TURUN
YLIOPISTO
UNIVERSITY
OF TURKU

MATHEMATICAL MODELLING AND NUMERICAL SIMULATION OF PHYSICAL CLOUD PROCESSES IN A WIDE RANGE OF SPATIOTEMPORAL SCALES

Jaakko Ahola

TURUN YLIOPISTON JULKAISUJA – ANNALES UNIVERSITATIS TURKUENSIS
SARJA - SER. AI OSA - TOM. 704 | ASTRONOMICA - CHEMICA - PHYSICA - MATHEMATICA | TURKU 2023

TURUN
YLIOPISTO
UNIVERSITY
OF TURKU

MATHEMATICAL MODELLING AND NUMERICAL SIMULATION OF PHYSICAL CLOUD PROCESSES IN A WIDE RANGE OF SPATIOTEMPORAL SCALES

Jaakko Ahola

TURUN YLIOPISTON JULKAISUJA – ANNALES UNIVERSITATIS TURKUENSIS
SARJA - SER. AI OSA - TOM. 704 | ASTRONOMICA - CHEMICA - PHYSICA - MATHEMATICA
| TURKU 2023

University of Turku

Faculty of Science
Department of Mathematics and Statistics
Applied mathematics
Doctoral Programme in Exact Sciences

Supervisors

Professor
Marko Mäkelä
University of Turku

Research Professor
Hannele Korhonen
Finnish Meteorological Institute

Associate Professor
Tomi Raatikainen
Finnish Meteorological Institute

Research Professor
Sami Romakkaniemi
Finnish Meteorological Institute

Reviewers

Professor
Michael Boy
University of Helsinki
Lappeenranta-Lahti University of Technology

Doctor
Lindsay Lee
Advanced Manufacturing Research Centre
University of Sheffield, UK

Opponent

Professor
Jari Hämäläinen
Lappeenranta-Lahti University of Technology

The originality of this thesis has been checked in accordance with the University of Turku quality assurance system using the Turnitin OriginalityCheck service.

ISBN 978-951-29-9544-8 (Print)
ISBN 978-951-29-9545-5 (Pdf)
ISSN 0082-7002 (Print)
ISSN 2343-3175 (Online)

PunaMusta Oy, Joensuu, 2023

To my family
”It’s over. I have the high ground.”

UNIVERSITY OF TURKU
Faculty of Science
Department of Mathematics and Statistics
Applied mathematics
AHOLA, JAAKKO: Mathematical modelling and numerical simulation of physical cloud processes in a wide range of spatiotemporal scales
Doctoral dissertation, 159 pp.
Doctoral Programme in Exact Sciences
November 2023

ABSTRACT

The mathematical modelling and numerical simulation of clouds and climate include numerous phenomena that are tough nuts to crack as they cover a wide range of spatiotemporal scales. In many ways, time is a vital factor, for instance, predicting the significance of a millisecond phenomenon for the future century is a major undertaking. Additionally, the computational time required by numerical models is a challenge. Luckily, we have a fine set of tools in our mathematical backpack. Here, we explore how a detailed cloud model can be improved to simulate the interactions with ice crystals. A new ice microphysics module is validated against a set of similar cloud models. Further on, the cloud model is shown to be an improvement over the previous generation of cloud models as it incorporates detailed aerosol-cloud interactions, which in our study is shown to impact cloud lifetime through ice nuclei recycling and marine ice nuclei import via updrafts. Additionally, the cloud model, which has a fine resolution in the order of meters, is harnessed to develop three different emulators to represent selected cloud processes in an improved detailed way. Emulators can be called also parametrisation or a machine learning model. Further on, created parameterisations are implemented within a global climate model, which has a much coarser resolution in the order of 10–100 kilometres. The implementation enables more precise climate simulations by having a more detailed subgrid scale description of cloud processes. As an adventurous side quest, we elaborate on how the proof-of-concept emulator could be embellished by showing an optimised way of creating the design of the simulation experiment in our applied case and we compare our results with the proof-of-concept method used in the study where the emulators were created.
KEYWORDS: cloud modelling, climate modelling, simulation, machine learning, parameterisation, design of experimentation, optimisation, large-eddy simulation, ice microphysics

TURUN YLIOPISTO
Matemaattis-luonnontieteellinen tiedekunta
Matematiikan ja tilastotieteen laitos
Sovellettu matematiikka
AHOLA, JAAKKO: Fysikaalisten pilviprosessien matemaattinen mallinnus ja numeerinen simulointi laajassa mittakaavassa
Väitöskirja, 159 s.
Eksaktien tieteiden tohtoriohjelma
Marraskuu 2023

TIIVISTELMÄ

Ilmaston ja pilvien matemaattisessa mallinnuksessa ja numeerisessa simuloinnissa on monia pähkinöitä purtavaksi, sillä niihin liittyy useassa eri avaruudellisessa ja ajallisessa mittakaavassa tapahtuvia ilmiöitä. Aika on monella tavalla kriittinen tekijä, koska esimerkiksi millisekunneissa tapahtuvan ilmiön merkityksen ennustaminen kuluvan vuosisadan ilmastolle on hankalaa. Samalla myös numeeristen mallien vaatima laskenta-aika on merkittävä haaste. Onneksi matemaattisesta työkalupakista löytyy toimivia työkaluja helpottamaan työkuormaa. Tässä tutkimuksessa esitämme, miten käyttämäämme pilvimallia voidaan parantaa, jotta sillä pystytään mallintamaan pilvissä tapahtuvia ilmiöitä myös jääkiteiden osalta. Osoitamme pilvimallin toimivuuden ja näytämme sen sisältävän parannuksia edelliseen pilvimallisukupolveen nähden, sillä malli kykenee käsittelemään aerosoli-pilvi-vuorovaikutuksia aiempaa yksityiskohtaisemmin. Mallin yksityiskohtaisuuden lisäyksen ansiosta voimme selvittää esimerkiksi, miten jääkiteet tai tuulen merestä nostattama merisuola vaikuttavat pilvien elinaikaan. Lisäksi resoluutioltaan metrien tai kymmenien metrien luokkaa oleva pilvimallimme valjastetaan luomaan valituista pilviprosesseista kolme erilaista emulaattoria, joita voidaan kutsua myös parametrisaatioiksi tai koneoppiviksi malleiksi. Koska uudet parametrisaatiot perustuvat yksityiskohtaiseen pilvimalliin, ne ovat aiemmin käytettyjä parametrisaatioita tarkempia. Uusia emulaattoreita hyödynnetään karkeamman resoluution globaalissa ilmastomallissa, jonka resoluutio on kymmenien tai jopa satojen kilometrien luokkaa. Tällainen parametrisaation toteutus ilmastomallissa mahdollistaa aiempaa paremmat ilmastosimulaatiot, sillä pilvien alihilaprosessit on kuvattu aikaisempaa tarkemmin. Lopuksi eräänlaisena sivutehtävänä tutkimme, miten esiteltyä emulaattorien luomisessa käytettyä simulaatioiden alkuarvojoukkoa voitaisiin edelleen optimoida. Saatuja optimoituja tuloksia verrataan edellä luotuihin emulaattorien alkuarvojoukkoihin.
ASIASANAT: mallinnus, simulointi, parametrisaatio, koneoppiminen, suurten pyörteiden menetelmä, simulaatioiden alkuarvojoukon optimointi, optimointi, pilvet, ilmasto, jäämikrofysiikka

Acknowledgements

This thesis is a product of participating in the research project: ”Emulation of subgrid-scale aerosol-cloud interactions in climate models: towards a realistic representation of aerosol indirect effect” (ECLAIR) that was funded by the European Research Council and was conducted by the Finnish Meteorological Institute during years 2015-2020 at the Climate System Modelling group.

First, I want to thank my supervisors Hannele, Tomi, Sami and Marko for their long-term support in this process. I am thankful for Professor Michael Boy and Doctor Lindsay Lee for reviewing this thesis with favourable statements. I feel privileged and grateful to Professor Jari Hämäläinen for agreeing to act as my opponent. I want to acknowledge and thank Professor Emeritus Hannu Aro for the mentorship.

I want to thank my former superiors Hannele, Antti-Ilari and Joonas for their support and openness to discuss about any issues. I am thankful for Kalle and Erika for their peer support. I also want to thank my FMI colleagues Laura, Ege, Jia, Petri, Marje, Jukka-Pekka, Harri, Antti K., Antti L., and Juha, with whom I have learned about science. Especially, I want to show my appreciation to Tommi B. for many shared coffees particularly during COVID-times; Svante for the laughs, discussions, honest peer support and empirical medieval studies in the staircase of Dynamicum; Anders for friendship, support, and shared sailing miles; and Anca, Giulia and Joni-Pekka for shared and alternating shenanigans and many laughs in and out of Star Wars universe. I also want to thank numerous FMI colleagues who make the FMI a diverse and professional institute and who made my time at the FMI pleasant and sometimes made me laugh to tears.

I express my thanks to my upper comprehensive school mathematics teacher Liisa Ranki for encouraging me and pushing me forward in the world of mathematics. To which she might say something similar to what Gandalf said about Bilbo: “If you’re referring to the incident with the Dragon, I was barely involved. All I did was give your uncle a little nudge out of the door.” Thank you for the nudge that made a big difference.

I am grateful to my friends, especially Vesa and Antti P. for their support, discussions, and friendship. I would like to express my gratitude for my brother and my parents for their long-term support. Specifically, I would like to mention the support that my parents, parents-in-law, and sisters-in-law provided as they took care our little kids when I was finalising this thesis.

Lastly, I am grateful to my wife Hanna for her solid and persistent support through the years, tears, and victories. Without your support, I would have never reached the finish line. But now, *we* have reached it and comes a time when I am not writing my thesis.

November 2023
Jaakko Ahola

Table of Contents

Abbreviations

ALGR	Adaptive Lattice Grid Refinement
BSP	Binary Space Partition
CCN	Cloud Condensation Nucleus/Nuclei
CDNC	Cloud Droplet Number Concentration
CoMinED	Constrained Minimum Energy Design
ECHAM6	a global general circulation model (Stevens et al., 2013)
ECHAM	in this study referred to ECHAM-HAMMOZ
ECHAM-HAMMOZ	a comprehensive 3-dimensional chemistry-climate model (Schultz et al., 2018), including ECHAM6, HAM and MOZART
GCM	General Circulation Model
GP	Gaussian Process
GPE	Gaussian Process Emulator
HAM	Hamburg Aerosol Model, an aerosol chemistry and microphysics package (Zhang et al., 2012)
LES	Large-Eddy Simulation/Simulator
LFRF	Linear Fit improved with Random Forest
LHS	Latin Hypercube Sampling
LWP	Liquid Water Path
MCMC	Markov Chain Monte Carlo
MinED	Minimum Energy Design
MOZART	Model of Ozone and Related Chemical Tracers, an atmospheric trace gas chemistry model (Rast et al., 2014)
SALSA	Sectional Aerosol module for Large Scale Applications, a bin microphysics scheme (Kokkola et al., 2008)
SB	Seifert & Beheng, a bulk microphysics scheme (Seifert and Beheng, 2006)
SCMC	Sequentially Constrained Monte Carlo
SMC	Sequential Monte Carlo
SGS	SubGrid-Scale
UCLALES	University of California Los Angeles Large Eddy Simulator, an LES model (Stevens et al., 1999, 2005)
UCLALES-SALSA	UCLALES coupled with SALSA microphysics scheme, a LES model (Tonttila et al., 2017)

List of Original Publications

This dissertation is based on the following original publications, which are referred to in the text by their Roman numerals:

I Ahola, J., Korhonen, H., Tonttila, J., Romakkaniemi, S., Kokkola, H., and Raatikainen, T.: Modelling mixed-phase clouds with the large-eddy model UCLALES–SALSA, Atmospheric Chemistry and Physics, 20, 11639–11654, https://doi.org/10.5194/acp-20-11639-2020, 2020.

II Raatikainen, T., Prank, M., Ahola, J., Kokkola, H., Tonttila, J., and Romakkaniemi, S.: The effect of marine ice-nucleating particles on mixed-phase clouds, Atmospheric Chemistry and Physics, 22, 3763–3778, https://doi.org/10.5194/acp-22-3763-2022, 2022.

III Ahola, J., Raatikainen, T., Alper, M. E., Keskinen, J.-P., Kokkola, H., Kukkurainen, A., Lipponen, A., Liu, J., Nordling, K., Partanen, A.-I., Romakkaniemi, S., Räisänen, P., Tonttila, J., and Korhonen, H.: Technical note: Parameterising cloud base updraft velocity of marine stratocumuli, Atmospheric Chemistry and Physics, 22, 4523–4537, https://doi.org/10.5194/acp-22-4523-2022, 2022.

IV Nordling, K., Keskinen, J.-P., Romakkaniemi, S., Kokkola, H., Räisänen, P., Lipponen, A., Partanen, A.-I., Ahola, J., Tonttila, J., Alper, M. E., Korhonen, H., and Raatikainen, T.: Technical note: Emulation of a large-eddy simulator for stratocumulus clouds in a general circulation model, EGUsphere [preprint], https://doi.org/10.5194/egusphere-2023-912, 2023.

1 Introduction

It has been said that the most interesting scientific research is at the intersection of different branches of science. At that intersection, the most rewarding inventions and the hardest problems are solved. Like in space exploration rocket scientists, theoretical physicists, astrophysicists, astrobiologists, mathematicians, psychologists, physicians, engineers, computer scientists, etc. are needed. Similarly, to tackle one of the largest threats, that is climate change, humankind has to face, a large variety of experts are required.

Based on the most recent scientific research, the IPCC (Intergovernmental Panel on Climate Change) report by Masson-Delmotte et al. (2021) provides an extensive assessment of the current state of climate change, including its impacts and potential future scenarios. It aims to inform policymakers and the public, guiding in decision-making to mitigate climate change and adapt to its effects. The IPCC report documents comprehensively the scientific information on global warming, radiative forcing and the underlying uncertainties of aerosols, ice crystals and clouds affecting climate. To answer these issues expressed by IPCC, we lay out the basis of this study.

The main objectives of this thesis:
This thesis aims to contribute to understanding the climate system by answering the following research questions.

Q1. What mathematical tools are useful to describe the complex phenomena of aerosol-cloud interactions that affect climate in multiple ways?

Q2. In the context of climate system models, can the uncertainty related to aerosol-cloud interactions be decreased?

Q3. Can the climate system models be improved by implementing machine learning methods for modelling cloud processes?

This study is in itself a mix of different branches of science put to practice where the application of mathematical tools is at the heart of all examinations (**Q1**). Since the scale of the climate phenomena as a whole is immense, they cannot be accommodated within the confines of a laboratory. Also, we have only one planet Earth that is impossible to expose to controlled experiments. Though, laboratory experiments of selected phenomena are plausible, like studying droplet freezing within a cloud chamber. Thus, mathematically abstract numerical laboratories need to be created.

There is a large variety of such laboratories in varying scales, like global (GCM), regional, or zero-dimensional, that are used to do controlled numerical experiments. These studies are needed to untangle the consequences of the uncontrolled experiments that have been going on in the climate since the early days of industrialisation (Kaper and Engler, 2013).

Climate, weather and cloud modelling is a relatively new scientific field as the related phenomena are so complex that studying climate and weather would not exist without modern computers. The first general circulation model that had both oceanic and atmospheric processes was developed in the late 1960s at the NOAA Geophysical Fluid Dynamics Laboratory (National Oceanic and Atmospheric Administration). Climate models have been developed to be more precise along with specialised models, like cloud models, ocean, and pollution advection models that have emerged to examine the relevant phenomena.

The second research question (**Q2**) focuses more on the perspective of natural sciences. Clouds, aerosols and ice crystals interact with the climate by several effects and feedback loops that can have either a warming or cooling effect on the climate. In climate models, the relevant interaction processes are usually highly parameterised due to their nature of high level of detail and computational cost. Hence, the aerosol-cloud interactions are poorly represented.

Large-eddy simulation (LES) models are a common way of studying cloud-scale phenomena. However, these cloud-scale models often lack the details for interacting with aerosols. UCLALES-SALSA is an LES cloud model that is used to model detailed aerosol-cloud interactions. UCLALES-SALSA is the crucial cornerstone of every modelling aspect in the studies presented here. To answer the research question, in this study UCLALES-SALSA is improved by implementing microphysics related to ice crystals that engage in aerosol-cloud-ice interactions.

The third research question (**Q3**) emerges from the fact that global climate models have coarse resolution in both spatial and temporal scales. Thus, cloud processes, which act even in sub-meter scales, are poorly represented. Coarse-resolution emerges from computational limitations if only traditional finite element methods are used to solve relevant partial differential equations. Computational limitations are emphasised if detailed cloud process descriptions would be used in long climate predictions. To circumvent the limitations of finite element methods, in this study, we employ machine learning methods to bridge the gap between detailed LES models and coarse GCMs. That means that we create new more accurate cloud process parameterisations, with the leverage that detailed UCLALES-SALSA provides, to be used in GCMs.

All these models have benefited from mathematical developments, such as more accurate finite element methods to solve complex partial differential equations, and increasing computational performance. However, first and foremost new mathematical tools and methods, like our novel way of implementing machine learning, are

necessary as we cannot solely trust Moore's law about ever-increasing computational power since eventually, the law will meet the boundaries of physical reality. This includes current technologies such as massively parallel systems such as GPUs. However, transforming the existing CPU-based models into GPU architecture is not a straightforward task. Technologies that could solve the problem, like quantum computing, are not yet technically feasible. Additionally, future prospects and expenses of novel technologies are uncertain and debatable.

In this study, we will discuss the mathematical choices that have been made to better understand aerosol-cloud interactions within the climate system. Additionally, the most problematic areas related to cloud modelling are addressed and what improvements could be made to the cloud models. We begin by giving relevant and further elaborated background in Chapter 2 to understand the baseline of the study. Chapter 3 focuses on **Paper I** and **Paper II** providing details on the cloud-scale model UCLALES-SALSA and how it was improved by implementing ice microphysics. Chapter 4 specifies the details of a newly applied mathematical tool of machine learning in cloud process modelling. This tool is first described in **Paper III** and further on applied in **Paper IV**. As a side quest, in Chapter 5 we provide a mathematical optimisation tool for a possible improvement of the created machine learning method by giving details for an optimised design of simulation experiments. Here we stand on the shoulders of giants, including Isaac Newton who described laws of motion by using differential equations, which are the foundation of every climate or cloud model.

2 Climate, clouds and how to model them

In this chapter, we will have an overlook of climate and clouds and how to model them. This is done to make the study easier to understand as we are putting our study in a larger context, sort of an extended introduction.

2.1 Climate

Weather represents the state of the atmosphere which can be described by the status of temperature, precipitation, cloudiness and other atmospheric conditions. Weather and climate are closely connected. Climate can be defined as statistics of weather or as the mean state of the climate system. Following, climate change refers to changes in the statistics of weather over time. To eliminate local variations and the random nature of weather, the climate is typically calculated by averaging weather data over 30 years. In other words, climate is the expectation but weather is what we get (Kaper and Engler, 2013).

Climate system consists of five components: the *atmosphere*, the *hydrosphere* (oceans, lakes and other bodies of water), the *cryosphere* (ice and snow), the *lithosphere* (land surface) and the *biosphere* (all living things) (Kaper and Engler, 2013). All these components are in constant interaction either directly or indirectly. The climate system as a whole is mainly powered by *solar radiation* and a little by geothermal heat. The state of the climate system evolves through its own dynamics in atmospheric circulation, ocean currents and other processes in all five components. Moreover, there are *climate forcings* or *climate drivers*, that impact the climate system. As per the fundamental principles of thermodynamics, when the Earth gains energy from the Sun, some of the radiation will reflect back (about 30%) and some of the energy will be absorbed by the planet. As a warm object in cold space, Earth will radiate heat out to space. This difference between the amount of incoming and outgoing radiation is called the planet's radiative forcing (RF) (NOAA predicting climate). Currently, more heat is coming in than coming out, the climate is warming. Climate forcings can be divided into human-induced (*anthropogenic*) and natural phenomena. Human-caused forcings can be changes in atmospheric composition (e.g. greenhouse gas emissions) and land use (e.g. deforestation) affecting albedo (how different surfaces reflect radiation). Natural climate forcings include volcanic eruptions, solar output variations (e.g. sunspot cycle), cyclical changes of

Earth's orbit (Milankovitch cycles), changes in albedo (like cloud coverage) and oscillations (quasi-periodic changes in surface pressure and sea surface temperature, e.g. El Niño) (Kaper and Engler, 2013). In conclusion, the climate system is highly complex and consists of a wide range of spatiotemporal scales from nanometres to thousands of kilometres and from microseconds to thousands of years therefore is not a simple task to model mathematically.

2.1.1 Modelling climate

To model the climate and its components, one needs to have variables describing the state of the system and the governing rules describing the evolution of the states. State variables include, for example, air temperature, air pressure, winds, humidity, aerosol and trace gases composition, the strength of ocean currents, rate of evaporation from vegetation cover, land use and vegetation patterns. Governing laws include laws of motion, chemical reaction laws and phase change laws, especially for water (Kaper and Engler, 2013). Laws and states are formulated into the language of mathematics usually in the form of a system of differential equations and often dividing the planet into a 3-dimensional discrete grid. The translation to mathematics is not necessarily well executed for all the components of the climate system.

The equations resulting from this process can often be extremely complex and may span several pages, making it nearly impossible to solve them analytically (i.e. with exact methods). In such cases, the best approach is to use numerical methods. This means to have a numerical solver integrating the differential equations. The solver is usually a Finite Element Method (FEM) based on Euler, Runge-Kutta or in this study the Leapfrog method. Instead of calculations done by hand, to help out with the numerous computations following from using a numerical solver, the model is written as computer software. The model is solved with desired initial and boundary conditions on a high-performance computer. Further on, the simulation with respect to the inputs is analysed. This approach allows for studying clouds in different conditions.

As this process is imitating a real-world process (e.g. cloud processes) or system (e.g. climate system) over time it is called a simulation (Banks et al., 1995). Simulations require using models. The model represents the key characteristics or behaviours of the selected system or process. On the other hand, the simulation represents the evolution of the model state over time (Wikipedia, f).

In this study, we have used the global climate model ECHAM-HAMMOZ and the cloud model UCLALES-SALSA. ECHAM-HAMMOZ can be used for example for climate predictions. The motivation to use UCLALES-SALSA, a large-eddy simulation (LES) model with a detailed aerosol description (SALSA), is that it can be used to model low-level clouds (not limited to) along with aerosol-cloud interactions. Following, the obtained UCLALES-SALSA simulation knowledge can used

to improve the coarser resolution model ECHAM-HAMMOZ that has known deficiencies. As UCLALES-SALSA is our primary simulation tool, Chapters 3 and 4 are dedicated to it. Here, we shortly introduce ECHAM-HAMMOZ.

2.1.2 Climate model ECHAM-HAMMOZ

ECHAM-HAMMOZ is one example of a climate model and it has been used in **Paper III** and **Paper IV**. To get an idea of how complex the model is, it consists of several hundreds of thousands of lines of code and equations that govern the model and fill some hundreds of manual pages. The most recent version of ECHAM-HAMMOZ model (ECHAM6.3-HAM2.3-MOZ1.0) consists of the latest versions of ECHAM6, an atmospheric general circulation model (Stevens et al., 2013), the HAM2, that is Hamburg Aerosol Model, an aerosol chemistry and microphysics package (Zhang et al., 2012), and the MOZ1 (= MOZART, Model of Ozone and Related Chemical Tracers), an atmospheric trace gas chemistry model (Rast et al., 2014). The description of the coupled ECHAM6–HAMMOZ model is provided in Schultz et al. (2018).

The atmospheric module ECHAM6 of the ECHAM-HAMMOZ computes winds, heat and mass transfers, radiative effects, relative humidity and cloud formation within each grid cell and evaluates interactions with neighbouring points. The dynamical part (i.e. winds, temperature, pressure) of ECHAM6 is formulated in spherical harmonics and each function defined on the surface of a sphere can be written as a sum of these spherical harmonics. The grid of ECHAM6 is a Gaussian grid where grid points at a given latitude are equally spaced, and grid points at a given longitude are unequally spaced. The spacing is defined by Gaussian quadrature. The horizontal resolution (= grid size) of ECHAM-HAMMOZ is in the order of hundred kilometres and the vertical resolution is in the order of kilometres. Like other climate models, the resolution has been improving significantly during its development history.

The aerosol module HAM includes processes like aerosol nucleation (= formation of small particles) and computation of emissions for sea salt and mineral dust. The MOZ1 trace gas chemistry model consists of a number of chemical reactions in the order of hundreds for tens of species.

2.2 Clouds

Clouds are a critical component of the climate system, and various aspects of clouds are the primary focus of interest in all the **Papers I, II, III, IV**. Clouds have a well-known impact on the hydrological cycle and the atmospheric radiation balance. The latter means that as clouds' albedo is high (they are visibly white), they reflect a lot of solar radiation, and as global cloud coverage is about 0.68 (Stubenrauch et al., 2013), their cooling effect is crucial for regulating the Earth's energy balance. Simultaneously, clouds have a warming effect by trapping some outgoing thermal radiation.

Therefore, clouds can alter the temperature of the climate. Cloud properties might also change in the warming atmosphere, and this effect, the net cloud feedback is estimated to be positive with the estimate of 0.42 ($W\ m^{-2}\ °C^{-1}$). Additionally, the net cloud feedback is considered to be negative with a very low probability (IPCC, 2021). The potential role of the cloud feedback for future climate estimates can be understood when compared with the total greenhouse gas effective radiative forcing (ERF), which is 3.317 $\pm$ 0.278 ($W\ m^{-2}$) from 1750 to 2019 (IPCC, 2021).

ERF takes into account both the instantaneous radiative forcing (IRF), which represents the initial change in energy balance caused by a specific driver, and the adjustments or feedbacks that occur within the climate system in response to that initial forcing.

Typically in the lower atmosphere clouds form as warm (= warmer than the environment) and moist (= containing water vapour) air parcels rise. As these air parcels ascend, they cool down due to the expansion of the air without heat exchange with their surroundings (adiabatic expansion). Ultimately, they reach lifting condensation level (LCL) where relative humidity (RH) exceeds 100% (= supersaturation). At this stage, the excess water vapour condenses onto available aerosols acting as cloud condensation nuclei (CCN) causing a sub-population of them to grow up to a size of a few micrometres at the cloud base. This process is called cloud activation or droplet activation. It should be noted that condensation occurs also at lower relative humidities, however, in the case of cloud activation, the condensation is notably more pronounced and vigorous. One of the most important factors of cloud formation is the rising or falling of the air parcel and it is affected by temperature differences in the atmosphere. That is referred to as the vertical temperature profile, which is influenced by the dynamic atmospheric behaviour ensuing from winds, solar heating and radiative cooling (Seinfeld and Pandis, 1998; Jacobson, 2005). Clouds are also affected directly or indirectly by surface fluxes (e.g. sensible and latent heat fluxes), advection and subsidence. The complex relationship between aerosols and clouds, referred to as aerosol-cloud interactions, is apparent in cloud formation.

The cloud droplets can grow by condensation or by coagulation. In condensation, water vapour condenses on liquid cloud droplets. In coagulation, the droplets collide with each other. Rain formation occurs when a sub-population of cloud droplets reaches a size that is sufficiently large to initiate gravitational settling. This settling enhances the collision rate between droplets, leading to an acceleration in droplet growth. Cloud droplets can become smaller or even dissipate through evaporation, which can happen for example if cloud droplets are mixed with dry air at the edge of the cloud, or when the drizzling droplet evaporates below the cloud.

Aerosols are tiny liquid or solid particles mixed in the air. Clouds and aerosols are strongly connected through aerosol-cloud interactions. Aerosols and cloud droplets impact the climate system through direct and indirect effects on the radiation budget (see Appendix 7.2.3). The direct effect means that aerosols scatter and absorb in-

coming solar radiation (Charlson et al., 1992). The indirect effect means that some particles can act as initial formation sites for cloud droplets (CCN) and ice crystals (ice nuclei, IN) and thereby affect the microphysics, dynamics, radiative properties and lifetime of clouds (Albrecht, 1989; Twomey, 1991; Stevens and Feingold, 2009).

As aerosols affect clouds, the clouds can also alter the atmospheric aerosol population via several processes. Clouds affect atmospheric aerosol population via cloud processing as aerosols are exposed to mixing, evaporation, condensation, coagulation and coalescence (concerns larger droplets) that alter aerosol properties and characteristics. Aqueous-phase chemical reactions change aerosol properties as specific particles can form and aerosol populations can evolve in a moist and chemically active environment (Ervens et al., 2008; Korhonen et al., 2008; Kulmala et al., 2013). Due to these processes, as clouds evaporate the composition and size of the released particles can differ distinctly from those of the original CCN population. Additionally, wet scavenging removes aerosol particles including CCN within and below a cloud, and is the most important removal mechanism of atmospheric particles. Furthermore, turbulent or convective updrafts transport aerosol particles between atmospheric layers and hence affect their role and radiative effects in the atmosphere. The updraft mechanism is one of the focal points in all the papers, especially in **Paper II** and **Paper III**.

As the interactions are complex and spatially varying, it is challenging to constrain the aerosol indirect effect. Satellite data presents good spatial coverage but it suffers from several detection uncertainties (Quaas et al., 2010; Arola et al., 2022). In addition, contrary to detailed cloud simulations and observations (Small et al., 2009; Christensen and Stephens, 2011) global climate models are prone to predict a higher increase in water content with increasing aerosol concentration compared to observations, which significantly affects the predicted cloud radiative properties. Due to these shortcomings, the aerosol indirect effect holds the single largest uncertainty in present estimates of radiative forcing (Masson-Delmotte et al., 2021). This large uncertainty in the past and present aerosol cooling narrows our understanding of the sensitivity of the climate system to an increase in carbon dioxide, which influences future climate predictions by imposing significant uncertainties (Andreae et al., 2005). Through the CMIP6 era (1850–2014), greenhouse gases have contributed the most significant positive radiative forcing, while aerosols have possessed the largest negative forcing. However, the global trend of increasing aerosol forcing has shifted towards a decreasing trend (Bauer et al., 2022).

2.2.1 Stratocumulus clouds

Stratocumulus clouds, especially marine stratocumuli are the focal point of this study concerning different cloud types (see Figure 1). Stratocumuli cover vast regions, approximately one-fifth of Earth's surface in the annual mean, 23% of the ocean

surface and 12% of the land surface. Concerning area coverage stratocumulus is the most dominant cloud type (Warren et al., 1986, 1988; Hahn and Warren, 2007; Wood, 2012). These two facts along with that stratocumuli are feasible to model with our detailed cloud model UCLALES-SALSA make stratocumuli the cloud type to study in this research. Further information about stratocumulus clouds can be found in Wood (2012).

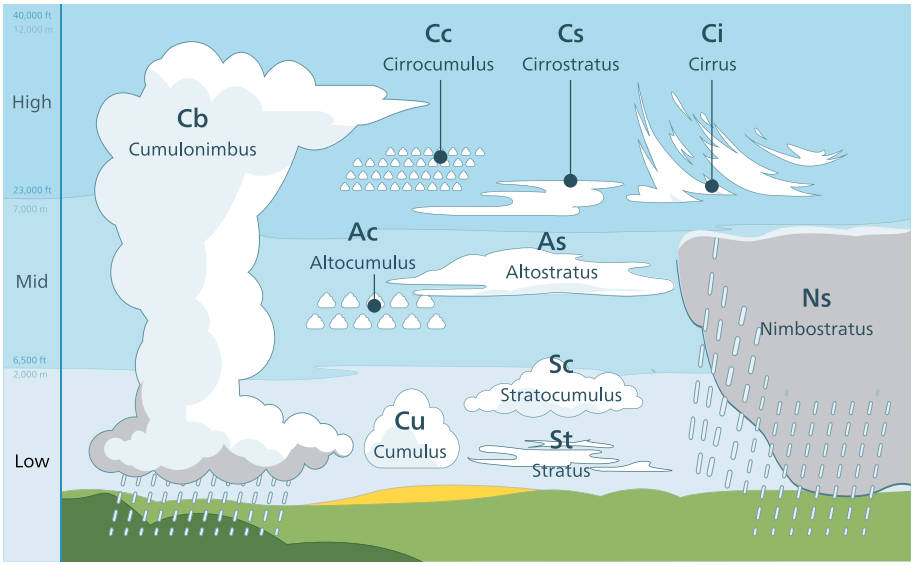

Figure 1. Figure credit: Valentin de Bruyn / Coton. This illustration has been created for Coton, the cloud identification guide for mobile. Figure reprinted under CC BY 3.0 licence (`https://creativecommons.org/licenses/by/3.0/`), via Wikimedia Commons.

The name stratocumulus originates from Latin, where *stratus* means ”layer” and *cumulus* means ”heap”. Stratocumulus forms up of an ensemble of individual convective elements that together compose a layered pattern (Wood, 2012). The layering is often obtained by a strong temperature inversion of only some ten meters thick that acts as an upper boundary. The heaping displays the convective nature of the cloud. Stratocumuli are often low-level and shallow clouds. The dynamics of stratocumuli are driven by convective instability (See Appendix 7.2.4) caused by cloud-top radiative cooling which separates stratocumulus from stratus by definition (Wood, 2012). Moreover, surface sensible heat flux is a much weaker source of turbulence compared to cloud-top radiative cooling. However, over land and in cold-air outbreaks, surface sensible heat flux can be as significant as cloud top radiative cooling (Atkinson and Wu Zhang, 1996; Wood, 2012)). Surface latent heat flux provides the main source of moisture in most stratocumuli (Wood, 2012).

The layer of stratocumuli can exhibit complex but recognisable mesoscale (horizontal scale from five to hundreds of kilometres) structures (Wood, 2012), which

can be classified typically to four categories: No MCC (MCC = mesoscale cellular convection), Closed MCC, Open MCC, Cellular but disorganised. To illustrate, No MCC can resemble a ploughed field, Closed MCC seem to be like soap bubbles squished next to each other, Open MCC can be thought of as a honeycomb and often they are even somewhat hexagonal, Cellular but disorganised refer to a group of convective cells that do not display any distinctive spatial pattern. These morphological structures can be seen in Figure 2 along with solid stratus, clustered cumulus and suppressed cumulus. The circled portion of the solid stratus resembles the structure of No MCC. Mohrmann et al. (2021) added clustered cumulus and suppressed cumulus to capture more cloud morphological variability in the tropics and subtropics.

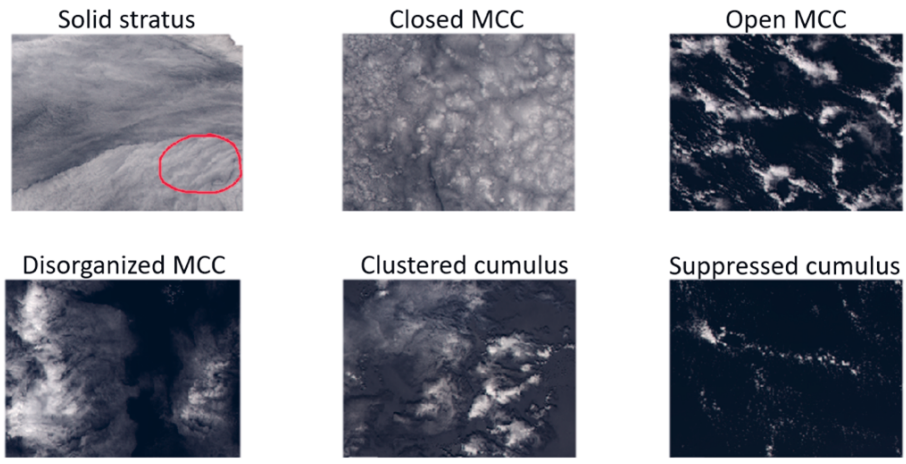

Figure 2. Cloud structures, image scale about 100 kilometres (Mohrmann et al., 2021). Figure reprinted under CC BY 4.0 licence (`https://creativecommons.org/licenses/by/4.0/`) with an added circle in stratus subfigure.

How to model clouds is the major subject of this study and is handled in detail in the following chapters.

2.2.2 Mixed-phase clouds and ice microphysics

One of the focus areas of **Paper I** and **Paper II** is the role of ice crystals in clouds as liquid-phase cloud processes are fairly well quantified but the ice microphysical processes, especially heterogeneous ice nucleation, dynamics and radiative effects of mixed-phase and ice clouds remain more inadequately constrained. When supercooled liquid droplets co-exist with ice crystals, the clouds formed are known as mixed-phase clouds. These clouds are most frequent at temperatures between -10 to $-25°$ (Filioglou et al., 2019) but can exist between -35 to $0°$ and require certain microphysical and dynamical conditions, like supersaturation with respect to ice, presence of ice nucleating particles along with updraft and turbulent mixing (An-

dronache, 2017). The role of ice in clouds can be significant as for instance, it can affect their lifetime and radiative properties. Cloud lifetime can be substantially reduced as a critical amount of ice can lead to cloud glaciation and dissipation (Rauber and Tokay, 1991; Harrington et al., 1999; Avramov and Harrington, 2010).

Ice crystals can form (= nucleate) either by homogeneous or heterogeneous freezing. In heterogeneous nucleation, freezing begins from the surface of the ice nucleating particle (INP) and can happen at higher temperatures than homogeneous nucleation (= freezing without INPs) which occurs at temperatures lower than $-38°$. Heterogeneous nucleation can further be categorised into immersion, deposition, contact and condensation freezing. These droplet freezing processes are not yet fully quantified despite extensive research (Phillips et al., 2008; Atkinson et al., 2013; DeMott et al., 2011; Kiselev et al., 2017; Iwata and Matsuki, 2018; Chatziparaschos et al., 2023). However, understanding of these processes grows iteratively. For instance, our study simulates droplet freezing processes while tracking the evolution of aerosol distribution and cloud dynamics.

2.3 The range of spatiotemporal scales

One key point in this study is the wide range of spatiotemporal scales in climate and cloud modelling. This challenge of scales emerges from the fact that for example clouds contain phenomena from microphysics of nanometre scale to cloud activation on a scale of micrometres and model resolutions of large-eddy simulation (LES) (see Chapter 3) subgrid dynamics ($< 10(m)$) where clouds and surrounding environments mix to global cloud model subgrid dynamics of mesoscale meteorology of hundreds of kilometres. The objective is to model and parameterise the physical processes that span over multiple orders of magnitude and to obtain physically representative results in a computationally feasible way.

For instance, although LES is highly detailed in terms of cloud dynamics, it cannot provide atmospherically representative results in cases where weather systems grow larger than what is feasible to simulate with the model. For example in Diamond et al. (2022), they showed that LES-scale modelling does not show full response for cloud changes. Nonetheless, at least a regional atmospheric model (of a larger length scale) is needed to obtain the effect of aerosols on atmospheric circulation. Furthermore, concerning microphysics, every single particle should fundamentally be simulated. Yet, the solutions that are feasible, like bulk and bin microphysics schemes (see Section 3.2.2), are essentially simplifications or parameterisations where an ensemble of similar particles is simulated instead of every single particle. These simplifications do not take into account that the largest droplets are not evenly distributed in turbulent air. However, the distribution of large droplets is accounted for to the extent permitted by the resolution of the model but is assumed to be even. Also Honnert et al. (2020) show that in atmospheric boundary

layer modelling in the resolution regime of a few hundred meters (*gray zone*) neither the techniques of high-resolution atmospheric modelling (a few tens of meters resolution) nor meteorological models (a few kilometres resolution) are convenient to solve turbulence structures as fundamental assumptions behind the parameterisations are violated. However, Honnert et al. (2020) admit that model simulations in this resolution regime of a few hundred meters may remain highly useful. Note well that the concern expressed by Honnert et al. (2020) does not concern our LES results where mostly all turbulent scales are solved. This is highlighted as care must be taken to ensure a parameterisation employed with different grid resolutions is suitable for its use. Essentially, a higher resolution can be used with any parameterisation, however, subgrid processes need to be properly handled. For instance, the microphysical packages in weather prediction models are altered as the resolution is changed.

The other part of the challenge is the range of temporal scales. Like in GCM, a typical time step is 10 minutes or more which can be compared to cloud activation of tens of seconds to minutes. In the detailed LES model, the time step is usually a few seconds, however for example the modelled water vapour condensation is calculated within a sub-time step process.

These discrepancies between realistic spatiotemporal scales and the limitations of feasible modelling lead to process parameterisations that can be inadequate representations of physical reality. Some processes or interactions might be missing altogether, especially in coarser models like GCMs. Thus, research is needed to further narrow down the gap between physical reality and climate system models.

3 Cloud Modelling with UCLALES-SALSA: methods and results

Clouds are a vital part of the climate system, as stated in Chapter 2, and since significant uncertainties are related to aerosol-cloud interactions we wanted to develop a state-of-the-art cloud model to quantify these interactions. In **Paper I** and **Paper II** a detailed cloud model UCLALES-SALSA was used to examine the aerosol-cloud interactions focusing on mixed-phase clouds, which means that interactions with ice crystals were included in the model.

The UCLALES-SALSA is a large-eddy simulator. It is a combination of UCLALES large-eddy simulator (Stevens et al., 1999, 2005) handling the dynamics (turbulence, fluxes, advections, etc.) and SALSA (Kokkola et al., 2008, 2018) managing microphysical processes. Novel descriptions for clouds and precipitation were added to SALSA, which was coupled with UCLALES and introduced as UCLALES-SALSA in Tonttila et al. (2017). Moreover, the SALSA module was extended with ice microphysics in **Paper I**, which was further on applied to study a mixed-phase cloud case in **Paper II**.

In this chapter, we dig into what major mathematical choices have been made to make UCLALES-SALSA work. Also, some shortcomings and compromises are discussed.

3.1 Large-eddy simulation

Large-eddy simulators (LES) have been widely used in the last two to three decades to study planetary boundary layer (PBL, see Appendix 7.2.2) phenomena (Maronga and Li, 2022). The idea of LES is to reduce the computational cost by parameterising the smallest length scales as opposed to Direct Numerical Simulation (DNS) solving all included length scales. In a LES model, the large-scale turbulence is resolved. Large-scale means spatial scales larger than the grid spacing (i.e. spatial scales $> \Delta$ = grid spacing). On smaller length scales ($< \Delta$), the impact of turbulence is accounted for with a subgrid-scale (SGS) turbulence closure (Maronga and Li, 2022). The UCLALES-SALSA model implements the Smagorinsky-Lilly subgrid model (Smagorinsky, 1963; Tonttila et al., 2017). The accuracy of the LES depends on the subgrid model, the numerical schemes, that is how the continuous equations are approximated and solved on a discrete grid (e.g. spectral or finite element meth-

ods), and the grid spacing (Pope, 2000). The separation of flows into large-scale and subgrid flows is usually executed by filtering the velocity field with a kernel $G_\Delta(\boldsymbol{x})$ (Leonard, 1974). The convolution kernel excludes scales smaller than Δ (Meneveau, 2010). Here, the LES filter is applied to a spatiotemporal field $\phi(\boldsymbol{x}, t)$. Following Pope (2000); Saugaut (2006) the filtered field (marked with a bar), is defined as

$$\overline{\phi(\boldsymbol{x}, t)} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \phi(\boldsymbol{r}, \tau) G(\boldsymbol{x} - \boldsymbol{r}, t - \tau) d\tau d\boldsymbol{r}, \tag{1}$$

where G is the filter convolution kernel. The Equation (1) can be formulated also as

$$\overline{\phi} = G \star \phi, \tag{2}$$

where $\star$ is the convolution operation. The filter kernel G has a corresponding cutoff length scale Δ and cutoff time scale τ_c. The smaller scales than the cutoff lengths are eliminated from $\overline{\phi}$, and denoted by ϕ'. For any field ϕ the following equation

$$\phi = \overline{\phi} + \phi' \tag{3}$$

holds showing that the larger scales are resolved and the smaller scales ϕ' are parameterised with the subgrid model. Hence, the choice of grid spacing (Δ) and the subgrid model is vital as they affect the accuracy and computational cost of the model.

In a large-eddy simulator, the domain size is usually in the order of ten kilometres in each horizontal direction and some kilometres in the vertical direction depending on the application. The grid resolution is from several meters to some tens of meters. In UCLALES-SALSA the horizontal domain, with uniform grid squares, is defined with doubly periodic boundary conditions, which means that the variables and their fluxes are identical on opposing boundaries. In other words, when an object passes the boundary, it reappears on the opposite side with the same velocity (like topological mapping onto a torus). In flow models in general, other possible boundary conditions are for instance solid walls and near-wall resolved turbulence (e.g. Freire (2022)). The bounded vertical domain is spanned by a stretchable grid. The vertical grid has a selected number (by default five) of topmost grid points acting as a sponge layer (Rayleigh friction), which damps unrealistically reflected gravity waves at the model top (Tonttila et al., 2017). In UCLALES-SALSA the advection of momentum variables (i.e. mass with direction and speed) is based on a fourth-order difference equation where time stepping is done with a leap-frog method. A simple forward time stepping is used for scalars describing a state, like temperature. The simulation time is usually from hours to several days but can also be some seconds or minutes. Large-eddy simulations often include a spin-up period from the start of the simulation to allow a dynamically and thermodynamically consistent state before starting the actual simulation or analysis. This includes enabling

the turbulence to become fully developed reaching the top of the boundary layer. In **Paper I**, **Paper II** and **Paper III** the spin-up period was chosen to be from one to two hours. During the spin-up period one or several processes, like microphysical processes can be switched off to prevent phoney effects on the cloud properties during the initial buildup of turbulent kinetic energy and settling of the boundary layer properties (Tonttila et al., 2017). Some processes are explicitly resolved but often parameterisations are needed, especially in sub-grid (e.g. turbulence) processes.

3.2 Microphysics

Apart from solving the dynamics, a major part of any LES modelling of the planetary boundary layer is the implemented microphysics scheme. There are several types of microphysics schemes: *bulk* (Khairoutdinov and Kogan, 2000; Golaz et al., 2005; Seifert and Beheng, 2001, 2006; Stevens et al., 2005; Savre et al., 2014), *bin* (Feingold et al., 1996; Feingold and Kreidenweis, 2002; Saleeby et al., 2015; Tonttila et al., 2017), bin-emulating bulk models where microphysical processes are parameterised to some level but only bulk scalars are transported within the model (Mansell et al., 2020) or Lagrangian particle-based methods (Shima et al., 2009), where a probabilistic microphysics was used.

Here, we focus on bulk and bin microphysics schemes. They are the prevalent schemes (Khain et al., 2015), and they are currently the only possible microphysics scheme choices in UCLALES-SALSA. In this study, both bulk and bin microphysics schemes have been employed. Typically in the bulk schemes, droplet mass is predicted along with prescribed or varying droplet number concentrations. In the bulk scheme used in UCLALES, the cloud droplets are diagnostic and rain water mass and rain droplet number concentrations are prognostic. The term *two moment* is used when both variables are predicted, if one droplet variable is predicted (the other is prescribed) then the term *one moment* is used. In bin microphysics, the droplet size distributions are divided into bins (= sections, see Figure 3). Within each bin, the droplet variables (mass, size, etc.) are predicted similarly to bulk microphysics. When choosing either bin or bulk microphysics the main trade-off is the choice between computational cost and accuracy in the representation of microphysical processes. Bin microphysics is more detailed as the shape of the size distribution is allowed to evolve. However, better details come with a higher computational cost. Bulk microphysics, where the shape of the size distribution is prescribed, has a lower computational cost but loose to bin microphysics in accuracy and in sensitivity to many microphysical processes such as the effects of aerosols on clouds (Khain et al., 2015).

In essence, as already noted in Section 2.3, both bin and bulk schemes are parameterisations since fundamentally every individual particle should be simulated. Additionally, by default, microphysics is ignorant of sub-grid processes. However,

sub-grid processes can be taken into consideration, like it has been done with coagulation kernels (See 7.2.5). However, UCLALES-SALSA produces well the average cloud properties, at least on the level that is obtainable from observations, for details see **Paper I** and Tonttila et al. (2017); Silva et al. (2021); Calderón et al. (2022).

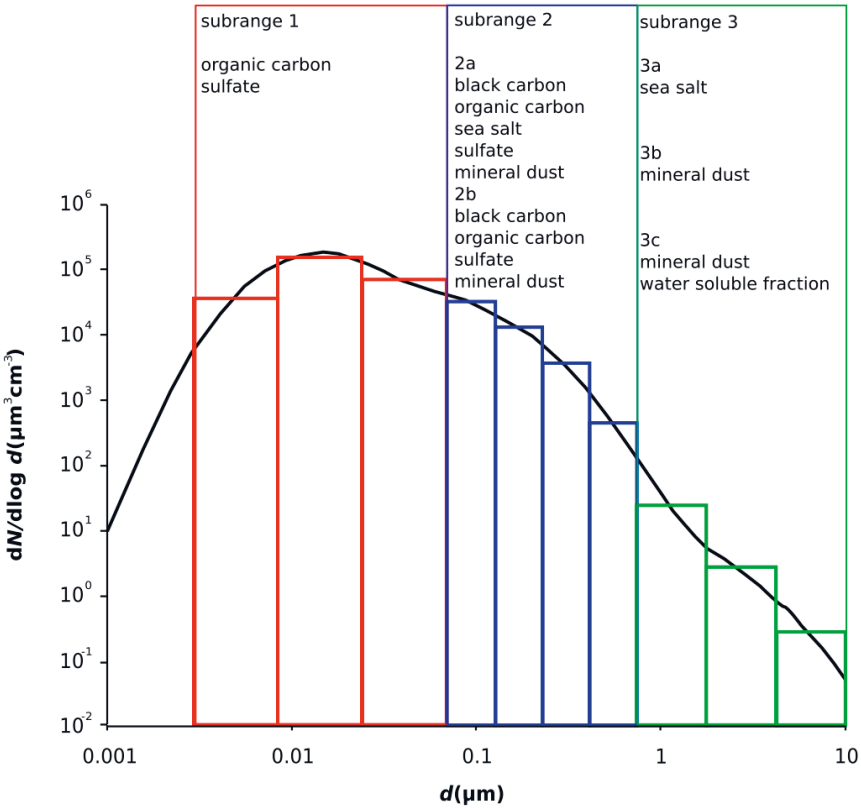

Figure 3. Illustration of bin microphysics (Kokkola et al., 2008). Figure reprinted under CC BY 3.0 licence (`https://creativecommons.org/licenses/by/3.0/`).

3.2.1 Bulk microphysics

When using bulk microphysics in UCLALES-SALSA, it shrinks back to UCLALES model (Stevens et al., 1999, 2005). In this Seifert & Beheng bulk microphysics (Seifert and Beheng, 2001) the cloud water mixing ratio is calculated with saturation adjustment method, where the relative humidity is set back to 100 % in case of supersaturation, the enthalpy of condensation is released and the temperature is increased just the right amount for 100% humidity, following with condensates as cloud droplets. The prognostic variables (See 7.2.6) are liquid water potential temperature θ_l, total water mixing ratio q_t and additionally other required prognostic

variables like rainwater mass and rain droplet number concentration. Only cloud condensate mixing ratio q_c and water-vapour mixing ratio q_v are diagnostic.

The bulk microphysics does not include a description of aerosols. Instead, the microphysical processes are governed by a prescribed cloud condensation nuclei (CCN), which is assumed to represent the number of cloud droplets throughout the whole cloud volume (cloud droplet number concentration, CDNC) (Tonttila et al., 2017). The drizzle formation follows Seifert and Beheng (2001) as

$$\frac{\partial q_r}{\partial t} = k_c q_c^2 x_c^2, \tag{4}$$

where q_r is the precipitation mixing ratio (for prognostic variable, see 7.2.6), q_c is the cloud condensate mixing ratio, $x_c = q_c/N_c$, where N_c is the cloud condensation nuclei (CCN) concentration and k_c is a coefficient taking into account the droplet size distribution width and non-equilibrium effects (Stevens and Seifert, 2008; Tonttila et al., 2017). Rain droplets can grow by coagulation, that is by colliding with each other. The rate of coagulation is defined with coagulation kernels (See 7.2.5) that are updated at each time step. Sedimentation of the cloud and rain droplets is calculated based on sedimentation velocity, which depends on the diagnosed droplet size (Tonttila et al., 2017).

3.2.2 SALSA

When the SALSA microphysics scheme is coupled to UCLALES (i.e. UCLALES-SALSA model), condensation and evaporation of water vapour on cloud droplets, raindrops and aerosols are explicitly resolved with analytical predictor of condensation (APC) scheme (Jacobson, 2005; Tonttila et al., 2017). As the bulk microphysics scheme used saturation adjustment for computing the prognostic total water mixing ratio (q_t), with the SALSA bin microphysics cloud condensate mixing ratio (q_c), rainwater mixing ratio (q_r) and the water vapour mixing ratio (q_v) are treated as separate prognostic variables. This enables realistic non-equilibrium conditions with respect to water (Tonttila et al., 2017).

With the SALSA module, the aerosol size distribution is discretised into n size bins according to the dry particle diameter (Bergman et al., 2012). Aerosol number, compound masses (sulphate, dust, organic carbon, sea salt, nitrate and ammonium) and the mass of condensed water are the prognostic variables for each bin. By default, the number of bins is 10 and the bins cover a diameter range from 3 (nm) to 10 (μm). The range is divided into subranges 1a and 2a (see Figures 3 and 4). The division aims to reduce the number of computed tracer variables by including only those compounds that are the atmospherically most significant in each subrange (Kokkola et al., 2008; Tonttila et al., 2017). The particles in individual subranges are assumed to be internally mixed. Incorporating a parallel subrange 2b enables the external mixing of particle populations. A typical setup is to have soluble compounds

in subrange 2a and insoluble compounds in subrange 2b. The spacing of the size bins is established as logarithmically equidistant within each of the subranges.

As a unique strategy, in UCLALES-SALSA cloud droplets are described based on the dry size of the activated aerosol with identical prognostic bin quantities as for the aerosol bins within their common size range, which is by default the 2a and 2b bins. Following, each cloud droplet bin has a parallel aerosol bin. This approach enables preserving the aerosol size distribution and the number concentration upon cloud activation and droplet evaporation (Tonttila et al., 2017).

Contrary to the cloud and aerosol size bins, the precipitation size bins follow wet drop diameter to allow realistic collection processes and surface precipitation. The number and mass of droplets to be transferred to precipitation size bins are determined by an autoconversion parameterisation based on a mathematically simple log-normal distribution. By default, the threshold diameter for drizzle droplets is $50(\mu m)$.

With these modelling choices, the spectral resolution is quite coarse but provides a good compromise between computational cost and model performance. However, the bin number can be changed and in **Paper II** and Prank et al. (2022) it has been increased. For instance, additional bins improve the size resolution for cloud droplets and ice particles.

The SALSA module calls sequentially for the implemented microphysical processes that are detailed and of key importance. These processes include for example sedimentation, coagulation (= collision-coalescence, i.e. particles colliding) and condensation of water vapour and aerosol precursor gases, which are implemented following from Jacobson (2005).

To accurately model the evolution of the aerosol size distribution through cloud processing and wet scavenging, a two-dimensional dry-wet diameter bin system would be needed. This derives from cloud activation being dependent on dry aerosol size distribution, whereas collision processes and sedimentation rates depend highly on the wet particle size. Such detailed two-dimensional model frameworks exist (Lebo and Seinfeld, 2011) but have a high computational cost in LES applications. SALSA aims to resolve the issue of computational cost by the compromise of having cloud droplet bins described with the dry size of the activated aerosol (i.e. CCN) and a parallel aerosol bin with identical prognostic bin quantities. This design enables the shape of the aerosol size distribution and the number concentration is conserved upon cloud droplet activation and upon droplet evaporation. However, drizzle droplets (rain) are tracked by wet droplet size as relevant drizzle processes depend on wet size. By this choice, information about the aerosol size distribution is not as accurate as with aerosols and cloud droplets. Nonetheless, the compromise of tracking rain droplets by wet size is reasonable as the number concentration of rain droplets is always much smaller than the number concentration of cloud droplets or aerosols (Tonttila et al., 2017).

In SALSA the cloud activation can be calculated with two different methods. The parameterisation method based on Abdul-Razzak and Ghan (2002) is more applicable in a coarser resolution regime (several tens of meters and above). The other method that has been used in all the **Papers** is the one based on resolving the wet aerosol particle diameter. The aerosol particle is activated once the wet diameter of the particle exceeds the critical diameter corresponding to the resolved supersaturation from the host model (Tonttila et al., 2017). Here, the condensation of water vapour onto aerosols is solved iteratively when the relative humidity is high. The critical diameter is also dependent on the chemical composition and the size of the dry CCN. The activated particles are moved to corresponding cloud bins that by default cover dry CCN size range from 50 (nm) to 10 (μm).

Since the spectral resolution of the aerosol bins is relatively coarse, it may cause some undesirable discontinuities in the activation spectrum with increasing saturation ratio due to the particle size discretisation (Tonttila et al., 2017). To reduce these discontinuities in cloud activation, the distribution of the particle mass and number in the critical aerosol size bin is adjusted by using linearly fitted slopes between the bin centres (Korhonen et al., 2005). Evaporation and deactivation of cloud droplets are modelled with resolved condensation, where activated aerosol particles are released back to the aerosol bin regime (Tonttila et al., 2017).

In **Paper I** ice microphysics was implemented within the UCLALES-SALSA bin microphysics scheme and further on applied in **Paper II**. Figure 4 shows the new bins associated with ice microphysics. Identical size bins 2a and 2b used for aerosols, cloud droplets and ice allow tracking of aerosol development through cloud activation, freezing and sublimation. The new ice microphysical processes implemented to UCLALES-SALSA are droplet freezing, deposition and sublimation of water vapour, melting when $T > 0^{\circ}C$, coagulation between different hydrometeors, sedimentation, and ice crystals interacting with radiation. Droplet freezing includes immersion, homogeneous, deposition, contact and condensation freezing. Immersion freezing is the focal point in both **Paper I** and **Paper II** as the other modes of ice nucleation are not applicable to the cloud conditions being simulated. The freezing rates are predicted using the stochastic freezing parameterisation that is based on classical nucleation theory (Khvorostyanov and Curry, 2000) with additional parameters from Jeffery and Austin (1997); Khvorostyanov and Curry (2004); Li et al. (2013). The freezing rates depend primarily on ambient conditions, the properties of the solid insoluble substrate and ice nucleating particle (INP) concentration. The ambient conditions include temperature and relative humidity over ice. The latter substrate properties are described with compound-specific ice nucleation parameters (e.g. contact angle).

In the first version of UCLALES-SALSA, (Tonttila et al., 2017) warm SALSA microphysics was coupled to UCLALES, where particles are assumed to be spherical. In the case of warm microphysics that is a reasonable assumption. The ice crys-

tals tend to be hexagonal or fractal shaped depending on temperature and in-cloud processes. As the saying goes, there are no identical snowflakes. The ice crystals in **Paper I** and **Paper II** are assumed to be spherical. However, concerning the application of modelling stratocumulus clouds, the ice shape assumption is fairly reasonable with the following definitions. In the UCLALES-SALSA ice microphysics description, ice shape is characterised by a mass-diameter parameterisation

$$m = a_m D^{b_m}, \tag{5}$$

where D is the maximum particle dimension related to capacitance C via $D = \pi C$ and $a_m = 44.2(kg\, m^{b_m})$ and $b_m = 3$. Capacitance is a measure acting as an effective radius for non-spherical particles (used also in condensation) and defined as

$$C = a_c m^{b_c}, \tag{6}$$

where $a_c = 0.09\ (m\ kg^{-b_c})$ and $b_c = \frac{1}{3}$. This parameterisation corresponds to spherical particles having a low effective density ($\rho = 84.5\ (kg\, m^{-3})$). Additionally, the ice fall speed is described with

$$V = a_v D^{b_v}, \tag{7}$$

where $a_v = 12\ (m^{1-b_v}\ s^{-1})$ and $b_v = 0.5$. This parameterisation is given in Ovchinnikov et al. (2014) and it represents an idealisation of dendrites as spheres of constant and low equivalent density. It is suitable for model intercomparison but does not account for changing aspect ratio frequently happening in growing crystals (Sulia and Harrington, 2011). We could have used a more detailed ice crystal parameterisation but this parameterisation was used in **Paper I** to allow comparison of our model to other models given in Ovchinnikov et al. (2014).

As we can see there are several parameterisations that had to be done in order to model ice crystals within a large-eddy simulator. Although UCLALES-SALSA has some limitations, it is still a useful modelling tool as shown in **Paper I** and **Paper II**. Furthermore, both papers showed the INP recycling within the boundary layer and **Paper II** also showed the importance of updrafts for importing INPs from the surface to the cloud layer. All things considered, UCLALES-SALSA has been shown to model the aerosol-cloud-ice-precipitation interaction well. The mathematical choices of ice microphysics of UCLALES-SALSA are discussed further in Section 6.1.2.

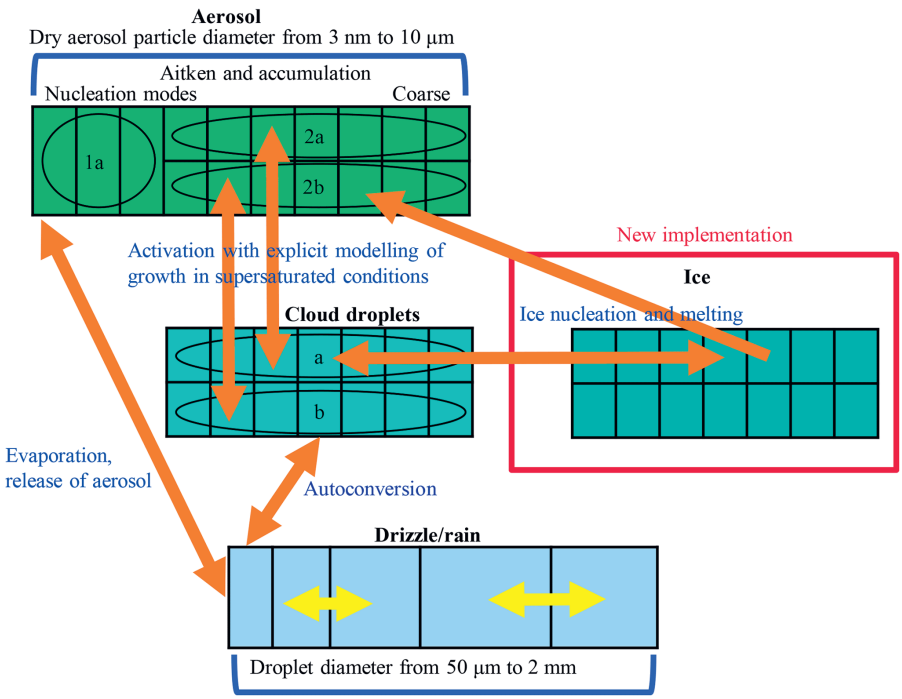

Figure 4. Illustration of SALSA microphysics bin scheme in **Paper I**. Figure reprinted under CC BY 4.0 licence without modifications (`https://creativecommons.org/licenses/by/4.0/`).

4 Parameterising cloud processes: methods and results

As already stated, clouds are one of the major features of the climate system causing uncertainties in predicting climate change. Clouds are particularly hard to model with global climate models as their resolution is coarse compared to the fine details of clouds (Honnert et al., 2020). Cloud droplets form when air cools and then water vapour condenses on small particles, aerosols. Regarding stratocumulus clouds, the cooling of air happens usually via air parcel rising (i.e. adiabatic expansion in the updraft) or radiative cooling in cloud top. Aerosols can be of either natural or human origin, like dust, sulphates, black carbon, and so forth. Cloud droplet number concentration forming at the cloud base is determined by aerosol properties and how fast the air parcel rises, that is updraft velocity. Updraft velocity is a component of turbulent air flows in the planetary boundary layer (i.e. near the Earth's surface). In general, turbulence is caused by excessive kinetic energy in some parts of a fluid flow, where the damping effect of the fluid's viscosity cannot keep the fluid flow laminar. In the atmosphere, the instability of the atmosphere is a major driving force for turbulent air flows. The stability of the atmosphere is defined by comparing the temperature of a rising or sinking air parcel to the environmental air temperature. An unstable atmosphere favours vertical motions. The stability of the atmosphere is affected by fluxes from the surface (heat fluxes, moisture fluxes) and changes caused by radiation at different altitudes. Latent heat has a greater impact on stability (i.e. temperature) than radiation except at cloud top. Radiative cooling of clouds and advection of cooler air masses can also destabilise the atmosphere (Seinfeld and Pandis, 1998; Jacobson, 2005).

In the current global climate models, the grid resolution is at its best tens of kilometres, which is two to three orders of magnitude too low to resolve cloud structural variability. Superparameterisations are a potential solution, where a high-resolution model, like LES, is run within every low-resolution column but it is highly computationally expensive and therefore too heavy for climate predictions. Instead of trying to fully resolve the cloud development with a superparameterisation, vital properties for cloud evolution, like updraft velocity, are parameterised as accurately as possible. To get an estimate for updraft velocity, most models employ a parameterisation based on a proxy which is often the estimate of turbulent kinetic energy (Golaz et al., 2011). Other possible means of updraft velocity parameterisation are based on

cloud-top radiative cooling or cloud-top height, where updraft velocity is presented as a linear function of the proxy, but the methods are not yet widely used (Zheng and Rosenfeld, 2015; Zheng et al., 2016).

To achieve high accuracy with a low computational cost, in **Paper III** we represented three different updraft parameterisation methods based on supervised machine learning where training data is obtained from detailed LES simulations. In **Paper IV** we used one of the created parameterisations, Gaussian process emulator (GPE), in the ECHAM-HAMMOZ global climate model (later on ECHAM abbreviation is used, see also Section 2.1.2). GPE was chosen in **Paper IV** as a reasonable compromise due to its simpler implementation compared to the Random Forest based method (LFRF) and expressive power compared to the simplest linear method (LF) as shown in intercomparison in **Paper III**. Additionally, GPE was the first method envisioned to be used. The other two methods emerged as we were writing **Paper III**. Moreover, the required Gaussian process libraries were available in FORTRAN, which is the programming language of the ECHAM model. In **Paper IV** , GPE was used for predicting values of updraft velocity and rainwater formation rate. In ECHAM, the latter directly affects precipitation. Precipitation is one of the major cloud microphysical processes affecting cloud water content, dynamics and lifetime. The chosen cloud processes are identified as one of the major sources of uncertainty in the cloud radiative forcing estimates in present climate models (Donner et al., 2016; Jing et al., 2019; Yoshioka et al., 2019; Bougiatioti et al., 2020).

Here we are examining the parameterisations from a detailed mathematical point of view. Creating the parameterisation follows the workflow shown in Figure 5. Parameterisation means here an emulator, a statistical model, that emulates the behaviour of a simulator (here: large-eddy simulator, LES) and it estimates simulation output at untried input combinations. According to a common naming convention in supervised machine learning, we name training input data, $\mathbf{X}$, as *feature vectors* and training output, $\mathbf{y}$ as *target values*. After training (i.e. fitting), we have a parameterisation

$$\tilde{f}(\mathbf{X}) = \mathbf{y}. \tag{8}$$

In optimal case, the parameterisation $\tilde{f}$ has the same output as LES with a fraction of computational cost.

4.1 Filtering ECHAM to create source data

This section covers how a large set of low marine clouds is filtered from climate model simulation data. This part of the workflow is labelled (A), 1. and (B) in Figure 5. Here we show how original raw data (A) is filtered (1.) to create a clean sample set (B).

The Initial ECHAM data was generated from one one-year ECHAM-HAMMOZ (ECHAM6.3-HAM2.3-MOZ1.0) AMIP (Atmospheric Model Intercomparison Project)

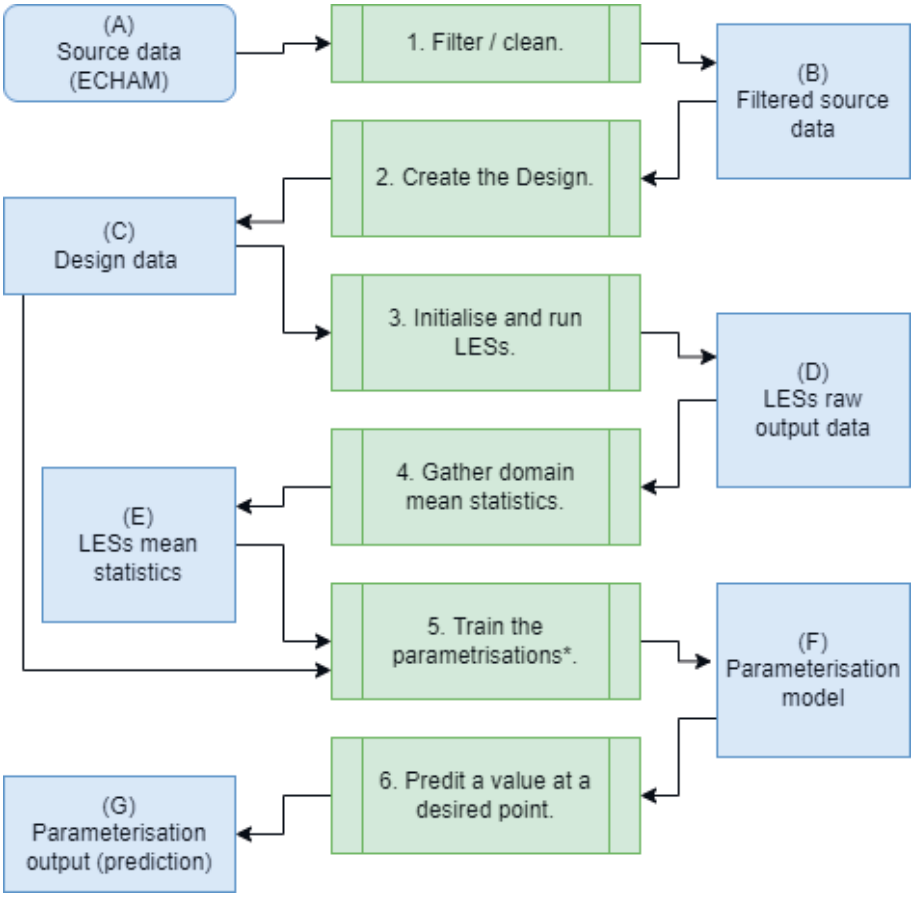

Figure 5. The workflow laid out for creating a cloud process parameterisation. Lettered (blue-coloured) boxes refer to data that can be input or output for a process. Numbered (green-coloured) boxes refer to processes. * means that usually (C) forms up the feature vectors (= training input data) and (E) are the target values (= training output data) but some variables in (E) can be used as features also.

type run. Filtering of the source data is given as a pseudocode in Algorithm 1. First, any marine column without sea ice is included in the set (lines 1-7). Next all low cloud columns are identified (lines 8-14). Here low-level cloud is defined as a column where there is more than 0.01 ($\mathrm{g\ kg^{-1}}$) cloud liquid water below 700 hPa pressure level. Fog columns are eliminated by excluding all columns that have cloud water in the first level above ground (lines 15-21). From the remaining columns, the lowest cloud layer is identified, and the Liquid Water Path (LWP) inside that lowest cloud is calculated (lines 22-25). Here, the low cloud is defined as the lowest continuous layer with one or more cloudy ECHAM levels below 700 hPa. Finally, a column is accepted as input data if more than half of the total column WP ($=$ LWP $+$ IWP, WP = Water Path, IWP = Ice Water Path) is inside the lower cloud layer, which means that

there would be no cloud above the low cloud. Such layered clouds were discarded as they affect radiative flux. Additionally, we planned that there would be no ice in the low cloud, which is defined so that IWP is less than 10% of the LWP in the low cloud region (lines 26-27). The boundary layer height was defined as the difference between sea level pressure and pressure at the detected cloud top (line 30). Additional limitations could be set in order to avoid clearly improper LES initial values, for example, thresholds for low cloud LWP, and temperature and humidity inversion, but we did not see a clear need for this.

Only low altitude and stratiform clouds are included since LES is capable of modelling them with detail and global models have known issues in modelling them. Additionally, the more convective clouds and cloud systems, like high or thick clouds, would require a larger simulation area and a more complicated simulation setup. Both requirements would increase the computational cost of the LES model and thus make the execution of simulation ensembles impractical.

Columns that do not have clouds but could potentially generate clouds would require a longer simulation time. An extended duration of simulation would contradict our objective of simulating the cloud evolution within an ECHAM time step. In addition, this would be problematic since the initial cloud state would differ excessively from the end state. The excessive difference in cloud states would mean that the initial state and end state would not be usable together as machine learning training data.

As the source data is cleaned with the Algorithm 1, we select appropriate variables that can be used to describe a cloud state and therefore can be used to initialise a cloud simulation. These variables and how they are calculated are shown in Table 1. The variables LWP and H_{PBL} are calculated in the Algorithm 1.

Algorithm 1 Filtering Source data (ECHAM)

while $p \in$ COLUMNS **do**
  if p is not above land **then**
    if p is not above sea ice **then**
      $p \in SC$ {p belongs to a group of columns where columns above land and sea ice are excluded}
    end if
  end if
end while
while $p \in SC$ **do**
  if pressure level $< 700\mathrm{hPa}$ **then**
    if cloud liquid water content $> 0.01\mathrm{kg\ m^{-3}}$ **then**
      $p \in CC$ {Identify p as a cloudy column (CC)}
    end if
  end if
end while
while $p \in CC$ **do**
  if lowest level contains cloud water **then**
    Exclude point p
  else
    $p \in$ NFCC {p is a non-foggy and cloudy column}
  end if
end while
while $p \in$ NFCC **do**
  Calculate total column water path (TC-WP)
  Identify lowest cloud layer (LCL)
  Calculate the liquid water path within lowest cloud layer (LCL-LWP)
  Calculate the ice water path within lowest cloud layer (LCL-IWP)
  if LCL-LWP $> 0.5 \times$ TC-LWP **then**
    if LCL-IWP $< 0.1 \times$ LCL-LWP **then**
      $p \in$ FILTERED {accept p as a column for Filtered data}
      Define boundary layer height (PBLH) as the difference between sea level pressure and pressure at detected cloud top
    end if
  end if
end while
Separate FILTERED data between night and daytime
return `FILTERED_NIGHT` **and** `FILTERED_DAY`

Variable name	Unit	Variable explanation	Way of retrieving from ECHAM	SB	SALSA	DAY	Phase
Δq_t	$g\,kg^{-1}$	jump in total water mass mixing ratio at the boundary layer top	difference of max and min values of the total water within two levels from the cloud top				1,2
θ_L	K	liquid water potential temperature in the boundary layer	the minimum value of potential temperature (the same levels as for Δq_t)				1,2
$\Delta\theta_L$	K	inversion strength of liquid water potential temperature	difference of max and min values of potential temperature (the same levels as for Δq_t)				1,2
LWP	$g\,m^{-2}$	liquid water path for the cloud	integrated from the surface up to the cloud top				1
H_{PBL}	hPa	planetary boundary layer height is described as a pressure difference from the surface	pressure difference from the surface up to the cloud top				1,2
$CDNC$	mg^{-1}	cloud droplet number concentration	averaged over the cloud	1			1,2
r_{eff}	nm	effective dry radius of accumulation mode	calculated based on values from the lowest level		1		2
N_{Ait}	mg^{-1}	aerosol number concentration in the Aitken mode	concentration from the lowest level		1		2
N_{acc}	mg^{-1}	aerosol number concentration in the accumulation mode	concentration from the lowest level		1		2
N_{coa}	mg^{-1}	aerosol number concentration in the coarse mode	concentration from the lowest level		1		2
$\cos\mu$	-	cosine of solar zenith angle	as is			1	2
CLW_{max}	$g\,kg^{-1}$	maximum cloud liquid water mixing ratio	From ECHAM standard output vphyscis stream, calculated inside pbl				1

Table 1. Variables to represent a cloud state and used to initialise the cloud simulations. SB refers to microphysical variables specific only to Seifert & Beheng -microphysics scheme (blue colour), SALSA refers to microphysical variables specific only to a more detailed SALSA microphysics scheme (orange colour). DAY means if the cloud simulation is to be run as a daytime simulation when $\cos\mu$ variable is needed (yellow colour). Phase refers either to the first design version (1) that was discarded later or the design version (2) used in **Paper III** and **Paper IV**. The green colour shows the variable that was used only in Phase I.

4.2 Sampling source data to create a set of initial states for LES runs

Once adequate source data (= sample set) is acquired, a representative subset of the source data is sampled. These subsets are called *designs*, which are used to initialise the LES simulations.

As a rule of thumb, the number of simulations (= samples in the design) needs to be at least 10 times the number of input parameters (Loeppky et al., 2009). For SB sets we used 500 simulations each. For SALSA sets we used 135 and 150 simulations, for nighttime and daytime simulations, respectively. The lower number of simulations in SALSA sets is chosen since the computational cost increases approximately 16-fold with SALSA microphysics.

All possible design features are listed with a description in Table 1. Selected design features for each design are listed in Table 2 along with the number of design features. There are several methods of creating a design which are explained in the following.

Design	Number of design features p	design features
LHS	6	$\Delta q_t, \theta_L, \Delta\theta_L, H_{PBL}$, $CDNC, CLW_{max}$
SB night	6	$\Delta q_t, \theta_L, \Delta\theta_L, H_{PBL}$, $CDNC, LWP$
SB day	7	$\Delta q_t, \theta_L, \Delta\theta_L, H_{PBL}$, $CDNC, LWP, \cos\mu$
SALSA night	9	$\Delta q_t, \theta_L, \Delta\theta_L, H_{PBL}$, LWP, r_{eff}, $N_{Ait}, N_{acc}, N_{coa}$
SALSA day	10	$\Delta q_t, \theta_L, \Delta\theta_L, H_{PBL}$, LWP, r_{eff}, $N_{Ait}, N_{acc}, N_{coa}$, $\cos\mu$

Table 2. The design features and the number of them for each design. The explanation of design features is given in Table 1.

4.2.1 Sampling method: Latin Hypercube Sampling

Latin Hypercube Sampling (LHS) is a statistical method where a near-random sample is generated from a multidimensional distribution (McKay et al., 1979a). Within statistical sampling, a Latin square is defined as a square grid where there is only one sample position in each row and each column. If we want to generalise this

concept to an arbitrary number of dimensions we call this a Latin hypercube, where each sample is the only one in that specific axis-aligned hyperplane (= non-collapsing). Independence is one of the main advantages of LHS. When creating a sample set, first the number of samples is needed. For each sample point, the dimensional coordinates must be recorded. It is like having N rooks (= number of samples) on a chessboard without threatening each other. LHS makes sure that the samples represent the real variability.

Following Atangana (2018), in formal terms, LHS is a sample with size N from the X variables $x_1, x_2, x_3, \ldots, x_p$. The range of each variable is partitioned into N non-overlapping intervals (= bins) based on the sample size with equal probability $1/N$. One value from each interval is chosen randomly according to the probability density in the interval. As follows, the N values from x_1 are paired randomly with the N values of x_2. Then these N pairs are coupled with the N values of x_3 to build Np-triplets until a set of Np-tuples is completely built. The set Np-tuples is the Latin hypercube sample. For a sample size N and p variables, there exist $(N!)^{p-1}$ possible interval combinations for an LHS.

4.2.2 Creating a design with LHS

In Phase I, we used LHS to sample design features $\Delta q_t, \theta_L, \Delta\theta_L, H_{PBL}$ and CLW_{max} to describe the meteorological state of the cloud and $CDNC$ to describe the microphysical state using Seifert & Beheng -microphysics scheme. These design features are listed also in Table 1, marked with 1 in the column *Phase I*. These design features are used to infer further parameters that are needed to describe the initial state of the cloud simulation. However, LHS does not inherently make sure that the interconnected design features are physically realistic. For example, the chosen sample points should not end up representing negative concentrations. For LHS designs, physical feasibility is ensured by filtering sample points with the following constraints.

First, we needed to make sure that the humidity jump at the boundary layer (Δq_t) was strong enough but not greater than the total water mixing ratio in the boundary layer (q_t), which would lead to impossible negative concentrations, that is

$$1\ (g\ kg^{-1}) < \Delta q_t < q_t, \tag{9}$$

where q_t is deduced from other design features.

Second, the constraint for the temperature jump,

$$1\ (K) < \Delta\theta_L < 15\ (K), \tag{10}$$

depicts a typical temperature jump seen in stratocumuli. Both constraints (9) and (10) enable a distinct enough planetary boundary layer that allows the existence of a stable cloud for the planned simulation period. The existence of a planetary boundary layer is essential for turbulent low-level clouds (e.g. marine stratocumuli).

Third, the planetary boundary layer height is to be lower than 3 kilometres as stratocumuli rarely extend beyond that and usually are lower than 2 (km) in height (World Meteorological Organization Cloud Atlas). Additionally, there should be more than one cloud-free layer as fog should not be included, that is

$$\begin{aligned} &30 + 50\ (m) < H_{PBL} < 3000\ (m) \\ &CB > 30\ (m). \end{aligned} \tag{11}$$

Fourth, a constraint makes sure that there is initially a cloud. Formally, the maximum cloud water content is limited by

$$CLW_{max} > 0.1(g/kg). \tag{12}$$

Nonetheless, using LHS turned out to be impractical as the constraints removed too many of the points and we could only use a fraction of the filtered source data (= low feasibility ratio). This was mostly due to the fact that moisture and temperature are interdependent physical variables (i.e. warmer air can hold more moisture) and therefore cannot be independently selected using LHS.

4.2.3 Sampling method: Binary space partitioning

To tackle the issues with LHS, we changed the design concept to use a simple stratified sampling method based on binary space partitioning (BSP) trees (Fuchs et al., 1980; Tóth, 2005).

The idea for BSP is to partition the space along a hyperplane into two half-spaces, and then both of these half-spaces are further recursively divided into half-spaces until every subproblem contains only a trivial fraction of the input objects. The input set consists of pairwise interior disjoint objects in $\mathbb{R}^p, p \in \mathbb{N}$.

The partition algorithm is similar to a binary tree. All the inputs of a recursive call of the BSP equates to a node. The root of the tree refers to the initial input set. The two children of a non-leaf node refer to the inputs of its two subproblems. The data structure of a BSP tree is as follows. Every leaf holds a maximum of one full-dimensional object which is the input of the corresponding subproblem. Every non-leaf node holds the splitting hyperplane and the (lower-dimensional) objects of the related subproblem that exist on the splitting hyperplane.

It is a custom that the non-leaf nodes stash only d-dimensional fragments of d-dimensional objects lying on the splitting hyperplane in $\mathbb{R}^p, 0 \leq d \leq p$. For example, if a splitting hyperplane h bisects an input segment s then the point $h \cap s$ is never stored, which means that optimally, the partition hyperplanes do not split the input objects (Tóth, 2005), however with our implementation algorithm this is not a concern. Algorithm description of BSP is given in Algorithm 2.

Algorithm 2 Binary Space Partitioning based sampling

```
Set initial partition as the whole collection
while number of partitions less than the desired amount do
    Create a list of randomly permuted input dimensions
    for each dimension in the permuted list do
        for each partition do
            if number of partitions less than the desired amount then
                Subdivide the partition into two, using the median along the current
                dimension
            end if
        end for
    end for
end while
Uniformly sample a point in each partition and add to the design
return Subsampled design with a given number of points
```

4.2.4 Creating a design with BSP

As we discarded LHS and used BSP to create a design, we also incorporated LWP as a feature and removed CLW_{max} from features. The change of features was implemented since LWP contains all the cloud water in ECHAM cloud profiles and using LHS-based design yielded unrealistic cloud profiles too frequently. In this second phase, we created four different design sets. Two design sets had detailed SALSA microphysics schemes where microphysical design features were r_{eff}, N_{Ait}, N_{acc} and N_{coa}. Two design sets had Seifert & Beheng -microphysics, which was already used in the LHS design. Each microphysics scheme is split into nighttime and daytime design sets (i.e. two design sets for each microphysics scheme). With daytime simulations, an additional design feature $\cos\mu$ (cosine of solar zenith angle) was implemented to incorporate solar radiation in the design. The splitting into daytime and nighttime design sets was applied to improve the cloud process parameterisation. Since $\cos\mu$ gets real number values during daytime but daytime and nighttime differ from each other in a binary way (i.e. non-differentiable, not smooth), the splitting improves the accuracy of the machine learning models.

4.3 LES runs

Now that we have a design, it is used to initialise the LES runs (Figure 5 process 3.). Then domain mean statistics are collected from the LESs raw data. The surface area of the LES domain is $10 \times 10 \times (km^2)$. The domain height varies depending on the simulation case from 200 (m) to 3000 (m). The resolution is greater than or equal to $50 \times 50 \times 10$ (m^3). The vertical resolution depends on the domain height. We used

UCLALES-SALSA as the LES (see Section 3), which is written in FORTRAN. The simulations were run on a Cray supercomputer with 100 CPU cores (= 4 nodes, 3 of which are used in full and 16 cores from one node) assigned for each simulation. Details about the supercomputer are given in Table 3. The total computational cost of all the LES runs is approximately 517 000 CPU hours as each set takes 3-14 days in real time on a supercomputer, see Table 4 for more details.

Available compute nodes	168
Cores	Intel 14-cores Haswell
Cores per node	28
Peak performance per compute node (GF)	1030.4 GF
Total peak performance (TF)	173.10 TF
Memory peak per compute node (GB)	128 GB
Total system compute memory (TB)	21.0 TB
Interconnect topology	Dragonfly

Table 3. Details about the used Cray supercomputer. GF means billion (10^9) and TF means trillion (10^12) floating point operations per second.

	LES run CPU cost (h)	Post-processing CPU cost (h)	Number of simulations	Total cost per simulation (h)
SB night	45 079	10	500	90
SB day	48 357	9	500	97
SALSA night	199 730	3	135	1 480
SALSA day	224 239	4	150	1 495
Total time	517 405	26	1 285	3 161

Table 4. The computational cost of LES runs and postprocessing of the LES outputs in CPU hours.

4.4 Creating parameterisations

Here, we present three parameterisations that were used to estimate cloud-base updraft velocity.

4.4.1 Linear Fit for updraft velocity

First, we create a simple Linear Fit (LF) for updraft velocity. In Zheng et al. (2016) updraft velocity is parameterised as

$$W_b = -0.44 \times CTRC + 22.30 \pm 13, \tag{13}$$

where W_b (updraft velocity at cloud base) and $CTRC$ (Cloud Top Radiative Cooling) have units of $\mathrm{cm\ s^{-1}}$ and $\mathrm{W\ m^{-2}}$, respectively. Here, negative $CTRC$ values indicate cooling. In Zheng et al. (2016) the parameterisation is based on data from meteorological observations. In a similar way, we created a parameterisation where feature vector $CTRC_{LES}$ and target values W_{LES} originate from LESs output data. Hence, we have linear regression, a simple approximation, for updraft velocity W_{LF} (compare with (8)):

$$\tilde{f}_{W_{LF}}(CTRC_{LES}) = W_{LES}. \tag{14}$$

Updraft velocity is affected by solar radiation, as stated in the beginning of Section 4, and Zheng et al. (2016) state that they found a statistically significant relationship between cloud-top radiative cooling and updraft velocity at cloud base. Additionally, marine stratocumuli often occur in areas, where surface sensible heat flux is low compared to how radiative cooling alters the temperature profile (Wood, 2012). These points lead us to conclude that this parameterisation has a justifiable physical basis.

4.4.2 Linear Fit improved with Random Forest (LFRF) for updraft velocity

The second parameterisation, LFRF, is based on the approximation error correction method, which was introduced in Lipponen et al. (2013, 2018). The idea of the method is to have a physics-based model that provides an estimate, and the estimate is corrected based on selected parameters. In Lipponen et al. (2018) they showed that this way of integrating the physics model into the prediction is more accurate than direct machine learning prediction with the selected parameters. Here, the physics-based model is the Linear Fit (LF, Section 4.4.1).

LFRF predicts the *approximation error*, that is the difference between the LF updraft velocity and the LES updraft velocity. Hence, the target value $\mathbf{y}$ (see Equations (13) and (14)) is defined as

$$\mathbf{y} = W_{LES} - W_{LF}. \tag{15}$$

The features are the design features augmented with $CTRC$ values from LES output. The error prediction is then used to correct the predictions of LF.

The training of the parameterisation (i.e. fitting) is executed with Random Forest regression model (Breiman, 2001) and the implementation is based on the Scikit-learn machine learning package written in Python (Pedregosa et al., 2011). Each

tree in a Random Forest is built from a sample drawn with replacement (i.e. bootstrapping) from the training set. When each node is split during the construction of a tree, the best split is found either from all input features or a random subset of features (Pedregosa et al., 2011; Breiman, 2001). A Random Forest regressor is an ensemble of binary regression trees and can be considered as a piecewise-defined constant function. Random Forests can learn nonlinear functions and are to some extent tolerant to overfitting (Hastie et al., 2008).

Formally, Random Forests for regression are created by growing trees depending on a random vector Θ such that the tree predictor $h(x, \Theta)$ intakes numerical values. The output values are numerical and the training set is assumed to be independently drawn from the distribution of the random vector Y, X. The mean-squared generalisation error for any numerical predictor $h(x)$ is

$$E_{X,Y}(Y - h(X))^2 \tag{16}$$

The Random Forest predictor is formed by taking the average over k of the trees $\{h(x, \Theta_k)\}$ (Breiman, 2001).

4.4.3 Gaussian process emulator for any LES output

The third method, the Gaussian process emulator (GPE) we used was the Gaussian Process (GP) emulator (O'Hagan, 1978; O'Hagan, 2006; Rasmussen and Williams, 2006), which was actually our initial idea for creating a parameterisation. GP is a stochastic process (random function) so that every finite collection of those random variables has a multivariate normal distribution. Gaussian distribution has mean and variance, GP on the other hand has a mean function and covariance function, and thus the GP can be thought of as a generalisation of Gaussian probability distributions to functions. An arbitrary smooth function f can be estimated with GP as

$$f(\mathbf{x}) \sim \mathbb{GP}\big(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')\big), \tag{17}$$

where $m(\mathbf{x})$ is mean function, $k(\mathbf{x}, \mathbf{x}')$ covariance function (= kernel). Any smooth function can be modelled with GP. A GP emulator correctly reproduces the model output at training data points. The benefit of the GP emulator is that it also provides uncertainty (variance) of the prediction. However, this feature was not applied in **Paper III** nor in **Paper IV**. Thus, the GP emulator is more than an approximation, since it makes a full probabilistic prediction of what the simulator would output. Along with the LF method, which is a simple machine learning method as it is based on linear regression, GPE is implemented here as a pure machine learning method and can be used to emulate any of the LESs outputs. A pure machine learning method means that it does not include any known dependency but it learns the input-output relationship based on only the training data. It is worth noting that GPE could also be implemented so that it includes a known dependency by introducing it through a

prior mean function similar to the LFRF setup. Random Forest could also be used as a pure machine learning method but according to Lipponen et al. (2013, 2018) incorporating a dependency improves the method.

We used GPE to parameterise updraft velocity in **Paper III** and in **Paper IV** parameterisation of precipitation formation was included as well. The implementation of GPE in **Paper III** is based on the Skicit-learn machine learning package written in Python (Pedregosa et al., 2011) which has a theoretical foundation provided in the Algorithm 2.1 in Rasmussen and Williams (2006). On the other hand, the implementation of GPE in **Paper IV** is written in FORTRAN and is based on the GPF library (GitHub:ots22/gpf) but has the same theoretical base (Rasmussen and Williams, 2006) as the Python implementation.

4.5 Parameterisation creation conclusions

In **Paper III** we obtained promising results, especially with LFRF and GPE. Our LFRF results are also in line with Lipponen et al. (2013, 2018) where they show that incorporating a dependency (i.e. improving a rough empirical model with Random Forest) improved the results compared to a pure Random Forest learning method. The improvement by the Random Forest shows that the LF does not adequately capture the relationship and further modelling of the errors is required. This is typical in statistical modelling and Random Forest is only one potential approach to address the issue.

Results with SB microphysics were slightly better than with SALSA. One possible reason for this is that a more detailed microphysics scheme increases the degrees of freedom and concurrently the internal variability of the model. The variability is also harder to obtain as the number of simulations was lower due to the higher computational cost.

Considering the differences between GPE and LFRF, it should be noted that GPE is a pure machine learning method that does not incorporate any physical dependencies, while LFRF entails a linear approximation of updraft velocity as a function of cloud top radiative cooling. The GPE as implemented here does not include any specific prior mean function and performs similarly to the LFRF. This is because it is also trying to represent all relationships in the model and not a single relationship like the LF. When extrapolating outside of the range of the training input data, the GPE prediction reduces to the mean of the training outputs. If GPE were used with a prior mean function similar to LFRF, it would reduce to the mean of the prior function. Here, LFRF reduces to the linear approximation of updraft velocity, which is still a physics-based estimation rather than the mean of training output with GPE. Furthermore, since LFRF can be considered as a piecewise-defined constant function, it allows for the possibility of outperforming GPE in certain subsets of inputs. This is particularly true when the subset is small and sharp-edged. Instead of a piecewise-defined function, GPE makes predictions based on the whole training point domain

and requires the predicted function to be smooth as irregularities can cause GPE to go grievously amiss. Compared to LFRF, GPE is a more general method as it can be used to emulate any process for which LES has an output. However, a parameterisation based on a pure Random Forest without any embedded dependencies could also be easily developed from the LES training data. Similarly, Linear Fit improved with the Gaussian Process Emulator or an autoconversion (= cloud droplets to rain droplets) dependency could be embedded along with the Random Forest. Thus, there are many possibilities and it is a matter of preference and scope of the study which method will be chosen. Like in **Paper IV** GPE was used to parameterise both updraft velocity and precipitation formation in ECHAM.

When applying the parameterisation in a coarser model in **Paper IV**, it should be noted that the LES domain size (10×10 (km^2)) did not overlap with the grid size of the coarser model ECHAM (100×100 (km)), which mean that such an area nor our simulation time of 3.5 hours does not fully support accounting for the organisation of clouds. However, Prank et al. (2022) state as the size of open cells exceeds 10 (km) the domain allowed to simulate the transition process with sufficient accuracy. Similarly, when applying the precipitation formation emulator in **Paper IV**, there was an issue as the emulator provided a single vertically integrated value for each column but clouds span several model levels in ECHAM. However, the vertical mismatch was resolved by dividing the emulated column precipitation into cloudy levels based on the autoconversion rate calculation of ECHAM (Khairoutdinov and Kogan, 2000). In addition, there were time dimension problems, as the end state of each LES run should represent the design, that is the initial state of each LES run, but the state might drift notably during the LES simulation time of 3.5 hours. Nudging toward the initial state was used to prevent excessive drifting.

It is notable that using model data (here, ECHAM) to generate the parameterisation training data limits the parameterisation to the physics of the source model, which means using limited processes and conditions. There are studies where machine learning parameterisations are developed based on observations (Rodriguez-Galiano et al., 2012; Schneider et al., 2017) but it should be noted that in that case several simultaneous metrics of the cloud meteorological and microphysical state would be needed. During the initial phases of our study, we explored this possibility but noticed that at that time there was a limited number of good quality global observations (i.e. satellite observations) that included both meteorological and aerosol states. Moreover, the parameterisations that would be used within a coarser model (i.e. global/larger model) should only use those input variables that are available in the coarser model.

5 Finding the optimal design: methods and results

Due to the inherent nature of research, there is always room for improvement. Although we obtained good and promising results in **Paper III**, being a proof-of-concept, there are many aspects to make it better. One of these is the method to select the representative sample for initialising the simulations (i.e. *design of experimentation*, or in short: *design*).

The sampling method (Binary Space Partition, BSP see Section 4.2.3) used in **Paper III** and **Paper IV** has room for improvement as the points are chosen stochastically from each partition and therefore no optimisation is used to select a design. The benefit of BSP is that it is simple, representative, inherently incorporates physical constraints as samples are from realistic data, and for example, outliers are not over-represented. However, the sample might not be well spread out while keeping it representative. Here, we show how we could improve the design by making it more spread out and possibly more importantly avoiding too similar sample points. This is done while accounting for the inherent physical constraints and maintaining the design representative and comprehensive. In this application, the constraint describes the interdependence between temperature and moisture. Improved designs are achieved following the methods described in Huang et al. (2021) which is our main literature resource for improving the results. To ease the analysis and intercomparison of design creation methods, the methods of Huang et al. (2021) are given in good detail in Sections 5.5.2, 5.2 and 5.3. The new designs are compared with the BSP.

In deterministic computer experiments, specific software is run on computers as simulations to examine the input/output relationship of complex for example physical, economical, or engineering models (Mak et al., 2018). However, these simulations are often highly time-consuming and computationally expensive. A way to circumvent this is to develop a computationally inexpensive surrogate model that mimics the behaviour of the expensive computer simulations (Santner et al., 2018; Huang et al., 2021). As the input/output relationship is presumed to be complex, it is vital that the design covers the input space (comprehensive) as well as mimics the densities of design features (representative). Often a space-filling design is used to build the experimental design, in which the design points are selected in a well-spread manner across the entire design region $\mathcal{X} \subseteq \mathbb{R}^p$ (Huang et al., 2021).

The optimal design is formulated as a mathematical optimisation problem. There

are several metrics that can be chosen as the objective function for the optimisation problem. Minimax and maximin are the two favoured space-filling measures recommended by Johnson et al. (1990). A minimax design intends to minimise the maximum distance from any point in $\mathcal{X}$ to the nearest design point, while a maximin design maximises the minimum distance between any two design points. Minimax would reduce the number of outlier samples and therefore favour more frequent values, which would be suitable for our application. However, we use a mapping from hypercube to the set of realistic values, which makes evenly distributed maximin design convenient for our case. Our numerical results in Section 5.5 also prove this. In addition, the maximin measure is frequently employed in the literature due to its computational feasibility. Yet, maximin designs are often collapsing, which means that certain design points share the same value in one-dimensional projections. Latin hypercube designs (LHD, see also Section 4.2.2) are built to entail a good projection of each design feature (McKay et al., 1979b) which has also been improved by incorporating it with other space-filling criteria like maximin (Morris and Mitchell, 1995). However, maximin LHDs can only provide good one-dimensional projections and full-dimensional space-fillings (Huang et al., 2021). The maximum projection (MaxPro) designs (Joseph et al., 2015b), on the contrary, are able to attain good space-filling properties on projections to all subsets of design features (Huang et al., 2021).

According to Huang et al. (2021), space-filling design literature often focus on bounded rectangular region $\mathcal{X} = \prod_{d=1}^{p}[a_d, b_d] \subseteq \mathbb{R}^p$. However, in many real-world applications, such as in our cloud simulation case, there is a need to deal with non-rectangular bounded design space:

$$\mathcal{X} = \left\{ x \in \prod_{d=1}^{p}[a_d, b_d] : g_k(x) \leq 0 \quad \forall k = 1, \ldots, K \right\}, \tag{18}$$

where $\{g_k(x) \leq 0\}_{k=1}^{K}$ are arbitrary K inequality constraints (Huang et al., 2021).

In Huang et al. (2021) they cite that there are two main approaches shown in the literature for generating space-filling design in non-rectangular design space. First is to directly use general purpose constrained optimisation techniques (Trosset, 1999; Stinstra et al., 2003; Kang, 2019). However, this method can be computationally expensive and can be limited by the type of constraints and design properties (e.g. projections) it can process. The second method is to have a two-step process.

- *Candidate generation*: construct a large set of uniformly distributed candidates in $\mathcal{X}$.
- *Design construction*: select points from the set of candidates with a chosen criterion.

This approach is flexible as it allows choosing both space-filling and non-collapsing properties in the designs. The way of producing good quality candidate points is

essential and the main challenge of this approach.

In the literature, there are a number of methods discussed for generating candidate points following Huang et al. (2021). One of those is acceptance/rejection sampling on a large set of uniformly distributed points in $[0, 1]^p$, such as a regular grid of points covering the whole p-dimensional space (Pratola et al., 2017), Latin hypercube samples (Wu et al., 2019) as we did earlier, and quasi-random points (Joseph, 2016). However, with a low feasibility ratio, acceptance/rejection sampling is a highly inefficient method. One way to improve is to iterate between acceptance/rejection sampling and candidate augmentation (Draguljić et al., 2012). The multi-step acceptance/rejection can be executed with the help of Simulated Annealing (Kirkpatrick et al., 1983) and a sequence of shrinking regions (Bect et al., 2017). This method is further developed by employing the probabilistic constraint, which leads to the Sequentially Constrained Monte Carlo (SCMC, Golchi and Loeppky (2015)), which is one of the methods used in this study. SCMC has the same weakness as Monte Carlo sampling where samples are often repeated or are too close to each other. If the samples are initially evenly distributed, fewer samples are required to cover the whole space. Having fewer samples but of good quality means that constraint functions are evaluated fewer times, which is valuable when the constraints are computationally expensive. Having complicated and expensive constraints is not a rarity in climate/cloud modelling, which applies to our case too. Minimum energy design (MinED) is a recently developed deterministic sampling method aimed to simulate well-spaced samples for any given distribution (Joseph et al., 2015a, 2019). The MinED is equivalent to the maximin design when the target distribution is uniform. By combining the probabilistic constraints from SCMC to MinED, Huang et al. (2021) proposed constrained minimum energy design (CoMinED) to create good-quality design candidate samples in arbitrarily constrained space.

5.1 Design construction

Following Huang et al. (2021), the definitive goal is to construct an n-point design $\mathcal{D}_n = \{x_i \in \mathcal{X}\}_{i=1}^n$ in $\mathcal{X}$ holding a good design property. We collect a finite set of $N(N \geq n)$ candidate points $\mathcal{C}_N = \{y_j \in \mathcal{X}\}_{j=1}^N$ from the candidate generation step (either adaptive SCMC or CoMinED) that are approximately uniformly distributed in $\mathcal{X}$. Subsequently, here the next step is to find the n samples from the candidate set that maximise a convenient design criterion ψ. Thus, we formally solve

$$\arg \max_{\mathcal{D}_n \subseteq \mathcal{C}_N} \psi(\mathcal{D}_n). \tag{19}$$

If we want to have a maximin design, we define

$$\psi(\mathcal{D}_n) = \min_{x_i, x_j \in \mathcal{D}_n : i \neq j} \|x_i - x_j\|_2, \tag{20}$$

where $\|\cdot\|_2$ is the Euclidean distance. There are multiple ways to solve (19), however, many of them are computationally expensive. Kennard and Stone (1969) proposed an alternative way of solving (19) by using a one-point-at-a-time greedy algorithm. The idea of the greedy algorithm is that, as we have a m-point design $\mathcal{D}_m (m < n)$, we generate the $(m+1)$-th point by

$$x_{m+1} = \arg \max_{x \in \mathcal{C}_{\mathcal{N}} \setminus \mathcal{D}_m} \psi(\mathcal{D}_m \cup \{x\}). \tag{21}$$

5.2 Adaptive Sequentially Constrained Monte Carlo

Following Huang et al. (2021), to improve Markov Chain Monte Carlo sampling from a sequence of shrinking regions, the hard constraint $g(x) \leq 0$ is relaxed with a probabilistic constraint, which leads to the Sequentially Constrained Monte Carlo (SCMC).

The relaxed constraint is formulated according to Golchi and Loeppky (2015) with the probit function

$$\rho_\tau(x) = \Phi(-\tau g(x)), \tag{22}$$

where Φ is the standard normal cumulative distribution function and $\tau \geq 0$ is the parameter that handles the rigidity of the constraint. When the constraint is met, the function ρ_τ assigns a value for x close to 1 and close to 0 when the constraint is not met. With the limit we have

$$\lim_{\tau \to \infty} \rho_\tau(x) = \lim_{\tau \to \infty} \Phi(-\tau g(x)) = \mathbb{1}(g(x) \leq 0). \tag{23}$$

The previous limit (23) can be generalised to multiple inequality constraints $\{g_k(x) \leq 0\}_{k=1}^K\}$ by

$$\rho_\tau(x) = \prod_{k=1}^{K} \Phi(-\tau g_k(x)). \tag{24}$$

The previous Expression (24) sets up the Sequentially Constrained Monte Carlo (SCMC) that replaces the sequence of indicator functions $\{\mathbb{1}_{\mathcal{X}_t}\}_{t=0}^T$ in the subset simulation by the sequence of probabilistic constraint functions $\{\rho_{\tau_t}\}_{t=0}^T$ defined in (24) with and increasing sequence $0 < \tau_0 < \tau_1 < \ldots < \tau_T$, where τ_T is a large constant, for example 10^6. First, the SCMC algorithm in Golchi and Loeppky (2015) provided a pre-fixed normal distribution proposal for the Markov kernel of the MCMC step. In Huang et al. (2021) they state that in this way picking the scale of the normal proposal is challenging for a high-dimensional problem with the small feasible region. Hence, Huang et al. (2021) improved the SCMC method by allowing adaptation of the Markov kernel. The scale (standard deviation) of the normal proposal is adjusted at each iteration (Algorithm 3). According to Huang et al. (2021) the adaptive kernel shows robust performance for the majority of the benchmark problems. Algorithm 3

shows the details of the adaptive SCMC algorithm given in Huang et al. (2021) for generating a large number of uniformly distributed samples from any design space $\mathcal{X}$ with any number of constraints. Before defining the algorithm, let us formulate the design space

$$\mathcal{X} = \{x \in [0,1]^p : g_k(x) \leq 0 \quad \forall k = 1, \ldots, K\}. \tag{25}$$

Algorithm 3 Adaptive Sequentially Constrained Monte Carlo (SCMC) (Huang et al., 2021)

Input: $\{g_k(x) \leq 0\}_{k=1}^K$ defines the design space $\mathcal{X}$ (Equation 25), and the increasing sequence of rigidity parameters $0 = \tau_0 < \tau_1 < \ldots < \tau_T$.

1: Simulate the initial M samples $\{x_m^{(0)}\}_{m=1}^M$ from $[0,1]^p$.
2: **for** $t = 1, \ldots, T$ **do**
3: **Weighting:** compute the importance weights $w_m^{(t)} = \rho_{\tau_t}(x_m^{(t-1)}) / \rho_{\tau_{t-1}}(x_m^{(t-1)})$, for all $m = 1, \ldots, M$, where $\rho_\tau(\cdot)$ is defined in Equation (24). Normalise the weights by $\overline{\mathrm{w}}_m^{(t)} = w_m^{(t)} / \sum_{i=1}^M w_i^{(t)}$ for all $m = 1, \ldots, M$.
4: **Resample:** draw M independent samples $\{y_m^{(0)}\}_{m=1}^M$ from $\sum_{m=1}^M \overline{\mathrm{w}}_m^{(t)} \delta_{x_m^{(t-1)}}$, where δ_x is the Dirac measure for any $x \in \mathcal{X}$.
5: **Sampling:** for $m = 1, \ldots, M$ draw $x_m^{(0)} \sim K_{\sigma^{(t)}}(y_m^t, \cdot)$, where $K_{\sigma^{(t)}}(y_m^t, \cdot)$ is a Markov kernel with target distribution ρ_{τ_t}, and adaptive scale $\sigma^{(t)}$ is 75th percentile (=Q_3) of $\{\min_{j \neq m} \|x_m^{(t-1)} - x_j^{(t-1)}\|\}_{m=1}^M$.
6: **end for**

Output: all particles $\{x_m^{(t)}\}_{m=1}^{M\ T}_{t=1}$ that are in $\mathcal{X}$.

The adaptive SCMC algorithm is defined in the Algorithm 3, where Q_3 means the 75th percentile of the distribution. The definitions of Dirac measure and Markov kernel are detailed in Appendices 7.1.3 and 7.1.6, respectively.

5.3 Constrained Minimum Energy Design

In this section, we explain how the Constrained Minimum Energy Design (CoMinED) algorithm provides a design. Following Joseph et al. (2015a), let us first define the *minimum energy design* (MinED). Here, the analogy is to have a physical system of electrically charged particles inside a box to motivate the proposed design, which is called minimum energy design.

According to the analogy, the particles have an explicit charge that is vital for mimicking the underlying distributions. If the charge of particles is of the same sign, they will repel each other and occupy positions inside the box in a way that minimises the total potential energy. Concerning the application, the box is the experimental region, each position taken by the charged particles is a design point and the charge

represents the experimental response (= observed or measured quantity). Therefore, all the positions occupied by the particles form the experimental design. As the design is obtained by minimising the potential energy, it is called minimum energy design.

The analogy of the repulsion principle requires all the particles to have the same sign of the charge. Without loss of generality, the particle charge is assumed to be positive. Let us define the potential energy $E_{i,j}$ between ith and jth particle:

$$E_{i,j} \propto \frac{q(x_i)q(x_j)}{\|x_i - x_j\|_2}, \tag{26}$$

where $q(x_i)$ is the charge of the particle at ith design point x_i. The charge of the particles $q(\cdot)$ can be chosen according to the objective (Joseph et al., 2015a). The different choices of charge of the particles are discussed further in (Joseph et al., 2015a). Equation (26) can be directly compared with Coulomb's law which states the magnitude of the electrostatic force of attraction or repulsion between two point charges, and the proportionality is defined with Coulomb's constant (Wikipedia, b).

Here, following Joseph et al. (2015a) and Huang et al. (2021) the formal definition of MinED is described in Definition 5.3.1.

***Definition* 5.3.1** (Minimum Energy Design, Joseph et al. (2015a))**.** *Let us have π as the target probability density function. We define that an n-point minimum energy design of π is the optimal solution of*

$$\arg\min_{\mathcal{D}_n \in \mathbb{D}_n} \sum_{\substack{x_i, x_j \in \mathcal{D}_n \\ i \neq j}} \frac{q(x_i)q(x_j)}{\|x_i - x_j\|_2}, \tag{27}$$

where $\mathbb{D}_n = \{\{x_i\}_{i=1}^n : x_i \in \mathbb{R}^p\}$ is the set of all unordered n-tuple in $\mathbb{R}^p$ and $q(\cdot) = 1/\pi^{1/(2p)}(\cdot)$ is the charge function.

In MinED, with the given proposed charge function, the limiting distribution of the design points converges to π.

Yet, the optimisation problem (27) is hard to solve and numerically unstable. To bypass the problem, Joseph et al. (2019) noticed that (27) is related to

$$\arg\min_{\mathcal{D}_n \in \mathbb{D}_n} \left[\sum_{\substack{x_i, x_j \in \mathcal{D}_n \\ i \neq j}} \left(\frac{q(x_i)q(x_j)}{\|x_i - x_j\|_2} \right)^k \right]^{1/k}, \tag{28}$$

for $k > 0$. As $k \to \infty$, the optimisation problem converges to

$$\arg\min_{\mathcal{D}_n \in \mathbb{D}_n} \max_{\substack{x_i, x_j \in \mathcal{D}_n \\ i \neq j}} \frac{q(x_i)q(x_j)}{\|x_i - x_j\|_2}. \tag{29}$$

If we substitute $q(\cdot) = 1/\pi^{(1/2p)}(\cdot)$ into (29) and apply logarithmic function, we get

$$\arg\max_{\mathcal{D}_n \in \mathbb{D}_n} \min_{\substack{x_i, x_j \in \mathcal{D}_n \\ i \neq j}} \frac{1}{2p} \log \gamma(x_i) + \frac{1}{2p} \log \gamma(x_j) + \log \|x_i - x_j\|_2. \tag{30}$$

It follows from the objective function of (30) that the chosen design points try to be as far as possible while focused in the high-density regions (Huang et al., 2021), which is valuable for creating a parameterisation.

Following Huang et al. (2021), we have an unnormalised probability density function γ proportional to π in some non-rectangular bounded space $\mathcal{X} = \{x \in [0,1]^p : g_k \leq 0 \forall k = 1, \dots K\}$, and we have a generalised distance

$$\|u\|_s = \left(\frac{1}{p} \sum_{l=1}^{p} |u_l|^s \right)^{1/s} \text{(Joseph et al., 2019).} \tag{31}$$

Thus, we want to create the relevant Minimum Energy Design and the optimisation problem is

$$\arg\max_{\mathcal{D}_n \in \mathbb{D}_n^{\mathcal{X}}} \min_{\substack{x_i, x_j \in \mathcal{D}_n \\ i \neq j}} \frac{1}{2p} \log \gamma(x_i) + \frac{1}{2p} \log \gamma(x_j) + \log \|x_i - x_j\|_s, \tag{32}$$

where $\mathbb{D}_n^{\mathcal{X}} = \{\{x_i\}_{i=1}^n : x_i \in \mathcal{X}\}$ is the set of all unordered n-tuple in $\mathcal{X}$. As constrained optimisation problem is often difficult to solve, especially in the case of nonlinear constraints, Huang et al. (2021) simplified the optimisation problem (32) by introducing the probabilistic relaxation ρ_τ defined in (24) for the inequality constraints $\{g_k\}_{k=1}^K$, which leads to the constrained minimum energy design (CoMinED) described in the Definition 5.3.2.

***Definition* 5.3.2** (Constrained Minimum Energy Design)**.** *Let us have the $\gamma \propto \pi$ as the target unnormalised probability density function. An n-point minimum energy design of π in an arbitrary non-rectangular bounded space $\mathcal{X} = \{x \in [0,1]^p : g_k(x) \leq \forall k = 1, \dots, K\}$ is*

$$\arg\max_{\mathcal{D}_n \in \mathbb{D}_n^u} \min_{\substack{x_i, x_j \in \mathcal{D}_n \\ i \neq j}} \frac{1}{2p} \log \widetilde{\gamma_\tau}(x_i) + \frac{1}{2p} \log \widetilde{\gamma_\tau}(x_j) + \log \|x_i - x_j\|_s, \tag{33}$$

where $\|\cdot\|_s$ is the distance measure defined in (31), and

$$\widetilde{\gamma_\tau} = \gamma(\cdot)\rho_\tau(\cdot) = \gamma(\cdot) \prod_{k=1}^{K} \Phi(-\tau g_k(\cdot)), \tag{34}$$

where τ controls the rigidity of the constraints (Huang et al., 2021).

To point out, as $\tau \to \infty$, (33) converges to (32). In practice, the parameter $\tau = 10^6$ is sufficient to reach the limit numerically if the constraints are properly scaled (Huang et al., 2021).

According to Huang et al. (2021), Joseph et al. (2015a) and Joseph et al. (2019) solving MinED optimisation directly using nonlinear programming solver is difficult and computationally expensive. To circumvent this, the proposal is to first generate the design from a set of candidate samples, and second, apply Simulated Annealing (see. Appendix 7.1.7) on τ by starting with a lighter problem and slowly increasing the rigidity of the constraints (applied also in the SCMC). Let us have a T step – Simulated Annealing, and we will define the increasing sequence of rigidity parameters $0 < \tau_0 < \tau_1 < \ldots < \tau_T = 10^6$. At each step, the n-point intermediate CoMinED is generated as follows. Let us have τ_t as the rigidity parameter, $\mathcal{C}^t = \{y_j^t\}_{j=1}^{N_t}$ as the candidate samples, and $\mathcal{D}^t = \{x_i^t\}_{i=1}^n \subseteq \mathcal{C}^t$ as the CoMinED at the t-th step. To construct $\mathcal{D}^{t+1}$ in a process called *adaptive lattice grid refinement* (ALGR), first the candidate samples are augmented to $\mathcal{C}^{t+1}$ by including the linear combinations of nearby points in $\mathcal{D}^t$, and then the one-point-at-a-time greedy algorithm (21) is applied to solve (33) with τ_{t+1} as the rigidity parameter. Details of the CoMinED algorithm are given in Algorithm 4.

5.4 Applying the CoMinED and the adaptive SCMC to modelling cloud processes

As previously stated, the adaptive SCMC and the CoMinED are represented in hypercube $\mathcal{H} = [0,1]^p$ and the constraints are of format $\{g_k(x) \leq 0\}_{k=1}^K$. In combination, the design space $\mathcal{X}$ is defined in Equation (25).

Concerning our application, we can scale values from the hypercube values to representative cloud state values or the other way around. While scaling the values, the individual variable density distribution can be reproduced according to the source data. We have close to 5.9 million samples ($\mathcal{O}(10^6)$) from ECHAM as source data (see Section 4.1) and the range of single design feature does not span more than three orders of magnitude (= $\mathcal{O}(10^3)$, see e.g. Figure 10 in Section 5.5.2). Since we have a mapping from a uniform distribution (a basis of adaptive SCMC and CoMinED) to realistic ECHAM values, we would have at least three significant digits, which is a good numerical accuracy concerning the application.

To scale hypercube values to representative cloud parameters or in the opposite direction, we used the following algorithmic approach. For each design feature $d = 1, \ldots, p$ (see Tables 1, 2) we can define the value x within the hypercube as

$$x \in \mathcal{H}_d = [0,1] \subseteq \mathcal{H}^p, \tag{38}$$

and similarly the realistic cloud state value ξ as

$$\xi \in \mathcal{R}_d = [a_d, b_d] \subseteq \mathcal{R}^p \subseteq \mathbb{R}^p. \tag{39}$$

Algorithm 4 n-point Constrained Minimum Energy Design (Huang et al., 2021)

Input: $\{g_k(x) \leq 0\}_{k=1}^K$ defining the design space $\mathcal{X}$ (Equation 25), the increasing sequence of rigidity parameter $0 = \tau_0 < \tau_1 < \ldots < \tau_T$, and the number of nearest neighbours $\mathcal{Q}$ suggested for candidate augmentation.

1: **Initialisation**: as the initial set of candidate sample $\mathcal{C}^1$ generate $N_1 > n$ (prime number) lattice points $\{y_j^{(1)}\}_{j=1}^{N_1}$ from $[0,1]^p$.

2: **for** $t = 1, \ldots, T$ **do**

3: **Construction**: solve (33) with $\tau = \tau_t$ with one-point-at-a-time greedy algorithm (21) to get the CoMinED $\mathcal{D}^t = \{x_i^t\}_{i=1}^n$, namely, with $\{x_1^t, \ldots, x_m^t\}, x_{m+1}^t$ is given by

$$x_{m+1}^t = \arg \max_{x \in \mathcal{C}^t \setminus \{x_l^t\}_{l=1}^m} \min_{i=1,\ldots,m} \frac{1}{2p} \sum_{k=1}^K \log \Phi(-\tau g_k(x)) + \frac{1}{2p} \sum_{k=1}^K \log \Phi(-\tau_t g_k(x_i)) + \log \|x_i - x_j\|_s. \tag{35}$$

4: **if** $t < T$ **then**

5: **Augmentation**: expand the set of candidate samples $\mathcal{C}^{t+1} = \mathcal{C}^t \cup \widetilde{\mathcal{C}^t}$ where $\widetilde{\mathcal{C}^t}$ is the set of linear combinations of nearby points in $\mathcal{D}^t$. The $\widetilde{\mathcal{C}^t}$ is constructed in the following.

6: **for** $i = 1, \ldots, n$ **do**

7: find the $\mathcal{Q}$ nearest neighbours of x_i^t in $\mathcal{D}^t$.

8: for each nearest neighbour $\widetilde{x}_{i,q} (q = 1, \ldots, \mathcal{Q})$, compute the mid-point

$$\widetilde{y}_{i,q}^{(m)} = x_i + \frac{1}{2}(\widetilde{x}_{i,q} - x_i) = \frac{x_i + \widetilde{x}_{i,q}}{2}, \tag{36}$$

9: and the reflection mid-point

$$\widetilde{y}_{i,q}^{(r)} = x_i - \frac{1}{2}(\widetilde{x}_{i,q} - x_i) = \frac{3x_i - \widetilde{x}_{i,q}}{2}. \tag{37}$$

10: Update $\widetilde{\mathcal{C}^t} = \widetilde{\mathcal{C}^t} \cup \{\widetilde{y}_{i,q}^{(m)}, \widetilde{y}_{i,q}^{(r)}\}$.

11: **end for**

12: Remove repeated points in $\widetilde{\mathcal{C}^t}$, keep points in $\widetilde{\mathcal{C}^t}$ that do not exist in $\mathcal{C}^t$.

13: **end if**

14: **end for**

Output: feasible candidate samples $\{y \in \mathcal{C}^T : y \in \mathcal{X}\}$, and the CoMinED: $\mathcal{D}^T$.

Let us create a lookup table $\mathcal{T}_d$ for each design feature d. First, the samples are sorted in an ascending order

$$\mathcal{T}_d = \{r_0 \leq r_1 \leq \cdots \leq r_i \leq \cdots \leq r_M\}, \tag{40}$$

where i is the index of the sample and M is the total number of samples. To reduce the number of identical values in source data, an ordinal number is appended to the decimal part. We tested that in practice, adding the ordinal number does not impact the numerical results.

Scaling values from $[0, 1]$ to realistic cloud state values goes according to Algorithm 5. In the Algorithm 5 the R is a function that rounds the input to the closest

Algorithm 5 Scaling hypercube values ($\mathcal{H}_d$) to representative cloud state values ($\mathcal{R}_d$)

Input: value $x \in \mathcal{H}_d = [0, 1]$.

1: M is the length of the lookup table $\mathcal{T}_d$.
2: For hypercube value x the corresponding cloud parameter value is ξ and the index of the lookup table is i.
3: $i \leftarrow R(x \cdot \lambda)$.
4: $\xi \leftarrow i$:th value of lookup table $\mathcal{T}_d$.

Output: a representative cloud parameter value $\xi \in [a_d, b_d] = \mathcal{R}_d \subseteq \mathbb{R}$.

integer, and λ is the total number of samples (close to 5.9 million). It would be possible to develop a more accurate method to find the realistic value ξ that corresponds with x but this method of finding the arithmetic middle point is fast and accurate enough as M is large, in the order of millions. One other solution would be to fit a relevant function between the indexes and their corresponding realistic values. The Algorithm 5 represents a function f_d, which in theory is a bijection due to the added ordinal numbers. However, due to the limitations of computers representing float numbers, there exist identical values, making it monotonically increasing.

$$f_d : \mathcal{H}_d \to \mathcal{R}_d. \tag{41}$$

However, we want every value within $[0, 1]$ to correspond to one index value i, which leads to one realistic value ξ. Taking into account the limitations of computers, let us define an inverse function

$$f_d^{-1} : \mathcal{R}_d \to \mathcal{H}_d. \tag{42}$$

in the Algorithm 6. From the Algorithm 6 follows that if there are several identical values, the index values are given at the upper end. As determined by test results, having the index values at the lower end does not impact the numerical results. Although, the function (41) is not a perfect bijection, the practical implications following our inverse function are negligible as we have an extensive number of samples

Algorithm 6 Scaling representative cloud parameter values ($\mathcal{R}_d$) to hypercube values ($\mathcal{H}_d$)

Input: a cloud parameter value $\xi \in [a_d, b_d] = \mathcal{R}_d \subseteq \mathbb{R}$.

1: M is the length of the lookup table $\mathcal{T}_d$.

2: With binary search algorithm (see Appendix 7.1.1) find the lower index $(i-1)$ and the upper index $(i+1)$ from the lookup table $\mathcal{T}_d$, for which $r_{i-1} \leq r_i < r_{i+1}$.

3: $x \leftarrow \frac{((i-1)+(i+1))/2}{M} = \frac{i}{M}$.

Output: value $x \in \mathcal{H}_d = [0, 1]$.

(close to six million), which gives us sufficient numerical accuracy. If we want to create a practically perfect bijection, the source data should be carefully sampled and stored in a suitable data format to prevent identical values.

Algorithm 5 is needed to get the designs obtained with both adaptive SCMC and CoMinED algorithms (Algorithms 3 and 4, respectively) as the constraint function g_1 is defined in Equation 43.

$$\begin{aligned} & q_t \leftarrow \text{solve\_rw\_lwp}(\theta_L, \text{LWP}, H_{\text{PBL}}) \\ & q_t - \Delta q_t \geq 1 \\ & g_1(x) = \Delta q_t - q_t + 1 \leq 0, \end{aligned} \tag{43}$$

where solve_rw_lwp is an iterative method to find the q_t (total water in planetary boundary layer) based on θ_L, LWP, H_{PBL}, and g_1 is the format of the constraint that the adaptive SCMC and CoMinED require.

With Algorithms 5 and 6, and constraint function (43) we have all we need to apply adaptive SCMC and CoMinED to our application of emulation of cloud processes. With Algorithm 6 it is possible to scale new BSP results along with the BSP results in **Paper III** to hypercube and also to get the value of the optimisation objective function (i.e. optimisation measure) for those designs in order for comparing all the results.

5.5 Numerical design results

Here, we create designs for four different sets (SB night, SB day, SALSA night, SALSA day) with BSP, adaptive SCMC and CoMinED methods by using both maximin and MaxPro optimisation measures. Maximin measure is defined with Equations (19) and (20). MaxPro measure (Joseph et al., 2015b) is given as

$$\arg\min_{\mathcal{D}_n \subseteq \mathcal{C}_N} \psi(\mathcal{D}_n) = \left\{ \frac{1}{\binom{n}{2}} \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \frac{1}{\prod_{l=1}^{p} (x_{il} - x_{jl})^2} \right\}^{(1/p)}. \tag{44}$$

Regarding Maxpro measure, for any l, if $x_{il} = x_{jl}$ for $i \neq j$, then $\psi(\mathcal{D}) = \infty$. Consequently, the Maxpro design minimising $\psi(\mathcal{D})$ must have n distinct levels for each feature. Since the denominator in Equation 44 has products of squared distances from all the features, any two design points cannot get close to each other in any of the projections. Hence, the design minimising $\psi(\mathcal{D})$ tends to maximise its projection capability in all subspaces of features (Joseph et al., 2015b).

SB (Seifert & Beheng) and SALSA refer to the used cloud microphysics scheme (see Section 3.2). SB and SALSA have different numbers of design features. Night or day indicates whether the cloud evolution is simulated when the Sun is either below or above the horizon. Daytime simulations use an additional design feature of cosine of solar zenith angle. Design variables are elaborated in Chapter 4.

5.5.1 Setup for design creation

We created the BSP, adaptive SCMC and CoMinED designs with 53, 101, 199, 307, 401 and 499 design points. The design creation algorithms are not limited to only prime numbers. The main idea of selecting these design points is to have enough data points to draw conclusions and to cover the original design points 135, 150, and 500 shown in **Paper III**. Thus, BSP designs are recreated as the purpose is to compare these different design creation methods. The designs from **Paper III** are shown along with these new designs as a reference. As in Huang et al. (2021), in all the numerical results CoMinED is run with $s = 2$ for the distance measure (31), namely the Euclidean distance. Likewise, let N_1 be the number of the initial candidate samples to be largest prime number that is less than the product of the number of CoMinED points n and the number of neighbours to be considered for candidate augmentation $\mathcal{Q}$. For SB design results, the $\mathcal{Q}$ parameter was set to 19. For SALSA results, the parameter $\mathcal{Q}$ was set to 23. The choice is based on Huang et al. (2021) where they suggest that $\mathcal{Q}$ is any number between $2p + 1$ and $3p + 1$ depending on the available computational resources. We set $T = 8$ and the set of rigidity parameters $\{\tau_t\}_{t=0}^{8} = \{0, e^1, e^2, e^3, e^4, e^5, e^6, e^7, 10^6\}$ similarly as in Huang et al. (2021) as these parameters show stable performance on both CoMinED and adaptive SCMC. Additionally, the set of rigidity parameters shows robust performance for dimensions from 2 to 13 (Huang et al., 2021), which is applicable to our case.

The number of samples per iteration in adaptive SCMC is the same as in Huang et al. (2021), which is

$$M = \max\{n\mathcal{Q}, \lceil \frac{N_T}{T+1} \rceil\}, \tag{45}$$

so that M is larger than the number of initial samples of CoMinED ($N_1 < nQ$) and the total number of adaptive SCMC samples $M(T + 1)$ is larger than the total CoMinED samples N_T. This choice made in Huang et al. (2021) puts CoMinED in a slightly disadvantageous position compared to SCMC and their intention was to highlight the effectiveness of CoMinED over adaptive SCMC. Although this is not

our objective since the constraint evaluations of both algorithms cannot be set equal, this choice keeps both methods reasonably well comparable.

As previously stated, the objective of Huang et al. (2021) study was to show the superior attributes of CoMinED compared to adaptive SCMC. However, the aim of this study is to find an improved design compared to BSP, and the simplest approach was to use the same hyperparameters as in Huang et al. (2021). Furthermore, there is no indication that the hyperparameters would be ineffective or nonfunctional.

The adaptive SCMC is iterated only once to enable a more meaningful comparison with the BSP method which is also a stochastic method. In Huang et al. (2021) they use 50 iterations for adaptive SCMC but we obtained good results already with one iteration (see Section 5.5.2). This can be seen with the six designs for each set that create a clear pattern. Additionally, 50 repetitions would require quite a high computational cost (several days with a high number of design points) due to the complexity of our constraint function. CoMinED is a deterministic algorithm using an initial candidate of lattice points and the one-point-at-a-time greedy algorithm for design construction (Huang et al., 2021), therefore single iteration is sufficient.

5.5.2 Design comparison results

Figure 6 shows the results of the maximin measure relative to the highest measure of all the designs. The maximin measure can be interpreted so that for each number of design points the higher the measure the better the result, as the target is to maximise the minimum distance. As the number of design points increases, the minimum distance is smaller but still, the higher measure is better since maximin tries to fill the whole space (= space-filling design). The maximin measure might not be the best choice considering that we want to develop a cloud emulator with the known density function as source data. However, as we apply the maximin with a function from hyperspace to realistic values (Algorithms 6 and 6), maximin ensures good coverage of the hyperspace, which leads the focus on the densest (i.e. vital) subspaces of the source data.

The feasibility ratio, namely the percentage of total samples that are feasible, is 0.6420 for all design sets as the constraint is independent of the microphysical scheme and solar zenith angle.

The CoMinED and adaptive SCMC display significant improvements compared to the non-optimised BSP results. The CoMinED and adaptive SCMC present almost equally good results. The stochastic character of SCMC might be the most important reason why neither CoMinED nor adaptive SCMC is consistently the best method. However, with a small number of design points the adaptive SCMC is the best choice in all sets except with SB night (Figure 6a). As the number of design points gets larger the difference between CoMinED and adaptive SCMC becomes smaller, and the difference is a bit larger with SB microphysics (Figure 6a and 6b) in favour for CoMinED. The CoMinED and adaptive SCMC also show an exponential decrease

of maximin measure as the number of design points increases. Adaptive SCMC results would probably improve with additional iterations. Yet, our main objective is to find alternatives for BSP. Thus, the chosen number of iterations is sufficient to draw the conclusion that both SCMC and CoMinED are improvements over BSP. In contrast, the BSP results show the expected random behaviour. However, BSP results might have identical measures within a design set since for research and repeatability purposes the BSP was set up to use a specific random seed number, which leads to having partitions that different designs share and therefore share the same measures. This can be seen with BSP for example in Figure 6d with design points 401 and 499. This same notion applies to Figures 7, 8 and 9.

Figure 7 illustrates the results of how designs are valued with MaxPro measure (Equation (44)). Again, both adaptive SCMC and CoMinED are optimised against the measure, and BSP design is only evaluated using the measure without any optimisation. With MaxPro, the better the design, the smaller the value. In addition, as the number of design points increases the MaxPro measure increases. Figure 7 tells more or less the same story as with maximin measure (Figure 6), namely that both CoMinED and adaptive SCMC are an improvement over BSP. However, in some cases BSP result is close to the other methods, for example in Figure 7b with 53 design points. Also, BSP results seem to be even more irregular with MaxPro than with the maximin measure. It is noteworthy to see that the MaxPro measure does not exist for **Paper III** (BSP) with SB Day. This is because the MaxPro measure is numerically unstable as design points with features too close to each other will yield to not-a-number value (NaN). Here, CoMinED and adaptive SCMC results are more consistently closer to each other than in maximin results (Figure 6).

Figures 8 and 9 show fill distance results for both maximin and MaxPro design. *Fill distance* is the largest distance of any point in $\mathcal{X}$ to the closest feasible samples. It is another metric to assess how well the algorithm explores the feasible region completely (Huang et al., 2021). The smaller the fill distance the better. Similarly, as in (Huang et al., 2021), the fill distance is approximated numerically with feasible samples from acceptance/rejection sampling on a large set ($2^{14} = 16384$) of Sobol points in unit hypercube (See Appendix 7.1.8). The number of Sobol points was chosen to be slightly above 10^4, which is the number used in Huang et al. (2021). Also, the Python function that we used (SciPy Sobol) was numerically more stable when using exponents of 2 to generate the Sobol points. These fill distance results seem to be quite irregular, however, adaptive SCMC seems to be performing well and holds the lowest (i.e. best) fill distance value in most of the cases, which is even more evident when considering MaxPro in Figure 9. With SALSA microphysics both maximin and MaxPro fill distances tend to be higher which probably relates most to the higher number of design features. Fill distances seem to be gradually increasing as the number of design points increases, which is clearer with maximin.

Figures 10 and 11 show the individual design feature distributions. Here, the idea is to see whether the design creation algorithms reproduce the distributions of

the source data. In both figures, all distributions are from the SB day set except for those design features that are SALSA specific (sub-figures g, h, i, j). The number of design points in all sub-figures is 499 except **Paper III** results with 500 design points with SB day and 150 with SALSA day. **Paper III** results are shown as a side-note since the focus is on method intercomparison where the number of sample points can be set equal. We can observe from Figures 10 and 11 that the designs closely replicate the original distributions (i.e. source data from ECHAM-HAMMOZ, see Chapter 4.1). However, there are higher peaks in the original distribution associated with the distributions of aerosol variables (sub-figures f, g, h, i, j). Also, H_{PBL} is not reproduced in the same way as in the original data. However, the source data shows a known numerical characteristic with H_{PBL} and it is better if it is not reproduced exactly. The numerical characteristic is caused by grid vertical layers in ECHAM that cause the high peaks in H_{PBL} distribution. In reality, stratocumulus clouds present a much smoother distribution of cloud top heights. In **Paper III** noise was added to the H_{PBL} to enable the desired smother distribution. Maximin distributions (Figure 10) do not show significant differences over MaxPro distributions (Figure 11).

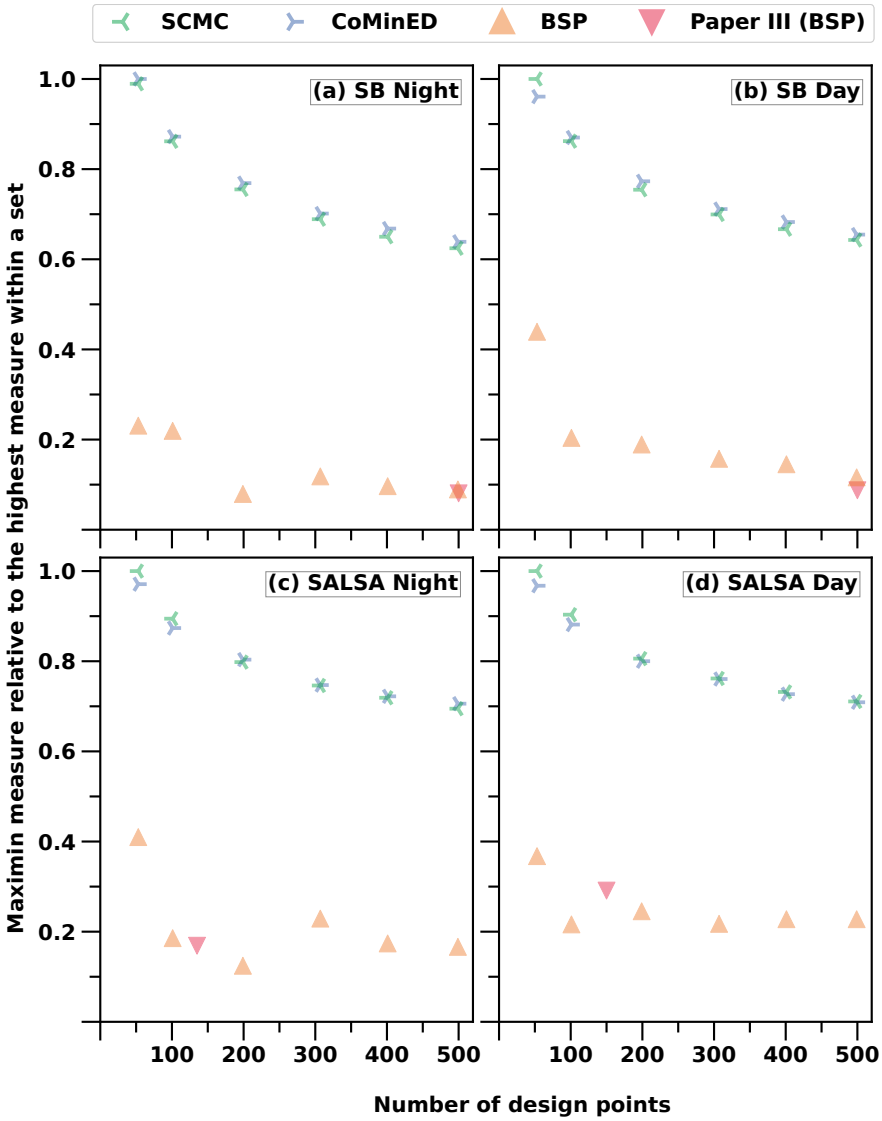

Figure 6. Maximin measures for all designs (Adaptive SCMC, CoMinED, BSP, **Paper III** (BSP)) in all simulation sets (SB Night, SB Day, SALSA Night, SALSA Day). Adaptive SCMC and CoMinED are optimised against the maximin measure. BSP results (both new and the ones from **Paper III**) are only measured with the maximin measure (i.e. the BSP designs are not optimised). Measure is calculated in hypercube $[0, 1]^p$ where p is the dimension of the set.

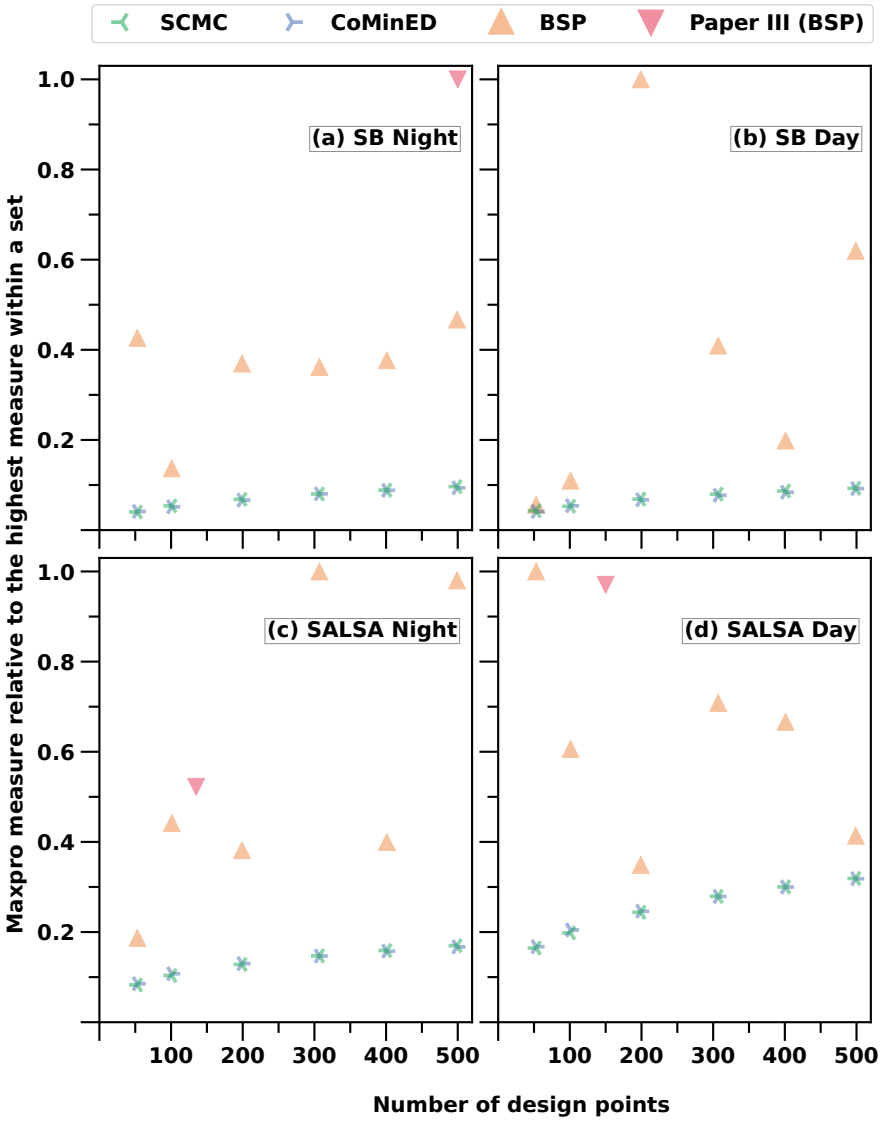

Figure 7. MaxPro measures for all designs (Adaptive SCMC, CoMinED, BSP, **Paper III** (BSP)) in all simulation sets (SB Night, SB Day, SALSA Night, SALSA Day). Adaptive SCMC and CoMinED are optimised against the MaxPro measure. BSP results (both new and the ones from **Paper III**) are only measured with the MaxPro measure (i.e. the BSP designs are not optimised). Measure is calculated in hypercube $[0, 1]^p$ where p is the dimension of the set. **Paper III** (BSP) with SB Day (b) is not shown as it yields NaN value.

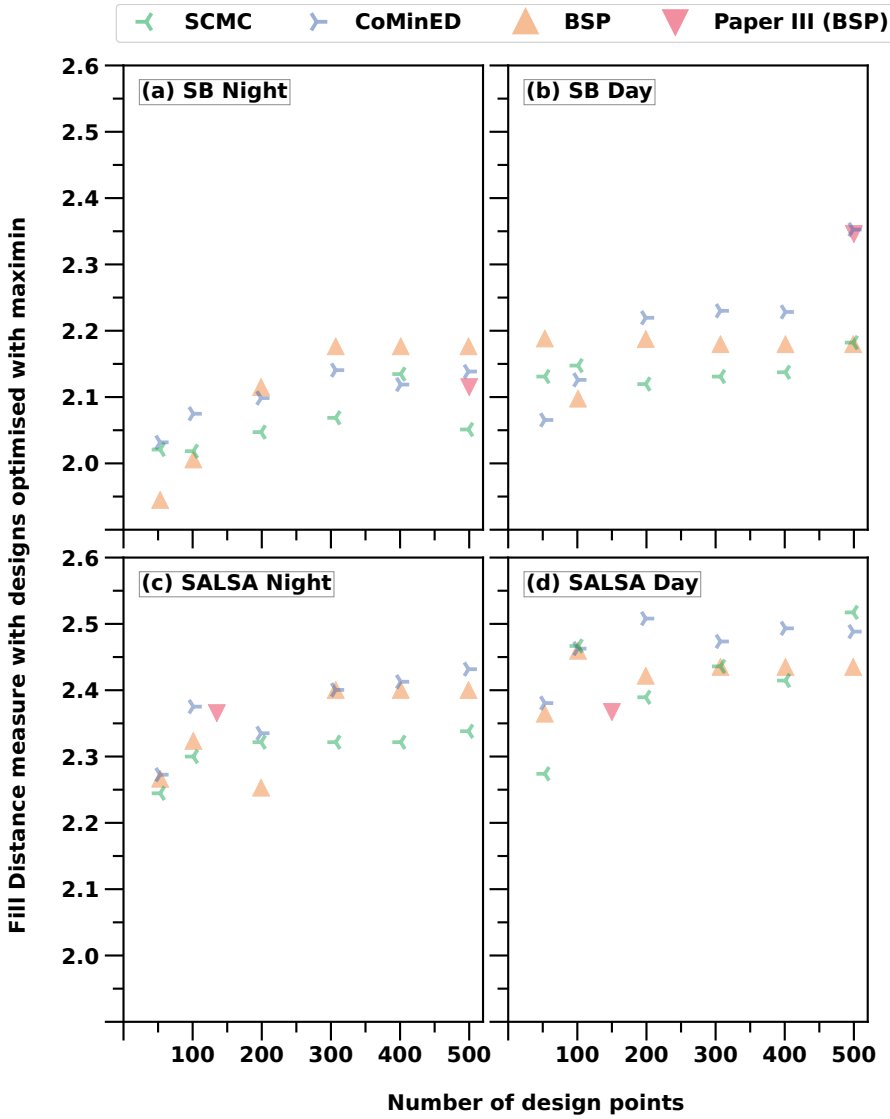

Figure 8. Fill distances for all designs (Adaptive SCMC, CoMinED, BSP, **Paper III** (BSP)) in all simulation sets (SB Night, SB Day, SALSA Night, SALSA Day). Fill distance is the largest distance of any point in $\mathcal{X}$ to the closest feasible samples. Adaptive SCMC and CoMinED are optimised against the maximin measure. Fill distance is calculated in hypercube $[0, 1]^p$ where p is the dimension of the set.

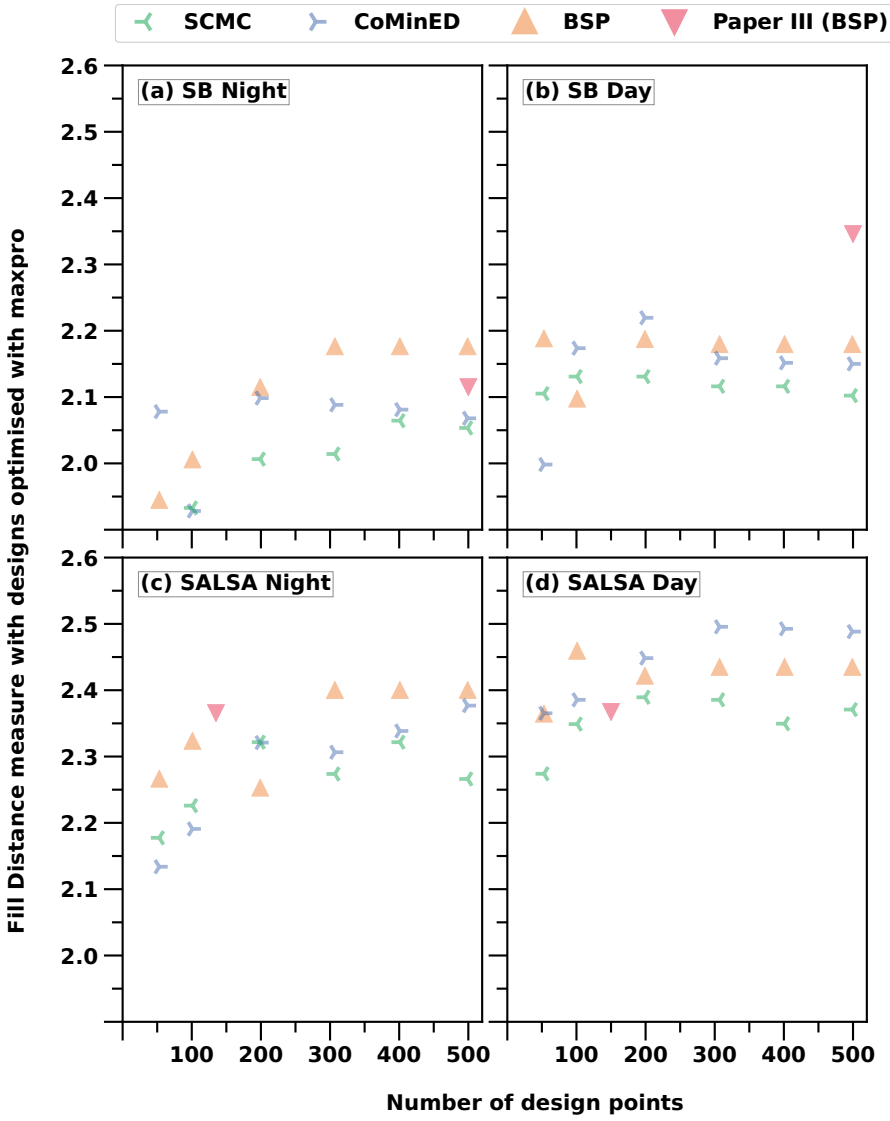

Figure 9. Fill distances for all designs (Adaptive SCMC, CoMinED, BSP, **Paper III** (BSP)) in all simulation sets (SB Night, SB Day, SALSA Night, SALSA Day). Fill distance is the largest distance of any point in $\mathcal{X}$ to the closest feasible samples. Adaptive SCMC and CoMinED are optimised against the MaxPro measure. Fill distance is calculated in hypercube $[0, 1]^p$ where p is the dimension of the set.

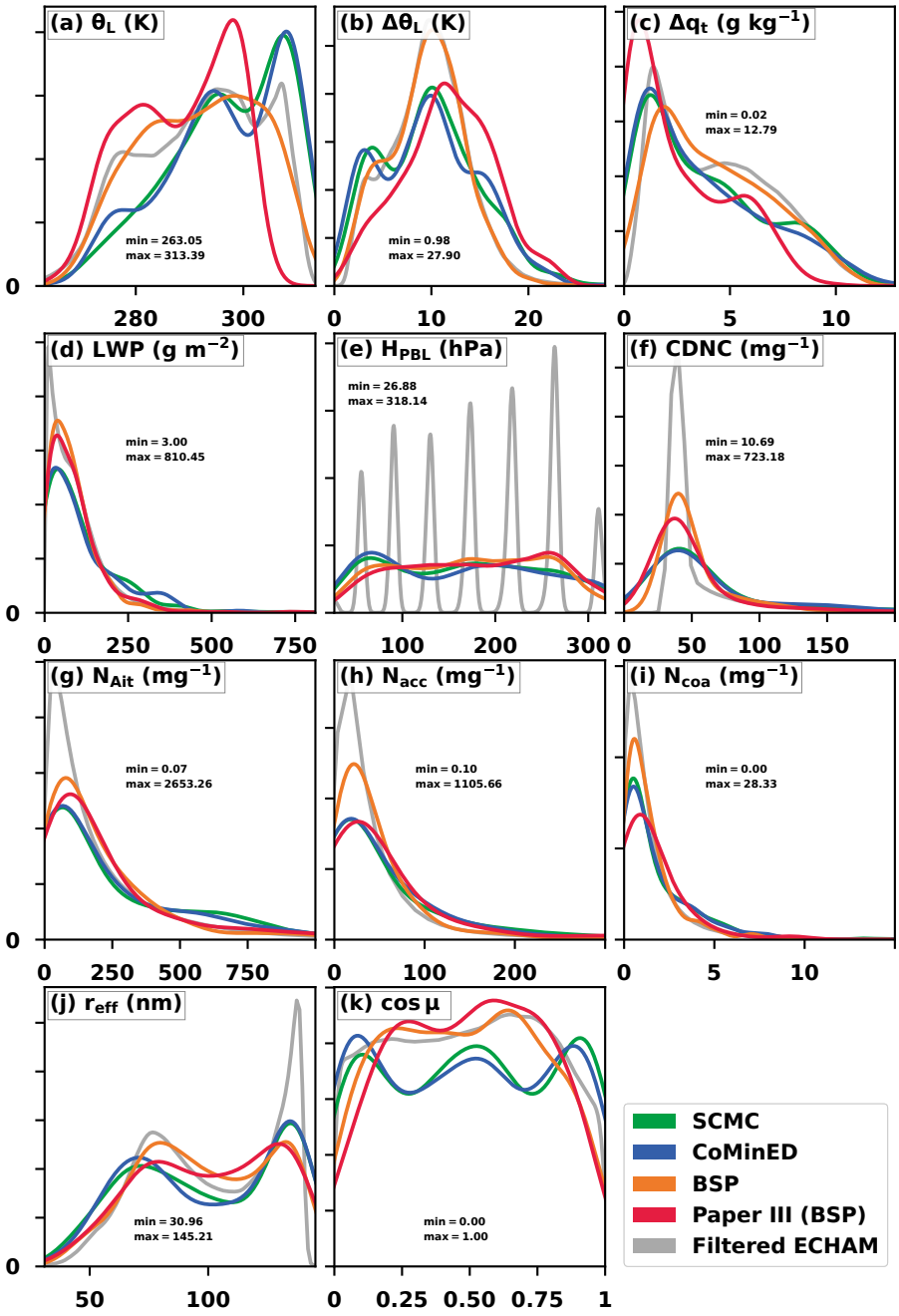

Figure 10. Design feature distribution for designs measured with maximin measure and created with adaptive SCMC, CoMinED, BSP (both new results and those from **Paper III**) and Filtered ECHAM (source data). The number of design points in all sub-figures is 499 except **Paper III** results with 500 design points with SB day and 150 with SALSA day. All distributions are from the SB day set except for those design features that are SALSA specific (sub-figures g, h, i, j).

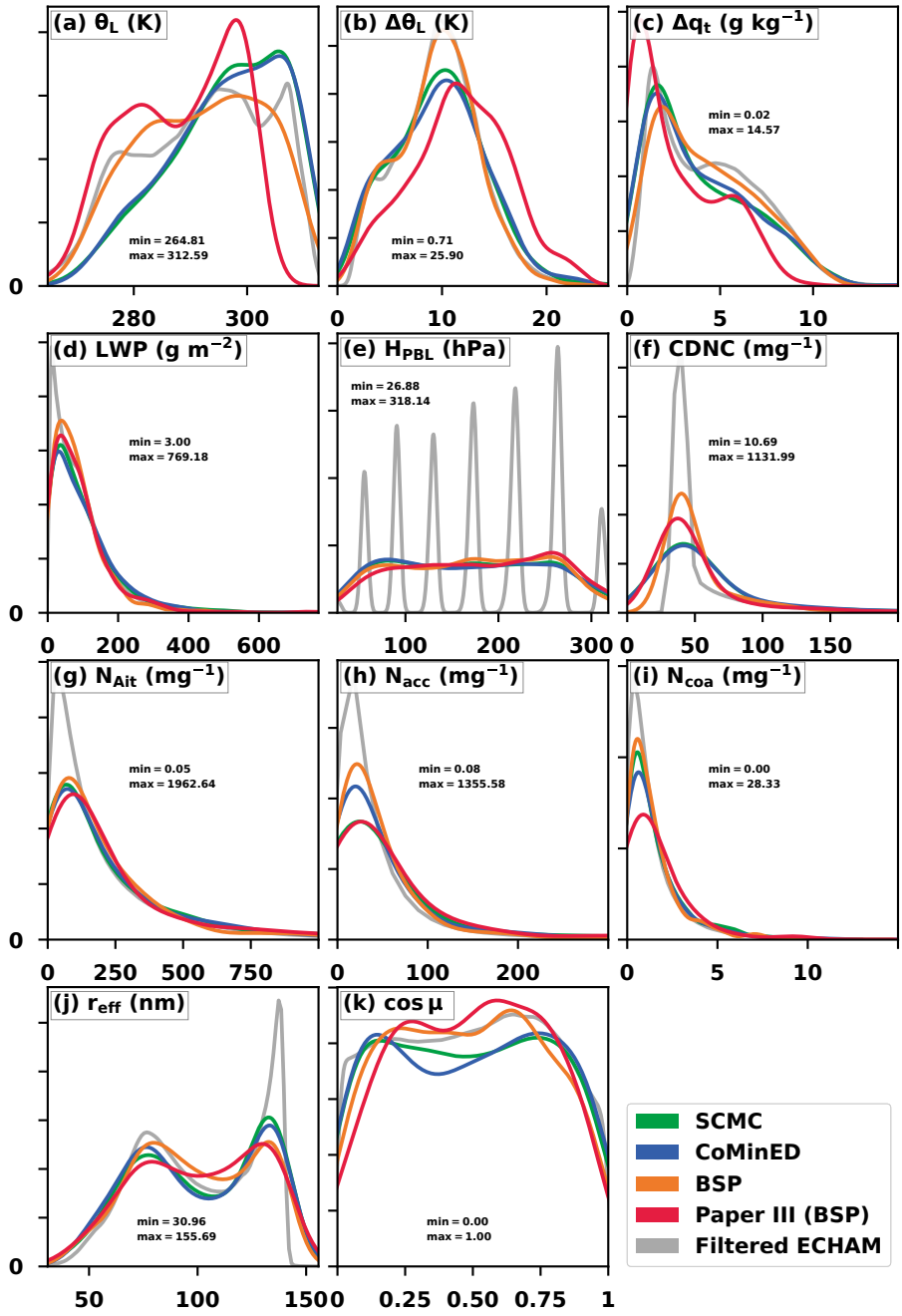

Figure 11. Design feature distribution for designs measured with MaxPro measure and created with adaptive SCMC, CoMinED, BSP (both new results and those from **Paper III**) and Filtered ECHAM (source data). The number of design points in all sub-figures is 499 except **Paper III** results with 500 design points with SB day and 150 with SALSA day. All distributions are from the SB day set except for those design features that are SALSA specific (sub-figures g, h, i, j).

6 Conclusions

This thesis aims to answer the following research questions:

Q1. What mathematical tools are useful to describe the complex phenomena of aerosol-cloud interactions that affect climate in multiple ways?

Q2. In the context of climate system models, can the uncertainty related to aerosol-cloud interactions be decreased?

Q3. Can the climate system models be improved by implementing machine learning methods for modelling cloud processes?

To answer the first broad question (**Q1**) we have explored a variety of tools such as a global climate model ECHAM, cloud-scale model UCLALES-SALSA, several machine learning methods to improve cloud process description and methods to upgrade choosing a design for computational experimentation. All those tools in their specific regimes are highly useful in studying climate. Yet, they are not perfect or comprehensive.

To resolve the second research question (**Q2**), in this study we have explored several methods that improve aerosol-cloud-interaction description in cloud and climate models. In **Paper I** and **Paper II** we have upgraded the cloud model UCLALES-SALSA by integrating ice crystals and cold microphysics to facilitate the simulations of both completely glaciated and mixed-phase clouds in addition to warm clouds. In **Paper IV** we improved the detail of processes related to aerosol-cloud interactions in a climate model by employing novel parameterisations.

To deal with the research question (**Q3**), in **Paper III** and **Paper IV** we have used UCLALES-SALSA to run numerous simulations in creating computationally faster and more accurate cloud process parameterisations for a global climate model by using machine learning methods. The chosen cloud processes to be improved were updraft velocity and warm rain formation which have been pointed as one of the major sources of uncertainty in the cloud radiative forcing estimates in present climate models (Donner et al., 2016; Jing et al., 2019; Yoshioka et al., 2019; Bougiatioti et al., 2020). We have also examined ways to improve the design of experimentation, which could be used to further upgrade the aforementioned parameterisations. Nevertheless, the created parameterisation shows favourable results. To sum up, we have developed the UCLALES-SALSA cloud model (Chapter 3), applied UCLALES-SALSA for a cloud process parameterisation creation case (Chapter 4)

and explored how an ensemble of cloud simulations could be initialised optimally (Chapter 5).

Climate system modelling requires a large variety of mathematical tools. The main mathematical tools used in this study are mathematical modelling, numerical simulation, machine learning and optimisation. Mathematical modelling is incorporated within the UCLALES-SALSA cloud model, where physical phenomena are translated into the language of mathematics which is mainly differential equations that are further implemented into software. Further on, the UCLALES-SALSA cloud model software is utilised to run an ensemble of different cloud simulations with varying initial aerosol concentrations and meteorological conditions. Next, machine learning methods are used to create parameterisations based on the initial conditions and simulation output data to provide a realistic representation of cloud properties in larger-scale models. Optimisation is carried out when finding the optimal initial conditions for the simulations. Overall this study provides several improved mathematical tools to study cloud processes in a wide range of spatiotemporal scales.

With these tools, in **Paper I** we showed for example how the lifetime of a mixed-phase cloud is extended when employing a more detailed aerosol-cloud-ice interaction scheme compared to a simpler ice nucleation. In **Paper II** we showed how updraughts transport marine ice nucleating particles up to the cloud and maintain the cloud. In **Paper III** we created a more accurate updraft velocity parameterisation. In **Paper IV** aforementioned updraft velocity parameterisation, along with parameterisation for rainwater formation, was applied in the global climate model ECHAM with small but statistically significant results by improving the cloud description in the subgrid scale.

6.1 Discussion

In this section, we discuss certain aspects of climate modelling from a bird's-eye view.

6.1.1 Scale dilemma

One of the biggest challenges of climate or cloud modelling is the combination of a wide range of spatiotemporal scales and limited computational power. This is a dilemma. Usually, more accurate details require more dimensions or higher resolution in the model, which then leads to a higher computational cost. Would there be a way to have more accurate details with less computational time? Are there some ingenious ways to overcome the scale issues? In this study, compared to previous methods based on turbulent kinetic energy, we have provided a more precise updraft velocity parameterisation, which requires initially large amount of computational time but once the parameterisation is created it can be further applied with negligible additional computational time compared. Thus, machine learning methods could

provide more accurate details while having decent computational cost.

Another method to enhance the level of detail without excessive computational time is to use fixed irregular grid sizes where applicable, which is plausible in UCLALES-SALSA. Also in UCLALES-SALSA the vertical resolution changes according to altitude in a simplistic fashion. The idea is to have a smaller grid size (i.e. better resolution) where relevant phenomena occur. Some kind of intelligent adaptable resolution could be one possibility to overcome the scale issues. However, a grid size and shape based on a single phenomenon would most likely lead to complications due to interactions and different length scales. The grid could be adapted for example according to the studied phenomena or geographical region. Different grid solutions exist, like horizontal grids separated into varying sizes of triangles, and for example, global numerical weather prediction model ICON uses an unstructured icosahedral grid instead of a standard structured orthogonal grid used in LES models (Dipankar et al., 2015). These grid solutions tend to be fixed before running the simulation as a dynamically adapting grid would increase computational complexity, yet in some specific cases dynamic grid might be worth exploring.

More detailed cloud/climate simulations can also be achieved as the computational power increases as technology improves. However, this might not be a lasting solution as eventually, Moore's law of increasing CPU power will meet the boundaries of physical reality. Admitting, there is quantum computing and massive parallelisation, like GPUs that will provide future possibilities. More detailed climate simulations are needed to predict future climate conditions and hence guide decision-making.

6.1.2 Ice microphysics

The necessity of detail is clear with ice crystals since they come in various shapes and sizes, and it is said that there does not exist identical snowflakes. Fundamentally, ice crystals are mostly hexagonal or fractal shaped depending on temperature and whether the crystal has gone through cloud processing. In this study, they are represented with spherical algorithms to simplify calculations by using some circumvent solutions like low effective densities. These $m - D - v$ parameterisations do not hold any major issues, as any particle can be described as a spherical object considering relevant shape factors when employing fall speed, effective cross-sectional area (coagulation processes) and surface area (condensation processes). Although, the simulation results represent observational clouds on average, simulating and studying ice crystals and their interactions as realistically as possible could reveal interesting yet unknown outlier events such as tipping points. Additionally, as another solution Predicted Particle Properties (P3) microphysics scheme has been developed to allow free size and shape evolution of ice particles without categorising them into hail, graupel, etc. (Milbrandt et al., 2021). Still, P3 offers smooth shapes for ice crystals, not fractals.

From a more philosophical perspective, weather phenomena, including ice nucleation, are inherently chaotic. Chaos means that a small change in initial conditions can lead to a much larger, diverging end result of several orders of magnitude. The chaotic behaviour leads to a question. Do we have the mathematics or simply enough prognostic variables or even dimensions to describe cloud processes or ice nucleation, since even ice nucleation is not yet fully understood on a molecular level (Kiselev et al., 2017). Potentially, there could be a need for a sort of unifying mathematical theory that crosses over several orders of spatiotemporal magnitude. It could be that we are missing one or several dimensions to fully describe ice nucleation. However, adding more dimensions leads to challenges with computational time.

7 Appendices

7.1 Mathematics

7.1.1 Binary search algorithm

Bisect algorithm divides the ordered list $\{r_0 < r_1 < \cdots < r_i < \cdots < r_m\}$ recursively in two parts according to the search value r_i. The recursion continues until it finds the indexes $i-1$ and $i+1$, where $r_{i-1} < r_i < r_{i+1}$ (Wikipedia, a; Louis F. Williams, 1976).

7.1.2 Indicator function

Indicator function of characteristic function of a subset of a set is a function that maps all elements of the subset to one and rest of the elements to zero. The indicator function of a subset A of a set X is a function

$$\mathbb{1}_A : X \mapsto \{0, 1\} \tag{46}$$

defined as

$$\mathbb{1}_A(x) := \begin{cases} 0, & x \notin A \\ 1, & x \in A. \end{cases} \tag{47}$$

7.1.3 Dirac measure

A Dirac measure δ_x assigns a size to a set based on whether it contains a fixed element x or not. Dirac measure is defined on a set X (with any σ-algebra of subset of X) with given $x \in X$ and any measurable set $A \subseteq X$ by

$$\delta_x = \mathbb{1}_A(x) = \begin{cases} 0, & x \notin A \\ 1, & x \in A, \end{cases} \tag{48}$$

where $\mathbb{1}_A$ is the indicator function of A.

7.1.4 Hyperplane

In geometry, a hyperplane is defined as subspace of dimension $n-1$ within an n-dimensional space. For example, 3-dimensional space has hyperplanes that are 2-dimensional planes.

A hyperplane of an n-dimensional affine space is a flat subset with dimension $n - 1$ and it separates the space into two half spaces. A hyperplane of an projective space does not hold this feature (Wikipedia, c).

7.1.5 Latin Hypercube

Within statistical sampling, a Latin *square* is a square grid where selected sample positions are so that there is only one sample for each row and for each column. It is like towers positioned on a chess board that threaten all squares but not each other. A Latin hypercube is the same idea but generalised to n-dimensions where each sample is unique in each axis-aligned hyperplane holding it.

7.1.6 Markov kernel

In probability theory, a Markov kernel is a map (Wikipedia, d):

Let $(X, \mathcal{A})$ and $(Y, \mathcal{B})$ be measurable spaces. Let $(X, \mathcal{A})$ be the source and $(Y, \mathcal{B})$ the target. A Markov kernel is a map $\kappa : \mathcal{B} \times X \to [0, 1]$ having properties:

1 For every (fixed) $B \in \mathcal{B}$, the map $x \to \kappa(B, x)$ is $\mathcal{A}$-measurable.

2 For every (fixed) $x \in \mathcal{X}$, the map $B \to \kappa(B, x)$ is a probability measure on $(Y, \mathcal{B})$.

7.1.7 Simulated annealing

Simulated annealing (SA) is a probabilistic technique for finding the approximate global optimum of a given function (Wikipedia, e). Formally it is defined as a meta-heuristic to approximate global optimisation in a large search space for an optimisation problem. For large numbers of local optima, SA can find the global optima (Simulated Annealing). Simulated Annealing is a common choice with optimisation problems having a discrete search space.

The name of the algorithm comes from annealing in metallurgy. Annealing is a technique, where heating and controlled cooling of a material are utilised to alter the physical properties of a metal.

7.1.8 Sobol sequence

Sobol points or sequence is a method for generating low-discrepancy quasi-random numbers, that is sort of points in a unit hypercube that spread randomly but somewhat evenly (Sobol, 1967; Joe and Kuo, 2008). In this study, Sobol points are implemented with Python's SciPy library.

7.2 Physics

7.2.1 ECHAM

An atmospheric general circulation model developed at the Max Planck Institute for Meteorology (MPIM). The climate model ECHAM has been developed from the ECMWF operational forecast model (therefore the two first letters of its name: EC) and an extensive parameterisation package developed at Hamburg (therefore the last three letters HAM).

7.2.2 Planetary boundary layer

Planetary boundary layer (PBL) known also as the atmospheric boundary layer or peplosphere, is the lowest part of the atmosphere, and it is in direct influence with the planetary surface. Above the PBL is the free troposphere.

7.2.3 Radiation budget

The components of the radiation budget are the energy entering, reflected and emitted by the climate system. A budget that is out of balance can cause the temperature of the climate system to increase or decrease.

7.2.4 Stability of atmosphere

The stability of the atmosphere is based on the density difference of a rising or sinking air parcel compared to the environmental air density. The density of air is affected by atmospheric pressure, temperature and humidity. Turbulence is caused the stability as for example unstable atmosphere favours vertical motions. The stability of the atmosphere is affected by fluxes from the surface (heat fluxes, moisture fluxes) and changes caused by radiation at different altitudes.

7.2.5 Coagulation kernel

Coagulation kernels are mathematical functions that describe the rate at which hydrometeors (=cloud droplets/ ice crystals) and aerosols collide and merge with each other to form larger particles through a process called coagulation.

Coagulation kernels typically depend on several factors, such as the size and composition of the cloud particles, their relative velocities, and the atmospheric conditions, such as temperature and humidity.

7.2.6 Prognostic and diagnostic variables

In a mathematical model, prognostic variables evolve independently over space and time and their future behaviour can be predicted through numerical simulations. Typically, prognostic variables represent physical properties like temperature, wind speed, atmospheric pressure, and humidity. In contrast, diagnostic variables are derived from prognostic variables at particular times and locations, and they do not evolve independently. Prognostic variables can be used as inputs for models and often diagnostic variables represent outputs of the model, like precipitation, cloud cover and atmospheric stability.

List of References

H. Abdul-Razzak and S. J. Ghan. A parameterization of aerosol activation 3. sectional representation. *Journal of Geophysical Research: Atmospheres*, 107(D3):AAC 1–1–AAC 1–6, 2002. doi: 10.1029/2001JD000483. URL https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2001JD000483.

B. A. Albrecht. Aerosols, cloud microphysics, and fractional cloudiness. *Science*, 245(4923):1227–1230, 1989. doi: 10.1126/science.245.4923.1227. URL https://www.science.org/doi/abs/10.1126/science.245.4923.1227.

M. O. Andreae, C. D. Jones, and P. M. Cox. Strong present-day aerosol cooling implies a hot future. *Nature*, 435(7046):1187–1190, Jun 2005. ISSN 1476-4687. doi: 10.1038/nature03671. URL https://doi.org/10.1038/nature03671.

C. Andronache. *Mixed-phase Clouds: Observations and Modeling*. Elsevier, Saint Louis, 2017. ISBN 9780128105498. URL https://ebookcentral.proquest.com/lib/fmi/detail.action?docID=5064425. ProQuest Ebook Central.

A. Arola, A. Lipponen, P. Kolmonen, T. H. Virtanen, N. Bellouin, D. P. Grosvenor, E. Gryspeerdt, J. Quaas, and H. Kokkola. Aerosol effects on clouds are concealed by natural cloud heterogeneity and satellite retrieval errors. *Nature Communications*, 13(1):7357, Nov 2022. ISSN 2041-1723. doi: 10.1038/s41467-022-34948-5. URL https://doi.org/10.1038/s41467-022-34948-5.

A. Atangana. Chapter 2 - principle of groundwater flow. In A. Atangana, editor, *Fractional Operators with Constant and Variable Order with Application to Geo-Hydrology*, pages 15–47. Academic Press, 2018. ISBN 978-0-12-809670-3. doi: 10.1016/B978-0-12-809670-3.00002-3. URL https://www.sciencedirect.com/science/article/pii/B9780128096703000023.

B. W. Atkinson and J. Wu Zhang. Mesoscale shallow convection in the atmosphere. *Reviews of Geophysics*, 34(4):403–431, 1996. doi: 10.1029/96RG02623. URL https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/96RG02623.

J. D. Atkinson, B. J. Murray, M. T. Woodhouse, T. F. Whale, K. J. Baustian, K. S. Carslaw, S. Dobbie, D. O'Sullivan, and T. L. Malkin. The importance of feldspar for ice nucleation by mineral dust in mixed-phase clouds. *Nature*, 498(7454):355–358, jun 2013. ISSN 0028-0836. doi: 10.1038/nature12278.

A. Avramov and J. Y. Harrington. Influence of parameterized ice habit on simulated mixed-phase arctic clouds. *Journal of Geophysical Research*, 115:D03205, 2010. doi: 10.1029/2009JD012108.

J. Banks, J. S. Carson, B. L. Nelson, and D. M. Nicol. Discrete-event system simulation. In *Discrete-Event System Simulation*, 1995. URL https://api.semanticscholar.org/CorpusID:122566976.

S. E. Bauer, K. Tsigaridis, G. Faluvegi, L. Nazarenko, R. L. Miller, M. Kelley, and G. Schmidt. The turning point of the aerosol era. *Journal of Advances in Modeling Earth Systems*, 14(12):e2022MS003070, 2022. doi: 10.1029/2022MS003070. URL https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2022MS003070. e2022MS003070 2022MS003070.

J. Bect, L. Li, and E. Vazquez. Bayesian subset simulation. *SIAM/ASA Journal on Uncertainty Quan-*

tification, 5(1):762–786, 2017. doi: 10.1137/16M1078276. URL https://doi.org/10.1137/16M1078276.

T. Bergman, V.-M. Kerminen, H. Korhonen, K. J. Lehtinen, R. Makkonen, A. Arola, T. Mielonen, S. Romakkaniemi, M. Kulmala, and H. Kokkola. Evaluation of the sectional aerosol microphysics module salsa implementation in echam5-ham aerosol-climate model. *Geoscientific Model Development*, 5(3):845–868, 2012. doi: 10.5194/gmd-5-845-2012.

A. Bougiatioti, A. Nenes, J. J. Lin, C. A. Brock, J. A. de Gouw, J. Liao, A. M. Middlebrook, and A. Welti. Drivers of cloud droplet number variability in the summertime in the southeastern united states. *Atmospheric Chemistry and Physics*, 20(20):12163–12176, 2020. doi: 10.5194/acp-20-12163-2020.

L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, Oct 2001. ISSN 1573-0565. doi: 10.1023/A:1010933404324.

S. M. Calderón, J. Tonttila, A. Buchholz, J. Joutsensaari, M. Komppula, A. Leskinen, L. Hao, D. Moisseev, I. Pullinen, P. Tiitta, J. Xu, A. Virtanen, H. Kokkola, and S. Romakkaniemi. Aerosol–stratocumulus interactions: towards a better process understanding using closures between observations and large eddy simulations. *Atmospheric Chemistry and Physics*, 22(18):12417–12441, 2022. doi: 10.5194/acp-22-12417-2022. URL https://acp.copernicus.org/articles/22/12417/2022/.

R. J. Charlson, S. E. Schwartz, J. M. Hales, R. D. Cess, J. A. Coakley, Jr., J. E. Hansen, and D. J. Hoffman. Climate forcing by anthropogenic aerosols. *Science*, 255:423–430, 1992. doi: 10.1126/science.255.5043.423.

M. Chatziparaschos, N. Daskalakis, S. Myriokefalitakis, N. Kalivitis, A. Nenes, M. Gonçalves Ageitos, M. Costa-Surós, C. Pérez García-Pando, M. Zanoli, M. Vrekoussis, and M. Kanakidou. Role of k-feldspar and quartz in global ice nucleation by mineral dust in mixed-phase clouds. *Atmospheric Chemistry and Physics*, 23(3):1785–1801, 2023. doi: 10.5194/acp-23-1785-2023. URL https://acp.copernicus.org/articles/23/1785/2023/.

M. W. Christensen and G. L. Stephens. Microphysical and macrophysical responses of marine stratocumulus polluted by underlying ships: Evidence of cloud deepening. *Journal of Geophysical Research: Atmospheres*, 116(D3), 2011. doi: 10.1029/2010JD014638. URL https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2010JD014638.

P. J. DeMott, O. Möhler, O. Stetzer, G. Vali, Z. Levin, M. D. Petters, M. Murakami, T. Leisner, U. Bundke, H. Klein, Z. A. Kanji, R. Cotton, H. Jones, S. Benz, M. Brinkmann, D. Rzesanke, H. Saathoff, M. Nicolet, A. Saito, B. Nillius, H. Bingemer, J. Abbatt, K. Ardon, E. Ganor, D. G. Georgakopoulos, and C. Saunders. Resurgence in ice nuclei measurement research. *Bulletin of the American Meteorological Society*, 92(12):1623–1635, 2011. doi: 10.1175/2011BAMS3119.1.

M. S. Diamond, P. E. Saide, P. Zuidema, A. S. Ackerman, S. J. Doherty, A. M. Fridlind, H. Gordon, C. Howes, J. Kazil, T. Yamaguchi, J. Zhang, G. Feingold, and R. Wood. Cloud adjustments from large-scale smoke–circulation interactions strongly modulate the southeastern atlantic stratocumulus-to-cumulus transition. *Atmospheric Chemistry and Physics*, 22(18):12113–12151, 2022. doi: 10.5194/acp-22-12113-2022. URL https://acp.copernicus.org/articles/22/12113/2022/.

A. Dipankar, B. Stevens, R. Heinze, C. Moseley, G. Zängl, M. Giorgetta, and S. Brdar. Large eddy simulation using the general circulation model icon. *Journal of Advances in Modeling Earth Systems*, 7(3):963–986, 2015. doi: 10.1002/2015MS000431. URL https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1002/2015MS000431.

L. J. Donner, T. A. O'Brien, D. Rieger, B. Vogel, and W. F. Cooke. Are atmospheric updrafts a key to unlocking climate forcing and sensitivity? *Atmospheric Chemistry and Physics*, 16(20):12983–12992, 2016. doi: 10.5194/acp-16-12983-2016. URL https://acp.copernicus.org/articles/16/12983/2016/.

D. Draguljić, T. J. Santner, and A. M. Dean. Noncollapsing space-filling designs for bounded non-

rectangular regions. *Technometrics*, 54, 2012. doi: 10.1080/00401706.2012.676951. URL https://doi.org/10.1080/00401706.2012.676951.

B. Ervens, A. G. Carlton, B. J. Turpin, K. E. Altieri, S. M. Kreidenweis, and G. Feingold. Secondary organic aerosol yields from cloud-processing of isoprene oxidation products. *Geophysical Research Letters*, 35(2), 2008. doi: 10.1029/2007GL031828. URL https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2007GL031828.

G. Feingold and S. M. Kreidenweis. Cloud processing of aerosol as modeled by a large eddy simulation with coupled microphysics and aqueous chemistry. *Journal of Geophysical Research: Atmospheres*, 107(D23):AAC 6–1–AAC 6–15, 2002. doi: 10.1029/2002JD002054. URL https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2002JD002054.

G. Feingold, S. M. Kreidenweis, B. Stevens, and W. R. Cotton. Numerical simulations of stratocumulus processing of cloud condensation nuclei through collision-coalescence. *Journal of Geophysical Research: Atmospheres*, 101(D16):21391–21402, 1996. doi: 10.1029/96JD01552. URL https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/96JD01552.

M. Filioglou, T. Mielonen, D. Balis, E. Giannakaki, A. Arola, H. Kokkola, M. Komppula, and S. Romakkaniemi. Aerosol effect on the cloud phase of low-level clouds over the arctic. *Journal of Geophysical Research: Atmospheres*, 124(14):7886–7899, 7 2019. ISSN 2169-897X. doi: 10.1029/2018JD030088.

L. S. Freire. Large-eddy simulation of the atmospheric boundary layer with near-wall resolved turbulence. *Boundary-Layer Meteorology*, 184(1):25–43, Jul 2022. ISSN 1573-1472. doi: 10.1007/s10546-022-00702-z. URL https://doi.org/10.1007/s10546-022-00702-z.

H. Fuchs, Z. M. Kedem, and B. F. Naylor. On visible surface generation by a priori tree structures. In *Proceedings of the 7th Annual Conference on Computer Graphics and Interactive Techniques*, volume 14 of *SIGGRAPH '80*, page 124–133, New York, NY, USA, jul 1980. Association for Computing Machinery. ISBN 0897910214. doi: 10.1145/965105.807481.

GitHub:ots22/gpf. Gpf: Fortran library for gaussian process regression. https://github.com/ots22/gpf, 2019. Accessed: 2022-10-25.

J.-C. Golaz, S. Wang, J. D. Doyle, and J. M. Schmidt. Coamps®-les: Model evaluation and analysis of second-and third-moment vertical velocity budgets. *Boundary-Layer Meteorology*, 116(3): 487–517, Sep 2005. ISSN 1573-1472. doi: 10.1007/s10546-004-7300-5. URL https://doi.org/10.1007/s10546-004-7300-5.

J.-C. Golaz, M. Salzmann, L. J. Donner, L. W. Horowitz, Y. Ming, and M. Zhao. Sensitivity of the Aerosol Indirect Effect to Subgrid Variability in the Cloud Parameterization of the GFDL Atmosphere General Circulation Model AM3. *J. Clim.*, 24(13):3145–3160, 2011. doi: 10.1175/2010JCLI3945.1.

S. Golchi and J. Loeppky. Monte Carlo based designs for constrained domains. *arXiv preprint*, 2015. URL http://arxiv.org/abs/1512.07328.

C. J. Hahn and S. G. Warren. A gridded climatology of clouds over land (1971-96) and ocean (1954-97) from surface observations worldwide. In *A Gridded Climatology of Clouds Over Land (1971-96) and Ocean (1954-97) from Surface Observations Worldwide*, 2007. URL https://api.semanticscholar.org/CorpusID:130382152.

J. Y. Harrington, T. Reisin, W. R. Cotton, and S. M. Kreidenweis. Cloud resolving simulations of arctic stratus—part ii: Transition-season clouds. *Atmospheric Research*, 51:45–75, 1999. doi: 10.1016/S0169-8095(98)00098-2.

T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning*. Springer, 2 edition, 2008. ISBN 0-387-95284-5.

R. Honnert, G. A. Efstathiou, R. J. Beare, J. Ito, A. Lock, R. Neggers, R. S. Plant, H. H. Shin, L. Tomassini, and B. Zhou. The atmospheric boundary layer and the "gray zone" of turbulence: A critical review. *Journal of Geophysical Research: Atmospheres*, 125(13), 2020. doi: 10.1029/2019JD030317.

C. Huang, V. R. Joseph, and D. M. Ray. Constrained minimum energy designs. *Statistics and Computing*, 31(6), Oct. 2021. doi: 10.1007/s11222-021-10054-2. URL https://doi.org/10.1007/s11222-021-10054-2.

IPCC. *Climate Change 2021: The Physical Science Basis*. Cambridge University Press, 2021. URL https://www.ipcc.ch/report/ar6/wg1/downloads/report/IPCC_AR6_WGI_Full_Report.pdf. Chapter 7: The Earth's Energy Budget, Climate Feedbacks, and Climate Sensitivity.

A. Iwata and A. Matsuki. Characterization of individual ice residual particles by the single droplet freezing method: a case study in the asian dust outflow region. *Atmospheric Chemistry and Physics*, 18(3):1785–1804, 2018. doi: 10.5194/acp-18-1785-2018. URL https://acp.copernicus.org/articles/18/1785/2018/.

M. Z. Jacobson. *Fundamentals of Atmospheric Modeling*. Cambridge University Press, 2 edition, 2005. doi: 10.1017/CBO9781139165389.

C. A. Jeffery and P. H. Austin. Homogeneous nucleation of supercooled water: Results from a new equation of state. *Journal of Geophysical Research: Atmospheres*, 102(D21):25269–25279, 1997. doi: 10.1029/97JD02243.

X. Jing, K. Suzuki, and T. Michibata. The key role of warm rain parameterization in determining the aerosol indirect effect in a global climate model. *Journal of Climate*, 32(14):4409 – 4430, 2019. doi: 10.1175/JCLI-D-18-0789.1. URL https://journals.ametsoc.org/view/journals/clim/32/14/jcli-d-18-0789.1.xml.

S. Joe and F. Y. Kuo. Constructing sobol sequences with better two-dimensional projections. *SIAM Journal on Scientific Computing*, 30(5):2635–2654, 2008. doi: 10.1137/070709359. URL https://doi.org/10.1137/070709359.

M. E. Johnson, L. M. Moore, and D. Ylvisaker. Minimax and maximin distance designs. *J. Stat. Plan. Inference*, 26, 1990. doi: 10.1016/0378-3758(90)90122-B. URL https://doi.org/10.1016/0378-3758(90)90122-B.

V. R. Joseph. Rejoinder. *Qual. Eng.*, 28, 2016. doi: 10.1080/08982112.2015.1100452. URL https://doi.org/10.1080/08982112.2015.1100452.

V. R. Joseph, T. Dasgupta, R. Tuo, and C. J. Wu. Sequential exploration of complex surfaces using minimum energy designs. *Technometrics*, 57, 2015a. doi: 10.1080/00401706.2014.881749. URL https://doi.org/10.1080/00401706.2014.881749.

V. R. Joseph, E. Gul, and S. Ba. Maximum projection designs for computer experiments. *Biometrika*, 102, 2015b. doi: 10.1093/biomet/asv002. URL https://doi.org/10.1093/biomet/asv002.

V. R. Joseph, D. Wang, L. Gu, S. Lyu, and R. Tuo. Deterministic sampling of expensive posteriors using minimum energy designs. *Technometrics*, 61, 2019. doi: 10.1080/00401706.2018.1552203. URL https://doi.org/10.1080/00401706.2018.1552203.

L. Kang. Stochastic coordinate-exchange optimal designs with complex constraints. *Qual. Eng.*, 31, 2019. doi: 10.1080/08982112.2018.1508695. URL https://doi.org/10.1080/08982112.2018.1508695.

H. Kaper and H. Engler. *Mathematics & Climate*. Society for Industrial and Applied Mathematics, Philadelphia, PA, 2013. ISBN 987-1-611972-60-3.

R. W. Kennard and L. A. Stone. Computer aided design of experiments. *Technometrics*, 11, 1969. doi: 10.1080/00401706.1969.10490666. URL https://doi.org/10.1080/00401706.1969.10490666.

A. P. Khain, K. D. Beheng, A. Heymsfield, A. Korolev, S. O. Krichak, Z. Levin, M. Pinsky, V. Phillips, T. Prabhakaran, A. Teller, S. C. van den Heever, and J.-I. Yano. Representation of microphysical processes in cloud-resolving models: Spectral (bin) microphysics versus bulk parameterization. *Reviews of Geophysics*, 53(2):247–322, 2015. doi: 10.1002/2014RG000468. URL https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1002/2014RG000468.

M. Khairoutdinov and Y. Kogan. A new cloud physics parameterization in a large-eddy simulation model of marine stratocumulus. *Monthly Weather Review*, 128(1):229 – 243, 2000. doi: 10.1175/1520-0493(2000)128⟨0229:ANCPPI⟩2.0.CO;2.

V. I. Khvorostyanov and J. A. Curry. Khvorostyanov, v. i. and j. a. curry, 2000. a new theory of heterogeneous ice nucleation for application in cloud and climate models. geophys. res. lett., 27, 4081-4084. *Geophysical Research Letters*, 27:4081–4084, 01 2000. doi: 10.1029/1999GL011211.

V. I. Khvorostyanov and J. A. Curry. The theory of ice nucleation by heterogeneous freezing of deliquescent mixed ccn. part i: Critical radius, energy, and nucleation rate. *Journal of the Atmospheric Sciences*, 61(22):2676–2691, 2004. doi: 10.1175/JAS3266.1.

S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by simulated annealing. *Science*, 220, 1983. doi: 10.1126/science.220.4598.671. URL https://doi.org/10.1126/science.220.4598.671.

A. Kiselev, F. Bachmann, P. Pedevilla, S. J. Cox, A. Michaelides, D. Gerthsen, and T. Leisner. Active sites in heterogeneous ice nucleation—the example of k-rich feldspars. *Science*, 355(6323):367–371, 2017. ISSN 0036-8075. doi: 10.1126/science.aai8034. URL http://science.sciencemag.org/content/355/6323/367.

H. Kokkola, H. Korhonen, K. E. J. Lehtinen, R. Makkonen, A. Asmi, S. Järvenoja, T. Anttila, A.-I. I. Partanen, M. Kulmala, H. Järvinen, A. Laaksonen, and V.-M. M. Kerminen. SALSA - a sectional aerosol module for large scale applications. *Atmos. Chem. Phys.*, 8(9):2469–2483, 2008. ISSN 1680-7316. doi: 10.5194/acp-8-2469-2008.

H. Kokkola, T. Kühn, A. Laakso, T. Bergman, K. E. J. Lehtinen, T. Mielonen, A. Arola, S. Stadtler, H. Korhonen, S. Ferrachat, U. Lohmann, D. Neubauer, I. Tegen, C. Siegenthaler-Le Drian, M. G. Schultz, I. Bey, P. Stier, N. Daskalakis, C. L. Heald, and S. Romakkaniemi. Salsa2.0: The sectional aerosol module of the aerosol–chemistry–climate model echam6.3.0-ham2.3-moz1.0. *Geoscientific Model Development*, 11(9):3833–3863, 2018. doi: 10.5194/gmd-11-3833-2018.

H. Korhonen, V.-M. Kerminen, K. E. J. Lehtinen, and M. Kulmala. Ccn activation and cloud processing in sectional aerosol models with low size resolution. *Atmospheric Chemistry and Physics*, 5(9):2561–2570, 2005. doi: 10.5194/acp-5-2561-2005. URL https://acp.copernicus.org/articles/5/2561/2005/.

H. Korhonen, K. S. Carslaw, D. V. Spracklen, G. W. Mann, and M. T. Woodhouse. Influence of oceanic dimethyl sulfide emissions on cloud condensation nuclei concentrations and seasonality over the remote southern hemisphere oceans: A global model study. *Journal of Geophysical Research: Atmospheres*, 113(D15), 2008. doi: 10.1029/2007JD009718. URL https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2007JD009718.

M. Kulmala, J. Kontkanen, H. Junninen, K. Lehtipalo, H. E. Manninen, T. Nieminen, T. Petäjä, M. Sipilä, S. Schobesberger, P. Rantala, A. Franchin, T. Jokinen, E. Järvinen, M. Äijälä, J. Kangasluoma, J. Hakala, P. P. Aalto, P. Paasonen, J. Mikkilä, J. Vanhanen, J. Aalto, H. Hakola, U. Makkonen, T. Ruuskanen, R. L. r. Mauldin, J. Duplissy, H. Vehkamäki, J. Bäck, A. Kortelainen, I. Riipinen, T. Kurtén, M. V. Johnston, J. N. Smith, M. Ehn, T. F. Mentel, K. E. J. Lehtinen, A. Laaksonen, V.-M. Kerminen, and D. R. Worsnop. Direct observations of atmospheric aerosol nucleation. *Science*, 339(6122):943–946, Feb 2013. doi: 10.1126/science.1227385. URL https://doi.org/10.1126/science.1227385.

Z. J. Lebo and J. H. Seinfeld. A continuous spectral aerosol-droplet microphysics model. *Atmospheric Chemistry and Physics*, 11(23):12297–12316, 2011. doi: 10.5194/acp-11-12297-2011. URL https://acp.copernicus.org/articles/11/12297/2011/.

A. Leonard. Energy cascade in large-eddy simulations of turbulent fluid flows. *Adv. in Geophysics*, pages 237–248, 1974.

Z. Li, H. Xue, and F. Yang. A modeling study of ice formation affected by aerosols. *Journal of Geophysical Research: Atmospheres*, 118(19):11,213–11,227, 2013. doi: 10.1002/jgrd.50861.

A. Lipponen, V. Kolehmainen, S. Romakkaniemi, and H. Kokkola. Correction of approximation errors with random forests applied to modelling of cloud droplet formation. *Geoscientific Model Development*, 6(6):2087–2098, 2013.

A. Lipponen, J. M. J. Huttunen, S. Romakkaniemi, H. Kokkola, and V. Kolehmainen. Correction of Model Reduction Errors in Simulations. *SIAM J. Sci. Comput.*, 40(1):B305–B327, 2018. doi: 10.1137/15M1052421.

J. L. Loeppky, J. Sacks, and W. J. Welch. Choosing the sample size of a computer experiment: A practical guide. *Technometrics*, 51(4):366–376, Nov. 2009. doi: 10.1198/tech.2009.08040.

J. Louis F. Williams. A modification to the half-interval search (binary search) method. In *Proceedings of the 14th ACM Southeast Conference*, pages 95–101. ACM, 1976. doi: 10.1145/503561.503582.

S. Mak, C. L. Sung, X. Wang, S. T. Yeh, Y. H. Chang, V. R. Joseph, V. Yang, and C. J. Wu. An efficient surrogate model for emulation and physics extraction of large eddy simulations. *J. Am. Stat. Assoc.*, 113, 2018. doi: 10.1080/01621459.2017.1409123. URL https://doi.org/10.1080/01621459.2017.1409123.

E. R. Mansell, D. T. D. II, and J. M. Straka. Bin-emulating hail melting in three-moment bulk microphysics. *Journal of the Atmospheric Sciences*, 77(10):3361 – 3385, 2020. doi: 10.1175/JAS-D-19-0268.1. URL https://journals.ametsoc.org/view/journals/atsc/77/10/jasD190268.xml.

B. Maronga and D. Li. An investigation of the grid sensitivity in large-eddy simulations of the stable boundary layer. *Boundary-Layer Meteorology*, 182(2):251–273, Feb 2022. ISSN 1573-1472. doi: 10.1007/s10546-021-00656-8. URL https://doi.org/10.1007/s10546-021-00656-8.

V. Masson-Delmotte, P. Zhai, A. Pirani, S. Connors, C. Péan, S. Berger, N. Caud, Y. Chen, L. Goldfarb, M. Gomis, M. Huang, K. Leitzell, E. Lonnoy, J. Matthews, T. Maycock, T. Waterfield, O. Yelekçi, R. Yu, and B. Zhou, editors. *Climate Change 2021: The Physical Science Basis*. Cambridge University Press, Cambridge, United Kingdom and New York, NY, USA, 2021. doi: 10.1017/9781009157896.

M. D. McKay, R. J. Beckman, and W. J. Conover. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 21(2):239–245, 2021/09/20/ 1979a. ISSN 00401706. doi: 10.2307/1268522. URL https://doi.org/10.2307/1268522. Full publication date: May, 1979.

M. D. McKay, R. J. Beckman, and W. J. Conover. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 21, 1979b.

C. Meneveau. Turbulence: Subgrid-Scale Modeling. *Scholarpedia*, 5(1):9489, 2010. doi: 10.4249/scholarpedia.9489. revision #153312.

J. A. Milbrandt, H. Morrison, D. T. D. II, and M. Paukert. A triple-moment representation of ice in the predicted particle properties (p3) microphysics scheme. *Journal of the Atmospheric Sciences*, 78(2):439 – 458, 2021. doi: 10.1175/JAS-D-20-0084.1. URL https://journals.ametsoc.org/view/journals/atsc/78/2/jas-d-20-0084.1.xml.

J. Mohrmann, R. Wood, T. Yuan, H. Song, R. Eastman, and L. Oreopoulos. Identifying meteorological influences on marine low-cloud mesoscale morphology using satellite classifications. *Atmospheric Chemistry and Physics*, 21(12):9629–9642, 2021. doi: 10.5194/acp-21-9629-2021. URL https://acp.copernicus.org/articles/21/9629/2021/.

M. D. Morris and T. J. Mitchell. Exploratory designs for computational experiments. *J. Stat. Plann. Inference*, 43, 1995. doi: 10.1016/0378-3758(94)00035-T. URL https://doi.org/10.1016/0378-3758(94)00035-T.

National Oceanic and Atmospheric Administration. The first climate model. https://celebrating200years.noaa.gov/breakthroughs/climate_model/welcome.html, 2022. Accessed: 2022-10-26.

NOAA predicting climate. Predicting climate: Climate forcing, national oceanic and atmospheric administration (noaa). https://www.climate.gov/maps-data/climate-data-primer/predicting-climate/climate-forcing, 2023. Last accessed 2023-09-22.

A. O'Hagan. Curve fitting and optimal design for prediction. *Journal of the Royal Statistical Society: Series B (Methodological)*, 40(1):1–24, 1978. doi: 10.1111/j.2517-6161.1978.tb01643.x.

M. Ovchinnikov, A. S. Ackerman, A. Avramov, A. Cheng, J. Fan, A. M. Fridlind, S. Ghan, J. Harrington, C. Hoose, A. Korolev, G. M. McFarquhar, H. Morrison, M. Paukert, J. Savre, B. J. Shipway, M. D. Shupe, A. Solomon, and K. Sulia. Intercomparison of large-eddy simulations of arctic mixed-phase clouds: Importance of ice size distribution assumptions. *J. Adv. Model. Earth Syst.*, 6(1):223–248, 2014. doi: 10.1002/2013MS000282.

A. O'Hagan. Bayesian analysis of computer code outputs: A tutorial. *Reliability Engineering & System Safety*, 91(10):1290–1300, 2006. ISSN 0951-8320. doi: 10.1016/j.ress.2005.11.025. The Fourth International Conference on Sensitivity Analysis of Model Output (SAMO 2004).

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

V. T. J. Phillips, P. J. DeMott, and C. Andronache. An empirical parameterization of heterogeneous ice nucleation for multiple chemical species of aerosol. *Journal of the Atmospheric Sciences*, 65(9): 2757–2783, 2008. doi: 10.1175/2007JAS2546.1.

S. Pope. *Turbulent flows*. Cambridge University Press, Cambridge, 2000. doi: 10.1017/CBO9780511840531. URL https://doi.org/10.1017/CBO9780511840531.

M. Prank, J. Tonttila, J. Ahola, H. Kokkola, T. Kühn, S. Romakkaniemi, and T. Raatikainen. Impacts of marine organic emissions on low-level stratiform clouds – a large eddy simulator study. *Atmospheric Chemistry and Physics*, 22(16):10971–10992, 2022. doi: 10.5194/acp-22-10971-2022. URL https://acp.copernicus.org/articles/22/10971/2022/.

M. T. Pratola, O. Harari, D. Bingham, and G. E. Flowers. Design and analysis of experiments on nonconvex regions. *Technometrics*, 59, 2017. doi: 10.1080/00401706.2015.1115674. URL https://doi.org/10.1080/00401706.2015.1115674.

J. Quaas, B. Stevens, P. Stier, and U. Lohmann. Interpreting the cloud cover – aerosol optical depth relationship found in satellite data using a general circulation model. *Atmospheric Chemistry and Physics*, 10(13):6129–6135, 2010. doi: 10.5194/acp-10-6129-2010. URL https://acp.copernicus.org/articles/10/6129/2010/.

C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. the MIT Press, 2006. ISBN 026218253X.

S. Rast, M. Schultz, I. Bey, T. van Noije, A. Aghedo, G. Brasseur, T. Diehl, M. Esch, L. Ganzeveld, I. Kirchner, L. Kornblueh, A. Rhodin, E. Roeckner, H. Schmidt, S. Schroeder, U. Schulzweida, P. Stier, K. Thomas, , and S. Walters. Evaluation of the tropospheric chemistry general circulation model echam5-moz and its application to the analysis of the chemical composition of the troposphere with an emphasis on the late retro period 1990–2000. *Berichte zur Erdsystemforschung*, page 74 pp., 2014. doi: 10.17617/2.2058065.

R. M. Rauber and A. Tokay. An explanation for the existence of supercooled water at the top of cold clouds. *Journal of the Atmospheric Sciences*, 48:1005–1023, 1991. URL https://api.semanticscholar.org/CorpusID:121118240.

V. Rodriguez-Galiano, B. Ghimire, J. Rogan, M. Chica-Olmo, and J. Rigol-Sanchez. An assessment of the effectiveness of a random forest classifier for land-cover classification. *ISPRS Journal of Photogrammetry and Remote Sensing*, 67:93–104, 2012. ISSN 0924-2716. doi: 10.1016/j.isprsjprs.2011.11.002. URL https://www.sciencedirect.com/science/article/pii/S0924271611001304.

S. M. Saleeby, S. R. Herbener, S. C. van den Heever, and T. L'Ecuyer. Impacts of cloud droplet–nucleating aerosols on shallow tropical convection. *Journal of the Atmospheric Sciences*, 72(4):1369 – 1385, 2015. doi: 10.1175/JAS-D-14-0153.1. URL https://journals.ametsoc.org/view/journals/atsc/72/4/jas-d-14-0153.1.xml.

T. J. Santner, B. J. Williams, and W. I. Notz. *The Design and Analysis of Computer Experiments*. Springer, Berlin, 2018. doi: 10.1007/978-1-4939-8847-1. URL https://doi.org/10.1007/978-1-4939-8847-1.

P. Saugaut. *Large Eddy Simulation for Incompressible Flows*. Springer, 3 edition, 2006. ISBN 978-3-540-26344-9.

J. Savre, A. M. L. Ekman, and G. Svensson. Technical note: Introduction to mimica, a large-eddy simulation solver for cloudy planetary boundary layers. *Journal of Advances in Modeling Earth Systems*, 6(3):630–649, 2014. doi: 10.1002/2013MS000292.

T. Schneider, S. Lan, A. Stuart, and J. Teixeira. Earth system modeling 2.0: A blueprint for models that learn from observations and targeted high-resolution simulations. *Geophysical Research Letters*, 44(24):12,396–12,417, 2017. doi: 10.1002/2017GL076101. URL https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1002/2017GL076101.

M. G. Schultz, S. Stadtler, S. Schröder, D. Taraborrelli, B. Franco, J. Krefting, A. Henrot, S. Ferrachat, U. Lohmann, D. Neubauer, C. Siegenthaler-Le Drian, S. Wahl, H. Kokkola, T. Kühn, S. Rast, H. Schmidt, P. Stier, D. Kinnison, G. S. Tyndall, J. J. Orlando, and C. Wespes. The chemistry–climate model echam6.3-ham2.3-moz1.0. *Geoscientific Model Development*, 11(5):1695–1723, 2018. doi: 10.5194/gmd-11-1695-2018. URL https://gmd.copernicus.org/articles/11/1695/2018/.

SciPy Sobol. Engine for generating (scrambled) sobol' sequences. https://docs.scipy.org/doc/scipy-1.9.1/reference/generated/scipy.stats.qmc.Sobol.html, 2022. Last accessed: 2022-10-29.

A. Seifert and K. D. Beheng. A double-moment parameterization for simulating autoconversion, accretion and selfcollection. *Atmospheric Research*, 59-60:265–281, 2001. ISSN 0169-8095. doi: 10.1016/S0169-8095(01)00126-0. URL https://www.sciencedirect.com/science/article/pii/S0169809501001260. 13th International Conference on Clouds and Precipitation.

A. Seifert and K. D. Beheng. A two-moment cloud microphysics parameterization for mixed-phase clouds. part 1: Model description. *Meteorology and Atmospheric Physics*, 92(1):45–66, Feb 2006. ISSN 1436-5065. doi: 10.1007/s00703-005-0112-4.

J. H. Seinfeld and S. N. Pandis. *Atmospheric Chemistry and Physics: From Air Pollution to Climate Change*. John Wiley & Sons inc., 1998. ISBN 0-471-17815-2.

S. Shima, K. Kusano, A. Kawano, T. Sugiyama, and S. Kawahara. The super-droplet method for the numerical simulation of clouds and precipitation: a particle-based and probabilistic microphysics model coupled with a non-hydrostatic model. *Quarterly Journal of the Royal Meteorological Society*, 135(642):1307–1320, 2009. doi: 10.1002/qj.441. URL https://rmets.onlinelibrary.wiley.com/doi/abs/10.1002/qj.441.

S. J. Silva, P.-L. Ma, J. C. Hardin, and D. Rothenberg. Physically regularized machine learning emulators of aerosol activation. *Geoscientific Model Development*, 14(5):3067–3077, 2021. doi: 10.5194/gmd-14-3067-2021.

Simulated Annealing. https://www.cs.cmu.edu/afs/cs.cmu.edu/project/learn-43/lib/photoz/.g/web/glossary/anneal.html, 2023. Last accessed 2023-10-07.

J. Smagorinsky. General circulation experiments with the primitive equation i the basic experiment. *Monthly Weather Review*, pages 99–164, 1963. doi: 10.1175/1520-0493(1963)091⟨0099:GCEWTP⟩2.3.CO;2.

J. D. Small, P. Y. Chuang, G. Feingold, and H. Jiang. Can aerosol decrease cloud lifetime? *Geophysical Research Letters*, 36(16), 2009. doi: 10.1029/2009GL038888. URL https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2009GL038888.

I. M. Sobol. The distribution of points in a cube and the accurate evaluation of integrals. *Zh. Vychisl. Mat. i Mat. Phys.*, pages 784–802, 1967.

B. Stevens and G. Feingold. Untangling aerosol effects on clouds and precipitation in a buffered system. *Nature*, 461(7264):607–613, 2009. ISSN 1476-4687. doi: 10.1038/nature08281. URL https://doi.org/10.1038/nature08281.

B. Stevens and A. Seifert. Understanding macrophysical outcomes of microphysical choices in simulations of shallow cumulus convection. *Journal of the Meteorological Society of Japan*, 86: 143–162, 2008.

B. Stevens, C.-H. Moeng, and P. P. Sullivan. Large-eddy simulations of radiatively driven convection: Sensitivities to the representation of small scales. *Journal of the Atmospheric Sciences*, 56(23): 3963–3984, 1999. doi: 10.1175/1520-0469(1999)056⟨3963:LESORD⟩2.0.CO;2.

B. Stevens, C.-H. Moeng, A. S. Ackerman, C. S. Bretherton, A. Chlond, S. de Roode, J. Edwards, J.-C. Golaz, H. Jiang, M. Khairoutdinov, M. P. Kirkpatrick, D. C. Lewellen, A. Lock, F. Müller, D. E. Stevens, E. Whelan, and P. Zhu. Evaluation of large-eddy simulations via observations of nocturnal marine stratocumulus. *Monthly Weather Review*, 133(6):1443–1462, 2005. doi: 10.1175/MWR2930.1.

B. Stevens, M. Giorgetta, M. Esch, T. Mauritsen, T. Crueger, S. Rast, M. Salzmann, H. Schmidt, J. Bader, K. Block, R. Brokopf, I. Fast, S. Kinne, L. Kornblueh, U. Lohmann, R. Pincus, T. Reichler, and E. Roeckner. Atmospheric component of the mpi-m earth system model: Echam6. *Journal of Advances in Modeling Earth Systems*, 5(2):146–172, 2013. doi: 10.1002/jame.20015. URL https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1002/jame.20015.

E. Stinstra, D. Hertog, P. Stehouwer, and A. Vestjens. Constrained maximin designs for computer experiments. *Technometrics*, 45, 2003. doi: 10.1198/004017003000000168. URL https://doi.org/10.1198/004017003000000168.

C. Stubenrauch, W. Rossow, S. Kinne, S. Ackerman, C. G., and et al. Assessment of global cloud datasets from satellites: Project and database initiated by the gewex radiation panel. *Bulletin of the American Meteorological Society, American Meteorological Society*, 94:1031–1049, 2013. doi: 10.1175/BAMS-D-12-00117.1.

K. J. Sulia and J. Y. Harrington. Ice aspect ratio influences on mixed-phase clouds: Impacts on phase partitioning in parcel models. *Journal of Geophysical Research: Atmospheres*, 116(D21), 2011. doi: 10.1029/2011JD016298. URL https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2011JD016298.

J. Tonttila, Z. Maalick, T. Raatikainen, H. Kokkola, T. Kühn, and S. Romakkaniemi. UCLALES–SALSA v1.0: a large-eddy model with interactive sectional microphysics for aerosol, clouds and precipitation. *Geosci. Model Dev.*, 10(1):169–188, jan 2017. ISSN 1991-9603. doi: 10.5194/gmd-10-169-2017.

M. Trosset. Approximate maximin distance designs. *In: Proceedings of the Section on Physical and Engineering Sciences*, page 223–227, 1999.

S. Twomey. Aerosols, clouds and radiation. *Atmospheric Environment. Part A. General Topics*, 25(11): 2435–2442, 1991. ISSN 0960-1686. doi: 10.1016/0960-1686(91)90159-5. URL https://www.sciencedirect.com/science/article/pii/0960168691901595. Symposium on Global Climatic Effects of Aerosols.

C. D. Tóth. Binary space partitions: recent developments. In J. E. Goodman, P. J., and E. Welzl, editors, *Combinatorial and Computational Geometry*, volume 52 of MSRI Publications, chapter 29, pages 529–556. Cambridge University Press, Cambridge, 2005.

S. G. Warren, C. J. Hahn, J. London, R. M. Chervin, and R. L. Jenne. Global distribution of total cloud cover and cloud type amounts over land. Technical Report NCAR/TN-273+STR, University Corporation for Atmospheric Research, 1986.

S. G. Warren, C. J. Hahn, J. London, R. M. Chervin, R. L. Jenne, C. Colorado Univ., Boulder, C. Colorado Univ., Boulder, and C. National Center for Atmospheric Research, Boulder. Global distribution of total cloud cover and cloud type amounts over the ocean. 12 1988. doi: 10.2172/5415329. URL https://www.osti.gov/biblio/5415329.

Wikipedia. Binary search algorithm. https://en.wikipedia.org/wiki/Binary_search_algorithm, 2023a. Last accessed 2023-09-24.

Wikipedia. Coulomb's law. https://en.wikipedia.org/wiki/Coulomb%27s_law, 2023b. Last accessed 2023-09-24.

Wikipedia. Hyperplane. https://en.wikipedia.org/wiki/Hyperplane, 2023c. Last accessed 2023-09-24.

Wikipedia. Markov kernel. https://en.wikipedia.org/wiki/Markov_kernel, 2023d. Last accessed 2023-09-24.

Wikipedia. Simulated annealing. https://en.wikipedia.org/wiki/Simulated_annealing, 2023e. Last accessed 2023-10-07.

Wikipedia. Simulation. https://en.wikipedia.org/wiki/Simulation, 2023f. Last accessed 2023-09-22.

R. Wood. Stratocumulus Clouds. *Mon. Weather Rev.*, 140(8):2373–2423, aug 2012. ISSN 0027-0644. doi: 10.1175/MWR-D-11-00121.1.

World Meteorological Organization Cloud Atlas. World meteorological organization: International cloud atlas — definitions of clouds. https://cloudatlas.wmo.int/en/clouds-definitions.html, 2022. Last accessed 2022-10-24.

Z. Wu, D. Wang, W. Wang, K. Zhao, P. N. Okolo, and W. Zhang. Space-filling experimental designs for constrained design spaces. *Eng. Optim.*, 51, 2019. doi: 10.1080/0305215X.2018.1542691. URL https://doi.org/10.1080/0305215X.2018.1542691.

M. Yoshioka, L. A. Regayre, K. J. Pringle, J. S. Johnson, G. W. Mann, D. G. Partridge, D. M. H. Sexton, G. M. S. Lister, N. Schutgens, P. Stier, Z. Kipling, N. Bellouin, J. Browse, B. B. B. Booth, C. E. Johnson, B. Johnson, J. D. P. Mollard, L. Lee, and K. S. Carslaw. Ensembles of global climate model variants designed for the quantification and constraint of uncertainty in aerosols and their radiative forcing. *Journal of Advances in Modeling Earth Systems*, 11(11):3728–3754, 2019. doi: 10.1029/2019MS001628.

K. Zhang, D. O'Donnell, J. Kazil, P. Stier, S. Kinne, U. Lohmann, S. Ferrachat, B. Croft, J. Quaas, H. Wan, S. Rast, and J. Feichter. The global aerosol-climate model echam-ham, version 2: sensitivity to improvements in process representations. *Atmospheric Chemistry and Physics*, 12(19): 8911–8949, 2012. doi: 10.5194/acp-12-8911-2012. URL https://acp.copernicus.org/articles/12/8911/2012/.

Y. Zheng and D. Rosenfeld. Linear relation between convective cloud base height and updrafts and application to satellite retrievals. *Geophys. Res. Lett.*, 42(15):6485–6491, 2015. doi: 10.1002/2015GL064809.

Y. Zheng, D. Rosenfeld, and Z. Li. Quantifying cloud base updraft speeds of marine stratocumulus from cloud top radiative cooling. *Geophys. Res. Lett.*, 43(21):11,407–11,413, nov 2016. ISSN 00948276. doi: 10.1002/2016GL071185.