in both directions it must have an entry in each direction.

In order to represent a graph in a form that can make use of the tools of linear algebra, we can use the adjacency matrix, A_{ij}. We say that two nodes are adjacent if they share an edge. For an undirected, unweighted network, the $n \times n$ adjacency matrix is defined by

$$A_{ij} = \begin{cases} 1 & \text{if node } v_i \text{ is adjacent to node } v_j, \\ 0 & \text{otherwise,} \end{cases} \tag{2.9}$$

and the matrix is symmetric. For a directed, unweighted network, the same definition holds, but the matrix will not be symmetric. In both cases, the diagonal is filled with zeros, unless the network contains self-edges. For weighted edges, the associated entry in the adjacency matrix takes the value of the weight instead of a 1.

We can also describe the network structure using the directed edge-incidence matrix,

$$B_{ij} = \begin{cases} -1 & \text{if edge } e_i \text{ goes from node } v_j, \\ 1 & \text{if edge } e_i \text{ goes to node } v_j, \\ 0 & \text{otherwise.} \end{cases} \tag{2.10}$$

The edge-incidence matrix is an $m \times n$ matrix that can be useful in mapping quantities that live on edges to those that live on nodes, and vice versa. For an undirected network, the edge-incidence matrix has values of $+1$ where there would be values -1 in the directed case. We describe a network as sparse if it has a small number of edges compared to the possible maximum amount of edges and dense if the relative number of edges is large. We define the degree k of a node v_i as the number of edges incident on it.

2.3.1.3 Walks and paths

Networks are often used to model the transport of something amongst individual entities, such as current on an electricity network, where the entities are generation stations and metropolitan areas (and increasingly include solar farms and wind turbines), or information on a social network, where the entities are people or organisations. We can define a walk on a network as a succession of adjacent nodes that one can travel to by travelling along edges. This was exactly what Euler was trying to model in the Königsberg bridge problem. A walk is defined to be a path if each node is only visited once (with the exception of the starting node if the walk also ends there). We may say that two nodes are connected if a path exists from one to

the other. For undirected networks, connectedness is symmetric, and transitive. If u is connected to v, then v is connected to u. If u is connected to v, and v is connected to w, then u is connected to w.

Directed networks are more complicated. If u is connected with a directed edge to v only, then a walker could go from u to v, but could not go from v to u. For a directed network we say two nodes are strongly connected if a path exists from one to the other in both directions. Two nodes are weakly connected if a path only exists each way between them if the direction of edges is ignored. A connected component (or just component) is a set of nodes for which any one node in the set is connected to each other node by a path. As for nodes, components in directed networks can be strongly or weakly connected.

We also define some structural features of graphs that we will meet again later. A connected component may have sufficient edges that a cycle forms. A cycle on an undirected network is defined as any set of edges that forms a closed loop path. In the example above, if u is connected to v, and v is connected to w, and by another edge w is connected to u, then this forms a cycle. A cycle on a directed network is defined as any path that forms a closed loop where the arrows on the edges in the path all face in the same direction around the loop. We call a graph or sub-graph a tree if it is connected and has no paths that are loops. If a network has no loops but is not fully connected we can call it 'tree-like'.

2.3.1.4 Network models

A wealth of models exist to generate networks mathematically in ways that are tractable to deep analysis, particularly of structure and dynamics. The simplest such model is a chain. In this case, a set of nodes is arranged in a line, and edges exist between nearest neighbours, as in Figure 2.2. For a chain, the degree k is 2, for all except the two end nodes. If we add an edge to link the two end nodes, we form a ring, for which $k = 2$ for all nodes. The idea of a ring, where each node in a line is connected to its nearest neighbours can be extended to a more general arrangement as a lattice, where the points do not necessarily lie in a line, but do have a repeating pattern. In the case of a square lattice, such as can be seen in Figure 2.2 (c) or Figure 2.3 (b), each node shares an edge with its four nearest neighbours (above, below, to the left, and to the right).

Real world networks are rarely as ordered as rings and lattices however, and are often non-planar. Their topologies tend to have a degree of randomness and many models, from simple to complex, have been developed to represent this randomness.

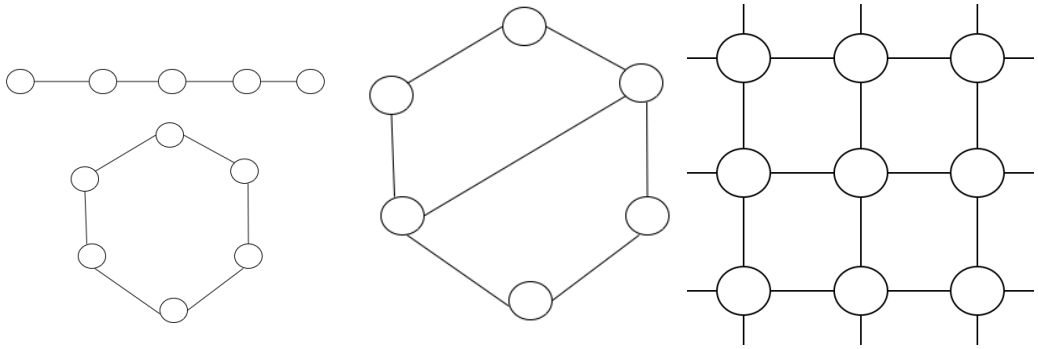


Figure 2.2: (a) A chain (top) and a ring (bottom). (b) A closed ring with one random edge, a simple version of the Watts-Strogatz model. (c) A lattice network.

Possibly the simplest non-planar network is the Erdős-Rényi or ER network, developed in 1959 [Erdős and Rényi, 1959]. The model starts with a set of isolated nodes N. Each pair of nodes is considered in turn and an edge is added between each pair according to a Bernoulli process with probability q. Self-edges and multi-edges are excluded. This forms an object $\mathcal{G}(N, q)$, a random graph with parameters N and q. In practice, ER graphs are used in an ensemble, where many $\mathcal{G}$'s are created numerically, and properties of the network are averaged over all members of the ensemble to generate robust results for this class of network. The averaging process removes the effects of any extreme behaviour that occurs by chance in any one simulated graph. We will also use this approach of averaging when simulating pond networks.

Another simple non-planar random network is the Watts-Strogatz model, developed to study the small world phenomenon [Newman, 2018]. The small world phenomenon is a statement of occurrences of unexpectedly short paths between nodes in certain types of random networks, such as social networks, whereby for instance, by passing a parcel through mutual friends, I could reach almost anyone on the planet within fewer than ten steps.[3] The simplest form of the Watts-Strogatz model starts with a ring of n nodes. Each node is considered in turn, and like in the ER model, an edge is added between that node and another at random with probability p.

[3]Or within a citation network, if mathematicians are nodes, and authorship of a paper defines an edge between mathematicians, I and Paul Erdős are only separated by four edges. You the reader may be even closer to him. This metric is called the Erdős number. A similar metric exists for films, called the Bacon number, where the target node is Kevin Bacon, which may sound familiar, as it is derived from the game 'six degrees of Kevin Bacon', a very neat example of the small-world effect. A composite metric called the Erdős-Bacon number even exists for those mathematicians whose lives are so interesting that they also have had an acting career at some point.

2.3.2 Dynamical systems on networks

When studying systems we are often as much concerned with dynamics as we are with structure. Networks afford us several useful ways to model and tools to investigate the dynamics of systems consisting of many interrelated entities. Well-known examples include SIR models (Susceptible-Infected-Recovered) for epidemiological processes on networks [Newman, 2018], or models of oscillators on networks, such as the Kuramoto model [Kuramoto, 1975]. Models of oscillators on networks are frequently used to investigate the phenomenon of synchronization in random systems, where the coupling of oscillating nodes that begin out of phase with one another can move to a state where they are all in phase with one another. Examples of the use of these models include the development of opinion formation among populations, or the firing of neurons. An in-depth review of synchronicity on networks can be found in Arenas et al. [2008].

To model dynamics on a network, we assign a dynamical variable which we call the activity $x_i(t)$ to each node. A dynamical system can be written on the network as

$$\dot{x_i} = f_i(x_i, t) + A_{ij} g_{ij}(x_i, x_j, t), \tag{2.11}$$

where f is a function of the node attributes and g represents how neighbouring nodes affect each other. In our model, we propose to write the water levels in catchments as the activity on each node and the water fluxes between catchments (and later the sea) as interaction terms (the $A_{ij} g_{ij}(x_i, x_j, t)$).

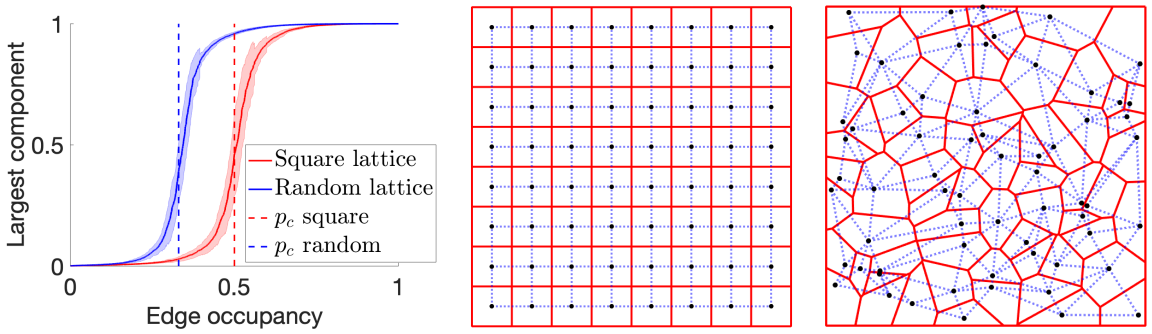
From equation (2.11), it is obvious that the network structure (encoded in A_{ij}) affects the dynamics of the system. It is also possible for the dynamics to affect the network structure, such as in a process in which edges can be switched on or off according to rules that depend on the activity of the nodes. Such a network is called an adaptive network, and we will make use of this kind of structure in building our model.

2.3.3 Percolation theory

We have discussed already that the field of network science explores phenomena of connectivity between discrete entities, focusing on structure, behaviour and function. The level of connectivity can affect how the network functions. Depending on the type of network, to be functional it may be preferable to have low levels of connectivity, such as in a contact network during a respiratory pandemic, or high connectivity, if instead we have a social network where it is important to spread information quickly

and extensively (such as the safety and availability of vaccines during a respiratory pandemic).

Suppose we have a set of isolated nodes. If we are allowed to add edges between nodes, it is clear that the network begins in a state where it has very few edges, where if a fluid or something similar (such as current, information, infections or memes) was allowed to flow between nodes along edges in the network it would not get very far. With continued addition of edges, it ends in a state of high connectivity, where the network could have sufficiently many edges that fluid could flow between any two nodes. Depending on how we add edges, in many cases, the network moves from a disconnected state to a connected state quickly, within the space of a small number of addition events, rather than steadily (such as a linear increase).



(a) (b) Square lattice (c) Random lattice

Figure 2.3: (a) Random percolation on square and random lattices, with example lattices shown in (b) and (c) for 64 nodes (the number of tiles on our example checker-board). In both simulations shown in (a), lattices of $25 \times 64 = 1600$ nodes are used, and an average is taken over 50 realisations of the random percolation experiment. In (a), solid lines represent the mean size of the GCC. Shaded regions are within one standard deviation of the mean, and the vertical dashed lines denote the theoretical locations of the respective percolation thresholds.

To demonstrate this, we consider a simple example. We take a checker-board and a box of matches. Each tile of the board, regardless of colour is considered a node. We decide that edges can exist between adjacent tiles, and the case where all possible edges exist in this way is called a square lattice network, shown in Figure 2.3 (b), where the edges are shown as the blue dashed lines, the nodes as black dots, and the tiles are divided by red lines. The tiles now represent a set of isolated nodes. We take a box of matches, and begin to lay them one by one uniformly at random across the lines dividing adjacent tiles on the checker-board, adding edges to the network. At the start of our exercise, we see mostly isolated tiles, and then small connected clusters

or paths begin to appear. Eventually, one large cluster of connected tiles spans the board from one side to another, such that a path exists through the cluster from one side of the board to the opposite side, with a small number of other disconnected clusters or isolated tiles.

We call this large cluster the giant connected component (GCC) and is sometimes referred to as the spanning cluster. If we plot the size of whichever component is the largest at each step, divided by the number of tiles on the board (the fractional largest component), against the number of matches lain, divided by the number of possible edges that can form (the edge occupancy), shown as the red curve in Figure 2.3 (a), we see a sudden increase around when the fraction of matches laid equals one half, from around 0.4 to 0.6. When this occurs, we also observe a path from one side of the board to the other emerging.

This suggests that there is a transition between the disconnected and connected states. We define the edge occupation density at which this transition happens (the emergence of the GCC) as the percolation threshold, p_c. This is a value that we may want to quantify, especially if we want to maximize the degree of connectivity in a network while minimizing the number of edges, such as in physical internet networks, where optical fibre is expensive and global connectivity is required. For some specific networks, p_c is known exactly, such as the square lattice, for which $p_c = 0.5$. For experiments on finite networks, the percolation transition can be estimated by averaging over many random realisations of the network model. This is also the point where, if we remove edges from a network, the connectedness, and in some cases the functionality, begins to break down. We refer to this process where, with the addition or removal of edges, the size of the largest component of a network changes quickly, around a transition point, as a percolation process. Percolation theory is related to the theory of phase-transitions in statistical mechanics and condensed matter physics, and the density of occupied edges can be considered the order parameter for the percolation process, the quantity that controls the connectivity, as in the Ising model for magnetization. Percolation processes also occur due to the removal of nodes from a network, or addition of nodes with a rule for the formation of edges, both of which are called node percolation.

The type of percolation we have described here is called edge or bond percolation, and we have focussed on percolation involving the addition of edges. We note that the addition of edges requires a rule to determine what edges can be added between nodes (in this case the rule was that edges could only form between nodes that lie beside one another on the board). We can generalise this process to much less restrictive

forms of choosing random edges, which will allow us to build any kind of network, in which case p_c will depend on the network topology, and particularly mean degree.

To illustrate how the percolation process changes with topology, we consider bond percolation in a lattice where sites are distributed uniformly at random on the $x - y$ plane, and the define cells or tiles by way of a Voronoi tessellation. In a Voronoi tessellation, cells for each site are defined such that all points within a cell are closer to the site of the cell than to any other site, using the Euclidean distance metric (or L_2 norm). In this case the graph of edges is defined by the Delaunay triangulation of the tessellation. To calculate the Delaunay triangulation, we take each pair of sites whose Voronoi cells touch, and for each pair, draw a line from one site to the other. The set of these lines forms the triangulation.

We will use this structure, along with the square lattice, throughout this work, and it will be referred to as the 'Voronoi tessellation' or 'random lattice'.[4] An example of the random lattice structure can be seen in Figure 2.3 (c). Cells (nodes), which in our model will correspond to catchment basins, are distinguished by solid red lines. The edges that are allowed to exist, as per the Delaunay triangulation, are shown dashed in blue. In panel (a), we see a comparison between the same percolation process on both a square lattice and the random lattice. The expected degree of a node in the random lattice is ~ 5.75 compared to ~ 4.0 for the square lattice (if we neglect domain-boundary effects). As edges are added to a network, the probability of an isolated node becoming connected to a connected component rises with the average degree of the nodes. Due to the higher average degree that nodes in the random lattice can attain compared to the square lattice, percolation occurs much earlier, with $p_c \approx 0.33$, seen clearly in the plot.

In this thesis, we will consider finite, spatial networks with polygonal boundaries, such as those shown in Figure 2.3 (b) and (c). Given that these networks have clearly defined sides, we will calculate the percolation threshold p_c as the earliest event (change to the edges in the graph) at which a cluster appears that spans from one side of the network to the other as edges are added. This is an appropriate and reliable condition for percolation in finite networks, as we can think of the graphs of finite networks as being subgraphs of the graphs of infinite networks (this is the main idea behind renormalization group methods). As we expect the GCC to form suddenly during percolation on an infinite graph, it must also do so in every finite

[4]Both structures are technically Voronoi tessellations, and both can be considered to be lattices, if we consider a lattice to be a structure of individual cells in a plane, arranged according to a particular rule, and assigned 'neighbours' according to another rule. In both cases the rule to determine which cells neighbour any particular cell is that used to create a Voronoi diagram.

subgraph (in this way we call the GCC self-similar, at percolation in all subgraphs). As we expect tree-like and spanning behaviour of the GCC at percolation, once the GCC forms, it will trace a path from one side of a rectangular subgraph of an infinite graph to the opposite side.

2.3.4 Alternative processes and explosive percolation

We have demonstrated in Figure 2.3 that differences in ways in which the edges to be added to the network are chosen can lead to large changes in the percolation process, especially the location of p_c. Further examples of percolation on random networks are covered in Newman [2018], including percolation on the Watts-Strogatz model, Erdős-Rényi graphs and the configuration model (where edges are wired at random between nodes but every node has a prescribed degree), and for these processes some analytical theory also exists.

More pertinent to this work is a set of models that deviate from a random choice of bonds to add or remove from the network, and instead show 'preference' at each step for the addition of one of a pair of edges chosen at random. This process was first suggested by Achlioptas et al. [2009]. This family of processes have become known as anomalous or explosive percolation, due to the propensity for percolation to happen with an apparent discontinuity (there has been much written on whether the jump seen in these processes is truly discontinuous [Riordan and Warnke, 2011]).

To illustrate how the processes that lead to conventional percolation differ from those that lead to anomalous percolation we consider the ER model. In 'conventional' percolation on the ER model, edges are added at random according to a Bernoulli process between a set of isolated nodes with a defined probability, and a smooth percolation transition occurs. A delayed and non-smooth transition can occur if we use a different addition rule or 'edge rule'. At every step, two random edges $\{e_1, e_2\}$ are chosen independently of one another. One is then added to the network according to a rule, and the other is discarded. We then choose another pair of possible edges, and so on. The rule introduced by Achlioptas et al. [2009] was to decide between e_1 and e_2, based on the sizes of the components that the nodes at the ends of the two edges chosen in the step, (e_1, e_2), belong to.

The authors found that when the rule minimises the size of the new component, this leads to suppression of the GCC initially, compared to percolation by the ER model. This edge rule instead leads to the emergence of several large clusters, none of which span the network. This is referred to as a 'powder keg', as once enough edges are added to the system, these clusters all join within a few steps and the

GCC appears almost fully formed. The authors found that other edge rules could be formulated that could hasten percolation. A variety of edge rules have been explored since [D'Souza and Nagler, 2015], and a huge variety of possible systems abounds. This is an idea that will be revisited in Section 7.6.

2.3.5 Fractals and ponds

The features that we examine in the geosciences, such as clouds, mountains or lakes, are often inadequately described by the straight lines and smooth curves of classical geometry. To accommodate some of the inherent complexity in geophysical features we can instead employ ideas from fractal geometry and dimension. The fundamental idea of fractal geometry is that of fractal dimension, whereby the inherent scaling exponent of a measure of the geometry is not an integer, as it would be for familiar shapes such as the circle or square in $\mathbb{R}^2$ (where for area a, and diameter d, $a \sim d^2$). In this section we give an introduction to fractal dimension, as we will use measurements of fractal dimension as a benchmark test of our model against observations in Chapter 7. Here, we present sufficient material such that the reader can understand the measurements and analysis in the section that covers the fractal dimension of synthetic ponds in that chapter, including a definition of fractal dimension, how it is calculated, and a worked example of this. A more rigorous introduction than the one presented here is given in Appendix C.

To measure the dimension of a set, we can compute the box-counting dimension, (an approximation of Hausdorff dimension, defined in the Appendix). The box counting measure $N_\delta(F)$ is a count of the number of square boxes of diameter δ it takes to completely cover a set F. Motivated by the definition of Hausdorff dimension given in Appendix C, assume that there is a power law relation between the box counting measure and box size

$$N_\delta(F) \sim c\delta^{-s}, \tag{2.12}$$

where c is a constant and s is the dimension of the set. Rearranging this power law relationship and taking the limit $\delta \to 0$, we can derive the box-counting dimension s as

$$s = \lim_{\delta \to 0} \frac{\log N_\delta(F)}{-\log \delta}. \tag{2.13}$$

We then define the lower box counting dimension

$$\underline{\dim}_B F = \underline{\lim}_{\delta \to 0} \frac{\log N_\delta(F)}{-\log \delta} \tag{2.14}$$

and upper box counting dimension

$$\overline{\dim}_B F = \overline{\lim}_{\delta \to 0} \frac{\log N_\delta(F)}{-\log \delta}, \tag{2.15}$$

where the underbar and overbar denote the limit infimum and limit supremum respectively. If $\underline{\dim}_B F = \overline{\dim}_B F$ we call this common value the box counting dimension. This is the more commonly seen definition of box counting, or fractal dimension,

$$D(F) = \dim_B F := \lim_{\delta \to 0} \frac{\log N_\delta(F)}{-\log \delta}. \tag{2.16}$$

To illustrate a set with non-integer dimension, we consider the Koch snowflake. The snowflake of order n is built iteratively, starting with an equilateral triangle. At each iteration, we take each straight section of the curve (the boundary of the snowflake), divide it in three, delete the middle third of the section and build a new equilateral triangle on this piece (without the deleted length), as shown in Figure (2.4). We note that newly built equilateral triangle on each section has side length equal to the piece deleted.

Figure 2.4: The 0^{th} to 3^{rd} order Koch snowflakes. Taken from Mandelbrot [1983].

To calculate the box-counting dimension of the curve, we consider a side of the curve at order n. If we rescale so that the line section of order $n-1$ has length 1, and take δ to be one third of the segment, we measure 4 of these pieces of length $1/3$ at order n along the line segment. According to equation (2.13),

$$s = \frac{\log 4}{-\log 1/3} \approx 1.26.$$

By construction, this holds at all orders. In the Koch snowflake, as we let $n \to \infty$, we observe self similarity at different scales and fine structure or roughness at arbitrarily small scales. These are common characteristics of fractal sets.

As mentioned above, the perimeters of coastlines and islands have fractal geometries. The same has been found to be true for lakes, the converse of an island. In practice, measuring the fractal dimension of perimeters of many such lakes or islands can be costly. Mandelbrot [1983] proposed an easier method than box counting for

this special case, making use of dimensional analysis arguments. The various measures of shapes, perimeter, area, volume, are known to scale consistently with the linear diameter d as $d \propto \text{perimeter}^1$, $d \propto \text{area}^{1/2}$, $d \propto \text{volume}^{1/3}$, (defined in equation (C.7) when using Hausdorff measure). For example, if we take a cube of side d, its volume is d^3, the area of one face is d^2 and the perimeter of such a face is $4d$. These hold for all d, as do the ratios of measures. If we consider then a shape with a fractal perimeter p and area a, such as an island or lake, for two self similar lakes,

$$\frac{p_1(\delta)^{1/D}}{a_1(\delta)^{1/2}} = \frac{p_2(\delta)^{1/D}}{a_2(\delta)^{1/2}} = \text{const}, \tag{2.17}$$

where subscripts denote different lakes. It can be shown this relation is also independent of the resolution at which we measure perimeter, i.e., for the same pond,

$$\frac{p(\delta_1)^{1/D}}{p(\delta_1)^{1/2}} = \frac{p(\delta_2)^{1/D}}{a(\delta_2)^{1/2}} = \text{const}, \tag{2.18}$$

where the subscripts denote different measurement scales. This allows us to write

$$p = ka^{D/2}, \tag{2.19}$$

with k a constant, which can be rearranged to

$$\log p = k + \frac{D}{2} \log a. \tag{2.20}$$

Given data on the areas and perimeter of many lakes at a given resolution, we can then estimate D by plotting $\log p$ against $\log a$ and finding the slope of the regression line.

Hohenegger et al. [2012] examined the areas and perimeters of a large number of ponds with regard to self-similarity and calculated a fractal dimension for pond perimeters using the method based on equation (2.19) across ponds at various stages of development. As mentioned earlier, the authors found that ponds in the early stages of development had simple, smooth boundaries, with perimeters that had fractal dimensions close to 1. The authors also found that the highly ramified clusters of ponds in the later stages of development were self similar and had perimeters that approached space-filling curves, with fractal dimensions close to 2. The authors proposed from the data that these scalings changed at an area scale of around 10^2 m^2.

The behaviour highlighted by Hohenegger et al. [2012] was replicated in analysis by Huang et al. [2016] (who also collated their own data from fieldwork) and has

since become a useful benchmark for models of pond geometry and behaviour, as was used by Popović et al. [2018], Scagliarini and Calzavarini [2018] and Horvat et al. [2020]. We will do the same with our model in Chapter 7. Hohenegger et al. [2012] suggested that the change in scaling implies that two different processes dominate behaviour in either regime, and that a change in the scales of the power-law relations and self-similarity behaviour point towards a percolation process. The structure of our model, a temporal network, will allow us to look at the behaviour of pond systems with different levels of connectivity and can be used to examine whether the change in D is associated with the hypothetical percolation phenomenon in melt pond growth.

2.4 Summary

In this chapter we have outlined the important literature on sea ice and melt ponds. We have given an indication as to what data is available, and what modelling work has been done, and summarised some of the most important findings from these models, especially those results that we will make use of, and models that we will build upon. Literature and theory for the thermodynamics of pond formation and evolution was discussed, and a rationale for parametrizing the melt rates in our model as a pair of constant melt rates was provided. An introduction to network science and the mathematical structures used in the study of networks was given. So too were introductions into dynamical systems on networks, percolation theory, and fractal dimension, as well as how fractal dimension is relevant to the geometry of melt ponds.

Chapter 3

A model for a single pond

3.1 Introduction

In this chapter we describe a model for the growth of a pond in a single catchment area of a floe of impermeable ice. This model will form the basic behaviour for growth and shrinkage of ponds and will form the activity on a node in the network model. It will later be extended to incorporate interactions with neighbouring catchments in Chapter 4 and porous drainage in Chapter 6. We develop the model in its most general form, then explore its behaviour in one and then two dimensions for the cases of axisymmetric and square-pyramidal catchments. We also discuss the dependence of the model on parameters. For clarity, we refer to three dimensional space as being the two dimensional case, as we are making reference to the dimension of the pond surface.

3.2 Mass balance model

3.2.1 Model derivation

As described earlier, sea ice floes can be treated as thin pieces of ice, floating on the sea surface. Due to the accumulation of snow and sculpting by wind and dynamic ice processes, we assume that a floe has an irregular topography, which we break up into individual catchments, in a similar way to how land surfaces can be divided into watersheds. In this chapter we consider one catchment of area a_0. The catchment has an initial topography $h(\mathbf{x}, 0) = h_0(\mathbf{x})$ at the start of the season, which is a smooth surface with only one local minimum, as can be seen in Figure 3.1. Note we are using bold face for vectors. We set up our coordinate system such that the x and y coordinates of the origin lie at the x and y coordinates of the local minimum of the

ice surface in the catchment. We place the z coordinate of the origin at a fixed point in the ice below the local minimum of the surface.

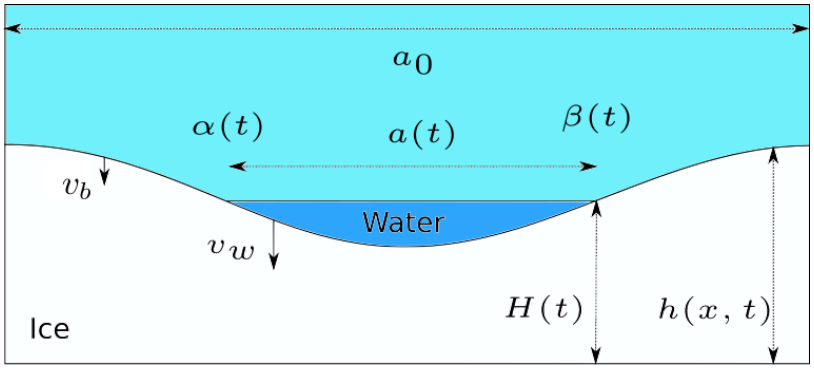


Figure 3.1: Schematic of the model for a single pond during the early stages of evolution, showing one whole catchment of area a_0. Water from the melting of bare ice settles in the local minimum (centre of figure) to a height $H(t)$, yielding a pond of area $a(t)$. The ice melts downwards at a rate v_b for bare ice and v_w for ice under water. The pond boundaries are denoted by α and β in one dimension. In two dimensions the boundaries become a curve rather than two points.

Melt-water forms and collects near the local minimum due to (i) melting of bare ice, with the liberated water flowing from areas of higher elevation to lower elevation, and (ii) melting of ice under pond water. We neglect snow cover on the ice, assuming it to be bare, and that all melt water comes from melting ice. This is a major simplification in comparison to other pond models, but is done with the goal of deriving a simple model for node behaviour. Melt water is assumed to be completely fresh, as sea ice rejects salt during the freezing process, and we neglect any residual salinity of the sea ice. Melt-water pools initially at the local minimum and fills the topography upwards to the water surface height $H(t)$ and has an area $a(t)$ (both to be determined). We assume for now that there is no seepage loss through the ice, or flow of melt-water into the sea and assume that the flow of melt-water downhill into the pond can be treated as instantaneous. This last assumption is made because the time scale for the flow of water within a catchment is much smaller than the time scale for melting. We further assume that bare ice melts downwards at a rate $v_b(t)$ and that ice beneath melt water melts at a rate $v_w(t)$ so that

$$\frac{\partial h}{\partial t}(\mathbf{x},t) = \begin{cases} -v_b(t), & h(\mathbf{x},t) \geq H(t) \\ -v_w(t), & h(\mathbf{x},t) < H(t), \end{cases} \tag{3.1}$$

where we have assumed only downwards melting as the ice topography is considered to be shallow. For now, we will only consider melt rates that are constant in time.

We distinguish between the melted volume of a catchment area and the volume that accumulates, because these will not be equal once ponds start to join (covered in the next chapter). The total volume of melt-water $V_m(t)$ in the catchment evolves according to

$$\rho_w \frac{\mathrm{d}V_m}{\mathrm{d}t} = \rho_i \iint_{a_0} -\frac{\partial h(\mathbf{x},t)}{\partial t}\,\mathrm{d}A, \tag{3.2}$$

where ρ_w and ρ_i are the densities of water and ice respectively. We assume that all of the melt water that forms in an isolated catchment area accumulates with a volume $V_A(t)$ around the local minimum to the height $H(t)$ such that

$$V_A(t) = \iint_{\Omega(t)} [H(t) - h(\mathbf{x},t)]\,\mathrm{d}A, \tag{3.3}$$

where $\Omega(t)$ is the pond region, defined by $\Omega(t) = \{\mathbf{x}\colon h(\mathbf{x},t) \leq H(t)\}$. By conservation of mass, neglecting flow of water between neighbouring catchments,

$$V_m(t) = V_A(t) \tag{3.4}$$

which we refer to as the filling equation.

It is useful to define the pond boundary $\partial\Omega(t)$ which is the projection of the pond boundary, the level set

$$h(\mathbf{x},t) = H(t) \tag{3.5}$$

onto the x–y plane. In one dimension the pond boundaries are denoted by the points $x = \alpha$ and $x = \beta$, as shown in Figure 3.1. We set an initial condition of $V_m(0) = V_A(0) = 0$.

3.2.2 Non-dimensionalization

To non-dimensionalize equations (3.2)–(3.5), we choose appropriate scales for the variables based on the topography and melt rates, based on the literature and my own observations taken while on fieldwork in the Beaufort Sea as part of the Healy 1902 cruise, as described in Chapter 1. We assume that the catchment area of a pond has characteristic lateral length scale $\tilde{x}$, such that $a_0 \sim \tilde{x}^D$, where D is the dimension ($D = 1$ or 2 when working in one or two dimensions). A typical value is $\tilde{x} \approx 5$ m, from measurements taken during fieldwork. This compares favourably with data from Landy and Ehn [2014]. We choose a thickness scale $\tilde{h}$. It is also useful to define the non-dimensional density ratio

$$\rho_{iw} = \frac{\rho_i}{\rho_w} \tag{3.6}$$

which for pure ice and fresh water takes the value $\rho_{iw} \approx 0.9$.

We then substitute into (3.2)–(3.5) the relations

$$\mathbf{x} = \tilde{x}\hat{\mathbf{x}} \qquad a_0 = \tilde{x}^D \hat{a}_0 \qquad h = \tilde{h}\hat{h} \qquad t = \frac{\tilde{h}}{\tilde{v}}\hat{t}$$
$$v_b = \tilde{v}\hat{v}_b \qquad v_w = \tilde{v}\hat{v}_w \qquad V = \tilde{x}^D \tilde{h}\hat{V},$$

where $\tilde{v}$ is a typical melt rate, and then drop the hats with the understanding that all variables and parameters are now non-dimensional (unless stated otherwise). The non-dimensional system arises:

$$\frac{\mathrm{d}V_m}{\mathrm{d}t} = \rho_{iw} \iint_{a_0} -\frac{\partial h}{\partial t}\,\mathrm{d}A, \tag{3.7}$$

$$\frac{\partial h}{\partial t}(\mathbf{x}, t) = \begin{cases} -v_b, & h \geq H \\ -v_w, & h < H \end{cases} \tag{3.8}$$

and

$$V_A = \iint_{\Omega} H - h\,\mathrm{d}A. \tag{3.9}$$

The dimensionless initial topography is taken to have dimensionless amplitude Δh, which in dimensional form can vary on the range of 10 cm to 30 cm depending on the age of the floe. Until Chapter 6, where we have to account for the finite thickness of ice, we can choose $\tilde{h}$ such that $\Delta h = 1$, since the amplitude is the natural vertical length scale, though in some cases it is more natural to choose $\tilde{h}$ to scale h such that the average ice thickness $\bar{h} = 1$. In either case, the vertical scale is not important until porous drainage is introduced and the finite thickness of floes needs to be accounted for. Until then, the vertical scale has little meaning, and given that it also depends on the thickness scale, neither does the time scale. In Chapters 6 and 7, when the finite thickness of ice becomes important, it is natural to use $\bar{h}$, which is typically between 1 and 1.5 m, for $\tilde{h}$, and closer attention must be paid to both the vertical scale and the time scale.

We will take the non-dimensional melt rate $v_b = 1$ (by choice of $\tilde{v}$) for the remainder of the thesis, but will keep it in the equations to aid understanding of the physics. For what follows in the remainder of the text, subscripts t and x denote partial differentiation with respect to the relevant variable, and an over-dot indicates an ordinary time derivative.

3.3 Evolution in one dimension

3.3.1 The half-pond

We now consider the one-dimensional case where $h_0(x)$ corresponds to a channel-like geometry extending into the page as in Figure 3.1. Such one-dimensional geometries will be used in this chapter and Chapter 4 to develop the model and examine specific aspects of it in a simplified setting. If we assume that the catchment area is symmetric we need only consider the problem for the right half-pond within a half-catchment, with domain $0 \leq x \leq a_0$, such that this catchment is one half of that shown in Figure 3.1, as if there were a wall at $x = 0$. The boundary contour $\partial\Omega(t)$ reduces to a point which we denote as $x = \beta(t)$, the right-hand boundary.

In this geometry the area integral (3.7) becomes a line integral that can be integrated directly by substitution of (3.8) to give

$$\dot{V}_m = \rho_{iw} v_b (a_0 - \beta) + \rho_{iw} v_w \beta. \tag{3.10}$$

Taking the time derivative of (3.9) gives

$$\dot{V}_A = \int_0^\beta \dot{H}(t) - h_t(x,t)\,\mathrm{d}s = (\dot{H} + v_w)\beta, \tag{3.11}$$

having noted that the derivative of the boundary term of the integral disappears as $h(x,t) = H(t)$ at $x = \beta(t)$, assuming continuity of h. The time derivative of (3.5) gives

$$\dot{H} = h_x(\beta,t)\dot{\beta} + h_t(\beta,t). \tag{3.12}$$

Care must be taken when evaluating this expression as h_t is discontinuous across $x = \beta$. If we take the limit from the right of (3.12),

$$\dot{H} = h_x^+(\beta,t)\dot{\beta} - v_b \tag{3.13}$$

and if we take the limit from the left,

$$\dot{H} = h_x^-(\beta,t)\dot{\beta} - v_w, \tag{3.14}$$

as the region to the right of β is bare ice and the region to the left is water-covered. We also note that while the pond is filling the slope of the topography to the right of the boundary is equal to its initial slope,

$$h_x^+(\beta,t) = h_{0x}(\beta). \tag{3.15}$$

Given that $H(t)$ is single-valued, we can equate (3.13) and (3.14) to obtain

$$h_x^-(\beta, t) = h_{0x}(\beta) - \frac{v_b - v_w}{\dot{\beta}} \tag{3.16}$$

which can be written as

$$[h_x(\beta, t)]_-^+ = -\frac{v_w - v_b}{\dot{\beta}}. \tag{3.17}$$

For $v_w \neq v_b$, this indicates that the gradient of the ice profile becomes discontinuous at the boundary. It also indicates that the gradient at the pond edge becomes infinite if $\dot{\beta} \to 0$, discussed further below. Equation (3.13) describes how increases in water level cause motion of the pond boundary. This motion is an interplay between changes in water level travelling up along the slope of the ice as the pond fills, and the translation downwards of the bare ice melting at the boundary also.

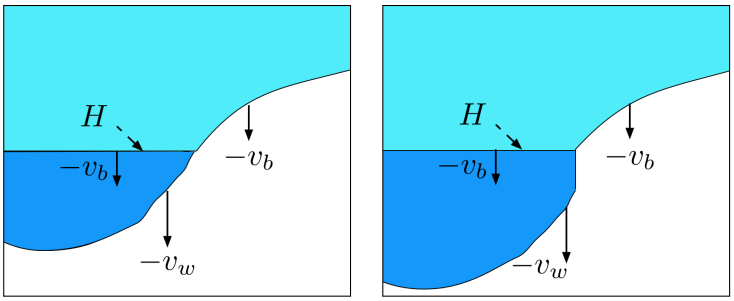
Equating the rate of melt-water production (3.10) with accumulation (3.11), and using (3.13) to eliminate $\dot{H}$, we can write an equation for the evolution of the pond boundary,

$$\dot{\beta} = \frac{\rho_{iw} v_b a_0 - (1 - \rho_{iw})(v_w - v_b)\beta}{\beta h_{0x}(\beta)}. \tag{3.18}$$

Equation (3.18) describes how the outward motion of the boundary is aided by influx of melt-water from the catchment area (the first term in the numerator), hindered by the contraction that occurs as ice melts beneath the pond (the second term in the numerator), and also affected by the steepness of the ice surface (the denominator). Intuitively, a steeper ice hollow will fill upwards faster than it fills outwards, reflected in equation (3.18).

3.3.2 The profile discontinuity at the boundary

So far we have assumed continuity in $h(x, t)$. But we have also found that the ice surface slope h_x becomes discontinuous, as we have introduced a discontinuous quantity into the system by our choice of different melt rates for bare ice and water-covered ice. Moreover, if $\dot{\beta} = 0$, the discontinuous melt rate can also generate a discontinuity in $h(x, t)$ itself. In that case, given the physics included in the model, the ice at the pond side of β would melt and the surface would move downwards at a faster rate than ice at the other side, which would cause a discontinuity in the profile - a cliff - to form. This is illustrated in Figure 3.2 We return to this in Chapter 4 Section 4.2.1, where we discuss how this happens when a pond begins to overflow from one catchment to another, and pond growth stalls.



(a) (b)

Figure 3.2: Schematic diagrams of a discontinuity forming in the slope of the pond floor. (a) The edge of a pond when the expansion of the pond stops ($\dot{\beta} \to 0$). (b) Some time after pond growth stalls ($\dot{\beta} = 0$). Note the cliff that forms at the pond boundary, due to the two different melt rates for ice inside and outside of the pond.

3.3.3 Behaviour at early time and initial conditions for the model

Examining equation (3.18) around the onset of melt, we find that $\dot{\beta}$ blows up to infinity as $\beta \to 0$, so the equation is singular here. To initiate the numerical solution, we look for an asymptotic solution for early time. We expand (3.18) for small β about the local minimum of the pond with $h_{0x}(0) = 0$, and $h_{0xx}(0) > 0$. This gives the approximation

$$\beta^2 h_{0xx}(0)\dot{\beta} = \rho_{iw} v_b a_0. \tag{3.19}$$

Integrating (3.19) gives the short time dependence for growth of

$$\beta \sim \left(\frac{3\rho_{iw} v_b a_0 t}{h_{0xx}(0)} \right)^{\frac{1}{3}}. \tag{3.20}$$

To initiate numerical solutions we use (3.20) to calculate $\beta(t_0)$ for some small $t_0 > 0$. In practice, $t_0 = 10^{-4}$ is large enough to avoid the singularity whilst maintaining β small enough for the linearisation to hold.

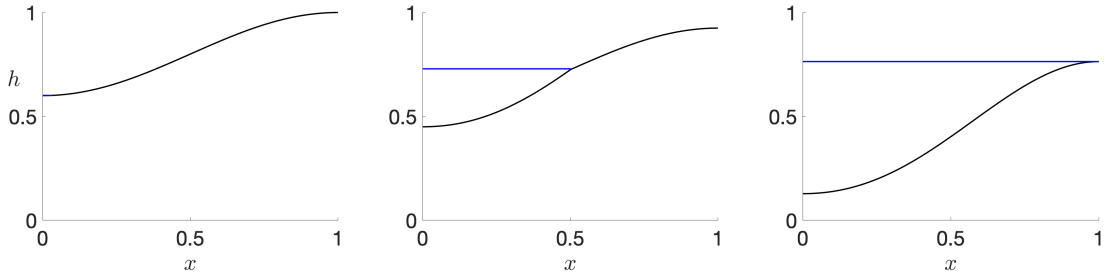
We now have enough information to run a simple numerical simulation of pond growth. Figure 3.3 shows renderings of the topography at several times from a numerical simulation of (3.18), and using the early time solution (3.20), in a half-catchment on a topography defined by

$$h_0(x) = \bar{h} - \Delta h \cos(\pi x). \tag{3.21}$$

In this case $\bar{h} = 0.8$ and $\Delta h = 0.2$. The topography was discretized into 200 points, and the heights of these points were calculated by solving equation (3.8), which allows

us to visualise the topography through time. The discontinuity of the slope of the topography due to the jump condition can be seen clearly in panel (b). In later time, once the catchment has flooded (panel (c)) we see that the centre of the pond has deepened significantly compared to the shore, though for this simulation, the shape of the ice surface at flooding remains similar to the initial topography.

Examining the behaviour of (3.20), using (3.21) we estimate that the area of a pond will evolve with a rate like $v_b a_0/\Delta h$, which suggests that steeper catchment basins will take longer to flood with melt-water than shallower ones, and that increasing the characteristic melt rate will cause the basin to flood faster.



(a) $t = 10^{-4}$ (b) $t = 0.075$ (c) $t = 0.235$

Figure 3.3: Plots of ice topography (black) and water level (blue) for a numerical simulation of the model in one dimension, equation (3.18), for a cosine shaped half-catchment during melting. Melt rates of $v_b = 1$ and $v_w = 2$ and a density ratio of $\rho_{iw} = 0.9$ were used.

In numerical simulations of the model throughout this thesis we use MATLAB's ODE15s to integrate the model equations. This function uses an adaptive time-step, but we set it to output time series results at a resolution of 10^{-4}, consistent with the t_0 used for the initial condition.

3.3.4 Steady states and stability

A steady state of equation (3.18) exists for $\beta = (\rho_{iw} v_b a_0)/[(v_w - v_b)(1 - \rho_{iw})]$, provided $\beta < a_0$. This represents a point at which the pond would stop widening and only deepen. To investigate the stability of this state we denote the location of the boundary as β_∞ and apply a small perturbation $\tilde{\beta}$ from the stationary state such that $\beta = \beta_\infty + \tilde{\beta}$. Taylor expanding (3.18) gives

$$\dot{\tilde{\beta}} = -\frac{\beta_\infty (1 - \rho_{iw})(v_w - v_b)}{\beta_\infty h_{0x}(\beta_\infty)} \tilde{\beta}, \tag{3.22}$$

and since the coefficient of $\tilde{\beta}$ on the RHS is negative, the stationary state $\beta = \beta_\infty$ is stable.

It is worth examining the location of the stationary state more carefully. Note that in the non-dimensional form of the model $\beta \leq a_0 = 1$, so for β_∞ to be in the relevant domain, we need

$$\frac{\rho_{iw}}{1-\rho_{iw}} \frac{v_b}{v_w - v_b} \leq 1 \tag{3.23}$$

or equivalently,

$$\frac{v_w}{v_b} \geq \frac{1}{1-\rho_{iw}}. \tag{3.24}$$

Typical values of 1000 kg m^{-3} and 916 kg m^{-3} for the densities of water and ice respectively gives

$$\frac{v_w}{v_b} \gtrsim 12, \tag{3.25}$$

so the melt rate would need to be at least an order of magnitude in difference, making the steady state unlikely to be relevant.

As mentioned in Section 2.2, melt rates may vary throughout the season and are typically of the order of 10^{-2} m/day [Fetterer and Untersteiner, 1998], but we do not expect the condition (3.25) to be met, except in the most extreme cases. However, we will see in Chapter 6 that the model for permeable ice behaves quite differently, and that this particular steady state only holds during the impermeable phase of the melt season.

3.3.5 Analytical solutions in one dimension

Analytical solutions to (3.18) exist for some very simple geometries, such as a notch, where $h_0(x) = |x|$, or a parabola, where $h_0(x) = x^2/2$. In both cases we write $-1 \leq x \leq 1$ to define the extent of the catchment in both cases, which gives $a_0 = 2$. If we write $c_1 = \rho_{iw} v_b a_0$ and $c_2 = (1-\rho_{iw})(v_w - v_b)$, the boundary evolution equation for a notch becomes

$$\dot{\beta} = \frac{c_1}{\beta} - c_2. \tag{3.26}$$

Likewise, for the parabola,

$$\dot{\beta} = \frac{c_1}{\beta^2} - \frac{c_2}{\beta}. \tag{3.27}$$

Both equations can be integrated with the initial condition $\beta(0) = 0$. The notch geometry then evolves according to the implicit expression

$$c_2 t = -\beta + \frac{c_1}{c_2} \log\left(\frac{c_1}{c_1 - c_2\beta}\right) \tag{3.28}$$

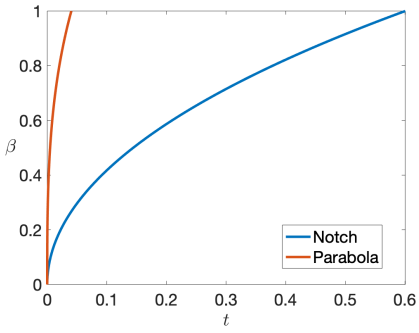


Figure 3.4: Plots of the evolution of the pond boundary in one dimension for, in blue, a pond growing in a notch shaped profile, according to equation (3.28) and in red, a pond growing in a parabolic profile, according to equation (3.29) Values of $v_b = 1$ and $v_w = 2$ were used.

and the parabola evolves according to

$$c_2 t = -\frac{\beta^2}{2} - \frac{c_1 \beta}{c_2} + \left(\frac{c_1}{c_2}\right)^2 \log\left(\frac{c_1}{c_1 - c_2\beta}\right). \tag{3.29}$$

Both analytic solutions provide some insight into the behaviour of pond growth, and we can use them to find the time to flooding of the catchment area by substituting $\beta = a_0$. Comparatively, the pond with the parabolic profile grows faster while the pond is still small, as the gradient of the pond is low around the minimum, as can be seen in Figure 3.4. Neither profile flattens towards the top, when the catchment area would flood, hence we see that the boundary tends to 1 steadily in both cases.

3.3.6 Asymmetric ponds in one dimension

We can derive growth equations for a pond in an asymmetric domain in much the same way as in Section 3.3.1. Now, we consider the full catchment area in one dimension to be of (unit-less) length a_0, with a continuous initial ice height profile $h_0(x)$ that has a minimum near the middle, where the pond will initially grow. Once the pond forms, we define its left boundary as $\alpha(t)$ and its right boundary as $\beta(t)$ as in Figure 3.1. In this geometry, conservation of mass (3.7) is a line integral that can be integrated directly by substitution of the melt rates (3.8) to give

$$\dot{V}_m = \rho_{iw} v_b [a_0 - (\beta - \alpha)] + \rho_{iw} v_w (\beta - \alpha). \tag{3.30}$$

The time derivative of the filling equation (3.9) gives

$$\dot{V}_A = \int_\alpha^\beta [\dot{H} - h_t(x,t)]\,\mathrm{d}x = (\beta - \alpha)(\dot{H} + v_w), \tag{3.31}$$

noting that the derivatives of both boundary terms of the integral disappear as $h(x,t) = H(t)$ at $x = \alpha(t)$ and at $x = \beta(t)$. Equating (3.30) and (3.31),

$$\dot{H} = \frac{\rho_{iw} v_b a_0 - [v_w - \rho_{iw}(v_w - v_b)]a}{a}, \tag{3.32}$$

where the pond area

$$a = \beta - \alpha. \tag{3.33}$$

The time derivative of (3.5) at each boundary gives, respectively,

$$\dot{H} = h_x(\alpha, t)\dot{\alpha} + h_t(\alpha, t) \tag{3.34}$$

and

$$\dot{H} = h_x(\beta, t)\dot{\beta} + h_t(\beta, t). \tag{3.35}$$

As before, (3.17) holds at $x = \beta$, with a similar condition at $x = \alpha$. We evaluate (3.34) and (3.35) at the ice side of either boundary where the slope is unchanged from its initial value and rearrange to give

$$\dot{\alpha} = \frac{\dot{H} + v_b}{h_{0x}(\alpha)}, \qquad \dot{\beta}(t) = \frac{\dot{H} + v_b}{h_{0x}(\beta)}. \tag{3.36}$$

Using (3.32), this can be rewritten explicitly as

$$\dot{\alpha} = \frac{\rho_{iw} v_b a_0 - (1 - \rho_{iw})(v_w - v_b)a}{h_{0x}(\alpha)a} \tag{3.37}$$

and

$$\dot{\beta} = \frac{\rho_{iw} v_b a_0 - (1 - \rho_{iw})(v_w - v_b)a}{h_{0x}(\beta)a}. \tag{3.38}$$

Thus, the pond area defined in (3.33) evolves according to

$$\dot{a} = \frac{\rho_{iw} v_b a_0 - (1 - \rho_{iw})(v_w - v_b)a}{a} \left(\frac{1}{h_{0x}(\beta)} - \frac{1}{h_{0x}(\alpha)} \right). \tag{3.39}$$

These equations are used in Chapter 4, Section 4.5 where we consider the growth of several coupled ponds in one-dimensional irregular notch-shaped catchments.

3.4 Simple ponds in two dimensions

3.4.1 Derivation in two dimensions

To derive expressions for the rates of change of water levels and pond areas in two dimensions we consider a pond in a catchment of area a_0. We restrict the topography

such that the pond boundary $\partial\Omega(t)$ is a simple curve, requiring that there be no discontinuities in either $h_0(\mathbf{x})$ or $\nabla h_0(\mathbf{x})$.

By taking the time derivative of the accumulated volume (3.9) and using the Leibniz integral rule, an expression for the rate of pond filling arises in the form

$$\dot{V}_A = \frac{\mathrm{d}}{\mathrm{d}t}\iint_{\Omega(t)} H - h(\mathbf{x},t)\,\mathrm{d}A = \iint_{\Omega}\frac{\partial}{\partial t}[H - h(\mathbf{x},t)]\,\mathrm{d}A + \int_{\partial\Omega(t)}[H - h(\mathbf{x},t)]u_n\,\mathrm{d}s, \tag{3.40}$$

where u_n is the normal velocity of the moving boundary $\partial\Omega$. The second term on the right of (3.40) disappears due to the boundary condition $H = h(\mathbf{x},t)$ on $\partial\Omega$. Equating this with (3.7), and using the constant melt rates gives

$$\rho_{iw}v_b(a_0 - a) + \rho_{iw}v_w a = (\dot{H} + v_w)a, \tag{3.41}$$

where the pond area is defined by

$$a = \iint_{\Omega}\mathrm{d}A. \tag{3.42}$$

This can be rearranged as

$$\dot{H} = \frac{\rho_{iw}v_b a_0 - [v_w - \rho_{iw}(v_w - v_b)]a}{a}. \tag{3.43}$$

Equation (3.43) would still hold if v_b and v_w varied temporally, but would not if either melt rate varied spatially. This equation can be further developed as a statement for the evolution of the moving boundary $\partial\Omega$ by differentiating (3.5) with respect to time. Evaluating the partial derivative of h_0 at the ice side of the boundary, where the surface slope does not change over time, yields

$$\dot{H} = u_n\mathbf{n}\cdot\nabla h_0 - v_b \tag{3.44}$$

so

$$u_n = \frac{\rho_{iw}v_b a_0 - (1-\rho_{iw})(v_w - v_b)a}{a}\frac{1}{\mathbf{n}\cdot\nabla h_0}. \tag{3.45}$$

Hence,

$$\dot{a} = \int_{\partial\Omega} u_n\,\mathrm{d}s = \frac{\rho_{iw}a_0 v_b - (1-\rho_{iw})(v_w - v_b)a}{a}\int_{\partial\Omega}\frac{\mathrm{d}s}{|\nabla h_0|}, \tag{3.46}$$

where s parametrizes arc length around $\delta\Omega$, a level set of H. In the next two sections, we use equations (3.43) and (3.46) to explore the model. However, once it comes to using this model in a simulation of many catchments, we find that it is easier to solve

for the water levels and pond area in a catchment by integrating equation (3.43) and finding the area according to the hypsometry of the initial topography,

$$a(t) = f(H, t), \tag{3.47}$$

where f depends on the initial topography $h_0(\mathbf{x})$. Specifically, $f(H, t)$ gives the area enclosed by the contour

$$h_0(\mathbf{x}) = H + \int_0^t v_b \, \mathrm{d}t.$$

From the end of this chapter onwards, equations of the form of (3.43) and (3.47) will make up the model for the growth of a single pond on impermeable ice.

3.4.2 Circular ponds

The integral in equation (3.46) is straightforward to evaluate for profiles with axisymmetric or elliptical symmetries. We will now use these as simple profiles to help us to build understanding of the behaviour of the model. Parametrizing the curve $\partial\Omega$ which lies in the $x - y$ plane, with angle τ, $0 \leq \tau \leq 2\pi$, as

$$\mathbf{x} = (x, y) = (g_1(\tau, t), g_2(\tau, t)), \tag{3.48}$$

we can write the integral on the right hand side of (3.46) as

$$I = \int_0^{2\pi} \frac{1}{|\nabla h_0|} \left| \frac{\mathrm{d}\mathbf{x}}{\mathrm{d}\tau} \right| \mathrm{d}\tau. \tag{3.49}$$

If the ice topography is axisymmetric about a local minimum, then circular ponds develop. We consider a circular catchment, of area a_0, where $h_0(\mathbf{x}) = f(\sqrt{x^2 + y^2})$ and parametrize the boundary $\partial\Omega(t)$ by

$$\begin{aligned} x &= \gamma(t) \cos \tau, \\ y &= \gamma(t) \sin \tau, \end{aligned} \tag{3.50}$$

where $\gamma(t)$ is the radius of the circle that the curve encloses. The integral (3.49) then reduces to

$$I = \frac{2\pi\gamma}{f'(\gamma)}. \tag{3.51}$$

Substituting this expression for the integral into the area equation (3.46) and noting that $a = \pi\gamma^2$ gives

$$\dot{\gamma} = \frac{\rho_{iw} v_b a_0 - (1 - \rho_{iw})(v_w - v_b)\pi\gamma^2}{\pi\gamma^2 f'(\gamma)}. \tag{3.52}$$

We determine the time dependence for small γ in the same way as we did for β for channels in Section 3.3, with the small time behaviour of (3.52) giving a similar expression to (3.20),

$$\gamma \sim \left(\frac{4\rho_{iw} v_b a_0 t}{\pi f''(0)} \right)^{\frac{1}{4}}, \tag{3.53}$$

having expanded around $\gamma(0) = 0$ and noting that $f'(0) = 0$ at the local minimum.

3.4.3 Sensitivity to parameters for axisymmetric ponds

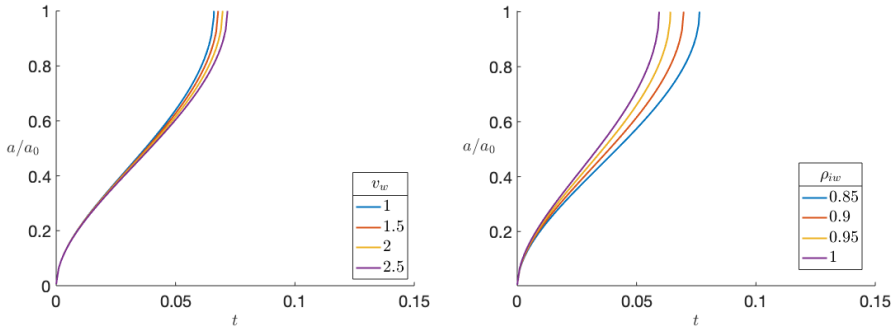
To explore the sensitivity of the model for a pond growing in a single catchment, we performed several numerical simulations of equation (3.43) and a hypsometry for a circular catchment of area π, with a cosine curve shaped topography, given by

$$h_0(r) = -\cos(\pi r), \tag{3.54}$$

with parameters ρ_{iw} and v_w varied. Results are shown in Figure 3.5. Making a general note on the behaviour of solutions, in all cases, there is an initial sharp increase in a/a_0 due to the shallow gradient of the topography near the minimum, before growth slows at intermediate times. Rapid changes are seen as the pond eventually floods and a/a_0 is close to 1, as the ice profile approaches a local maximum at the edge of the catchment basin. Frame (a) shows the dependence of the expansion of the pond area fraction, $a(t)/a_0$, on the melt rate. Dispersion of solutions only occurs at late time, once the topography shallows. Somewhat counter-intuitively, a greater degree of enhanced melting in the pond tends to slow pond growth. This can be rationalised by inspecting the contraction term in equation (3.43) - the more melting internal to the pond, the more of a void is caused due to the contraction of volume during the phase change from ice to water. This void has to be filled from melt water flowing in from the bare ice regions of the catchment. Frame (b) shows the effects of changes in the density parameter, which are more pronounced than the melt rate. However, we do not expect the density parameter to change appreciably in reality, so we keep a fixed $\rho_{iw} = 0.9$ for the rest of the thesis.

3.4.4 A square pyramid

Similarly to the notch in one dimension, in two dimensions, it is possible to derive an analytical solution to the model if we consider a square pyramidal topography. We consider a catchment in the shape of a square pyramid of side length 1 (such that $a_0 = 1$), perpendicular height 1 and face slope 2, shown in Figure 3.6(a). For



(a) (b)

Figure 3.5: Solutions of equation (3.46) for a circular catchment of area π, with a cosine wave topography, equation (3.54). (a) Variation in the area fraction over time due to changes in the melt parameter. (b) Variations due to changes in the density parameter. Parameters $v_w = 2$ and $\rho_{iw} = 0.9$ were used for values kept constant (e.g. ρ_{iw} in frame (a)).

this case, we can solve the equation for water levels implicitly. Modifying equation (3.46) for the area of a two dimensional catchment, the area of a square pyramidal catchment with base perimeter p grows as

$$\dot{a}(t) = \left(\frac{\rho_{iw} v_b a_0}{a(t)} - [1 - \rho_{iw}][v_w - v_b] \right) \frac{p}{2}. \tag{3.55}$$

We write the perimeter $p = 4\sqrt{a}$, and (3.55) can be rewritten

$$\dot{a} = \frac{\zeta}{\sqrt{a}} - \eta\sqrt{a}, \tag{3.56}$$

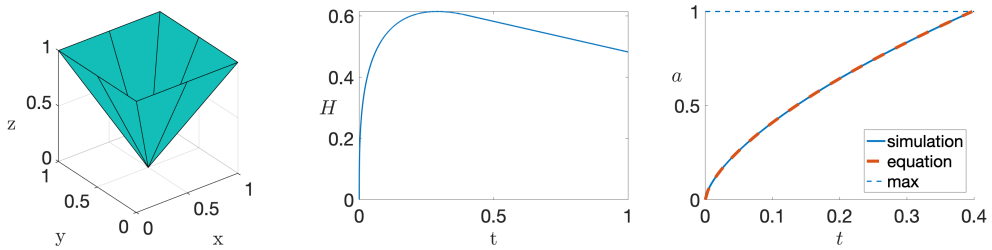
where $\zeta = 2\rho_{iw} v_b a_0$ and $\eta = 2(1 - \rho_{iw})(v_w - v_i)$. Equation (3.56) has implicit solution

$$t = \frac{2}{\eta}\sqrt{\frac{\zeta}{\eta}} \operatorname{arctanh}\left(\sqrt{\frac{\eta a(t)}{\zeta}} \right) - 2\frac{\sqrt{a(t)}}{\eta}. \tag{3.57}$$

Figure 3.6(b) and (c) show the behaviour of the numerical simulation of the single pyramid, with parameters $\rho_{iw} = 0.9$ and $v_w = 2$, where we have integrated the equation for the water level, equation (3.43) and evaluated the area at time t using a hypsometry,

$$a(t) := f(H, t) = (H + v_b t)^2 a_0. \tag{3.58}$$

This hypsometry is derived by shifting the water level to a moving frame with the ice, and using this shifted water level to calculate the width of the pond, which is twice the shifted height divided by the slope (which is 2). The factor of a_0 is kept



(a) (b) (c)

Figure 3.6: (a) A single square pyramid of side 1 and height 1, the initial topography for an analytical and numerical test of the model. (b) Evolution of water level. Melt rate parameter $v_w = 2$ was used. (c) Evolution of water area, compared with equation (3.57).

for physical clarity as the factor $(H + v_b t)^2$ can be thought of as a scale factor for the pond area fraction.

We see the fast increase in water level at early time, however as the simulation progresses, the interplay between diminishing amounts of bare ice melt and the translation of the whole ice surface downwards causes the increase in water level to slow down. Eventually, due to the contraction of volume in melting, the water level (relative to a fixed point deep in the ice, $z = 0$) begins to fall, because the pond base falls faster than new water volume is generated. Figure 3.6(c) compares $a(t)$ from a numerical simulation of the model with the implicit solution (3.57). We see that these agree exactly.

3.5 Summary

In this chapter we have considered the behaviour of a single pond growing in an isolated catchment on impermeable ice, first in one dimension and then in two dimensions. We derived equations for the evolution of the water level and pond area, which encapsulate the behaviour of individual catchments that will be used to determine the activity on nodes in our network model. For the asymmetric notch, these are equations (3.32) and (3.47). For the general two dimensional catchment, these are equations (3.43) and (3.47). The node behaviour depends directly on the slope of the initial topography, the melt rates and the ice-water density ratio. We found strong dependence on the density ratio, and that for a larger ratio, ponds will fill more quickly. This suggests that changes in salinity (which would change this melt ratio) could affect the behaviour of pond growth. However, in our model, we keep the

densities fixed. We found a weaker sensitivity to the ratio of the melt rates v_w/v_b, but that for a larger melt rate ratio, ponds fill more slowly. This indicates that the ratio v_w/v_b may not be the most important parameter in our model, and may be less important to the physics of pond systems than other aspects, such as the melting scale $\tilde{v}$ or the roughness of the topography Δh. We also found a steady state for the model, and analytical solutions for particular simple topographies. We find that the model is badly behaved as $a(t) \to 0$ and has a singularity here, so we found the asymptotic behaviour to use as an initial condition for the numerical integrations. In the model, discontinuities or cliffs can form in the ice topography if pond growth stops. This may help to explain the particular geometries displayed by pond floors, which steepen considerably near the edge. This is explored in more detail in Appendix B.

Chapter 4

Interactions amongst ponds and building a network

In this chapter we model ponding on a whole floe as a network of neighbouring catchments. We modify equation (3.4), conservation of mass for an individual pond (action on the node), to include fluxes from other ponds (action on the edges). We identify the different forms that the water flux between catchments can take: catchments can be isolated with no flow, water can overflow from one catchment to another, and eventually ponds in catchments adjacent to one another can join together. We first consider the model in one dimension to identify constraints on each type of flux and use these constraints to derive expressions for the fluxes. We then use the ideas developed for one dimension to derive a model in two dimensions. We also identify some intricacies of the model, and write it in matrix form for computational convenience.

4.1 Building a network

We consider the section of a floe shown in Figure 4.1. Here we have divided the ice surface into catchments (divided by dashed lines), each of area a_{0i}. A catchment is the area around a local minimum from where surface water that is free to move under gravity will move towards that local minimum, as determined by the gradient of the initial topography $\nabla h_0(\mathbf{x})$. For simplicity we will choose $h_0(\mathbf{x})$ such that it forms catchments in the shape of polygons, and that all of the boundaries between catchments, the dashed lines in Figure 4.1, are straight lines. The location of catchment boundaries is dependent on the highest points between neighbouring catchments, and the points at which water from a catchment can overflow into a neighbour. Due to melting of the ice surface, the positions of the highest points between catchments,

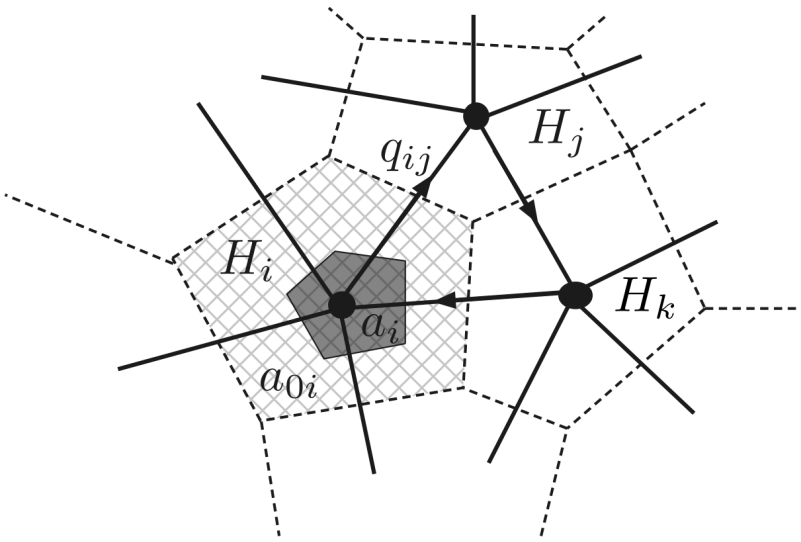


Figure 4.1: Diagram of catchments in two dimensions, with network architecture labelled, including nodes (dotted) and edges (arrows). Dashed lines mark the separation of catchments. The hatched area denotes the catchment i, a_{0i}. The shaded area denotes water covered area in i, a_i.

and the points over which overflowing occurs, may change, so it is possible that the catchment boundaries may change with time. This is discussed in Section 4.2.1.

Later, in simulations, we will focus on topographies where catchments either form Voronoi tessellations based on randomly placed points for catchment centres, or square lattices, where the centres are placed uniformly. The catchments are our nodes, which we label 1 to n. We label the centres $\mathbf{x}_{0i}$. The behaviour of the nodes is characterised by the pond volume V_{Ai} and water level H_i, which in turn, determine the pond area a_i. These, along with the catchment area a_{0i} are the node attributes.

As catchments start to fill above their boundaries, we allow fluxes q_{ij} between nodes, defined on edges. An edge exists between each neighbouring node pair across the boundary line between them. The fluxes are regulated by the ice height at the saddle $\mathbf{x}_{ij}$, the lowest point on the catchment boundary between catchments i and j. For simplicity we will at times write $h(\mathbf{x}_{ij})$ as the shorthand h_{ij}. In the same way that the initial pond is the physical feature represented by a node, the saddle can be viewed as the physical feature represented by an edge. The edge attributes are then the saddle heights h_{ij} and water fluxes q_{ij}. Considering each of the m saddles in turn, we write an edge list for the network, based on the ordering of the node list. In the case of the subgraph of catchments i, j and k shown in Figure 4.1, the edge list is given by $(\mathbf{l}, \mathbf{r}) = [(i, j); (j, k); (k, i)]$. The column vectors $\mathbf{l}$ and $\mathbf{r}$ assign a direction to edges and a valence (in and out, or 'left' and 'right') to the nodes at either side of an

edge. The valences of nodes associated with edges are chosen based on the ordering of the entries in the edge list. The valences are important in defining a convention for the direction of flow of water in the network, but do not have any further physical meaning. From here on, in order to describe the valence of nodes associated with edges we will refer to ponds as being to the left or to the right of edges, and set up our convention such that a flow from left to right is positive.

To visualise this, we can inspect the simple example shown in Figure 4.2. Each catchment is a node, labelled 1 to 3, based around a local minimum, x_{0i}. Each catchment has two neighbouring saddles (shown by dashed lines; in one dimension, saddles are the same as the local maxima). Each of these maxima is shared with another catchment and defines an edge, called a saddle, as this is how the point appears in the topography in two dimensions. In this particular case our network has $m = n$ edges. Starting at the edge from nodes 1 to 2 we write an edge list for the network. In the case of the ring in Figure 4.2, the edge list is given by $(\mathbf{l}, \mathbf{r}) = [(1, 2); (2, 3); (3, 1)]$.

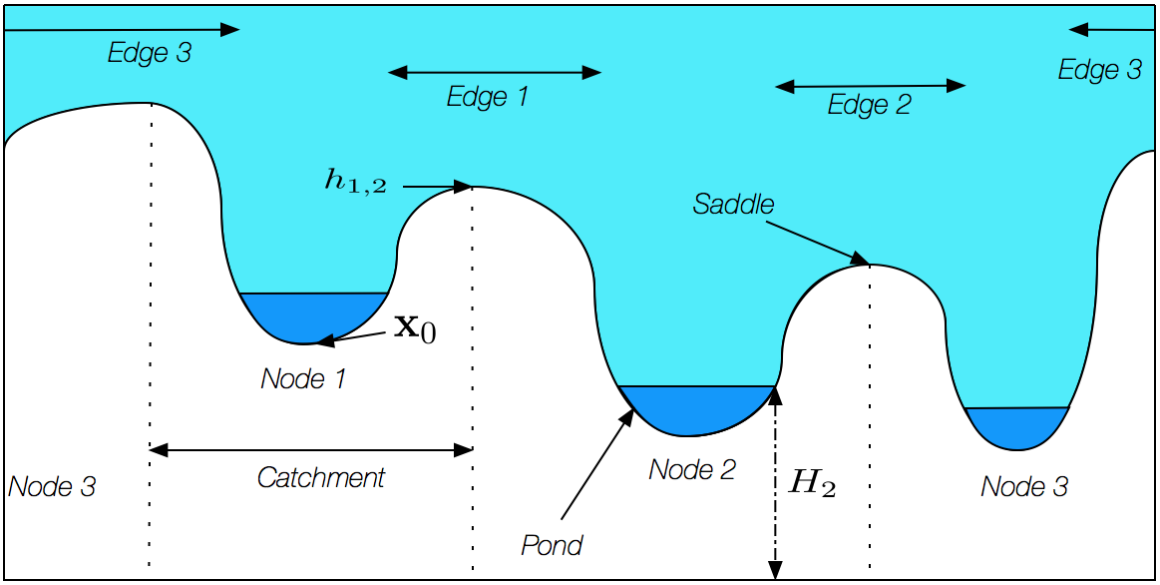


Figure 4.2: Diagram of a periodic ring of ponds in one dimension, with network architecture labelled, including nodes (pond centres) and edges (the higher points of ice, referred to as saddles due to their location in two dimensional topographies. Dashed lines mark the separation of catchments.

From the edge list column vectors we can write the $m \times n$ directed edge incidence matrix $\mathbf{B}$, a useful structure that maps edges (of which there are m) to nodes (of which there are n), defined in Chapter 2. We can decompose $\mathbf{B}$ into the difference of

two matrices $\mathbf{L}$ and $\mathbf{R}$, $\mathbf{B} = \mathbf{R} - \mathbf{L}$. $\mathbf{L}$ is the $m \times n$ matrix with a 1 in position i, j if the node j lies to the left of (or leaves) edge i, and zeros elsewhere. Likewise, $\mathbf{R}$ is the $m \times n$ matrix with a 1 in position i, j if the node j lies to the right of (or enters) edge i, and zeros elsewhere. $\mathbf{L}$ and $\mathbf{R}$ encode the network structure of the catchments of our floe, and must be set at the start of any numerical simulation. For what follows, we write scalars in lower case, vectors in bold lower case and matrices in bold upper case print, with the exception of H, which is bold for a vector and normal print for a scalar.

Nodes in our network behave according to mass conservation, equation (3.4) from Chapter 3, modified to incorporate interactions with other catchments,

$$\dot{V}_{Ai} = \dot{V}_{mi} - \sum_j q_{ij} \tag{4.1}$$

where q_{ij} is the water flux from pond i to j. We define the direction of q_{ij} to be positive if water is flowing from i to j, and hence $q_{ij} = -q_{ji}$. Using the expressions for water accumulation (3.9) and production (3.7), this becomes

$$a_i \dot{H}_i = \rho_{iw} v_b a_{0i} - [v_w - \rho_{iw}(v_w - v_b)]a_i - \sum_j q_{ij}. \tag{4.2}$$

The next section will focus on how we can constrain $\dot{H}_i$ in some ponds, depending on their behaviour, so that we can solve for each q_{ij} in the network, and determine $\dot{H}_i$ in all ponds. We can then determine $H_i(t)$, $a_i(t)$, $\dot{V}_{Ai}$ and $\dot{V}_{mi}$ which will be required in numerical simulations in later chapters.

4.2 Interactions in one dimension

In order to determine the behaviour of pond interactions, we consider the model in one dimension. If we consider two neighbouring catchments, shown in Figure 4.3, we identify two different types of interactions (and a state of non-interaction). In panel (a), both ponds have low water levels, and there is no interaction. The edge corresponding to these two ponds is inactive. In panel (b), the water level of the left pond reaches the lip of the saddle. After this point, seen in panel (c), the left pond starts to overflow with flux q_{ij} into the right pond. The water level in the left pond, H_i, is equal to the saddle height and the water level in the right pond is lower. The edge is now active and directed from i to j. Finally, as the water level in the right increases, and the saddle height decreases, the two ponds join and act as one pond

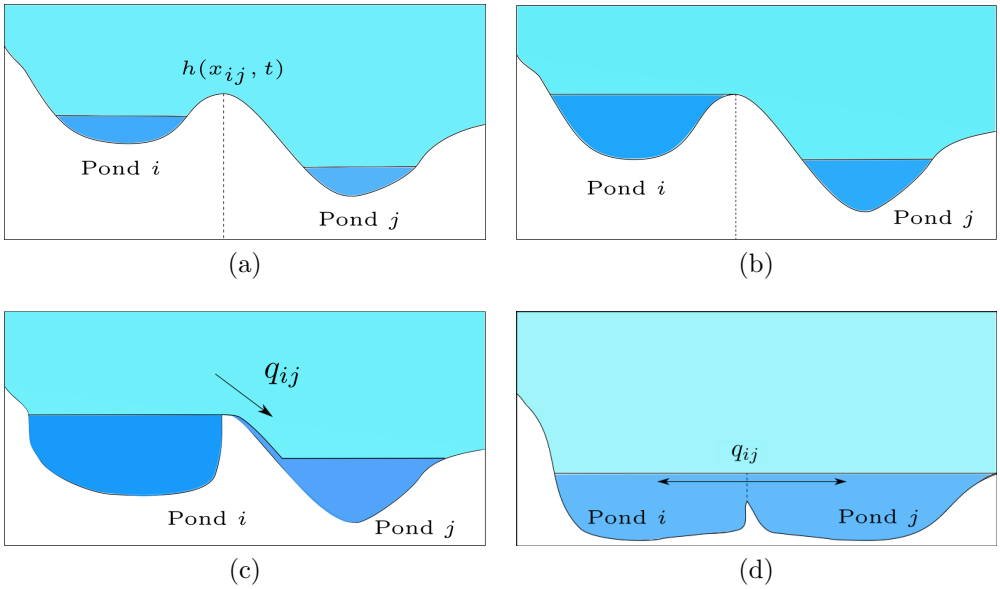


Figure 4.3: Schematic of the progression of coupling ponds. In panel (a), ponds grow independently. In panel (b), the left pond i grows so that it begins to overflow at its boundary into the right pond j. In panel (c) pond i is overflowing into j. In panel (d) the two ponds are joined and act effectively as one pond.

(panel (d)). Both water levels must be the same, and higher than the saddle height. The edge changes to an active undirected edge.

With reference to our network and the scenarios outlined, we say that an edge can exist in one of four possible states, labelled 1 to 4:

1. No flow across the saddle, an undirected, inactive edge,
2. Directed overflow from left to right,
3. Directed overflow from right to left,
4. Undirected joining, with possible flow in either direction.

The state of an edge (i, j), depends on the water levels H_i, H_j and the saddle height h_{ij}, and we can write this dependence as a set of logical conditions. A saddle between catchments i and j exists in state

1. if $H_i < h_{ij}$ and $H_j < h_{ij}$,
2. if $H_i \geq h_{ij}$ and $H_j < h_{ij}$,
3. if $H_i < h_{ij}$ and $H_j \geq h_{ij}$,

4. if $H_i \geq h_{ij}$ and $H_j \geq h_{ij}$.

It is also useful to define indicator vectors (also referred to as state vectors) $\mathbf{i}_1$, $\mathbf{i}_2$, $\mathbf{i}_3$ and $\mathbf{i}_4$, each of size $m \times 1$, for edges in these states. For, $\mathbf{i}_1$ the l^{th} entry is equal to 1 if the l^{th} edge in the edge list for the network is in state 1 and zero otherwise. The indicators $\mathbf{i}_2$, $\mathbf{i}_3$ and $\mathbf{i}_4$ are each defined in a corresponding way.

For an edge (i, j) in state 1, where there is no interaction, $q_{ij} = 0$. We now investigate how to determine q_{ij} in the other states.

4.2.1 Overflow

To investigate the overflowing of a pond we consider the saddle shown in panel (c) of Figure 4.3, with a pond i to the left with higher water level than pond j to the right. In this state the water level H_i is constrained to be equal to the height of the saddle, h_{ij}, and therefore need

$$\dot{H}_i = \frac{\mathrm{d}h_{ij}}{\mathrm{d}t}. \tag{4.3}$$

The flux of water $q_{ij}(t)$ must be such as to satisfy this constraint. In particular, since $\dot{H}_i$ is governed by the water conservation equation (4.2), the flux becomes

$$q_{ij} = \rho_{iw} v_b a_{0i} - \left[\frac{\mathrm{d}h_{ij}}{\mathrm{d}t} + v_w - \rho_{iw}(v_w - v_b)\right] a_i - \sum_{k \neq j} q_{ik}, \tag{4.4}$$

allowing other possible fluxes from pond k into pond i.

This model for the overspilling process will work for any saddle where the water level on one side begins to exceed it, and the water level at the other side lies below it. However, to determine the overflowing flux we now have to specify $\mathrm{d}h_{ij}/\mathrm{d}t$.

Choice of melt rate at the overspilling saddle

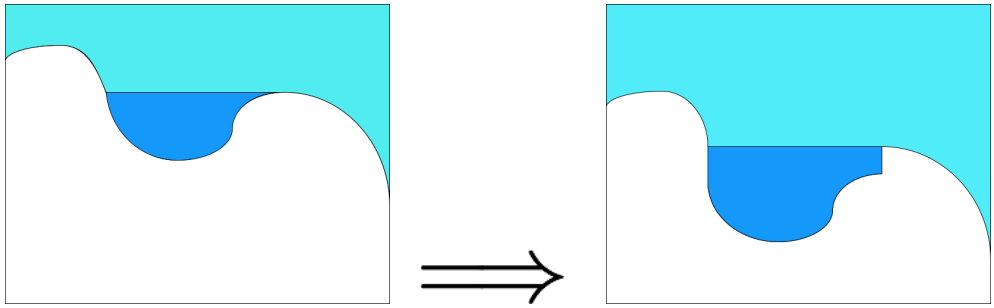
In the overflowing scenario that we present, $x_{ij} = \beta$ is the right hand boundary of pond i, so $h_{ij} = h(\beta(t), t)$. If we expand the derivative of the condition on the overspill point (4.3), it becomes

$$\dot{H} = h_t(\beta(t), t) + h_x(\beta(t), t)\dot{\beta}. \tag{4.5}$$

As was the case for equation (3.12), we must be careful how we evaluate the derivatives here. If we evaluate them at the right hand side (the bare ice side) of the boundary, we get

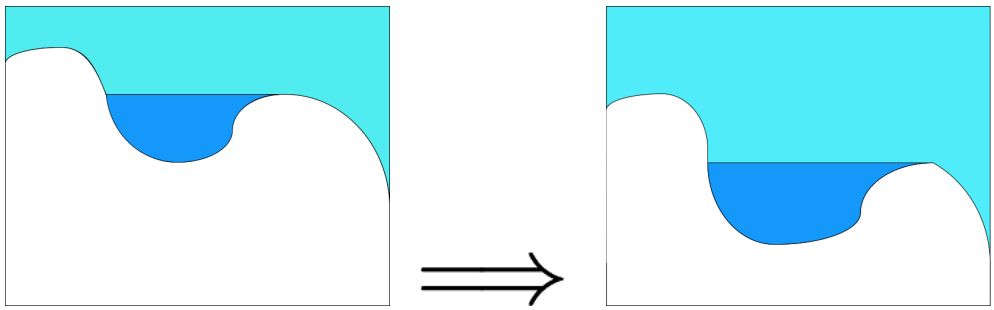
$$\dot{H} = h_{x0}(\beta(t), t)\dot{\beta} - v_b. \tag{4.6}$$

Previously, $\dot{H}$ was calculated by considering the melting and conservation of mass within the pond, (4.2), and then $\dot{\beta}$ was determined by rearranging (4.6). Now that both $\dot{H}$ and $\dot{\beta}$ are effectively determined by the same equation we must look elsewhere to constrain $\dot{\beta}$. As mentioned in Section 3.3.2, there is a solution to (4.6) where $\dot{\beta} = 0$ and $\dot{H} = -v_b$, which causes a cliff edge to form at the boundary, as illustrated in Figure 4.4. Another solution exists where $\dot{\beta} > 0$ and the overspill point advances, moving down the other side of the saddle and causing the water level to fall at a rate faster than v_b. A schematic of the behaviour of this solution can be seen in Figure 4.5.



(a) (b)

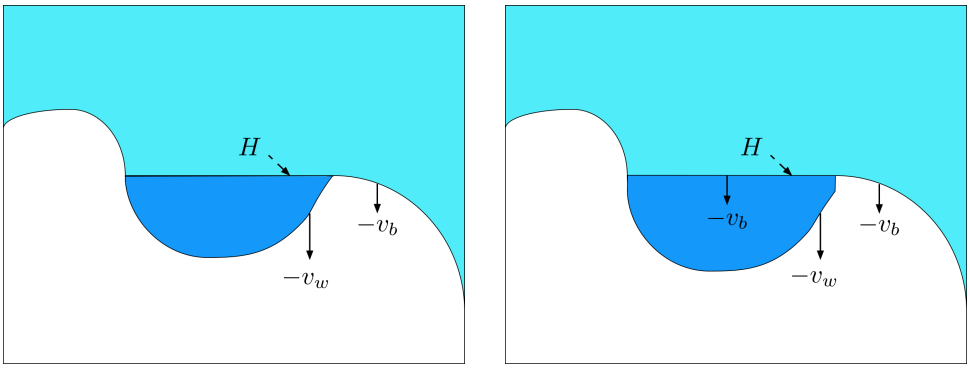
Figure 4.4: Schematic of behaviour of an overflowing pond where $\dot{\beta}(t) = 0$. The ice profile at the onset of the overflow is shown in panel (a). Panel (b) shows the profile after some time, where the pond has developed vertical walls due to the discontinuity in the melt rates.



(a) (b)

Figure 4.5: Schematic of behaviour of an overflowing pond where $\dot{\beta}(t) > 0$. The ice profile at the onset of the overflow is shown in panel (a). Panel (b) shows the profile after some time, where the overspill point has migrated to the right and created a sharp change in the ice profile.

To determine which behaviour of the overflow point is physically and mathematically consistent for our model, we consider the pictorial argument shown in Figure



(a) (b)

Figure 4.6: Annotated schematic of behaviour of an overflowing pond. (a) The ice profile at the onset of the overflow. (b) The profile after some time, when the pond has developed vertical walls due to the discontinuity in the melt rates.

4.6. We assume that water cannot accumulate at the local maximum h_{ij}, as it would flow under gravity over to the left or right to a lower position (we neglect wetting or adhesion, considering this to be negligible, like the film of melted water on bare ice). Therefore, we consider the saddle point to be bare ice, which melts with rate $-v_b$. As a reminder, the main difference between bare ice and pond-covered ice that accounts for the different melt rates is the absorption of short-wave solar energy over a sufficient depth of water. This means that we can consider ice with a negligible depth of surface water to be bare (from a thermodynamic point of view).

We assume that the water flowing out of the pond over bare ice has negligible depth, like the thin film of water that flows from the bare ice of the catchment to the pond. Therefore, the bare ice to the right of the overflow also melts at a rate $-v_b$. We note that this is spatially uniform, so all of the bare ice melts at this rate. The ice within the pond melts at $-v_w$. This too is a spatially uniform melt rate. As discussed in Chapter 2, this is due to the pond having a sufficient depth to absorb solar radiation, and to being well-mixed, therefore distributing heat in the pond such that the temperature is spatially uniform. The pond and bare ice are separated by the saddle, a local maximum. This is also now the location of the right pond boundary, β.

As shown in the figure, water cannot accumulate on the right side of the saddle, which we denote h_{ij}^+, but can on the left hand side h_{ij}^-. The different melt rates at either side of the saddle $h_{ij}^+ = -v_b$ and $h_{ij}^- = -v_w$ lead to a cliff, as shown in panel

(b). In the pond, the condition on the water level

$$\dot{H} = \frac{\partial h_{ij}^{-}}{\partial x}\dot{\beta} - v_b \tag{4.7}$$

still holds. Given that the water level in the pond moves with the saddle point, such that $\dot{H} = -v_b$, for (4.7) to hold, $\dot{\beta}$ must be zero, and the saddle cannot move. Therefore, in our model the boundaries and extents of the catchments remain stationary always.

Channelization

We neglect channelization during overflowing, and any erosion of the saddles (beyond melting at the bare ice melt rate), and argue that it happens at a negligible rate compared to the bare and pond covered melt rates, and the difference between them. If we were to include channelization due to turbulent melting, we would write

$$\dot{h}_{ij} = -v_b + \dot{h}_m \tag{4.8}$$

where $\dot{h}_m$ is the incision rate, and would use this $\dot{h}_{ij}$ in equations (4.3) and (4.4) as the model for overflowing, paired with an hypsometry function $\dot{a} = f(H, t)$ to determine the pond area.

To estimate the incision rate we consider the model for glacial lake draining presented by Kingslake et al. [2015]. That model describes the outflow over the lowest point on the boundary of a glacial lake with stagnant water, and the authors derive an incision rate for the uniformly sloping channel that emanates from the lake, assuming pseudo-steady plug flow in the channel. The authors also assume that the water level in the lake is slightly higher than the ice at the top of the channel, which drives the flow. We use the incision rate derived in the paper, and direct the interested reader to Sections 3 and 4 of Kingslake et al. [2015]. From the paper,

$$\dot{h}_m = \alpha \xi^{3/2} \tag{4.9}$$

where ξ is the height difference between the ice at the top of the channel (in our case the saddle point) and the water level in the lake, and

$$\alpha = \frac{f_R \rho_w}{8 L \rho_i} \left(\frac{2g}{1 + \frac{f_R}{4\phi_b}} \right)^{3/2}. \tag{4.10}$$

The densities ρ_w and ρ_i are those of water and glacier ice, L is the latent heat of fusion of ice, g is acceleration due to gravity, f_R is the hydraulic roughness of the channel and ϕ_b is the slope of the ice along the channel.

To estimate $\dot{h}_m$ and compare it to the melt rates in our model, we return momentarily to dimensional quantities. Using the velocity scale of 1 cm/day, $v_b = -1.15 \times 10^{-7}$ ms^{-1}. We assume the same hydraulic roughness used by Kingslake et al. [2015] of $f_R = 0.25$, and take $\xi = 0.005$ m, assuming it is small because the overflowing pond fills steadily and the overflow at the saddle does not overly restrict the outflow of water.

Using the scale lengths $\tilde{x} \sim 5$ m and $\tilde{h} = 1$ m, and assuming that the topographic deviation $\Delta h = 0.2$, we take $\phi_b = 0.04$. With these parameters, equations (4.9) and (4.10) give $\dot{h}_m \approx -7.7 \times 10^{-10}$ ms^{-1}, which we find to be negligible in comparison to v_b, v_w and the difference between them.

The overflowing flux

Having ascertained the behaviour at the saddle for an overflowing catchment we return to the catchments shown in Figure 4.3(a) to (c). With an appropriate choice of $\mathrm{d}h_{ij}/\mathrm{d}t = -v_b$, equation (4.4) for the flux out of catchment i becomes

$$q_{ij} = v_b \rho_{iw} a_{0i} - (1 - \rho_{iw})(v_w - v_b) a_i + \sum_{k \neq j} q_{ki}, \tag{4.11}$$

which describes the edge behaviour for any edge in state 2. Switching indices i and j gives the behaviour of any edge in state 3, by symmetry.

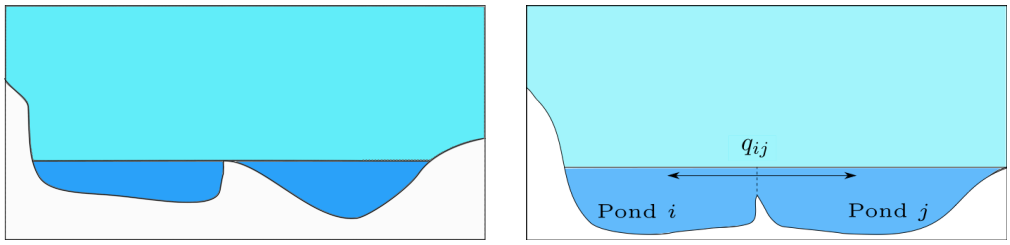
Due to our choice of melt rates, interesting behaviour occurs at the non-overspilling boundary of the pond, which we denote $x = \alpha$. With $\dot{H} = -v_b$, (3.36) shows that

$$\dot{\alpha} = \frac{-v_b + v_b}{h_{0x}(\alpha, t)} = 0. \tag{4.12}$$

Therefore, when the pond overspills at $x = \beta_i$, $\dot{\alpha}_i = 0$ and both boundaries remain static. As illustrated in Figure 4.4 a static pond develops cliff edges at boundaries. Hence the area of the pond does not change during overspilling. The arresting of the migration of the pond boundary has an important consequence for the network model, as it means that only one overspill point can develop for any pond, (until it joins with another, as discussed below). Therefore each pond can have at most one outward edge in the network at any time. There is an exception to this result in the unlikely case that a pond has two saddles at exactly the same height. However, we avoid this exception in numerical simulations. A strict control on the number of permissible active edges has consequences for the behaviour of the network model, even if the control is only on an interim step in pond evolution. This is an important result, and the consequences of this will be explored in some more detail in Chapter 7.

4.2.2 Joining

Figure 4.7 shows a schematic of the behaviour of ponds i and j, once they join fully and can potentially exchange water in either direction. In the notation of the network, the saddle between i and j is now in state 4. We expect that the new compound pond continues to grow, as show in Figures 4.7 (a) and (b), with the constraint that their water levels are now equal.



(a) (b)

Figure 4.7: Schematic of the onset of a join (a) and a join edge at a later stage (b).

When the ponds join, $\alpha_j = \beta_i$. Thereafter β_i and α_j remain fixed at the saddle point between points i and j. The outer boundaries, α_i and β_j, which now define the moving boundaries of the compound pond will start to move outwards as water from the surrounding, melting ice continues to flow into the pond according to

$$\dot{\alpha}_i = \frac{\dot{H}_i + v_b}{h_{0x}(\alpha_i)}, \tag{4.13}$$

$$\dot{\beta}_j = \frac{\dot{H}_j + v_b}{h_{0x}(\beta_j)}. \tag{4.14}$$

Since the water levels are equal, necessarily, $\dot{H}_i = \dot{H}_j$. This provides a constraint from which to determine the flux q_{ij}, which we now show. From mass conservation, (4.2), we have, for pond i,

$$a_i \dot{H}_i = \rho_{iw} v_b a_{0i} - q_{ij} - \sum_{k \neq j} q_{ik} - (v_w - \rho_{iw}[v_w - v_b])a_i, \tag{4.15}$$

and for pond j,

$$a_j \dot{H}_j = \rho_{iw} v_b a_{0j} - q_{ji} - \sum_{k \neq i} q_{jk} - (v_w - \rho_{iw}[v_w - v_b])a_j, \tag{4.16}$$

where in each case k is a dummy variable for elements in the set of neighbouring ponds. Equating $\dot{H}_i$ with $\dot{H}_j$ and recalling that $q_{ij} = -q_{ji}$, gives

$$q_{ij} = \frac{\rho_{iw} v_b (a_{0i} a_j - a_{0j} a_i) - a_j \sum_{k \neq j} q_{ik} + a_i \sum_{k \neq i} q_{jk}}{a_i + a_j}. \tag{4.17}$$

By extending this argument to a compound pond made up of n joined ponds, we obtain a similar constraint on each of the $(n-1)$ fluxes between them. The evolution of pond boundaries depends on the pond's location in the chain. For the pond at the left end of the chain, denoted as i,

$$\dot{\alpha}_i = \frac{\dot{H}_i + v_b}{h_{0x}(\alpha_i)} \tag{4.18}$$

and $\dot{\beta}_i = 0$. For the pond at the right end of the chain, denoted as n,

$$\dot{\beta}_n = \frac{\dot{H}_n + v_b}{h_{0x}(\beta_n)} \tag{4.19}$$

and $\dot{\alpha}_n = 0$. For any pond in the middle of the chain, $\dot{\alpha} = 0$ and $\dot{\beta} = 0$.

4.3 Interactions in two dimensions

4.3.1 Overflow

For an overflowing catchment in two dimensions, the constraints are the same as in one dimension, except that the overflowing catchment may be subject to incoming fluxes from other neighbouring catchments, rather than at most two others. The condition for an overflow to occur across a saddle from catchment i to catchment j remains the same as before. The area of the overflowing pond does not increase once overflow begins, since its boundary stops migrating outwards, as discussed for the one dimensional case in Section 4.2.1. The water level in the overflowing pond moves in tandem with the bare ice melting around it. The model equations for the overflowing catchment i and saddle ij become:

$$\dot{H}_i = -v_b, \tag{4.20}$$

$$\frac{\mathrm{d}h_{ij}}{\mathrm{d}t} = -v_b, \tag{4.21}$$

and this gives rise to the flux constraint as in (4.22),

$$\rho_{iw} v_b a_{0i} - (1 - \rho_{iw})(v_w - v_b) a_i = \sum_k q_{ik}, \tag{4.22}$$

where we have pulled all of the fluxes in and out of i into one term.

4.3.2 Joins

The behaviour of joining ponds in two dimensions is also almost identical to the one dimensional case. If we consider a pair of ponds, i and j in two dimensions that join, we have the same constraints on water level, $H_i = H_j$. As before, conservation of mass in each catchment satisfies (4.2) and equating $\dot{H}_i$ with $\dot{H}_j$ gives

$$\rho_{iw} v_b (a_{0i} a_j - a_{0j} a_i) = a_j \sum_k q_{ik} - a_i \sum_k q_{jk}, \tag{4.23}$$

as in (4.17).

4.3.3 Summary of the network model

Our model consists of a network of catchments with four different states of behaviour on edges. On each node, we solve the equation for water level

$$\dot{H}_i = \frac{\rho_{iw} v_b a_{0i} - \sum_k q_{ik}}{a_i} - [v_w - \rho_{iw}(v_w - v_b)]. \tag{4.24}$$

To solve this, we require the area a_i, the fluxes q_{ij}, and the heights of saddles, h_{ij}, which determine the states of edges. For a_i, we use an equation dependent on the topography and water level H_i, which will be discussed in Chapter 5. Assuming i to be on the left and j to be on the right, each edge has a flux q_{ij} and exists in one of four states:

1. no flow, if $H_i < h_{ij}$ and $H_j < h_{ij}$,
2. flow over the saddle from left to right, if $H_i \geq h_{ij}$ and $H_j < h_{ij}$,
3. flow over the saddle from right to left, $H_i < h_{ij}$ and $H_j \geq h_{ij}$,
4. joining, with possible flow in either direction, $H_i \geq h_{ij}$ and $H_j \geq h_{ij}$.

For an edge in state 1, $q_{ij} = 0$ and $\dot{h}_{ij} = -v_b$. For an edge in state 2, the flux q_{ij} is constrained by (4.22),

$$\rho_{iw} v_b a_{0i} - (1 - \rho_{iw})(v_w - v_b) a_i = \sum_k q_{ik}, \tag{4.25}$$

and $\dot{h}_{ij} = -v_b$. For an edge in state 3, the flux q_{ij} must satisfy

$$\rho_{iw} v_b a_{0j} - (1 - \rho_{iw})(v_w - v_b) a_j = \sum_k q_{jk}, \tag{4.26}$$

and $\dot{h}_{ij} = -v_b$. For an edge in state 4, the flux q_{ij} satisfies

$$\rho_{iw} v_b (a_{0i} a_j - a_{0j} a_i) = a_j \sum_k q_{ik} - a_i \sum_k q_{jk}, \tag{4.27}$$

and $\dot{h}_{ij} = -v_w$.

4.3.4 Loop formation

The model exhibits some inconvenient behaviour in the case of ponds joining in a loop. We first illustrate this behaviour by considering the specific example of three ponds in a triangle that have all joined, shown in Figure 4.8. In this particular example, the one and two dimensional cases are equivalent.

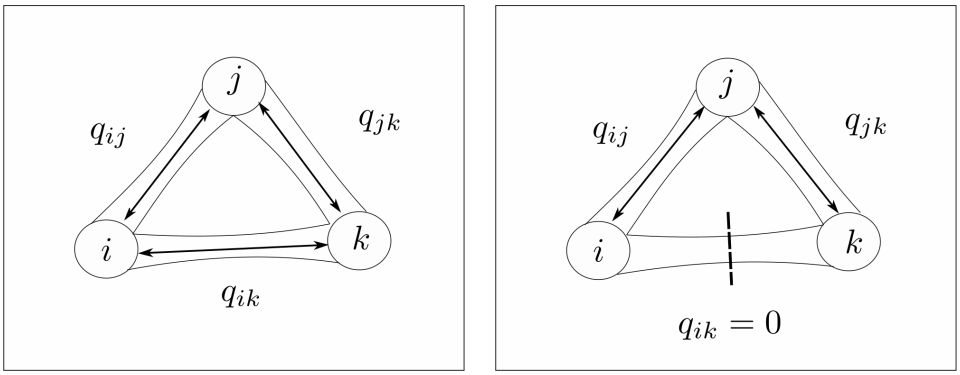


Figure 4.8: A triad of joined ponds (a) and the same triad with one edge cut (b).

Water levels in the triad evolve according to

$$\dot{H}_i = \frac{\rho_{iw} v_b a_{0i} + \sum_l q_{li}}{a_i} - [v_w - \rho_{iw}(v_w - v_b)], \tag{4.28}$$

with corresponding equations for $\dot{H}_j$ and $\dot{H}_k$. These water levels are constrained by

$$\dot{H}_i = \dot{H}_j = \dot{H}_k, \tag{4.29}$$

as the water level in the whole triad is equal. According to the constraints, (4.27), the fluxes on all three edges must satisfy

$$\begin{bmatrix} (a_i + a_j) & -a_i & -a_j \\ -a_k & (a_j + a_k) & -a_j \\ -a_k & -a_i & (a_i + a_k) \end{bmatrix} \begin{bmatrix} q_{ij} \\ q_{jk} \\ q_{k_i} \end{bmatrix} = \rho_{iw} v_b \begin{bmatrix} a_{0i} a_j - a_{0j} a_i \\ a_{0j} a_k - a_{0k} a_j \\ a_{0k} a_i - a_{0i} a_k \end{bmatrix}. \tag{4.30}$$

However, this system for the three fluxes does not have a unique solution, since the matrix on the left hand side is singular. This is because there is degeneracy in the

three constraints that have gone into (4.30). Any two of them imply the other by transitivity:

$$\dot{H}_i = \dot{H}_j, \qquad \dot{H}_j = \dot{H}_k \qquad \Rightarrow \qquad \dot{H}_i = \dot{H}_k. \tag{4.31}$$

This result applies to any such loop of arbitrary length, as can be seen by extending the transitivity argument through each edge in the loop.

The physical interpretation of a multiplicity of solutions in a loop is the possibility of a circulation flux that exchanges equal fluxes of water from one catchment to the next around the loop, in addition to the fluxes needed to maintain consistent water levels. The circulation flux could be determined with the introduction of a new constraint, such as a resistance to flow across edges. Instead, it is sufficient to break one edge in any loop, setting its state to inactive, setting the associated flux to zero and allowing the other constraints in the loop to set the water level, as shown in Figure 4.8(b). As ponds grow, these broken edges are not reintroduced (however, once we introduce drainage in Chapter 6 and water levels fall, they need to be accounted for). The broken edges are recorded in simulations and used in analysis in Chapter 7, in places where we examine the 'true' graph of ponding.

If we consider a graph of joined ponds with several loops, by inspection we find that there is a transitivity relation for each face except the external face (i.e. the boundary of the graph), requiring the breaking of $f - 1$ edges for a graph with f faces. By Euler's formula,

$$v - e + f = 2 \tag{4.32}$$

where e is the number of edges and v is the number of nodes. Hence if we remove $f - 1$ edges then the number of remaining edges is $v - 1$, and the graph takes the form of a tree.

4.4 Coupling a whole floe and a matrix representation of the model

We have seen in Section 4.3.4 that in our network any flux can depend on a number of other fluxes, so that the water level and hence water area in any catchment can be coupled to that of a number of other catchments. In order to solve for the water level in all catchments, given the possibility of widespread coupling of fluxes, we write the equations and conditions summarised in 4.3.3 in matrix form. To do this we return to the network architecture described in Section 4.1. The water level for a catchment

i evolves as

$$\dot{H}_i = \rho_{iw} v_b a_{0i} a_i^{-1} - [v_w - \rho_{iw}(v_w - v_b)] - a_i^{-1} \sum_j q_{ij}. \tag{4.33}$$

Vectorising this equation such that i could refer to any catchment in the network gives an equation for the vector $\dot{\mathbf{H}}$ of size $n \times 1$ (for n catchments/nodes), dependent on vectors $\mathbf{a}$ and $\mathbf{a}_0$ (pond and catchment areas respectively), each of length n, but also on a vector $\mathbf{q}$ of length m (for m edges). To ensure the product $a_{0i}a_i^{-1}$ in the first term on the left-hand side has size $n \times 1$, we write the a_i^{-1} terms as the $n \times n$ diagonal matrix $\mathbf{A}^{-1}$ with entries a_i^{-1}.

We must take care when linking edge behaviour to node behaviour. In the network we can map quantities that live on edges to their associated nodes using the directed edge incidence matrix $\mathbf{B}$, which we decompose into the outward directed edge matrix $\mathbf{L}$ and the inward directed edge matrix $\mathbf{R}$, described in Section 4.1. In this way, if we left multiply an $m \times 1$ column vector of a variable that lives on edges by $\mathbf{L}^T$, we can map the edge variables to those nodes to which they are incident from the left. The equivalent for the right side of edges can be done using $-\mathbf{R}^T$.

Using these edge matrices to map between catchments and fluxes in this way, the water level equation can then be written

$$\dot{\mathbf{H}} = \rho_{iw} v_b \mathbf{A}^{-1}\mathbf{a}_0 - [v_w - \rho_{iw}(v_w - v_b)]\mathbf{1} + \mathbf{A}^{-1}(\mathbf{R} - \mathbf{L})^T\mathbf{q}, \tag{4.34}$$

where $\mathbf{1}$ is a column vector of ones of length n. Following the decision made in Section 4.2.1 for the melting behaviour of saddles, we can write the rate of ice loss at each saddle as

$$\dot{\mathbf{h}}_s = -v_b(\mathbf{i}_1 + \mathbf{i}_2 + \mathbf{i}_3) - v_w\mathbf{i}_4, \tag{4.35}$$

where $\mathbf{h}_s$ is the $m \times 1$ vector of saddle heights, the h_{ij}'s, and we say that any saddle with associated edge in states 1, 2 or 3 melts with rate v_b and any in state 4 melts at rate v_w. Recall that the $\mathbf{i}$'s are indicator functions for each state.

For individual q's we write a matrix equation of the form

$$\mathbf{M}\mathbf{q} = \mathbf{b}, \tag{4.36}$$

where the rows of these equations correspond to equations (4.25), (4.26) and (4.27), so that we can solve for all m fluxes at once. We have already identified that any edge can exist in one of four possible states. In order to represent these states in this matrix equation, we write four $m \times m$ indicator matrices based on the indicator vectors $\mathbf{i}_1$ to $\mathbf{i}_4$ derived earlier, whereby we define $\mathbf{I}_j = \text{diag}(\mathbf{i}_j)$ for $j = 1, \ldots, 4$. For

each edge in state 1, corresponding rows of (4.36) need to state that $q_{ij} = 0$. Using the indicator matrix we can write the set of these equations as

$$\mathbf{I}_1\mathbf{q} = \mathbf{0}. \tag{4.37}$$

Fluxes across edges in states 2 and 3 contribute equations of the form of (4.25) and (4.26), but we must take care to represent the flux direction correctly using the edge matrices $\mathbf{L}$ for state 2 and $\mathbf{R}$ for state 3. The rows of (4.36) corresponding to edges in state 2 are supplied by

$$\mathbf{I}_2\mathbf{L}(\mathbf{A}^{-1}(\mathbf{R}-\mathbf{L})^T)\mathbf{q} = \mathbf{I}_2[(1-\rho_{iw})(v_w - v_b)\mathbf{1} - \rho_{iw}v_b\mathbf{L}\mathbf{A}^{-1}\mathbf{a}_0], \tag{4.38}$$

and for those in state 3 by,

$$\mathbf{I}_3\mathbf{R}(\mathbf{A}^{-1}(\mathbf{R}-\mathbf{L})^T)\mathbf{q} = \mathbf{I}_3[(1-\rho_{iw})(v_w - v_b)\mathbf{1} - \rho_{iw}v_b\mathbf{R}\mathbf{A}^{-1}\mathbf{a}_0]. \tag{4.39}$$

Each edge in state 4 contributes an equation in the form of (4.23) to the system. In matrix form, we write these as

$$\mathbf{I}_4(\mathbf{L}-\mathbf{R})\mathbf{A}^{-1}(\mathbf{R}-\mathbf{L})^T\mathbf{q} = -\mathbf{I}_4\rho_{iw}v_b(\mathbf{L}-\mathbf{R})\mathbf{A}^{-1}\mathbf{a}_0. \tag{4.40}$$

In (4.40), the factor $\mathbf{L}-\mathbf{R}$ on the right hand side produces the $(a_{0i}a_j - a_{0j}a_i)$ term from (4.27). Each of the $\mathbf{L}-\mathbf{R}$ factors on the left hand side contributes one of the other fluxes, $a_j\sum_k q_{ik}$ and $a_i\sum_k q_{jk}$. Summing all four equations (4.37), (4.38), (4.39) and (4.40) we effectively assign one equation to each row of (4.36) (since every edge is in one of the 4 states), which becomes

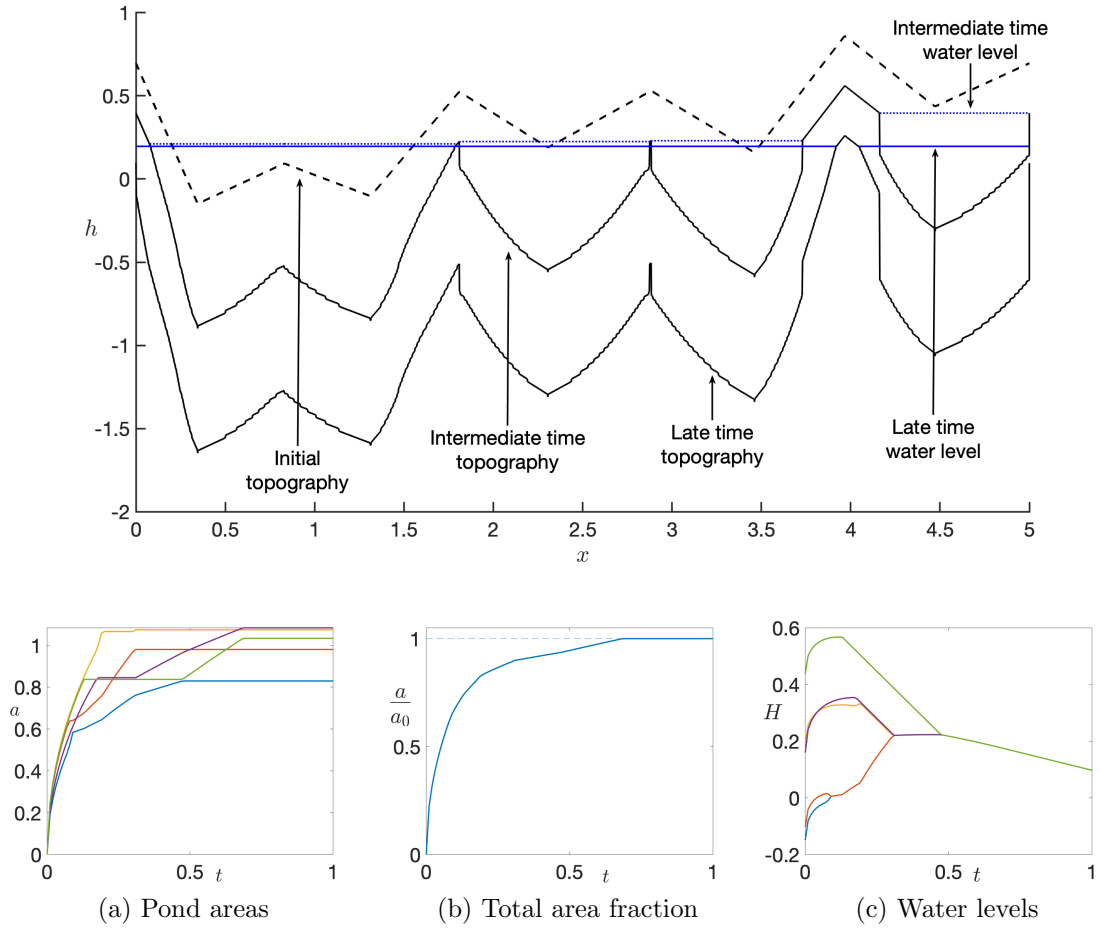
$$\begin{aligned}
&\left[\mathbf{I}_1 + \mathbf{I}_2\mathbf{L}\mathbf{A}^{-1}(\mathbf{R}-\mathbf{L})^T + \mathbf{I}_3\mathbf{R}\mathbf{A}^{-1}(\mathbf{R}-\mathbf{L})^T + \mathbf{I}_4(\mathbf{L}-\mathbf{R})\mathbf{A}^{-1}(\mathbf{R}-\mathbf{L})^T\right]\mathbf{q} = \\
&\mathbf{I}_2((1-\rho_{iw})(v_w - v_b)\mathbf{1} - \rho_{iw}v_b\mathbf{L}\mathbf{A}^{-1}\mathbf{a}_0) \\
&+ \mathbf{I}_3((1-\rho_{iw})(v_w - v_b)\mathbf{1} - \rho_{iw}v_b\mathbf{R}\mathbf{A}^{-1}\mathbf{a}_0) \\
&- \mathbf{I}_4(\rho_{iw}v_b(\mathbf{L}-\mathbf{R})\mathbf{A}^{-1}\mathbf{a}_0).
\end{aligned} \tag{4.41}$$

Equation (4.41) can be solved uniquely for $\mathbf{q}$ assuming the absence of loops in the network of join edges, which would cause the matrix to be singular as discussed earlier. In Appendix A and Chapter 6 we discuss how to ensure no loops form in the network. Finally, in the network model, it is also useful to solve for the melt volume produced in each catchment over time. Writing equation (3.2) in vector form, the rate of melt volume production in each catchment is given by

$$\dot{\mathbf{V}}_m = \rho_{iw}v_b(\mathbf{a}_0 - \mathbf{a}) + \rho_{iw}v_w\mathbf{a}. \tag{4.42}$$

4.5 Simulations in one dimension

To demonstrate a simple case of the network model, we present here a one-dimensional simulation that uses an initial topography of notch shaped ponds arranged in a line, with periodic boundaries. This forms a ring network, one of the simplest networks and a typical first step in building network models. This initial topography is shown on the black dashed line in Figure 4.9(a).



(a) Pond areas (b) Total area fraction (c) Water levels

Figure 4.9: (Top panel) Topographies and water levels for a one dimensional simulations of five catchments, calculated at various times, given an initial topography shown by the dashed line. 'Intermediate time' corresponds to $t \approx 0.3$ and 'late time' corresponds to $t \approx 0.6$. (a) Pond areas. (b) Total area fraction. (c) Time series of water levels. Melt rate parameter $v_w = 2$ was used.

4.5.1 One dimensional area functions

In this one-dimensional example, the extent a_0 of a catchment is the distance between the saddles either side of the centre. The pond extent in a catchment can be calculated by considering the intersection of the horizontal line at $z = \tilde{H}(t) = H(t) + \int_0^t v_b(t')\mathrm{d}t'$ with the left and right sloping faces of the catchment in the initial topography $h_0(x)$. Using the slope $s = (z_s - z_0)/(x_s - x_0)$ of each face, the contribution to pond extent from each face is $(\tilde{H}(t) - z_0)s$, and the pond area in each catchment is the sum of the contributions from its two faces.

4.5.2 Numerical considerations

Initial conditions must be offset slightly in time, as discussed in Chapter 3, and the initial singularity is treated in the same way as there. The other thing we must consider for a simulation in one dimension is the formation of loops. For the one dimensional ring, identifying the formation of a loop is trivial because one can only occur when all catchments are connected, and this can only happen when the highest saddle becomes submerged. To ensure no loops can form, we identify the highest saddle in the topography and fix the associated edge in the graph in state 1 permanently. We note that this edge could never become an overflow edge, and will only become a join edge once the whole floe floods, and at which point the loop would form.

4.5.3 An example of five notches

Figure 4.9 shows results of a simulation of five catchments with notch geometries in one dimension. Panels (b) and (d) show distinct signatures of edge changes (overflows or joins forming) which will be discussed in more detail for two dimensional simulations in Chapter 5 . The total area fraction, shown in panel (b), reaches a maximum of 1 in finite time, as in the case of the single catchment. The noticeable difference between the coupled model of several catchments and the single catchment is the existence of sudden changes in slope of the area fraction plot as catchments begin to interact more, from the period of $t = 0.2$ up to when the floe floods at around $t = 0.7$.

The one-dimensional simulation is computationally efficient enough to allow us to track changes in the topography of pond floors, seen in the solid black lines in Figure 4.9. To do this, we discretize the x axis into intervals of equal width. In each timestep, we adjust the height of each interval by $-v_b\Delta t$ if the centre-point of the interval is bare ice, where Δt is the size of the time-step, and $-v_w\Delta t$ if it is pond-covered. The top dashed line shows the initial topography. The middle solid black line shows the

topography at an intermediate time, $t \approx 0.3$. The dotted blue line shows the water levels at this time. Finally the bottom solid black line shows the topography at late time, $t \approx 0.6$. The corresponding water levels are shown in solid blue.

Looking at panel (d), at $t = 0.3$ we see two distinct water levels, corresponding to two distinct ponds, which are also seen in panel (a). By $t = 0.6$, only one water level exists, and the floe has flooded, save for the second last peak on the right. Water levels rise initially as ponds fill. Once all ponds have joined, the water level falls, as there is little incoming water from bare ice, and due to the melting of the ice at the pond floor, the density ratio ρ_{iw} causes a contraction in total volume (ice and water). It is also instructive to note the obvious cliffs that form in the topography due to overflow edges (including a cliff at the very end of the topography, $x = 5$, where it maps back onto the start, $x = 0$), in agreement with the behaviour described in Section 4.2.1. A nice feature of this particular example is the very thin cliff that forms at the saddle located near $x = 3$, due to the fact that in intermediate time (for which the water levels are marked with dashed lines) it is in the middle of a cascade, with the catchment to its right overflowing into the catchment to its left, and the catchment to its left overflowing in turn to the catchment to the left of that catchment again. This causes cliffs on both sides of the saddle, as shown in the topography.

4.6 Summary

In this chapter we have begun to examine how a network of multiple catchments on a floe behave as melting progresses. We first identified the possible ways in which fluxes of water can occur across the saddle points between catchments. We determined that there could be: no flow; flow over the saddle in one direction (overflow); or flow across the saddle in either direction (joining). By focusing on individual saddles we then derived the conditions for each of these states and the associated constraints that determine the fluxes.

In this analysis we found that in our model a catchment can only overflow across one saddle at any one time, and that during overflow, the area of the catchment stays constant. This constraint on the states that particular edges can have while overflow edges are prevalent in the system has the potential to be a strong control on the behaviour of the network. The cessation in pond growth during overflow was found to lead to the formation of cliff edges at the pond boundary in those that overflow. As seen in Figure 4.9(a), this leads to steep walls in ponds, rather than shallow, beach-like shores (note that the aspect ratio in this panel is exaggerated; the

x axis is squeezed, making cliffs more prominent in the figure than they would be in reality). This is in agreement with observations of ponds [Fetterer and Untersteiner, 1998], that have near vertical side walls and even form overhanging lips, as opposed to smooth beaches that are often seen at the edge of water formations on land, such as lakes.

In deriving conditions and flux equations for joining ponds, we found that our model cannot incorporate loops of joined ponds, as loops cause degeneracy in the flux equations. In using the model to perform simulations, we need to ensure that loops are prevented from forming. Finally, by making use of a set of indicator matrices and the edge-incidence matrices for the network of catchments we wrote a version of the model for a whole floe of individual but interacting catchments as a set of matrix equations. In the next chapter, we will use these equations to simulate systems of ponds.

Chapter 5

Simulating a floe

In this chapter we build two-dimensional numerical simulations of an entire floe of impermeable ice, focussing on building computational infrastructure for solving the system of equations developed in Chapter 4. We begin by designing synthetic initial topographies based on a triangular mesh, and compare these with observations of sea ice. We derive functions to relate pond area to water levels for the triangular mesh, and describe how the singularity at zero pond depth and area is treated. We describe simple test simulations based on square pyramids, for which we can calculate solutions exactly, then consider simulations of increasing complexity. These test simulations help to build intuition for the features seen in the model output, which will help us to examine the results of simulations more quantitatively in Chapters 6 and 7. We also discuss how model behaviour depends on what is assumed about network edges at the exterior boundary of the floe.

5.1 Defining the initial topography

5.1.1 Defining catchments and the network structure

To define a topography we first choose n, the number of catchment basins. We define a square domain of area a_{floe} such that we have a pre-determined average catchment area $\bar{a}_{0i} = a_{floe}/n$. Unless stated otherwise, we will choose a_{floe} such that $\bar{a}_{0i} = 1$. To create 'random lattice' topographies, we scatter n points uniformly at random in this domain, which will define the local minima, the 'centres' of the catchments, shown in Figure 5.1(a). The coordinates of these points are denoted $\mathbf{x}_0$. The boundaries for the catchments are defined by calculating the Voronoi tessellation for the points laid down, using Euclidean distance, as shown in Figure 5.1(b). Voronoi edges intersecting the exterior of the floe are stopped where they intersect with the square boundary. For

any cell that lies along the exterior, the square boundary lines are used to complete the cell polygons (catchment areas). The Voronoi tessellation defines the vertices

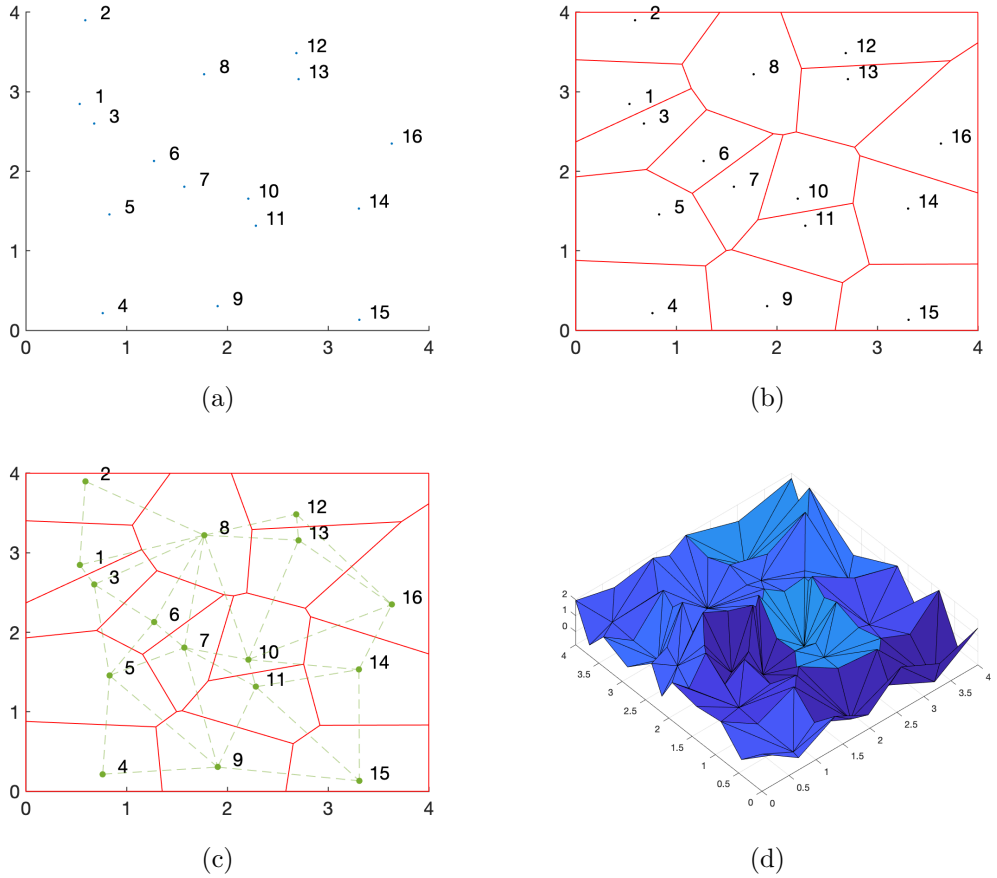


Figure 5.1: Schematic of the development of a synthetic topography. In panel (a), we lay down points for catchment centres uniformly at random (numbered). In panel (b), boundaries of catchments are calculated as a Voronoi tessellation (solid red lines). In panel (c) the edge structure of the network of possible fluxes $q_{i,j}$ between catchments is determined by the Delaunay triangulation of the tessellation (dashed green lines). (Note that the dashed lines represent the network structure, not the exact path of water fluxes, which have to cross the saddle points lying on the ridges between catchments.) In panel (d) a mesh of $h_0(\mathbf{x})$, is formed, by setting heights for centres, peaks and saddles and interpolating between them. Different shades of blue in (d) denote different catchments.

that bound each catchment, which we take to be the local maxima of the initial topography (referred to as 'peaks'). These have coordinates $\mathbf{x}_{pj}$. Saddles are placed at the mid-point of each Voronoi edge. For any one catchment, the location of the centre point, the vertices of the Voronoi cell, and the saddles, along with the heights of each of these points, define the initial topography. We use flat triangular faces to

interpolate the surface between each of these set points, an example of which can be seen in Figure 5.2(a). To build the network structure described in Section 4.1, we consider each Voronoi edge in the tessellation in turn and add this to the edge list. From there we build the edge incidence matrices $\mathbf{L}$ and $\mathbf{R}$.

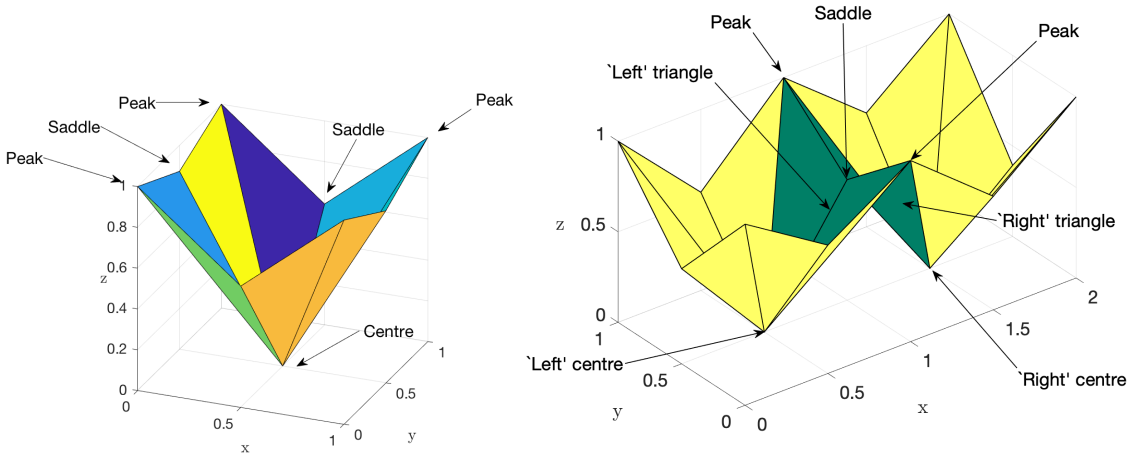


Figure 5.2: (a) An example triangular mesh of one catchment, labelling some peaks, some saddles and the catchment centre. (b) Example topography of two catchments, with points and faces along the shared boundary labelled. The faces either side of the saddle are coloured for significance.

In some cases, we will use more ordered 'square lattice' topographies. To create these topographies, the centres are placed at points on a square grid, as shown in Figure 2.3(b). The Voronoi cells, set points and other structures for these topographies are then assigned in the same way as for random lattice topographies.

5.1.2 Choosing heights of points

We now describe a simple method used to choose the initial heights of points in the topography. Point heights are generated at random, but in such a way as to ensure that in any one catchment, the centre is the lowest point, and that the saddle between any two neighbouring peaks is lower than either of these peaks, while being higher than the two neighbouring centres. We assume that the peaks all lie above the mean floe thickness, known as the mean line $\bar{h}$, and are bounded by a roughness parameter Δh, the maximum deviation from the mean. Similarly we assume that all centres lie below the centre line and lie at most Δh below it. The heights of each peak z_{pj} in the

floe are chosen first, uniformly at random between $\bar{h}$ and $\bar{h} + \Delta h$, $z_{pj} \sim U[\bar{h}, \bar{h} + \Delta h]$. The heights of the centres, z_{0j} are chosen in the same way as the peaks, uniformly at random between $\bar{h}$ and $\bar{h} - \Delta h$, $z_{0j} \sim U[\bar{h} - \Delta h, \bar{h}]$.

The saddle point heights, z_{sj} are chosen uniformly at random from the interval $[a_j, b_j]$ where a_j is the maximum of the centre heights of the two neighbouring catchments and b_j is the minimum of the two neighbouring peak heights, i.e. $z_{sj} \sim U[a_j, b_j]$ where $a_j = \max\{z_c : \text{centres on either side of saddle } j\}$ and $b_j = \min\{z_p : \text{peaks on either side of saddle } j\}$.

5.1.3 Behaviour of exterior saddles

At the exterior of the domain, we have two choices - to allow water to flow into the sea, or design our simulation such that our topography has high walls at the exterior (such as would be the case for a floe with extensive ridging - a common configuration). In the latter case we define a 'dummy edge' on each boundary edge, to be used for plotting and calculating area changes in the integrator of the numerical scheme. We choose the heights of vertices and the heights and position of saddle points along the border in the same manner as for points inside the domain, except that for these points, the height of only one centre restricts the saddle height. The dummy edges do not appear in the edge list and do not have any fluxes across them, so no catchment will lose or gain water across them, but they do appear in arrays that represent the geometry of the catchment basins (as discussed in Section 5.3).

In the case where drainage over the exterior is considered, we connect the boundary edges from each boundary catchment to a 'sea node', where any excess water can flow. The sea node is labelled $n + 1$ and, for now, its geometric extent and area are unimportant. We neglect buoyancy while modelling the impermeable phase of pond evolution. Buoyancy is introduced in Chapter 6, where a model with porous drainage is described, and the effect of buoyancy is important. In experiments which do not consider buoyancy (this chapter), the water level in the sea node is set to be below the height of the lowest saddle on the exterior of the floe, so that no flux from the sea to the floe can occur. In these experiments the exterior edges allow fluxes of water into the sea node in the same way as water would flow over the sides of an ice floe. In later experiments where buoyancy is considered (from Chapter 6 onwards) the water level of the sea node is allowed to change as water is shed from the floe, which rebounds buoyantly.

For the majority of this chapter we adopt the first of these choices for the exterior boundary and allow no flow. In Section 5.6 we examine what happens when we allow flow into the sea node.

5.2 Comparison of synthetic topographies against data

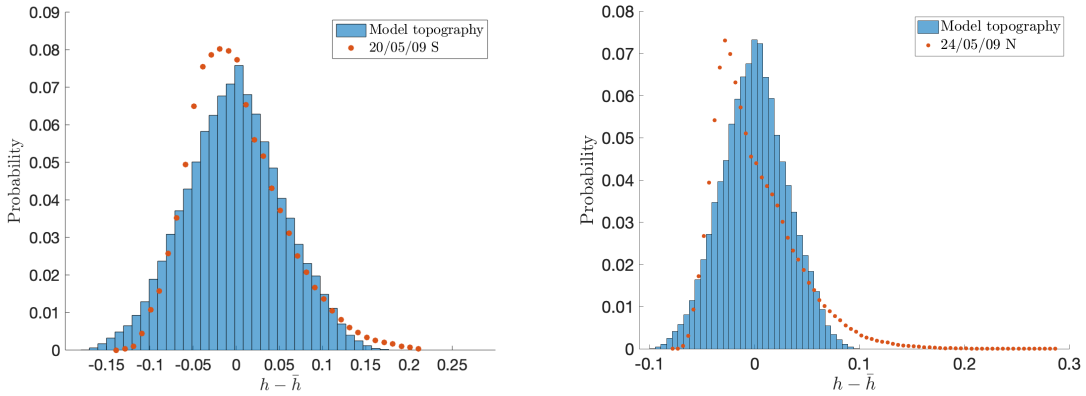
5.2.1 Surface height distribution

In creating random topographies, we have chosen a method that is straightforward to develop and understand, fits neatly into the structure of a network, can be scaled to create ice that resembles both FYI and MYI floes, and for which the area can be easily calculated as a function of water levels and time. Unfortunately, reality and computational simplicity are not necessarily always found together, and we must ensure that our synthetic topographies resemble real ice floes. To do this, we compare a synthetic topography to data taken by Polashenski et al. [2012], which includes a distribution of ice surface heights at a field camp in the Arctic, recorded between May 20^{th} and 24^{th} 2009, at two sites, labelled 'north' and 'south'.[1] In this section we will refer to this data as the 'field data'.

To form an initial surface height distribution, we took a random topography generated by the method described in Section 5.1, with amplitude Δh_1 and picked n_h points uniformly at random on the topography and recorded the point heights. We chose n_h to be the same as the number of data points used in the field data, and calculated both the standard deviation of the field data σ_{field} and the random topography σ_{test}. We then created a new random topography, where $\bar{h}$ was taken to be the mean of the distribution of the field data and the amplitude was calculated as $\Delta h = \Delta h_1(\sigma_{field}/\sigma_{test})$ so that the standard deviation of h for the new random topography was the same as that of the field data. Finally, we resampled, picking n_h points uniformly at random from the whole of the new topography,. The histogram of these points is shown in Figure 5.3(a). Note that the plot is shifted such that the average height is zero. Units are in metres. Bins were chosen so that the number of bins was the same as for the field data distribution.

The synthetic topography distribution broadly matches the observations. We note however that the random topography misses some mass near the peak of the real distribution, and is biased towards higher ice near the mean line, $\bar{h}$ (or zero

[1] This data is freely available online at `https://chrispolashenski.com/data.php`, along with a distribution of ice surface heights calculated by Polashenski et al. [2012].



(a) (b)

Figure 5.3: Distribution of initial topography heights at random points from a simulation using Δh scaled to data from the start of the melt season in 2009 at (a) the south site and (b) the north site, as measured by Polashenski et al. [2012]. Shown also is their data. Units are in metres. Bins were chosen so that the number of bins was the same as for the distribution presented in the data.

on the figure), as it has lower skewness than the field data. It also has has deeper hollows (more mass at the far left of the distribution). The excess mass of the random topography distribution at the left suggests that the pointed centres of our catchments are not entirely realistic, as the ice surface at its deepest points would be flatter. The missing mass at the right suggests that peaks should be more pointed, especially for panel (b), the north site. I note from my own observations of floes, made during the Healy 1902 cruise, that these features are more ridge-like in reality. This can also be seen in photography from other fieldwork, including those presented by Perovich et al. [2009]. These issues notwithstanding, we deem our synthetic topographies good enough to use in numerical experiments. Fitting an idealised topography to observations is not the specific aim of this work.

Other forms of random topography, based on different distributions from which point heights were chosen, were analysed briefly during the development of our model, however no large changes in behaviour in simulations were seen. Alterations involved modifying the distributions used to choose the heights of centres, peaks and saddles on an interval $[a, b]$. Six different topographies were tried that did not use uniform distributions. One used normal distributions for each point height, with $\mu = (a+b)/2$ and mean $\sigma = 0.1(a - b)$, truncated between a and b. Another used $\mu = (a + b)/2$ and $\sigma = 0.2(b - a)$ truncated between a and b. A third used $\mu = (a + b)/2$ and $\sigma = 0.5(b - a)$ truncated between a and b. Three forms of topographies using logistic

distributions for point heights were also investigated, with $\mu = (a+b)/2$ and scale factor $\sigma = 0.05(b-a)$, $\sigma = 0.1(b-a)$, and $\sigma = 0.2(b-a)$, truncated between a and b again in each case. This lack of dependence of the behaviour of pond coverage to the distributions of the point heights indicates that the behaviour is dominated by the means of the point heights.

Although we cannot derive an exact expression for the distribution of points, the simplicity of the method we have chosen creates topographies with some convenient features, and we will use these to develop approximations for the time scales of processes, and changes in pond extent in Chapter 7. From our choice of uniform distributions for peaks and centres, the average peak height is

$$\mathbb{E}(z_p) = \bar{h} + \Delta h/2, \tag{5.1}$$

the average centre height is

$$\mathbb{E}(z_0) = \bar{h} - \Delta h/2, \tag{5.2}$$

and the average saddle height is

$$\mathbb{E}(z_s) = \bar{h}. \tag{5.3}$$

These expressions emerge naturally from the properties of the uniform distribution:

$$\mathbb{E}(U[a,b]) = \frac{a+b}{2}, \tag{5.4}$$

since we have chosen the peak heights from $U[\bar{h}, \bar{h}+\Delta h]$ and the centre heights from $U[\bar{h}-\Delta h, \bar{h}]$. The saddle heights are chosen to lie uniformly at random between the maximum of two centres and the minima of two peaks, so by symmetry of the distributions of the minima and the maxima, the expected height of the saddles is $\bar{h}$.

5.2.2 Surface roughness

To analyse our synthetic topographies, we calculate surface roughness measures from transects of random topographies of 400 catchments, across different height amplitudes Δh. As is done in calculating surface roughness in the field, to collect the data in order to quantify the surface roughness of our synthetic topography, we measure the ice surface height at 500 evenly spaced intervals (of length 0.04) along a transect of the surface, and repeat this for many transects. We define the roughness as

$$R_a = \frac{1}{l_r} \int_0^{l_r} |z(x) - \bar{z}| \, \mathrm{d}x \tag{5.5}$$

and the root-mean-square (RMS) roughness

$$R_q = \sqrt{\frac{1}{l_r}\int_0^{l_r} [z(x) - \bar{z}]^2 \, \mathrm{d}x}, \tag{5.6}$$

where l_r is the length of the transect, $z(x)$ are the point heights taken and $\bar{z}$ is a reference height. The first, R_a, is the roughness measure by the L_1 norm, while R_q uses the L_2 norm. Both roughness measures scale with Δh, indicating that floes with larger surface amplitudes are rougher. We find that on average over several realisations with different Δh, $R_a/\Delta h = 0.24$ and $R_q/\Delta h = 0.31$.

The average skew and kurtosis, R_{sk} and R_{ku}, which are the second and third moment equivalents of equation (5.5) were $R_{sk} = 0.24$ and $R_{ku} = 2.87$. Here

$$R_{sk} = \frac{1}{R_q^3}\frac{1}{l_r}\int_0^{l_r} [z(x) - \bar{z}]^3 \, \mathrm{d}x \tag{5.7}$$

and

$$R_{ku} = \frac{1}{R_q^4}\frac{1}{l_r}\int_0^{l_r} [z(x) - \bar{z}]^4 \, \mathrm{d}x. \tag{5.8}$$

The low skewness is in line with what we see in Figure 5.3. The kurtosis close to, but less than 3 suggests that there is a small amount more weight in the peak than in the tails.

If we scale $\bar{h}$ with $\tilde{h} \sim 1$ m and take $\tilde{x} \sim 5$ m (having used $\bar{a}_0 = 1$), then the roughness measures R_a and R_q are on the order of centimetres for smooth ice, and tens of centimetres for rough ice, which are comparable with observations [Nolin and Mar, 2019]. The scaling in roughness with height amplitude suggests that for simulations we can use Δh as a proxy for surface roughness. We will do this in later discussions and informally call Δh 'the roughness'.

5.2.3 Surface correlations

As well as coastlines, mentioned earlier, geophysical surfaces have been shown to have fractal properties. Surfaces such as landforms and sea ice and can be modelled as fractal Brownian surfaces, the first of which is discussed in Mandelbrot and Wheeler [1983] and Family and Vicsek [1991]. These models make use of self-affine fractals, whereby landscapes tend to be random, but self repeating with correlations existing between the heights of points in different locations of the surface. The extent of these correlations can by investigated using an autocorrelation function. This was done by Popović et al. [2020a] to compare the synthetic ice and snow topographies that they

developed to the LIDAR data from the field experiment described by Polashenski et al. [2012].

We define the normalised autocorrelation for surface heights,

$$C_h(\mathbf{l}) = \frac{\langle h(\mathbf{x})h(\mathbf{x}+\mathbf{l})\rangle - \langle h\rangle^2}{\sigma^2(h)} \tag{5.9}$$

where σ^2 is the variance, $\mathbf{l}$ is a lag between surface points and angle brackets denote an expectation over the whole surface. In calculating the autocorrelation function, we essentially shift a copy of the topography by the vector $\mathbf{l}$, and compare how well the shifted topography lines up with the original. By using a range of $\mathbf{l}$ we can determine whether the topography has a propensity to self-repeat, or shows some degree of self-similarity.

The autocorrelation functions for the synthetic topography and LIDAR data used to produce Figure 5.3 are shown in Figure 5.4, plotted against a range of $\mathbf{l}$ in the x direction (the function for the field data is taken directly from Popović et al. [2020a]). To make a comparison with the LIDAR data, we scale the lag by the horizontal length scale for catchments, $\tilde{x} = 5$ m. Due to an expected anisotropy in the field data caused by winds, the autocorrelation is averaged across all directions. In contrast, for the synthetic topography we only present the autocorrelation in the x direction. No significant difference is seen in the y direction, as the method for choosing surface peaks, centres and surface heights is not expected to cause any isotropy.

While the correlation function for the model presented by Popović et al. [2020a] agrees very closely with the field data, correlations for our synthetic topography fall much faster and display very low levels of correlation at any lag greater than around the extent of a catchment. The fast decay of the function could be due to the steepness of the ice at the centres of catchments, which we expect to be smoother in reality, as discussed earlier. The function for our synthetic topography does show some structure and anti-correlation at larger lags, which may be due to peaks lining up with centres.

The synthetic topography that we use does not recreate the surface correlations seen in real ice. However, in Chapters 6 and 7 we show that the model accurately recreates the important behaviour of pond evolution and coverage. This indicates that surface correlations may not be the most important statistic to recreate in synthetic ice topographies for use in models of melt ponds.

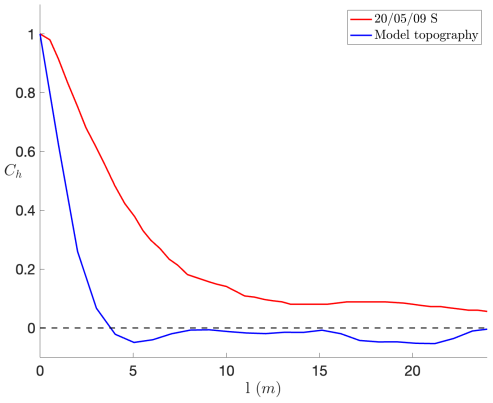
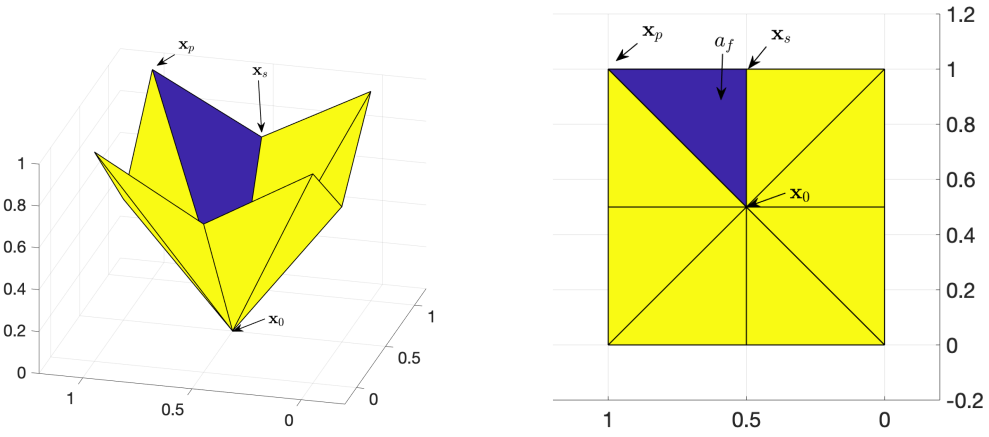


Figure 5.4: Correlation function of surface heights for the synthetic topography (blue) and for the field sites described by Polashenski et al. [2017] (red), with result taken from Popović et al. [2020a].

5.3 Calculating pond area for the triangular mesh

By simplifying the topography to a triangular mesh, we can determine the pond area at time t from the water level $H(t)$. We consider the catchment shown in Figure



(a) (b)

Figure 5.5: A single catchment of a triangular mesh topography, in oblique view (a) and plan view (b). To calculate areas we consider each triangular face in turn. In this figure we have coloured one in blue for reference, and marked important points for reference, including $\mathbf{x}_p$, the location of the peak of this face, $\mathbf{x}_s$, the saddle on this face, and $\mathbf{x}_0$, the centre of this catchment. The total horizontal area of the face, a_f is marked in (b). This illustrative example happens to be a regular square catchment, but in general all 8 (or more) faces may be of different sizes.

5.5. The faces of the mesh divide it into eight distinct regions, seen in the plan view of Figure 5.5(b). In a single catchment area, the water-covered area is the sum of the water covered areas in each face region. We have coloured one face blue for reference and marked important points in the initial topography; $\mathbf{x}_p$, the location of the peak of this face; $\mathbf{x}_s$, the saddle associated with this face, and $\mathbf{x}_0$, the centre of this catchment. Each face contributes to the pond area by an amount equal to the area of the part of the horizontal plane at a height $\tilde{H}(t) = H(t) + H_b(t)$ that lies above the initial topography height $h_0(\mathbf{x})$ (see Figure 5.6(a) and (b)). The translated height $\tilde{H}$ accounts for the downward motion of the whole ice surface by an amount $H_b = \int_0^t v_b \, \mathrm{d}t$. For the constant melt rates used in this chapter, $H_b = v_b t$. We label the two intersection points of the edges of the face and the plane $z = \tilde{H}(t)$ as $\mathbf{p}_p$ and $\mathbf{p}_s$, as shown in Figure 5.6.

The intersection of the horizontal plane with a face can create three possible shapes:

- A triangle, at any stage before the associated edge becomes a join edge, i.e. when $\tilde{H} > z_0$, and $\tilde{H} \leq z_s$. An example of this shape can be seen in Figure 5.6(a) and (b). This shape is the case when the associated edge is in states 1 (inactive), 2 (overflowing to the left) or 3 (overflowing to the right). For the case where the catchment is overflowing by the associated edge, $\mathbf{x}_s$ and $\mathbf{p}_s$ are coincident.

- A quadrilateral, once the associated edge becomes a join edge (state 4), for $\tilde{H} > z_s$ and $\tilde{H} < z_p$. The quadrilateral consists of the saddle point, the pond centre, the point of intersection between the pond boundary, and the peak to centre line $\mathbf{p}_p$, and the point of intersection between the pond edge and saddle-to-centre line $\mathbf{p}_s$. This shape can be seen in Figure 5.6 (c) and (d).

- A triangle consisting of the entire section of catchment, when $\tilde{H} \geq z_p$, the point at which this face floods. This is the area a_f marked in Figure 5.5.

The full area of the face a_f projected onto the plane $z = \tilde{H}$ can be calculated using the cross product of the vectors $(\mathbf{x}_p - \mathbf{x}_0)$ and $(\mathbf{x}_s - \mathbf{x}_0)$ to find the vector area, and then projecting this into the $x - y$ plane, by dotting with the normal $\mathbf{k}$, the unit vector in the vertical direction. Then

$$a_f = \frac{1}{2} \left| (\mathbf{x}_p - \mathbf{x}_0) \wedge (\mathbf{x}_s - \mathbf{x}_0) \cdot \mathbf{k} \right|. \tag{5.10}$$

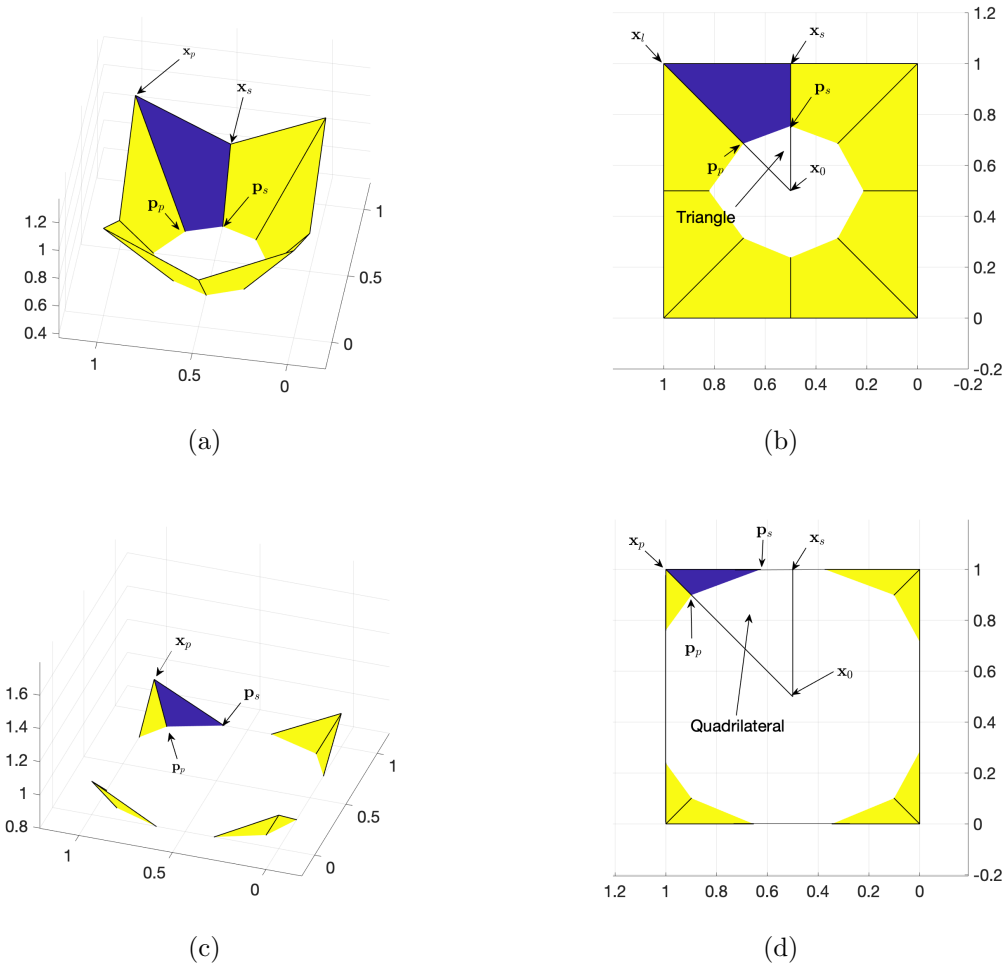


Figure 5.6: A single catchment, where the water-covered part of the blue face has shape that is either a triangle, shown in (a) and (b), or a quadrilateral, shown in (c) and (d). In (a) and (b) the network edge associated with the saddle $\mathbf{x}_s$ is still inactive (state 1). In (c) and (d) the edge associated with the blue face is a joined edge (state 4). The catchment is shown in oblique view (a) and (c) and plan view (b) and (d). The intersection points of the edges of the face and the plane of the water surface are labelled $\mathbf{p}_p$ and $\mathbf{p}_s$.

The triangle shape for segments of catchments, shown in 5.6(b), is a fraction of the full area. To calculate this fraction we scale each vector in the cross product by the ratio of the water level in the pond (relative to the initial centre), to the vertical length of these vectors, giving

$$a = \frac{1}{2}\frac{(\tilde{H} - z_0)^2}{(z_s - z_0)(z_p - z_0)}|(\mathbf{x}_p - \mathbf{x}_0) \wedge (\mathbf{x}_s - \mathbf{x}_0) \cdot \mathbf{k}| = \frac{(\tilde{H} - z_0)^2}{(z_s - z_0)(z_p - z_0)} a_f. \quad (5.11)$$

The quadrilateral shape is shown in Figure 5.6(c) and (d). The intersection point of the plane formed by the water level with the line between the centre and peak,

$\mathbf{p}_p$ is found in the same way as for the triangle, scaling the vector $(\mathbf{x}_p - \mathbf{x}_0)$ by $(\tilde{H} - z_0)/(z_p - z_0)$. The other intersection point $\mathbf{p}_s$, now along the line between the peak and saddle point, is determined by a similar scaling. It lies along the vector $\mathbf{x}_p - \mathbf{x}_s$, with the z component scaled by the fraction $(\tilde{H} - z_s)/(z_p - z_s)$. The intersection point has the position vector $\mathbf{x}_s + (\mathbf{x}_p - \mathbf{x}_s)(\tilde{H} - z_s)/(z_p - z_s)$. The area of the quadrilateral is then given by the sum of the cross products $|(\mathbf{p}_p - \mathbf{x}_0) \wedge (\mathbf{p}_s - \mathbf{x}_0)|$ and $|(\mathbf{p}_s - \mathbf{x}_0) \wedge (\mathbf{x}_s - \mathbf{x}_0)|$. After some manipulation, we find

$$a = a_f \left\{ 1 - \frac{(\tilde{H} - z_p)^2}{(z_p - z_s)(z_p - z_0)} \right\}, \tag{5.12}$$

which can also be thought of as subtracting the blue area in Figure 5.6(d) from the whole area of the face.

We can use the logical restrictions of the different possible shapes to derive the general equation for the area contributed by a face of the mesh. Each of these regions would have the area

$$a_h = \begin{cases} a_f \left[\frac{(\tilde{H} - z_0)^2}{(z_s - z_0)(z_p - z_0)} \right] & z_0 \leq \tilde{H} < z_s \\ a_f \left[1 - \frac{(\tilde{H} - z_p)^2}{(z_p - z_s)(z_p - z_0)} \right] & z_s \leq \tilde{H} < z_p \\ a_f & \tilde{H} \geq z_p, \end{cases} \tag{5.13}$$

and we then calculate the water covered area for a catchment by summing the a_h for each face in the catchment. To summarize, this method gives us a way to calculate the area a_i for each catchment in terms of the initial topography and the time-dependent water levels $H_i(t)$. We refer to this method as the 'area function' and write $a_i(t) = f(H_i(t), t)$ as in (3.47) and (3.58).

5.4 Initial water levels

To avoid the singularity in the water level equation when $a_i = 0$, we use an asymptotic solution to start simulations at a small time $t_0 > 0$, as was done in Chapter 3. Provided t_0 is small enough that no saddles have been reached, (5.13) indicates that all the areas are of the form

$$a_i = (H_i - z_{0i})^2 C_i, \tag{5.14}$$

where $C_i = \sum c_k$ with the sum taken over all faces in catchment i, and the c_k for each face region are given (as in (5.13)) by

$$c_k = \frac{1}{2} \frac{|(\mathbf{x}_p - \mathbf{x}_0) \wedge (\mathbf{x}_s - \mathbf{x}_0)| \cdot \mathbf{k}}{(z_s - z_0)(z_p - z_0)}. \tag{5.15}$$

We can write the water level equation (4.33) in the form

$$\dot{H}_i = \frac{\rho_{iw} v_b a_{0i}}{(H_i - z_{0i})^2 C_i} - \eta, \tag{5.16}$$

where $\eta = [v_w - \rho_{iw}(v_w - v_b)]$. For early times, the first term on the right dominates, so

$$H_i(t) - z_{0i} = \left(3\frac{\rho_{iw} v_b a_{0i} t}{C_i}\right)^{1/3}, \tag{5.17}$$

from which we calculate initial water levels at a small time (for example $t_0 = 10^{-4}$), and start the simulation from there. Alternately, equation (5.16) above has, for $H_i(0) = z_{0i}$, the implicit solution

$$t = \frac{1}{\eta}\sqrt{\frac{\rho_{iw} v_b a_{0i}}{\eta C_i}}\operatorname{arctanh}\left([H_i - z_{0i}]\sqrt{\frac{\eta C_i}{\rho_{iw} v_b a_{0i}}}\right) - \frac{H_i - z_{0i}}{\eta}. \tag{5.18}$$

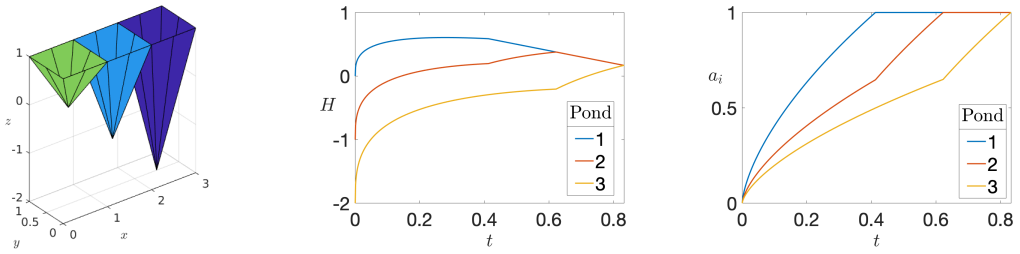
Using a root-finding method we can calculate the small water level in each catchment, $H_i(t)$ according to equation (5.18), for a small time t before pond interactions begin. In practice, this method is more accurate than the expansion for small time.

5.5 Example model runs

This section describes several example simulations. Details of the numerical implementation are given in Appendix A, including how the model equations are integrated, how the different events (changes or edge states) are located and accounted for, and how loops are broken. Section 5.5.1 describes an analytical test of the simulation algorithm using a floe of three square pyramids with uniform saddle heights, which tests how effectively the algorithm deals with coupling of catchments and changes of state for edges. Section 5.5.2 considers a more complicated floe, a 3 by 3 array of nine square pyramids with non-uniform saddle heights. This experiment is complex enough to show the possible features of a simulation while still allowing the reader to distinguish individual events (the beginning of overflow of a catchment, the creation of a joined pond, etc.) in order to build intuition for the data produced in a simulation. Finally Section 5.5.3 describes a simulation with the topography of one hundred catchments built as described in Section 5.1. This simulation shows the behaviour of the data we expect for the more targeted experiments described in Chapters 6 and 7, particularly analysing the effects of drainage and exploring the behaviour of the model using different parameters.

5.5.1 Analytical tests

The behaviour of the model in a single catchment, in the shape of an inverted square pyramid was tested analytically in Chapter 3. To test the connectivity features of the algorithm - overflow fluxes and join fluxes - we build a topography on three adjoining pyramids. Each has side length 1, and the depths are 1, 2, and 3 units, arranged side by side one after the other, as shown in Figure 5.7. We define edges from pyramids 1 to 2 and 2 to 3. The simulation is designed such that the leftmost pyramid (1) will fill first, overflow into the middle (2), which later joins with the leftmost and overflows into the rightmost (3), testing each type of flux.



(a) (b) (c)

Figure 5.7: (a) Initial topography for a simulation using three square pyramids each of side 1 and of heights 1 (pyramid 1, in green), 2, (pyramid 2, in blue) and 3 (pyramid 3, in purple). Edges exist from 1 to 2 and from 2 to 3, such that water can flow amongst the three catchments. Panels (b) and (c) show evolution of water levels and water covered areas respectively. Melt rate parameter $v_w = 2.5$ was used.

Modifying equation (3.46) for the area of a two dimensional catchment labelled i, the area of a pond with perimeter p_i in an inverted square pyramidal catchment of face slope s_i grows as

$$\dot{a}_i = \left(\frac{\rho_{iw} v_b a_{0i} - \sum_j q_{ij}}{a_i} - [1 - \rho_{iw}][v_w - v_i] \right) \frac{4\sqrt{a_i}}{s_i}, \tag{5.19}$$

and (5.19) can be rewritten

$$\dot{a}_i = \frac{\zeta_i}{\sqrt{a_i}} - \eta_i \sqrt{a_i}, \tag{5.20}$$

where $\zeta_i = 4(\rho_{iw} v_b a_{0i} - \sum_j q_{ij})/s_i$ and $\eta = 4(1 - \rho_{iw})(v_w - v_i)/s_i$. Equation (5.20) has implicit solution

$$t = \frac{2}{\eta_i}\sqrt{\frac{\zeta_i}{\eta_i}} \operatorname{arctanh}\left(\sqrt{\frac{\eta_i a_i(t)}{\zeta_i}} \right) - 2\frac{\sqrt{a_i}}{\eta_i} + \phi_i \tag{5.21}$$

where we use the initial condition expression

$$\phi_i = t' - \frac{2}{\eta_i}\sqrt{\frac{\zeta_i}{\eta_i}}\text{arctanh}\left(\sqrt{\frac{\eta_i a_i(t')}{\zeta_i}}\right) + 2\frac{\sqrt{a_i(t')}}{\eta_i} \tag{5.22}$$

for some initial time t' and corresponding area $a_i(t')$.

The initial conditions for numerical simulations were calculated as in Section 5.4. Figure 5.7 shows the behaviour of the three pyramids simulation. The expected events are visible in both figures: pyramid 1 fills first, at time t_1, then overflows into 2, causing an acceleration of its water level increase and area growth. At this point

$$q_{12} = \rho_{iw} v_b a_{01} - (1 - \rho_{iw})(v_w - v_b)a_{01} \tag{5.23}$$

with $a_1 = a_{01}$ as catchment 1 has flooded. These pyramids then join at time $t_1 + t_2$ (seen by the area of pyramid 2 reaching a value of 1), and then overflow into pyramid 3, at which point we see the largest change in area growth. Then,

$$q_{23} = \rho_{iw} v_b (a_{01} + a_{02}) - (1 - \rho_{iw})(v_w - v_b)(a_{01} + a_{02}). \tag{5.24}$$

Once pond 3 joins with 2 at time $t_1 + t_2 + t_3$, the whole domain is flooded, and we end the simulation.

The time for the three pyramids to flood was calculated by stitching together stages of equation (5.21) calculating fluxes and initial conditions at the events identified for the simulation. The time t_1 was calculated as the time taken for catchment 1 to fill ($a_1 = 1$), with $q_{12} = 0$. At this point, $a_2(t_1)$ and $a_3(t_1)$ were calculated. To find t_2, the time taken from t_1 to the time when catchment 2 filled was calculated, given initial condition $a_2(t_1)$ and flux (5.23). The area $a_3(t_2)$ was calculated, given the initial condition $a_3(t_1)$. Finally, the time t_3 was calculated as the time from t_2 to when catchment 3 fills, subject to initial condition $a_3(t_2)$ and incoming flux (5.24). The calculated time to flood ($t_1 + t_2 + t_3$) and simulated time to flood agreed to within 10^{-8}.

5.5.2 A regular lattice topography

We next consider an experiment using a topography generated by the method outlined in Section 5.1. For simplicity we choose a regular 3 by 3 lattice of square ponds, each of side length 1 (instead of randomly assigning pond centre locations and using the Voronoi tessellation) on which to run the simulation. Catchments in this topography have constrained random catchment depths and saddle heights, as described in Section

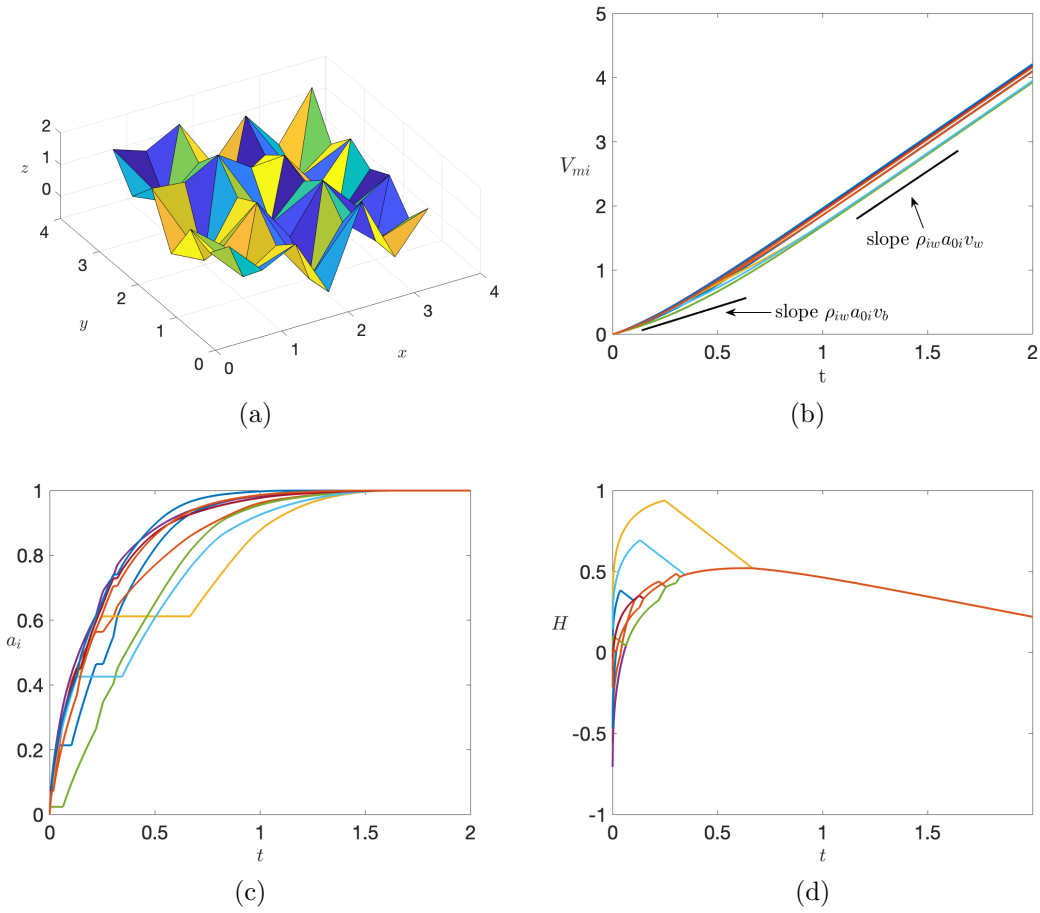


Figure 5.8: Results from a numerical simulation on a floe of nine catchments. (a) Initial topography of nine square catchment areas (note, different colours within catchments merely mark different faces). (b) Volume of melt produced in individual catchments for the nine catchments in the simulation. Asymptotes for the rate of change of melt volume are marked. These are based on observations that bare ice melt dominates at early time, while water covered melt dominates at later times. (c) Areas and (d) water levels, respectively, of individual catchments. Melt rate parameter $v_w = 2.5$ was used.

5.1.2. Plots of the melt volume, pond area and water levels in each catchment are shown in Figure 5.8, as well as a rendering of the initial topography. Figures 5.9 and 5.10 show an overhead view of the floe at each event (change of state of an edge) during the simulation. Pond extent and the state of each edge are marked in the overhead views.

Examining the melt volume in Figure 5.8(b), we see low rates of increase at early times, as there is little water cover on the ice, and the rate of change of volume

asymptotes to

$$\dot{V}_{mi} \approx \rho_{iw} a_{0i} v_b, \tag{5.25}$$

which is marked in the plot. After 0.5 time units, we see a steepening, as melting within ponds starts to dominate, as most (and later all) of the floe is melting with rat that asymptotes to

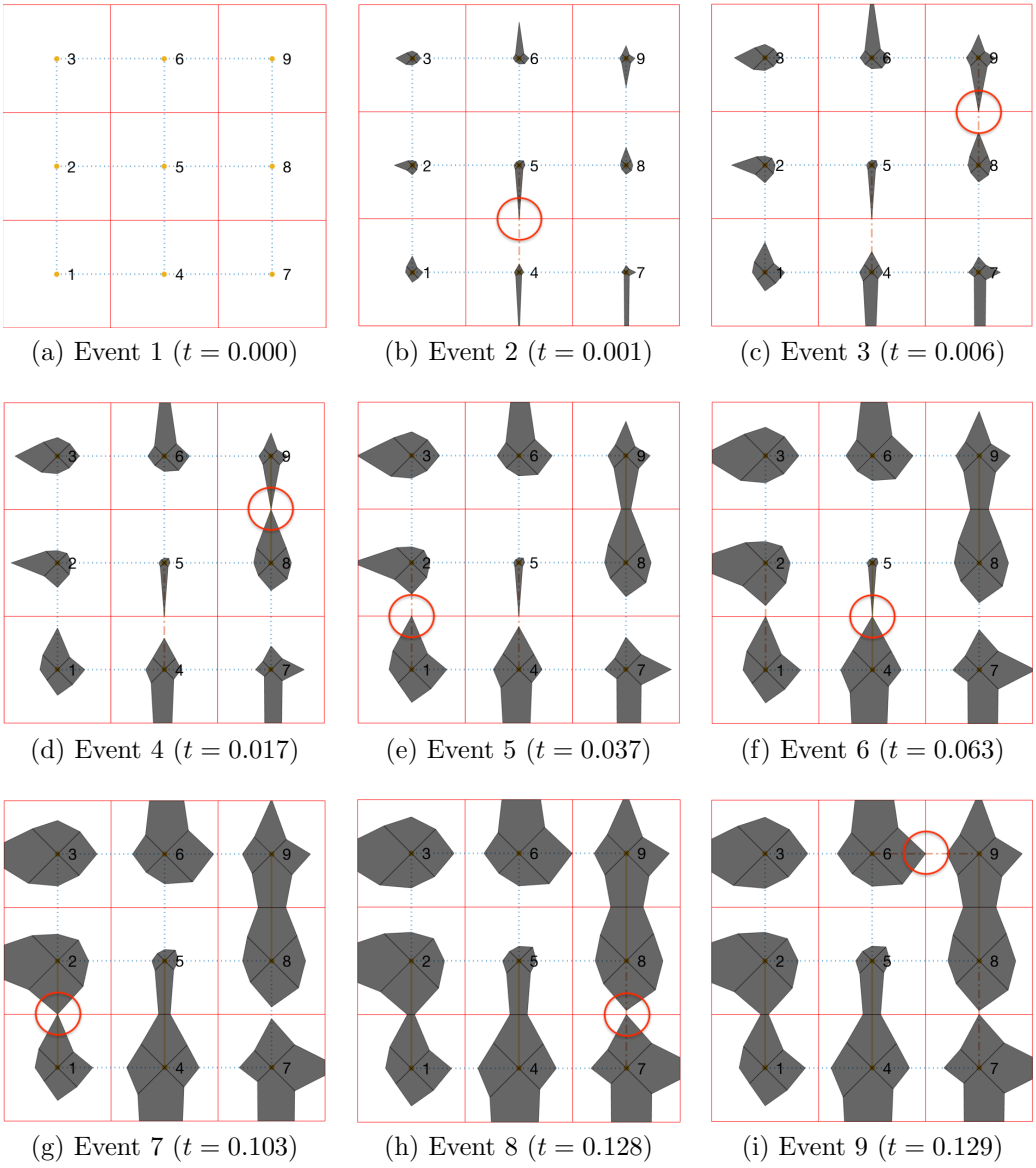
$$\dot{V}_{mi} \approx \rho_{iw} a_{0i} v_w. \tag{5.26}$$

We note that all catchment areas a_{0i} are equal in this example.

Examining the evolution of the area cover over time, we see from Figure 5.8(c) that the total area fraction goes to 1 in finite time and (by construction) goes no further. We can also see the fingerprints of each event (overflows, joins occurring), as described below. All catchments display a fast increase in area cover at early times, and much slower increases at later times as the area fraction approaches 1. Smooth increases denote catchments that are either independent, or joined to other catchments, and not overflowing to others. Horizontal segments correspond to ponds that are overflowing. A sudden steepening of an area curve denotes that it is the recipient of an overflow flux. The timing of these events can be seen clearly where there are discontinuities in the slopes of water level plots (d). A shallowing of a recipient catchment's curve in (c) denotes the emergence of a join edge where the overflow was. Likewise an elbow in the curve of an overflowing catchment, from flat to increasing denotes that the overflowing edge has become a join.

Similar signatures can be seen in the plot of water levels, Figure 5.8(d). Smooth increases in level denote independence. Steepening of increases denotes an incoming overflow flux. A straight downward-sloped segment of the curve denotes overflow into another catchment. The start of an overflow and the end can both be seen when there is a discontinuity in the slope of water level trajectories. Finally, convergence of curves denotes that the catchments have joined, as the levels of the catchments have equalised, as expected from the model.

Each of the events that can be identified from Figures 5.8(c) and (d) can be seen in plan view (from above), with the network structure and a visualisation of pond extent in Figures 5.9 and 5.10. For example, in event 2, catchment 5 overflows into catchment 4. In event 4, catchments 8 and 9 join. Event 19 refers to the joining of catchments 3 and 6, and is the more prominent coalescing of yellow and red curves around $t \approx 0.7$ in Figure 5.8(d).

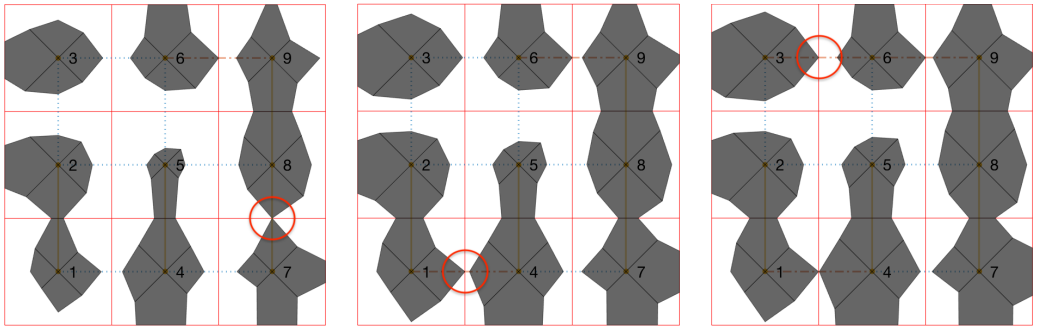


(a) Event 1 ($t = 0.000$) (b) Event 2 ($t = 0.001$) (c) Event 3 ($t = 0.006$)

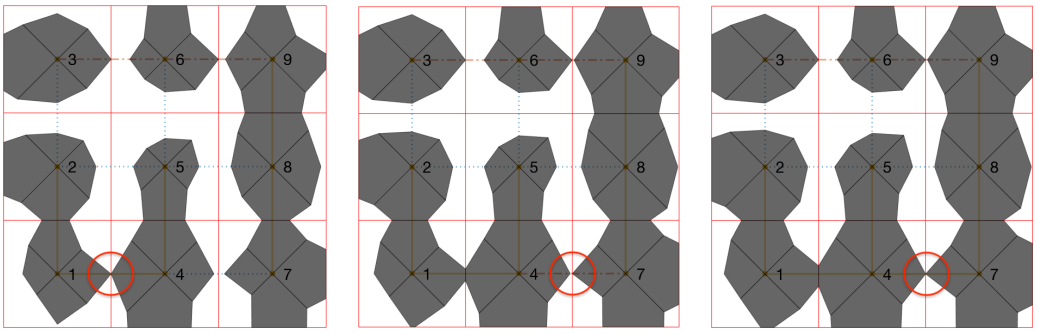
(d) Event 4 ($t = 0.017$) (e) Event 5 ($t = 0.037$) (f) Event 6 ($t = 0.063$)

(g) Event 7 ($t = 0.103$) (h) Event 8 ($t = 0.128$) (i) Event 9 ($t = 0.129$)

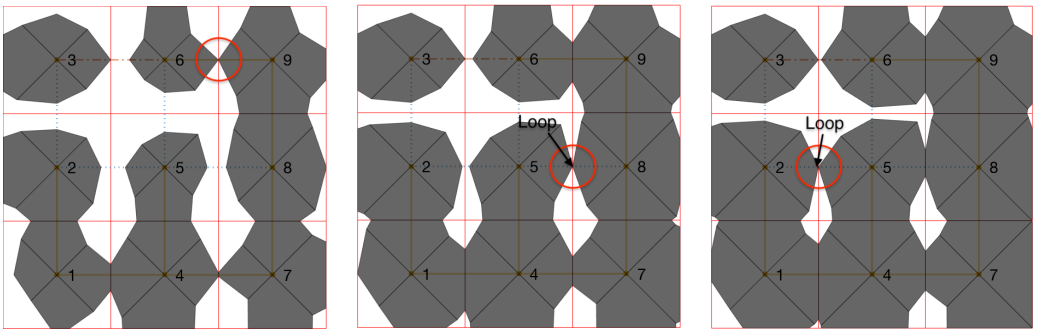
Figure 5.9: Events 1 to 9 for the nine catchment experiment, showing network structure and area cover of ponding. Solid lines between centres denote join edges, dashed lines denote overflow edges. Dotted lines denote inactive edges. Greyed areas denote ponding. The solid diagonal lines within ponds denote the lines along the topography from centres to peaks. Events are circled in red. Example events include an overflow beginning (event 3) where catchment 9 overflows into catchment 6, and the same overflow becoming a join (event 4). Note also that events are not equally spaced in time. No overflow is allowed into the sea.



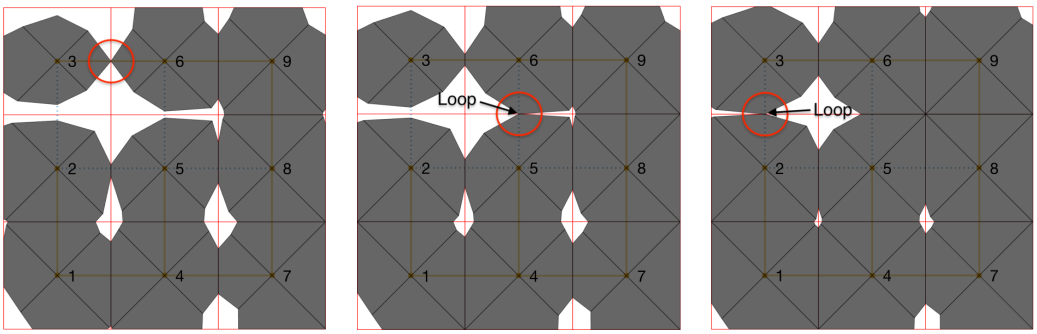
(a) Event 10 ($t = 0.147$) (b) Event 11 ($t = 0.219$) (c) Event 12 ($t = 0.248$)



(d) Event 13 ($t = 0.253$) (e) Event 14 ($t = 0.302$) (f) Event 15 ($t = 0.321$)



(g) Event 16 ($t = 0.344$) (h) Event 17 ($t = 0.525$) (i) Event 18 ($t = 0.611$)



(j) Event 19 ($t = 0.667$) (k) Event 20 ($t = 0.752$) (l) Event 21 ($t = 0.923$)

Figure 5.10: Events 10 to 21 for the nine catchment experiment. Note events without edge changes such as event 17, where loops have been cut. Otherwise event 17 would have represented a join forming between catchments 5 and 8.

5.5.3 A larger, more irregular topography

Figure 5.11 shows output of a simulation on a floe with 100 catchments. The centre points are distributed on the plane uniformly at random as described in Section 5.1.1. The same behaviours as in the simpler case of 9 catchments are seen for total area and total volume of water produced. The behaviours of the individual ponds differ, as the topography of random centres is much more irregular. The number of catchments makes the plots of water levels, areas and volumes too complicated to be useful, so we will pay more attention to the bulk measure of the 'pond fraction', $\sum_i^n a_i / \sum_i^n a_{0i}$. When carrying out experiments on these large simulations later we will explore different analytical and statistical techniques, including analysing the network structure, and using methods from percolation theory.

Figure 5.11(d) shows the number of edges in each state during the simulation. Note, in this figure, join edges that would usually be made inactive due to loop formation are counted as active join edges, in order to represent the system as it should be physically, rather than how we need to write it in order to solve for the fluxes. For simplicity we have included both directions of overflow edge (leftwards and rightwards) together as overflow edges. We see that at early time there is a small spike in overflow edges as water levels rise, with an accompanied fall in the number of inactive edges. The number of overflow edges then decays to zero over time as join edges begin to form and overflows are eventually covered by rising water levels. Meanwhile, the number of inactive edges continues to decrease. The number of independent edges decays to zero as the floe floods, and every edge becomes a join edge by $t = 1.5$. At this point the configuration becomes static.

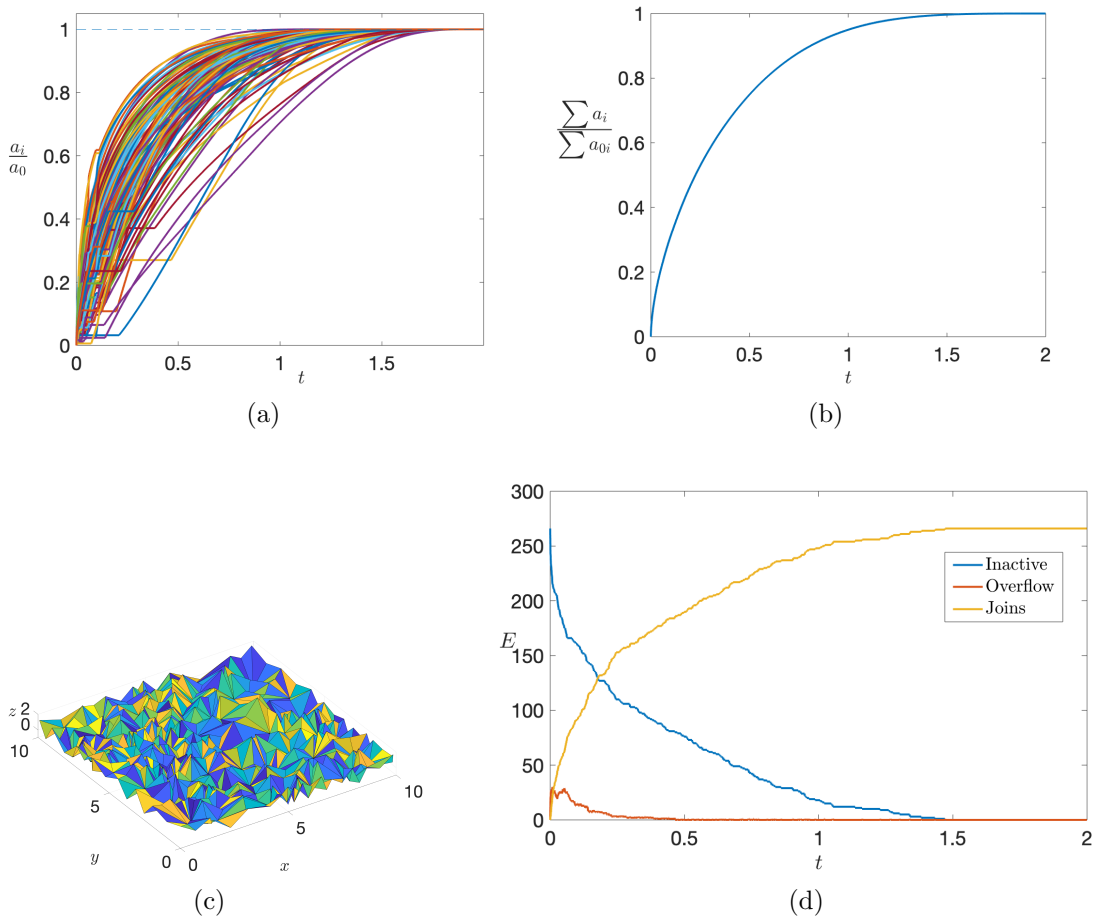


Figure 5.11: Results for a numerical simulation on a floe of 100 catchments. (a) Area fraction in individual catchments. (b) Total area fraction of ponding. (c) Initial topography. Colours denote different faces of the surface. (d) Plot of number of edges in each state. Independent edges are shown in blue, overflows in red and joins in yellow. Melt rate parameter $v_w = 2.5$ was used.

5.6 Behaviour at the exterior boundaries

We return to the possibilities discussed in Section 5.1.3, whereby two choices present themselves for the behaviour of saddles on the exterior boundaries of the floe. The first and simplest case is to assume vertical walls at the exterior, and not connect the edges associated with these saddles to any other nodes. The second is to connect each of the edges to another 'sea' node. To compare these, we set up experiments, each with nine realisations of random initial topographies, and ran simulations both with and without connecting boundary catchments to the ocean node.

Figure 5.12 shows the mean trajectories of area fraction (a) and melt volume

(b) with ensemble standard deviations shaded for three sets of nine topographies. Each set uses a different numbers of nodes. In panel (a), the upper set of curves, the trajectory for simulations without drainage, always shows the system moving towards flooding of the domain. Meanwhile, the lower set of curves shows an area fraction that plateaus around the area at which the pond network percolates, $a(p_c)$, shown by the horizontal dashed lines. We define $a(p_c)$ as the area fraction of the pond system at the earliest time at which a connected component of the network forms that spans from one side of the domain to the opposite side (this includes overflow and join edges). The plateau of area corresponds to a state in which each catchment is connected by a join or overflow to a catchment at a side of the floe.[2] Due to the way in which overflowing limits area expansion, the area fraction becomes constant in this state. The average maximum area for each set of experiments is close to but generally below the average area at percolation. This might be because when each side of the floe is open, a spanning cluster (a large pond or system of connected ponds that would span from one side of the floe to the opposite side) need not develop in order for every catchment to drain.

The open area fraction trajectories show some small finite-size effects for the $n = 100$ mean trajectory, which deviates slightly from the $n = 200$ and $n = 300$ trajectories. The $n = 200$ and $n = 300$ trajectories are much closer together, indicating that for simulations of these sizes, finite size effects have almost disappeared. Given that the mean trajectory of the $n = 100$ experiment set lies within one standard deviation of the $n = 200$ and $n = 300$ trajectories, we also conclude that finite-size effects can be considered negligible for experiments with $n = 100$.

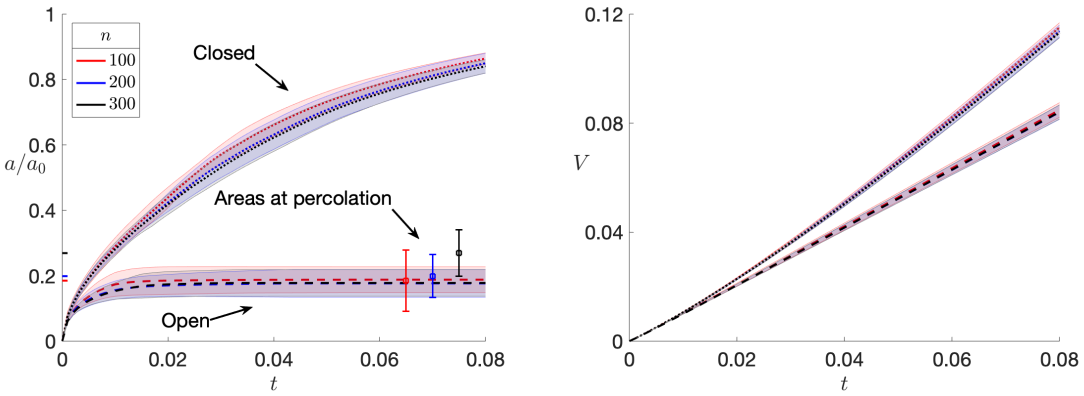
The melted volume, shown in Figure 5.12(b) is lower for the open case, which follows naturally from the suppressed area fraction. Due to the plateau in area fraction, at later time the melt volume increases linearly. The melt volume changes according to equation (4.42)

$$\dot{\mathbf{V}}_m = \rho_{iw} v_b (\mathbf{a}_0 - \mathbf{a}) + \rho_{iw} v_w \mathbf{a},$$

which is a constant rate when all elements of $\mathbf{a}$ are constant.

Figure 5.13 shows the edge fractions for each type of edge for the experiments with 300 catchments, averaging over nine ensemble members. The case without drainage is shown in (a) and the case with drainage is shown in (b). Panel (a) shows the type of behaviour shown in example simulations from Section 5.5.2, whereby the number of inactive edges decays throughout the experiment, while the number of joined edges

[2]The system described here would resemble a cascading water feature, an elaborate fountain, perhaps.

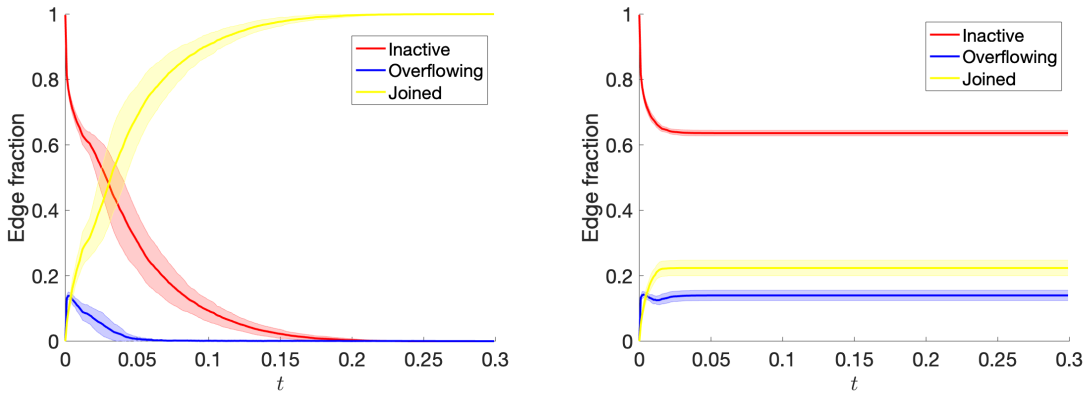


(a) (b)

Figure 5.12: Trajectories of average areas (a) and melt volume (b) for nine ensemble experiments with no exterior boundary drainage (upper branches, dotted), referred to as 'closed', and with drainage (lower branches, dashed), referred to as 'open'. Shaded areas denote the region within one standard deviation of the ensemble. The line colours indicate number of catchments, and are consistent between the two plots. Error bars in (a) show the means and standard deviations of the areas at which percolation occurs for the topographies used. For clarity, the means are also marked on the y axis. The melt rate parameter $v_w = 2$ was used.

increases. The network only contains a significant number of overflow edges for a brief period at the start of the simulations while pond systems are developing. In contrast, because of the water loss to the ocean, in panel (b) the numbers of each type of edge plateaus to finite values. There is a considerable amount of overflow occurring even at later times as water flows continuously into the sea. We note that the total fraction of these overflowing and joined edges is around the edge fraction of random percolation (0.33 for a Delaunay triangulation), so we might initially expect the area fractions seen in Figure 5.12(a) for draining simulations to be at or above that of the percolation threshold. However, we will explain in Chapter 7 that in our model, percolation is suppressed due to the behaviour of overflowing edges, and occurs at a higher edge fraction than expected.

This locking of the state of the system in the way shown here has not been seen in observations and is unlikely to be relevant for real ice due to the short lived nature of the impermeable phase of pond evolution. This will be addressed in Chapter 6 when we make the ice thickness finite and introduce drainage through the base. No difference in the plateau phase of area fraction or edge states were found in experiments with side drainage when either the enhanced melt rate or the surface



(a) Without drainage (closed)

(b) With drainage (open)

Figure 5.13: (a) Average edge fractions for the experiments of 300 catchments, with standard deviations shaded and no drainage. Nine ensemble members were used. Inactive edge fractions are represented in red, overflow edges in blue and join edges in yellow. (b) Average edge fractions for the same experiment with drainage. Note that edges that would form loops are included in the join edges.

roughness were changed.

We can conclude that drainage over the side of floes bounds the maximum area fraction during the initial melt phase while ice is impermeable. We note from Figure 5.13(a) that in each case presented, the average area at which floes that are allowed to drain plateau is at or less than the corresponding area at percolation for those floes, suggesting that in each case, the systems might not reach percolation.

5.7 Summary

In this chapter we have described the method used to construct synthetic topographies on which we can run numerical simulations of our model. We have designed the triangular-mesh-based topography with the aim of easily superimposing the network structure for our model on it. The topography was compared to field data, and was found to adequately represent the surface of real ice.

We discussed in detail how simulations are set up and run for a network of ponds. An analytical test of the numerical simulation was carried out for a simple topography with just three catchments in the shape of inverted square pyramids. The agreement of the results of this simulation with analytical predictions provides confidence in the numerical implementation of our model. A simulation of intermediate complexity, consisting of a square lattice of 9 catchments, and a more complicated simulation

with 100 catchments with randomly placed centres, were also considered in order to illustrate the metrics we will use to analyse the behaviour of our model. In particular, in the coming chapters we will make use of data from the model on pond area, water levels and melt volumes, and the states of the edges in the network.

We also introduced a simple form of drainage at the boundary edges, and carried out experiments that correspond to floes that are open to drain to the sea over their sides. These experiments showed that when drainage is allowed, the pond fraction is much smaller and reaches a plateau, which differs from the continual growth seen when the melt water is contained. This lower area fraction is shown to be around the area at percolation for the system.

Chapter 6

Drainage

In this chapter we tackle one of the characteristics of sea ice that causes much of the unusual behaviour and some difficulty for the modeller - the fact that the ice becomes porous after an initial impermeable stage, which allows pond water to drain. We also address the reality of sea ice as a layer of finite thickness, floating on the ocean surface. Ponds will eventually melt through this finite layer, which we have not yet considered. Once the ice becomes porous and can exchange water with the ocean, determining the draft of the ice floe becomes important to predict the appropriate hydrostatically balanced state. We explore how porous drainage affects systems of ponds across a whole floe.

6.1 Modelling porous drainage

6.1.1 A drainage model for a single catchment

As described by Polashenski et al. [2012], after an initial period of growth, ponds eventually drain to sea level, either through cracks or through sufficiently porous ice. Given that this is an important control of pond extent, we need to incorporate the physics of porous drainage in our model. So far we have also neglected the effects of buoyancy and the finite thickness of the ice. Since the model without drainage conserves the combined mass of ice and water, sea level was irrelevant; however, it becomes important once we allow ponds to drain, as the draft of the floating ice floe changes to accommodate the decreasing weight of the floe. Further, the water levels in draining ponds equilibrate to a level slightly above sea level [Polashenski et al., 2012], creating a feedback process between drainage and buoyancy which we must take care to represent correctly.

We consider a floe with one catchment, shown in Figure 6.1, of base area a_0 and ice surface elevation $h(\mathbf{x}, t)$. For now, we return to dimensional variables. We now measure heights from the base of the floe (shown in Figure 6.1) as opposed to measuring from an arbitrary point in the ice as done before, and denote the draft, (sea level relative to the bottom of the ice) $b(t)$. As in Chapter 4, the water level H

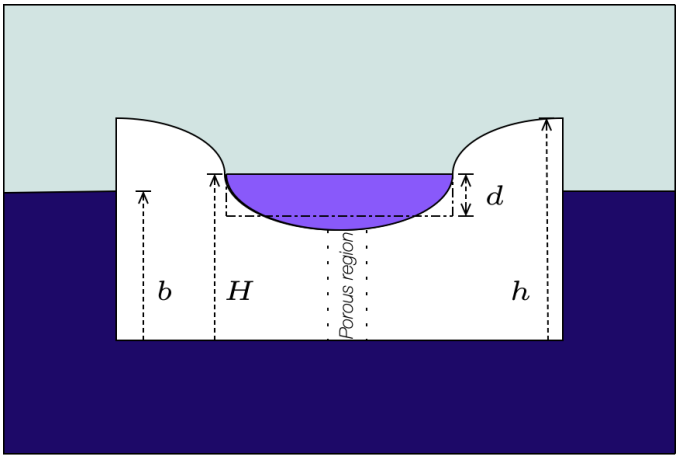


Figure 6.1: Diagram of a single catchment. Heights are measured from the base of the floe, including the draft b, ice surface height h and water level H. Also marked is the volume averaged depth d. The pond will eventually drain through the porous region near the centre.

behaves according to equation (4.24),

$$\dot{H} = \frac{\rho_{iw} v_b a_0 - q}{a} - [v_w - \rho_{iw}(v_w - v_b)] \tag{6.1}$$

where q is the flux of water volume per unit time out of the pond. Here, as earlier, $\rho_{iw} = \rho_i/\rho_w$, and likewise, later in the chapter we will use $\rho_{sw} = \rho_s/\rho_w$, where ρ_s is the density of ocean water (ρ_{sw} is slightly larger than 1 at around 1.03). In this case of a single catchment, q is the drainage flux through the base of the pond.

According to Archimedes, the floe will sit with draft $b(t)$ such that the upthrust matches the combined weight of the ice and pond water,

$$\rho_s g a_0 b(t) = \rho_i g \iint_{a_0} h(\mathbf{x}, t) \mathrm{d}a + \rho_w g \iint_{a} [H(t) - h(\mathbf{x}, t)] \mathrm{d}a, \tag{6.2}$$

where we have assumed that the floe has vertical sides. We recall that the ice surface melts according to

$$\frac{\partial h}{\partial t}(\mathbf{x}, t) = \begin{cases} -v_b, & h(\mathbf{x}, t) \geq H(t) \\ -v_w, & h(\mathbf{x}, t) < H(t). \end{cases} \tag{6.3}$$

We assume that there is no melting of the base of the floe. By differentiating (6.2) and using (6.3) and (6.1),

$$\rho_s g a_0 \dot{b} = -\rho_w g q. \tag{6.4}$$

Hence,

$$\dot{b} = -\frac{q}{\rho_{sw} a_0}. \tag{6.5}$$

We assume that the flux q can be related to the water level H and draft b, by the expression

$$q = K(H - \rho_{sw} b), \tag{6.6}$$

where K is a constant. To motivate this choice, we consider vertical Darcy flow through a porous cylinder of ice starting at the floor of the pond, with cross-sectional area a_d. If the porous medium has permeability k and the viscosity of the water is μ, then Darcy's law states

$$q = \frac{a_d k}{\mu}\left(\rho_w g + \frac{\partial p}{\partial z}\right). \tag{6.7}$$

Here p is the pressure, which varies from $p = \rho_w g(H - h_c)$ at the bottom of the pond to $p = \rho_s g b$ at the bottom of the floe ($z = 0$). We neglect any effects of possible temporal or spatial variability in atmospheric pressure. The draining area a_d is assumed small enough that the ice thickness can be approximated by its value h_c at the pond centre. Note that we ignore the effects of brine flushing [Notz and Worster, 2009], assuming that pores in the ice are only occupied by fresh pond water once drainage begins, rather than initially occupied by brine which is flushed through by the pond water after a short period. To calculate the total flux, we rearrange and integrate (6.7) across the ice thickness, giving

$$\int_0^{h_c} \frac{\mu}{a_d k} q - \rho_w g \, \mathrm{d}z = \int_0^{h_c} \frac{\partial p}{\partial z} \mathrm{d}z = \rho_w g(H - h_c) - \rho_s g b, \tag{6.8}$$

which gives

$$q = \frac{a_d g k}{\mu h_c}(\rho_w H - \rho_s b). \tag{6.9}$$

This can be written as equation (6.6) with $K = a_d g k \rho_w / \mu h_c$. Although h_c (and in principle also a_d) will change through time (so that K likely increases with time), we treat it as constant, for simplicity. Based on field measurements taken by Polashenski et al. [2017] and on some numerical experiments in the next section, the drainage parameter can be taken to be 'large'. We will see that its precise value is of relatively little importance. We take as standard values $\mu = 1.8 \times 10^{-3}$ Pa s, $\rho_w = 1000$ kg m^{-3} and gravity $g = 9.8$ m s^{-2}. From Zhu et al. [2006], the permeability $k \sim 3\phi^3 \times 10^{-8}$

m^2, where ϕ is the porosity of the porous medium, and from measurements described by Polashenski et al. [2017], the permeability ranges from $\mathcal{O}(10^{-9})$ to $\mathcal{O}(10^{-12})$ m^2. For the porous region, we choose $k \sim 3\times10^{-10}$ m^2 (from Polashenski et al. [2017]), and assume a cross-sectional area of 1 m^2. Given the uncertainty in the ice thickness at which drainage begins, we take h_c to be 0.5 m. This is chosen based on the thickness scale for the floe, as a compromise between the possibility of very thin and very thick porous regions. Using these example values, $K \approx 1.1 \times 10^{-3}$ m^2 s^{-1}.

We now return to non-dimensional variables. To non-dimensionalise, (6.6), we scale q with $\tilde{v}\tilde{x}^2$ and H and b with $\tilde{h}$ and the equation becomes

$$q = \hat{K}(H - \rho_{sw} b), \tag{6.10}$$

where

$$\hat{K} = \frac{K\tilde{h}}{\tilde{v}\tilde{x}^2}. \tag{6.11}$$

Equations (6.1), (6.3) and (6.5) appear the same in non-dimensional form. We choose the thickness scale to be the average floe thickness, such that $\tilde{h} \sim 1$ m, $\bar{h} = 1$ and Δh is a fraction of $\bar{h}$. Using the typical scales from Chapter 3 of $\tilde{v}$ of 1 cm/day, or 1.15×10^{-7} m s^{-1}, $\tilde{x}$ of 5 m and the value of K obtained above, we find $\hat{K} \approx 10^3$.

According to Golden [2003], in the absence of desalination, the porosity of sea ice is primarily controlled by temperature. With the choice of constant melt rates, our model is not directly dependent on temperature. The analysis of Taylor and Feltham [2004] shows that under solar forcing, the ice temperature under a pond is strongly dependent on ice thickness. Hence we conclude that as the ice underneath the pond gets thinner, it gets warmer and therefore the porosity likely increases. This was investigated in detail by Polashenski et al. [2017], who found that the permeability increases to between $\mathcal{O}(10^{-10})$ to $\mathcal{O}(10^{-12})$ m^2 once the bulk ice underneath the pond warms to -1.6 C and becomes considerably larger once the temperature increases to -0.9 C.

With this in mind, we will use ice thickness as a proxy for temperature, and we assume that drainage begins once the ice thickness at the pond centre reaches a critical value h_{crit}. At present, the value of h_{crit} is treated as unknown, but the thickness may be somewhere as deep as 60 cm below the ice surface, which in our non-dimensional model would correspond to a value of 0.4 [Polashenski et al., 2012]. However, the thickness might be very close to the surface, corresponding to non-dimensional values as high as 0.8 [Eicken, 2002].

In this case we can take the flux q to be given by

$$q = \begin{cases} 0, & h_c > h_{crit} \\ \hat{K}(H - \frac{\rho_s}{\rho_w} b), & h_c \leq h_{crit}. \end{cases} \tag{6.12}$$

As long as a pond exists in the catchment, the thickness at the pond centre, h_c evolves according to

$$\dot{h}_c = -v_w, \tag{6.13}$$

so with this model there is a unique time at which we transition from $h_c > h_{crit}$ (no drainage) to $h_c \leq h_{crit}$ (drainage begins).

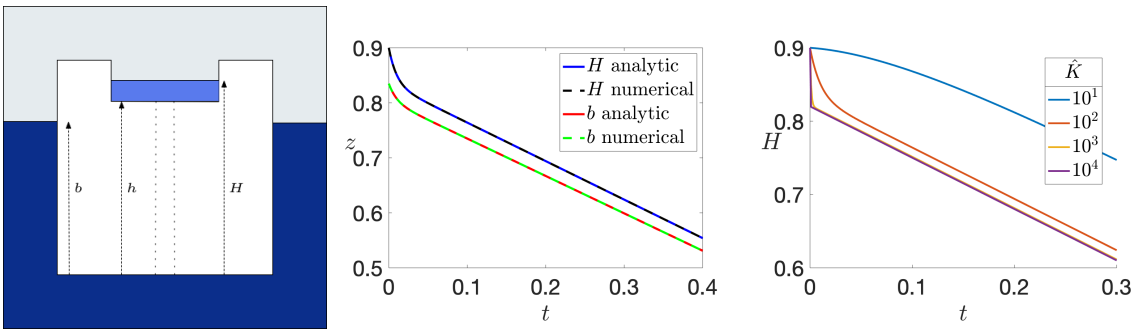
In our choice of model for drainage, we have assumed that the sea water under the floe is salty and water in the ice is fresh. Given how these layers will sit one on top of the other, we are led to ask whether mixing would occur by diffusion, influencing the density in the freshwater layer and requiring a modification to equation (6.6). However, the molecular diffusivity of NaCl salt in water is around 1×10^{-9} m^{2}s^{-1}. By diffusion alone, the salt would take more than a year to reach the top of a pond a metre above the base of the floe. We also neglect any horizontal permeability in the ice, arguing that by the time this becomes important, the water levels will have relaxed to sea level, and the vertical porous drainage will capture any changes to the behaviour that a horizontal permeability would make.

6.1.2 An analytic solution

Unlike in previous chapters, we cannot easily use the square pyramid to derive an analytical solution for the model with drainage. During pond filling, the pond area was always increasing or static, and the growth of the pond only depended on the initial slope of the bare ice that the pond boundary would move over. The history of melting in the pond floor was not accounted for. However, once drainage begins, the pond area can recede, and the pond boundary moves back over what was previously the pond floor, which has a different slope to the initial topography due to preferential melting in the pond (and the non-constant rate of travel of the pond boundary as the pond grew). To derive an analytical example of drainage, we therefore fix the pond area, effectively considering a cylindrical pond with vertical sides (see Figure 6.2). For simplicity we allow drainage to begin immediately at time $t = 0$, with the pond already filled to water level H_0.

The model for this scenario is given by equations (6.1), (6.5) and (6.10). Using (6.10) we can write (6.1) as

$$\dot{H} = c_c - \frac{\hat{K}}{a}(H - \rho_{sw} b) \tag{6.14}$$



(a) (b) (c)

Figure 6.2: Results of a numerical simulation of equations (6.1), (6.5) and (6.10) for a cylindrical pond, together with analytical solutions. (a) Schematic of the pond, with vertical walls and fixed area. From time $t = 0$ the pond drains through the porous region between the dotted lines. (b) Pond water level H and draft b of the floe, compared to analytical solutions, equations (6.17) and (6.18). The cylinder has constant area $a = 0.5$, initial pond centre height $h_c = 0.8$, in a floe of area $a_0 = 1$ and peak height $z_p = 1$, with initial water level $H(0) = 0.9$, and a $h_{crit} = 0.8$ so that the pond drains immediately, with $\hat{K} = 100$. Melt rates $v_w = 2$ and $v_b = 1$ are used. (c) Plots of water level according to the analytical solution, equation (6.17), varying $\hat{K}$.

where for brevity we have defined $c_c = \rho_{iw} v_b a_0/a - [v_w - \rho_{iw}(v_w - v_b)]$. If we differentiate (6.14), then use (6.5), (6.10) and (6.14) to eliminate $\dot{b}$ in favour of $\dot{H}$, we can write this as

$$\ddot{H} + c_1 \dot{H} = -\frac{\hat{K} c_c}{a_0}, \tag{6.15}$$

where

$$c_1 = \frac{\hat{K}(a - a_0)}{a_0 a}. \tag{6.16}$$

Equation 6.15 has solution

$$H(t) = \frac{k_1}{c_1}(e^{-c_1 t} - 1) + \left(\frac{c_c a}{a - a_0}\right) t + H_0, \tag{6.17}$$

where k_1 is a constant of integration. If we differentiate (6.17) and make use of (6.14) we derive the expression for the draft,

$$\rho_{sw} b(t) = \left(\frac{1}{c_1} - \frac{a}{\hat{K}}\right) k_1 e^{-c_1 t} + \left(\frac{c_c a}{a - a_0}\right) t + \frac{a c_c}{\hat{K}} \left(\frac{a}{a - a_0} - 1\right) + H_0 - \frac{k_1}{c_1}. \tag{6.18}$$

The constant k_1 is determined from the condition $b(0) = b_0$ to be

$$k_1 = c_c \left(\frac{a}{a - a_0} - 1\right) + \frac{\hat{K}}{a}(H_0 - \rho_{sw} b_0). \tag{6.19}$$

Results from a numerical simulation of the filling and drainage behaviour of this cylinder-shaped pond are shown in Figure 6.2(b), compared to the behaviour of equations (6.17) and (6.18). These agree to within the tolerance of the numerical scheme used. Panel (c) illustrates the dependence on the drainage parameter. For $\hat{K} = 10$, drainage occurs too slowly for the water level to equilibrate to sea level. In comparison, this is shown to occur quickly for $\hat{K} = 10^3$, with the relaxation being so fast as to be almost instantaneous compared to the time scale for melting through the floe. The difference between $\hat{K}$ of 10^3 and 10^4 is only noticeable during a short period as drainage of the catchment ends. We therefore conclude that a dimensionless drainage parameter of $\mathcal{O}(10^3)$ captures the observed behaviour of ponds draining over a period of days, much smaller than the length of the melt season of around 90 days.

6.1.3 How much do changes to the topography matter?

Our algorithm calculates pond area correctly when the water levels are increasing, as the slope and curvature of the bare ice topography does not change over time. However, this ceases to be true once the water level and pond area recedes, as the topography will have changed as ponds grow and experience different melt rates over time. The network model does not record how the submerged topography changes as ponds grow, and therefore we cannot accurately determine area as a function of water depth once drainage begins and the water level decreases. It is simple and computationally efficient to use the gradient of the initial topography when evaluating changes in the pond boundaries, or relating pond areas to water levels, as we have previously in numerical simulations. For this reason we wish to continue to do so in numerical simulations of the model with drainage. However, we expect that this method is inaccurate.

In order to check how much ignoring changes to the topography affects a simulation, we compare two simulations: one that uses the method that records changes to the pond floor, which we call the 'discretized' simulation, and another that does not, which we refer to as the 'unmodified' simulation. In a discretized simulation, the water level evolves according to (6.1). We consider a half pond in one dimension as in Chapter 3. The right boundary moves according to

$$\dot{\beta} = \frac{\dot{H} + v_b}{h_x(\beta^+, t)} \tag{6.20}$$

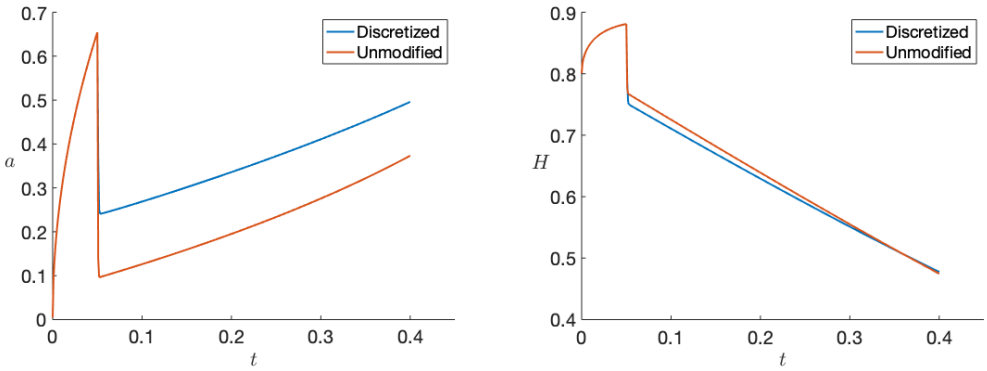
if the water level is increasing relative to the ice, $\dot{H} \geq -v_b$. Before the pond begins to recede, the gradient $h_x(\beta^+, t)$ is equal to h_{0x}, the initial gradient. The superscript

of β^+ has the same meaning as in Chapter 3, in this case referring to the dry side of the boundary. The unmodified simulation continues to use this even when $\dot{H} < -v_b$. However, in reality if $-v_w < \dot{H} < -v_b$ then the pond boundary remains fixed, and if $\dot{H} < -v_w$, we need

$$\dot{\beta} = \frac{\dot{H} + v_w}{h_x(\beta^-, t)}, \tag{6.21}$$

as the gradient of the topography needs to be evaluated on the inside of the boundary, because the boundary is moving in that direction.

To determine the topography $h(\mathbf{x}, t)$, we discretize the full topography and evolve the heights at each point according to equation (6.3). The slope h_x of each interval element of the discretization of the boundary is then known at all times, not just initially, and can then be used in (6.20) and (6.21) to evolve the pond boundary. This also allows us to visualise how the shape of the floor of a pond changes over time, which is discussed in Appendix B.



(a) Areas (b) Water levels

Figure 6.3: Time series of pond areas (a) and water levels (b) for numerical simulations of filling and drainage of a catchment of half-width 1 in one dimension, shaped as a notch, with $\bar{h} = 1$, $\Delta h = 0.2$. Parameters $\hat{K} = 1000$, $h_{crit} = 0.7$, and $v_w = 2$ were used. Two simulations were carried out. In the 'discretized' simulation, the topography was discretized and modified through time. In the 'unmodified' simulation, the initial topography was used throughout, as described in the text.

Results from these two numerical simulations are shown in Figure 6.3. Panel (a) shows that even though the unmodified simulation underestimates pond area after drainage, it does so by a nearly constant amount through time. Panel (b) shows the water levels in each simulation, which are almost the same throughout. Similar differences to those seen in Figure 6.3 are seen between the behaviour of the two cases as we vary Δh or h_{crit}. From the consistent difference in area coverage after

drainage seen between the two methods, we can conclude that leaving the slope of the topography unchanged should have little effect on the interactions between ponds, and therefore little effect on the behaviour of the network.

In Chapters 3, 4 and 5 we demonstrated that simulations in one and two dimensions display qualitatively similar behaviour. In particular, the equations for the motion of the boundary of a one dimensional pond (3.18) and a two-dimensional pond (3.45) have the same dependence on the gradient of the ice topography. We therefore conclude that it is an acceptable modelling decision and simplification to ignore changes in the topography throughout the simulation when calculating pond area, taking $h_x(x,t) = h_{0x}$ always in (6.20), and we will do this in all numerical simulations for the remainder of this thesis. We do however have to acknowledge that this leads to an underestimation in pond area at later times.

6.2 Incorporating drainage in a simulation of pond systems

6.2.1 Modifying the model equations for a whole floe

In a similar way to incorporating drainage over the sides of a floe, to incorporate drainage through the base of the floe into a model for a pond network, we add edges connecting each catchment in the floe to the extra 'sea' node at the end of the edge list, the $(n+1)^{th}$ node, denoted by a subscript s. This adds an extra n rows to the edge list. Each of these rows has the catchment index in the first column and the sea node in the second. Incorporating the possibility of a drainage flux q_{ij} on these additional edges, the water levels are still given (as in Chapter 5), by

$$\dot{H}_i = \frac{\rho_{iw} v_b a_{0i} - \sum_j q_{ij}}{a_i} - [v_w - \rho_{iw}(v_w - v_b)], \tag{6.22}$$

for $i = 1, \ldots, n$ and where the sum over j includes the sea node s. We tie the sea node to sea level, such that

$$\dot{H}_s = \dot{b}, \tag{6.23}$$

and $H_s(0) = b(0)$, given that we measure heights from the base of the floe. We also give the sea node a constant area of 1 and 'catchment area' of 1, so as not to interfere with terms that depend on these quantities in equations in matrix form. The draft changes according to

$$\dot{b} = -\frac{\sum_i q_{is}}{\rho_{sw} \sum a_0}. \tag{6.24}$$

Following the discussion in the previous section, areas are calculated as described in Chapter 5,

$$a_i(t) = f_i(\tilde{H}_i(t)) \tag{6.25}$$

where f_i is the hypsometry of the initial topography, and

$$\tilde{H} = H_i + \int_0^t v_b \, \mathrm{d}t. \tag{6.26}$$

We partition our edge list into the original e edges between ponds ('surface edges') and $e_d = n$ edges between ponds and the sea. The indicators are now of length $e + e_d$. The last e_d entries of $\mathbf{i}_2$, $\mathbf{i}_3$ and $\mathbf{i}_4$ are restricted to have values of 0 at all times. The indicator $\mathbf{i}_1$ represents inactive edges across the whole edge list. We leave the draining edges, the last e_d edges, inactive until $h_c < h_{crit}$. As the draining edges behave in a different way to edges between ponds when active, we define another indicator $\mathbf{i}_5$ of length $e + e_d$, which has value zero always for the first e edges, and either 1 for any of the others while the corresponding $h_c <= h_{crit}$, or 0 otherwise. With this indicator we build the diagonal matrix $\mathbf{I}_5$ which we use to incorporate equation (6.12) for drainage fluxes into the system of equations for $\mathbf{q}$. The expanded system of equations for $\mathbf{q}$ now has $e + e_d$ equations for that many q's. The conditions to determine the states of surface edges, and their associated fluxes are the same as described in Chapter 4.

The system for the whole floe, including drainage can be written as

$$\begin{aligned}\left[\mathbf{I}_1 + \mathbf{I}_2\mathbf{L}\mathbf{A}^{-1}(\mathbf{R}-\mathbf{L})^T + \mathbf{I}_3\mathbf{R}\mathbf{A}^{-1}(\mathbf{R}-\mathbf{L})^T + \mathbf{I}_4(\mathbf{L}-\mathbf{R})\mathbf{A}^{-1}(\mathbf{R}-\mathbf{L})^T + \mathbf{I}_5\right]\mathbf{q} = \\ \mathbf{I}_2((1-\rho_{iw})(v_w - v_b)\mathbf{1} - \rho_{iw}v_b\mathbf{L}\mathbf{A}^{-1}\mathbf{a}_0) + \mathbf{I}_3((1-\rho_{iw})(v_w - v_b)\mathbf{1} - \rho_{iw}v_b\mathbf{R}\mathbf{A}^{-1}\mathbf{a}_0) \\ - \mathbf{I}_4(\rho_{iw}v_b(\mathbf{L}-\mathbf{R})\mathbf{A}^{-1}\mathbf{a}_0) + \mathbf{I}_5\mathbf{L}K(\mathbf{H} - \rho_{sw}b\mathbf{1}).\end{aligned} \tag{6.27}$$

The change in draft is dependent on the sum of the water fluxes out of the floe, from (6.24),

$$\dot{b} = -\mathrm{tr}(\mathbf{I}_5\mathbf{q})\rho_{sw}^{-1}[\mathrm{tr}(\mathbf{A}_0)]^{-1} \tag{6.28}$$

where the matrix $\mathbf{A}_0$ is the $(n+1) \times (n+1)$ matrix with catchment areas along the diagonal. The model now consists of equations (6.22) and (6.25) for the water levels and pond area, fluxes given by equation (6.27), and draft given by (6.28). We have decided that in the model the area function does not track changes to the ice topography due to preferential melting in ponds. The melting behaviour at saddles

must be amended so as to be consistent with this choice. Therefore, at all times, all saddle heights h_{ij} move downwards according to

$$\dot{h}_{ij} = -v_b \tag{6.29}$$

even when ponds are joined. The melting behaviour of the pond centres is described in Section 6.2.4.

Aside from the densities, which are fixed, the non-dimensional model now has four parameters: the relative melt rate v_w, the drainage parameter $\hat{K}$, the critical depth h_{crit} and the topographic amplitude Δh. We will explore changes to these parameters in the sensitivity analysis, Section 6.3.2. We will later consider two of them, Δh and h_{crit} to be variables for a more in-depth analysis in Chapter 7.

6.2.2 Breaking loops before and after drainage

Our algorithm evolves the water level, flux and hypsometry equations, pausing between events (where an edge changes state). Upon a change of state, the state vectors (the **i**'s) are updated according to the conditions for surface edges described in Chapter 4 and the condition for catchment-to-sea edges described in the previous section. We break loops in the graph of e surface edges in the same way as before, discussed in Appendix A (but allow loops in the full graph, as fluxes from ponds to the sea are determined by their own equation, rather than implicitly and do not cause any indeterminacy in solving the system of flux equations). To break loops, we identify the changed edges by comparing the new state vectors (those at the current event) to the old state vectors (those from the previous event), and identify their type. We add these one by one (in the case where there is more than one edge change in an event, which is possible but unlikely) to the graph built from the previous state. We then check if this new graph has any loops. If not, we keep the new network. If so, we erase the change to the new indicator vectors and keep the old graph. This is repeated for all edge changes.

While this seems at first glance to be computationally inefficient, when running simulations of the model, especially when using an integrator for the model equations that utilizes an adaptive time-step (which could make the time-step very small), it is more efficient than attempting to identify and possibly change the state of the network at each step of the integrator. MATLAB, which we use to perform simulations of the model, uses an events function that can pause the integrator on meeting a condition. Once paused, it is then quick and easy to change the state of the network and restart the integrator.

6.2.3 Overflow formation during drainage

After the onset of drainage, overflows can form 'in reverse' if the ponds at both sides of the overflow begin to drain. We consider the scenario shown in Figure 6.4(a), where pond j has begun to drain with flux q_{js} and an overflow has formed from i to j with flux q_{ij}. If pond i now also begins to drain, with flux q_{is}, as shown in panel (b), this can cause the overflow flux q_{ij} to reverse in direction, along with the drainage flux q_{js}. This is due to the constraint $H_i = h_{ij}$ (that the water level in i remains fixed at the saddle height), and the fact that $q_{is} \gg q_{js}$ as $H_i - \rho_{sw} \gg H_j - \rho_s w$, and q_{is} dominates the other terms when solving for the fluxes. This is not allowed, so once drainage begins we monitor for uphill overflow fluxes, (negative for a left-to-right overflow and positive for a right-to-left overflow), and switch the overflow edge to an inactive edge if this does occur.

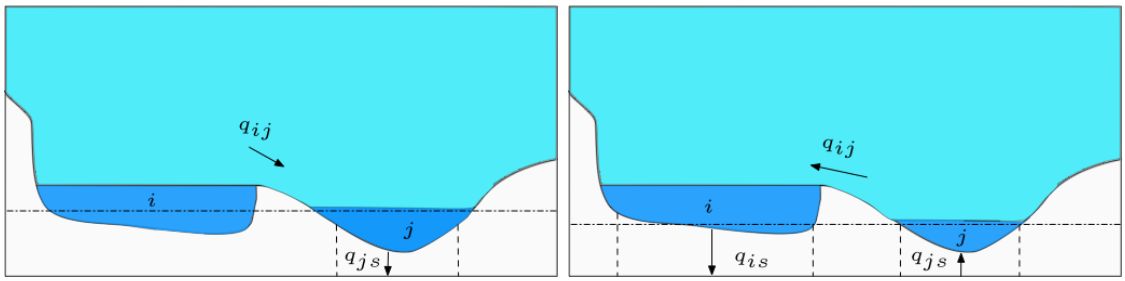


Figure 6.4: Schematic of the reversal of an overflow flux and drainage flux due to 'pinning' of the water level at the saddle height for an overflowing pond. Porous regions are shown between dashed vertical lines. Sea level is marked by the dash-dot line. Both panels are in a reference frame where the ice surface is stationary (a) Pond j begins to drain with flux q_{js} and pond i begins to overflow into j with flux q_{ij}. (b) Pond i now also begins to drain, with flux q_{is}. The larger drainage flux q_{is} and the constraint $H_i = h_{ij}$ cause a reversal of the directions of q_{ij} and q_{js} when solving for the fluxes. This needs to be controlled for in numerical simulations.

6.2.4 Accounting for total drainage of ponds

In the regime where the critical thickness for drainage is large, or the amplitude of ice surface topography is small, drainage occurs quickly and can fully drain a catchment. This catchment may not recover any pond area again, except perhaps at the end of a summer period, as the change in draft will keep the pond centres high in these 'dead' ponds. Due to the singularity in equation (6.22) for water level as $a_i \to 0$, the numerical scheme struggles when ponds completely drain. For the beginning of

the simulation with an initial condition of $a(0) = 0$ we dealt with the singularity by making use of an asymptotic expansion of (6.22) for small times (see Chapter 5). However this is not so straightforward later in a simulation, as the equations for each catchment need to be solved together as a linear system. Instead, we assume that once the area in a catchment becomes 'small enough' which we denote by a threshold value ϵa_0, and the water level is above sea level, the catchment is considered fully drained. For such catchments, we state that the water level changes at the rate of bare ice,

$$\dot{H}_i = -v_b. \tag{6.30}$$

We are now in a better position to define how the heights of pond centres evolve in the network model with drainage. These now move according to

$$\dot{h}_{ci} = \begin{cases} -v_b & a_i \leq \epsilon a_0 \\ -v_w & a_i > \epsilon a_0. \end{cases} \tag{6.31}$$

We take $\epsilon a_0 = 10^{-4}$, and allow all fluxes into the catchment, and any melt water formed in the catchment to drain, such that

$$q_{is} = \rho_{iw} v_b a_{0i} - \sum_j q_{ij}. \tag{6.32}$$

With this possibility included in the model, the porous drainage equation becomes

$$q_{is} = \begin{cases} 0, & h_{ci} > h_{crit} \\ K(H_i - \frac{\rho_s}{\rho_w} b), & h_{ci} \leq h_{crit} \ \& \ a_i > \epsilon a_0 \\ \rho_{iw} v_b a_{0i} - \sum_j q_{ij}, & h_{ci} \leq h_{crit} \ \& \ a_i \leq \epsilon a_0. \end{cases} \tag{6.33}$$

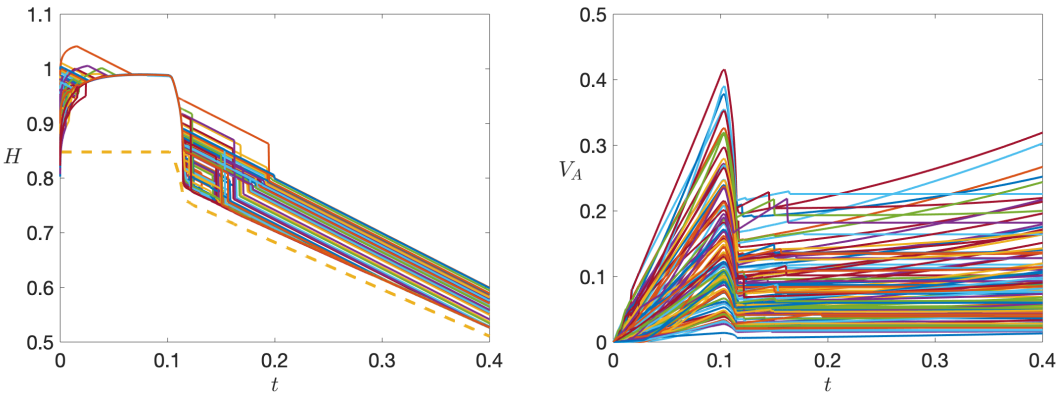
6.2.5 Ponds breaching the base

Given that the model now assumes finite ice thickness, we must account for the possibility of a pond melting all of the way through the floe. Once this happens, which we call 'breaching', no further ice exists to melt at the points on the floe where a pond breaches the base. This has consequences for the mass balance within each catchment, and would require a more rigorous treatment of modifications to the topography of the pond floor than what we have decided to use, given that we only record the locations of the pond centres, ice peaks and saddles. For a floe of average thickness 1 m, and assuming a melt rate of 2 cm/day in the pond, we only expect a breach after approximately fifty days. This is long enough to run our simulation such that we can examine the state of a floe in late summer. Therefore, in order to avoid accounting for breaches, we decide to end our simulations once the first pond floor breaches the base of the floe.

6.3 Analysis of the model with drainage

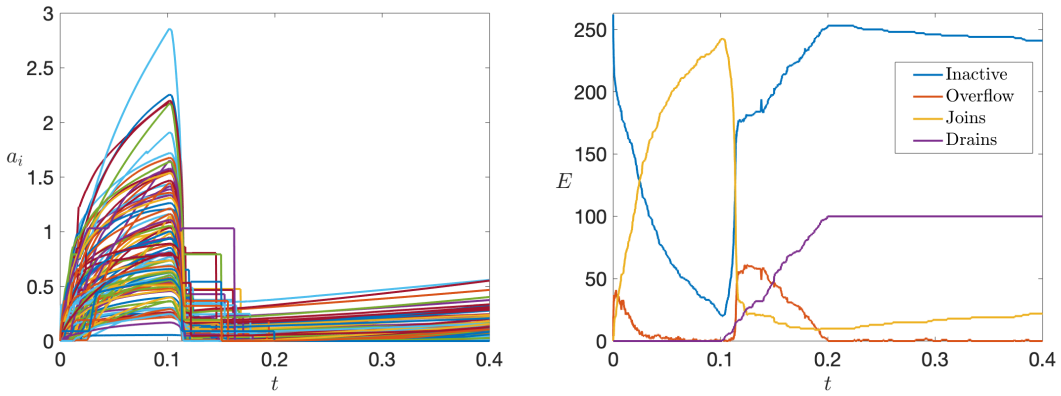
6.3.1 A single example

We now discuss output from an example simulation of 100 nodes on a Voronoi tessellation, which is summarised in Figures 6.5, 6.6, 6.7 and 6.8. Parameters $\hat{K} = 1000$, $v_w = 2$, $\Delta h = 0.2$, and $h_{crit} = 0.6$ are used. Here we take one particular example in order to explore the behaviour and identify the key physical phenomena seen in the model, and examine the sensitivity to key parameters. A detailed analysis of the model, including percolation in systems of ponds and the fractal dimensions of pond geometries is given in the next chapter.



(a) Water levels in each catchment

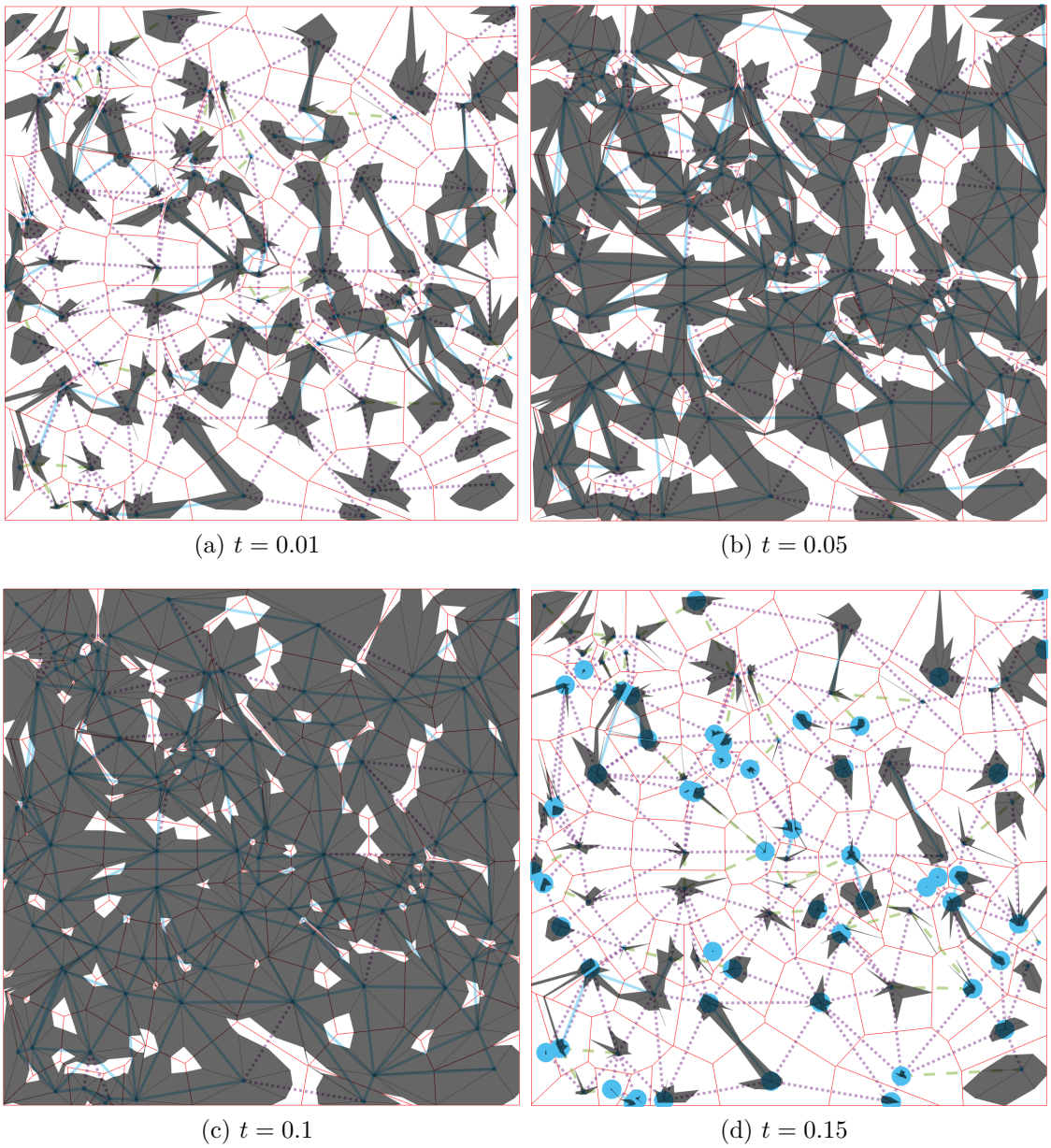
(b) Pond volume in each catchment



(c) Pond area in each catchment

(d) Numbers of edges in each state

Figure 6.5: (a) Water levels (solid) and sea level (dashed) for simulations of 100 catchments in a Voronoi tessellation. (b) Accumulated volumes of water. (c) Pond area for each catchment. (d) Time series of edges in each state. Parameters are $\hat{K} = 1000$, $v_w = 2$, $\Delta h = 0.2$, and $h_{crit} = 0.6$.



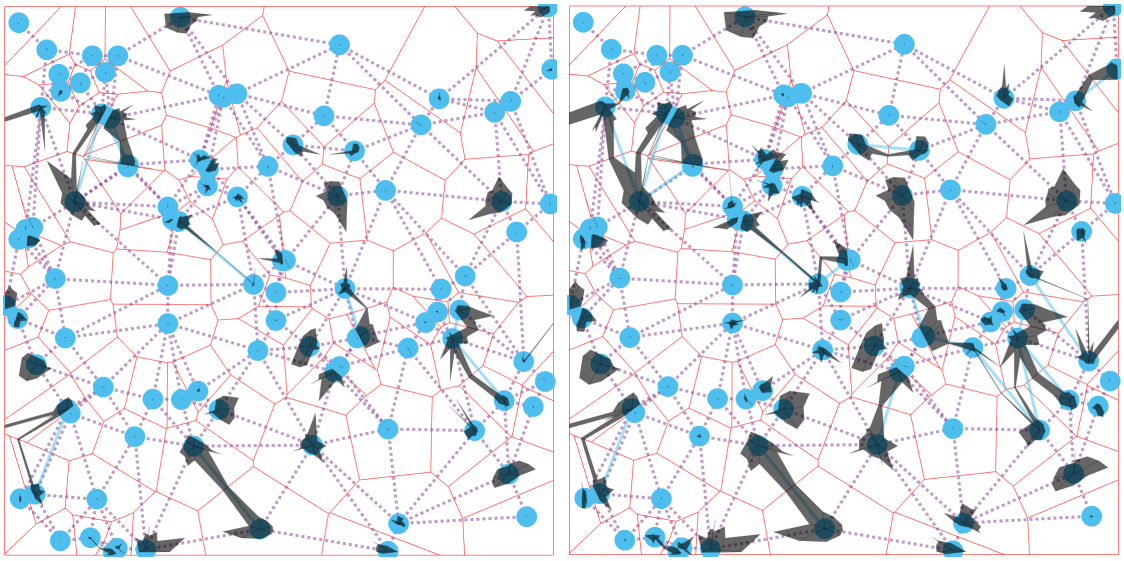
(a) $t = 0.01$ (b) $t = 0.05$

(c) $t = 0.1$ (d) $t = 0.15$

Figure 6.6: Visualisation of floe at time slices throughout development for the simulation shown in Figure 6.5, up until time $t = 0.15$, midway through drainage. Drainage begins around $t = 0.1$. Blue circles denote draining catchments. Dotted lines denote potential edges, solid blue denote joins, and dashed yellow lines denote overflows. Grey areas denote ponded regions. Red lines denote the boundaries between catchments.

Water levels and melt volumes

Water levels for catchments relative to the base of the floe are shown in panel (a) of Figure 6.5. Panels (b) and (c) show the volume of water accumulated and pond area in



(a) $t = 0.2$ (b) $t = 0.4$

Figure 6.7: Visualisation of floe at time slices throughout development for the simulation shown in Figure 6.5, from the end of drainage to the end of the simulation. Blue circles denote draining catchments. Dotted lines denote potential edges, solid blue denote joins, and dashed yellow lines denote overflows. Grey areas denote ponded regions. Red lines denote the boundaries between catchments.

each catchment, respectively. Note in (a), the draft b (also relative to the base of the ice) is shown by the dashed line and is initially flat during the impermeable phase up to $t \approx 0.1$. After this impermeable phase where the water levels initially rise, ponds begin to overflow into one another then merge until all ponds follow a single water level trajectory (as discussed in Chapter 5), we see the behaviour presented earlier for drainage of a single catchment, but for the entire floe. The water level falls, initially smoothly, then rapidly, and eventually moves downwards with the melting ice. We also notice the typical behaviour of overflows during the draining period (water levels falling exactly at a rate $-v_b$ for a period of time). In the case where the pond network becomes densely connected again, we expect the water levels to all equalise to a height of $\rho_{sw} b(t)$. However, in this simulation most catchments drain completely, and the water levels of these catchments get fixed at the height of their catchment centres. Only the active ponds with the lowest water levels (and centres) tend towards the height of $\rho_{sw} b(t)$. Once drainage begins, the draft decreases quickly (and the waterline moves towards the base of the floe), as ponds begin with water levels above sea level and the floe loses weight of water that drains from ponds. Once the initial drainage stage ends, the draft continues to decrease due to continued melting, the

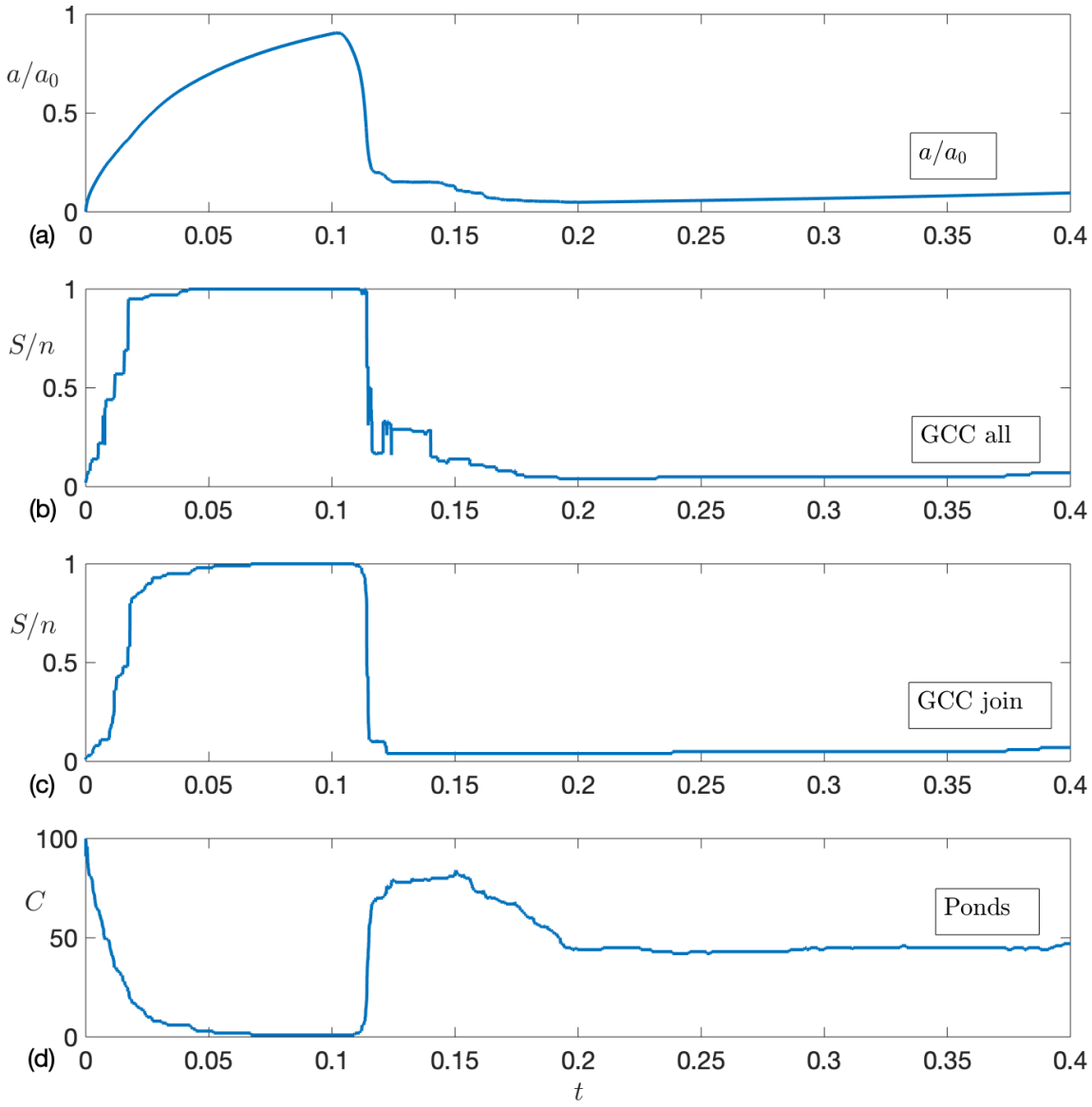


Figure 6.8: Time series results for for the simulation shown in Figure 6.5. Panel (a) total area fraction, (b) fractional size of giant component for the graph of joins and overflows (considering both join and overflow edges as active) and (c) fractional size of giant component for the graph of just joins (considering only join edges as active). (d) The number of individual ponds on the floe at each time-step (number of components in the graph of join edges minus number of dead catchments).

water from which drains into the sea.

Pond volumes, shown in (b), change through time in much the same way as pond coverage does. The panel shows the typical fast increase pre-drainage and a precipitous drop at drainage across all catchments. It also shows some catchments

filling post-drainage due to collection of meltwater from the surrounding bare ice, while also being disconnected from regions that are draining. Such continued filling ends by $t \approx 0.2$, the time at which all ice becomes porous, when the highest pond centre reaches the critical thickness. Thereafter there is a steady increase in pond volume for catchments whose floors lie below sea level and whose water levels move with the changing draft of the floe. The panel also shows many catchments plateauing with little to no volume post drainage. These have become empty (though the finite area given to them by the model has a corresponding finite accumulated volume).

Edge types

The behaviour of individual ponds, catchments and saddles during the drainage process is seen clearly when we examine the time series of active and inactive edges, Figure 6.5(d). Edges in the impermeable period ($t = 0$ to $t = 0.1$) change as we have seen already in Chapter 5. At the point of the first drain we also see a rapid increase in the number of independent edges, and a decline in join edges as ponds drain and saddles re-emerge. We also see a sharp increase in overflow edges, as water from components not joined to draining regions overflows into the draining regions. This resembles the cascade effect seen as a consequence of boundary drainage (Section 5.6), and explains why the peak of overflowing edges post-drainage is larger than the initial peak due to filling. We note that the process of filling and drainage are not direct opposites of one another, and the draining state is capable of sustaining more overflow edges than filling. During drainage, the water levels cannot rise to cover overflows, so overflows can persist for longer. Overflows eventually decay with the appearance of the remaining drains, and previously overflowing edges become inactive.

Eventually the system settles down once the whole floe is sufficiently porous. At this point the number of overflows decays to zero, the number of independent edges peaks before slowly decaying and the number of join edges starts to increase again as ponds start to refill due to the relative increase of sea level. Note also that we see tiny increases in the number of overflow edges during refill that quickly disappear again as they become join edges.

Pond areas and floe visualisation

It is useful to examine the evolution of pond cover in individual catchments, shown in Figure 6.5(c). We see large differences between catchments, partly because of the differences in catchment size due to the random nature of the topography, but also due to levels of connectivity, and the times at which different catchments drain.

This is not a synchronous process. Due to the random distribution of initial pond centre heights, each will become porous at a different time. From early time, the same behaviour of filling during the impermeable stage is seen, as was described in Chapter 5. We then see precipitous drops in all pond areas simultaneously around $t \approx 0.1$ once drainage begins in the first catchment. All catchments lose area quickly, some completely, while others eventually plateau, due to the formation of overflowing edges connecting regions that are draining to regions not yet doing so. These plateauing regions see precipitous falls in area cover again once they too become porous. Once all regions of the floe have become porous, pond cover increases slowly and smoothly due to refilling of ponds as sea level increases up the floe topography with continual melting of the ice.

Figures 6.6 and 6.7 show visualisations of the floe at set times. Grey areas show pond cover. Red lines mark the boundaries of catchments. Blue circles denote draining catchments. Dotted lines denote potential edges, blue solid denote joins, and dashed yellow lines denote overflows. Figure 6.6(a) shows the system early in pond development, as ponds overflow into one another and begin to join. In (b), the system of ponds is well developed and is moving towards a state near flooding shown in panel (c), which is immediately prior to drainage. Panel (d) shows the system midway through drainage, when overflows have reformed and pond coverage has shrunk considerably. In 6.7(a) we see the system as drainage ends, and pond coverage is at a minimum. There are a small number of compound ponds, no overflows, many catchments are essentially completely empty. Finally, in panel (b), the remaining ponds have grown somewhat and some have reappeared, which we call 'refilling'.

Pond system measures

Figure 6.8(a) shows how the total area fraction of ponds varies through time. Panel (b) shows the fractional size of the largest pond system (the largest component of the graph, the GCC), where we denote cluster size as S, and count all active edges. Panel (c) shows the equivalent fractional size of the largest pond, when only join edges are counted. Finally, panel (d) shows the total number of individual ponds (components, C), in the graph of join edges, minus any 'dead' ponds. Presenting the system in this way shows how the pond network changes throughout the development of ponding during the summer.

As ponds initially form, the system rapidly becomes densely connected, with percolation occurring early (around $a/a_0 \approx 0.33$ as discussed in Chapter 5, Section 5.6). We note that the percolation process looks slightly different depending on whether

the system is defined as having overflows or not. These connections do not cause the visible appearance of ponds, but are still important to the function of pond systems, a point we will explore in more detail in Section 7.6.

It is clear that by $t \approx 0.05$ the whole system is connected and water flows freely from any one region to another on the floe, as can also be seen in Figure 6.6(b). Just after $t = 0.1$, this changes rapidly as the first catchment becomes porous. Because every catchment is connected to the porous one, all drain until saddles begin to re-emerge at $t \approx 0.11$ and some clusters of catchments become disconnected from those that are draining. This happens within each connected region in turn as further catchments become porous, and the size of the largest component drops quickly (seen in the second and third panels) with a corresponding increase in the number of individual ponds. The densely connected nature of the pond system in the early period of melt aids in the quick shedding of water from the surface once drainage begins. This is in agreement with analysis on drainage of pond systems by Popović et al. [2020a], who considered a statistical model of pond drainage in an initially connected system by the introduction of randomly seeded holes.

Altogether, this simulation tells a story of initial pond filling on impermeable ice and flooding of the floe, a rapid drainage phase as pond area retreats, and then a slow refilling of mostly porous ice with ponds just above sea level. We find that the process of drainage is somewhat noisy, due to the nature of the path to drainage we have chosen, where not all regions of the floe become porous at the same time. As a consequence of this choice, in the model, drainage takes longer for a system of ponds than for a single, isolated pond, as was explored in Section 6.1.2. This may be physically realistic, as the phenomenon of ponds on the same floe draining at different times and rates was mentioned by Polashenski et al. [2012].

6.3.2 Sensitivity analysis

Figure 6.9(a) shows the dependence of the evolution of area fraction on the drainage parameter, $\hat{K}$, with a range of values corresponding to permeabilities of 10^{-10} to 10^{-12} m^2, in line with data recorded by Polashenski et al. [2017]. A drainage parameter $\hat{K} = 10$ is seen to lead to insufficient drainage, and the whole floe floods at late time. A parameter value of $\hat{K} = 100$, corresponding to the lower end of what was recorded by Polashenski et al. [2017] in the later part of the field experiment as the permeability of the ice increased, leads to smooth drainage as the time scale for drainage in this case is slower than the rate at which the ice become porous. A parameter value of $\hat{K} = 1000$ leads to a staggered decrease in area cover, as drainage

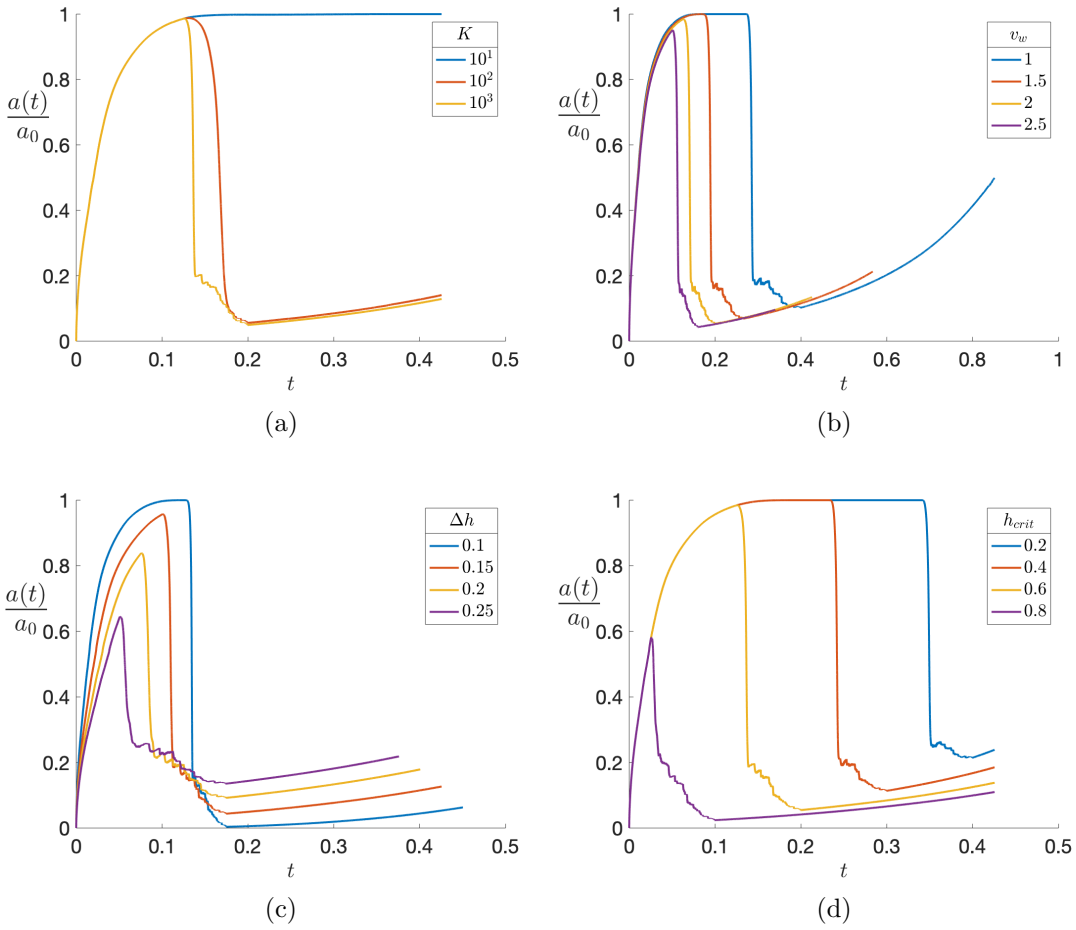


Figure 6.9: Sensitivity analysis of total area fraction for simulations of 100 catchments of a square lattice topography. Benchmark parameters are $v_w = 2$, $\hat{K} = 1000$, $\Delta h = 0.15$ and $h_{crit} = 0.6$. (a) Varying $\hat{K}$. (b) Varying v_w. (c) Varying Δh. (d) Varying h_{crit}.

happens at a rate much faster than that at which isolated regions become porous, as mentioned in Section 6.2.3. In this case, the area coverage is controlled more by the pond interactions than by drainage itself. This behaviour was also found to hold across other topographies and critical thicknesses.

Figure 6.9 (b) shows the dependence of the model on the rate of enhanced melting v_w. The qualitative pattern of the behaviour to enhanced melting is similar across a range of enhanced melt rates, with the pond base reaching the onset of drainage later and breaching of the base of the pond occurring later for a lower melt parameter. Lowering the melt ratio led to a higher propensity for flooding of the floe. With the exception of the case where the melt rates are equal, area coverage traces the same path during refilling, suggesting that changes in melt ratio should not affect

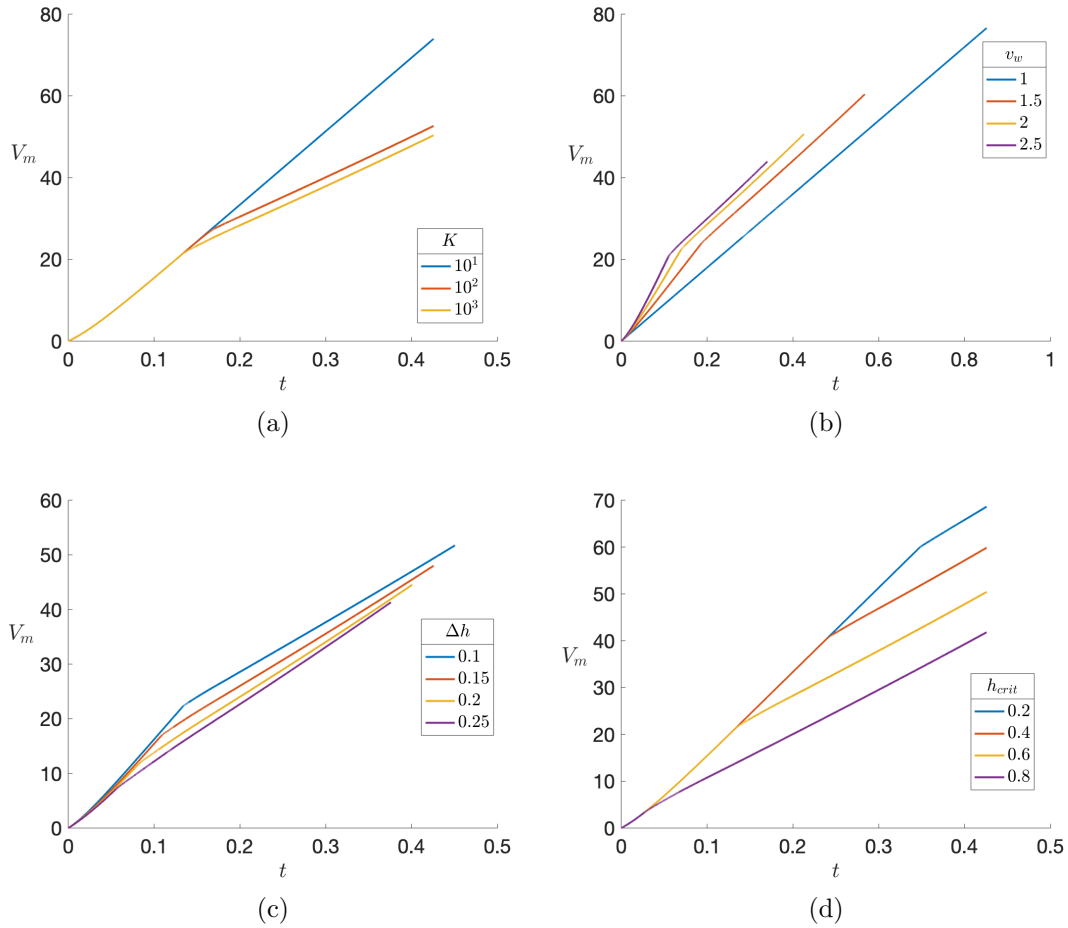


Figure 6.10: Sensitivity analysis of total melt volume (water) for simulations of 100 catchments in a square lattice. Benchmark parameters are $v_w = 2$, $\hat{K} = 1000$, $\Delta h = 0.15$ and $h_{crit} = 0.6$. (a) Varying $\hat{K}$. (b) Varying v_w. (c) Varying Δh. (d) Varying h_{crit}.

the behaviour during this period, except to make simulations with higher ratios end earlier. For experiments with higher melt ratios, pond floors breach the base of the floe faster. Volume melted (measured in water volume liberated), shown in Figure 6.10(b), varies considerably with changes in the melt parameter, as would be expected intuitively, despite the differences seen due to experiments finishing at different times.

Changes due to surface roughness can be seen in panel (c) of Figure 6.9. This has a strong influence on how pond coverage changes through the melt season on a floe, with different sensitivity at early and later times. Smoother ice tends to have larger pond area in the early part of the melt season, but is less likely to maintain pond area later into the melt season, with many ponds disappearing completely on smooth ice. The opposite is true for rough ice. This indicates that Δh is one of the stronger

controls on pond area fraction. Smaller differences are seen in the associated melt volume in Figure 6.10(c). The dependence of the model on the roughness Δh will be explored in more detail in the next chapter. The disappearance of ponds on smooth ice is investigated in more detail in Chapter 7, Section 7.3.

Figure 6.9(d) and 6.10(d) show the dependence on the critical thickness h_{crit}. This parameter affects the timing of stages of pond evolution. Lower critical thicknesses cause drainage to occur later, resulting in larger amounts of enhanced melting, seen in Figure 6.10(d). Changes in the time of the onset of drainage can impact whether the floe partially or completely floods before drainage, and is therefore a strong control on the maximum pond area. This could have a significant impact on the seasonally integrated albedo, especially in models that attempt to resolve ice and ponds over the course of several years, and need to account for several annual cycles of energy fluxes (such as CICE).

It is geophysically interesting to re-dimensionalize the time scale in Figure 6.9(b) and (d) and examine how long after the onset of ponding drainage begins, as this has a strong control over the time-averaged albedo. If we choose $\tilde{h} \sim 1$ m and $\tilde{v} \sim 1$ cm/day, then each full unit of time in the simulations shown correspond to 100 days. In Figure 6.9(b), if the melt ratio is as high as 2.5, drainage begins around 10 days after the onset of drainage. If the melt rates are equal, drainage will only begin a month after ponds begin to form. In Figure 6.9(d), ice that drains early, with $h_{crit} = 0.8$ will drain several days after the appearance of ponding. In the less plausible scenario that ponds do not drain until they have almost melted the full way through the floe, $h_{crit} = 0.2$, floes would remain flooded until 35 days after ponds begin to form. These differences are large and consequential, and may better inform us as to what are sensible parameter values for our model. The time scales are investigated in more detail in Chapter 7, Section 7.2.2.

6.4 Pairing methods of drainage

We have considered two of the main methods of drainage separately in this chapter and the previous one. Drainage over the floe side was shown to limit pond coverage to near the percolation threshold throughout time. Porous drainage was shown to have more complicated effects on the evolution of area coverage, with effects that depend on the depth and time of the onset of drainage, and on the surface roughness of the ice. For completeness we pair both types of drainage in a simulation, and provide a qualitative analysis of the results here. There are several difficulties in side drainage

that were not addressed in Chapter 5, Section 5.6. That section did not account for sea level, or for the potential of seawater flooding into the ponds (a join edge forming with the sea). Also that experiment was designed so that loops would not form in the graph comprised of catchments and the sea node. These issues become important once we incorporate buoyancy and account for the role of sea level, as join edges between ponds and the sea could occur, and these could cause a loop that would lead to indeterminacy in the system of equations used to determine the fluxes. Further, multi-edges (repeated edges between the same two nodes) with the sea form at corners, which we must disallow.

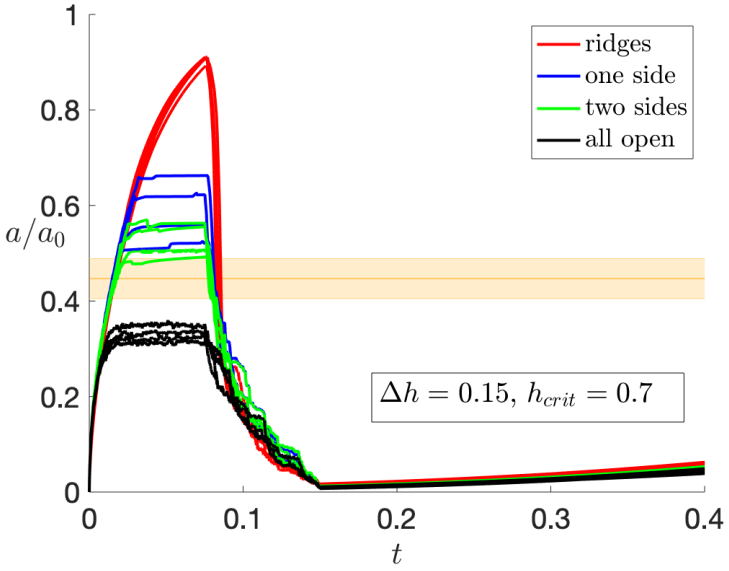


Figure 6.11: Experiments that incorporate both side drainage and porous drainage. Four experiments are shown, each using the same four random topographies, square lattices of 64 nodes with $\Delta h = 0.15$. Parameters $v_w = 2$, , $h_{crit} = 0.7$ and $\hat{K} = 1000$ were used in each. The 'ridge' experiment has no side drainage, only porous drainage. The 'one side' experiment allows the lowest saddle on one side of the floe to drain. The 'two side' experiment allows drainage over the lowest saddle on each of two opposite sides. The 'all open' experiment allows drainage over all saddles on all sides. Shown also in orange is the average area at which percolation of the pond system during filling occurs, calculated as an average across the four topographies used, as described in Section 5.6 (note that this is higher than shown in that section, due to changes in network structure). The shaded region is one standard deviation.

To avoid these intricacies we decide to choose only certain boundary edges at which to allow overflow into the sea. Once the ice in the associated catchment becomes porous however, we disallow side drainage, so that each catchment can only

drain by any one mechanism at a time. Water levels relax to near sea level after drainage begins, removing the need to account for lateral flow over the floe's edge. For algorithmic simplicity, the flux is calculated using the same equation for porous drainage, but with a different condition:

$$q_{is} = \begin{cases} 0, & H_i \leq h_{io} \text{ and } h_{ci} > h_{crit} \\ \hat{K}(H_i - \rho_{is} b), & H_i > h_{io} \text{ or } h_{ci} \leq h_{crit} \end{cases} \tag{6.34}$$

where h_{io} are the saddle heights of the chosen boundary edges. This allows q_{is} to account for both drainage and porous drainage.

Figure 6.11 shows the results of several experiments on a set of four random topographies. By design, side drainage only plays a role until porous drainage begins. We see that in all cases, just as in Section 5.6, side drainage puts a cap on the pond fraction that can be reached on the floe, and again, this is dominated by the percolation properties of the graph of catchments. In all cases, this control of the pond fraction brings the results of the experiment into line with some observed maximum area fractions, such as those reported by Polashenski et al. [2012], which include observations of $a/a_0 \sim 0.5$. It should be noted that side drainage is a factor in those sets of observations, but may not be for observations of ponding in the central Arctic Ocean, further from the marginal ice zone.

Experiments are distinguished by the number of open catchments along the sides that are allowed to drain. In each case we show the results from realisations of the model on four random topographies. In Figure 6.11, the red 'ridges' experiment is the control where there are no openings along any side, equivalent to a field of flat ice completely surrounded by ridges. The blue lines show pond fraction when the lowest saddle along one side is allowed to drain only. This may represent a field mostly hemmed in by ridges. For the green, the lowest saddle on each of two opposing sides are allowed to drain and the black lines show a completely open floe, where all catchments along the sides of the floe are allowed to overflow to the sea (to avoid multi-edges at the corners, only the lowest saddle on a corner catchment is allowed to overflow).

For the experiments allowing only one overflow we see a significant difference in the maximum pond fraction reached, both compared to the case with no overflow, and between ensemble members in the simulation. This variance between different ensemble members narrows once another side is opened up. The restriction on what boundary catchments overflow is allowed on changes the effect that percolation in the pond system has on side drainage, as in this case the system has to connect to a small

number of particular catchments, rather than any catchment along particular sides. We see that for the open floe (black), pond fraction only reaches a point near (but less than) the area at the percolation threshold. We can rationalise by noting that during the impermeable phase, even the most central catchment within the floe can shed water to the sea without needing to be connected to catchments on two sides of the floe. Note that these curves also display more noise than the others, as the abundance of overflowing catchments leads to numerical overshoot in equation (6.34) for the side drainage flux, as the water levels constantly move above and below the heights of the boundary saddles until drainage begins.

A system size of 64 nodes was used in these experiments, as the incorporation of a drainage flux over the sides of the floe is quite costly. This is due to the fact that drainage fluxes quickly turn on and off due to numerical drift of the integrator (this is seen in the 'jitter' of the trajectories in Figure 6.11 while side drainage occurs. We acknowledge that this smaller system size does introduce some finite-size effects. However, we expect much of the behaviour to stay the same as we increase the system size, with the exception that for any given simulation, the pond fraction seen in the plateau during side drainage tends to $a(p_c)$, the area fraction at percolation, as $n \to \infty$.

6.5 Summary

In this chapter we have incorporated the effects of buoyancy, the finite thickness of the ice, and the eventual porous drainage of accumulated pond water through the ice into the model. The modelling of this process in one dimension and later two dimensions was described in Section 6.1. This included a test of the model equations against a numerical simulation of a cylindrical shaped pond in one dimension. We found that beyond a value of $\hat{K} = 1000$, increases in the drainage parameter do not affect the behaviour of the model. We also addressed the adaptation of the ice topography due to preferential melting in ponds, and found that it was acceptable as an approximation to leave the slope of the ice topography unchanged for the purposes of calculating the relationship between water level and pond area.

In Section 6.2, we described how the model for drainage can be incorporated into the network model and discussed some numerical and algorithmic considerations needed to keep the physics of a numerical simulation realistic. We also described the phenomenon of 'dead ponds', catchments that fill initially and then drain completely, and how we account for this behaviour in the model.

The output from one example simulation of the network model was presented and discussed in Section 6.3.1. In the example, the model was shown to clearly resolve all of the major stages of pond evolution: filling, drainage and refilling. A sensitivity analysis of the model was shown in Section 6.3.2. The results show that the pond extent, the rate of ice loss and the timing and duration of stages of pond evolution are all strongly controlled by the parameters, and a wide range of behaviour is seen across the parameter space investigated. These changes are explored in more detail in the next chapter. The variety of behaviour observed in the model output suggests that changes in the parameters across different regions of the Arctic have the potential to lead to large differences in observed pond extent and volume loss, and that changes in the conditions in the Arctic in the future may affect the extent and behaviour of pond systems. We discuss this in more detail in Chapter 8.

Finally, in Section 6.4, we incorporated both side drainage and porous drainage into the model. While this was found to be computationally expensive, a small set of experiments showed that the side drainage had strong control on the maximum area fraction reached before drainage begins, but almost no effect thereafter. This indicates that a lower maximum pond fraction is likely to be seen on floes in the marginal ice zone, where floes tend to be smaller, and edge effects are more consequential.

Chapter 7

Analysis of the model

7.1 Introduction

With the introduction of drainage in Chapter 6, we have a model that mathematically represents the evolution of ponds from early to late summer. A key goal of modelling is to aid our understanding of the systems they are built to represent. For the most part, this amounts to furthering our knowledge of how the systems work, in order to illuminate processes and behaviours that were previously hidden. In some cases a model might even be good enough to predict aspects or possible behaviours that have not been observed before, or even to predict how a system might behave in the future under new conditions. With this in mind, in this chapter we use our model to pursue several routes of investigation and answer several questions on pond systems.

Section 7.2 describes a set of experiments where we explore model output across the parameter space of the roughness scale Δh and the ice thickness threshold for porous drainage h_{crit} in order to probe the range of behaviours for the temporal evolution of pond extent. With help from these experiments we develop a taxonomy of time scales for the different processes involved in our model, and derive estimates for these time scales and the corresponding pond extents. We also derive an approximate expression for volume loss throughout the melt season.

In Section 7.3 we investigate why it is sometimes observed that there are floes without ponds near floes with ponds, by quantifying the fraction of ponds that drain completely in the experiments described in Section 7.2. In Section 7.4 we compare our model to the field experiment carried out by Polashenski et al. [2012] in order to test how closely our model replicates the reality it was designed to represent and interrogate.

In Section 7.5 we compare the output of our model to the geometric behaviour of pond systems described by Hohenegger et al. [2012], which is represented by two

regimes of pond, each with a different perimeter roughness, or fractal dimension D, which we defined in Section 2.3.5. We also explore a way to delineate between the two regimes. Finally, in Section 7.6, we explore the different percolation processes identified in Chapters 5 and 6 and discuss what consequences they have for network function and the behaviour of the melting process.

7.2 Dependence of pond extent on surface roughness and the onset of drainage

7.2.1 Exploring the parameter space

In Section 6.3.2, we presented a sensitivity analysis to four of the variables in our model, and observed a wide range of behaviours. In this section, we explore in more detail the dependence of the model to the two variables that we expect to change most between floes, and identify and catalogue possible behaviours that we may see in real pond systems. In developing the model, and by a comparison with the literature, we are able to deduce that variation in the behaviour of the model is mostly dependent on the variables Δh and h_{crit}. This is done by a process of elimination. In Chapter 6 we estimated that a $\hat{K}$ of 1000 is consistent with the permeabilities measured by [Polashenski et al., 2017]. The densities of ice, fresh water and sea water are not expected to change enough to affect the model. From the observations in Fetterer and Untersteiner [1998] we expect that the melt rate of ponded ice is consistently around twice that of bare ice. We know that melt rates in the Arctic depend on latitude and can also vary year to year with atmospheric conditions but this will impact melting of bare and pond covered ice similarly. As we have used the bare ice melt rate to scale time, this rate will not affect the behaviour of our dimensionless model. We also demonstrated in Chapter 6 that pond extent in the later part of the melt season is much less dependent on changes in v_w than on the other three possible variables. We saw that the behaviour of the pond fraction changed more consistently with changes in v_w than with some of the other parameters. Therefore, we do not need to explore variations in the melt rates further. Thus, we decide that $\hat{K}$ and v_w will be held fixed in this chapter (along with the densities of ice, fresh-water and sea-water), and Δh and h_{crit} varied.

In order to capture the expected range of behaviour of the model, we perform a set of experiments. Each experiment is composed of six synthetic topographies of 100 catchments on a square lattice (results in this section give similar behaviour for both

square and random lattice topographies). In each experiment, the initial random number generator seed is fixed and identical, such that comparable topographies are generated for each experiment, except for the roughness. We vary Δh from 0.05 to 0.2 in increments of 0.025, and vary h_{crit} from 0.4 and 0.8 in increments of 0.05, and carry out numerical experiments so as to cover both ranges (for a total of 63 experiments). The range of Δh is chosen from observations taken during fieldwork, and from the literature to represent both very flat first-year ice, and deformed multi-year ice, assuming the scale length for ice thickness $\tilde{h} \sim 1 - 1.5$ m, and using data from [Nolin and Mar, 2019] and [Polashenski et al., 2012], the second of which is also discussed in Section 7.4. The range of h_{crit} is more speculative. The upper end is chosen to coincide with $\bar{h} - \Delta h$ for the roughest ice, such that the lowest centre would start to drain immediately after the start of the simulation. The lower end is chosen speculatively, assuming that drainage is likely to start when more than half of the thickness of the floe has melted away from the pond floor.

Figure 7.1 shows the area-fraction time series for experiments which use the ends of the ranges in the parameter sweep. They show four distinct types of behaviour of the model. The top left, (a), represents smooth and comparatively impermeable ice that floods quickly, remains flooded for a period of time, then drains late. Drainage is fast and drains ponds completely, but pond coverage eventually rebounds before the end of the simulation. The top right, (b), representing smooth ice that drains early, shows the same fast filling but with little time spent flooded, and only recovers any ponding much later in the summer (this is not visible in the plot, as the horizontal axis in this panel has been truncated to be consistent with the others). In this case, due to the complete emptying of ponds, it takes far longer for the ice surface to melt all the way through. The ponds begin to drain before the pond floors reach sea level, and once drainage ends, almost all of the ice is bare, and from then onwards melts at the slower bare ice melt rate. Some small amount of increase in pond area is seen, but only once the ice is less than $0.25\,\bar{h}$ thick.

The bottom left panel, (c), shows rough ice that drains late. Here we see ponds flooding, but then draining immediately, and, later into the season, retaining a larger amount of pond area compared to smoother ice. However, the existence of ponds throughout the season leads to their melting through the ice much faster than those floes that lose ponding. Finally, the lower right, (d), shows rough ice that drains early. In this case, only a low extent of ponding is seen in the impermeable phase before draining. The floe loses pond extent quickly, then slowly regains a small amount as the summer progresses. For simulations in (c) and (d) the area growth post-drainage

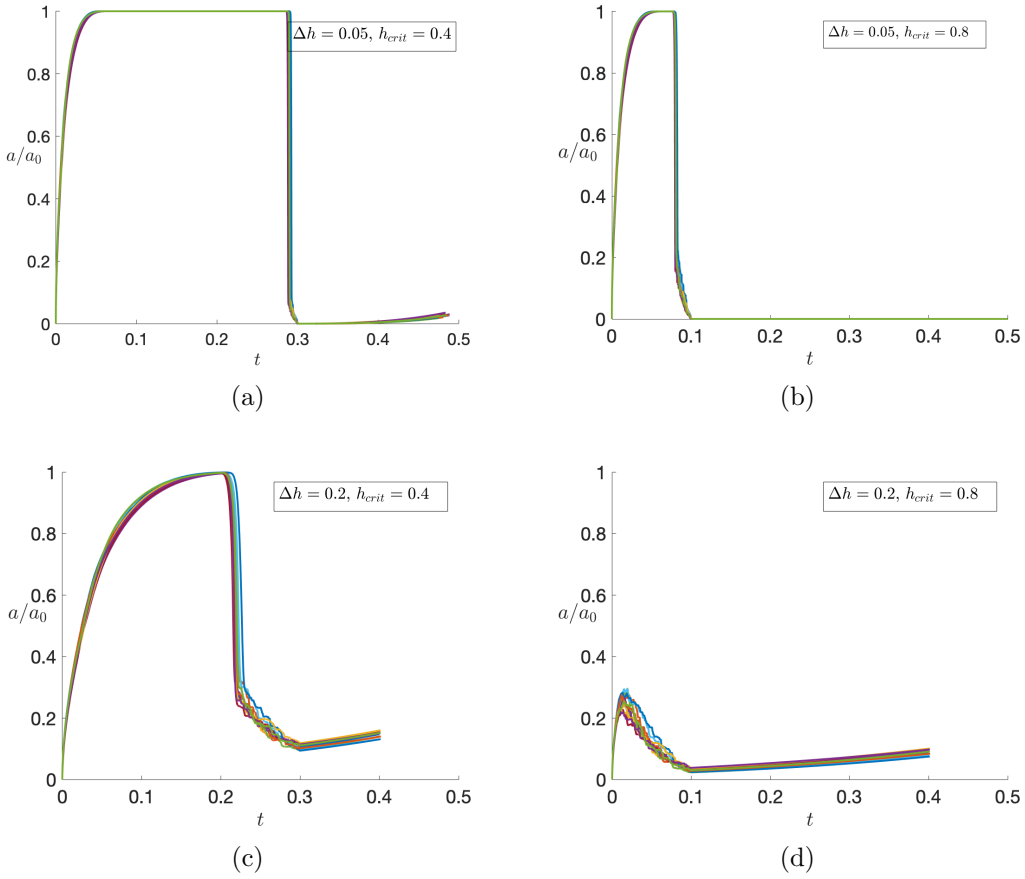


Figure 7.1: The end members of a set of experiments spanning the parameter space of surface roughness Δh and critical thickness for drainage h_{crit}. Each plot shows 12 realisations of a simulation of a floe of 100 square lattice catchments. A melt parameter of $v_w = 2$ was used.

is slower due to the higher ice roughness. As during the filling stage (c.f. Chapter 3), the rate of pond growth during refilling approximately scales with $1/\Delta h$. This can also be seen in Figure 6.9(c). The higher roughness is also responsible for the amount of residual pond area still left immediately once drainage ends, as the floors of ponds that grow on rougher ice will sit lower in the water than those that grow on flat ice.

These four end members of our set of experiments each show distinct forms of pond evolution and coverage. For floes that have intermediate values of Δh and h_{crit} the behaviour of pond evolution moves steadily between these forms. This will be the focus of the next sections.

7.2.2 Deriving scalings and approximations for pond evolution from the model

In order to understand the range of behaviour of model described above and shown in Figure 7.1, and to draw conclusions for the future of the ice pack in Chapter 8, we derive time scales for each of the important processes in our model, approximations for the area fraction during filling and refilling, and an approximation for the volume of water melted during the course of the simulation.

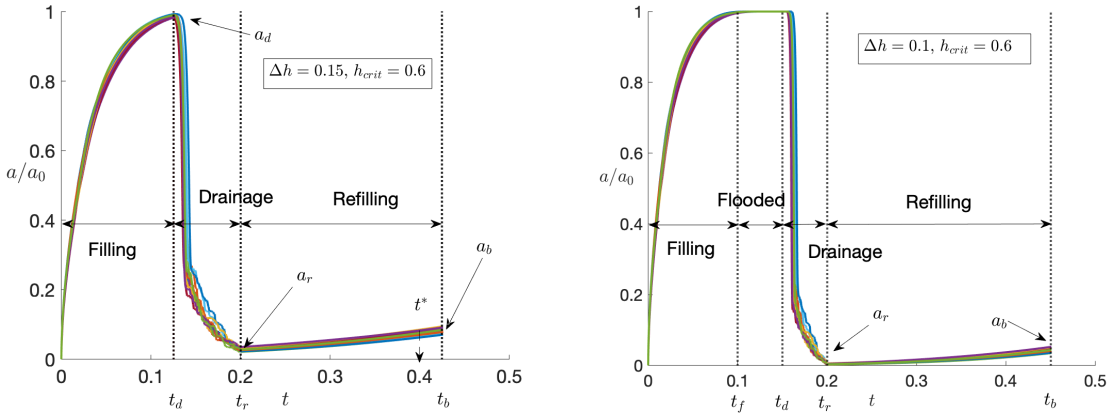


Figure 7.2: Example area fraction plot describing important features. (a) A case that neither floods nor drains completely. (b) A case that floods and briefly drains completely. Timescales for flooding t_f, onset of drainage t_d, end of drainage t_r and breach of the base of the floe t_b are illustrated, with corresponding areas.

We divide the evolution into three sections, as shown in Figure 7.2. Filling (and in some cases flooding) occurs during the initial impermeable phase. We define drainage as the period from the time at which the first catchment begins to drain to the time at which the last one does, at which point the whole floe is permeable. This approximately coincides with the minimum of area after the ice becomes permeable. Refilling occurs during the remaining period until the first of the pond floors reach the base of the floe. For flat ice, there are also periods where the floe is flooded, which we consider part of the filling phase. There may also be a time post-drainage where catchments are empty, which we consider to be part of the refilling phase. We label the time of flooding t_f, the time at which drainage starts t_d, the time at the end of drainage t_r and the end of the simulation (the 'breach') t_b. These have corresponding areas with the same subscripts. On panel (a) of the figure we also label a time t^*, taken to be 0.4, a common time before the end of each simulation at which we make

comparisons of area fraction and melted volume (volume loss from the floe), which is also used in Section 7.3 to explore the complete loss of ponding.

In our analysis, we first assess time scales for the processes from the details of the model. We then check our scalings. For the time of flooding, we do this by rescaling the time series of area fractions for several experiments across the parameter space of Δh and h_{crit}, shown in Figure 7.3(b). In that figure, the unscaled time series are shown in the top panel. For the other time scales, we compare the times of the events in the simulations used in (a) with the corresponding calculated time scales. These are shown in panels (c) to (e). Finally, with the aid of these time scales, we derive approximations for the area fraction and melt volume during the summer-time evolution of ponds.

7.2.3 Time scalings

In the early melt period, the floe floods once the highest peak is covered by water. The water level in each catchment is the same at this time, and the melt-water must fill all of the cavities. The volume to be filled scales with Δh and in the limit $\rho_{iw} \approx 1$, the water level once these cavities have been filled is $\bar{h}$, assuming equal volumes of ice below $\bar{h} - \Delta h$ and above $\bar{h} + \Delta h$. The time taken for the highest point, which is expected to have an initial height $\bar{h} + \Delta h$, to reach $\bar{h}$, given that it is bare ice until the point of flooding, is then

$$t_f \approx \frac{\Delta h}{v_b}. \tag{7.1}$$

Figure 7.3(b) demonstrates that this is a valid approximation, as we have rescaled time by t_f here, and the curves for all experiments allign for the impermeable period.

To estimate the time at which drainage starts, we reason that this occurs once the lowest centre melts to the ice thickness h_{crit}. Given the expected lowest centre starts with height $h_c \approx \bar{h} - \Delta h$, we reason that this occurs at time

$$t_d \approx \frac{\bar{h} - h_{crit} - \Delta h}{v_w}. \tag{7.2}$$

Figure 7.3(c) shows that this time scale matches the calculated time taken for drainage to begin for most of the experiments shown, and closely matches the rest.

By the same logic, drainage ends once the highest centre, expected to lie initially at $\bar{h}$ in the limit of infinitely many catchments, reaches h_{crit}, giving

$$t_r \approx \frac{\bar{h} - h_{crit}}{v_w}. \tag{7.3}$$

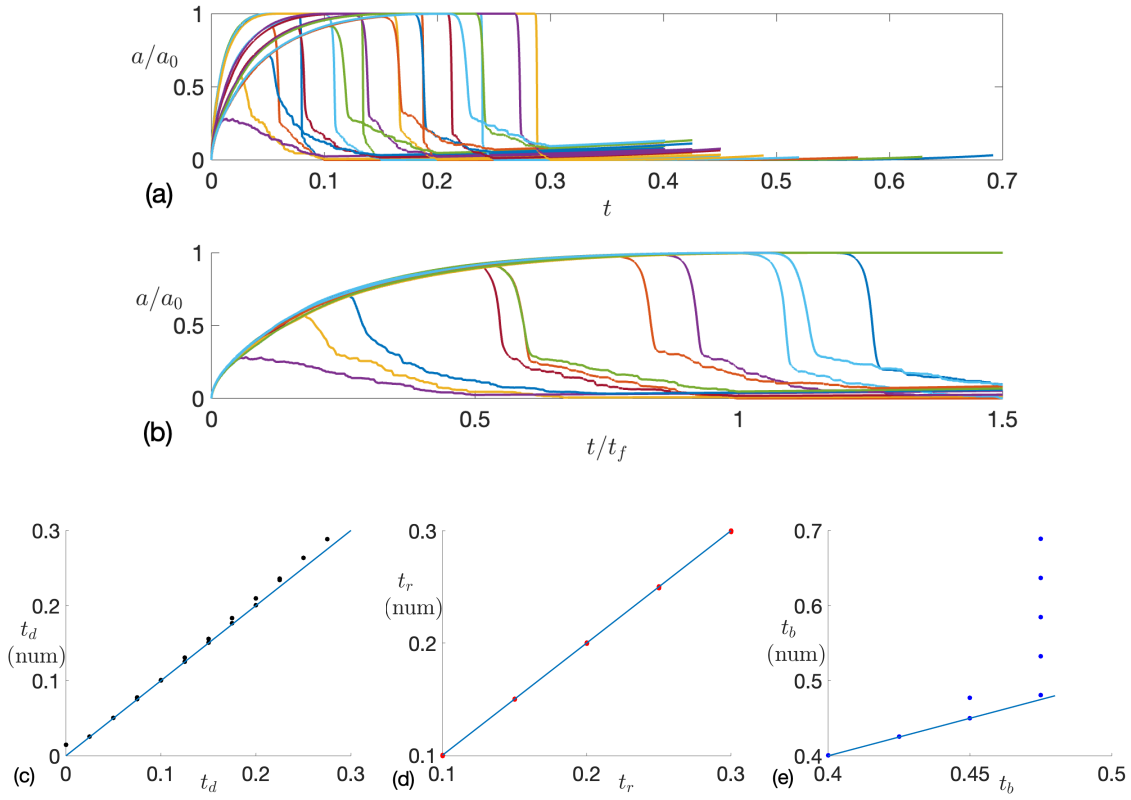


Figure 7.3: (a) Area fraction time series for the experiments described in 7.2.2, square lattice topographies of 100 nodes, with $v_b = 1$, $v_w = 2$, $K = 1000$, and across $h_{crit} = \{0.4, 0.5, 0, 6, 0.7, 0.8\}$ and $\Delta h = \{0.05, 0.1, 0.15, 0.2\}$. (b)An ensemble member was chosen at random for each experiment, and rescaled by: (b) equation (7.1) the flooding timescale. (c) Equation (7.2), the onset of drainage time scale compared with the average calculated onset of drainage for each set of simulations (t_d num). Also plotted is a line of slope 1 and intercept $(0, 0)$. (d) Equation (7.3), the end of drainage time scale compared with the average calculated end of drainage. (e) Equation (7.4), the breach of the base time scale compared with the average calculated time to breach.

Figure 7.3(d) shows that this time scale matches the calculated time taken for drainage to end for all of the experiments shown.

For a rough topography, the time to breach (the time taken for the lowest pond centre to reach the base of the floe) is

$$t_b \approx \frac{\bar{h} - \Delta h}{v_w}. \tag{7.4}$$

For a smooth topography, this is more difficult to reason, as ponds are absent for much of the summer. We may reason that in this case that

$$t_b \approx \frac{\bar{h} - \Delta h}{v_b}, \tag{7.5}$$

which assumes for most of the summer the whole floe is bare, given that initial flooding and drainage happens quickly, such as in Figure 7.1(b). Figure 7.3(e) shows the comparison of equation (7.4), the time scale for rough ice with the average calculated time to breach for the selected experiments shown in (a). The time scale agrees with the calculated times for experiments that do not completely drain. Points that do not agree with the time scale represent experiments where the floe drains completely for at least some period of time. In these cases, the ice at the lowest points on floes melts with velocity $-v_b$ for some period of time, which accounts for the discrepancy between calculated time to breach and the time scale for breaching on rough ice.

7.2.4 Area fraction scaling

We now derive approximations for the area fraction at the start of pond growth and near the flooding of floes. Our method relies on an assumption that during these periods, the whole floe behaves in a similar manner to how an average catchment behaves.

At early times, we can use the asymptotic solution derived in Chapter 5 to estimate the area fraction, with some simplifications. We note that at early time, ponds are unlikely to have joined, so we can assume that each catchment behaves independently to others. We find that approximating that the lower part of each catchment has the shape of a square pyramid improves the accuracy of our approximation for area, which implies that we must take the saddle heights and peak heights to have the same height $\bar{h}$ and the pond centres to have height Δh.

The approximation (5.17) for water level at early times yields

$$H(t) = \left(3\frac{\rho_{iw} v_b a_0 t}{C}\right)^{1/3}. \tag{7.6}$$

The scale factor for the square pyramid is $C = a_0/\Delta h^2$, and the area is related to the water level by

$$\frac{a}{a_0} = \frac{H^2}{\Delta h^2}. \tag{7.7}$$

The approximation for the area fraction at early times for our array of pyramids is then

$$\frac{a}{a_0} \simeq \left(\frac{3\rho_{iw} v_b t}{\Delta h}\right)^{2/3}. \tag{7.8}$$

For late times, we can employ a similar method, whereby we rewrite the equations in terms of $r = 1 - a/a_0$, and aim to expand for $r \ll 1$ near flooding. For this case we assume that all ponds are joined, and all catchment segments have shape

3 (as described in Section 5.3), a quadrilateral, where the segment is almost completely covered apart from a region around the associated peak. The area fractions of catchments are given by

$$\frac{a}{a_0} = 1 - \frac{(\tilde{H} - z_p)^2}{(z_p - z_s)(z_p - z_0)}. \tag{7.9}$$

Using the averages for point heights derived in Section 5.2, we approximate $z_p - z_s \simeq \Delta h/2$ and $z_p - z_0 \simeq \Delta h$, so we can rearrange and differentiate (7.9) to find

$$\dot{H} + v_b \simeq \frac{\Delta h}{2\sqrt{2}} r^{-1/2} \dot{r}. \tag{7.10}$$

The water level equation for a catchment (5.16) can be written in terms of r as

$$\dot{H} + v_b = \frac{\rho_{iw} v_b}{1 - r} - (v_w - v_b)(1 - \rho_{iw}), \tag{7.11}$$

so equating with (7.10) gives

$$\frac{\Delta h}{2\sqrt{2}} \dot{r} = \rho_{iw} v_b \frac{r^{1/2}}{1 - r} - (v_w - v_b)(1 - \rho_{iw}) r^{1/2}. \tag{7.12}$$

Expanding for small r, we find

$$r^{1/2} \approx \frac{\sqrt{2}}{\Delta h} \left[\rho_{iw} v_b - (v_w - v_b)(1 - \rho_{iw})\right] (t_f - t) \tag{7.13}$$

where we have used the condition $r = 0$ at $t = t_f$. Our approximation for the area fraction close to flooding is therefore

$$\frac{a}{a_0} \simeq 1 - \left\{ \frac{\sqrt{2}}{\Delta h} \left[\rho_{iw} v_b - (v_w - v_b)(1 - \rho_{iw})\right] (t_f - t) \right\}^2. \tag{7.14}$$

Solutions for equations (7.8) and (7.14) are plotted in Figure 7.4, against data from a simulation using a random lattice topography and $\Delta h = 0.15$, which shows good agreement in both cases.

From a networks perspective, the method of averaging and approximating we have described here is equivalent to a node-based mean-field approximation. For the early stage of the impermeable phase we have been able to do this because ponds can be considered independent. For the later part near flooding, the system is in effect fully coupled and the activity at each node converges to a consensus or become synchronised with one another, which again, allows us to look at the mean field. We cannot assume anything equivalent during the middle period of filling, as nodes are coupled to some

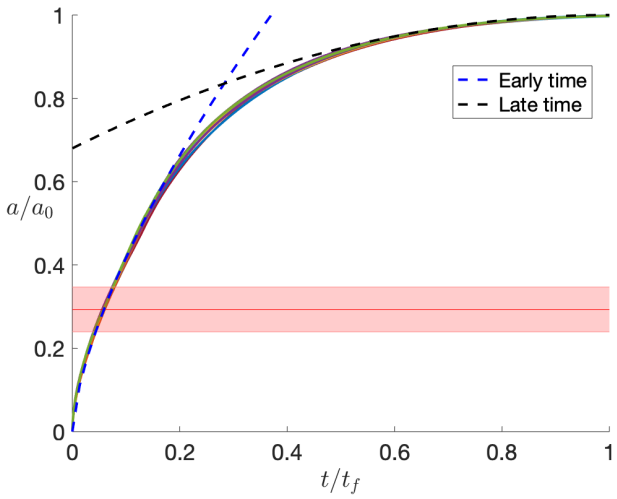


Figure 7.4: Pond fraction during the impermeable phase, for a Voronoi tessellation of 225 catchments with randomly placed centres, using $v_w = 2$, $\Delta h = 0.15$, with 12 realisations shown. Time is scaled by the filling timescale t_f, as in figure 7.3(b). Marked are asymptotic solutions for $a/a_0 \ll 1$, equation (7.8), the dashed blue line marked 'early time', and $1 - a/a_0 \ll 1$, equation (7.14), the black dashed line marked 'late time'. The plot also shows the average area at percolation, the solid, horizontal red line, calculated from the data. The shaded area represents one standard deviation.

of their neighbours and the couplings are too non-linear for us to easily derive an approximation for the coupling, or a statement of moment closure. The same is true of the draining phase, so we do not present an analytical approximation for the area during this period.

It should be possible to use a mean-field approximation to estimate the behaviour post-drainage, as this is another period of synchronization. During this time the network behaviour is dominated by interactions between catchments and the sea node, through volume loss and changes in buoyancy. However, to estimate this behaviour, we need to know the water level and draft immediately after drainage ends, which are difficult to estimate. For this reason, we do not present a detailed attempt at approximating the behaviour post-drainage.

Porter and Gleeson [2016] discuss how difficulties emerge in deriving mean-field behaviour for dynamical systems on networks when the time scales for changes in network structure are comparable to changes in the dynamics, as is the case for our model mid-way through filling and especially during drainage. However, given that our model displays consistency in behaviour during these periods across much of the parameter space, our model may prove a useful one in which to explore methods to derive mean-field behaviours while both the dynamics and network topology are

rapidly changing.

7.2.5 Volume loss

As discussed in Chapter 3 the melted volume in each catchment can be expressed as

$$\dot{V}_m(t) = \rho_{iw} v_b a_0 + \rho_{iw}(v_w - v_b)a(t). \tag{7.15}$$

Hence

$$V_m(t) = \int_0^t \dot{V}_m(t')\mathrm{d}t' = \rho_{iw} v_b a_0 t + \rho_{iw}(v_w - v_b)\int_0^t a(t')\mathrm{d}t' + V_m(0). \tag{7.16}$$

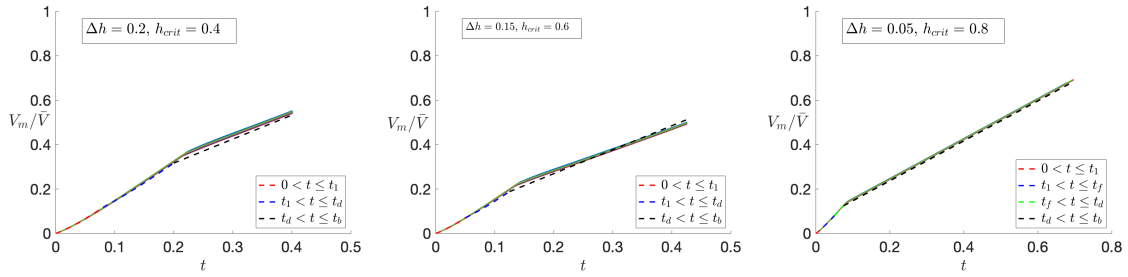
We can use the approximations for $a(t)$ from the previous section to compute $V_m(t)$ up to t_f, or t_d in the case $t_d < t_f$. For the draining and refilling period, we make an approximation that the area fraction remains constant throughout, i.e. $a \approx a_b$, and quantify a_b by inspecting the results of simulations. Based on the results shown in Figure 7.1, we approximate that $a_b \approx 0$ for simulations in which complete drainage of floes occurs, and $a_b \approx 0.2$ for simulations in which complete drainage does not occur, and use these values during the refilling period, from drainage onwards. An investigation into the conditions for when $a_b \approx 0$ is presented in the next section.

To summarise, for the purpose of calculating the melted volume through time we approximate area fraction at time t to be

$$\frac{a(t)}{a_0} \simeq \begin{cases} \left(\frac{3\rho_{iw} v_b t}{\Delta h}\right)^{2/3} & 0 \le t < t_1 \\ 1 - \left\{\frac{\sqrt{2}}{\Delta h}\left[\rho_{iw} v_b - (v_w - v_b)(1-\rho_{iw})\right](t - t_f)\right\}^2 & t_1 < t \le \min(t_f, t_d) \\ 1 & t_f \le t \le t_d, \\ 0 & t_d < t \le t_b \,\text{and}\, a_r = 0, \\ 0.2 & t_d < t \le t_b \,\text{and}\, a_r > 0 \end{cases} \tag{7.17}$$

where we have chosen a switching time $t_1 = 0.4 t_f$ between the two approximations for area fraction during the impermeable stage. Note also that the third case in 7.17 is only valid if $t_f < t_d$. We then estimate the volume loss by integrating equation (7.16), using (7.17), and summing over all catchments.

Figure 7.5 shows melted volume from three sets of simulations, across the parameter space, with equation (7.16) (using (7.17)) overlain. Good agreement between simulations and approximations is seen for each set, as long as we choose a_b appropriately.



(a) (b) (c)

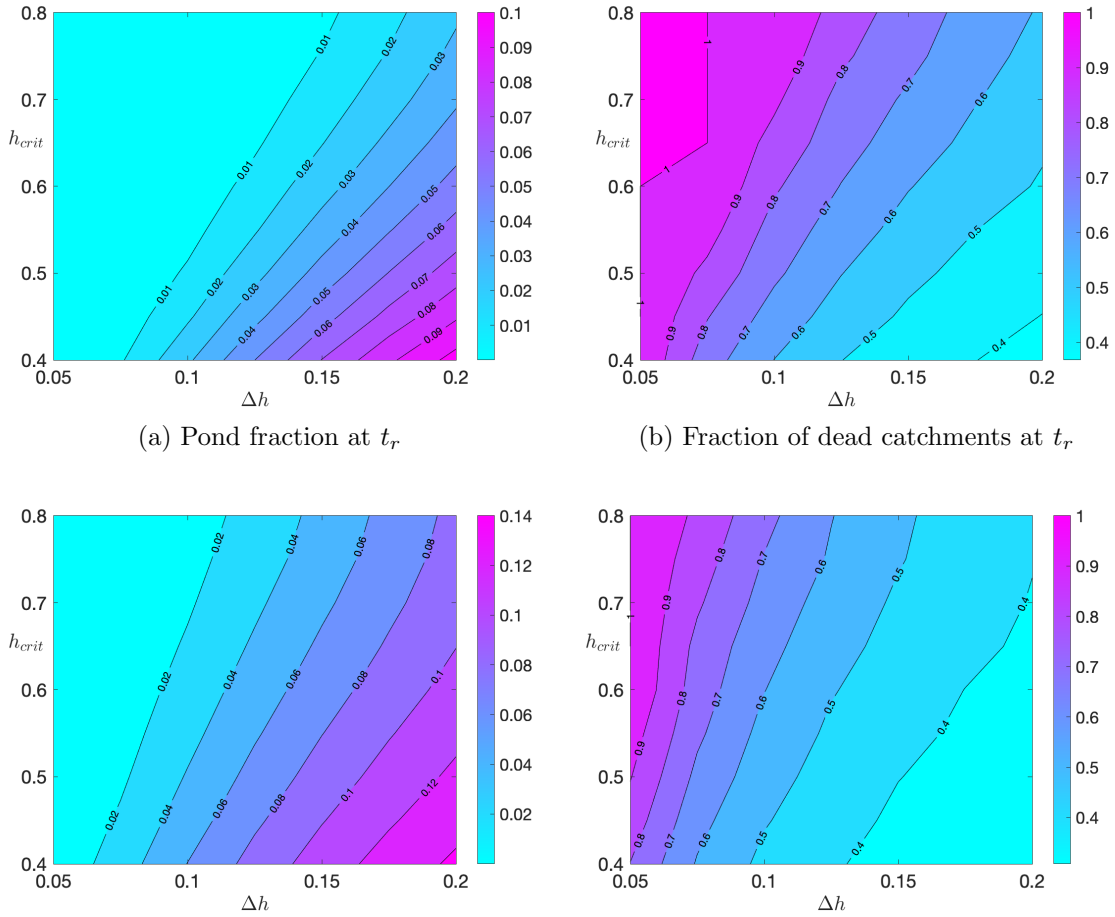
Figure 7.5: Melt production as a fraction of total initial ice volume for three simulations (solid lines) with different parameter sets Δh, h_{crit}. Each simulation has six ensemble members and is composed of a square lattice topography of 100 catchments. Here $v_w = 2$ and $K = 1000$. Dashed lines show the approximate solution, equations (7.16) and (7.17). Individual coloured solid lines may be too close to distinguish. In (a) and (b), we choose $a_b = 0.2$. In (c) we choose $a_b = 0$.

7.3 Why are there floes without ponds?

If one sails through the pack ice past the marginal ice zone to where ponds form, there is a striking curiosity. Ponds do not form on all undeformed regions of floes, and one may observe heavily ponded floes, and floes with a complete absence of ponds in the same day of sailing. The absence of ponds on some floes has been noted in the community [Perovich et al., 2002]. However, it does not seem to have been addressed in any modelling papers, save for the work of Popović et al. [2020a], who estimated conditions for the persistence of ponds based on a thermodynamic and statistical model of ponding during the impermeable stage. They concluded that smooth ice, and ice in regions with low melt rates are more likely to remain pond free.

This question is of interest for two reasons. Firstly, it is a nice geophysical curiosity to attempt to explain. Secondly, the absence of ponds in areas where they would be expected would lead to an important error in parametrizing pond coverage in climate models. Parametrizations may overestimate pond coverage in this case. This is especially relevant for any parametrization that assigns a constant pond fraction on floes in all grid cells, or assigns a time dependent pond fraction function that is the same across all floes in all grid cells. Therefore, determining the conditions under which ponds initially grow and then disappear completely could lead to important corrections and improvements to pond parametrizations. In this section, we use our model to determine under what conditions ponds initially form on ice and then disappear during drainage, never to re-emerge. We argue that the combination of smooth

ice and early drainage leads to the absence of ponds where they might otherwise be expected.



(a) Pond fraction at t_r

(b) Fraction of dead catchments at t_r

(c) Pond fraction at $t^* = 0.4$

(d) Fraction of dead catchments at $t^* = 0.4$

Figure 7.6: Contour plots from sets of simulations varying Δh and h_{crit}. Each simulation consists of six ensemble members of different topographies of 100 catchments on a square lattice. $v_w = 2$ and $K = 1000$ were used. (a) Pond fraction at the end of drainage. (b) Fraction of dead catchments determined by area at the end of drainage, determined by counting the number of catchments with area fraction below a threshold $\epsilon a = 10^{-4}$. (c) Pond fraction at the chosen common end point $t^* = 0.4$. (d) Fraction of dead catchments determined by area at the chosen common end point $t^* = 0.4$. The contour plots are made up of 7 values of Δh and 9 values of h_{crit}, with each data point being the average over six ensemble members.

Given the observation of ponded floes existing close to unponded floes, we assume that the rate of enhanced melting is not a major factor, since prevailing solar and meteorological conditions should be similar for neighbouring floes. At the beginning of this chapter we explained that the behaviour of ponding is least dependent on

v_w, and, above a certain value, changes in the drainage parameter $\hat{K}$ do not affect behaviour greatly. In regions of the ice where atmospheric conditions are sufficient for melting, we propose that the presence or absence of ponds late in the season are caused by differences in surface roughness and the onset of drainage, controlled by Δh and h_{crit}.

To quantify the size of these changes we derive a measure for 'dead' ponds, that grew early in the season, but do not reappear post drainage. We define the 'dead fraction', to be the fraction of catchments that have zero area at time t,

$$n_d(t) = \frac{1}{n}\sum_{i=1}^{n} \mathbb{I}[a_i(t) \leq \epsilon a_0]. \tag{7.18}$$

Specifically, we call this the 'area based dead fraction'. Here n is the number of catchments in the floe, ϵa_0 is the negligible area assigned to ponds with lowest points that lie above sea level, as described in Chapter 6, and we are making use of an indicator function $\mathbb{I}$ that satisfies the condition in braces. Recall that in numerical simulations we set a minimum area $\epsilon a_0 = 10^{-4}$ to avoid the singularity in the equation for $\dot{H}$ at zero area, and a catchment with this much pond area is deemed to be empty in reality.

Considering the set of experiments on a square lattice floe described in Section 7.2, Figure 7.6 shows contours of the area fraction and dead fraction in panels (a) and (b) respectively at the end of drainage, for changing Δh and h_{crit}. Panel (a) quantifies and expands on the differences in area fraction seen across simulations with different parameters in Figure 7.1 at the time of drainage, such that we can see the area fraction this time across the whole range we have explored. Here we find that early drainage and flatter ice both promote a reduced area fraction and larger fraction of dead catchments. Figure 7.6 (c) and (d) show contours of the area fraction and dead fraction respectively at time $t^* = 0.4$, after drainage and a long period of refilling (and immediately before the deepest pond across the whole set of experiments would breach the base of the floe). These tell the same story qualitatively, but with more extensive ponding, and some refilling of empty catchments.

However, does this conclusion hold if we account for the compromise we have made by ignoring changes to the topography due to preferential melting in ponds? In Section 6.1.3 we addressed the difficult issue of accurately representing the hypsometry of the ice surface during refill, given that the initial melting modifies the initial topography. Figure 6.3(a) captures the differences in the evolution of the pond fraction between a model that accurately accounts for changes in the hypsometry due

to preferential melting and the simplified catchment model used in the experiments presented in this section. During refilling, the two models behave qualitatively similarly, tracing similar paths, but with the more accurate model having a consistently higher pond fraction. Therefore, these results in effect show a lower bound on the area fraction after refilling, and an upper bound on the dead fraction.

7.4 Comparison with field data

Our model was built based on the behaviour described in observations, and has been shown so far to be broadly consistent with them. However, we must still evaluate whether it can qualitatively replicate the data on pond evolution. The richest dataset on pond evolution is presented by Polashenski et al. [2012]. We have already used the Lidar data from their field experiment to evaluate our synthetic topographies in Chapter 5.

In this section, we estimate parameters for our model from the data taken during 2009 on two sites, and compare the area fraction from a model run using these parameters to the area fraction calculated from aerial photography taken during the same field expedition. Data from two sites, denoted the 'north' and 'south' sites, are shown in Figures 7.7 (a) and (b), respectively. For the north site, the pond fraction time series was taken from Polashenski et al. [2012]. For the south site, aerial photographs were retrieved from the online archive of data from Polashenski's field campaigns.[1] These were classified using the OSSP algorithm, a machine learning classifier for sea ice and melt ponds [Wright and Polashenski, 2018], from which we were able to determine pond fraction.

The mean ice thickness at the north and south sites was estimated, from the Lidar data and the length of ice cores taken at the two sites, to be 1.25 m and 1 m, respectively. The surface deviation Δh for each was calculated by the method described in Section 5.2 to be 0.25 m and 0.18 m. This gives a non-dimensional $\Delta h/\bar{h} \approx 0.2$ for both experiments. The topographic parameters we calculate are in line with calculations by Popović et al. [2020a], where they used the same data to get scale lengths for their model. Polashenski et al. [2012] note that during the experiment, the atmospheric and solar conditions created the potential to melt a maximum possible 0.023 m/day of ice, with a 2 : 1 ratio of ice albedo to pond albedo. Based on the observations of Fetterer and Untersteiner [1998], and melt rate measurements derived

[1] https://chrispolashenski.com/data.php

from data taken by the CRREL-Dartmouth Mass Balance Buoy Program we take the melt rates v_b and v_w to be 0.01 and 0.02 m/day respectively.[2]

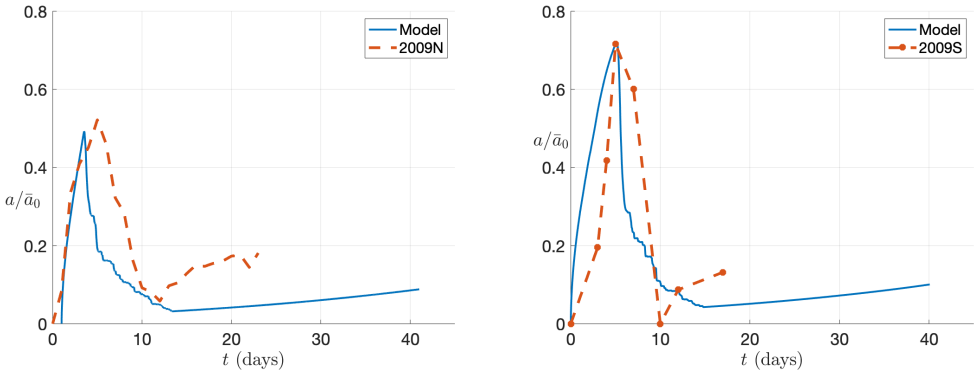
We take $\hat{K} = 1000$ as we have for simulations in much of the rest of this work, based on the chosen melt rates and the thickness scale for the floe $\bar{h}$. Based on the order of magnitude of the number of catchments seen in aerial photographs of the field site, we run a simulation of 100 randomly placed catchments on a random lattice (Voronoi tessellation). There is sparse information in the literature to estimate h_{crit}, the last undetermined parameter in the test against observations. We investigated several values of h_{crit} and find that for a non-dimensional value of 0.75 for the north site, and 0.7 for the south site, the model fits the data well. The list of parameters used is summarised in Table 7.1. Simulations terminate when the ponds should break through the ice, which is controlled by the topography and melt rates. The field data ends when the group had to leave the ice as it became too thin to safely work on.

Parameter	North	South
v_b	0.01 m/day	0.01 m/day
v_w	0.02 m/day	0.02 m/day
$\bar{h}$	1.25 m	1.0 m
Δh	0.25 m	0.18 m
$\Delta h/\bar{h}$	0.2	0.2
$\hat{K}$	1000	1000
$h_{crit}/\bar{h}$	0.75	0.7

Table 7.1: Parameters used in test against data.

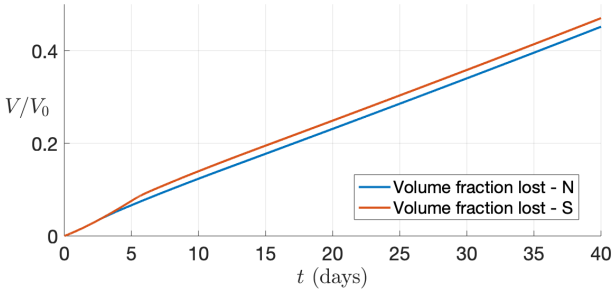
We see in Figure 7.7 that in both cases, the model represents all of the major processes involved and resolves all of the major features seen in the data. For both datasets, we rescale time to the approximate date of the start of pond formation. There is some difficulty in aligning the start of the simulation data to the start of the field data, as the exact time of the start of pond growth is unknown. The simulation data has an arbitrarily small resolution, whereas the resolutions of the field data is only single days (at best). Therefore the date of the onset of ponding for our model has an uncertainty of a day compared to the field data in panel (a), and possibly more in (b), in which the field data has fewer data points.

[2]http://imb-crrel-dartmouth.org/archived-data/



(a) North site

(b) South site



(c)

Figure 7.7: Simulation of the model (solid) compared against data from 2009 described in Polashenski et al. [2012]. (a) North site. (b) South site. (c) Calculated volume losses from the model runs. Note that the time of the onset of ponding is not exact in (a) and (b), due to the lower temporal resolution of the field data compared to the simulation data.

With the parameters we have chosen, we see that the model replicates the impermeable phase for the north site well. The model reaches almost the same maximum pond coverage, and drains in a similar manner, reaching a minimum at approximately the same time. However, from the minimum, once refill starts, we find that the model significantly underestimates pond fraction. We discussed the likely reasons for this in Section 6.1.3. Due to the method we use to calculate area based on water level, ignoring reshaping of the topography caused by preferential melting in ponds, the area fraction during refilling should be approximately 0.1 more that shown in the model output. This would put the model closer to the data.

The south site provides a sparser dataset for pond fraction. However, we see that the basic shape of the graph is the same, and our model replicates it similarly well, within the limits of the temporal resolution of observations. Both comparisons show that our model captures the correct behaviour for, and is an accurate representation

of pond systems. We admit however, that this is contingent on our tuned h_{crit} being relevant to real world conditions. This parameter has proven difficult to quantify. Disappointingly, we cannot determine how well either the derived time to flood or area fraction near flooding compare to this data.

Polashenski et al. [2012] state that by the date at which it became unsafe to work on the ice due to ice loss, about 25 days after the appearance of ponds, the ice had lost approximately half of its mass. This gives us an estimate with which to test volume loss as calculated by our model. Figure 7.7 (c) shows the fractional volume loss (with reference to initial ice volume V_0) for the two simulations that we have used to compare to the data, which is equivalent to the fractional mass loss. At 25 days, both simulations show volume loss of around a quarter of the initial floe. While the figure of a half from the authors is only an approximation, our model clearly underestimates volume loss, which is understandable given the underestimation of area fraction during refill. In summary, we conclude that our model is able to satisfactorily match the data.

7.5 Fractal dimension of the pond network

7.5.1 Background

In Chapter 2 we defined fractal dimension D of a set F using a box counting method to be

$$D(F) = \lim_{\delta \to 0} \frac{\log N_\delta(F)}{-\log \delta}, \tag{7.19}$$

where N_δ is the number of boxes of diameter δ taken to completely cover F. We also related perimeter p, area a and fractal dimension for a shape in two dimensions by

$$p = ka^{D/2}. \tag{7.20}$$

Having analysed large volumes of aerial photography data of ponds, and measured the fractal dimensions of those ponds, Hohenegger et al. [2012] concluded that the geometry of ponds shows two separate fractal dimension scalings for D, reflecting some ponds with simple geometries and $D \approx 1$, and some with complex geometries and $D \approx 2$. They also proposed that these observed properties of pond systems could serve as a useful benchmark for models and numerical simulations of ponds. This suggestion is in line with one of the major uses of D suggested by Mandelbrot, as a parameter that computational scientists could use in order to create synthetic versions of objects from nature, such as clouds and mountains [Barnsley et al., 1988].

In this section we compare output from our model to values of D measured for ponds, as a further test of realism.

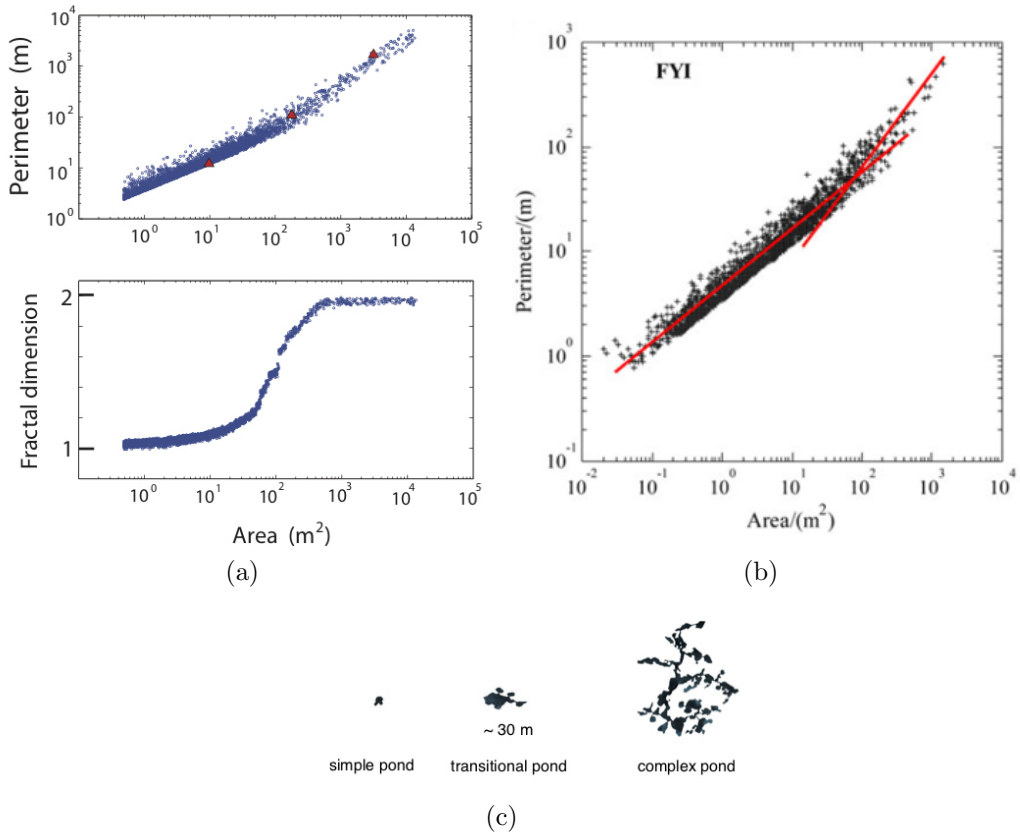


Figure 7.8: (a) Upper panel: scatter-plot of pond areas and perimeters from images taken on August 22nd 2005 on the HOTRAX expedition. Lower panel: Calculated fractal dimension of ponds from the upper panel, plotted against area. Red triangles mark examples of typical ponds from the three types shown in (c). The bottom left triangle is a simple pond, the middle triangle is a transitional pond, and the upper right triangle is a complex pond. Reproduced from Hohenegger et al. [2012]. (b) Pond areas and perimeters from aerial surveys of first year ice in the high Arctic in late summer of 2014, reproduced from Huang et al. [2016]. The red lines are general trends of two partitions of the data showing regions of $D \approx 1$ (lower left) and $D \approx 2$ (upper right) scalings. (c) Black and white images of ponds of the two different regimes for pond scaling - simple, single ponds (left), complex ponds (right) and transitional ponds (centre). Reproduced from Hohenegger et al. [2012].

Several modelling papers have already used fractal dimension as a model benchmark. In most cases, the aim is to capture the behaviour shown in Figure 7.8(a), produced by analysing aerial photographs of ponds. To initially demonstrate this behaviour, Hohenegger et al. [2012] first ordered and binned pond data into ranges of area. They then calculated values of D using pairs of ponds in adjacent bins. For any

pair $(a_{n,i}, p_{n,i})$ and $(a_{n+1,j}, p_{n+1,j})$, where n is the bin number, D was calculated as twice the log-slope between the two points, $2(\log p_{n+1,j} - \log p_{n,i})/(\log a_{n+1,j} - \log a_{n,i})$. Values of D were calculated across all i's and j's in each pair of adjoining bins. Negative slopes were disregarded, and points (a, D) were plotted to produce the lower panel of Figure 7.8 (a), which shows $D \approx 1$ for small ponds of area up to 10 m^2, $D \approx 2$ for ponds with area more than 10^3 m^2, and a transition region between the two. Panel (c), also reproduced from Hohenegger et al. [2012] shows what ponds in each of the regimes look like.

The upper panel of (a) and panel (b) of Figure 7.8 show the signature of this dual scaling behaviour in a simple scatter-plot of $\log p$ against $\log a$ for individual ponds from aerial photographs of real ponds. In (a), ponds from the two regimes, and a pond from the estimated transition between the two are marked by triangles in red. In (b), Huang et al. [2016] overlay general trend-lines in subregions of the area-perimeter scatter-plot in order to demonstrate that same behaviour, with the trend-lines in the upper and lower subregions showing $D \approx 1$ and $D \approx 2$ respectively.

In all attempts to use D in analysing pond systems, the aim is to exploit the relationship, equation (7.20). For this reason, one might be tempted to plot

$$D = 2\left(\frac{\log p - \log k}{\log a}\right), \tag{7.21}$$

where k is an appropriately chosen constant, against area a for synthetic ponds and expect to see a definite separation of the data into the two regimes $D = 1, 2$. However this is not what is seen when we attempt this with model data. The plot is too noisy to show any separation of scales, or be of any use, having an appearance similar to the plot of fractal dimension against area presented by Huang et al. [2016], in which they calculated D using a box counting method (not shown here). The approach was attempted too by Popović et al. [2018] who also found it too noisy and dismissed it. To assess the scaling behaviour of synthetic ponds they had created, Popović et al. [2018] first fitted a function with an elbow to the scatter-plot of the log of perimeter and area. They then differentiated this function to find D as a function of area, and plotted D against area to reproduce the splitting of data into the two regimes with different values of D, with similar results to the lower plot in Figure 7.8 (a).

Horvat et al. [2020] used an alternative approach to describing fractal dimension by a box-counting algorithm that weight-averaged counts of dimension across all box sizes. The authors developed this method in order to rank the complexity of different configurations of synthetic ponds. While this can be useful for comparing synthetic

models against data, it is not as useful a way to interrogate the physics of pond systems as simpler area-perimeter regression methods, so is not pursued here.

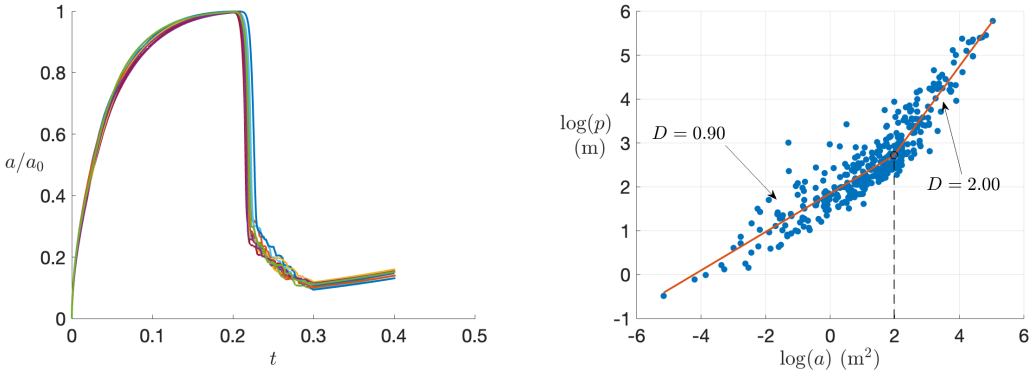
As we have discussed above, attempts to calculate D by differencing methods tend to produce results that are too noisy to be of use, or rely on very convoluted methods of fitting. Instead, we attempt a simpler method, whereby we try to calculate the two fractal dimensions by appropriately partitioning the dataset, and producing trendlines like those shown in Figure 7.8 (b). In doing this we aim to exploit the simplicity of the perimeter-area relation, equation (7.20), by dividing the scatter-plot of $\log p$ against $\log a$ into two subsets, and then fitting a simple regression to the data in each. This should give us two fractal dimensions for our system at different scales (each equal to twice the slope of plot of $\log p$-$\log a$ in each subset). The question now emerges of how to partition the dataset.

7.5.2 Fractal dimension of the modelled ponds

In choosing the model output to compare to observations, we look at the system in late summer, once ponds have drained and spent some time refilling, as this is the time of year at which the photographs used by Huang et al. [2016] and Hohenegger et al. [2012] were taken. In this case we examine the output from the end of a simulation with relatively rough ice, which can be seen in Figure 7.9. The calculated area fraction of ponds at late times is $a/a_0 \sim 0.18$ in line with observations, that $a/a_0 \sim 0.2$ in late summer. From this dataset, we identify each pond in each realisation to generate a set of ponds for which we know the area and perimeter. We then order these ponds by area and partition the dataset into two, again by area, whereby we have a 'lower' partition of small ponds (which we expect may have $D \approx 1$), and an 'upper' partition of large ponds, (which may have $D \approx 2$).

In order to choose where to partition the dataset, we use a multiple splines method - 'Multiple Autoregressive Splines' or MARS analysis, shown in Figure 7.9(b). An explanation of the method and the basis for its use is given in [Friedman, 1991]. The fit gives two knots that are very close together, and we use the lower of the two to partition the dataset, located at 1.97 m^2. The two main splines recreate the scaling behaviour described by Hohenegger et al. [2012], especially within the mean error of the fit, which is 0.135.

We now set out to give a reason for why the change in scaling occurs around the range we observe in Figure 7.9(b). We pose that the system moves to the connected phase, where most catchments are connected and ponds are large, when, on average, each catchment has an area at which is is statistically expected to be just about to



(a) (b)

Figure 7.9: (a) Total area fraction for a numerical experiment with drainage on 12 realisations of a random topography, with $v_w = 2$, $n = 100$, $\Delta h = 0.20$, and $h_{crit} = 0.4$. Perimeter versus area scaling for the same data at $t = 0.4$. Shown is a multiple splines fit to the data using a 'Multiple Autogregressive Splines' (MARS) regression with two main splines separated by two knots (very close together). Fits for D (twice the slope of the regression line) are shown for two main splines. The fit has a mean error of 0.135. Locations of the knots are marked by the dotted red line.

join to another catchment. On inspecting the perimeter-area plots in Figure 7.8 we notice that the change in trend of the cloud of points is most noticeable along the underside of the cloud - along points representing ponds with the smallest perimeter for a given area. To investigate this further, we log-bin our dataset by area, taking 50 equally spaced bins spanning the minimum to the maximum pond area. On Figure 7.10(a) we then plot points of the minimum perimeter in each bin on the scatter of $\log p$ and $\log a$ in blue, with the area for each point positioned at the centre of the bin. We also plot corresponding points for the mean perimeter in red.

At the right end of the plot the scatter deviates from the clear trend in the main part of the plot, around 1, above the $D = 1$ trend line. The points on the left have a trend close to 1, whereas those to the right side have a trend closer to 2, in line with the observations of [Huang et al., 2016] and the analysis of [Hohenegger et al., 2012]. The physical reason for the change in trend is the transition from isolated to joined catchments. Figure 7.11 shows a mosaic of pond geometries from four realisations of the dataset from which we see that isolated ponds have simple geometries, and relatively small perimeters for a given area. Joined ponds, especially across low-lying saddles, on the other hand, generally have long, thin regions that lead to large perimeters, but do not contribute substantially to the area.

We now provide an argument for choosing the area at which to separate the dataset

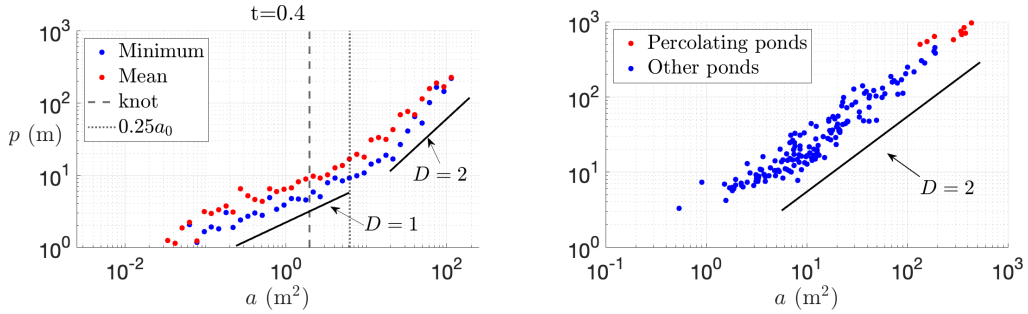


Figure 7.10: Perimeter-area scaling for the data shown in Figure 7.9. Ponds areas and perimeters are made dimensional using the scale length $\tilde{x} = 5$ m. (a) Binned minima and mean perimeters at the end of the simulation. Areas are log binned, using 50 bins. The pond with the minimum perimeter in each bin is plotted in blue, and the mean perimeter in each bin in red. The dotted line marks the expected area at which the transition occurs, $0.25\bar{a}_0$. The dashed line shows the location of the lower knot of the MARS analysis shown in Figure 7.9. (b) Individual ponds at the time of percolation during the initial period before drainage. The percolating clusters from each realisation are shown in red. Other ponds are shown in blue.

to calculate the two trend lines shown in Figure 7.9(b). To partition the dataset, we assume that the scaling changes at an area just before ponds begin to join together. If we pick a saddle at random, under what conditions do we expect it to form a join with ponds i and j on either side of it? We require that $H_i = h_{ij}$ and $H_j = h_{ij}$. During refilling, $H_i = H_j$, due to the relaxation towards sea level. The expected initial height of the saddle $\mathbb{E}(h_{ij}) = \bar{h}$, the expected initial height of a random peak $\mathbb{E}(h_p) = \bar{h} + \Delta h/2$ and the expected height of a random pond centre $\mathbb{E}(h_c) = \bar{h} - \Delta h/2$ (c.f. Section 5.2). We have chosen our area scale such that $\mathbb{E}(a_0) = \bar{a} = 1$. Given that we are using the initial hypsometry to calculate area, we can use the averaged square pyramid expression for area from Section 5.3 up to the point of joining. Replacing each height by its expected value, we estimate the area a_c at which the ponds tend to connect,

$$\begin{aligned} a \leq \bar{a}_0 \left(\frac{H - h_c}{h_p - h_c} \right)^2 &\approx a_0 \left(\frac{\mathbb{E}(h_{ij}) - \mathbb{E}(h_c))}{\mathbb{E}(h_p) - \mathbb{E}(h_c)} \right)^2 \\ = \bar{a}_0 \left(\frac{\bar{h} - (\bar{h} - \Delta h)}{(\bar{h} + \Delta h/2) - (\bar{h} - \Delta h/2)} \right)^2 &= \frac{\bar{a}_0}{4}. \end{aligned} \tag{7.22}$$

Using a scale length $\tilde{x} = 5$ m, which gives dimensional average catchment size, $\bar{a}_0 = 25$ m^2, we estimate that the transition to highly connected ponds $0.25\bar{a}_0$ begins

Figure 7.11: Mosaic of the simulated pond geometries at time $t = 0.4$ for four realisations of the simulation used to generate Figure 7.10.

at a pond area of 6.25 m^2. For a single catchment, this is in line with the order of magnitude at which joins begin to occur as reported in Hohenegger et al. [2012] at around 10 m^2, and with the location of the lower knot calculated using the spline fit in Figure 7.9(b). We have not used the observed fractal dimension of pond systems to create synthetic topographies, so the fact that it reproduces the transition in fractal scaling provides independent evidence that our mechanistic model usefully captures important aspects of the physics of the system.

In contrast to static models that focus more closely on the statistics of ice, snow

and pond geometries, [Popović et al., 2018], our model is capable of exploring how and why the scaling and roughness measure D of ponds changes during pond evolution. An exploration of how the perimeter area scaling for ponds changes throughout the melt season can be found in appendix C. As an example, Figure 7.10(b) shows the snapshot of pond areas and perimeters at the time of percolation during the initial pre-drainage filling phase. At this time, most ponds are joined and we see that they all display the more complex $D \approx 2$ scaling.

7.6 Topology of the pond network - Percolation processes

7.6.1 Motivation and definitions

Due to the ways in which the edges in our model change during pond evolution, the system goes through a bond percolation process up to three times. This is seen in Section 6.3.1, where the network becomes densely connected as ponds grow in the initial filling period, becomes disconnected during drainage, and starts to become connected again during refilling. We will now use a number of tools from percolation theory to explore these processes in our model in more detail. We restrict our discussion to bond percolation, for clarity. As discussed in Section 2.3.3, statistical mechanics and network science make use of percolation theory but take slightly different approaches. In statistical mechanics, the theory focuses on 'universality', based on the exponents to power law scalings and on behaviour very close to the phase transition. This approach was taken by Popović et al. [2020b] and Ma et al. [2019] in the context of melt-pond models. In contrast, network science uses percolation theory to investigate the resilience of networks to removal of edges, and how the function of a network changes with changes in connectivity, by looking at the behaviour of the largest component.

While percolation theory in statistical mechanics often assumes that the addition or removal of edges occur uniformly at random, and that the occupation probability of any two edges are independent, in network models these assumptions do not necessarily hold, and a greater variety of models are used. The state of different edges are not independent of one another in our model, and the probability of any given edge being active is not uniform. In this section we investigate the percolation processes that occur in our model by considering the system as a temporal network rather than a lattice system with a phase transition. We discuss the network function for our

pond system and how the percolation processes affect this function. We also discuss the consequences of lack of independence and uniformity in the percolation processes.

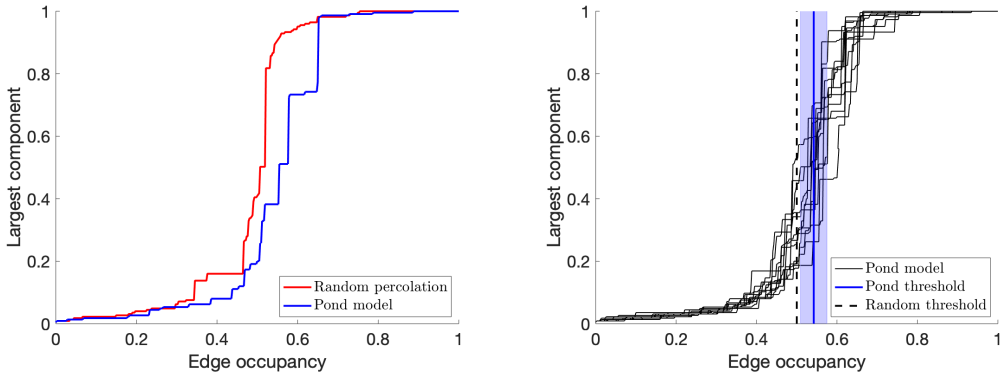
The 'function' of a network typically describes the ability of the network to transport something, be it electricity in a power grid, or information in a social network, across the entire network. In percolation theory, we explore how network function changes as we add or remove edges. For a power grid, we might wonder how many transformers, or generation stations/installations can fail before a blackout occurs. For an epidemic, we would explore how many people we have to vaccinate in order to control outbreaks. Importantly, a percolation process is highly dependent on network structure, especially $\langle k \rangle$, the mean degree of the network (the average number of edges of a node).

The 'function' of the pond network is the ability to evenly distribute water on the surface. Specifically we might define a pond network to be 'functional' if we can draw a line through a system of ponds (join edges) and channels (the overflow edges) from one side of the floe to the opposite side. According to this definition, if we start from the dry state in spring, and allow edges to sequentially become active, the function of the network improves, as the largest component in the graph grows. An example of this process is shown in Figure 7.12, which displays a sharp jump in size of the largest component, as discussed in Section 2.3.3, suggesting a percolation transition.

Our pond model has two different types of active edges for the surface of the floe, overflows and joins (ignoring the differences in overflow edges due to directedness). The rules derived in Section 4 for when these edges change state remain the same at all times, however in Chapter 6.3.1 we noted that during the different stages of pond development, very different behaviour of the sets of active edges are seen. From this we argue, and will now demonstrate, that the processes of filling, draining and refilling present different mechanisms that lead to the network moving between connected and disconnected states, and thereby represent three different percolation processes.

7.6.2 The filling process

In discussing several percolation processes, we require a null model to make comparisons against. For this we choose random percolation of edges on the floe, similar to the checker board example described in Section 2.3.3. The null models of bond percolation on the square lattice and the Voronoi tessellation (the two types of catchment geometry we have used in our model) are well understood, and have percolation transitions at edge fractions of 0.5 and 0.33, respectively.



(a) (b)

Figure 7.12: (a) Percolation during the filling stage of the pond model compared with random percolation on a square lattice floe of 225 catchments. In each case only one realisation is shown. Here, $v_w = 2$. Plotted for each are the fractional size (fraction of total number of nodes) of the largest component against the fractional number of active edges (number of join and overflow edges at each event, divided by the number of potential edges in the graph of catchments) as active edges are added. (b) Percolation of 12 realisations of the full pond model (joins and overflows) during filling on a square lattice of 225 nodes, $v_w = 2$. The vertical dashed line marks the edge fraction at which the random model percolates. The solid blue line is the calculated edge fraction at which the pond model percolates, averaged over the 12 realisations. The shaded region denotes one standard deviation.

For simplicity, in discussing percolation, we look at a square lattice topography. This choice benefits from the fact that the bond percolation threshold for a square lattice is known to be exactly 0.5, and the degree of every internal catchment is 4, providing the simplest system to examine. We will later consider percolation on the Voronoi tessellation. To visualise the percolation process, we start from a state where all edges are inactive, and consider successive edge activations until all potential edges are active. During this process we track the size of the largest component in the system and plot it (as a function of system size) against edge occupancy, as was done to generate Figure 2.3.

Figure 7.12(a) shows the plot of the percolation process for the pre-drainage phase of a simulation of 225 nodes on a square lattice. The plot also shows the equivalent output from a simulation of the null model, adding edges on a square lattice of the same size in a uniform, random order (every inactive edge has the same probability of being the next edge chosen). We see a qualitative difference between the two processes. The random process percolates earlier, with the last large jump around

the theoretical transition of 0.5 edge occupancy, and then a smoothing of changes in the size of the largest component. In contrast, the filling model shows delayed percolation, with stages of large jumps.

To identify when percolation happens for these trajectories, we define the percolation transition for our model as the event when the largest component first spans from one side to an opposite side of the floe. We express this in terms of the number of active edges (both join and overflow edges) as a fraction of the total potential surface edges (not related to porous drainage or drainage over the side) in the system at the time of the event. For simplicity, we ignore the direction of overflow edges. For accuracy (the difference shown in Figure 7.12(a) could be by chance), we average the calculated percolation thresholds for a number of simulations. We also define the percolation area as the area fraction of ponds at the time of percolation. Again, each time we quote this value we use the average of several ensemble members of a simulation.

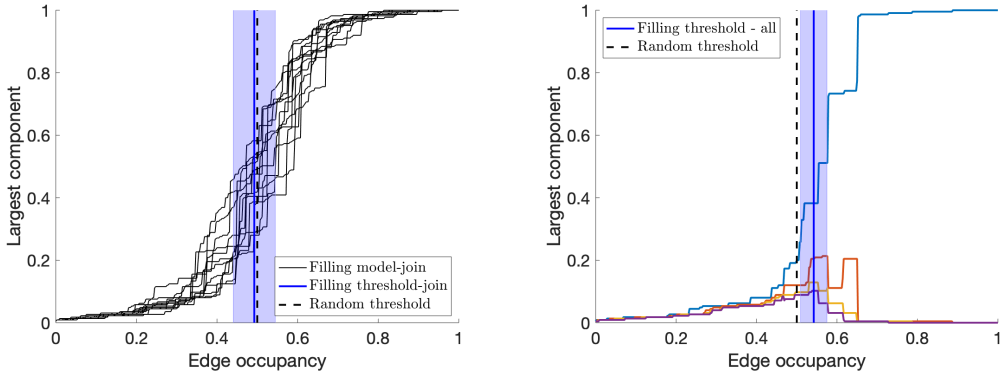
In Figure 7.12(b) we show 12 realisations of percolation of the filling model run on a set of 12 randomly generated square lattice topographies of, again, 225 nodes. This resembles anomalous percolation, discussed in Section 2.3.4, whereby the choice of rules for the addition of edges can delay percolation compared to a random uniform process [D'Souza and Nagler, 2015].

7.6.3 The role of overflow edges in network function

What might account for the anomalous behaviour of the pond model? What acts to suppress the growth of the largest component? Given that in our model we have distinguished between different types of active edges, we can also probe whether differences exist in percolation on the full graph of active edges which includes joins and overflows (representing channels), and percolation on the graph of ponds alone (just joins). By discriminating between the graph of all active edges, and the graph of join edges alone, we ask whether channels are significant to the connectivity and function of systems of ponds, and if so, how significant. Figure 7.13(a) shows the equivalent process as in Figure 7.12(b), for the same set of simulations, but where we consider the graph of join edges only, and with a corresponding calculated average percolation transition marked.

Comparing Figure 7.13(a) with 7.12(b), the trajectories of the largest component size are smoother for the join model, with a larger variation. These trajectories are qualitatively similar to random models (such as percolation on the configuration

model, described in Newman [2018]), and have a much lower percolation transition, almost exactly at the random transition.



(a) (b)

Figure 7.13: (a) Join percolation (counting only join edges) during filling, on a square lattice of 225 nodes, $v_w = 2$. (b) The largest four components of the graph at any one time during the percolation process for one simulation. Also marked are the percolation threshold for the random model (dashed) and the percolation threshold for the filling model in blue (counting both joins and overflows as active), calculated by averaging over the 12 realisations used in (a). The shaded area denotes one standard deviation.

From this we infer that the anomalous behaviour is caused by the existence of channels, the overflow edges, which cause two opposing behaviours. In terms of network function, channels spread melt-water from regions that would flood early to ones that are lower and would flood later, helping to distribute water, which we would expect to hasten percolation. However, the degree rule on overflow edges - whereby each catchment can only have one outward overflow edge - limits the number of edges in the system while these are active, in a similar way to cluster size limiting processes seen in other models that display anomalous behaviour [D'Souza and Nagler, 2015]. This degree rule breaks the independence of edges assumed in the random model. The rule can also be thought of as a less efficient way to cause percolation in the network while adding active edges to the system than a purely random process.

This rule means that taking one cluster of neighbouring nodes in the network, join edges can be added between the nodes in the cluster until all potential edges in the cluster are active. However, the cluster can only have one overflow edge that would join that cluster to another. This puts a constraint on how large clusters can become, even if they have a large number of active edges (larger edge occupancy), at least until the average water level rises sufficiently.

The process of the initial suppression and later explosion of the size of the largest component can be seen more clearly in 7.13(b), which shows component size trajectories of the four largest components for one of the simulations that made up panel (a), but using the full graph of active edges. Before percolation, the largest components all grow at similar rates until the percolation threshold, when the components of the graph are so large that further addition of active edges must eventually join some of these components together, creating the giant component that then percolates the system, suddenly (shown in blue). We find that this anomalous behaviour is seen across values of Δh, v_w, lattice type, system size, and number of realisations used.

7.6.4 Percolation during drainage

Does something similar occur in reverse once drainage begins? How does the structure and connectivity of the network break down as we remove edges during drainage (reverse percolation)? To answer these questions we choose a simulation on a square lattice with drainage that both floods and empties completely (very low Δh), and track changes in the network backwards through the drainage stage to develop a plot of the largest component as a function of edge occupancy. We also calculate a percolation threshold for the drainage process by tracking when the spanning component appears, going backwards in time from the end to the beginning of drainage. Figure 7.14(a) presents the trajectories of this process, with the calculated percolation threshold for drainage shown in red (the solid line is the average across the 8 ensemble members of the simulation, and the shaded area represents one standard deviation). The random threshold is marked by the dashed line. The percolation threshold from the filling stage of the same simulation is shown in blue (again, an average, with standard deviation in shading). Panel (b) shows the same, but with overflow edges excluded.

The results show that drainage percolation, in this case the loss of the continuous path from one side of the floe to another, happens at a lower edge occupancy fraction than its emergence during filling. This demonstrates the kind of hysteresis often seen in models that display phase transitions [Brokate and Sprekels, 1996] or anomalous percolation [Bastas et al., 2011], and bolsters the argument that overflow edges, which reappear during the drainage process, work to spread water widely across the floe. This can also be seen in Figure 6.5(d), the time series of edge type counts for a draining simulation.

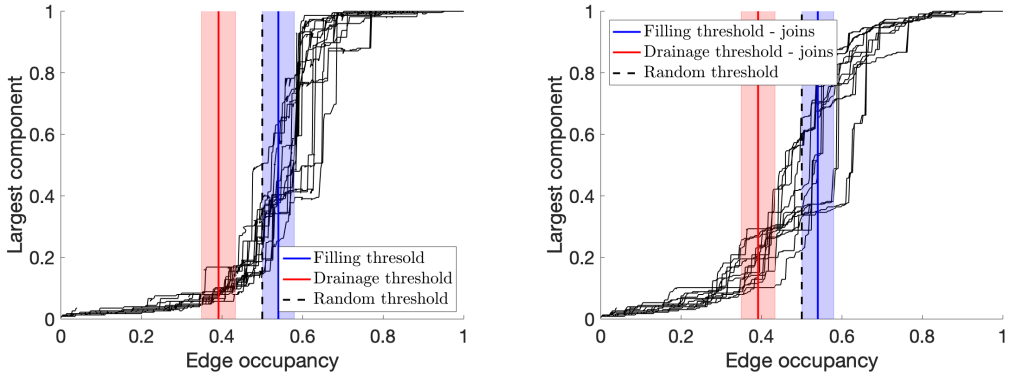


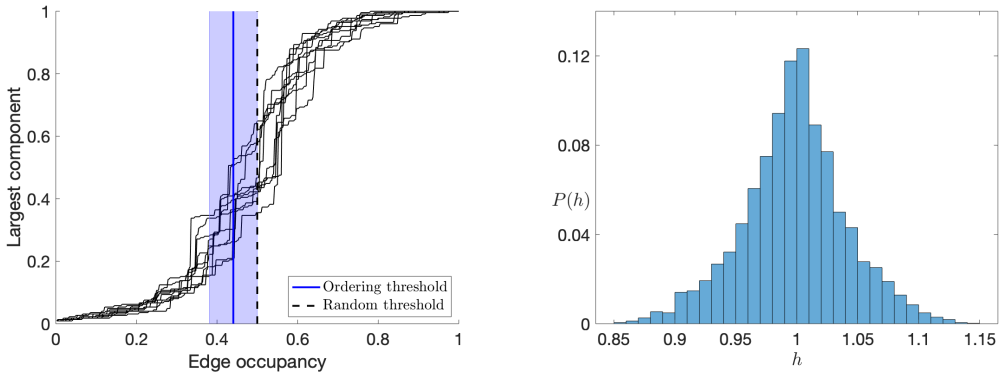
Figure 7.14: (a) Full graph percolation and (b) join percolation (ponds minus channels/overflows) for drainage processes on a square lattice of 225 nodes, $v_w = 2$, $\Delta h = 0.1$, $h_{crit} = 0.7$. The percolation thresholds for filling and draining are shown in each case in blue and red respectively. Shaded areas show one standard deviation.

7.6.5 Refilling, and the saddle ordering model

Revisiting Section 6.3.1 and in particular Figure 6.5(d), we concluded that the refilling stage displayed a different form of behaviour for the active edges, whereby the network regains connectivity, but without the formation of overflow edges, as the water level in all catchments is the same throughout. This is analogous to percolation by an ordering of saddle height. We will call this 'saddle-ordering percolation'. Trajectories and a calculated percolation threshold for this process can be seen in Figure 7.15. To generate these trajectories we took a set of random topographies in turn (using 225 nodes and $\Delta h = 0.15$, however, the behaviour is independent of this parameter). We then ordered the saddles in the topography by height from lowest to highest, and made edges active in turn in this order.

As seen in Chapter 6, during refilling only a small fraction of edges become active again, so we cannot see the whole percolation process during refilling in the data from simulations. The water level never gets high enough during this stage to cover all saddles, which is why we model this process using this saddle-ordering method, rather than using data from the main simulations. In reality, we do not expect the system to reach the percolation threshold by this process in late summer; however, it is worth exploring, as the model for percolation during this period of our model is similar to that which would be seen for the level set model for pond growth of Bowen et al. [2018].

The results show a lower threshold than for the filling process, and lower than for random percolation. As discussed in Section 5.2, the distribution of saddle heights is not uniform, evidence of which is shown in Figure 7.15(b). This may account for the deviation of the ordering threshold from the random percolation, which assumes a uniformly random process.



(a) (b)

Figure 7.15: Saddle ordering percolation (the level set method) on a square lattice of 225 nodes, $v_w = 2$. (c) The distribution of saddle heights for a set of 12 topographies for which $\Delta h = 0.15$.

7.6.6 Differences due to network structure

So far we have explored the percolation behaviour of the model on a square lattice. We now briefly explore a different network. As mentioned in the introduction, the percolation statistics of a network depend on mean degree, with consequences for the type of network we choose. The square lattice network has mean degree of 4. The networks on the Voronoi tessellation of randomly placed catchment centres are calculated to have nodes with mean degree of 5.75 (accounting for boundary edges). Figure 7.16 shows the percolation process plots for the more disordered floes using Voronoi tessellations. The calculated percolation thresholds shift towards 0.33, in line with the random threshold, and higher average degree, while conserving the relative behaviour of the percolation processes. The envelopes of trajectories are also shifted to a lower edge occupancy.

In line with other classical percolation models, percolation in our filling model depends strongly on the structure of our network of potential edges. We even see that if

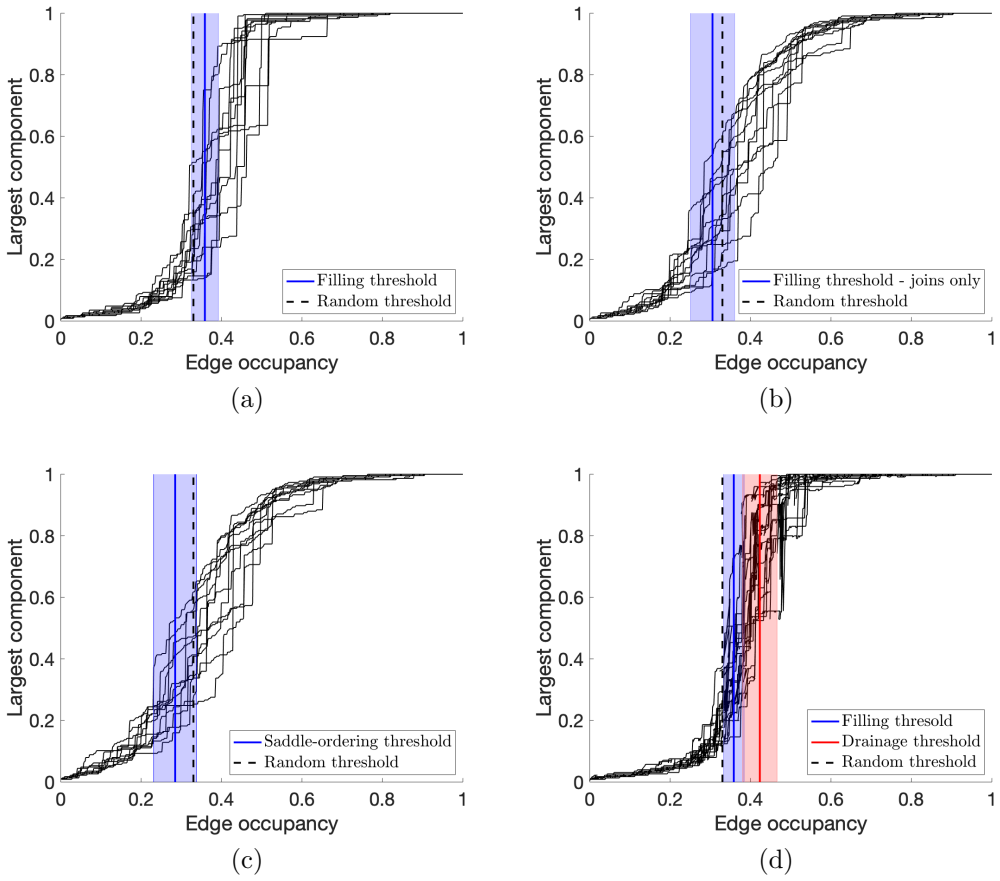
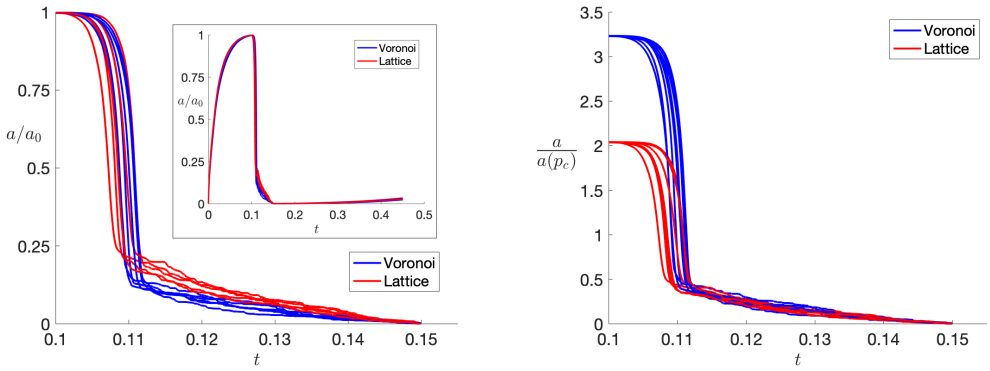


Figure 7.16: Percolation of (a) the full pond model during filling (joins and overflows) (b) just joined catchments during filling, (c) saddles by height and (d) during drainage on a random lattice of 225 nodes, $v_w = 2$. For the simulations used to generate (d), $\Delta h = 0.1$, $h_{crit} = 0.7$.

we compare Figure 7.16(d) to 7.14(a), increases in connectivity change the relationship between the percolation threshold during filling and the percolation threshold during emptying, reversing the hysteresis. The more saddles on average each catchments has, the faster percolation will occur. This difference in network topology has consequences during drainage for the speed of the initial, smooth part of drainage when the graph is mostly intact, before the secondary stage where the graph starts to break up and many different regions either drain or overflow into regions that drain (discussed in Section 6.2.3). A more densely connected random topography holds a high area fraction for longer once drainage begins. The break up of the main pond that floods the floe occurs at a lower area fraction for the random topography, as can be seen in Figure 7.17. The topology also controls the maximum area fraction

that can be reached if water is free to flow over the sides of the floe, with a lower maximum area fraction reached for the more connected random topography. However, these differences due to connectivity diminish if we rescale the area fraction during the drainage process by the area fraction at percolation, $a(p_c)/a_0$, as is shown in Figure 7.17(b).

The collapse of the pond fraction trajectories onto one curve indicates a scaling law for drainage, as was briefly discussed in Section 2.3.3, and explored by Popović et al. [2020b] who explored percolation during the drainage phase using a lattice model with simpler water flux rules.



(a) (b)

Figure 7.17: (a) Area fractions during drainage for Voronoi tessellation and square lattice topographies for simulations of $\Delta h = 0.1$ and $h_{crit} = 0.7$. The inset is the same data but for the full period of the simulations. (b) Area fractions during drainage rescaled by $a(p_c)$, the area at the percolation threshold.

We conclude with a comment on the network size and parameters used for this investigation. Increasing the number of realisations and the number of nodes improved convergence of the statistics (as is well known in the literature) and strengthened the case for anomalous percolation. However, a system size of 225 was large enough to generate accurate statistics for percolation during the filling or refilling phases. No differences were seen in the percolation thresholds due to variation of either Δh or v_w during the filling or refilling phases.

7.7 Summary

In this chapter, we have compared our model to observational data, and found that it adequately captures the geometry (Section 7.5) and the behaviour of ponds (Section

7.4). In Section 7.2 we carried out a set of numerical simulations to explore how model behaviour varies with two key parameters - the roughness Δh and critical ice thickness for pond drainage h_{crit}. From the results of these experiments, we derived time scales for the different stages of pond evolution, and approximations for the area fraction during these stages. The derived time scales depend on the melt rates, the average ice thickness, the surface roughness, and the critical thickness for drainage. This indicates that accurately measuring these properties and parameters, through field experiments or remote sensing can help us better understand and predict when each stage of pond development will occur in different parts of the Arctic in a given year. Estimates of these time scales can prove valuable to oceanographers and climate scientists. At the time of drainage, large quantities of freshwater flush into the upper layer of the ocean at this time, which can have a profound effect on the behaviour of the mixed layer, due to the accompanying changes in salinity. The timing of the breaching of ponds through the base can have an effect on the break-up of floes. Our estimates of time scales for pond behaviour, and of the associated pond coverage throughout the melt season may prove a useful way to parametrize the effects of ponds on the albedo of the ice. We were also able to derive a simple parametrization for the volume of melt produced, and therefore the amount of ice loss.

In Section 7.3 we investigated why in some cases floes might not produce ponds even if surface melting occurs. We found that for flat ice that drains early, ponds that form in early summer can drain completely, leaving the floe absent of ponds in mid to late summer. In Section 7.5, we found that the change in the perimeter-area scaling of pond geometries occurs near the area at which the average catchment forms a join edge with a neighbour (the expectation value of area for the formation of join edges). The change is scaling was not found to depend on the melt rates, or the roughness of the ice, or the drainage dynamics, but on the statistics of the heights of saddles, and on the pond fraction.

Finally, in Section 7.6, we explored the percolation processes in our model, and tied these to pond evolution in Section 7.6. We found that during flooding and drainage, the inclusion of overflowing fluxes in our model act to distribute water across the floe, but delay percolation as a function of edge occupancy, as they restrict the configurations that can occur on the graph of edges. This leads to higher possible pond fractions during flooding if there are flaws in or sides to a floe, and hastens the processes of drainage once it begins. We also found that the percolation process that occurs during refilling is not the same as during flooding or drainage. In a simple square lattice geometry, percolation during refilling occurs at a lower edge

occupancy fraction than the filling threshold, or random percolation, but at a higher fraction than the drainage threshold. However, the difference between the thresholds for the refilling and drainage process can reverse in the more ramified random lattice geometry.

Chapter 8

Conclusions

In this chapter we present a summary and synthesis of the thesis, and discuss possible future work. We also discuss how the conclusions of our model may help us to predict how Arctic ice will change in the warming environment of the coming decades, and what changes we might expect to see in the extent and behaviour of melt ponds in the future. Summaries of individual chapters are also given at the ends of Chapters 3 to 7.

8.1 Summary, and future work

In Chapter 3, we set out a model for a single pond, on impermeable ice, which formed the basis of node behaviour later in the network model. In the model, the melt rates differ between bare and pond-covered ice, and this causes a discontinuity in the gradient of the ice topography at the boundary of the pond. According to the model, the slope of the pond floor steepens when the growth rate of the pond slows, which can lead to the formation of 'cliffs' at the pond edge if the expansion of the pond stops. This can go some way to explaining the particular pond geometries seen in observations [Fetterer and Untersteiner, 1998], where ponds have steep sides, rather than beach-like shorelines. An asymptotic expansion of the equation for pond growth at early time was derived, to deal with the singular nature of that equation, which can be used in numerical simulations of the network model. A steady state solution to the model for a single catchment was found, as well as analytic solutions for some particular one-dimensional geometries.

In Chapter 4, we derived equations for the different types of water fluxes that can occur between neighbouring catchments, depending on the height of the lowest point of ice along the boundary between catchments. In developing these constraints, it was found that if water from one catchment overflows into another, this stops the

growth of the pond in the overflowing catchment. The stalling of area growth also introduces a constraint to the network topology, whereby only one overflow edge can exist out of any pond (or connected component of the graph). This chapter also focussed on writing the fluxes as a system of equations that depend on the network structure and activity on the nodes (water levels and pond areas). It was found that this system is indeterminate for a pond network that contains loops. However, it is possible to locate and remove edges that cause loops from the pond network in a numerical implementation of the model, allowing solutions of the system of fluxes to be found without changing the physics in the model.

With the model for impermeable ice developed, in Chapter 5 we outlined the method by which we create synthetic topographies, in such a manner that the network structure of the model can be easily imposed on them. These synthetic topographies were compared to field observations, and found to closely (though not perfectly) fit the data. The uniform distributions from which the maxima and minima were chosen, and from which the saddle points were developed, were intended to be simple, so that they captured the variation in the topography, but also so that the output of the model could be analysed easily, and so that relationships could be identified between the behaviour of pond systems and the underlying topography. The linear faces chosen to represent the topography were chosen to ensure computational economy, so that calculating the pond areas was cheap and straightforward. Future iterations of the model could investigate more sophisticated ways of building the topography, such as choosing point heights from a distribution fitted to field data, as was done by Popović et al. [2020a].

In Chapter 5, we also presented several example numerical simulations to demonstrate the behaviour of the model of connected ponds on impermeable ice. We tested the response of the system to drainage over the sides of the floe, and found that this process is a strong control on the maximum pond fraction that can be achieved, and that this maximum area is around or below the area at percolation.

In Chapter 6, the model was modified to incorporate porous drainage. Motivated by previous work on modelling drainage through porous sea ice [Eicken, 2002, Polashenski et al., 2017], a simple functional form of the drainage flux was chosen based on Darcy flow in a porous medium, driven by the hydrostatic pressure head of fresh water in ponds. The condition for the ice to become porous was represented by a critical ice thickness h_{crit}, or equivalently, a corresponding time at which the ice heats up enough to become porous.

To improve the model, it could be instructive to actually model the temperature (and associated permeability) of the ice. For example, the slab model for ponded ice presented by Taylor and Feltham [2004] could be used to calculate the bulk temperature in the ice, dependent on the solar forcing and thickness of the ice. The model could be expanded to solve for the minimum temperature in the ice at each time step and in each catchment, and drainage would begin once the minimum temperature reaches a critical temperature. Experiments in the field by Polashenski et al. [2017] suggest that an appropriate critical temperature is -0.9 C. Alternately, the drainage model could be used for all times in a similar fashion to that used by Skyllingstad et al. [2015], with the drainage parameter $\hat{K}$ written as a function of temperature. A low value of permeability could be used for cold ice at early time, and as the ice warms, the permeability increases, parametrised in a similar way as in [Scott and Feltham, 2010], to account for the effect of temperature and salinity on porosity in line with mushy layer theory. Modifying the model in this way could also allow for the tracking of salinity dependent changes to brine density in the ice as drainage begins, which would improve the calculation of the hydrostatic head, and the estimate of how and when the permeability changes.

Improvements could also be made to the way in which area is calculated, such that changes to the ice topography within ponds due to enhanced melting are properly accounted for. However, given that we do not wish to build a version of our model where the whole floe is discretized and changes to a full spatial distribution of ice heights are tracked (as was done in some one-dimensional examples), this would still need to involve some approximations. A discretization of the catchment in polygons enclosed by the contours of each saddle, could be used to evolve the approximate hypsometry in line with cliffs that form during overflow periods. This would entail tracking two heights for each saddle point: the saddle itself, and the height of the bottom of any cliff that would form due to the saddle overflowing (which would move downwards at a rate $-v_w$ during overflow).

The inclusion of porous drainage completed the model for pond evolution from the onset of ponding to the time at which the first pond breaches the base of the floe. With this addition, the model replicated the observed pattern of pond behaviour: initial growth for several days, followed by rapid drainage over the course of as little as a single day where pond water levels drop to near sea level and pond extent shrinks, and finally a long period during which pond surfaces are fixed just above sea level, and during which small increases in pond area are seen. The timing and duration of these stages, as well as the pond extents reached during each stage were found to be

heavily influenced by the parameters. A wide variety of different behaviours, within the distinct patterns of behaviour for the stages, was seen in the model output. The wide variety of possible behaviours seen in the sensitivity analysis indicates that a wide variety of pond evolution is possible now, and further, new behaviour might be possible in the future, if the state of the ice pack or the prevailing meteorological conditions change.

We found that the most important parameters in the model are the surface roughness Δh and the condition for the change in permeability, i.e. the critical thickness for drainage h_{crit}, both of which accounted for most of the variation in pond behaviour. The drainage parameter $\hat{K}$ was only found to affect behaviour of pond extent up to a value of 10^3. This value is consistent with observed values for the permeability of ice in late summer, around 10^{-10} m^2 [Polashenski et al., 2017]. Changes to the melt rates were found to shift the timing of events, but not greatly affect the behaviour. Numerical simulations were also performed with two forms of drainage. As in the simulations with impermeable ice, drainage over the sides of the floe was found to restrict the maximum pond fraction reached. The size of this maximum fraction was seen to vary with the number of catchments along the exterior of the floe out of which surface water could drain. Drainage over the sides was seen to play no further part once porous drainage began.

Simulations of the full model were analysed in detail in Chapter 7. We found that the model recreates the change in the scaling relations of perimeter and area of ponds discussed by Hohenegger et al. [2012]. Good agreement was found between observations and model output for the two observed fractal dimensions for pond systems, $D = 1$ for a disconnected phase of simple pond geometries and $D = 2$ for a connected phase of sprawling, ramified geometries. Good agreement between observations and model output was also found for the area at which the scaling changes. According to our model, the change in scaling of D for an individual pond occurs at the pond area at which the average catchment is statistically expected form a join edge, $\bar{a}_0/4$. This is the state of the system immediately preceding the percolation threshold. When dimensionalized with a representative scale for the lateral extent of catchments, this transition area was found to be similar to the observed area at which 'transitional' ponds with more ramified geometries begin to form. Further, the transition area was found to be independent of the melt rates and the roughness of the ice surface.

Focussing on Δh and h_{crit}, we explored the dependence of the model behaviour, particularly pond extent and melt volume, on changes to the parameters. Based on

this analysis, we derived time scales for the important processes in pond development, particularly flooding of the floe, the onset of drainage, the end of drainage, and the time at which the deepest pond would melt through the floe. For our model, these time scales are dominated by the melt rates, surface roughness and the onset of drainage. If we take scales $\tilde{v} \sim 1$ cm/day, $\tilde{h} \sim 1$ m, and a floe that has $\bar{h} = 1$, $\Delta h = 0.2$ and $h_{crit} = 0.7$, then according to the derived time scales, in the absence of drainage, flooding would occur at day 20 after the onset of ponding. Drainage would occur before this, at day 5, and would end on day 15. Pond floors would begin to breach the base on day 40. Using a mean-field approach, we were also able to derive the asymptotic behaviour for pond extent at early time and times near when floes flood, and for melt volume production for all times.

Using the model, we investigated the conditions under which we would expect to see catchments draining completely, and found that this occurs for very flat ice that drains early, and for which the ice becomes porous while ponds are still shallow. We also compared the model to data from the fieldwork of Polashenski et al. [2012], and found that, given a judicious choice of the critical thickness for drainage, good agreement between the data and the behaviour of our model could be achieved. Given the good agreement with observations, the model may prove useful in deriving improved parametrizations of pond extent and ice volume loss due to ponding, for use in climate models. In future development of the model it would be useful to make use of other datasets, such as the SODA dataset [Lee et al., 2016]. Given that satellite data and data from buoys deployed on ice floes covering a greater range of conditions can be obtained, this data should be used in testing any future iterations of the model, so as to compare the model to a wider range of ice types and meteorological conditions.

Due to the different physical processes that dominate filling, drainage, and refilling, we found that there are three different percolation transitions in the process of pond evolution. The percolation threshold for the filling process is delayed compared to random percolation, with percolation occurring at a higher edge density than expected, and represents a form of anomalous percolation. The anomalous percolation can be explained by the constraint on edge formation introduced by the presence of overflow edges. Percolation during refilling occurs at a lower edge density than random percolation (i.e. fewer edges need to be added). The reverse percolation during drainage (network breakdown) also occurs at a lower edge density than for random percolation (i.e. more edges need to be removed) for the less densely connected square lattice, and at a higher edge density for the more densely connected random lattice. The percolation processes in our model led to small differences in how pond extent

changes throughout the season amongst floes with different network topologies, especially during filling and draining. For example, the early stage of drainage can occur much faster on a floe in which catchments are disordered and more densely connected to one another, such as floes with randomly placed catchment centres, compared to a very ordered topography, like the square lattice. These processes bear further investigation, as they may represent unexplored variations of edge rules for network formation.

8.2 Possible improvements and comparisons with other work

The model we have developed is of intermediate complexity, falling in the middle of a hierarchy of models for ponds. It is more sophisticated and includes more physics than the models described by Bowen et al. [2018], Ma et al. [2019], and Popović et al. [2018], and represents both the surface topography, pond geometry and physics in a different way than Popović et al. [2020a] and Popović et al. [2020b]. We also argue that our model accounts for pond behaviour over longer time periods than those last two models, each of which represent either the period before or after drainage, but not both. It is less sophisticated and includes less physics than the models presented by Scott and Feltham [2010] and Skyllingstad et al. [2015], but still resolves all of the major behaviour of the summertime evolution of ponds. In making comparisons between our model and those mentioned above, there are both points of agreement and disagreement. The model agrees with Scott and Feltham [2010] that the initial surface roughness of the underlying ice is a strong control on pond extent and geometry later in the summer. The model differs from Skyllingstad et al. [2015] in that it neglects snow cover and considers ice that is not flat. Our model disagrees with the model of Skyllingstad et al. [2015], in that Skyllingstad et al. [2015] found that the exact permeability of the ice has a very strong influence on pond coverage post-drainage. Our model, however, found that above a certain value, the permeability of ice is much less relevant than the depth of ponds by the time drainage starts. It should be noted that this does depend on the assumption that we have made whereby the permeability of the ice changes from negligible to substantial almost instantaneously once pond floors reach a critical thickness above the base of the floe. This differs from the assumptions made by both Scott and Feltham [2010] and Skyllingstad et al. [2015].

Our model agrees with [Popović et al., 2020b], that the area at the percolation threshold, $a(p_c)$, is a plausible upper bound for pond fraction post-drainage. However, our model found that there is a different percolation threshold for each different phase of filling, draining and refilling, complicating the picture somewhat, but opening up further questions about what percolation theory can tell us about the growth and behaviour of pond systems. Although the analysis in Chapter 7 took a different approach to understanding what factors control whether ponds remain or do not remain after drainage than the probabilistic model of Popović et al. [2020a], the results are qualitatively similar, concluding that the strongest control is the initial surface roughness of the ice. This gives added confidence that this result may be robust and independent of model architecture. There is also some agreement between the behaviour of our model during the refilling period and the analysis presented by Popović and Abbot [2017] which investigated pond growth on permeable ice. Both models found that, assuming high permeability, ponds recover more area after drainage on rougher ice.

The modular nature of our model also allowed us to investigate many different aspects of the physics involved in pond growth. Our model indicates that drainage over the sides of floes (or 'edge effects' as they were referred to in [Scott and Feltham, 2010]) are only important if the ice is very flat and smooth, the floe is quite small (having few catchments), or is very impermeable (i.e. drains late). The model showed that the precise value of the permeability once drainage begins is unimportant beyond a value of around 10^{-10} m^2, and that the time at which drainage begins has a much larger effect on the behaviour of the system. The model also neglected horizontal permeability of the ice. We argue that this would not become important until the vertical permeability is also significant, but at that stage the vertical transport of water dominates.

Several other processes were neglected, including snow cover, and the initial leaching of water into brine channels. Instead, we assumed that snow cover was negligible, and that initial melting and refreezing had filled any brine channels, at which point ponds begin to appear on the surface. Melting at the base of the pond was neglected, even though this may have an effect on how the draft changes in time due to ice volume losses at the base. This should not affect the surface dynamics until after drainage, but, if accounted for, would change the behaviour during refilling.

The most important simplification made in the model is the assumption of constant melt rates. Time dependent melt rates were investigated, but, so long as v_w and v_b changed at the same rates, time dependent melt rates had no major effect on

the behaviour of the model, except to scale time during simulations, and change the timing between events (for this reason, results from simulations with varying melt rates were not presented). The ratio v_w/v_b was shown to be of only moderate importance to the behaviour of the model, especially when changes to the slope of the pond floor were neglected. The model was only run for the period of one summertime melt season. In the future, it would be both feasible and useful to incorporate some form of re-freezing of melt ponds, and run simulations over several annual cycles, to investigate how pond extent, and hence ice albedo, change over several years according to the model.

The model we have presented could be used to explore several questions that have not featured in this thesis. One of the more obvious future applications for the model is to expand on the work of Arntsen et al. [2015], who showed that ponding induces the break-up of floes, and that floes are likely to fracture along the long axis of ponds, due to the ice being thinner below the centre of ponds. By running our model beyond the time at which the first pond breaks though the floe, we could map out where floes are likely to fracture, by identifying lines along which the ice is thinnest late in the melt season. Particular questions to answer in such work would include finding the expected distribution of child floes into which a parent floe fractures, based on the initial topography and the melt rates.

There are several other further applications for this model, and the ideas that underpin the model. The one-dimensional model presented at the end of Chapter 4 could be extended to model systems of lakes on ice shelves and ice sheets, if drainage through fractures is added. Observations of these features have increased over recent years, and they are likely to have a destabilising effect on the ice [Kingslake et al., 2017]. The two dimensional pond model resolves changes in pond geometry through time, and could be extended to model how the irradiance under the floe changes through the melt season due to pond cover, similar to the work done by Horvat et al. [2020]. This is of particular interest to marine biologists, as phytoplankton and other organisms have been shown to exist in increasing abundance in the waters underneath ice with melt ponds, where the radiation field that penetrates through the ice is strong enough to allow photosynthesis [Arrigo et al., 2012]. Phytoplankton have also been found in ponds [Lee et al., 2012]. Our network model for pond evolution is well suited to modelling the diffusion of nutrients, salt and micro-organisms in systems of melt ponds. A model for the diffusion of salt would be a particular useful application of our model, as the (albeit low) salinity of pond water (which we have neglected), can increase after drainage. Evidence suggests that the accumulation of salt in ponds can

have a non-trivial impact on the energy budget of the ice-ocean system [Kim et al., 2018].

The agreement of the model with data indicates that a catchment-based method based on differential equations on networks may provide a useful way to parametrize ponds. In analysing the behaviour of the model we found that there are periods where fluxes between catchments have a negligible influence on the behaviour of the system, such as at early time, and during refilling, and the behaviour of the system is similar to the behaviour of the average catchment. Fluxes between catchments are important near flooding, but during this time the activities on nodes in the network are synchronized. For this reason, near flooding, the behaviour can be described using a mean-field approximation. Despite some of the changes in the system occurring during the other periods - midway through filling, and during drainage - the agreement of mean-field approximations with large periods of the model output indicates that it may be possible to parametrize the evolution of pond area in the form of a set of ODEs using a such a method.

8.3 Future possibilities for pond extent in a warming world

What consequences do the conclusions we have drawn from our model have for the future Arctic, given that it is changing rapidly? According to multi-decadal trends [Kwok and Untersteiner, 2011], the Arctic ice cover is shrinking in extent, and becoming younger, with multi-year ice being steadily replaced with short-lived first-year ice [Kwok and Cunningham, 2015], which tends to be flatter, but has a greater brine fraction which can make it more permeable [Golden et al., 2007].

The melt season is also elongating, [Perovich et al., 2020]. The summer of 2020 was the longest on record, with the Laptev sea only starting to refreeze in late October. Air temperatures in the Arctic are warmer now than in decades past, especially during the winter months, but also during the summer [Ballinger et al., 2020]. The ocean itself is warming, due to warm water from both the Pacific and Atlantic bringing heat northwards [Kodaira et al., 2020]. This raises the question of how are these changes relevant to the parameters in our model, and can we use the model to inform how we expect pond coverage, and annual ice loss, to change as the planet warms in the short to medium term.

The picture is complicated. Areas in the far north which previously saw little ponding, due to insufficient air and ice temperatures to cause extensive melting, will

see more ponding as this region warms. For areas where ponds form already, we explore several scenarios, shown schematically in Figure 8.1: a base case (blue), a case with higher atmospheric temperatures (red), a case with higher temperatures and flatter ice (yellow), and a case with higher temperatures, flatter ice and earlier drainage (purple). The data for these cases were chosen from amongst the experiments presented in Section 7.2. The flattening of the ice cover, reflected in a decrease in Δh could produce either less ponding if the ice is more porous, or more ponding if the ice porosity is unchanged, as shown in panel (a). While the case in yellow (hotter and flatter) displays more initial flooding of the floe, ponds disappear post-drainage, and only recover a small amount of area during refilling compared to the cases in blue and red (the base case and case that is just hotter). The increasing temperatures, both at the surface and the base, may lead to an increase in h_{crit}. According to panel (a), both of these changes would lead to a lower overall pond fraction in late summer. Further, in the case of thinner first year ice, the breaching of pond floors through floes, and the subsequent disintegration of floes may occur earlier. This would make the spring peak, shown in yellow on day 5 in panel (a), more significant.

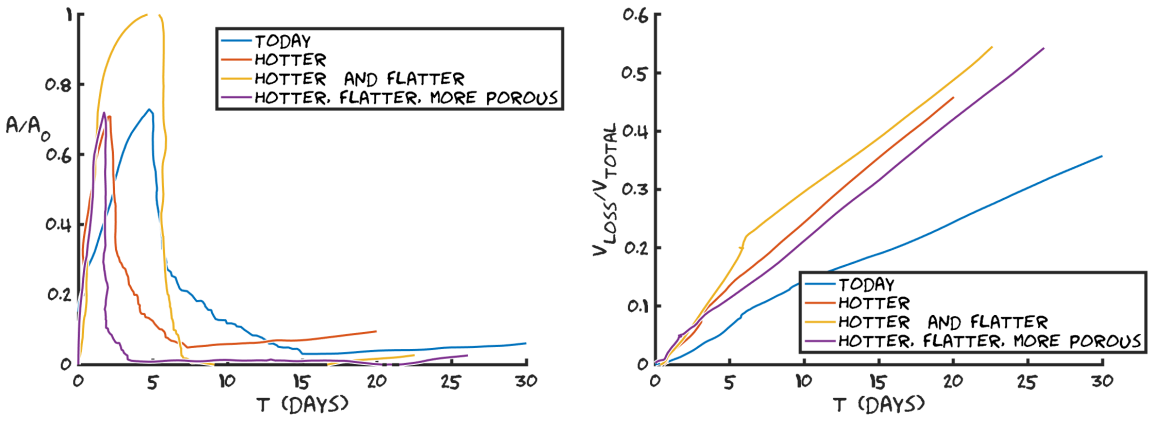


Figure 8.1: Cartoon illustrations of possible future scenarios (a) for pond extent and (b) Fractional volume of ice loss, for a range of scenarios. Time $t = 0$ refers to beginning of pond formation (late May, early June). The red, yellow and purple plots end when the base of the floe is breached by a pond floor. Each plot consist of data taken from the experiments described in Section 7.2 using one initial topography of 100 nodes. 'Today' refers to an experiment with $v_b = 1$, $v_w = 2$, $\hat{K} = 10^3$, $\Delta h = 0.2$, $\bar{h} = 1$, $h_{crit} = 0.7$ and a scaling whereby $v_b = \tilde{v}\hat{v}_b$ and $\tilde{v} = 1$ cm/day. 'Hotter' refers to the same data but with $\tilde{v} = 2$ cm/day. 'Hotter and flatter' uses $\tilde{v} = 2$ cm/day and $\Delta h = 0.1$. 'Hotter, flatter, more porous' uses $\tilde{v} = 2$ cm/day, $\Delta h = 0.1$ and $h_{crit} = 0.85$.

According to Figure 8.1 (b), which shows fractional ice volume loss for the same scenarios, a larger proportion of flatter first-year ice, shown in yellow, would lead to increases in volume loss along with increases due to higher summer temperatures. The scenario shown by the purple line suggests that an earlier onset of drainage could offset increases in volume loss due to higher temperatures and flatter ice. This is due to the reduction in the extent of the ice-albedo feedback. Given how early drainage occurs in the test cases described in Section 7.4, the potential for increases in h_{crit} is low (real values for h_{crit} are probably close to $\bar{h} - \Delta h$ so the difference of the pond floor to the depth at which drainage begins probably cannot decrease much more). Therefore offsetting in volume loss may be marginal.

While the ice has thinned and decreased in extent over the last two decades, the extent of the marginal ice zone, the area near the exterior of the Arctic ocean made up of between 15% and 80% ice cover, has remained almost constant [Rolph et al., 2020]. This in effect is an increase in the extent of the MIZ relative to total ice cover. In the MIZ floes tend to be smaller and more fractured, due to a lower ice concentration here than in the central Arctic. These floes tend to be rougher, due to being subject to greater dynamic effects of surface waves that are strong at the ice edge, but damp strongly as they propagate in towards the central ice pack, as the floes at the MIZ are free to collide with one another and form ridges. According to the model with side drainage discussed in Chapter 6, this fracturing of floes would limit the maximum pond area reached before drainage.

To summarise, the future Arctic is on a well understood path to being ice-free in the summer-time by mid-century, possibly as early as 2035 [Guarino et al., 2020]. Along the path to this state, according to our model, there is likely to be more ponding in the far north, or very central Arctic due to warming. In the mid-north, between the MIZ and the central Arctic north of 85° there is likely to be less ponding due to a larger abundance of flatter first year ice, with more porous ice throughout, and higher summertime temperatures. Volume loss will likely accelerate in all cases, primarily due to predicted increases in surface air temperature (increasing the melt rates v_b and v_w), and increasing ocean temperatures which cause enhanced melting at the base of the floe. All stages in pond evolution are set to be shorter and shorter-lived as the years progress, due to increases in summertime temperatures everywhere, and subsequent increases in melt rates. That said, given that these changes do not disrupt ice-formation in winter, if society can successfully de-carbonise by mid-century, it is possible for the ice to eventually recover in extent and volume, and eventually we

may even again see the great melt lakes described by Zubov [1943] in the early 20th century.

Appendices

Appendix A

Numerical considerations

A.1 Evolving the model for impermeable ice

In performing a simulation, we first define a topography and network structure as described in Section 5.1, and determine the initial conditions t_0 the initial time, $\mathbf{H}(t_0)$ the vector of water levels in each catchment at time t_0, and $\mathbf{H}_s(t_0)$, the vector of saddle heights at time t_0.

Having gathered the initial conditions, we determine the initial logical vectors $\mathbf{i}_1$ to $\mathbf{i}_4$ described in Section 4.4, based on the initial pond water levels and saddle heights. We then go about breaking any possible loops (for cases where a simulation is started with a sufficient amount of water such that some saddles may already be submerged). To do this we define the graph of joined edges G_4 as the graph with set of nodes (1 to n) and the set of edges in state 4 (those edges for which the corresponding entry in $\mathbf{i}_4$ is equal to 1) and then break any loops in G_4. To do this we construct a minimum spanning tree of G_4, using Prim's algorithm (for which MATLAB has an inbuilt function) and move any edges included in G_4 but absent from the spanning tree, to the graph of inactive edges, G_1 – the graph with set of nodes 1 to n and the set of edges in state 1. For any edges moved, we set the corresponding entries in $\mathbf{i}_4$ to 0 and the corresponding entry in $\mathbf{i}_1$ to 1.

With the initial conditions and state of the edges determined, we integrate the model described by equations (4.34) and (4.35) using MATLAB's ODE15s. We search for a change of state of the network (an 'event') using the events function

$$\mathcal{F}(\mathbf{H}, \mathbf{H}_s) = \begin{pmatrix} \mathbf{H}_s \\ \mathbf{H}_s \end{pmatrix} - \begin{pmatrix} \mathbf{LH} \\ \mathbf{RH} \end{pmatrix} \tag{A.1}$$

which will exit the integrator when any entry of $\mathcal{F}$ passes through 0 from above or below, and returns the output of the integrator up until that point. This catches

events for both filling and draining of ponds. The vector **LH** is a vector of catchment heights ordered by the left column of the edge list, generated by the mapping of levels to the left of their associated edges (it is therefore of length m, the number of edges, rather than n the number of nodes); **RH** is the equivalent for the right column. We store the output, then reset the state of the edges by determining the indicators $\mathbf{i}_1$ to $\mathbf{i}_4$ and break any loops in the new graph of active edges, $G_a = G_2 + G_3 + G_4$, where G_2 is the graph of edges in state 2 and so on. To do this, we consider each edge as it changes state, and disallow the change if it causes a loop in G_a. We also change the entry of the disallowed edge from 1 to 0 in the appropriate active edge indicator $\mathbf{i}_2$, $\mathbf{i}_3$ or $\mathbf{i}_4$ and the entry in $\mathbf{i}_1$ from 0 to 1. These broken loops are reinstated during drainage once the topology forming the loop disappears. We cover this in Chapter 6. In order to eliminate numerical drift, water levels are adjusted each time there is a change of edge state such that levels in overspilling ponds are equal to the saddles they are overspilling, and that at the point a join edge forms, the water levels of the two joining ponds are equal to the height of the saddle between them.

We also disallow more than one overflow edge out of any one connected component in the graph of join edges, as we deemed this to be unphysical in Chapter 4. To do this we consider each individual pond (connected component of G_a) in turn and count the number of outwards overflows. If this number is greater than one for any pond we switch off all except the one with the lowest associated saddle. If the pond has two or more (out) overflowing saddles with the same height, we choose which overflow to keep at random.

After an event, once we have set the states of the edges such that they represent the new configuration of ponds, and such that there are no loops in the network, we reinitialise the integrator with the new state of the edges and with initial conditions from the end of the previous run. This process continues until the integrator reaches a specified time. From the integration we store the water levels $\mathbf{H}(t)$, water-covered areas $\mathbf{a}(t)$, melt volume created in each catchment $\mathbf{V}_m(t)$, and saddle heights $\mathbf{H}_s(t)$ across the whole time interval. We also record the state of the edges at every event (the change of state of any edge), the time of each event and the water levels, water-covered areas and saddle heights at the time of each event.

Appendix B

The pond floor

Figure B.1 shows drilling transects taken from refrozen ponds during the Healy 1902 expedition in September of 2019. The pond floors (in blue) all show smooth, relatively flat profiles, but with a steep incline near the shore. This shape is well known to oceanographers, with ponds later in the season forming near vertical walls with water levels below the lip of these cliffs. Late in the season, these sheer sides can develop into overhangs due to lateral melting. The development of this geometry is described schematically in figure B.2. Little else has been written about the formation of these shapes.

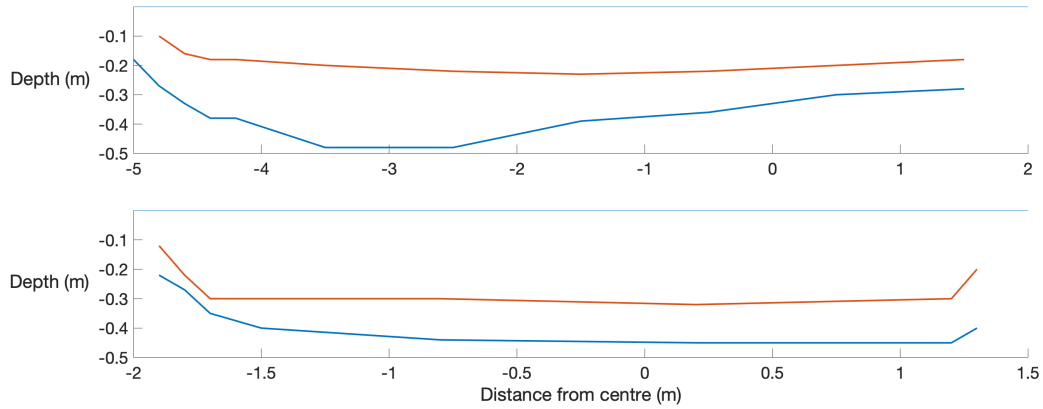


Figure B.1: Transects of two partially refrozen ponds taken from Ice Stations 1 from the Healy 1902 expedition on September 17^{th} 2019, at 78 53.2′N 141 50.8′W. The zero of height is taken to be the flat ice surface of the refrozen ice cap. Red lines are the base of the refrozen ice cap. Blue lines are the base of the pond. Distances are measured from a reference centre-point in each pond.

In this section, we build on the discretisation method for tracking and accounting for topographic changes used in Chapter 6, Section 6.1.3, equations (6.1), (6.3), (6.5) and (6.21) to analyse the one dimensional model, so that we can explore how the

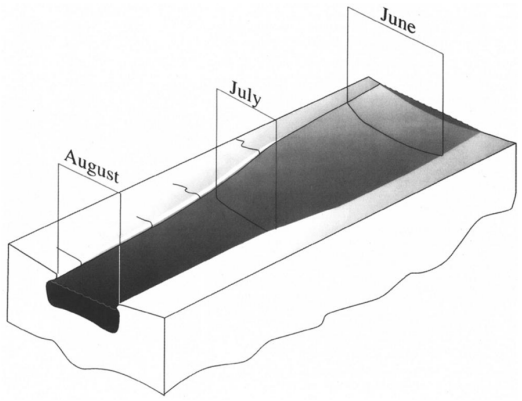
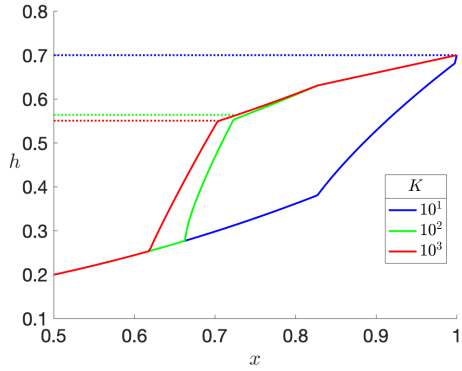


Figure B.2: Schematic of pond geometry through the melt season, taken from [Fetterer and Untersteiner, 1998].

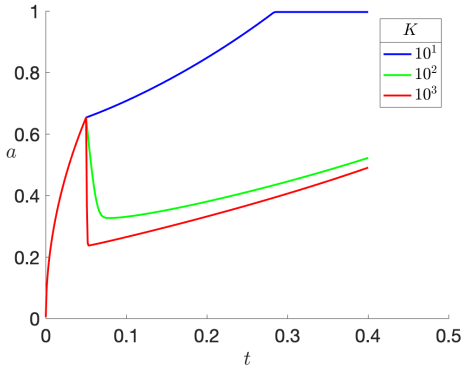
shape of the pond floor changes through the drainage period. By considering one half of symmetric topographies in one dimension, we explore how the shape of the pond floor changes between the onset of melting and late summer, particularly how this depends on the parameters. We also investigate whether there is a marked difference in the pond floor in late summer between a half-catchment with a flat sloping face, and one with a cosine curve, similar to the two topographies we examined in Chapter 3.

Using the discretization method, we carried out several experiments on a one dimensional floe of unit height and area to investigate how the different aspects of the melting process change the shape of the pond floor, and investigate how the changes depend on the parameters in our model. For these experiments we used a reference set of parameters $K = 1000$, $\Delta h = 0.2$, $h_{crit} = 0.7$, and $v_w = 2$. Profiles are taken at a common time $t = 0.3$ for all simulations, as some simulations last longer than others, and in examining differences in shape of pond floors, we want to make a fair comparison between simulations.

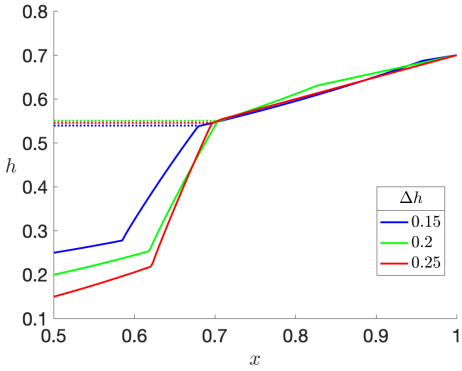
Figures B.3 and B.4 show a set of experiments on an initially half-notch shaped topography for a catchment with half-width 1, varying parameters. Examining the dependence on $\hat{K}$, shown in Figure B.3 (a) and (b), we see behaviour in line with the results of experiments from previous sections whereby $\hat{K} = 10$ is too low. No substantial drainage is observed from (b), and once the catchment fills, a cliff begins to forms, seen in (a). The curves for the other two values of $\hat{K}$ show similar gradual slopes forming close to the pond boundary, even though the values of $\hat{K}$ differ by factors of ten. The behaviour of pond extent for these two cases is also much more realistic, with a definite drop in pond extent at the onset of drainage, and a slow



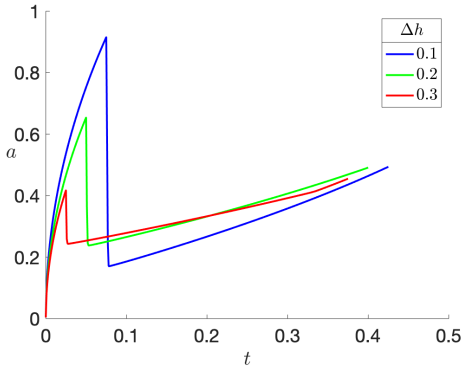
(a) Profiles, varying K



(b) Areas, varying K



(c) Profiles, varying Δh

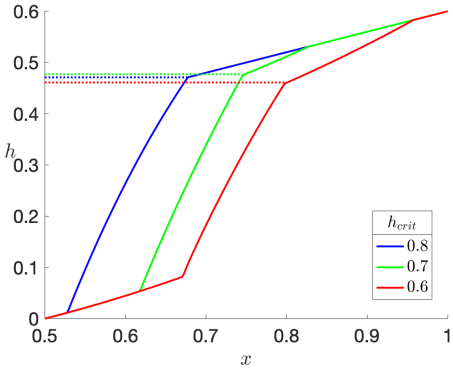


(d) Areas, varying Δh

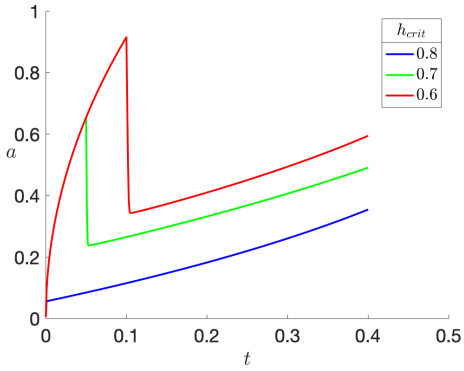
Figure B.3: Half pond floor profiles (one half of a symmetric profile in one dimension), shown in solid lines, (left) taken at time 0.3, and areas (right) for experiments with drainage. A standard set of parameters of $K = 1000$, $\Delta h = 0.2$, $h_{crit} = 0.7$, and $v_w = 2$ is used as a base case. For the profiles, sea level is just below the water levels, which are marked by the dotted lines.

increase thereafter, with values similar to those shown Figure 7.7(a) and (b), data from Polashenski et al. [2012].

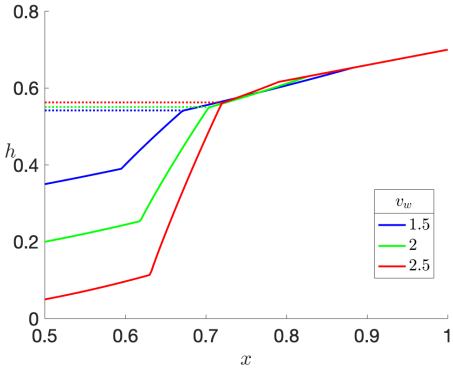
Panels (c) and (d) show the dependence on the surface roughness, Δh. Across changes in Δh, little difference is seen in the shapes of the pond floors. However, there is a wide variety of behaviours of the evolution of pond extent, with, in the smoothest case, the catchment filling almost completely, while the other two cases only achieve pond extents of around half of the catchment at any time in the simulation. Interestingly, each case has around the same final pond area, which means that it is possible that no matter the roughness of the initial ice, we end up with the same shape of pond floor. It must be noted however, that the distance that the lowest point in the catchment, has to travel before the onset of drainage is different for each



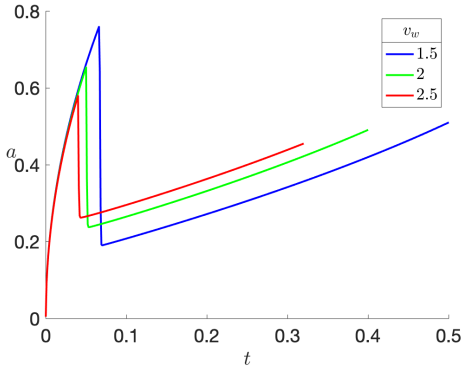
(a) Profiles, varying h_{crit}



(b) Areas, varying h_{crit}



(c) Profiles, varying v_w



(d) Areas, varying v_w

Figure B.4: Half pond floor profiles (one half of a symmetric profile in one dimension), shown in solid lines, (left) taken at time 0.3, and areas (right) for experiments with drainage. A standard set of parameters of $K = 1000$, $\Delta h = 0.2$, $h_{crit} = 0.7$, and $v_w = 2$ is used as a base case. For the profiles, sea level is just below the water levels, which are marked by the dotted lines.

of the three cases presented here.

The blue curve of panel (a) of B.4 demonstrates that ice that drains immediately forms a pond floor shaped like a deep notch rather that a floor that is wider and flatter in the middle, with steep sides (such as the red curve, and shown in Figure B.2). In fact, of panel (a) of Figure B.4, the red curve, with the smallest critical thickness and therefore latest onset of drainage shows the closest behaviour to the transects in Figure B.1. This demonstrates that it is not just the permeability of the ice that causes extensive ponds with steep sides, but the initial impermeable phase of melting and pond formation followed by a permeable phase of drainage and further melting. The transition in permeability is therefore important to the formation of ponds as we observe them, and h_{crit} could be low in places where we see well developed ponds.

Panel (c) of B.4 shows that preferential melting is important in forming the steep

sides that we see in ponds, whereby $v_w = 1.5$ is too small to produce the steep sides seen in observations. This suggests that it is likely that $v_w > 1.5v_b$, and closer to $2v_b$ or higher. Panel (d) shows small difference in the final areas of ponds, but a big difference in how quickly the pond floor melts through to the base of the floe.

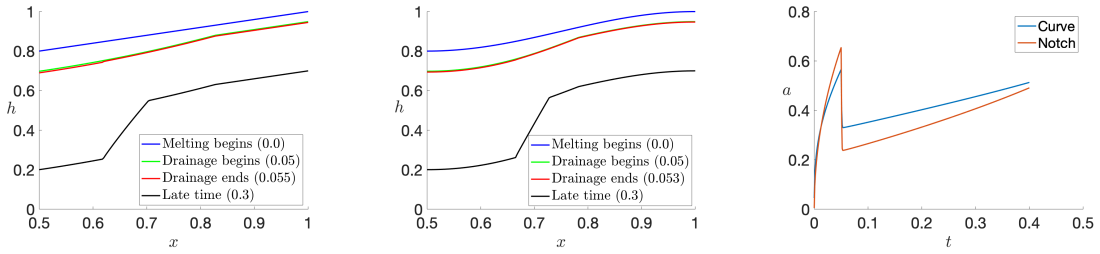
Figure B.5 contrasts the evolution of topographies for notch shaped and curve shaped initial topographies for catchments with half-width 1. The equation for the half-notch profile is

$$\bar{h} + \frac{\Delta h}{2}(x - 1). \tag{B.1}$$

The initial topography curve is given by

$$h(x) = \bar{h} + \frac{\Delta h}{2}[\cos(2\pi x) - 1]. \tag{B.2}$$

The figure shows that they behave similarly, forming almost identical underwater profiles in late time, and similar area coverage throughout. This further justifies the triangular mesh approximation for pond topographies used in the network model.



(a) Notch profile (b) Sinusoidal profile (c) Areas

Figure B.5: Half profiles at the time of some specific events as indicated in the legend for two different initial topographies in catchments with half-width 1 (exact times are given in the legend). (a) A notch, defined by equation (B.1) (b) A sinusoidal curve, defined by (B.2). (c) shows the time series of pond areas for the two experiments. Parameters used are $K = 1000$, $\Delta h = 0.2$, $h_{crit} = 0.7$, and $v_w = 2$.

In this analysis we have ignored lateral melt, however the conclusions are relevant where lateral melting is present, as the cliffs formed during overflow and the steep sides formed after drainage can be exploited by lateral heat fluxes to further steepen and even undercut the sides of the ponds due to lateral energy fluxes in the pond caused by turbulent mixing. We find that the greatest driver of steepening of the pond floor at the edges is the period after drainage begins, which leads to deepening of the pond and the formation of the side walls. Any period in which the pond overflows also works to steepen the walls early on.

Appendix C

Fractals

C.1 Introduction to Hausdorff and fractal dimension

For a rigorous explanation we first consider Lebesque measure, defined as

$$\mathcal{L}^n(F) = \inf\left\{\sum \mathrm{vol}^n(F_i) : F \subset \bigcup_{i=1}^{\infty} F_i\right\} \tag{C.1}$$

for a set F, which can be divided into coordinate parallelepipeds

$$F_i = \{(x_1, \ldots, x_n) \in \Re^n : a_j \leq x_j \leq b_j\}, \tag{C.2}$$

where the a's and b's mark opposite ends along dimension j of individual parallelepipeds. Here inf is the infimum and the n-dimensional volume of F_i is

$$\mathrm{vol}^n(F_i) = (b_1 - a_1)(b_2 - a_2)\ldots(b_n - a_n). \tag{C.3}$$

The Lebesque measure of dimension n covers completely a set with smaller shapes, and sums the n-dimensional volume (area for $n = 2$, volume for $n = 3$) of these shapes. This defines one of the simplest measures of a set with a smooth boundary, and generalises the concepts of length, area, volume, and so on into higher dimensions. We define the diameter of a set $|F| = \sup\{|x - y| : x, y \in F\}$ where sup is the supremum. From this we also define a δ-cover as a countable collection of sets of diameter at most δ that cover U, $U \subset \cup_{i=1}^{\infty} U_i$, $0 < |U_i| < \delta$.

We can further generalise Lebesque measure by the Hausdorff measure, central to the ideas in fractal geometries. First, we consider a δ-cover of a set F, such that at a scale δ we can define the measure

$$\mathcal{H}^s_\delta(F) = \inf\{\Sigma_{i=1}^{\infty}|U_i|^s : \{U_i\}\text{is a } \delta\text{-cover of } F\}. \tag{C.4}$$

With this, we measure the tiling of covering shapes of dimension s and diameter δ needed to cover the set F. In the context of measuring a coastline length or island perimeter, we would calculate this measure summing how many times we would need to lay a ruler of length δ along the coastline to traverse its entire length. This gives the length of coastline to a resolution of δ. The Hausdorff measure is the idealised limit of this measurement technique, such that

$$\mathcal{H}^s(F) = \lim_{\delta \to 0} \mathcal{H}^s_\delta(F) \tag{C.5}$$

For any given set this measure is not necessarily finite or well defined for all s. In fact, it turns out that there is a value of s which divides those values for which $\mathcal{H}^s$ is zero from those for which it is infinite, and we define this value to be the Hausdorff dimension of F. That is, $\dim_H F = \inf\{s : \mathcal{H}^s(F) = 0\} = \sup\{s : \mathcal{H}^s(F) = \infty\}$ so that

$$\mathcal{H}^s(F) = \begin{cases} \infty & \text{if } s < \dim_H F \\ 0 & \text{if } s > \dim_H F. \end{cases} \tag{C.6}$$

A fractal set is one for which the Hausdorff dimension, the exponent on the Hausdorff measure, is not an integer. As an example, consider F to be the flat disc of unit radius in $\mathbb{R}^3$. Here,

$$\mathcal{H}^1(F) = \text{length}(F) = \infty$$
$$0 < \mathcal{H}^2(F) = \frac{4}{\pi} \cdot \text{area}(F) < \infty$$
$$\mathcal{H}^3(F) = 6\pi \cdot \text{vol}(F) = 0$$

from which we determine that $1 < s < 3$. We know however, from the standard definition of dimension that $\dim_H F = 2$ for a flat disk, rather than the range $1 < s < 3$.

From this simple example we see that Hausdorff measure and dimension are useful but not necessarily easy to calculate. If F is a Borel subset of $\mathbb{R}^n$, where a Borel set is defined as a set that can be formed from the countable union, intersection or relative complement of closed or open sets, then

$$\mathcal{H}^n(F) = c_n \text{vol}^n(F) \tag{C.7}$$

where c_n is a shape constant. From this relation we can return to our δ-scale Hausdorff measure and write

$$\mathcal{H}^n_\delta(F) = c_n \delta^n. \tag{C.8}$$

This is the widely used power law expression of a measure.[1]

Hausdorff measure is rarely used in practice. Instead we derive the box-counting dimension by writing the power law relation for measure (C.8) as

$$N_\delta(F) \sim c\delta^{-s} \tag{C.9}$$

which we can rearrange as

$$s = \lim_{\delta \to 0} \frac{\log N_\delta(F)}{-\log \delta} \tag{C.10}$$

We compute the measure N_δ as the number of boxes of diameter δ taken to completely cover the set F, making the more abstract Hausdorff measure more usable by constraining the shapes used in the δ-cover. We then define the lower box counting dimension

$$\underline{\dim}_B F = \underline{\lim}_{\delta \to 0} \frac{\log N_\delta(F)}{-\log \delta} \tag{C.11}$$

and upper box counting dimension

$$\overline{\dim}_B F = \overline{\lim}_{\delta \to 0} \frac{\log N_\delta(F)}{-\log \delta}, \tag{C.12}$$

where the underbar and overbar denote the limit inifmum and limit supremum respectively. If $\underline{\dim}_B F = \overline{\dim}_B F$ we call this common value the box counting dimension. This is the more commonly seen definition of box counting, or fractal dimension,

$$D(F) = \dim_B F := \lim_{\delta \to 0} \frac{\log N_\delta(F)}{-\log \delta}. \tag{C.13}$$

It can be shown that by covering F by N_δ sets of diameter δ,

$$\mathcal{H}^s_\delta \leq N_\delta(F)\delta^s, \tag{C.14}$$

which follows from equation (C.4), as the boxes form a δ cover. We can then infer that

$$\dim_H(F) \approx \dim_B(F). \tag{C.15}$$

If F is a subset of $\mathbb{R}^n$ then we can write the sometimes useful form of the lower box-counting dimension

$$\underline{\dim}_B F = n - \underline{\lim}_{\delta \to 0} \frac{\log \mathrm{vol}^n(F_\delta)}{-\log \delta} \tag{C.16}$$

[1]This is the equation used to motivate the power law formulation of box counting dimension in Chapter 2.

and upper box counting dimension

$$\overline{\dim}_B F = n - \overline{\lim}_{\delta \to 0} \frac{\log \mathrm{vol}^n(F_\delta)}{-\log \delta} \tag{C.17}$$

where F_δ is the δ-parallel body to F,

$$F_\delta = \{x : |x - y| \leq \delta \text{ for some } y \text{ in } F\}$$

The δ-parallel body in effect coats the set in a shell of thickness δ. The result above is the reason for sometimes referring to the box counting dimension as Minkowski dimension, due to Minkowski's development of the Minkowski curve a, δ-parallel body used to measure the length of curves.

C.2 Time slices of perimeter area scaling - an example

To explore how the scaling behaviour and fractal dimension of ponds changes over time, we investigate simulations at distinct time slices. To examine the changes in scaling during the filling stage in more detail we take output from a simulation on impermeable ice at specific times and put each pond on an area - perimeter plot, as in Figure 7.9(b). The output from this process is shown in Figure C.1, at time intervals of 2.5×10^{-3}. Due to the very fast pond growth at early times for the model, there are very small ponds present by $t = \delta t > 0$, Figure C.1(a). As described in Chapter 5, we begin the simulation slightly after time zero, at time δt. By $t = 0.0025$, Figure C.1(b), they have grown in size to form a large cluster of points in the centre of the plot. As we go from $t = 0.0025$, Figure C.1(b) to $t = 0.0225$ Figure C.1(j), the number of individual ponds has decreased dramatically as more joins have formed, and we see the largest clusters in each ensemble member appear in the top right of the plot. The positions of these points are limited by the size of the domain. No one pond can have area greater than $n\bar{a}_0$. The maximum perimeter does not have the same restriction. Even for a fixed total area it may go to infinity if n does, as in the case of a Peano (space-filling) curve. The ponds not connected to the largest pond appear to be either single ponds or more ramified ponds. With more simulation members here it would be possible to fit the two trend lines similar to Figure 7.8. Through $t = 0.0225$, Figure C.1(j), and $t = 0.05$, Figure C.1(o), yet more of the smaller ponds are absorbed into the spanning cluster, which grows in area, moving to the right of the plot, but not far up or down. At $t = 0.035$, Figure C.1(o), only

a handful of ponds remain disconnected from the largest, and by $t = 0.05$, Figure C.2(a), these have disappeared completely as the system moves towards flooding. We see a decrease in perimeter relative to Figure C.1(a), with little change in area, due to shores disappearing as the last peaks are covered.

To examine the changes in scaling over the whole cycle of filling, draining and refilling, we take output from several realisations of one of the experiments described in Section 7.2 and plot the output in the same way as for Figure C.1. Figure C.2 shows perimeter - area plots from when the pond system is almost totally connected, through drainage, to the end of the simulation, at time slices 0.05 apart. In (b) and (c) we see the process of the floe flooding: small increases in area but decreases in perimeter as the remaining peaks become small islands that steadily disappear. Panel (e) $t = 0.25$ occurs during the drainage process (see Figure 7.1 (c)), and at this point, ponds are decreasing in size and separating. The path through drainage is short lived and complicated, and the cluster of points in (e) shows little evidence of structure.

Panels (f), (g) and (h) show ponds during refilling, and here we see the kind of behaviour described in the observations emerging, with, in each plot, a cluster of ponds that to the left (smaller ponds) appear to scale with $D \approx 1$, and to the right (larger ponds) appears to deviate from this scaling towards $D \approx 2$.

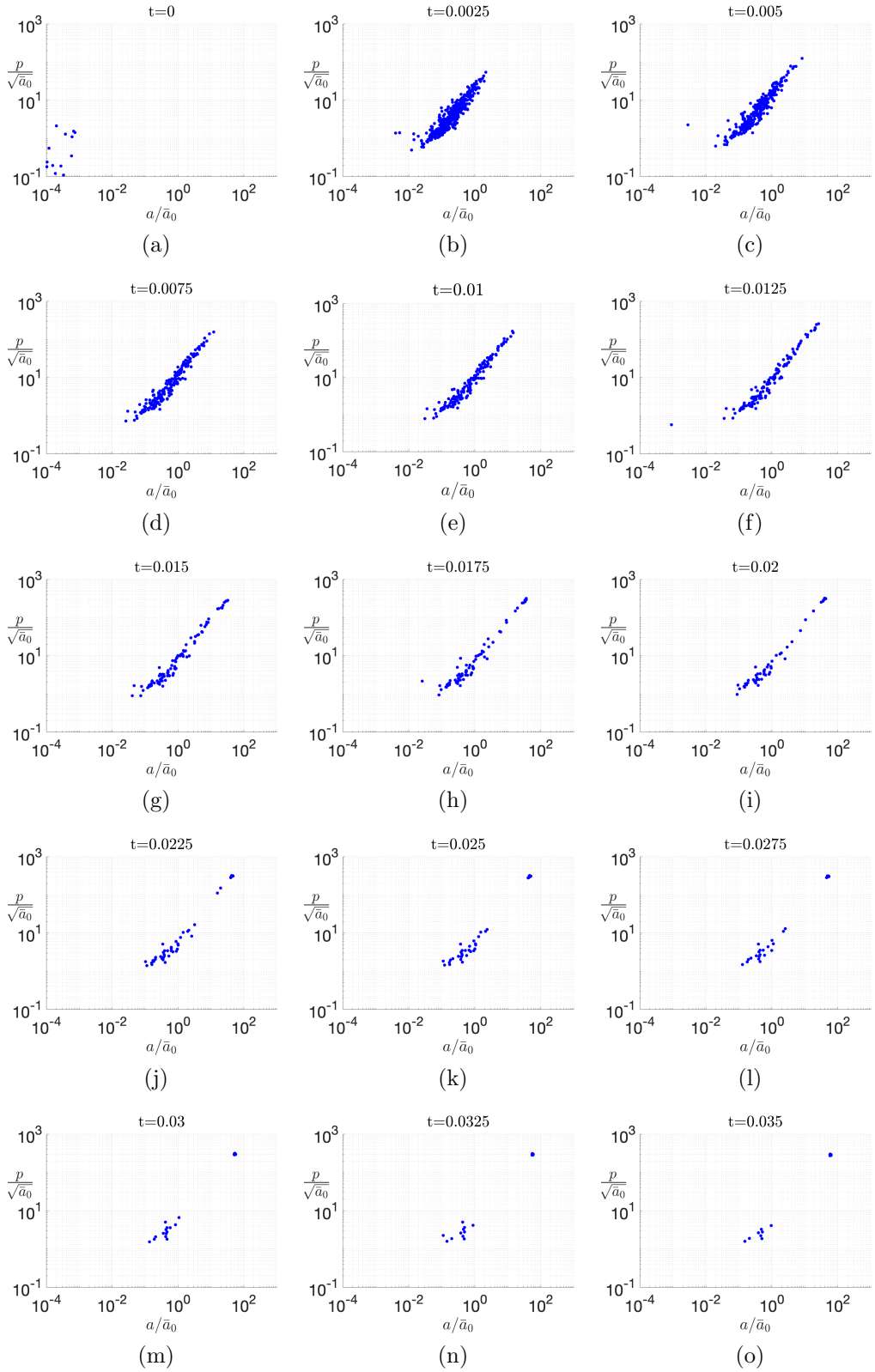


Figure C.1: Perimeters and areas of ponds at intervals during simulations on impermeable ice, with 225 catchments on a Voronoi tessellation, using $v_w = 2$, $\Delta h = 0.15$ and twelve ensemble members.

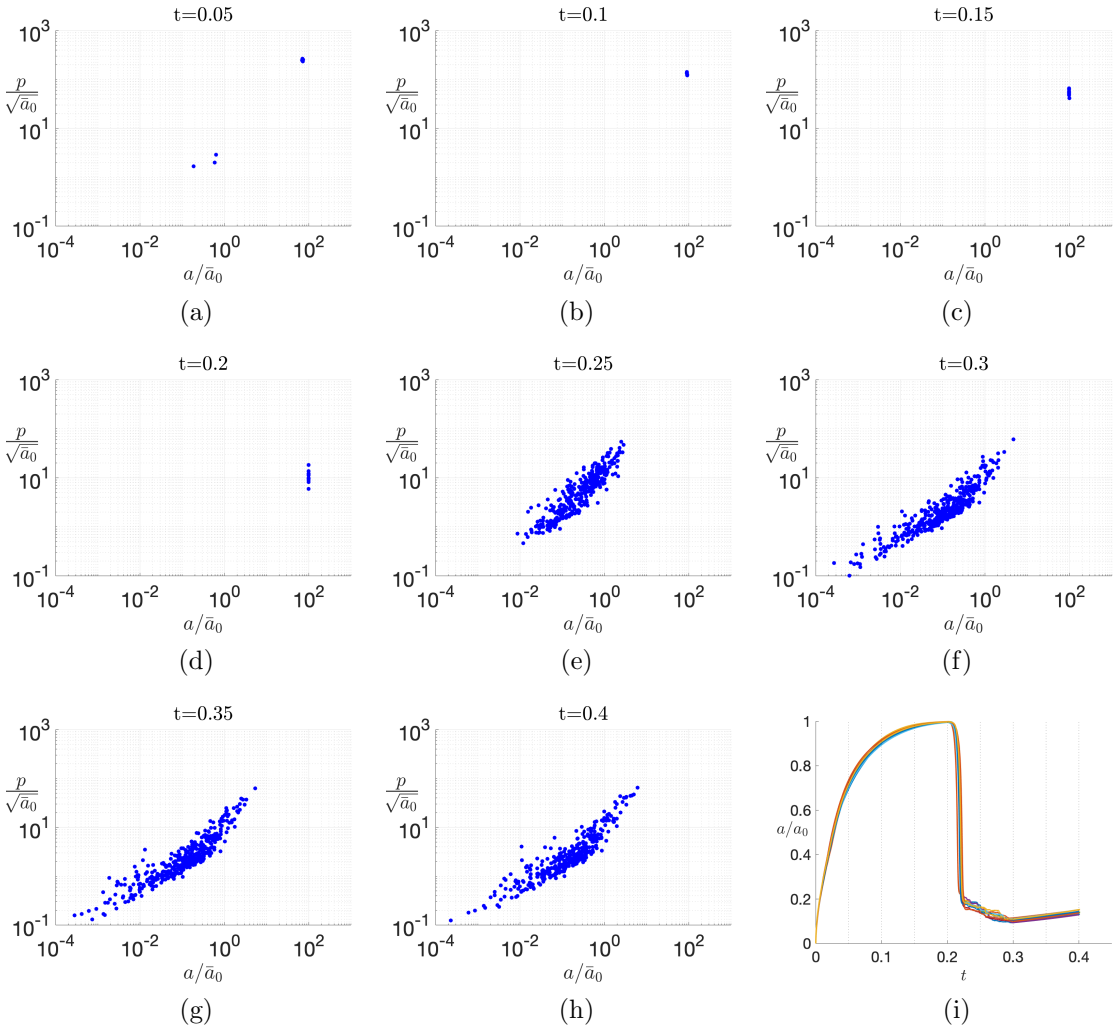


Figure C.2: (a) to (h): Perimeters and areas of ponds at intervals during simulations with drainage as described in Section 7.2, with 100 catchments on a Voronoi tessellation, using $v_w = 2$, $\Delta h = 0.2$ and $h_{crit} = 0.4$ and ten ensemble members. Area fractions for the simulations are shown in panel (i).

Bibliography

D. Achlioptas, R. M. D'Souza, and J. Spencer. Explosive percolation in random networks. *Science*, 323(5920):1453–1455, 2009. ISSN 00368075. doi: 10.1126/science.1167782.

S. Agarwal, W. Moon, and J. S. Wettlaufer. Decadal to seasonal variability of Arctic sea ice albedo. *Geophys. Res. Lett.*, 38(20):1–5, 2011. ISSN 00948276. doi: 10.1029/2011GL049109.

A. Arenas, A. Díaz-Guilera, J. Kurths, Y. Moreno, and C. Zhou. Synchronization in complex networks. *Phys. Rep.*, 469(3):93–153, 2008. ISSN 03701573. doi: 10.1016/j.physrep.2008.09.002.

A. E. Arntsen, A. J. Song, D. K. Perovich, and J. A. Richter-Menge. Observations of the summer breakup of an Arctic sea ice cover. *Geophys. Res. Lett.*, 42(19):8057–8063, 2015. ISSN 19448007. doi: 10.1002/2015GL065224.

K. R. Arrigo, D. K. Perovich, R. S. Pickart, Z. W. Brown, G. L. Van Dijken, K. E. Lowry, M. M. Mills, M. A. Palmer, W. M. Balch, F. Bahr, N. R. Bates, C. Benitez-Nelson, B. Bowler, E. Brownlee, J. K. Ehn, K. E. Frey, R. Garley, S. R. Laney, L. Lubelczyk, J. Mathis, A. Matsuoka, B. G. Mitchell, G. W. Moore, E. Ortega-Retuerta, S. Pal, C. M. Polashenski, R. A. Reynolds, B. Schieber, H. M. Sosik, M. Stephens, and J. H. Swift. Massive phytoplankton blooms under arctic sea ice. *Science (80-.).*, 336(6087):1408, 2012. ISSN 10959203. doi: 10.1126/science.1215065.

T. J. Ballinger, J. E. Overland, M. Wang, U. S. Bhatt, E. Hanna, I. Hanssen-Bauer, S. J. Kim, R. L. Thoman, and J. E. Walsh. Arctic report card 2020: surface air temperature. Technical report, 2020.

A.-L. Barabási and M. Pósfai. *Network science*. Cambridge University Press, Cambridge, 2016. ISBN 9781107076266.

M. Barjatia, T. Tasdizen, B. Song, C. Sampson, and K. M. Golden. Network modeling of Arctic melt ponds. *Cold Reg. Sci. Technol.*, 124:40–53, 2016. ISSN 0165232X. doi: 10.1016/j.coldregions.2015.11.019.

M. F. Barnsley, D. Saupe, and H.-O. Peitgen. *The Science of fractal images*. Springer-Verlag, New York, 1988. ISBN 0387966080.

N. Bastas, K. Kosmidis, and P. Argyrakis. Explosive site percolation and finite-size hysteresis. *Phys. Rev. E - Stat. Nonlinear, Soft Matter Phys.*, 84(6):1–6, 2011. ISSN 15393755. doi: 10.1103/PhysRevE.84.066112.

B. Bollobás and O. Riordan. The critical probability for random Voronoi percolation in the plane is 1/2. *Probab. Theory Relat. Fields*, 136(3):417–468, 2006. ISSN 01788051. doi: 10.1007/s00440-005-0490-z.

B. Bowen, C. Strong, and K. M. Golden. Modeling the fractal geometry of Arctic melt ponds using the level sets of random surfaces. *J. Fractal Geom.*, 5(2), 2018.

S. R. Broadbent and J. M. Hammersley. Percolation processes. 1. Crystals and mazes. *Proc. Cambridge Phil. Soc.*, 53:629–641, 1957. doi: 10.1017/S0305004100032680.

M. Brokate and J. Sprekels. *Hysteresis and phase transitions*. Applied mathematical sciences (Springer-Verlag New York Inc.). Springer, New York, 1996. ISBN 9780387947631.

B. W. Cunliffe. *The extraordinary voyage of Pytheas the Greek*. Allen Lane, London, 2001. ISBN 9780713995091.

C. Deser, J. E. Walsh, and M. S. Timlin. Arctic sea ice variability in the context of recent atmospheric circulation trends. *J. Clim.*, 13(3):617–633, 2000. ISSN 08948755. doi: 10.1175/1520-0442(2000)013⟨0617:ASIVIT⟩2.0.CO;2.

A. Diaz, J. K. Ehn, J. C. Landy, B. G. Else, K. Campbell, and T. N. Papakyriakou. The energetics of extensive meltwater flooding of level Arctic sea ice. *J. Geophys. Res. Ocean.*, 123(12):8730–8748, 2018. ISSN 21699291. doi: 10.1029/2018JC014045.

R. M. D'Souza and J. Nagler. Anomalous critical and supercritical phenomena in explosive percolation. *Nat. Phys.*, 11(7):531–538, 2015. ISSN 17452481. doi: 10.1038/nphys3378.

R. M. D'Souza, J. Gómez-Gardeñes, J. Nagler, and A. Arenas. Explosive phenomena in complex networks. *Adv. Phys.*, 68(3):123–223, 2019. ISSN 14606976. doi: 10.1080/00018732.2019.1650450.

J. K. Ehn, C. J. Mundy, D. G. Barber, H. Hop, A. Rossnagel, and J. Stewart. Impact of horizontal spreading on light propagation in melt pond covered seasonal sea ice in the Canadian Arctic. *J. Geophys. Res. Ocean.*, 116(9):1–15, 2011. ISSN 21699291. doi: 10.1029/2010JC006908.

H. Eicken. Tracer studies of pathways and rates of meltwater transport through Arctic summer sea ice. *J. Geophys. Res.*, 107(C10):8046, 2002. ISSN 0148-0227. doi: 10.1029/2000JC000583.

P. Erdős and A. Rényi. On random graphs. *Publ. Math.*, 6:290–297, 1959.

K. Falconer. *Fractal Geometry*. John Wiley and Sons, Chichester, UK, 1990.

F. Family and T. Vicsek. *Dynamics of fractal surfaces*. World Scientific, Singapore, 1991. ISBN 9789810207205.

F. Fetterer and N. Untersteiner. Observations of melt ponds on Arctic sea ice. *J. Geophys. Res.*, 103(Cll), 1998.

D. Flocco and D. L. Feltham. A continuum model of melt pond evolution on Arctic sea ice. *J. Geophys. Res.*, 112, 2007. doi: 10.1029/2006JC003836.

D. Flocco, D. L. Feltham, and A. K. Turner. Incorporation of a physically based melt pond scheme into the sea ice component of a climate model. *J. Geophys. Res. Ocean.*, 115(8), 2010. ISSN 21699291. doi: 10.1029/2009JC005568.

D. Flocco, D. Schroeder, D. L. Feltham, and E. C. Hunke. Impact of melt ponds on Arctic sea ice simulations from 1990 to 2007. *J. Geophys. Res. Ocean.*, 117(9), 2012. ISSN 21699291. doi: 10.1029/2012JC008195.

D. Flocco, D. L. Feltham, E. Bailey, and D. Schroeder. The refreezing of melt ponds on Arctic sea ice. *J. Geophys. Res. Ocean.*, 2015. doi: 10.1002/2014JC010140.

A. S. Fors, D. V. Divine, A. P. Doulgeris, A. H. Renner, and S. Gerland. Signature of Arctic first-year ice melt pond fraction in X-band SAR imagery. *Cryosphere*, 11 (2):755–771, 2017. ISSN 19940424. doi: 10.5194/tc-11-755-2017.

J. H. Friedman. Multivariate adaptive regression splines. *Ann. Stat.*, 1991.

K. M. Golden. Critical behavior of transport in sea ice. *Phys. B Condens. Matter*, 338(1-4):274–283, 2003. ISSN 09214526. doi: 10.1016/j.physb.2003.08.007.

K. M. Golden, S. F. Ackley, and V. I. Lytle. The percolation phase transition in sea ice. *Science*, 282(5397):2238–2241, 1998. ISSN 00368075. doi: 10.1126/science.282.5397.2238.

K. M. Golden, H. Eicken, A. L. Heaton, J. Miner, D. J. Pringle, and J. Zhu. Thermal evolution of permeability and microstructure in sea ice. *Geophys. Res. Lett.*, 34 (16):2–7, 2007. ISSN 00948276. doi: 10.1029/2007GL030447.

T. C. Grenfell and G. A. Maykut. The optical properties of ice and snow in the Arctic Basin. *J. Glaciol.*, 18(80):445–463, 1977. ISSN 0022-1430. doi: 10.3189/S0022143000021122.

M. V. Guarino, L. C. Sime, D. Schröeder, I. Malmierca-Vallet, E. Rosenblum, M. Ringer, J. Ridley, D. Feltham, C. Bitz, E. J. Steig, E. Wolff, J. Stroeve, and A. Sellar. Sea-ice-free Arctic during the Last Interglacial supports fast future loss. *Nat. Clim. Chang.*, 10(10):928–932, 2020. ISSN 17586798. doi: 10.1038/s41558-020-0865-2.

G. M. Hale and M. R. Querry. Optical constants of water in the 200-nm to 200-μm wavelength region. *Appl. Opt.*, 12(3):555, 1973. ISSN 0003-6935. doi: 10.1364/AO.12.000555.

H. Han, J. Im, M. Kim, S. Sim, J. Kim, D. J. Kim, and S. H. Kang. Retrieval of melt ponds on arctic multiyear sea ice in summer from TerraSAR-X dual-polarization data using machine learning approaches: A case study in the Chukchi Sea with mid-incidence angle data. *Remote Sens.*, 8(1), 2016. ISSN 20724292. doi: 10.3390/rs8010057.

J. M. Hanesiak, D. G. Barber, R. A. De Abreu, and J. J. Yackel. Local and regional albedo observations of arctic first-year sea ice during melt ponding. *J. Geophys. Res. Ocean.*, 106(C1):1005–1016, 2001. ISSN 01480227. doi: 10.1029/1999JC000068.

C. F. C. Hawkes. *Pytheas: Europe and the Greek explorers, a lecture delivered at New College, Oxford on 20th May, 1975, revised and amplified with ten maps.* J.L. Myres memorial lecture. Blackwell, Classics Department for the Board of Management of the Myres Memorial Fund, Oxford, 1977. ISBN 9780903563079.

C. Hohenegger, B. Alali, K. R. Steffen, D. K. Perovich, and K. M. Golden. Transition in the fractal geometry of Arctic melt ponds. *Cryosph.*, 6(5):1157–1162, 2012. ISSN 19940416. doi: 10.5194/tc-6-1157-2012.

M. M. Holland, D. A. Bailey, B. P. Briegleb, B. Light, and E. Hunke. Improved sea ice shortwave radiation physics in CCSM4: The impact of melt ponds and aerosols on Arctic sea ice. *J. Clim.*, 25(5):1413–1430, 2012. ISSN 08948755. doi: 10.1175/JCLI-D-11-00078.1.

C. Horvat, D. Flocco, D. W. Rees Jones, L. Roach, and K. M. Golden. The Effect of Melt Pond Geometry on the Distribution of Solar Energy Under First-Year Sea Ice. *Geophys. Res. Lett.*, 47(4):1–10, 2020. ISSN 19448007. doi: 10.1029/2019GL085956.

W. Huang, P. Lu, R. Lei, H. Xie, and Z. Li. Melt pond distribution and geometry in high Arctic sea ice derived from aerial investigations. *Ann. Glaciol.*, 57(73): 105–118, 2016. ISSN 02603055. doi: 10.1017/aog.2016.30.

J.-h. Kim, W. Moon, A. J. Wells, J. P. Wilkinson, B. Hwang, M. A. Granskog, and D. R. Jones. Salinity control of thermal evolution of late summer melt ponds on Arctic sea ice. *Geophys. Res. Lett.*, pages 1–10, 2018. ISSN 00948276. doi: 10.1029/2018GL078077.

J. Kingslake, F. Ng, and A. Sole. Modelling channelized surface drainage of supraglacial lakes. *J. Glaciol.*, 61(225):185–199, 2015. ISSN 00221430. doi: 10.3189/2015JoG14J158.

J. Kingslake, J. C. Ely, I. Das, and R. E. Bell. Widespread movement of meltwater onto and across Antarctic ice shelves. *Nature*, 544(7650):349–352, 2017. ISSN 14764687. doi: 10.1038/nature22049.

T. Kodaira, T. Waseda, T. Nose, and J. Inoue. Record high Pacific Arctic seawater temperatures and delayed sea ice advance in response to episodic atmospheric blocking. *Sci. Rep.*, 10(1):1–12, 2020. ISSN 20452322. doi: 10.1038/s41598-020-77488-y.

Y. Kuramoto. *Self-entrainment of a population of coupled non-linear oscillators*, volume 39, pages 420–422. 1975. doi: 10.1007/BFb0013365.

R. Kwok and G. F. Cunningham. Variability of Arctic sea ice thickness and volume from CryoSat-2. *Philos. Trans. R. Soc. A Math. Phys. Eng. Sci.*, 373(2045): 20140157, 2015. ISSN 1364-503X. doi: 10.1098/rsta.2014.0157.

R. Kwok and N. Untersteiner. The thinning of Arctic sea ice. *Phys. Today*, 64:36, 2011.

R. Kwok, G. F. Cunningham, M. Wensnahan, I. Rigor, H. J. Zwally, and D. Yi. Thinning and volume loss of the Arctic Ocean sea ice cover: 2003-2008. *J. Geophys. Res. Ocean.*, 114(7):2003–2008, 2009. ISSN 21699291. doi: 10.1029/2009JC005312.

J. Landy and J. K. Ehn. Surface and melt pond evolution on landfast first-year sea ice in the Canadian Arctic Archipelago. *J. Geophys. Res. Ocean.*, pages 1365–1382, 2014. doi: 10.1002/2013JC009525.

C. M. Lee, S. Cole, M. Doble, J. D. Guthrie, S. Harper, J. Mackinnon, J. Morison, R. Musgrave, T. Peacock, L. Rainville, T. Stanton, J. Toole, J. Thomson, J. Wilkinson, M. Alford, D. Chayes, L. Freitag, S. Jayne, J.-h. Park, H. Simmons, and O. Sun. Stratified Ocean Dynamics of the Arctic: Science and Experiment Plan. Technical report, 2016.

S. H. Lee, D. A. Stockwell, H. M. Joo, Y. B. Son, C. K. Kang, and T. E. Whitledge. Phytoplankton production from melting ponds on Arctic sea ice. *J. Geophys. Res. Ocean.*, 117(4):1–11, 2012. ISSN 21699291. doi: 10.1029/2011JC007717.

J.-F. Lemieux, S. Bouillon, F. Dupont, G. Flato, M. Losch, P. Rampal, L.-B. Tremblay, M. Vancoppenolle, and T. Williams. *Sea Ice Physics and Modelling*, pages 1–9. Cambridge University Press, 2017. doi: 10.1017/9781108277600.003.

P. Lu, M. Leppäranta, B. Cheng, Z. Li, L. Istomina, and G. Heygster. The color of melt ponds on Arctic sea ice. *Cryosphere*, 12(4):1331–1345, 2018. ISSN 19940424. doi: 10.5194/tc-12-1331-2018.

M. Lüthje, D. L. Feltham, P. D. Taylor, and M. G. Worster. Modeling the summertime evolution of sea-ice melt ponds. *J. Geophys. Res. Ocean.*, 111(2):1–17, 2006. ISSN 21699291. doi: 10.1029/2004JC002818.

Y. P. Ma, I. Sudakov, C. Strong, and K. M. Golden. Ising model for melt ponds on Arctic sea ice. *New J. Phys.*, 21(6), 2019. ISSN 13672630. doi: 10.1088/1367-2630/ab26db.

B. B. Mandelbrot. *The Fractal Geometry of Nature*. WH Freeman, New York, 1983.

B. B. Mandelbrot and J. A. Wheeler. The Fractal Geometry of Nature, 1983. ISSN 0002-9505.

G. A. Maykut and N. Untersteiner. Some results from a time-dependent thermodynamic model of sea ice. *J. Geophys. Res.*, 76(6):1550–1575, 1971. ISSN 01480227. doi: 10.1029/JC076i006p01550.

G. A. Maykut, N. Untersteiner, and N. Maykut, G. A.; Untersteiner. Numerical prediction of the thermodynamic response of Arctic sea ice to environmental changes. Technical report, 1969.

J. McCannon. *A history of the arctic: nature, exploration and exploitation*. Reaktion Books, London, 2012.

M. P. Morassutti and B. F. Ledrew. Albedo and depth of melt ponds on sea-ice. *Int. J. Climatol.*, 16(7):817–838, 1996. ISSN 08998418. doi: 10.1002/(SICI)1097-0088(199607)16:7⟨817::AID-JOC44⟩3.0.CO;2-5.

M. Newman. *Networks*. Oxford University Press, 2nd edition, 2018.

A. W. Nolin and E. Mar. Arctic sea ice surface roughness estimated from multi-angular reflectance satellite imagery. *Remote Sens.*, 11(1):1–12, 2019. ISSN 20724292. doi: 10.3390/rs11010050.

D. Notz. Challenges in simulating sea ice in Earth System Models. *Wiley Interdiscip. Rev. Clim. Chang.*, 3(6):509–526, 2012. ISSN 17577799. doi: 10.1002/wcc.189.

D. Notz and M. G. Worster. Desalination processes of sea ice revisited. *J. Geophys. Res. Ocean.*, 114(5):1–10, 2009. ISSN 21699291. doi: 10.1029/2008JC004885.

D. Perovich, W. Meier, M. Tschudi, S. Hendricks, A. A. Petty, D. Divine, S. Farrell, S. Gerland, C. Haas, L. Kaleschke, O. Pavlova, R. Ricker, X. Tian-Kunze, M. Webster, and K. Wood. Arctic report card 2020: sea ice. Technical report, 2020.

D. K. Perovich, W. B. Tucker, and K. A. Ligett. Aerial observations of the evolution of ice surface conditions during summer. *J. Geophys. Res. Ocean.*, 107(10):1–14, 2002. ISSN 21699291. doi: 10.1029/2000jc000449.

D. K. Perovich, T. C. Grenfell, B. Light, B. C. Elder, J. Harbeck, C. Polashenski, W. B. Tucker, and C. Stelmach. Transpolar observations of the morphological properties of Arctic sea ice. *J. Geophys. Res. Ocean.*, 114(1), 2009. ISSN 21699291. doi: 10.1029/2008JC004892.

C. Petrich, H. Eicken, C. M. Polashenski, M. Sturm, J. P. Harbeck, D. K. Perovich, and D. C. Finnegan. Snow dunes: A controlling factor of melt pond distribution on Arctic sea ice. *J. Geophys. Res. Ocean.*, 117(9):1–10, 2012. ISSN 21699291. doi: 10.1029/2012JC008192.

C. Polashenski, D. Perovich, and Z. Courville. The mechanisms of sea ice melt pond formation and evolution. *J. Geophys. Res. Ocean.*, 117(1):1–23, 2012. ISSN 21699291. doi: 10.1029/2011JC007231.

C. Polashenski, K. M. Golden, and D. K. Perovich. Percolation blockage: A process that enablesmelt pond formation on first year Arctic sea ice. *J. Geophys. Res. Ocean.*, 122:1–28, 2017. doi: 10.1002/2016JC011994.

P. Popović and D. Abbot. A simple model for the evolution of melt pond coverage on permeable Arctic sea ice. *Cryosph.*, 11(3):1149–1172, 2017. ISSN 19940424. doi: 10.5194/tc-11-1149-2017.

P. Popović, B. B. Cael, M. Silber, and D. S. Abbot. Simple rules govern the patterns of Arctic sea ice melt ponds. *Phys. Rev. Lett.*, 120(14):1–5, 2018. ISSN 10797114. doi: 10.1103/PhysRevLett.120.148701.

P. Popović, J. Finkel, M. C. Silber, and D. S. Abbot. Snow topography on undeformed Arctic sea ice captured by an idealized "snow dune" model. *J. Geophys. Res. Ocean.*, 125(9):1–19, 2020a. ISSN 21699291. doi: 10.1029/2019JC016034.

P. Popović, M. C. Silber, and D. S. Abbot. Critical percolation threshold restricts late-summer Arctic sea ice melt pond coverage. *J. Geophys. Res. Ocean.*, 125(8): 1–16, 2020b. ISSN 21699291. doi: 10.1029/2019JC016029.

M. A. Porter and J. P. Gleeson. *Dynamical systems on networks : a tutorial*. Frontiers in applied dynamical systems. Springer, Cham, 2016. ISBN 9783319266411.

V. F. Radionov, N. N. Bryazgin, and E. I. Alexandrov. The snow cover of the Arctic Basin. *Tech. Report, APL-UW TR 9701*, (March):98, 1997.

O. Riordan and L. Warnke. Explosive percolation is continuous. *Science*, 333(6040): 322–324, 2011. ISSN 00368075. doi: 10.1126/science.1206241.

E. Roeckner, T. Mauritsen, M. Esch, and R. Brokopf. Impact of melt ponds on Arctic sea ice in past and future climates as simulated by MPI-ESM. *J. Adv. Model. Earth Syst.*, 4(9):1989–1995, 2012. ISSN 19422466. doi: 10.1029/2012MS000157.

R. J. Rolph, D. L. Feltham, and D. Schröder. Changes of the Arctic marginal ice zone during the satellite era. *Cryosphere*, 14(6):1971–1984, 2020. ISSN 19940424. doi: 10.5194/tc-14-1971-2020.

A. Rösel and L. Kaleschke. Exceptional melt pond occurrence in the years 2007 and 2011 on the Arctic sea ice revealed from MODIS satellite data. *J. Geophys. Res. Ocean.*, 117(5):1–8, 2012. ISSN 21699291. doi: 10.1029/2011JC007869.

P. Sankelo, J. Haapala, I. Heiler, and E. Rinne. Melt pond formation and temporal evolution at the drifting station Tara during summer 2007. *Polar Res.*, 29(3): 311–321, 2010. ISSN 08000395. doi: 10.1111/j.1751-8369.2010.00161.x.

A. Scagliarini and E. Calzavarini. Modelling sea ice and melt ponds evolution: sensitivity to microscale heat transfer mechanisms. *arXiv:1809.06924 [physics.ao-ph]*, 2018.

D. Schröder, D. L. Feltham, D. Flocco, and M. Tsamados. September Arctic sea-ice minimum predicted by spring melt-pond fraction. *Nat. Clim. Chang.*, 4(5):353–357, 2014. ISSN 17586798. doi: 10.1038/nclimate2203.

F. Scott and D. Feltham. A model of the three dimensional evolution of Arctic melt ponds on first year and multiyear sea ice. *J. Geophys. Res.*, 115(July):1–37, 2010. doi: 10.1029/2010JC006156.

E. D. Skyllingstad and C. A. Paulson. A numerical study of melt ponds. *J. Geophys. Res. Ocean.*, 112(8):1–20, 2007. ISSN 21699291. doi: 10.1029/2006JC003729.

E. D. Skyllingstad, C. A. Paulson, and D. K. Perovich. Simulation of melt pond evolution on level ice. *J. Geophys. Res. Ocean.*, 114(12):1–15, 2009. ISSN 21699291. doi: 10.1029/2009JC005363.

E. D. Skyllingstad, K. M. Shell, L. Collins, and C. Polashenski. Simulation of the melt season using a resolved sea ice model with snow cover and melt ponds. *J. Geophys. Res. C Ocean.*, 2015. ISSN 21699291. doi: 10.1002/2014JC010569.

D. Stauffer and A. Amnon. *Introduction to Percolation Theory*. Taylor and Francis, London, 2003. ISBN 0 7484 0253 5.

J. Stefan. Uber die Theorie der Eisbildung, insbesondere uber die Eisbildung im Polarmeere. *Ann. der Phys. und Chemie*, 42:269–286, 1891.

R. Strachan. *Contributions to our knowledge of the meteorology of the arctic regions, Part 4*. Authority of the Meteorology, London, 1885.

P. D. Taylor and D. L. Feltham. A model of melt pond evolution on sea ice. *J. Geophys. Res. C Ocean.*, 109(12):1–19, 2004. ISSN 01480227. doi: 10.1029/2004JC002361.

A. Thorndike. Sea ice thickness as a stochastic process. *J. Geophys. Res.*, 105(C1): 1311–1313, 2000. ISSN 0148-0227. doi: 10.1029/1999JC900271.

A. S. Thorndike. Estimates of sea ice thickness distribution using observations and theory. *J. Geophys. Res.*, 97(C8):601–605, 1992. ISSN 0148-0227. doi: 10.1029/92JC01199.

A. S. Thorndike, D. A. Rothrock, G. A. Maykut, and R. Colony. The thickness distribution of sea ice. *J. Geophys. Res.*, 80(33):4501, 1975. ISSN 0148-0227. doi: 10.1029/JC080i033p04501.

S. Toppaladoddi and J. S. Wettlaufer. Theory of the sea ice thickness distribution. *Phys. Rev. Lett.*, 115(14):2–5, 2015. ISSN 10797114. doi: 10.1103/PhysRevLett.115.148501.

M. A. Tschudi, J. A. Maslanik, and D. K. Perovich. Derivation of melt pond coverage on Arctic sea ice using MODIS observations. *Remote Sens. Environ.*, 112(5):2605–2614, 2008. ISSN 00344257. doi: 10.1016/j.rse.2007.12.009.

D. L. Turcotte. *Fractals and Chaos in Geology and Geophysics*. Cambridge University Press, Cambridge, UK, 1997.

N. Untersteiner. On the mass and heat budget of arctic sea ice. *Arch. fur Meteorol. Geophys. und Bioklimatologie Ser. A*, 12(2):151–182, 1961. ISSN 00666416. doi: 10.1007/BF02247491.

T. Uttal, J. Curry, M. McPhee, D. Perovich, R. Moritz, J. Maslanik, P. Guest, H. Stern, J. Moore, R. Turenne, A. Heiberg, M. Serreze, D. Wylie, O. Persson, C. Paulson, C. Halle, J. Morison, P. Wheeler, A. Makshtas, H. Welch, M. Shupe, J. Intrieri, K. Stamnes, R. Lindsey, R. Pinkel, W. Pegau, T. Stanton, and T. Grenfeld. Surface heat budget of the Arctic Ocean. *Bull. Amer. Meteor. Soc.*, 83 (February):255–276, 2002.

T. Vihma. Effects of Arctic sea ice decline on weather and climate: a review. *Surv. Geophys.*, 35(5):1175–1214, 2014. ISSN 01693298. doi: 10.1007/s10712-014-9284-0.

M. A. Webster, I. G. Rigor, D. K. Perovich, J. A. Richter-menge, C. M. Polashenski, and B. Light. Seasonal evolution of melt ponds on Arctic sea ice. *J. Geophys. Res. Ocean.*, 120:1–15, 2015. ISSN 21699291. doi: 10.1002/2015JC011030.

W. F. Weeks and W. D. Hibler. *On Sea Ice*. University of Alaska Press, Fairbanks, 2014. ISBN 9781602230798.

N. C. Wright and C. M. Polashenski. Open-source algorithm for detecting sea ice surface features in high-resolution optical imagery. *Cryosphere*, 12(4):1307–1329, apr 2018. ISSN 19940424. doi: 10.5194/tc-12-1307-2018.

J. Zhu, A. Jabini, K. M. Golden, H. Eicken, and M. Morris. A network model for fluid transport through sea ice. *Ann. Glaciol.*, 44(1):129–133, 2006. ISSN 02603055. doi: 10.3189/172756406781811141.

N. N. Zubov. *Arctic ice*. U.S. Navy Oceanographic Office and American Meteorological Society, 1943. doi: 10.5962/bhl.title.39101.