

# **MACHINE LEARNING EMULATORS FOR NUMERICAL WEATHER PREDICTION**

*Applications to parametrization schemes*



David Meyer  
*Department of Meteorology*  
*University of Reading*

Thesis submitted for the degree of  
*Doctor of Philosophy*

April 2022

## **DECLARATION OF AUTHORSHIP**

I, David Meyer, confirm that this is my own work and the use of material from other sources has been fully acknowledged.

## **COPYRIGHT AND LICENCE**

Copyright © 2020-2022 David Meyer  
Creative Commons Attribution 4.0 International  
<https://creativecommons.org/licenses/by/4.0>

# ABSTRACT

Weather prediction hinges on mathematical models implemented into software to predict the future state of the atmosphere. Despite remarkable progress, computational constraints and user demands are choking this progress. In this PhD thesis, new machine learning (ML) methods are presented to improve the parametrization of two common schemes used within numerical weather prediction (NWP): radiation and urban land surface. First, a fast and accurate ML emulator for simulating three-dimensional cloud effects as a correction term to a fast parametrization scheme is developed rather than replacing the entire radiation scheme. Second, as ML emulators' training data can be scarce or expensive, a cheap method based on statistical copulas is implemented to generate data like the original across variables and dimensions. Third, the urban land surface model Town Energy Balance (TEB) is coupled to the Weather Research and Forecasting (WRF) model through a modular implementation, verified by an integration test, to evaluate a newly devised urban neural network (UNN) emulator. By training the UNN on the mean output from several urban land surface schemes, the UNN is more accurate, cheaper to run, and simpler to set up than TEB. Furthermore, when coupled to WRF, the UNN is numerically stable with lower errors than the reference WRF-TEB implementation.

# CONTENTS

---

<b>FIGURE CAPTIONS</b>	<b>vii</b>
<b>TABLE CAPTIONS</b>	<b>ix</b>
<b>PUBLICATIONS</b>	<b>x</b>
<b>CHAPTER</b>	
<b>1 Introduction</b>	<b>1</b>
1.1 Background and Motivation	2
1.2 Aim and Objectives	4
1.3 Thesis Structure	5
<b>2 Background</b>	<b>6</b>
2.1 A Brief History of Weather Forecasting	7
2.2 Numerical Weather Prediction Models and Parametrization Schemes	9
2.3 Machine Learning and Neural Networks	13
<b>3 A Faster, More Accurate Radiation Scheme</b>	<b>16</b>
3.1 Introduction	17
3.2 Methods	21
3.2.1 <i>Reference Model and Data</i>	21
3.2.2 <i>Neural Network Emulator</i>	22
3.3 Results and Discussion	26
3.3.1 <i>Qualitative Evaluation</i>	26
3.3.2 <i>Quantitative Evaluation</i>	29
3.3.3 <i>Runtime Performance</i>	33
3.4 Conclusion	34
<b>4 Boost Data, Boost Predictions</b>	<b>36</b>
4.1 Introduction	37
4.2 Materials and Methods	39
4.2.1 <i>Overview</i>	39
4.2.2 <i>Source Data</i>	41
4.2.3 <i>Data Generation</i>	43
4.2.4 <i>Target Generation</i>	46
4.2.5 <i>Estimator Training</i>	47
4.3 Results	49
4.3.1 <i>Copula</i>	49

4.3.2	<i>Machine Learning</i>	51
4.4	Discussion and Conclusion	53
<b>5</b>	<b>A Reference Urban Land Surface Model</b>	<b>56</b>
5.1	Introduction	57
5.2	Model and Software	59
5.2.1	<i>Weather Research and Forecasting</i>	59
5.2.2	<i>Town Energy Balance</i>	60
5.3	Coupling Approach	62
5.4	Software Implementation	66
5.4.1	<i>Modularity</i>	67
5.4.2	<i>Clarity</i>	69
5.4.3	<i>Reliability</i>	69
5.5	Scientific Reproducibility	70
5.6	Model Evaluation	72
5.6.1	<i>Site and Observational Data</i>	72
5.6.2	<i>Model Setup</i>	75
5.6.3	<i>Integration Test</i>	77
5.6.4	<i>Meteorological Evaluation</i>	81
5.7	Concluding Remarks	83
<b>6</b>	<b>A Faster, More Accurate Urban Land Surface Model</b>	<b>85</b>
6.1	Introduction	86
6.2	Methods	88
6.2.1	<i>General Problem</i>	88
6.2.2	<i>Urban Neural Network</i>	89
6.2.3	<i>Town Energy Balance</i>	90
6.2.4	<i>Weather and Research Forecasting (WRF) Coupling</i>	91
6.2.5	<i>Data and Model Setup</i>	92
6.2.6	<i>Evaluation Metrics</i>	98
6.3	Results and Discussion	98
6.3.1	<i>Multi-model Ensemble Mean</i>	98
6.3.2	<i>Offline Simulations</i>	99
6.3.3	<i>Online Simulations</i>	102
6.3.4	<i>Computational Performance</i>	105
6.4	Conclusion	105
<b>7</b>	<b>Conclusions</b>	<b>107</b>
7.1	Conclusions and Contributions	108
7.2	Lessons Learnt	110
7.3	Outlook	112

<b>APPENDICES</b>	<b>113</b>
<b>A</b> Postprocessing Methods for Chapter 3	114
<b>B</b> Statistical Metrics for Chapter 5	116
<b>C</b> Stage Selection for Chapter 6	117
<b>SUPPLEMENTARY INFORMATION</b>	<b>118</b>
<b>CODE AND DATA AVAILABILITY</b>	<b>122</b>
<b>ACKNOWLEDGEMENTS</b>	<b>123</b>
<b>REFERENCES</b>	<b>124</b>

# FIGURE CAPTIONS

---

1.1	Idealized present and future numerical weather prediction modelling workflow	3
1.2	Idealised processing workflow implementable to create surrogate ML models and as implemented in this work	5
2.1	The Electronic Numerical Integrator and Computer (ENIAC) main control panel at the Moore School of Electrical Engineering at the University of Pennsylvania	9
2.2	Idealised global NWP model and its parametrization schemes	11
3.1	Mean absolute errors resulting from different hyperparameters configurations	25
3.2	Typical zero- and one-dimensional SPARTACUS and Tripleclouds inputs	27
3.3	Qualitative evaluation of a slice of the atmosphere	28
3.4	Comparison of 3D signal and 3D prediction	29
3.5	Per-level statistics of fluxes	31
3.6	Per-level statistics of heating rates	32
4.1	General strategies for training ML emulators	40
4.2	Atmospheric profiles used in this study	42
4.3	Statistical evaluation of synthetically generated data	49
4.4	Qualitative evaluation of synthetically generated data	50
4.5	Errors for baseline and copula cases	52
4.6	Qualitative evaluation of emulated profiles for baseline and copula cases	53
5.1	Coupling approach	64
5.2	Software coupling	67
5.3	The scientific reproducibility approach taken in this paper	71
5.4	The four WRF nested domains used in online simulations	73

5.5	View of Toulouse downtown roofs from the terrace of the central site	74
5.6	Integration test workflow	79
5.7	Results from the integration test used to verify the coupling	80
5.8	Observations and simulations results from the different models	82
6.1	Conceptual sketch of surface energy balance exchanges for a clear-sky day	88
6.2	Conceptual sketch of the urban neural network	89
6.3	Call to Tensorflow Lite (TFLite) C API from WRF as implemented for this study	92
6.4	Meteorological data	96
6.5	Study location and domains	97
6.6	Comparison of UNN to MEM	99
6.7	Offline simulated and observed 30-min flux densities with interquartile range	101
6.8	Offline simulated and observed 30-min flux densities	102
6.9	As Figure 6.8, but for online simulations	103
6.10	Spatial comparison of online simulations	104
C.1	Model metrics used in the stage selection	117
S.1	Momentum flux density for the original and reduced number of models outputs data set	118
S.2	Fluxes modelled by 32 urban land surface models or configurations for all four Stages for four fluxes	118
S.3	As Figure S2, but for the selected 22 urban land surface models	118
S.4	Boxplot of 100 mean nMAE UNN iterations	119

# TABLE CAPTIONS

---

F.1	Author-specific contributions for co-authored papers	x
3.1	Inputs and outputs used in the two NN emulators	25
3.2	Bulk error statistics	32
4.1	Profiles of input and output quantities used in this study	42
4.2	Errors for baseline and copula cases	52
5.1	Models and software used in the coupling	60
5.2	Variables used in the coupling	65
5.3	General WPS/WRF configuration settings used in integration test and meteorological evaluation	76
5.4	TEB configuration differences between the integration test and the meteorological evaluation	77
5.5	Meteorological evaluation of half-hourly values for 2–5 July 2004	81
6.1	Inputs and outputs from data sets and models used to conduct the simulations	90
6.2	Results summary	100
S.1	Configuration used for the grid search	119
S.2	Model participants in the Grimmond et al. (2011) study	120
S.3	Morphological parameters provided for the four different Stages in Grimmond et al. (2011)	121
S.4	Main WPS/WRF configuration settings used with the model timestep for each domain	121
D.1	Links and references of permanent data archives used in this PhD thesis	122

# PUBLICATIONS

---

This PhD thesis consists of four research chapters (Chapters 3-7) published as papers (P1-P4). My and other author contributions are shown in Table F.1.

- P1.** Meyer, D., Hogan, R. J., Dueben, P. D., & Mason, S. L. (2022). Machine Learning Emulation of 3D Cloud Radiative Effects. *J. Adv. Model. Earth Syst.*, 14(3). <https://doi.org/10.1029/2021MS002550>
- P2.** Meyer, D., Nagler, T., & Hogan, R. J. (2021). Copula-based synthetic data augmentation for machine-learning emulators. *Geosci. Model Dev.*, 14(8), 5205–5215. <https://doi.org/10.5194/gmd-14-5205-2021>
- P3.** Meyer, D., Schoetter, R., Riechert, M., Verrelle, A., Tewari, M., Dudhia, J., Masson, V., van Reeuwijk, M., & Grimmond, S. (2020). WRF-TEB: Implementation and Evaluation of the Coupled Weather Research and Forecasting (WRF) and Town Energy Balance (TEB) Model. *J. Adv. Model. Earth Syst.*, 12(8). <https://doi.org/10.1029/2019MS001961>
- P4.** Meyer, D., Grimmond, S., Dueben, P., Hogan, R., & van Reeuwijk, M. (2022). Machine Learning Emulation of Urban Land Surface Processes. *J. Adv. Model. Earth Syst.*, 14(3). <https://doi.org/10.1029/2021MS002744>

**Table F.1 | Author-specific contributions for co-authored papers.** Contributions are shown using the Contributor Role Taxonomy (CRediT; Allen et al., 2019) for D. Meyer (DM), S. Grimmond (SG), R. Hogan (RH), P. Dueben (PD), R. Schoetter (RS), M. Riechert (MR), M. Tewari (MT), V. Masson (VM), T. Nagler (TN), A. Verrelle (AV), J. Dudhia (JD), M. van Reeuwijk (MvR). Original draft (OD). Review and editing (RE).

	<b>P1</b>	<b>P2</b>	<b>P3</b>	<b>P4</b>
Conceptualization	RH, PD	DM	DM	DM
Data curation	DM	DM	DM	DM
Formal analysis	DM	DM, TN	DM	DM
Investigation	DM	DM	DM	DM
Methodology	DM, RH, PD	DM, TN, RH	DM, AV	DM
Software	DM	DM	DM, MR, AV	DM
Resources	DM	DM	DM	DM
Validation	DM, PD	DM	DM	DM
Visualization	DM, SM	DM	DM	DM
Writing (OD)	DM, RH, PD	DM, TN, RH	DM, RS	DM
Writing (RE)	<i>all</i>	<i>all</i>	<i>all</i>	<i>all</i>

# CHAPTER 1

---

## Introduction

*I begin the first chapter outlining background information on numerical weather prediction, research motivations, aim and objectives, and thesis structure.*

# 1.1 Background and Motivation

Weather<sup>1</sup> and climate<sup>2</sup> matter. In 2021, global economic losses from weather-related disasters were estimated at about USD 258 billion—up 29 % from the twenty-first-century average (Aon, 2021). This year, the World Economic Forum has again ranked *climate action failure* and *extreme weather* as the top two long-term risks to our economy and society (World Economic Forum, 2022). Hinging on the use of numerical weather prediction (NWP) models<sup>3</sup>, weather predictions and climate projections are essential in mitigating the effects of weather and climate on our economy and society (Katz & Murphy, 1997; Maunder, 2019).

Although advances in modelling weather and climate phenomena have allowed us to represent more processes, in more detail (Alley et al., 2019; Bauer et al., 2015; Benjamin et al., 2016; Knutti & Sedláček, 2013), the slowing pace of progress in transistor technology combined with the heightened demand for higher resolution over a greater number of scenarios (e.g., Bauer, Stevens, et al., 2021) underlines the need to overhaul our current modelling approaches (Bauer, Dueben, et al., 2021; Lawrence et al., 2018). Today, most traditional NWP models consist of several hundred thousand to millions of lines of hand-tuned architecture-specific code (Fuhrer et al., 2014; Gettelman & Rood, 2016) that makes adaptation to new hardware architectures and accelerators challenging (Mozdzynski et al., 2015; Thaler et al., 2019). As such, several research groups are exploring alternative approaches such as domain-specific software frameworks and machine learning (ML)—whereas domain-

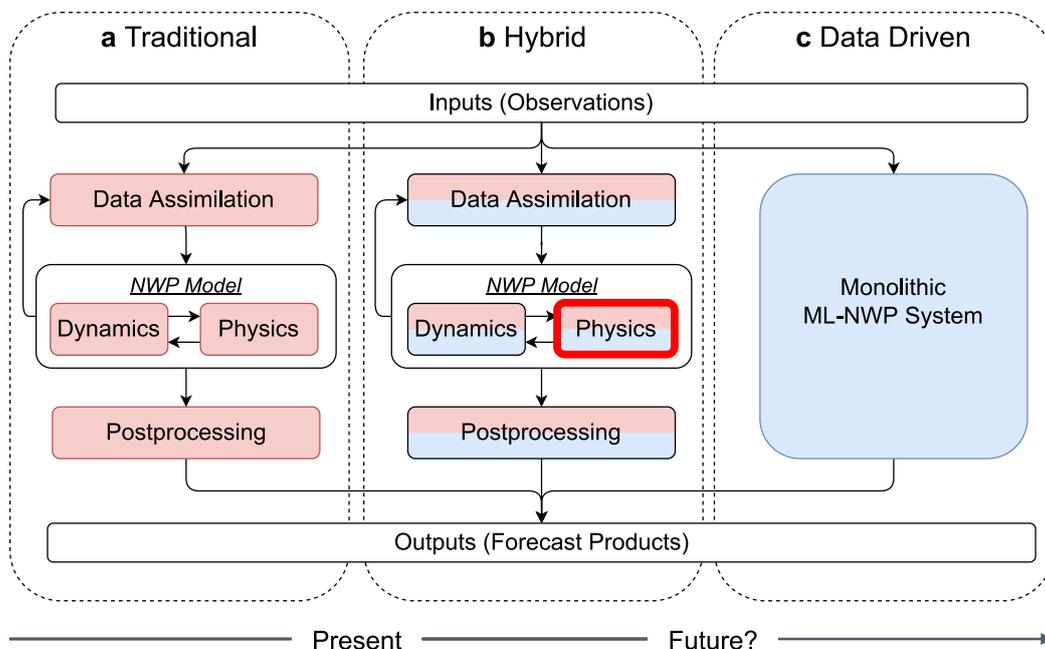
---

<sup>1</sup> Short-term changes (minutes to days) in the atmosphere (American Meteorological Society, 2021b).

<sup>2</sup> Slow-varying (month or more) aspects of the atmosphere–hydrosphere–land-surface system (American Meteorological Society, 2021a).

<sup>3</sup> The term NWP is used to refer to weather prediction. For climate projections, models are referred to as general circulation models (GCMs) or as Earth system model when GCMs are fully coupled to atmosphere, cryosphere, land, and ocean components (Benjamin et al., 2019; Randall et al., 2019; Watson-Parris, 2021). See Chapter 2.2 for more details.

specific software frameworks require hardware-specific code to be rewritten, ML approaches have come to the fore in recent years as they can offer a speedup for a trade-off in accuracy with little to no code rewrite (Bauer, Dueben, et al., 2021). ML approaches (Figure 1.1b) have already shown benefits over traditional (Figure 1.1a) approaches in several areas of weather and climate modelling (e.g., in nowcasting and observation processing, data assimilation, emulation of parametrization schemes and parameter tuning, and post-processing of model outputs) leading some (e.g., Schultz et al., 2021) to question if ML models may, one day, replace the entire weather prediction modelling workflow (Figure 1.1c).



**Figure 1.1 | Idealized present and future numerical weather prediction modelling workflow.** (a) ‘traditional’ workflow (i.e., as used today), (b) next-generation weather prediction with individual components substituted or augmented by ML and (c) purely data driven (i.e., ML) weather forecasting system. The red box shows where the contributions of this PhD research fit in. Modified after Figure 1 in Schultz et al. (2021).

Central to NWP models are two main components: a numerical core (dynamics; Figure 1.1) for solving the fluid motion in the atmosphere and a set of parametrization schemes (physics; Figure 1.1) used for processes that are too complex or too computationally expensive to be explicitly resolved (e.g.,

sub-grid scale turbulence, atmospheric radiative transfer, cloud microphysics, and land-surface processes; Ehrendorfer, 2011; Stensrud, 2007; Warner, 2010). Although the dynamical core is one of the single slowest components in NWP models, together, parametrization schemes make up for a large fraction of the total runtime (e.g., approximately 25 % of the ECMWF operational forecast; Dueben et al. 2020). As such, this PhD thesis focuses on emulating two parametrization schemes that are fundamental to weather and climate: radiation and urban land surface. Radiation is explored as it is the fundamental driver of atmospheric circulation and, typically, the largest energy source in the surface energy balance (Randall et al., 2019; Stensrud, 2007); urban land surface is explored as—with 54 % of the world’s population currently living in cities, projected to rise to 85 % by 2100 (United Nations, 2019)—it is the most relevant to most people (Oke et al., 2017).

## **1.2 Aim and Objectives**

The research presented in this PhD thesis aims to improve NWP models by cheaply representing 3D cloud radiative effects and urban land surface processes. Specific objectives are:

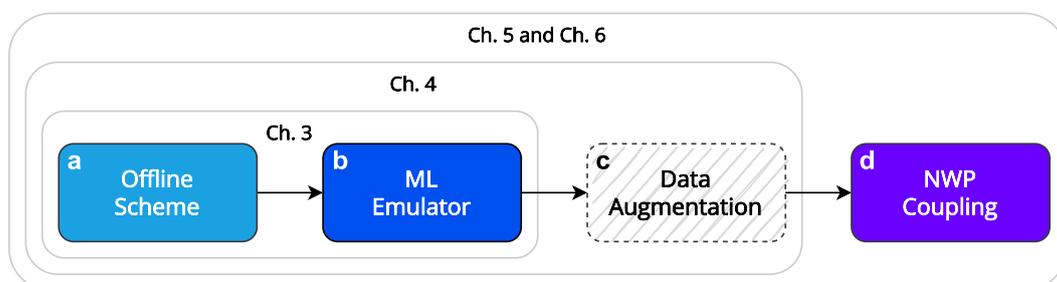
1. To develop and evaluate an offline machine learning (ML) emulator of atmospheric radiation.
2. To investigate synthetically generated data to improve the emulation of atmospheric longwave radiation.
3. To develop and evaluate an offline ML emulator for urban land surface processes coupled to an NWP model.

These are addressed as follows:

1. A neural network emulator is developed and assessed for 3D cloud radiative effects (Chapter 3).
2. A fast method to augment the training inputs of a neural network emulator is developed using a toy model of downwelling longwave radiation (Chapter 4).
3. A method to couple an offline urban land surface model to an NWP model is developed and assessed (Chapter 5) before being used as a baseline to assess a newly developed neural network emulator of urban land surface processes (Chapter 6).

## 1.3 Thesis Structure

A brief background (Chapter 2) outlines weather forecasting, parametrization scheme, and neural network terms central to this PhD research. The research (Chapters 3-6) follows an idealized processing workflow to create surrogate ML models (Figure 1.2). The overall conclusions, contributions, and lessons learnt are given in chapter 7.



**Figure 1.2 | Idealised processing workflow to create surrogate ML models as implemented in this work.** The general workflow for creating a surrogate ML model begins with (a) an offline parametrization scheme. From this (b) the corresponding ML emulator (i.e., the surrogate scheme) is generated. If the input samples are scarce (c) the ML emulator may be improved with more samples before being (d) coupled to the NWP model.

# CHAPTER 2

---

## Background<sup>4</sup>

*In the second introductory chapter, I give a high-level overview of the main topics covered in this thesis, namely parametrization schemes and machine learning. First, I give a brief history of weather forecasting, followed by an overview of numerical models and associated terms in weather and climate modelling, while outlining the main differences between the two. Second, I give a background of parametrization schemes and machine learning.*

---

<sup>4</sup> This chapter is based on the background section in the PhD thesis by Rasp (2019) and Larraondo (2019).

## 2.1 A Brief History of Weather Forecasting<sup>5</sup>

Attempts to forecast the weather have been made since antiquity: the Babylonians and Chaldeans attempted to link the weather to the motion of heavenly bodies, and the Egyptians had a sky religion (Frisinger, 1971). However, it was not until mathematicians and natural philosophers of Ancient Greece that a more reason-based approach was taken. Thales of Miletus (c. 624 – c. 548 BC) tried to understand the weather using reasoning, Anaximander (c. 611 – c. 547 BC) had astute observations leading him to define the wind as the flow of air, and Anaxagoras's (c. 499 – c. 427 BC) surprisingly modern beliefs that hail was caused by clouds being forced into the upper atmosphere and frozen (Frisinger, 1971). These ideas eventually culminated in what is today regarded as the oldest comprehensive treatise in Meteorology—*Meteorologica* (Aristotle, 1952). This book, probably by Aristotle (384–322 BC; see Furley, 1954), became the unquestioned authority in western meteorological theory (Frisinger, 1972) until the end of the seventeenth century (Hellmann, 1908). Indeed, the work of Classical Greek scholars such as Aristotle showed that their work was that of natural philosophers but not that of natural scientists in that their developed theories and philosophical thinking often outweighed experimental evidence, which had to wait until the seventeenth century Renaissance to change thanks to the development of experimental science and precise instruments (Frisinger, 1972).

---

<sup>5</sup> As the number of contributions in meteorology is immense—spanning fields of science and engineering—a great deal is omitted here. For a more comprehensive history of weather forecasting, readers are encouraged to refer to Burton (1986) for an early history of the Meteorological office in the UK, Randall et al. (2019) for a general overview of Earth system model developments, Thompson (1983) for a historical view of numerical weather prediction models in the US, Lorenz (2006) for reflection on the birth and development of numerical weather prediction, and Lynch (2014) for a discussion on the fundamental of numerical weather prediction and its history.

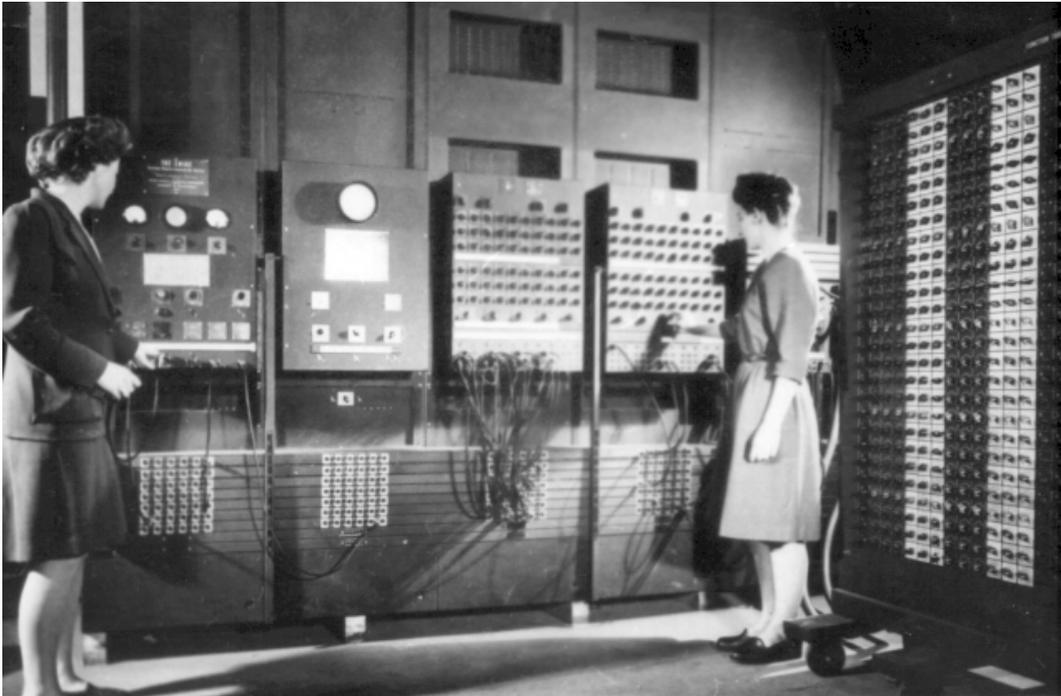
The years between the seventeenth and eighteenth centuries saw the development of physical and mathematical theories, such as calculus and fluid mechanics, and improvements in observational instruments. However, it was not until the second half of the nineteenth century that the humble electric telegraph made weather forecasting<sup>6</sup> a practical possibility (Craft, 1999, 2001). By the end of the 1840s, telegraphic daily weather reports were published daily (Craft, 2001; Monmonier, 1999). By the beginning of the 1850s, one E. E. Merriam published daily weather forecasts for New York partially based on his observations in Brooklyn but mostly on telegraphic reports published in local papers (Abbe, 1909). By the beginning of the 1860s, storm forecasts began being issued in Europe (Craft, 2001). All these forecasts were mainly based on empirical evidence and little skill—that is, they were far from the mathematical rigour used in today’s weather forecasting but were nevertheless predictions of a future state of the atmosphere, something that had not been possible before the establishment of the electric telegraph.

By the beginning of the twentieth century, modern meteorology was born: physicists began applying physical laws developed in previous centuries to forecast the weather. These steps were first documented by Cleveland Abbe (1901) and Vilhelm Bjorkens (1904, 2009), formalised by Lewis Fry Richardson (1922), and implemented by a group led by Charney, Fjörtoft, and von Neumann (1950) at the Institute for Advanced Study of Princeton University using the Electronic Numerical Integrator and Computer (ENIAC) at the Moore School of Electrical Engineering at the University of Pennsylvania (Figure 2.1). After Charney, Fjörtoft von Neumann (1950)'s first-ever successful numerical weather simulation of the continental US, *operational* but *limited-area* numerical weather prediction followed shortly after in Sweden, the US, and Japan. However, it was not until the 1970s that *global* numerical weather

---

<sup>6</sup> Accurate records of meteorological data had been kept since much earlier (see Lamb, 1995; Pfister et al., 1999), but it was only after the invention of the electric telegraph that the weather could be timely inferred from observations made windward.

prediction models began to be used operationally in the US and Europe (Lynch, 2014; Randall et al., 2019). Today, public and private entities run more sophisticated operational numerical weather prediction models several times a day to produce daily weather forecasts.



**Figure 2.1 | The Electronic Numerical Integrator and Computer (ENIAC) main control panel at the Moore School of Electrical Engineering at the University of Pennsylvania.** Mathematicians Betty Jennings (left) and Frances Bilas (right) run the ENIAC's main control panel at the Moore School of Electrical Engineering (Northwest Missouri State University, 2022; United States Army Research Laboratory, 2022). Credit: United States Army Research Laboratory.

## 2.2 Numerical Weather Prediction Models and Parametrization Schemes<sup>7</sup>

As noted in the introduction, modern weather forecasting—"the prediction of the weather through application of the principles of physics, supplemented by a variety of statistical and empirical techniques" (Cahir, 2019)—hinges on the use of computer software programs based on the mathematical equations of

---

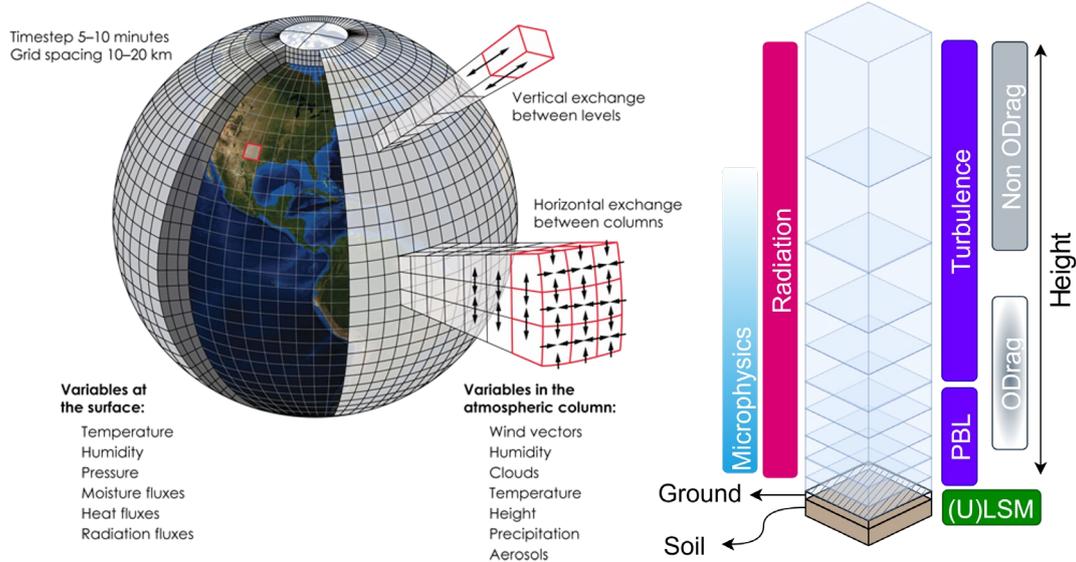
<sup>7</sup> The main source in this section is from Stensrud (2007).

motion describing the flow of fluids. Given the current atmospheric state, these software models predict the atmospheric state at a future moment in time. In the literature, the term NWP refers to weather prediction. For climate projections, models are referred to as general circulation models (GCMs) or as Earth system models (ESMs) when fully coupled to atmosphere, cryosphere, land, and ocean components. Indeed, all share two main components<sup>8</sup>: a dynamical core to account for the transport of heat and momentum (Figure 2.2a) and one or more parametrization schemes to account for processes that are too complex or (currently) too computationally expensive to be explicitly resolved (e.g., sub-grid turbulence, radiative transfer, cloud microphysics, and land-surface; Figure 2.2b).

Although weather and climate models may share common code and framework, they differ in their purpose, applications, and degree of parametrization. Weather models are applied to forecasting shorter temporal and spatial scales (e.g., maximum of two weeks into the future at an operational horizontal resolution of order 1-10 kilometres). On the other hand, climate models run from sub-seasonal to multi-year periods with horizontal resolution at their finest of order 10 km and for more extended periods at a resolution of order 100 km. Indeed, while weather models focus on predicting slight changes in the current state of the atmosphere, climate models focus on the effects of external changes to the earth systems, such as how increasing levels of carbon dioxide in the atmosphere change the long-term variation of temperatures across the globe. Here, the focus of the research is on weather.

---

<sup>8</sup> Here, I assume physically based NWPs. Statistical/ML emulators may not have a dynamical core.



**Figure 2.2 | Idealised global NWP model and its parametrization schemes.** (a) example of a weather or climate model by the American Geosciences Institute © Copyright 2019 is licensed under Creative Commons Attribution 4.0 International, and (b) Idealized vertical column of a model grid cell, with vertical layers showing atmospheric and soil layers is modified after Stensrud (2007). Areas where specific parametrizations are important for radiation, convection, microphysics, planetary boundary layer (PBL), turbulence, orographic drag (ODrag), and vegetation (and bare soil and water) are highlighted.

As previously mentioned, the representation of sub-grid processes is critical to atmospheric models—as such, NWP models need to be supplemented with parametrization schemes for solar and infrared radiation, cloud, soil, vegetation, and terrain effects. A parametrization represents the effect of a sub-grid process on the grid scale—it takes the resolved model state as input and returns the grid-scale tendency as output. A technical but essential aspect of most parametrization schemes is that, given the anisotropy of the grid cells in weather models, horizontal interactions can be neglected, allowing physical parametrizations to be treated as independent columns (Clement et al., 2018). Thus, most parametrization schemes today are developed and tested standalone, as separate components (offline mode), before being coupled to NWP models (online mode). Although NWP models have parametrizations for several processes not discussed here—such as orographic gravity wave drag or chemistry—in the following paragraph, I shall only focus on the parametrization of radiation and urban land surface as they are central to this PhD thesis.

Solar radiation is the primary driver for atmospheric weather systems via its role in establishing the equator-to-pole temperature gradient. In contrast, the interaction of thermal infrared radiation with greenhouse gases is the critical driver for anthropogenic climate change. In addition, the transfer of radiation through the atmosphere strongly influences surface temperatures. For clear skies, the primary problem is for gases varying by several orders of magnitude in tiny spectral regions. At the same time, when clouds are present, their properties change much more rapidly in time and space but overall over only three or four orders of magnitude (Randall et al., 2019). As such, the detailed representation of these processes in radiation schemes is vital for weather and climate models (Stensrud, 2007; Stephens, 1984). Radiation transfer methods currently used in NWP models can be traced back to the early twentieth century (Randall et al., 2019; Schuster, 1905). They predict the upwelling and downwelling flux profile through the atmosphere in NWP models and, therefore, the surface and atmosphere's heating (or cooling). Radiation schemes need to be able to accurately and efficiently represent the flow of radiation through the atmosphere (i.e., heating or cooling rates) as well as the fluxes emitted or absorbed at the surface in the case of weather-related applications but focus on the net energy input at the top of the atmosphere for climate-related applications (Randall et al., 2019). Thus, a radiation parametrization aims to compute the change in surface fluxes and that of heating (or cooling) of the atmosphere by solar and infrared radiation. Mathematically, the rate of heating (or cooling) of the atmosphere by radiation is represented by the heating (or cooling) rate  $dT/dt$  and expressed as:

$$\frac{dT}{dt} = \frac{1}{\rho c_p} \frac{d}{dz} (F^\downarrow - F^\uparrow), \quad (2.1)$$

with  $F^\downarrow$  and  $F^\uparrow$  the downwelling and upwelling flux, respectively,  $\rho$  the air density, and  $c_p$  the specific heat capacity of dry air.

Differently from radiation, the parametrization of urban land surface came much later, when it was realised in the 1980s that land surface processes play a crucial role in atmospheric processes at all scales (Stensrud, 2007). At the

core of any land surface model is the surface energy balance (SEB), a general statement of energy conservation with applications to surfaces and volumes of all temporal scales (Oke et al., 2017). SEB defines the energy balance at the infinitesimally thin Earth–atmosphere boundary, and it is composed of four main terms: net all-wave radiation  $Q^*$ , consisting of the net (downwelling minus upwelling) solar and infrared radiation flux, sensible heat flux  $Q_H$ , latent heat flux  $Q_E$ , and ground heat flux  $Q_G$ :

$$Q^* = Q_H + Q_E + Q_G. \quad (2.2)$$

In urban areas, similar models are used. However, they focus on processes more common to urban areas, such as buildings and their interaction with people and vegetation (Oke et al., 2017).

## 2.3 Machine Learning and Neural Networks<sup>9</sup>

Machine learning (ML), a subset of artificial intelligence (AI)<sup>10</sup>, is defined as “*the study of computer algorithms that can improve automatically through experience and by the use of data*” (Mitchell, 1997). ML can be split into supervised (e.g. finding matching dogs; the focus of this thesis) and unsupervised (e.g. clustering data into groups). In other words, the goal of supervised ML algorithms is to find a function  $f$  that maps an input  $x$  to the corresponding output  $y$ , that is,  $y = f(x)$ . This is done by maximising or minimising a loss function  $L$ , such as the mean square error between the predicted  $y'$  and the target  $y$  variable. For example, for the mean square error loss function  $L_{MSE}$ , this is done as  $1/N \sum (y_i - y'_i)^2$  with  $N$  the number of samples and the subscript  $i$  the  $i$ th sample. Although one of the simplest supervised ML algorithms in ML is linear regression (LR), in choosing the type of ML algorithm, I looked at

---

<sup>9</sup> The main sources in this section are Bishop (2006) and Goodfellow et al. (2016).

<sup>10</sup> There is an ongoing debate about whether ML is a subset of artificial intelligence (AI) or whether it is separate or intersects with that of AI. In this PhD thesis, I treat ML as a subfield of AI (IBM, 2021) but this this is not critical to this work.

several ML techniques with the simplest algorithm for the problem at hand. LR is the simplest ML technique, but as it cannot model nonlinearities typical of the problems described in this thesis, it was rejected (i.e., radiation, convection; note that a first assessment using LR was conducted in chapter 3 but showed poor results). Gaussian processes (GPs) are an excellent choice for modelling uncertainties. However, they suffer from cubic complexity to the dataset's number of samples, limiting their applicability. Finally, although tree-based algorithms such as random forest algorithms have shown promising results in the past with the added benefit of being more interpretable than Neural Networks (NNs), NNs had already been used and found to be a suitable algorithm to model radiative transfer problems (Chapters 3 and 4). Therefore, in this thesis, I shall focus on using the multilayer perceptron (MLP), one of the simplest types of artificial neural networks (ANN) widely used in ML and atmospheric sciences, capable of capturing nonlinearities between input and outputs.

An MLP-based NN is composed of a sequence of layers connected and with each layer forming a collection of neurons. Inputs move through a series of one or more hidden layers to the output layer. A single neuron can be represented mathematically as  $y = f(w^T x)$  with  $f$  the activation function and  $w$  the weight (including the bias). Activation functions introduce nonlinearity beyond just linear functions. Several non-linear activation functions exist, but one of the most common activation functions used for regression problems is the rectified linear unit (ReLU), defined as  $\max(0, x)$ . Although linear regression can be solved analytically using the normal equation, more complex models such as MLPs, need to be solved iteratively using numerical algorithms such as stochastic gradient descent (SGD) which takes small batches of data to make the training with large datasets computationally feasible. In this research, I use the Adam algorithm (Kingma & Ba, 2015)—an adaptive learning rate algorithm based on SGD.

Training MLPs require data to be split into train, evaluation (usually referred to as validation), and test parts with data reshaped to two-dimensional matrices with each sample (observation) as row and flattened quantities (e.g. air temperature, humidity) as column (i.e. feature). While train data are used as input to the optimisation algorithm, evaluation data are used to verify the performance of different network configurations (e.g., number of layers or neurons) and their impact on the results. Test data are used to ensure that the network is learning beyond the training data; that is, they do not overfit and are general enough to be used with future, unseen data. This is generally achieved using one or more regularizers that act to penalise the model and stop it from over-fitting. The most common types are L1 and L2: the former adds the absolute value of the weight to the loss function while the latter adds the squared weight.

As noted in chapter 2.2, although weather forecasting and climate projection are based on similar models, they answer fundamentally different questions. As noted in Watson-Parris (2021) the key uncertainties in weather prediction are to do with the internal variability due to the chaotic nature of the atmosphere over different time scales. In contrast, in climate projection, the critical uncertainties are to do with the incomplete representation of processes, the uncertainty of input parameters and assumptions, and incomplete knowledge of the emissions pathways of greenhouse gases, aerosol, and other short-lived climate forcings are critical in climate projections. (Hawkins & Sutton, 2009; Watson-Parris, 2021; Wilcox et al., 2020). As such, the development and testing of emulators need to be distinct. Here, however, the focus is on emulating parametrizations for weather applications, as are the evaluations.

## CHAPTER 3

---

# A Faster, More Accurate Radiation Scheme<sup>11</sup>

*In the first research chapter, I begin with the application of ML to radiation. Specifically, I look at a way to improve a radiation scheme used operationally by including three-dimensional cloud effects.*

---

<sup>11</sup> This chapter is based on Meyer, D., Hogan, R. J., Dueben, P. D., & Mason, S. L. (2022). Machine Learning Emulation of 3D Cloud Radiative Effects. *Journal of Advances in Modeling Earth Systems*, 14(3). <https://doi.org/10.1029/2021MS002550>

# 3.1 Introduction

Solar (hereafter, shortwave) and thermal-infrared (hereafter, longwave) radiation are the primary drivers of atmospheric weather systems via their role in creating the equator-to-pole temperature gradient, while their interaction with greenhouse gases drives anthropogenic climate change. As such, their detailed representation in radiation schemes is essential for both weather and climate (e.g., Stephens 1984).

Historically viewed as one of the slowest components of atmospheric models, radiation schemes have been among the prime candidates for acceleration via machine learning (ML). Cheruy et al. (1996) are one of the first to develop a neural network (NN) emulator of longwave radiation, reporting accurate results for speedups of 3 and 3 000 times the reference broadband and narrowband models, respectively. Chevallier et al. (1998, 2000) extend Cheruy et al. (1996)'s work to the European Centre for Medium-Range Weather Forecasts' (ECMWF) 50-level longwave and shortwave radiation scheme, reporting promising results for a sixfold reduction in computational costs. Krasnopolsky et al. (2005) develop a longwave NN emulator in the NCAR's (National Center for Atmospheric Research) Community Atmosphere Model, reporting predicted heating rate root-mean-square errors between 0.26 and 0.33 K d<sup>-1</sup>, for a runtime reduction between 35 and 80 times the original schemes, while Krasnopolsky et al. (2008)'s emulation of shortwave radiation report successful multi-decadal simulations with offline scheme speedups of 150 and 20 times the original scheme, and errors of 0.34 and 0.19 K d<sup>-1</sup> for longwave and shortwave heating rates, respectively. More recently, Pal et al. (2019)'s emulation of shortwave and longwave radiation in the Super-Parameterized Energy Exascale Earth System Model report a speedup of about one order of magnitude while retaining between 90 % and 95 % of the original scheme's accuracy.

Although these findings are encouraging, an assessment of Chevallier et al. (1998, 2000)'s emulators by Morcrette et al. (2008) report degraded accuracy and performance when the number of levels increases above the original 50 levels. Ukkonen et al. (2020) note that differences in radiative fluxes may sometimes be larger than the internal variability of the original scheme or with regional errors in annual-mean surface net fluxes of  $20 \text{ W m}^{-2}$ . Furthermore, although Roh and Song (2020) report average root-mean-square errors of 1.0 and  $0.49 \text{ K d}^{-1}$  for the longwave and shortwave heating rates, respectively, and 1.6 and  $14 \text{ W m}^{-2}$  for the longwave and shortwave fluxes, respectively, large deviations of about  $20 \text{ W m}^{-2}$  occur. Indeed, comparing these results is challenging as studies report their results using specific datasets and summarize them with different statistical metrics.

An important point to note when seeking applications of ML in radiative transfer is that radiation schemes are no longer a particularly slow component of atmospheric models, for example, in the ECMWF Integrated Forecast System (IFS), Hogan et al. (2017) report that the fractional time in the highest operational resolution model dropped from 19 % in 2007 to 5 % in 2017. On the other hand, several simplifications are still made such as:

1. operational radiation schemes cannot afford to represent the  $O(10^5)$  spectral lines explicitly and typically approximate the spectral variation of gas absorption by  $O(10^2)$  quasi-monochromatic radiative transfer calculations (Hogan and Matricardi 2020);
2. the accuracy of radiative forcing calculations due to changes in greenhouse gases in many schemes is questionable, with Soden et al. (2018) reporting that the spread of radiative forcing estimates due to increased  $\text{CO}_2$  is about 35 % of the mean;
3. the ways in which radiation interacts with cloud structure in radiation schemes is generally quite crude. Specifically, all radiation schemes used routinely in weather and climate models are "1D"; that is, they neglect the

full 3D interaction of radiation with clouds and therefore ignore the interception of direct sunlight by cloud sides, the trapping of sunlight beneath clouds, and the emission of thermal radiation from the sides of clouds (e.g., Hogan & Shonk, 2013; Varnai & Davies, 1999).

For cumulus clouds, the missing processes in reported in point 3 above—here defined as 3D cloud effects—can change the magnitude of instantaneous cloud radiative effects (i.e., the difference between fluxes in the presence of clouds and in the equivalent clear-sky conditions) by 30 % in the longwave (e.g., Heidinger and Cox 1996) and by up to 60 % in the shortwave, depending on sun angle (e.g., Pincus et al. 2005).

Although the method of Jakub and Mayer (2015) may be more appropriate for (sub)kilometer-resolution where radiative exchanges between atmospheric columns become important, to date, the fastest method we are aware of to represent 3D interactions of radiation with clouds within a model column suitable for large-scale models with horizontal resolution no finer than 5–10 km is the Speedy Algorithm for Radiative Transfer through Cloud Sides (SPARTACUS; Hogan et al., 2016). Despite this, SPARTACUS is approximately five times slower than the radiation scheme currently used at the ECMWF (Hogan & Bozzo, 2018)—far too slow to be considered for operational use.

As an alternative method to the emulation of an entire radiation scheme, Ukkonen et al. (2020) and Veerman et al. (2021) show a different approach whereby specific parts of a scheme, such as the gas optics in the RTE-RRTMGP (Radiative Transfer for Energetics and Rapid and accurate Radiative Transfer Model for General circulation models applications – Parallel; Pincus et al., 2019) framework, are emulated, retaining the original radiative transfer solver. Similar to the results by Veerman et al. (2021), Ukkonen et al. (2020) report heating rates and top-of-atmosphere longwave and shortwave radiative forcing root-mean-square errors relative to benchmark line-by-line radiation

calculations typically below  $0.1 \text{ K d}^{-1}$  and  $0.5 \text{ W m}^{-2}$ , respectively, relative to benchmark line-by-line radiation calculations (with smaller errors relative to RTE-RRTMGP), and speedups for clear-sky longwave and shortwave fluxes of 3.5 and 1.8 times the original scheme, respectively.

In the same spirit, here we investigate how to improve the representation of 3D cloud effects with a hybrid physical-ML method. Rather than replacing the entire radiation scheme, we run the existing 1D radiation scheme in parallel with an emulator trained on the difference between SPARTACUS and the 1D scheme. As the computational cost of the emulator is expected to be a fraction of that of SPARTACUS, we hope to achieve a similar accuracy for a fraction of the cost. Moreover, as heating rates are susceptible to vertical changes in fluxes, by only correcting profiles within the troposphere, we expect this approach to be more tolerant to errors in heating rates for higher parts of the atmosphere where low values of atmospheric pressure exacerbate comparatively small errors in predicted fluxes.

The following sections describe the general method, with specific details about reference models and data (Section 3.2.1) used to develop and train the ML emulators (Section 3.2.2). We follow with a qualitative (Section 3.3.1) and quantitative (Section 3.3.2) evaluation of the results, as well as a runtime performance analysis of the emulators (Section 3.3.3), before concluding with a summary and prospects for future work (Section 3.4).

## 3.2 Methods

### 3.2.1 Reference Model and Data

Reference simulations use the open-source atmospheric radiative transfer software ecRad (Hogan & Bozzo, 2018) version 1.3.0 (ECMWF, 2020). ecRad computes profiles of up- and downwelling, long- and shortwave radiative fluxes (with downwelling shortwave having both total and direct components) from zero- or one-dimensional (i.e., profiles) inputs of meteorological variables such as dry-bulb air temperature, cloud fraction, mixing ratios of water vapor, liquid water, ice cloud, snow, and trace gases. 3D cloud effects are computed as the difference between ecRad's 3D solver SPARTACUS and ecRad's 1D solver Tripleclouds (Shonk & Hogan, 2008). Although deterministic forecasts in the ECMWF Integrated Forecast System (IFS) use ecRad's 1D solver McICA (Monte Carlo Independent Column Approximation; Pincus et al. 2003). Tripleclouds is used here as (a) its flux predictions are noise-free—that is it does not introduce conditional random errors shown by McICA (Hill et al., 2011)—and (b) its underlying assumptions in cloud structure and overlap are the same as in SPARTACUS.

EcRad is forced with inputs from the EUMETSAT Numerical Weather Prediction Satellite Application Facility (NWP-SAF) data set (Eresmaa & McNally, 2014). This data set contains 25 000 atmospheric profiles representative of yearly, global, present-day atmospheric conditions on 137 atmospheric levels (surface to 0.01 hPa) from ECMWF operational forecasts between 2013 and 2014. Profiles of aerosol mixing ratio and greenhouse gas concentration are from the climatology of Bozzo et al. (2020) as a function of longitude, latitude, and month for the former, and latitude and month, for the latter. The prescribed horizontal cloud scale in SPARTACUS uses the parametrization of Fielding et al. (2020).

### 3.2.2 Neural Network Emulator

Two separate NNs to emulate short- and longwave 3D cloud effects are developed using the multilayer perceptron (MLP) architecture—a standard form of NN with inputs traveling via one or more hidden layers towards the outputs (Bishop, 2006)—following poor results from a preliminary investigation using linear regression (not reported). Both NNs are implemented in Python with TensorFlow (Abadi et al., 2015) version 2.4.1.

To capture the interaction of radiation with clouds, we compute the cloud optical depth  $\tau_c$  in the large particle limit where geometric optics is applicable, albeit ignoring small spectral dependences, as  $\frac{3}{2} \frac{\Delta p}{g} \left( \frac{q_l}{\rho_l r_l} + \frac{q_i}{\rho_i r_i} \right)$ , with  $\Delta p$  the difference in atmospheric pressure between two atmospheric layers,  $g$  the gravitational acceleration constant ( $9.81 \text{ m s}^{-2}$ ),  $q_l$  and  $q_i$  the liquid and ice mass mixing ratios,  $\rho_l$  and  $\rho_i$  the densities of liquid water and ice, and  $r_l$  and  $r_i$  the liquid and ice effective radii. Gas and aerosol properties, known to affect 3D effects minimally, are ignored. Heating rates  $dT/dt$  are computed from the *net* (downwelling *minus* upwelling) flux for an atmospheric layer  $i$  as  $-\frac{g}{c_p} \frac{F_{i+1/2}^n - F_{i-1/2}^n}{p_{i-1/2} - p_{i+1/2}}$ , with  $p_{i\pm 1/2}$  and  $F_{i\pm 1/2}^n$  the atmospheric pressure and net flux, respectively, at the layer interface  $i + 1/2$  and  $i - 1/2$  (counting down from the top of the atmosphere), and  $c_p$  the specific heat of dry air ( $1004 \text{ J kg}^{-1} \text{ K}^{-1}$ ).

As heating rates are proportional to the vertical derivative of the net flux, noise in predicted fluxes can amplify the errors in computed heating rates. Although training the NNs using fluxes and heating rates can partially mitigate this issue, predictions can no longer conserve energy. To avoid this issue, here we instead predict the 3D *scalar* (downwelling *plus* upwelling) flux and corresponding heating rates, as well as the direct downwelling shortwave flux, and postprocess the outputs in a separate step (Appendix A) to obtain energy-consistent downwelling and upwelling fluxes and heating rates. As we aim to

predict the 3D cloud effects, only levels between the surface and 50 hPa (i.e., assuming no clouds above the troposphere) are used. The full profiles, spanning 137 levels, are recovered by setting values between 50 and 0 hPa to zero for the downwelling component, and extending the last predicted value at 50 hPa to all levels between 50 and 0 hPa for the upwelling component.

Train, validation, and test datasets contain a random 60 % (13 702), 20 % (4 568), 20 % (4 568) selection of NWP-SAF profiles as inputs, and corresponding ecRad computed 3D cloud effect profiles (SPARTACUS minus Tripleclouds; Section 3.2.1) as outputs. Before being fed to the NNs, profiles are reshaped to two-dimensional matrices with each profile as row (sample) and flattened level and quantity as column (feature).

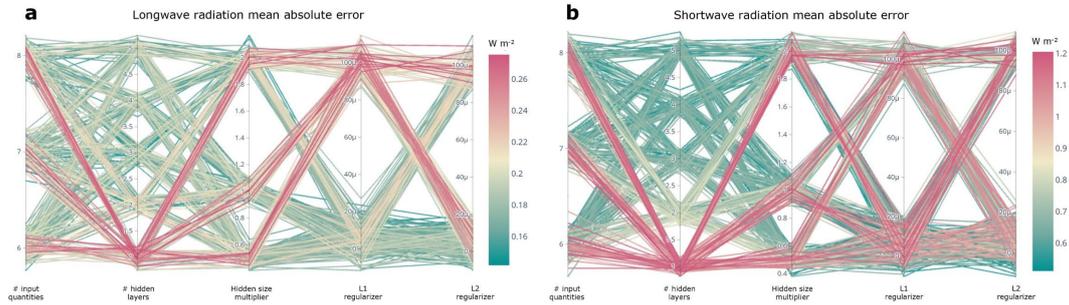
To determine the sensitivity to different choices of hyperparameters and input quantities, a grid search is conducted. In it, NNs are trained with NWP-SAF profiles of dry-bulb air temperature  $T$ , cloud fraction  $f_c$ , surface temperature  $T_s$ , surface albedo  $\alpha$ , cloud optical depth  $\tau_c$ , cosine of solar zenith angle  $\mu_0$ , specific humidity  $q$ , and vertical layer thickness  $\Delta z$  as inputs (Table 3.1a), and corresponding ecRad-computed 3D cloud effect profiles of scalar fluxes, heating rates, and direct downwelling shortwave as outputs (Table 3.1b). All configurations use the Exponential Linear Unit activation function, Adam optimiser with mean squared error on all outputs, and 1 000 epoch-limit with early stopping patience set to 50 epochs. The surface emissivity  $\varepsilon$  is not used as it is constant across profiles. Iterations are repeated 10 times to account for the stochasticity of the training algorithm. Hyperparameter choices are (a)  $\{f_c, \tau_c, T, T_s, \alpha, \mu\}$ ,  $\{f_c, \tau_c, T, T_s, \alpha, \mu, q\}$ , or  $\{f_c, \tau_c, T, T_s, \alpha, \mu, q, \Delta z\}$  for input quantities; (b)  $\{1, 2, 3, 4, 5\}$  for number of hidden layers; (c)  $\{0.5, 1, 2\}$  for hidden (neuron) size multipliers; and (d)  $\{10^{-6}, 10^{-5}, 10^{-4}\}$  for L1 and L2 regularization factors. The number of neurons in hidden layers is computed by multiplying the number of inputs (182 for shortwave and 271 for longwave) by the hidden size multiplier.

Results are visually inspected (Figure 1) and the simplest NN configuration (e.g., fewer neurons and input quantities) with the lowest mean absolute is chosen. For both longwave and shortwave components, this ‘optimal’ configuration is found to have three hidden layers, each with 217 and 182 neurons per hidden layer for longwave and shortwave, respectively, and L1 and L2 regularization factors set to  $10^{-5}$  (Figure 3.1). The most sensitive input quantities are:  $f_c$ ,  $T$ ,  $T_s$ ,  $\alpha$ ,  $\tau_c$ , and  $\mu_0$  (Table 3.1a).  $q$  and  $\Delta z$  are not used as they do not improve predictions (Figure 3.1). This is reasonable as (a) the cloud layer optical depth, which is proportional to layer thickness (for same cloud water mixing ratio), is already an input variable to the NNs, and (b) any dependence on humidity is likely captured by the dry-bulb air temperature for cloudy parts of the atmosphere. An increase in either the network size or the number of layers does not improve the overall accuracy (Figure 3.1). Convergence is achieved after approximately 100 epochs.

To improve the results further, the two NNs with the above-determined configuration are trained with more data. For this, we use Synthia (Meyer & Nagler, 2021) version 0.3.0 (Meyer & Nagler, 2020) as outlined in Meyer, Nagler, et al. (2021) but only for independent inputs. Thus, we (a) generate nine synthetic copies of the surface albedo  $\alpha$  and cosine of the solar zenith angle  $\mu_0$ , (b) randomly re-assign them to plain copies of NWP-SAF train-fraction profiles, and (c) collate them together to the original data set to form a total of 137 020 profiles for training (i.e., the original 13 702 profiles and 123 318 modified profiles). These augmented profiles are then used in ecRad to generate corresponding training outputs, and both augmented inputs and outputs to train the NNs. To account for the variability in the results given by the NN’s training algorithm, training (and inference) is run 20 times (10 times with and 10 times without data augmentation), varying random seeds between repeats. From this, the short- and longwave emulator with median mean absolute error are chosen. With this simple augmentation, the shortwave error is found to improve by about 18 %.

**Table 3.1 | Inputs and outputs used in the two NN emulators.** Vector quantities are either at the interface between two model layers (half level; HL), or at the model layer (full level; FL). The superscript “<sup>L</sup>” or “<sup>S</sup>” denotes if the input is used in the longwave or shortwave NN. The scalar flux is defined as downwelling plus upwelling flux.

Symbol	Name	Unit	Dimension
<i>(a) Inputs</i>			
$f_c$	<sup>L,S</sup> Cloud fraction	1	FL
$\tau_c$	<sup>L,S</sup> Cloud optical depth	1	FL
$T$	<sup>L</sup> Dry-bulb air temperature	K	FL
$T_s$	<sup>L</sup> Surface temperature	K	Scalar
$\alpha$	<sup>S</sup> Surface (shortwave) albedo	1	Scalar
$\mu_0$	<sup>S</sup> Cosine of solar zenith angle	1	Scalar
<i>(b) Outputs</i>			
$L^S$	3D effect on scalar longwave radiative flux density	$W m^{-2}$	HL
$S^S$	3D effect on scalar shortwave radiative flux density	$W m^{-2}$	HL
$S^\downarrow$	3D effect on downwelling direct shortwave radiative flux density	$W m^{-2}$	HL
$L^H$	3D effect on longwave heating rate	$K s^{-1}$	FL
$S^H$	3D effect on shortwave heating rate	$K s^{-1}$	FL



**Figure 3.1 | Mean absolute errors resulting from different hyperparameter configurations.** These are for the (a) longwave and (b) shortwave neural network. Each line represents a realization from a different hyperparameter configuration. Lines are shown slightly offset in the vertical axis for clarity. The search is conducted for configurations of (i) input quantities:  $\{f_c, \tau_c, T, T_s, \alpha, \mu\}$ ,  $\{f_c, \tau_c, T, T_s, \alpha, \mu, q\}$ ,  $\{f_c, \tau_c, T, T_s, \alpha, \mu, q, \Delta z\}$  shown as 6, 7, and 8, respectively; (ii) hidden layer size:  $\{1, 2, 3, 4, 5\}$ ; (iii) hidden (neuron) size multipliers:  $\{0.5, 1, 2\}$ ; and (iv) L1 and L2 regularization factors:  $\{10^{-6}, 10^{-5}, 10^{-4}\}$ . Hidden size multipliers are multiplied by the number of inputs (182 for shortwave and 271 for longwave) to obtain the number of neurons in each hidden layer.

## 3.3 Results and Discussion

### 3.3.1 Qualitative Evaluation

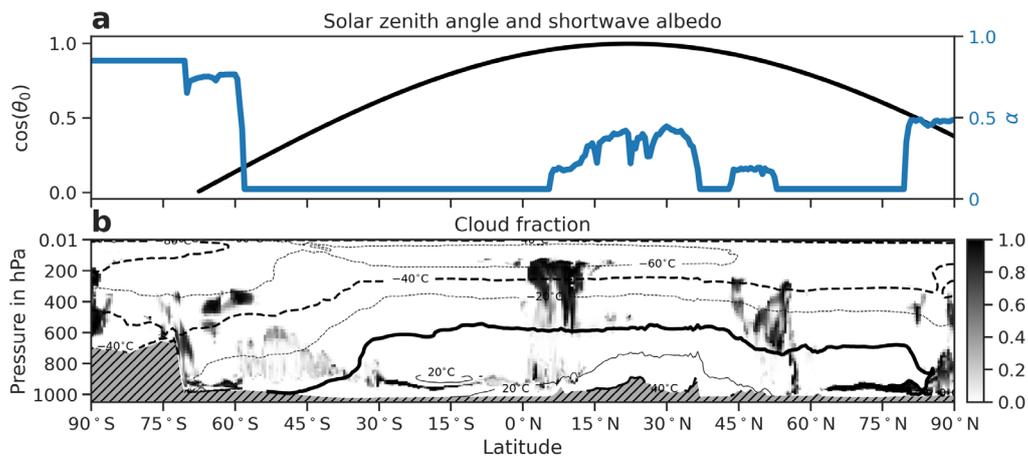
First, a separate visual inspection is conducted using an atmospheric slice of ERA5 reanalysis data (Hersbach et al., 2020), extending from north to south poles at a longitude of 5 °E at 12:00 UTC (Coordinated Universal Time) on 11 July 2019. This includes the response of radiation to Saharan dust, marine stratocumulus, deep convection, and Arctic stratus. The surface albedo, cosine of the solar zenith angle, and cloud fraction are shown in Figure 3.2. Figure 3.3 shows the outputs from SPARTACUS (left), reference 3D cloud effects (3D signal; SPARTACUS minus Tripleclouds; middle), and NN predictions (right), respectively.

The longwave effect of clouds (Figures 3.3a and 3.3d) is to warm the Earth system by reducing the upwelling radiation to space and increasing it towards the surface. When the 3D effects are simulated, clouds can not only interact with radiation through their base and top, but also through their sides. Thus, they further reduce the upwelling longwave radiation to space and further increase it towards the surface (Figures 3.3b and 3.3e).

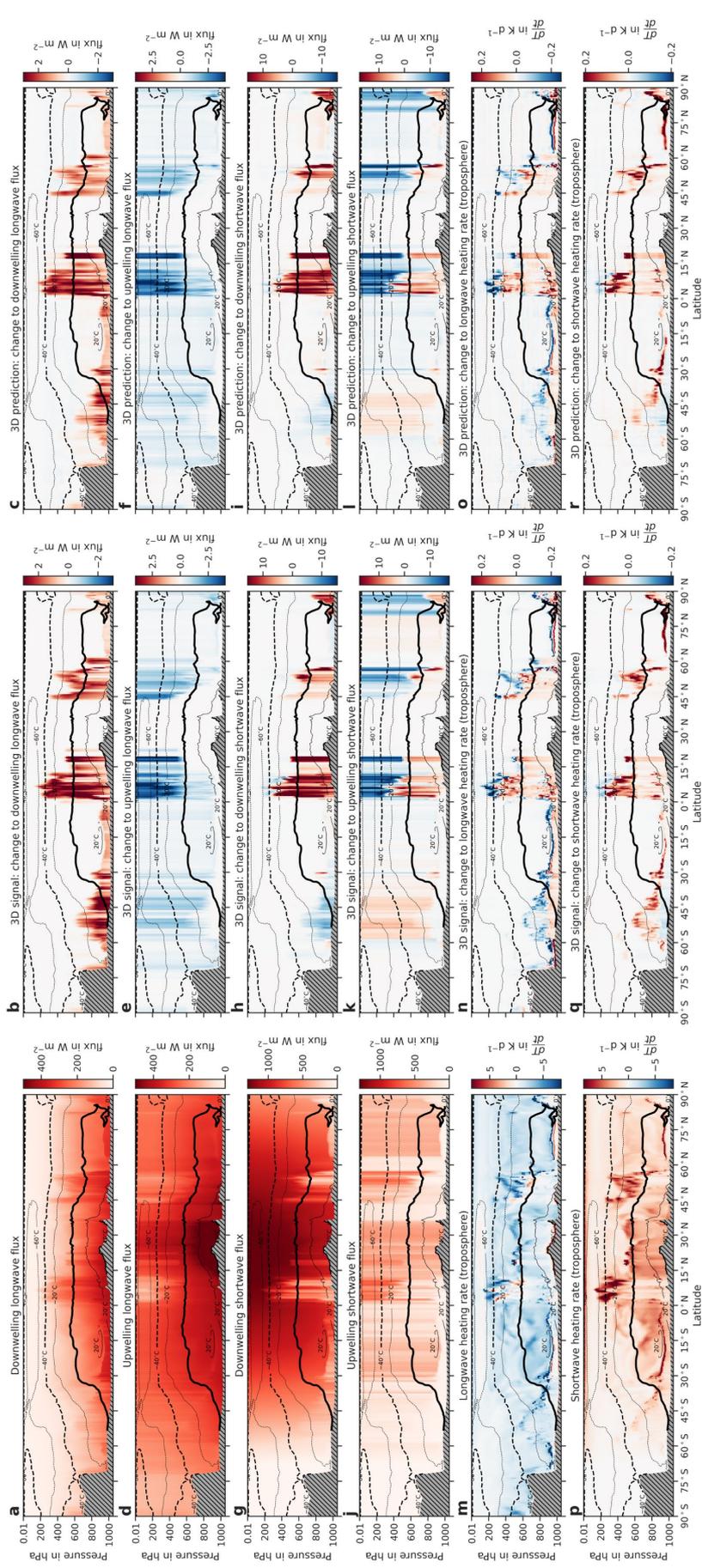
Figure 3.3n shows that the longwave heating rate signal of clouds is also amplified, increasing the magnitudes of cooling at cloud tops, and of warming at cloud bases (see Schafer et al., 2016 for further discussion). In the shortwave, the sign of the impact is dependent on solar zenith angle (Figure 3.3k): when the Sun is near its zenith, at the tropics, the 3D cloud effect acts to reduce the upwelling radiation reflected into space from cloud tops, but to increase it when near the horizon, over the Southern Ocean. These behaviors can be explained by the mechanisms of entrapment and side-illumination, respectively (Hogan et al., 2019). Although the vertical structure of heating rate is smoothed somewhat vertically in both the longwave

(Figure 3.3o) and shortwave (Figure 3.3r), the sign and size predicted by the NN is captured for high and low clouds and for high and low sun angles.

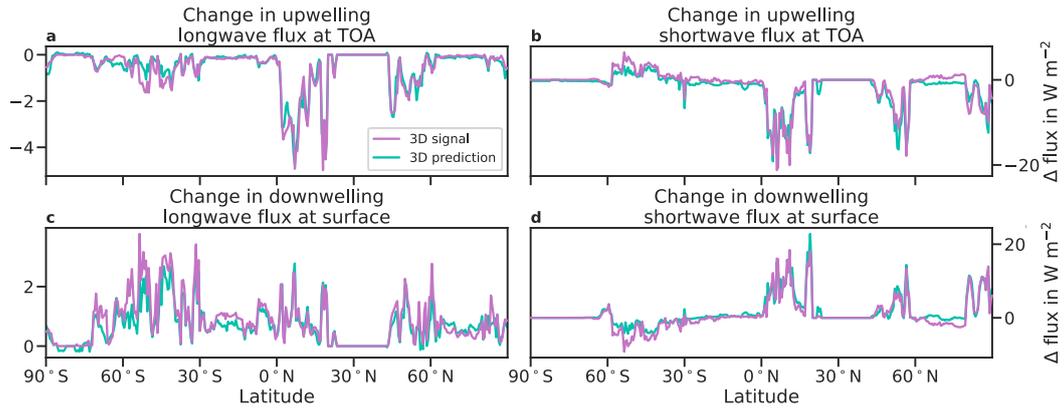
Figure 3.4 compares the 3D effects at top-of-atmosphere (TOA) upwelling fluxes and surface downwelling fluxes between the reference ecRad calculations and NN-predicted 3D cloud effects with generally good agreement across the range of latitudes.



**Figure 3.2 | Typical zero- and one-dimensional SPARTACUS and Tripleclouds inputs.** Here shown the (a) surface albedo ( $\alpha$ ; blue line), and cosine of the solar zenith angle  $\cos(\theta_0)$ ; black line; scalar quantities, and (b) cloud fraction (vector quantity) at 5°E on 11 July 2019 12:00 UTC from ERA5 reanalysis data (Hersbach et al., 2020). Vector quantities consist of 137 vertical levels, here shown using atmospheric pressure as coordinate. The hatched area shows the topography. Temperature contours are shown using dashed and dotted lines



**Figure 3.3 | Qualitative evaluation of a slice of the atmosphere.** (a-f) Longwave and (g-l) shortwave fluxes; and (m-o) longwave and (p-r) shortwave heating rates, computed with pole-to-pole slice at 5°E on 11 July 2019 12:00 UTC from ERA5 reanalysis data (Hersbach et al., 2020), of (a,d,g,j,m,p) SPARTACUS, (b,e,h,k,n,q) 3D signal (SPARTACUS minus Tripleclouds), and (c,f,i,l,o,r) 3D NN predictions. The slice includes the response of radiation to Saharan dust, marine stratocumulus, deep convection, and Arctic stratocumulus.



**Figure 3.4 | Comparison of 3D signal (SPARTACUS minus Tripleclouds; magenta) and 3D prediction (NN; cyan).** This is done for (a-b) top-of-atmosphere (TOA) upwelling flux and (c-d) surface downwelling flux using pole-to-pole slice at 5°E on 11 July 2019 12:00 UTC from ERA5 reanalysis data (Hersbach et al., 2020).

### 3.3.2 Quantitative Evaluation

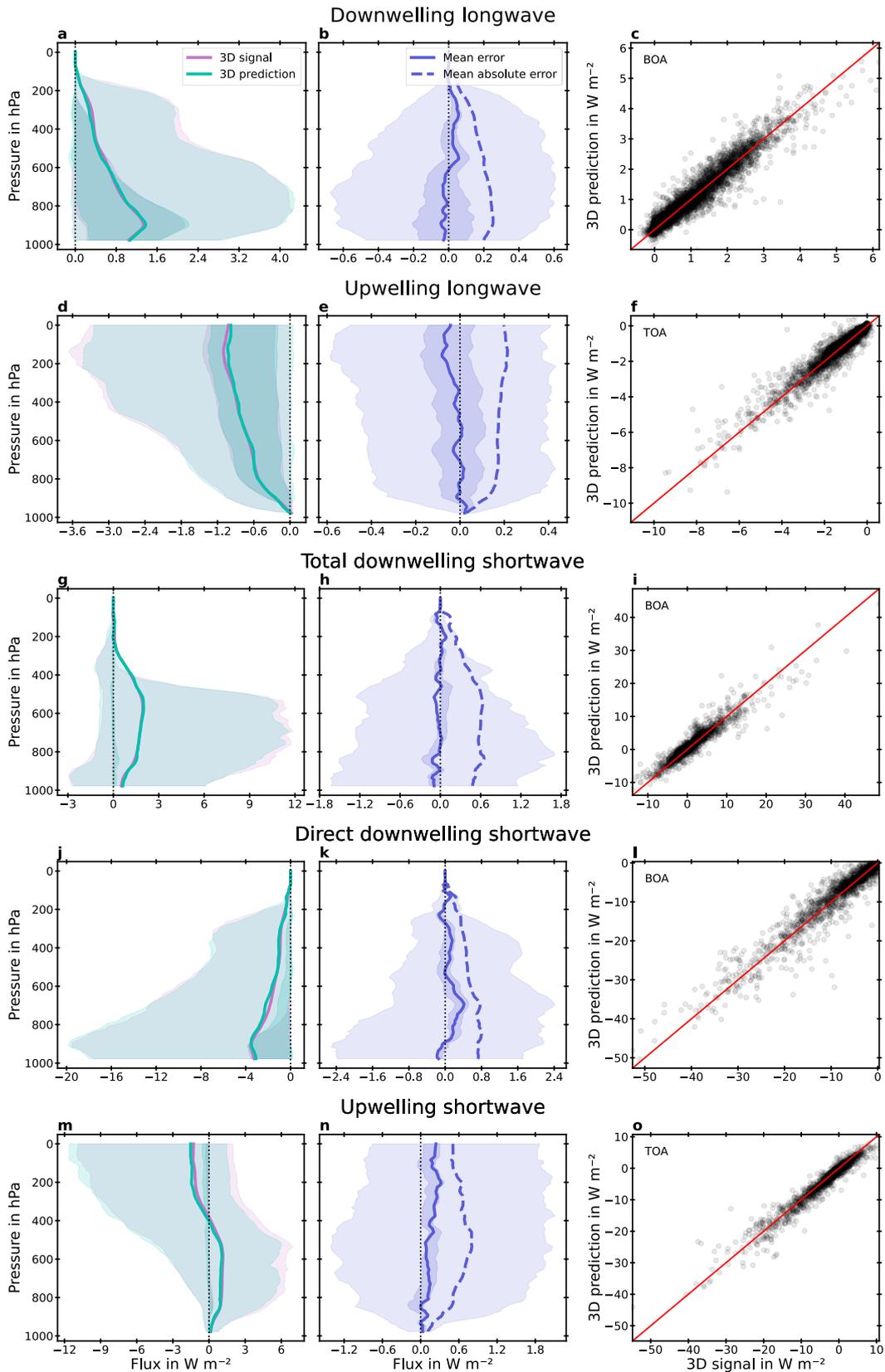
Second, a quantitative evaluation is made by comparing NN-predicted 3D cloud effects to reference ecRad calculations (3D signal; SPARTACUS minus Tripleclouds) using the test fraction (Section 3.2.1). Evaluation metrics are computed using profiles of either 3D signal, 3D predictions, or error (i.e., prediction minus signal) separately for each level, or with no distinction in vertical levels (hereafter referred to as bulk), for a vector  $\mathbf{y}$  of  $1 \dots N$  samples for the mean ( $\frac{1}{N} \sum_{i=1}^N y_i$ ) or mean absolute ( $\frac{1}{N} \sum_{i=1}^N |y_i|$ ). Per-level statistics are shown in Figure 3.5 for fluxes, and in Figure 3.6 for heating rates. Bulk error statistics are summarized in Table 3.2.

The first column in Figure 3.5 shows per-level means, and 50 % and 90 % quantiles of 3D signal and NN predictions. On average, the 3D signal is approximately  $1 \text{ W m}^{-2}$  for the longwave (Figures 3.5a and 3.5d) and about  $3 \text{ W m}^{-2}$  at the surface for the shortwave (Figure 3.5j). To put these numbers into context, the radiative forcing from doubling carbon dioxide concentrations from preindustrial levels is around  $3.7 \text{ W m}^{-2}$  (Forster et al. 2007). Visually, NN predictions are close to the mean reference 3D signal (Figures 3.5a, 3.5d, 3.5g, 3.5j, and 3.5m). The 3D error for the mean (solid) and mean absolute (dashed)

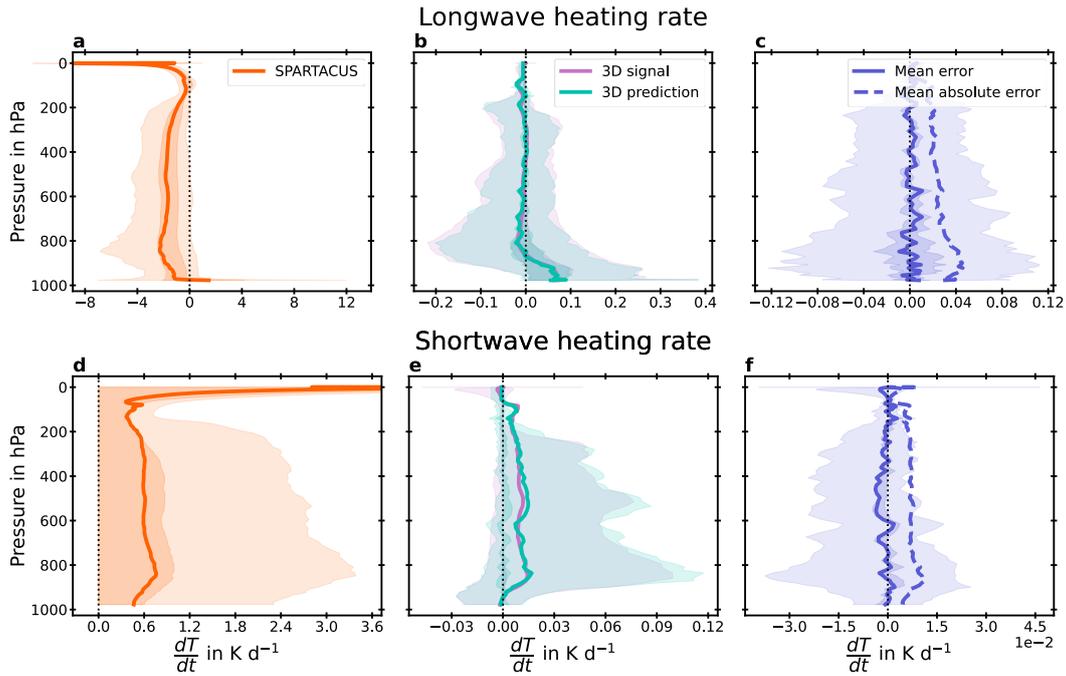
is shown in the second column of Figure 3.5. This reaches about  $0.2 \text{ W m}^{-2}$  for the longwave (Figures 3.5b and 3.5e) and about  $0.6 \text{ W m}^{-2}$  for the shortwave (Figures 3.5h, 3.5k, and 3.5n). Similarly to the qualitative assessment in Figure 3.4, scatterplots of top-of-atmosphere upwelling, and surface downwelling (Figures 3.5c, 3.5f, 3.5i, 3.5l, and 3.5o) flux predictions are close to reference calculations across the range of values.

The third column in Figure 3.6 shows larger errors in the vertical structure of the 3D effects on heating rates (Figures 3.6c and 3.6f). The size of 3D effects on heating rates (Figures 3.6b and 3.6e) is, however, about two orders of magnitude smaller than the absolute heating rates from SPARTACUS (Figures 3.6a and 3.6d).

Table 3.2 summarizes bulk error statistics for fluxes and heating rates. NN errors are generally small. The mean percentage error is below 20 %, except for the upwelling shortwave where it is  $-96 \%$ . This latter result is not particularly interesting, however, as the mean 3D cloud effect for the entire upwelling component is about  $-0.16 \text{ W m}^{-2}$ —much smaller than that at the top of the atmosphere of  $-1.3 \text{ W m}^{-2}$ . The mean absolute percentage error of fluxes is about 20 % and 30 %; in other words, NN predictions capture about 70 % and 80 % of the 3D effects predicted by SPARTACUS. For heating rates, the mean and mean absolute percentage errors are 15 % and 66 % for the longwave, and  $-6.1 \%$  and 62 % for the shortwave. This latter result is not particularly important as 3D effects on heating rates are small, about  $0.01 \text{ K d}^{-1}$  for the shortwave. Indeed, the primary means by which shortwave 3D effects influence the Earth system is via a change in surface fluxes, and from there the surface temperature.



**Figure 3.5 | Per-level statistics of fluxes.** Mean (a,d,g,j,m), 3D signal (SPARTACUS minus Tripleclouds) and 3D prediction (NN), (b,e,h,k,n) mean (continuous) and mean absolute (dashed) error (3D prediction minus 3D signal), and (c,f,i,l,o) scatterplots of 3D top-of-atmosphere (TOA) and bottom-of-atmosphere (BOA) fluxes, computed using the test fraction. 50 % (lighter) and 90 % (darker) quantiles are shown for right and middle panels.



**Figure 3.6 | Per-level statistics of heating rates.** (a,d) mean absolute heating rates from SPARTACUS, (b,e) mean 3D signal and prediction, and (c,f) mean (continuous) and mean absolute (dashed) error (3D prediction minus 3D signal), computed using the test fraction. 50 % (lighter) and 90 % (darker) quantiles are shown for all panels.

**Table 3.2 | Bulk error statistics.** Bulk mean and mean absolute 3D signal (SPARTACUS minus Tripleclouds), 3D error (prediction minus signal), and percentage error (3D error divided by 3D signal, multiplied by 100) for (a) longwave ( $L$ ) and shortwave ( $S$ ) fluxes, and (b) corresponding heating rates ( $L^H$ ,  $S^H$ ). Flux components are shown for the total upwelling ( $\uparrow$ ), downwelling ( $\downarrow$ ), and direct downwelling ( $\Downarrow$ ), and separately for top-of-atmosphere (TOA), and bottom-of-atmosphere (BOA).

	Mean			Mean Absolute		
	3D signal	3D error	Percentage error	3D signal	3D error	Percentage error
<i>(a) Fluxes</i>						
	$\text{W m}^{-2}$	$\text{W m}^{-2}$	%	$\text{W m}^{-2}$	$\text{W m}^{-2}$	%
$L\downarrow$	0.55	0.00048	0.087	0.56	0.14	25
$L_{\text{BOA}}\downarrow$	1.0	-0.029	-2.8	1.1	0.2	19
$L\uparrow$	-0.72	-0.025	3.4	0.73	0.17	23
$L_{\text{TOA}}\uparrow$	-1.1	-0.071	6.7	1.1	0.2	19
$S\downarrow$	0.73	-0.032	-4.4	1.1	0.38	33
$S_{\text{BOA}}\downarrow$	0.53	-0.092	-17	1.4	0.48	35
$S\Downarrow$	-1.5	0.037	-2.5	1.5	0.45	30
$S_{\text{BOA}}\Downarrow$	-3.2	-0.15	4.5	3.2	0.72	22
$S\uparrow$	-0.16	0.15	-96	1.6	0.52	32
$S_{\text{TOA}}\uparrow$	-1.3	0.22	-17	1.9	0.51	27
<i>(b) Heating Rates</i>						
	$\text{K d}^{-1}$	$\text{K d}^{-1}$	%	$\text{K d}^{-1}$	$\text{K d}^{-1}$	%
$L^H$	0.0069	0.001	15	0.037	0.024	66
$S^H$	0.0066	-0.0004	-6.1	0.0099	0.0062	62

### 3.3.3 Runtime Performance

The emulators' runtime performance is assessed using a normalized runtime performance metric defined as total model runtime divided by total number of profiles. The total model runtime includes measurements of data normalization, inference with TensorFlow in Python, data denormalization, and postprocessing (Appendix A). To reduce input, output, and runtime overheads during measurements, the input data is replicated 10 times and the batch size of TensorFlow set to 50 000. Both ecRad and the NN are run fully single-threaded in Singularity (Kurtzer et al., 2017) with Ubuntu 18.04, GNU Fortran 7.5.0 compiler, Anaconda Python 3.8 and TensorFlow 2.4.1 on a shared AMD EPYC 7742 node with 32 CPUs and 124 GiB of system memory on a shared cluster.

SPARTACUS is about 4.58 times slower than Tripleclouds with an average of  $11.6 \pm 0.0196$  ms per profile, compared to  $2.53 \pm 0.00854$  ms for Tripleclouds. In comparison, the two NNs predicting 3D effects take  $0.0257 \pm 0.0000372$  ms per profile. Thus, the combined time for running both Tripleclouds and the two NNs is  $2.56 \pm 0.00856$  ms per profile, an increase of about 1.19 % of Tripleclouds' runtime. A key reason for the NNs being so fast is that they predict broadband quantities directly, rather than integrating over many spectral intervals (140 in the longwave and 112 in the shortwave) as done in Tripleclouds and SPARTACUS.

While these absolute runtimes are expected to change when run on different hardware, or coupled to the IFS, relative differences are indicative of the order of speedup. With graphics processing units (GPUs) likely playing a significant role in future high-performance computing systems (Bauer, Dueben, et al., 2021), switching to GPUs is generally a trivial task with ML libraries such as TensorFlow.

## 3.4 Conclusion

In this paper we propose a hybrid physical machine learning approach to correct a fast but less accurate 1D radiative transfer scheme with two neural network emulators of shortwave and longwave 3D cloud effects. The emulators are trained on the difference between a 3D (SPARTACUS) and a 1D (Tripleclouds) solver.

Results show that the 3D effects on fluxes are captured with bulk mean absolute errors between 20 % and 30 % of the 3D signal (Figures 3.3–3.5; Table 3.2). To put these results into perspective, Hogan et al. (2019) report the same error range, albeit with biases of about  $0.3 \text{ W m}^{-2}$ , for comparing the shortwave component of SPARTACUS to Monte Carlo simulations of 65 3D cloud scenes. Although profiles of heating rates show large mean absolute errors of up to 66 % (Table 3.2), the impact of 3D cloud effects on heating rates is up to two orders of magnitude smaller than that of the absolute heating rates (cf. Figures 3.6a–3.6d vs. Figures 3.6c–3.6f). As the 3D effects for top-of-atmosphere upwelling fluxes and surface downwelling fluxes are constantly improved, this hybrid physical machine learning approach may be valuable in operational settings, where the computational performance of a parametrization scheme is often a limiting factor for its uptake. Here, clear-sky fluxes are efficiently and accurately computed using Tripleclouds, and cloudy profiles are corrected with neural network emulators that have a negligible impact on Tripleclouds' runtime performance ( $\sim 1 \%$ ; Section 3.3.3).

Although further improvements in emulating radiative transfer processes may be achieved with other types of network architectures (e.g., Ukkonen, 2021), the use of large domain-specific datasets such as those recently published as part of the MACHinE Learning for Scalable meTeORology and climate project (see A3 in Dueben et al., 2021), or of data augmentation strategies (e.g., as implemented by Meyer, Nagler, et al., 2021) may help to further improve the

accuracy and generalization of current emulators. As the number of vertical levels in the current emulator is fixed, retraining may be necessary if levels in the atmospheric model increase. However, we expect this to be a minor limitation as changes in operational components are often on a much longer time scale (i.e., a few years) than those needed to retrain and retest emulators. While we show that the emulation of 3D cloud effects is a promising area of research, it is only the first step toward operationalization. As new model capabilities may only be used operationally at the ECMWF if found to improve forecast skills, online evaluations within the ECMWF Integrated Forecast System, need to assess our findings in the broader context on skill scores and numerical stability: compensating errors in cloud-radiation interactions mean that changes in their representation may degrade forecast scores unless accompanied by other modifications (Haiden et al., 2018; Martin et al., 2010) and further influence a model's stability.

Current research highlights challenges with NN emulators coupled to Earth system models, reporting degraded performance and unstable simulations under some circumstances (Brenowitz & Bretherton, 2019; Rasp et al., 2018). While our recent experience in emulating gravity wave drag (Chantry et al., 2021) and urban land surface (Meyer, Grimmond, et al., 2022) schemes was positive, long coupled evaluations are required to better assess these type of models for operational use.

*Although in this chapter basic uncorrelated synthetic data are generated using the Synthia tool to augment atmospheric profiles of shortwave radiation, in chapter 4 a new, more complex method to improve the accuracy of ML emulators is investigated. This involves generating dependent data for zero- and one-dimensional cases, typical of current parametrization schemes used within NWP models.*

# CHAPTER 4

---

## Boost Data, Boost Predictions<sup>12</sup>

*Chapter 3 discusses the benefits of enriching the input dataset for an ML emulator of shortwave radiation by generating profiles with a greater variety of surface albedo and cosine of the solar zenith angle. In chapter 4, I develop a general approach for generating complex, statistically dependent data and demonstrate it using a toy longwave radiation model.*

---

<sup>12</sup> This chapter is based on Meyer, D., Nagler, T., & Hogan, R. J. (2021). Copula-based synthetic data augmentation for machine-learning emulators. *Geoscientific Model Development*, 14(8), 5205–5215. <https://doi.org/10.5194/gmd-14-5205-2021>

# 4.1 Introduction

The use of machine learning (ML) in weather and climate is becoming increasingly popular (Huntingford et al., 2019; Reichstein et al., 2019). ML approaches are being applied to an increasingly diverse range of problems for improving the modelling of radiation (e.g., Cheruy et al., 1996; Chevallier et al., 1998, 2000; Krasnopolsky et al., 2005; Meyer et al., 2021; Ukkonen et al., 2020; Veerman et al., 2021), ocean (e.g., Bolton and Zanna, 2019; Krasnopolsky et al., 2005), chemistry (e.g., Nowack et al., 2018), and convection (e.g., Krasnopolsky et al., 2013), as well as the representation of sub-grid processes (e.g., Brenowitz and Bretherton, 2018; Gentine et al., 2018; O’Gorman and Dwyer, 2018; Rasp et al., 2018), and the post-processing of model outputs (e.g., Krasnopolsky and Lin, 2012; Rasp and Lerch, 2018).

When it comes to training ML models for weather and climate applications two main strategies may be identified: one in which input and output pairs are directly provided (e.g., both come from observations) and a second in which inputs are provided but corresponding outputs are generated through a *physical model* (e.g., parametrization schemes or even a whole weather and climate model). Although the former may be considered the most common training strategy in use today, when the underlying physical processes are well understood (e.g., radiative transfer) and numerical codes are available, the latter may be of particular interest for developing one-to-one *emulators* (i.e., statistical surrogates of their physical counterparts), which can be used to improve computational performance for a trade-off in accuracy (e.g., Chevallier et al., 1998; Meyer et al., 2021; Ukkonen et al., 2020; Veerman et al., 2021). Here, for clarity, we will only be focusing on the latter case and refer to them as emulators.

In ML, the best way to make a model more generalizable is to train it on more data (Goodfellow et al., 2016). However, depending on the specific field and

application, input data may be scarce, representative of only a subset of situations and domains, or, in the case of synthetically generated data, require large computational resources, bespoke infrastructures, and specific domain knowledge. For example, generating atmospheric profiles using a general circulation model (GCM) may require in-depth knowledge of the GCM and large computational resources (e.g., data used in Meyer, Hogan, et al., 2022).

A possible solution to these issues may be found by augmenting the available input dataset with more samples. Although this may be a straightforward task for classification problems (e.g., by translating or adding noise to an image), this may not be the case for parametrizations of physical processes used in weather and climate models. In this context, it is common to work with high-dimensional and strongly dependent data (e.g., between physical quantities such as air temperature, humidity, and pressure across grid points). Although this dependence may be well approximated by simple physical laws (e.g., the ideal gas law for conditions found in the Earth's atmosphere), the generation of representative data across multiple dimensions for most weather and climate applications is challenging (e.g., the nonlinear relationship between cloud properties, humidity, and temperature).

To serve a similar purpose as real data, synthetically generated data thus need to preserve the statistical properties of real data with respect to (1) the individual behaviour of variables (e.g., the dry-bulb air temperature at a specific level) and (2) the dependence across variables and dimensions (e.g., the dry-bulb air temperature across two levels). Copulas are statistical models that allow these two aims to be disentangled (Trivedi & Zimmer, 2006; Joe, 2014) and to generate new samples that are statistically similar to the original data in terms of their individual behaviour and dependence.

Aside from copulas models (e.g., Patki et al., 2016), other methods may be suitable for generating synthetic data such as variational autoencoders (e.g.,

Wan et al., 2017), and, more recently, generative adversarial networks (GANs; e.g., Xu and Veeramachaneni, 2018). Although the use of GANs for data generation is becoming increasingly popular among the core ML community, they require multiple models to be trained, leading to difficulties and computational burden (Tagasovska et al., 2019). Variational approaches, on the other hand, make strong distributional assumptions that are potentially detrimental to generative models (Tagasovska et al., 2019). Compared to black-box deep-learning models, the training of (vine) copulas is relatively easy and robust, while taking away a lot of guesswork in specifying hyperparameters and network architecture. Furthermore, copula models give a direct representation of statistical distributions, making them easier to interpret and tweak after training. As such, copula-based models have been shown to be effective for generating synthetic data comparable to real data in the context of privacy protection (Patki et al., 2016).

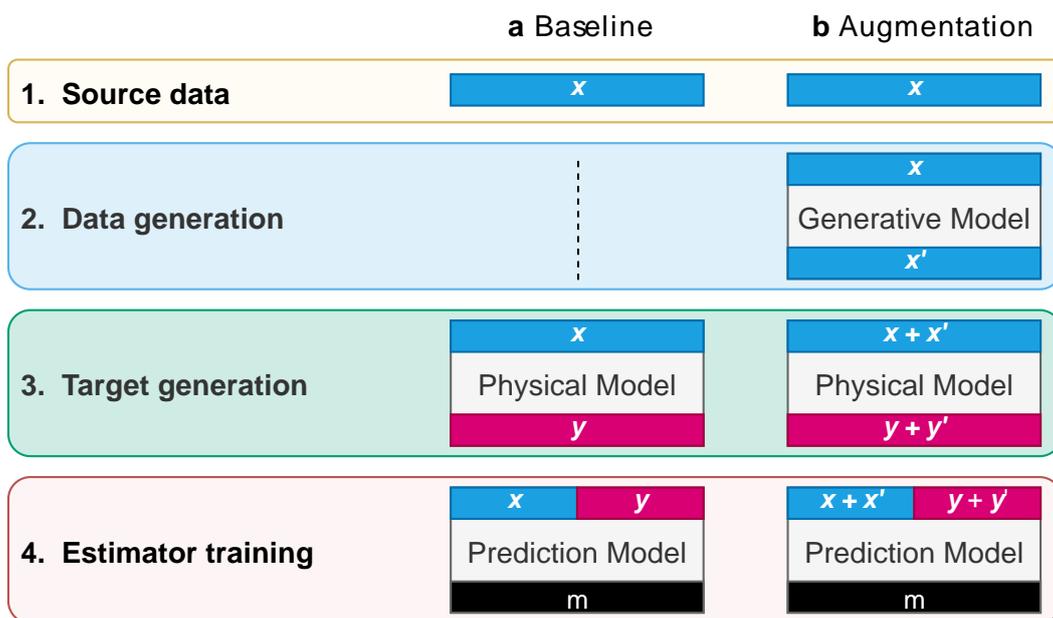
The goal of this paper is to improve ML emulators by augmenting the physical model’s inputs using copulas. We give a brief overview of methods in Section 4.2.1 with specific implementation details in Section 4.2.2–4.2.5. Results are shown in Section 4.3, with a focus on evaluating synthetically generated data in Section 4.3.1 and ML predictions in Section 4.3.2. We conclude with a discussion and prospects for future research in Section 4.4.

## 4.2 Materials and Methods

### 4.2.1 Overview

The general method for *training* an ML emulator for a set of  $N$  samples involves the use of paired *inputs*  $\mathbf{x} = \{x_1, \dots, x_N\}$  and *outputs*  $\mathbf{y} = \{y_1, \dots, y_N\}$  to find the best function approximation for a specific architecture and configuration. For *inference*, the trained ML emulator is then used to predict new outputs  $\mathbf{y}^*$  from inputs  $\mathbf{x}^*$ . Outputs  $\mathbf{y}$  are generated through a physical model from  $\mathbf{x}$  and fed

to the ML emulator for training (Figure 4.1a). In this paper we introduce an additional step: augmentation through copula-based synthetic data generation (Figure 4.1b). The method is demonstrated with a toy model of downwelling radiation as the physical model (Section 4.2.4) and a simple feed-forward neural network (FNN) as the ML emulator (Section 4.2.5). To evaluate the impact of copula-generated synthetic data on predictions we focus on predicting vertical profiles of longwave radiation from those of dry-bulb air



**Figure 4.1 | General strategies for training ML emulators.** (a) Inputs  $x$  are fed to the physical model to generate corresponding outputs  $y$ ;  $x$  and  $y$  are used to train the ML emulator. (b) A data generation model (here copula) is fitted to inputs  $x$  to generate synthetic inputs  $x'$ ; inputs  $x$  and  $x'$  are fed to the physical model to generate corresponding outputs  $y$  and  $y'$ ; both  $x$ ,  $x'$  and  $y$ ,  $y'$  are used to train the ML emulator. After training, the model ( $m$ ; e.g., architecture and weights) is saved and used for inference on new data.

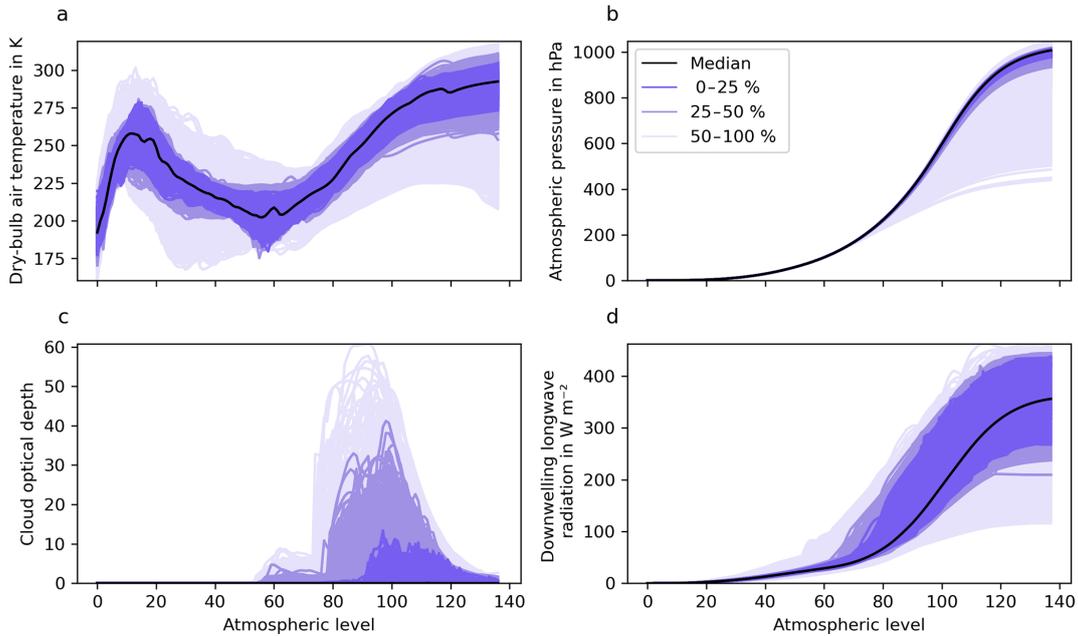
temperature, atmospheric pressure, and cloud optical depth (other parameters affecting longwave radiative transfer, such as gas optical depth, are treated as constant in the simple model described in Section 4.2.4). This task is chosen as it allows us to (1) evaluate copula-based models for generating correlated multidimensional data (e.g., with dependence across several quantities and grid points), some of which (e.g., cloud optical depth) are highly non-Gaussian; (2) develop a simple and fast toy physical model that

may be representative of other physical parametrizations such as radiation, (urban) land surface, cloud, or convection schemes; and (3) develop a fast and simple ML emulator used to compute representative statistics. Here we define case (a) as the *baseline* and generate six different subcases for case (b) using (1) three levels of data *augmentation factors* (i.e., either 1x, 5x, or 10x the number of profiles in the real dataset) (2) generated from three different copula types. In the following Sections we give background information and specific implementation details about the general method used for setting up the source data (Section 4.2.2), data generation (Section 4.2.3), target generation (Section 4.2.4), and estimator training (Section 4.2.5) as shown in Figure 4.1b.

## 4.2.2 Source Data

Inputs are derived from the EUMETSAT Numerical Weather Prediction Satellite Application Facility (NWP-SAF; Eresmaa and McNally, 2014) dataset. This contains a representative collection of 25 000 atmospheric profiles previously used to evaluate the performance of radiation models (e.g., Hocking et al., 2021; Hogan and Matricardi, 2020). Profiles were derived from 137 vertical-level global operational short-range ECMWF forecasts correlated in more than one dimension (between quantities and spatially across levels) and extending from the top of the atmosphere (TOA; 0.01 hPa; level 1) to the surface (bottom of the atmosphere; BOA; level 137). Inputs consist of profiles of dry-bulb air temperature ( $T$  in K; Figure 4.2a), atmospheric pressure ( $p$  in hPa; Figure 4.2b), and cloud layer optical depth ( $\tau_c$ ; Figure 4.2c).  $\tau_c$  is derived from other quantities to simplify the development of models as described in Section 4.2.4. Dry-bulb air temperature, atmospheric pressure, and cloud layer optical depth are then used as inputs to the physical model (Section 4.2.4) to compute outputs containing profiles of downwelling longwave radiation ( $L^\downarrow$  in  $W\ m^{-2}$ ; Figure 4.2d). As both copula models and ML emulator work on two-dimensional data, data are reshaped to input  $\mathbf{X}$  and output  $\mathbf{Y}$  matrices with each profile as row (sample) and flattened level and quantity as column

(feature) and reconstructed to their original shape where required. Prior to being used, source data are shuffled at random and split into three batches of 10 000 profiles (40 %) for training ( $\mathbf{X}_{\text{train}}, \mathbf{Y}_{\text{train}}$ ), 5 000 (20 %) for validation ( $\mathbf{X}_{\text{val}}, \mathbf{Y}_{\text{val}}$ ), and 10 000 (40 %) for testing ( $\mathbf{X}_{\text{test}}, \mathbf{Y}_{\text{test}}$ ).



**Figure 4.2 | Atmospheric profiles used in this study.** (a) dry-bulb air temperature, (b) atmospheric pressure, and (c) cloud layer optical depth from the NWP-SAF dataset (25 000 profiles; Eresmaa and McNally, 2014) as well as (d) corresponding profiles of longwave radiation computed using the toy physical model described in Section 4.2.4. Profiles are ordered using band depth statistics (López-Pintado and Romo, 2009), shown for their most central (median) profile, and grouped for the central 0 %–25 %, 25 %–50 %, and 50 %–100 %.

**Table 4.1 | Profiles of input and output quantities used in this study.** Input quantities are dry-bulb air temperature  $T$ , atmospheric temperature  $p$ , and cloud layer optical depth  $\tau_c$ .  $T$  and  $p$  are taken directly from the NWP-SAF dataset (Eresmaa and McNally, 2014), and  $\tau_c$  is derived from other quantities as described in Section 4.2.4. The output quantity downwelling longwave radiation  $L^\downarrow$  is computed using the physical model described in Section 4.2.4. Atmospheric model levels are 137 for full levels (FLs) and 138 for half-levels (HLs).

Symbol	Name	Unit	Dimension
<i>Inputs</i>			
$T$	Dry-bulb air temperature	K	FL
$p$	Atmospheric pressure	Pa	FL
$\tau_c$	Cloud optical depth	1	FL
<i>Output</i>			
$L^\downarrow$	Downwelling longwave radiation	$\text{W m}^{-2}$	HL

## 4.2.3 Data Generation

Data generation is used to generate additional input samples (here atmospheric profiles) to be fed to the physical model (Section 4.2.4) and ML (Section 4.2.5) emulator. As mentioned in the Introduction (Section 4.1) copula-generated synthetic data should thus resemble the original data as closely as possible with respect to the individual behaviour of variables and dependence across variables and dimensions.

### 4.2.3.1 Background on Copula Models

Suppose we want to generate synthetic data from a probabilistic model for  $n$  variables  $Z_1, \dots, Z_n$ . To achieve the first aim, we need to find appropriate *marginal cumulative distributions*  $F, \dots, F_n$ . A simple approach is to approximate them by the corresponding empirical distribution functions. To achieve the second aim, however, we need to build a model for the *joint distribution function*  $F(z_1, \dots, z_n)$ . The key result, Sklar's theorem (Sklar, 1959), states that any joint distribution function can be written as

$$F(z_1, \dots, z_n) = C(F_1(z_1), \dots, F_n(z_n)). \quad (4.1)$$

The function  $C$  is called the copula and encodes the dependence between variables.

Copulas are distribution functions themselves. More precisely, if all variables are continuous,  $C$  is the joint distribution of the variables  $U_1 = F_1(Z_1), \dots, U_n = F_n(Z_n)$ . This helps the estimation and simulation from the model. To estimate the copula function  $C$ , we:

1. estimate marginal distributions  $\hat{F}_1, \dots, \hat{F}_n$ ,
2. construct *pseudo-observations*  $\hat{U}_1 = \hat{F}_1(Z_1), \dots, \hat{U}_n = \hat{F}_n(Z_n)$ , and
3. estimate  $C$  from the pseudo-observations.

Then, given estimated models  $\hat{C}$  and  $\hat{F}_1, \dots, \hat{F}_n$  for the copula and marginal distributions, we can generate synthetic data as follows.

1. Simulate random variables  $U_1, \dots, U_n$  from the estimated copula  $\hat{C}$ .
2. Define  $Z_1 = \hat{F}_1^{-1}(U_1), \dots, Z_n = \hat{F}_n^{-1}(U_n)$ .

### 4.2.3.2 Parametric Copula Families

In practice, it is common to only consider sub-families of copulas that are conveniently parameterized. There are a variety of such parametric copula families. Such families can be derived from existing models for multivariate distributions by inverting the equation of Sklar's theorem:

$$C(u_1, \dots, u_n) = F(F_1^{-1}(u_1), \dots, F_n^{-1}(u_n)). \quad (4.2)$$

For example, we can take  $F$  as the joint distribution function of a multivariate Gaussian and  $F_1, \dots, F_n$  as the corresponding marginal distributions. Then Equation 4.2 yields a model for the copula called the Gaussian copula, which is parameterized by a correlation matrix. The Gaussian copula model includes all possible dependence structure in a multivariate Gaussian distribution. The benefit comes from the fact that we can combine a given copula with any type of marginal distribution, not just the ones the copula was derived from. That way, we can build flexible models with arbitrary marginal distributions and Gaussian-like dependence. The same principle applies to other multivariate distributions and many copula models have been derived, most prominently the Student's  $t$  copula and Archimedean families. A comprehensive list can be found in Joe (2014).

### 4.2.3.3 Vine Copula Models

When there are more than two variables ( $n > 2$ ) the type of dependence structure these models can generate is rather limited. Gaussian and Student copulas only allow for symmetric dependencies between variables. Quite often, dependence is asymmetric, however. For example, dependence between  $Z_1$  and  $Z_2$  may be stronger when both variables take large values.

Many Archimedean families allow for such asymmetries but require all pairs of variables to have the same type and strength of dependence.

Vine copula models (Aas et al., 2009; Czado, 2019) are a popular solution to this issue. The idea is to build a large dependence model from only two-dimensional building blocks. We can explain this with a simple example with just three variables:  $Z_1, Z_2,$  and  $Z_3$ . We can model the dependence between  $Z_1$  and  $Z_2$  by a two-dimensional copula  $C_{1,2}$  and the dependence between  $Z_2$  and  $Z_3$  by another, possibly different, copula  $C_{2,3}$ . These two copulas already contain some information about the dependence between  $Z_1$  and  $Z_3$ , the part of the dependence that is induced by  $Z_2$ . The missing piece is the dependence between  $Z_1$  and  $Z_3$  after the effect of  $Z_2$  has been removed. Mathematically, this is the conditional dependence between  $Z_1$  and  $Z_3$  given  $Z_2$  and can be modelled by yet another two-dimensional copula  $C_{1,3|2}$ . The principle is easily extended to an arbitrary number of variables  $Z_1, \dots, Z_n$ . Algorithms for simulation and selection of the right conditioning order and parametric families for each (conditional) pair are given in Dißman et al. (2013).

Because all two-dimensional copulas can be specified independently, such models are extremely flexible and allow for highly heterogeneous dependence structures. Using parametric models for pairwise dependencies remains a limiting factor, however. If necessary, it is also possible to use nonparametric models for the two-dimensional building blocks. Here, the joint distribution of pseudo-observations  $(\hat{U}_1, \hat{U}_2)$  is estimated by a suitable kernel density estimator (see Nagler et al., 2017).

#### 4.2.3.4 Implementation

Here we use Synthia (Meyer & Nagler, 2021) version 0.3.0 (Meyer & Nagler, 2020) with pyvinecopulib (Nagler & Vatter, 2020) to fit three different copula types: Gaussian, vine-parametric, and vine-nonparametric. Vine-parametric fits a parametric model for each pair in the model from the catalogue of Gaussian, Student, Clayton, Gumbel, Frank, Joe, BB1, BB6, BB7, and BB8 copula families and their rotations (see Joe, 2014, for details on these families) using the Akaike information criterion (AIC). Vine-nonparametric uses transformation local quadratic likelihood fitting as explained in Nagler et al. (2017). Copulas are fitted to  $\mathbf{X}_{\text{train}}$  to generate synthetic training sets  $\mathbf{X}'_{\text{train}}$  using three augmentation factors (i.e., each containing either 1x, 5x, or 10x the number of profiles in  $\mathbf{X}_{\text{train}}$ ).  $\mathbf{X}_{\text{train}}$  plus  $\mathbf{X}'_{\text{train}}$  form augmented training sets containing 20 000 profiles (or double the amount of training data) for 1x augmentation factor and 60 000 and 110 000 profiles for 5x and 10x augmentation factors, respectively. As the generation of new profiles with copula models is random, the generation is also repeated 10 times for each case to allow meaningful statistics to be computed.

#### 4.2.4 Target Generation

Target generation is used to generate outputs from corresponding inputs using a physical model. Here, outputs are computed using a simple toy model based on Schwarzschild's equation (e.g., Petty, 2006) to estimate the downwelling longwave radiation under the assumption that atmospheric absorption does not vary with wavelength as

$$\frac{dL^\downarrow}{dz} = a(z)[B(z) - L^\downarrow], \quad (4.3)$$

where  $z$  is the geometric height,  $B$  is the Planck function at level  $z$  (i.e.,  $B = \sigma_{\text{SB}} T^4$ , where  $\sigma_{\text{SB}}$  is the Stefan-Boltzmann constant; giving the flux in  $\text{W m}^{-2}$  emitted from a horizontal black body surface), and  $a$  is the rate at which radiation is intercepted and/or emitted. A common approximation is to treat longwave radiation travelling at all angles as if it were all travelling with a zenith

angle of  $53^\circ$  (Elsasser, 1942): in this case  $a = D\beta_e$ , where  $\beta_e$  is the extinction coefficient of the medium, and  $D = 1/\cos(53) = 1.66$  is the diffusivity factor, which accounts for the fact that the effective path length of radiation passing through a layer of thickness  $\Delta z$  is on average  $1.66\Delta z$  due to the multiple different angles of propagation. In the context of ML,  $a(z)$  and  $B(z)$  are known and  $F(z)$  is to be predicted. Here we use the difference in two atmospheric pressures expressed in sigma coordinates ( $\Delta\sigma$ , where  $\sigma$  is the pressure  $p$  at a particular height divided by the surface pressure  $p_0$ ) instead of  $z$ . The layer optical depth  $\tau = \beta_e\Delta z$  is calculated from the total-column gas optical depth  $\tau_g$  and cloud layer optical depth  $\tau_c$  as  $\tau = \tau_c + \tau_g \Delta\sigma_i$ , since  $\Delta\sigma$  is the fraction of mass of the full atmospheric column in layer  $i$ . Then, as the downwelling flux at the top of the atmosphere is 0, the equation is discretized as follows assuming  $B$  and  $a$  are constant within a layer:

$$L_{i-1/2}^\downarrow = L_{i+1/2}^\downarrow (1 - \epsilon_i) + B_i \epsilon_i, \quad (4.4)$$

where  $B_i$  is the Planck function of layer  $i$ ,  $\epsilon_i = 1 - e^{-a_i\Delta z} = 1 - e^{-D\tau}$  is the emissivity of layer  $i$ ,  $L_{i+1/2}^\downarrow$  is the downwelling flux at the top of layer  $i$ , and  $L_{i-1/2}^\downarrow$  is the downwelling flux at the bottom of layer  $i$ . We compute  $L^\downarrow$  from  $T$ ,  $p$ , and  $\tau_c$  using the real NWP-SAF ( $\mathbf{X}_{\text{train}}$ ) or augmented ( $\mathbf{X}_{\text{train}}$  plus  $\mathbf{X}'_{\text{train}}$ ) data. To reduce, and thus simplify, the number of quantities used in the physical model and ML emulator (Section 4.2.5),  $\tau_c$  is pre-computed and used instead of vertical profiles of liquid and ice mixing ratios ( $q_l$  and  $q_i$ ) and effective radius ( $r_l$  and  $r_i$  in m) as  $\frac{3}{2} \frac{\Delta p}{g} \left( \frac{q_l}{\rho_l r_l} + \frac{q_i}{\rho_i r_i} \right)$ , where  $\rho_l$  is the density of liquid water ( $1000 \text{ kg m}^{-3}$ ),  $\rho_i$  is the density of ice ( $917 \text{ kg m}^{-3}$ ),  $g$  is the standard gravitational acceleration ( $9.81 \text{ m s}^{-2}$ ). For  $\tau_g$  we use a constant value of 1.7 determined by minimizing the absolute error between profiles computed with this simple model and the comprehensive atmospheric radiation scheme ecRad (Hogan & Bozzo, 2018).

## 4.2.5 Estimator Training

As the goal of this paper is to determine whether the use of synthetic data improves the prediction of ML emulators, here we implement a simple feed-

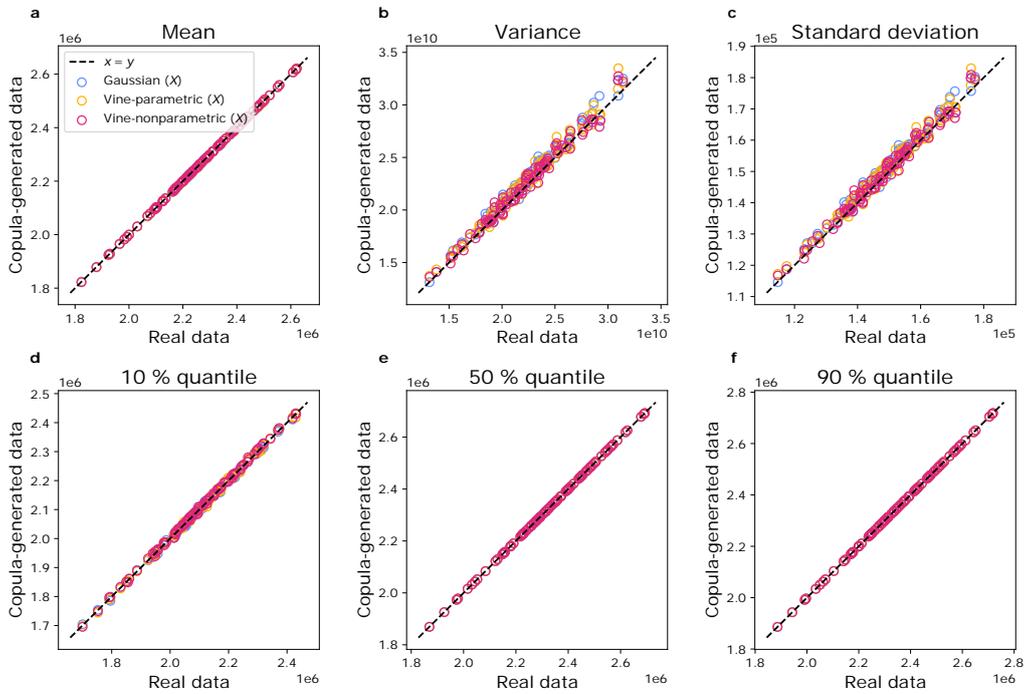
forward neural network (FNN). FNNs are one of the simplest and most common neural networks used in ML (Goodfellow et al., 2016) and have been previously used in similar weather and climate applications (e.g., Chevallier et al., 1998; Krasnopolsky et al., 2002). FNNs are composed of artificial neurons (conceptually derived from biological neurons) connected with each other; information moves forward from the input nodes through hidden nodes. The multilayer perceptron (MLP) is a type of FNN composed of at least three layers of nodes: an input layer, a hidden layer, and an output layer, with all but the input nodes using a nonlinear activation function.

Here we implement a simple MLP consisting of three hidden layers with 512 neurons each. This is implemented in TensorFlow (Abadi et al., 2016), and configured with the Exponential Linear Unit activation function, Adam optimiser, Huber loss, 1 000-epoch limit, and early stopping with patience of 25 epochs. The MLP is trained with profiles of dry-bulb air temperature (Figure 4.2a), atmospheric pressure (Figure 4.2b), and layer cloud optical depth (Figure 4.2c) as inputs and profiles of downwelling longwave radiation (Figure 4.2d) as outputs. Inputs are normalized and both inputs and outputs are flattened into two-dimensional matrices as described in Section 4.2.2. The baseline case (Figure 4.1a) uses 10 000 input profiles without data augmentation for training, and copula-based cases (Figure 4.1b) use either 20 000, 60 000, or 110 000 profiles. The validation dataset  $\mathbf{Y}_{\text{val}}$  of 5 000 profiles is used as input for the early stopping mechanism, while the test dataset  $\mathbf{Y}_{\text{test}}$  of 10 000 profiles is used to compute statistics (Section 4.3.2). Because of the stochastic nature of MLPs, training (and inference) is repeated 10 times for each case to allow meaningful statistics to be computed. Given that the generation of random profiles in the case of augmented datasets is also repeated 10 times (see Section 4.2.3.4), any case using data generation includes 100 iterations in total (i.e., for each data generation run, the estimator is trained 10 times).

## 4.3 Results

### 4.3.1 Copula

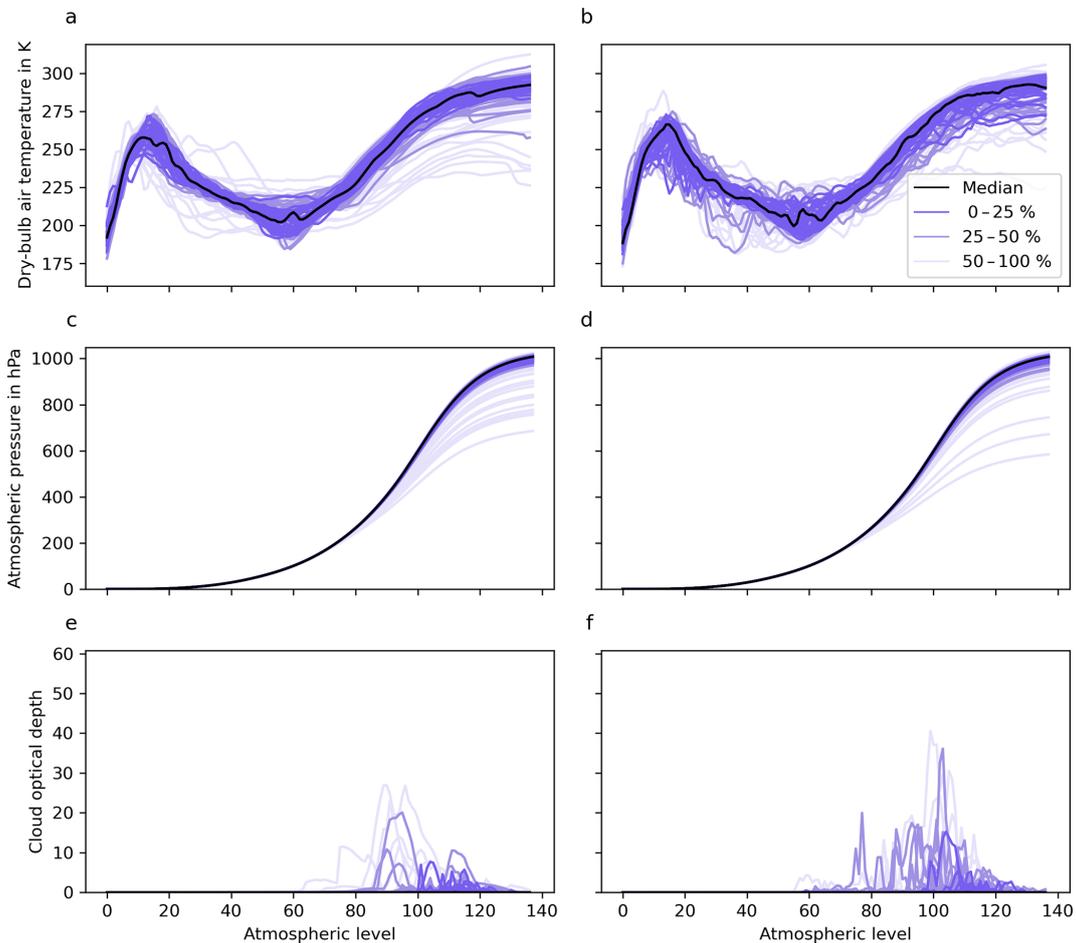
The quality of synthetic data is assessed in terms of summary statistics (e.g., Seitola et al., 2014) between the training  $\mathbf{X}_{\text{train}}$  and copula-simulated  $\mathbf{X}'_{\text{train}}$  datasets. For each copula type we compute a vector of summary statistics  $\mathbf{s}_i = f(\mathbf{p}_i)$  where  $f$  is the statistic function and  $\mathbf{p}_i = \mathbf{D}\mathbf{w}$ , with  $\mathbf{D}$  a matrix of flattened source or simulated data and  $\mathbf{w}$  a vector of random numbers for the  $i$ th iteration.



**Figure 4.3 | Statistical evaluation of synthetically generated data.** Summary statistics  $s_i$  from 100 iterations for (a) mean, (b) variance, (c) standard deviation, and (d) 10 %, (e) 50 %, and (f) 90 % quantiles. Each point corresponds to a statistic for a single iteration in arbitrary units. The x-axis represents the projection of real NWP-SAF  $\mathbf{X}_{\text{train}}$ , while the y-axis represents that of the copula-generated data  $\mathbf{X}'_{\text{train}}$ . Results are reported for Gaussian, vine-parametric, and vine-nonparametric copulas (see legend for keys).

Summary statistics are computed for mean, variance, and quantiles, iterating 100 times to allow meaningful statistics to be computed. As we consider random linear combinations of variables in source and copula-generated data,

we expect these summaries to coincide only if both marginal distributions and dependence between variables are captured. Figure 4.3 shows scatterplots of summary statistics  $\mathbf{s}_i$  for (a) mean, (b) variance, (c) standard deviation, and (d) 10 %, (e) 50 %, and (f) 90 % quantiles. Real NWP-SAF data are shown on the x-axis and copula generated data on the y-axis, with each point corresponding to a random projection as described earlier (100 points in total total). For a perfect copula model, we expect all points to fall on the main diagonal, where  $x = y$ . Figure 4.3 shows that for all copula models, synthetically generated data are close to the real data, with larger errors in variance and standard deviation. Qualitatively, we can evaluate copula-generated profiles in terms of their overall shape and smoothness across multiple levels, as well as range and



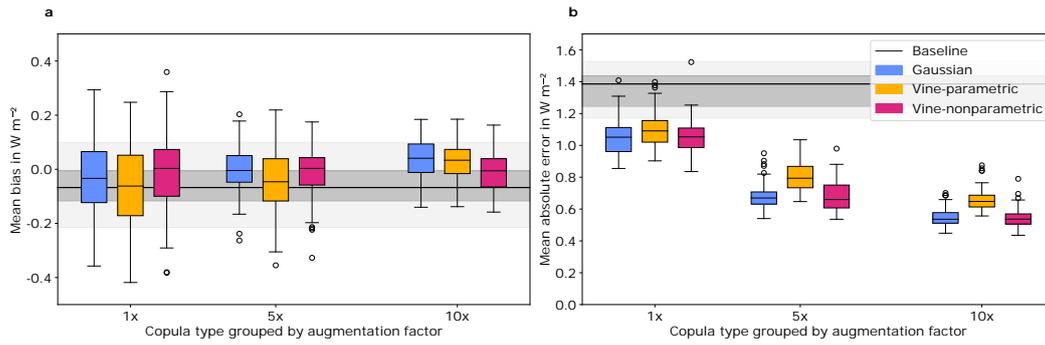
**Figure 4.4 | Qualitative evaluation of synthetically generated data.** Profiles of (a, c, e) real NWP-SAF and (b, d, f) Gaussian-copula-generated data for (a–b) dry-bulb air temperature, (c–d) atmospheric pressure, and (e–f) cloud optical depth. The median profile is shown in black, with a random selection of 90 profiles grouped in batches of 3 (i.e., each having 30 profiles) for the central 0%–25%, outer 25%–50%, and 50%–100% calculated with band depth statistics (López-Pintado and Romo, 2009).

density at each level. To this end we plot a side-by-side comparison of source (Figure 4.4, left panel) and Gaussian-copula-generated (Figure 4.4, right panel) profiles showing the median profile and random selection of 90 profiles grouped in batches of 3 (i.e., each having 30 profiles) for the central 0 %–25 %, outer 25 %–50 %, and 50 %–100 % quantiles calculated with band depth statistics (López-Pintado & Romo, 2009). Simulated profiles of dry-bulb air temperature (Figure 4.4b) appear less smooth than the real ones across levels (Figure 4.4a); however, both density and range are simulated well at each level. Simulated profiles of atmospheric pressure (Figure 4.4d) are simulated well: they are smooth across all levels with similar range and density (Figure 4.4c). The highly non-Gaussian and spiky profiles of cloud optical depth (Figure 4.4e) make qualitative comparisons difficult; however, simulated profiles (Figure 4.4f) have a similar range and density, with high density for low values, and most range between levels 80 and 120.

### 4.3.2 Machine Learning

To evaluate whether ML emulators trained on augmented datasets have lower prediction errors compared to the baseline, here we use the test dataset  $\mathbf{X}_{\text{test}}$  of 10 000 profiles defined in Section 4.2.2. Statistics are computed based on a vector of differences  $\mathbf{d}$  between the physically predicted baseline  $\mathbf{Y}_{\text{test}}$  and ML-emulated  $\mathbf{Y}'_{\text{test}}$  (i.e.,  $\mathbf{d} = \mathbf{Y}_{\text{test}} - \mathbf{Y}'_{\text{test}}$ ). From this, the mean bias (MB =  $\frac{1}{N} \sum_{i=1}^N d_i$ ) and mean absolute error (MAE =  $\frac{1}{N} \sum_{i=1}^N |d_i|$ ) for the set of  $N$  profiles are computed. Box plots of MB and MAE are shown in Figure 4.5. Summary MB and MAE for the ML emulator with the lowest MAE using an augmentation factor of 10x are reported in Table 4.2. A qualitative side-by-side comparison of baseline and ML-predicted profiles using Gaussian-copula-generated profiles with an augmentation factor of 10x is shown in Figure 4.6.

MBs (Figure 4.5a) across all copula types and augmentation factors are generally improved, with median MBs and respective spreads decreasing with larger augmentation factors. Overall, the Gaussian copula model performs



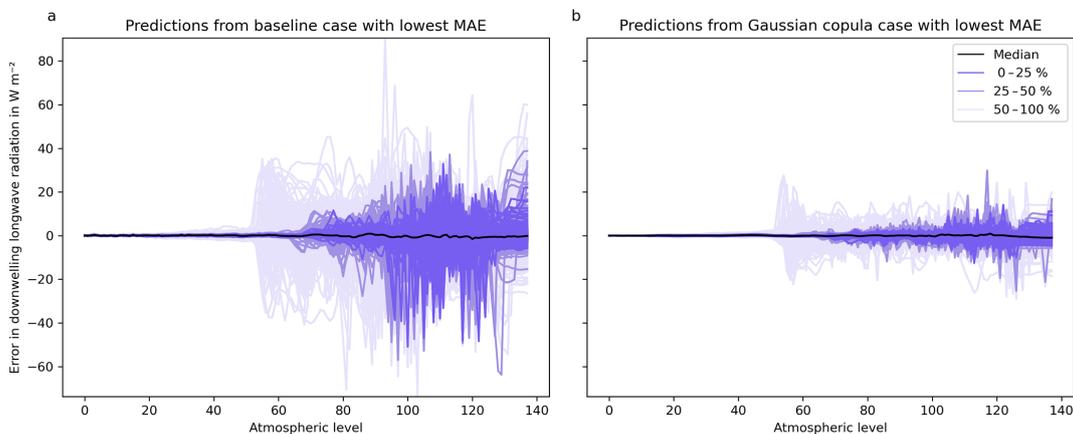
**Figure 4.5 | Errors for baseline and copula cases.** These are grouped by different copula types (Gaussian: blue, vine-parametric: yellow, vine-nonparametric: red) and augmentation factors (1x, 5x, 10x) for the mean bias (MB; **a**) and mean absolute error (MAE; **b**). The median for the baseline case is shown in black, and the range is shaded in grey.

better than vine-parametric and vine-nonparametric models. MAEs (Figure 4.5b) show a net improvement from the baseline across all copula models and augmentation factors. When using an augmentation factor of 1x (i.e., with double the amount of training data), the median MAE is reduced to approximately  $1.1 W m^{-2}$  from a baseline of approximately  $1.4 W m^{-2}$  and further reduced with increasing augmentation factors. In the best case, corresponding to an augmentation factor of 10x (i.e., with an additional 100 000 synthetic profiles), the copula and ML emulator combinations with the lowest MAE (Table 4.2) show that MBs are reduced from a baseline of  $0.08 W m^{-2}$  to  $-0.02$  and  $-0.05 W m^{-2}$  for Gaussian and vine-nonparametric, respectively, but increased to  $0.10 W m^{-2}$  for vine-parametric. MAEs are reduced from a baseline of  $1.17 W m^{-2}$  to  $0.45$ ,  $0.56$ , and  $0.44 W m^{-2}$  for Gaussian, vine-parametric, and vine-nonparametric copula types, respectively.

**Table 4.2 | Errors for baseline and copula cases.** Mean bias (MB) and mean absolute error (MAE) for baseline and copula cases. Statistics shown for the ML emulator combination with the lowest MAE. Baseline trained using 10 000 real NWP-SAF profiles. Copula cases were trained using 110 000 profiles (10 000 real and 100 000 synthetic), i.e., with an augmentation factor of 10x. Bold indicates the lowest error

Case	MB ( $W m^{-2}$ )	MAE ( $W m^{-2}$ )
Baseline	0.08	1.17
Gaussian	<b>-0.02</b>	0.45
Vine-parametric	0.10	0.56
Vine-nonparametric	-0.05	<b>0.44</b>

The ML training configuration with the lowest overall MB and MAE combination during inference corresponds to a Gaussian copula and augmentation factor of 10x (Table 4.2). Errors between the physically predicted  $\mathbf{Y}_{\text{test}}$  and ML-predicted  $\mathbf{Y}'_{\text{test}}$  are shown for the baseline (Figure 4.6a) and Gaussian copula case (Figure 4.6b). These are shown grouped by their central 0%–25%, outer 25%–50%, and 50%–100%. Qualitatively, most ML-generated profiles show improvements. The most central 25% profiles are within  $\pm 20 \text{ W m}^{-2}$  for the Gaussian copula case and about  $\pm 40 \text{ W m}^{-2}$  for the baseline case. Near-surface errors (levels 130-BOA) are reduced to approximately  $\pm 5 \text{ W m}^{-2}$  from approximately  $\pm 10 \text{ W m}^{-2}$ .



**Figure 4.6 | Qualitative evaluation of emulated profiles for baseline and copula cases.** Prediction errors for (a) baseline and (b) data-augmented emulator using 110 000 profiles (10x augmentation factor; Gaussian copula). The median (most central) profile is shown in black, and the most central 25%, outer 25%–50%, and 50%–100% profiles are computed using band depth statistics and shown in shades of blue.

## 4.4 Discussion and Conclusion

Results from a qualitative comparison of synthetically generated profiles (Figure 4.4) show that synthetic profiles tend to be less smooth and noisier than the real NWP-SAF. Nevertheless, a machine-learning evaluation shows that errors for emulators trained with augmented datasets are cut by up to 75

% for the mean bias (from 0.08 to -0.02 W m<sup>-2</sup>; Table 4.2) and by up to 62 % for the mean absolute error (from 1.17 to 0.44 W m<sup>-2</sup>; Table 4.2).

In this study, we show how copula-based models may be used to improve the prediction of ML emulators by generating augmented datasets that have statistically similar profiles in terms of their individual behaviour and dependence across variables (e.g., dry-bulb air temperature at a specific level and across several levels). Although the focus of this paper is to evaluate copula-based data generation models to improve predictions of ML emulators, we speculate that the same or similar methods of data generation have the potential to be used in several other ML-related applications, such as to (1) test architectures (e.g., instead of cross-validation, one may generate synthetic datasets of different size to test the effect of sample size on different ML architectures), (2) generate data for un-encountered conditions (e.g., for climate change scenarios by extending data ranges or relaxing marginal distributions), and (3) compress data (e.g., by storing reduced parameterized versions of the data if the number of samples is much larger than the number of features).

Although so far, we have only highlighted the main benefits of copula-based models, several limiting factors should also be considered. A key factor for very high-dimensional data is that both Gaussian and vine copula models scale quadratically in the number of features—in terms of both memory and computational complexity. This can be alleviated by imposing structural constraints on the model, for example using structured covariance matrix or truncating the vine after a fixed number of trees. However, this limits their flexibility and adds some arbitrariness to the modelling process. A second drawback compared to GANs is that the model architecture cannot be tailored to a specific problem, like images. For such cases, a preliminary data compression step as in Tagasovska et al. (2019) may be necessary.

As highlighted here, data augmentation for ML emulators may be of particular interest to scientists and practitioners looking to achieve a better generalization of their ML emulators (i.e., synthetic data may act as a regularizer to reduce overfitting; Shorten & Khoshgoftaar, 2019). Although a comprehensive analysis of prediction errors using different ML architectures is out of scope, our work is a first step towards further research in this area. Moreover, although we did not explore the generation of data for unencountered conditions (e.g., by extending the range of air temperature profiles while keeping a meaningful dependency across other quantities and levels), the use of copula-based synthetic data generation may prove useful to make emulators more resistant to outliers (e.g., in climate change scenario settings) and should be investigated in future research.

*In the following two chapters, we will look at the applications of ML emulators in urban land surface schemes. In the next chapter, I will show a method to couple a physically based urban land surface model to a weather model, used to compare the ML emulator developed in chapter 6.*

## CHAPTER 5

---

# A Reference Urban Land Surface Model<sup>13</sup>

*In chapter 2, we have seen that the surface energy balance used in urban land surface models includes different physical processes such as radiation, conduction, and convection. Thus, urban land surface models can be regarded as simple candidates for evaluating the emulation of all these three processes relevant to atmospheric modelling. Before moving to the emulation of urban land surface processes, however, in this chapter, I will first couple a mature urban energy balance to a mature weather research model so that it can be used as a baseline in chapter 6.*

---

<sup>13</sup> This chapter is based on Meyer, D., Schoetter, R., Riechert, M., Verrelle, A., Tewari, M., Dudhia, J., Masson, V., van Reeuwijk, M., & Grimmond, S. (2020). WRF-TEB: Implementation and Evaluation of the Coupled Weather Research and Forecasting (WRF) and Town Energy Balance (TEB) Model. *Journal of Advances in Modeling Earth Systems*, 12(8). <https://doi.org/10.1029/2019MS001961>.

# 5.1 Introduction

With increasing urbanization (United Nations, 2019) and climate change (Collins et al., 2013), the study of urban atmospheric phenomena such as the spatial variation of temperature (Arnfield, 2003), or the impact of the urban environment on moisture (Unger, 1999), precipitation (Liu & Niyogi, 2019; Shepherd, 2005), wind fields (Martilli, 2002; Moonen et al., 2012), boundary layer (Y. Chen et al., 2009; Lin et al., 2008; Masson, 2006), air conditioning (Salamanca et al., 2013, 2014; Takane et al., 2017), and heating energy demand (Santamouris et al., 2001), are increasingly relevant.

Coupled numerical weather prediction (NWP) and urban land surface models allow a diverse range of urban climate phenomena to be studied (e.g., Best, 2005; F. Chen et al., 2011; Hamdi et al., 2012). While NWP models simulate the prevailing meteorological conditions at kilometre resolution, land surface models (LSM) parameterize sub grid surface processes that are too small-scale, or (currently) too complex, to be explicitly modelled. Urban land surface models (ULSM) or urban canopy models (UCM), aim to capture the urban form (and sometimes function) created by buildings, roads, and vegetation.

UCMs, applicable to horizontal scales of the order 1–10 km, provide surface radiative and turbulent fluxes to NWPs. Parameters for the different aerodynamic (e.g., roughness length, drag force), radiative (e.g., albedo, emissivity), and thermal (e.g., heat capacity, conductivity) processes are required to capture radiative shading and trapping, large storage heat fluxes, and strongly varying turbulent heat fluxes. The energy balance may be solved for individual facets (e.g., roof, walls, roads) with different levels of complexity (Grimmond et al., 2010, 2011).

A coupled NWP-UCM may treat the surface as a single vertical layer (single layer UCM), where the entire urban canopy layer is collapsed into a single

point, or as multiple layers (multilayer UCM), where the UCM is “immersed” in the NWP to account for the interaction between bluff-bodies (e.g., buildings) and the atmosphere (F. Chen et al., 2011). The assumptions and simplifications can also vary from treating buildings as being arranged to create infinitely long street canyons (e.g., Kondo et al., 2005; Kusaka et al., 2001; Martilli et al., 2002; Masson, 2000), or as cuboids (e.g., Mills, 1997).

Complex models may not perform systematically better than simpler ones (Grimmond et al., 2010, 2011). However, simpler models tend to lack features, thus limiting the study of specific urban climate processes (e.g., estimation of building energy consumption or details of it such as air conditioners energy demand, energy production from solar photovoltaic panels), which may be of interest to the broad urban climate community.

Previous NWP-UCM coupling work implemented and evaluated the linkage between the single layer Town Energy Balance (TEB; Masson, 2000 and subsequent papers) and NWPs (e.g., Freitas et al., 2007; Hamdi et al., 2012; Lemonsu et al., 2009; Lemonsu & Masson, 2002; Rozoff et al., 2003) or the Weather Research and Forecasting (WRF; Skamarock et al., 2019) model and UCMs (e.g., Kikegawa et al., 2014; Martilli et al., 2002). However, to our knowledge, none have employed tests to verify the effects of coupling alone.

Here we both outline a technical approach to couple and verify model components and link TEB with WRF to add to other UCM options already available in WRF (UCAR, 2020). To date, these are: bulk urban parametrization within the Noah-LSM (F. Chen & Dudhia, 2001; Ek et al., 2003), single layer Urban Canopy Model (SLUCM; F. Chen et al., 2011), and the multi-layer Building Effect Parameterization (BEP; Martilli et al., 2002) with optional Building Energy Model (BEP-BEM; Salamanca et al., 2010). Given the conclusions in Grimmond et al. (2010, 2011)'s model comparison study, we do not expect WRF-TEB to perform systematically better than other models currently available in WRF

but we have undertaken this work to (a) offer researchers and practitioners a greater number of features currently unavailable in other models (Section 5.2.2), (b) simplify the evaluation of offline and online TEB in future research and comparison projects (Section 5.4.1), and (c) simplify the integration with future TEB developments (Section 5.4.1).

We describe the coupling between WRF and TEB (Section 5.2) both conceptually (Section 5.3) and technically (Section 5.4) in a way that may be generalizable beyond the scope of WRF (or WRF-TEB). We release the complete source code, data, and tools to make our results reproducible (Section 5.5) and evaluate the model (Section 5.6) with a technical integration test (Section 5.6.3) and meteorological observations (Section 5.6.4).

## **5.2 Models and Software**

### **5.2.1 Weather Research and Forecasting**

WRF is a popular atmospheric model used in research and NWP applications (Powers et al., 2017). It has been developed under two variants: the Advanced Research WRF (ARW; Skamarock et al., 2008, 2019), and the Nonhydrostatic Mesoscale Model (NMM; Janjic, 2003; Janjic et al., 2001). The support for the latter recently ended (see Developmental Testbed Center, 2018). Here, we exclusively refer to the WRF-ARW variant (as WRF).

WRF-TEB is developed using WRF-CMake version 4.1.5 (Riechert & Meyer, 2019b, 2020; Table 5.1) as it adds CMake (Kitware Inc., 2019a) support to the latest versions of WRF to simplify the configuration and build process of WRF and WPS (WRF Preprocessing System). Although, WRF-CMake version 4.1.5 does not include support for WRF-Chem (Grell et al., 2005), WRF-DA (Huang et al., 2009), WRFPLUS (Guerrette & Henze, 2015), or WRF-Hydro (Gochis et al., 2018), its benefits may outweigh these limitations to model developers, code

**Table 5.1 | Models and software used in the coupling.** “Model” refers to the science (i.e., as outlined in the literature), “Software” refers to the actual software and “Version” the exact software-version used in running a simulation.

Model	Reference	Software	Reference	Version	Reference
WRF	Skamarock et al. (2019)	WRF-CMake	Riechert & Meyer (2019)	4.1.5	Riechert & Meyer (2020)
TEB	Masson (2000)	TEB	Meyer et al. (2020)	4.0.1	Masson et al. (2020)

maintainers, and end-users wishing to build WRF, as it includes: robust incremental rebuilds, dependency analysis of Fortran code, flexible library dependency discovery, integrated support for shared (Open Multi-Processing; OpenMP) and distributed (Message Passing Interface; MPI) memory, support for automated testing using continuous integration (CI), and availability of experimental prebuilt binary releases for Linux, macOS, and Windows from the project's GitHub page or through the integration with GIS4WRF (Meyer & Riechert, 2019a), a QGIS (QGIS Development Team, 2019) toolkit for pre- and post-processing, visualizing, and running simulations in WRF. Here we refer to both the physical model and the software (i.e., WRF-CMake) as WRF, unless highlighting specific software features.

## 5.2.2 Town Energy Balance

The physically based single-layer UCM TEB (Masson, 2000) characterizes cities by their surface area of building roofs, walls, roads, and integrated vegetation using a simplified infinite street canyon geometry. The energy balance of impervious and pervious (vegetation) surfaces are calculated independently before being aggregated. To characterize the urban area, TEB requires a surface fraction of vegetation/garden, building, and road area, building height and vertical to horizontal surface ratio. For the calculation of shadowing effects and radiative trapping, the street orientation is assumed isotropic.

The outer surface of each facet is assumed to be sufficiently thin that the layer-averaged temperature can be used to determine the radiative and turbulent surface flux densities (i.e., the impervious skin temperature equals that of first layer-averaged temperatures). Thermal diffusion into materials is calculated

using the thermal properties and thickness of the specified layers. The momentum flux is calculated for the whole canopy using a representative roughness length of the city (at model grid point scale), whereas thermal and hydrological fluxes for impervious areas are computed using an aerodynamic resistances network that considers local energy exchange within and above the canyon. Turbulent exchanges inside the urban canyon, and those between the canyon and the atmosphere above, depend on an aerodynamic resistances network with exchange coefficients that depend on wind speed and stability conditions (see Figure 1 in Lemonsu et al., 2004). Other TEB original features include the following: a water reservoir on roofs and road, and a snow mantel on roofs and roads (Masson, 2000), but more recent TEB developments now also include the following:

1. Building Energy Model (BEM; Bueno et al., 2012): internal building energy balance (indoor air, floor, and internal mass), windows, heat-ventilation-air-conditioning (HVAC), infiltration, shading devices, and natural ventilation (opening of windows).
2. Road orientation (Lemonsu et al., 2012): specified road orientation, and separate energy balance for adjacent walls.
3. Gardens (Lemonsu et al., 2012): vegetation inside canyons.
4. Green roofs (de Munck et al., 2013).
5. Human behaviour related to building energy consumption (Schoetter et al., 2017).
6. Calculation of urban carbon dioxide fluxes (Goret et al., 2019).
7. Irrigation (de Munck et al., 2013): irrigation of green roofs, gardens, and watering of roads.
8. Solar panels (Masson et al., 2014) for hot water and/or photovoltaic (PV).

To implement WRF-TEB, TEB (Meyer, Schoetter, Masson, et al., 2020) version 4.0.1 (Masson et al., 2020; Table 5.1) is used as it includes MinimalDX (Meyer & Raustad, 2020) to improve the modelling of air conditioners (AC), and

Psychrolib (Meyer & Thevenard, 2019) to calculate psychrometric functions. Furthermore, support for Linux, macOS, and Windows with CMake allows direct integration in WRF-CMake. In TEB 4.0.1, Features 3 and 4 use a simplified vegetation scheme with a fixed albedo and Bowen ratio, whereas Lemonsu et al. (2012) treats the vegetation by coupling to ISBA (Interaction Soil Biosphere Atmosphere; Noilhan & Planton, 1989; see simplifications in Section 5.3). Furthermore, features 5 and 6 are not available in TEB 4.0.1.

## 5.3 Coupling Approach

TEB is coupled to WRF following the generalized coupling methodology described in Best et al. (2004), where atmospheric quantities from the NWP's lowest model level are passed to the LSM to improve the calculation of surface fluxes (Best et al., 2004). The current implementation of WRF-TEB is designed to work in both WRF (UCAR, 2020) and WRF-CMake (Riechert & Meyer, 2019b) alongside the Yonsei University (YSU) planetary boundary layer (PBL) scheme (Hong et al., 2006; Noh et al., 2003), the revised MM5 Monin-Obukhov surface layer scheme (Jiménez et al., 2012), and the Noah-LSM as they have been shown to perform reasonably in recent comparison studies for several types of environments (Greve et al., 2013; Hari Prasad et al., 2016; Hu et al., 2010, 2013; Shin & Dudhia, 2016; Xie et al., 2012).

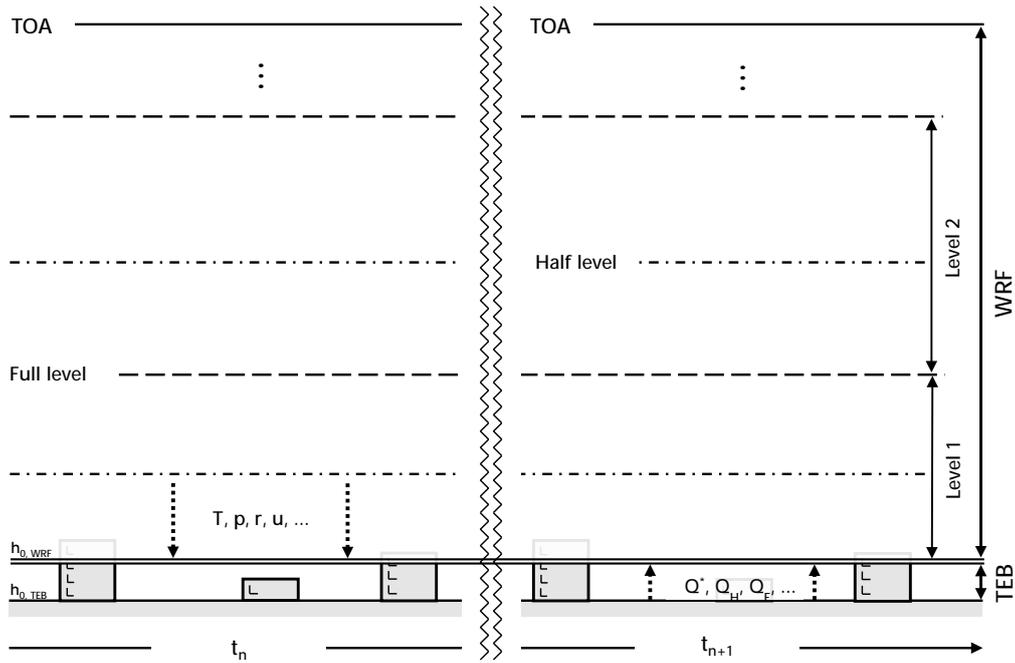
The general workflow used in WRF-TEB is as follows: (1) for each grid cell defined as urban, and each time step, WRF provides TEB with atmospheric and model-specific quantities from the lowest model level (Table 5.2a), and site-specific characteristics such as grid cell spatial coordinates and surface parameters (e.g., building height and roof albedo) from a lookup table in WRF (URBPARAM.TBL). (2) From these, TEB computes area-averaged surface quantities (Table 5.2b) and passes them to WRF (Figure 5.1). (3) Surface

diagnostics (Table 5.2c) are calculated in TEB and passed directly to WRF as outputs without affecting calculations in the WRF dynamical core.

As with other TEB features (Section 5.2.2), anthropogenic heat flux options in WRF-TEB are specified in the `URBPARM` file and modelled as follows: (a) traffic and industrial heat fluxes are user specified, (b) heating and air conditioning from buildings are a function of meteorological conditions, physical characteristics of the building envelope, internal heat release from electrical appliances, and heating/air conditioning set points. All options available in TEB (and therefore WRF-TEB) are given in the TEB section of the Surfex scientific documentation (Le Moigne et al., 2018).

Given the different nature of the two models, with atmospheric equations solved explicitly (WRF) and processes parameterized (TEB), several assumptions and simplifications are made:

In TEB, the urban canopy layer (UCL) is represented as a point. This results in a mismatch in elevation between TEB and WRF (i.e.,  $h_{0,TEB}$  and  $h_{0,WRF}$ ; Figure 5.1). As with other single-layer UCMs,  $h_{0,WRF}$  is located at the mean building height (given in `URBPARM.TBL`). This means that the urban canopy layer is located below the surface of the NWP model. This is an important assumption of any single-layer UCM and rarely explicitly stated. This assumption may be acceptable if buildings are low to midrise (e.g., < 20 m) and of uniform height, such as typical of extensive suburban areas. However, areas with tall and/or variable height roughness elements (e.g., skyscrapers), as in central business districts in many cities worldwide, may not suit the use of a single-layer UCM. Depending on the user's needs and model configuration, outputs from WRF-TEB may require postprocessing to account for this assumption.



**Figure 5.1 | Coupling approach.** For each urban grid cells, and at each time step ( $t_n$ ), atmospheric quantities from the first model level such as dry-bulb air temperature ( $T$ ), atmospheric pressure ( $p$ ), mass mixing ratio of water vapor ( $r$ ), and horizontal wind components ( $u, v$ ; Table 5.2a) are passed to TEB. In turn, TEB computes area-averaged surface quantities such as net all-wave radiation flux density ( $Q^*$ ), and turbulent sensible and latent heat flux density ( $Q_H$  and  $Q_E$ ; Table 5.2b) and passes them to WRF at the surface ( $t_{n+1}$ ). Under the single-layer UCM assumption, the vertical extent of the atmospheric model extend from  $h_{0,WRF}$  to the top of the atmosphere (TOA) whereas the UCM is assumed below the ground. This therefore creates a mismatch between the real ground level  $h_{0,TEB}$  and the ground level as seen by the atmospheric model  $h_{0,WRF}$ .

Surface diagnostics in WRF are given at standard World Meteorological Organization weather station heights (i.e., 10 m for wind, and 2 m for air temperature and humidity), thus representing quantities below the mean building height in the case of single layer UCMs. In WRF-TEB, we do not calculate these explicitly but simply rely on TEB's surface diagnostics (i.e., representative at half building height; see Masson, 2000). Given the single-layer assumption, WRF-TEB's surface diagnostics are, in effect, representative at half the building height below the first model level.

**Table 5.2 | Variables used in the coupling.** Their Outputs (O), Inputs (I) and their respective units are shown separately. The full list of parameters are reported in URBAPARM.TBL. <sup>a</sup>WRF atmospheric surface pressure is assumed to be the same as the atmospheric pressure at the roof level in TEB due to the coupling assumption (see  $h_{0,TEB}$  and  $h_{0,WRF}$ ; Figure 5.1). WRF-TEB assumes the atmospheric pressure to be the uniform throughout the canyon. <sup>b</sup> $Q^* = S^\downarrow - S^\uparrow + L^\downarrow - L^\uparrow$ , where  $S^\downarrow$  is the (total) downwelling shortwave radiation flux density calculated as  $S^\downarrow + S^\downarrow$ , and  $S^\uparrow$  and  $L^\uparrow$  is the upwelling shortwave and longwave radiation flux density.

Symbol	Name	Unit	WRF	TEB
<i>(a) WRF outputs, TEB inputs</i>				
$T$	Dry-bulb air temperature	K	O	I
$P_0$	Atmospheric pressure at the surface <sup>a</sup>	Pa	O	I
$P$	Atmospheric pressure at forcing level	Pa	O	I
$r$	Mass mixing ratio of water vapor	kg kg <sup>-1</sup>	O	I
$u$	Zonal component of wind velocity	m s <sup>-1</sup>	O	I
$v$	Meridional component of wind velocity	m s <sup>-1</sup>	O	I
$S^\downarrow$	Downwelling direct shortwave radiation flux density	W m <sup>-2</sup>	O	I
$S^\downarrow$	Downwelling diffuse shortwave radiation flux density	W m <sup>-2</sup>	O	I
$L^\downarrow$	Downwelling longwave radiation flux density	W m <sup>-2</sup>	O	I
RR	Rainfall rate	kg m <sup>-2</sup> s <sup>-1</sup>	O	I
SR	Snowfall rate	kg m <sup>-2</sup> s <sup>-1</sup>	O	I
$Z$	Solar zenith angle	rad	O	I
$\Delta t$	Time step	s	O	I
$t_{local}$	Current (local) time	s	O	I
<i>(b) WRF inputs, TEB outputs</i>				
$Q_H$	Turbulent sensible heat flux density	W m <sup>-2</sup>	I	O
$Q_E$	Turbulent latent heat flux density	W m <sup>-2</sup>	I	O
$E$	Evaporation mass flux density	kg m <sup>-2</sup> s <sup>-1</sup>	I	O
$Q_G$	Ground heat flux density	W m <sup>-2</sup>	I	O
$\alpha$	Surface albedo	1	I	O
$\varepsilon$	Surface emissivity	1	I	O
$T_s$	Surface (skin) temperature	K	I	O
$w_s$	Mass mixing ratio of water vapor	kg kg <sup>-1</sup>	I	O
$u_*$	Shear (friction) velocity	m s <sup>-1</sup>	I	O
<i>(c) Diagnostic outputs</i>				
$Q^*$	Net all-wave radiation flux density <sup>b</sup>	W m <sup>-2</sup>	O	O
$T_{canyon}$	Dry-bulb air temperature at half building height	K	O	O
$r_{canyon}$	Mass mixing ratio of water vapor at half building height	kg kg <sup>-1</sup>	O	O
$u_{canyon}$	Zonal component of wind velocity at half building height	m s <sup>-1</sup>	O	O
$v_{canyon}$	Meridional component of wind velocity at half building height	m s <sup>-1</sup>	O	O
$Q_{cooling}$	Buildings' power demand for cooling	W m <sup>-2</sup>	O	O
$Q_{heating}$	Buildings' power demand for heating	W m <sup>-2</sup>	O	O
$Q_{thermal}$	Thermal power production of solar panels on roofs	W m <sup>-2</sup>	O	O
$Q_{electric}$	Electric power production of solar panels on roofs	W m <sup>-2</sup>	O	O

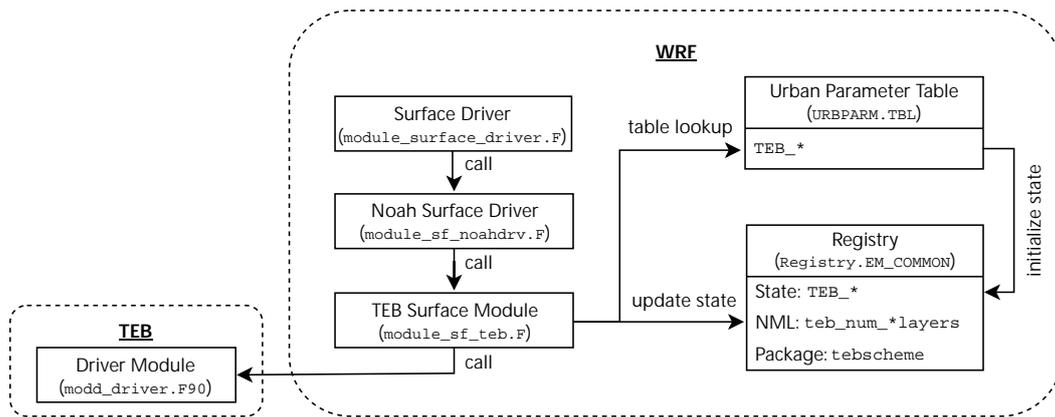
The treatment of vegetation in WRF-TEB can be done in three ways. Users have the option to define the fraction of vegetation as (a) integrated (i.e., vegetation inside canyons Lemonsu et al., 2012 or as part of green roofs de Munck et al., 2013), (b) non-integrated, or (c) both (i.e., integrated, and non-integrated). In (a), WRF-TEB uses a simple urban vegetation scheme with time-constant Bowen ratio (default 0.25) and albedo (default 0.15). These values can be modified (e.g., higher for drought conditions), but are constant during a model simulation. The temporal evolution of soil temperature, surface water storage or vegetation is not represented. The approach of Lemonsu et al. (2012) to

simulate in-canyon vegetation with the ISBA model is expected to be more universally applicable than the approach implemented in the current WRF-TEB. The default option in WRF-TEB (b) is to calculate the surface energy balance separately for urban and vegetated areas and to aggregate the fluxes. For the vegetated areas, the Noah-LSM cropland (MODIS class 12 in `VEGPARM.TBL`) class is assumed, whereas for built areas, TEB is used assuming no vegetation. In (c), users define the fraction of vegetation to model as integrated and non-integrated, respectively.

## 5.4 Software Implementation

The implementation of model to software and its testing are critical aspects of any model development. In WRF-TEB (Figure 5.2), the data flow and sequence of Fortran subroutine invocations is similar to existing couplings (e.g., SLUCM, BEM) and can be coupled serially as parallelization will be inherited from WRF over the number of grid cells in the domain(s). The main differences are in its code structure, implementation, and location. Although a generalizable framework (Common Community Physics Package; CCpp) is currently being developed (see Developmental Testbed Center, 2019), WRF does not yet provide a clear extension or plugin mechanism for integrating external models. Here we summarize the main techniques and issues encountered during its development that may aid other future meteorological model developments:

1. Modularity: As existing coupled models are tightly interwoven into WRF and partly modified, deviations from the original code can be hard to detect. By treating models as libraries we increase modularity.



**Figure 5.2 | Software coupling.** When TEB is enabled (`sf_urban_physics = 4` in `namelist.input`), the Noah Surface Driver calls the TEB subroutine in the TEB Surface Module. This calls the `TEB_DRIVER` subroutine in the Driver Module in the TEB library. The TEB driver is invoked for each urban grid cell at every time step. TEB parameters and initial conditions (IC; e.g., initial wall temperature) are read from a lookup table (`URBPARM.TBL`) and state variables are initialized with those IC at the start of the simulation. Conversions between units and dimensions are done in TEB Surface Module.

2. Clarity: As different couplings are typically in one large Fortran file or subroutine, understanding existing coupling is difficult. This is complicated further by reuse of some variables and parameters. By separating the coupling code, using consistent naming conventions, and introducing a small amount of duplication, we aim to increase clarity and reduce the time to undertake the coupling.
3. Reliability: WRF allows simulations to run even when required inputs are omitted, resulting in uninitialized values that may change results. By using a stricter input validation, we increase reliability of model output.

### 5.4.1 Modularity

Copying the coupled model's source code (e.g., TEB) directly into the WRF codebase is likely to cause code fragmentation, ultimately hindering model development, collaboration, and (possibly) the formation of a strong coherent community around that model.

Often, coupled models are not readily available in both offline and online versions (e.g., SLUCM, BEP, BEP-BEM, PBL), or are modified from their original (i.e., offline) version (e.g., Noah-LSM). For this reason it can be challenging to evaluate both versions.

To prevent this, we (a) keep TEB in its original repository on GitHub (Meyer, Schoetter, Masson, et al., 2020), (b) include TEB as a library dependency in WRF (similar to what is done with other external libraries such as `libnetcdf` (Unidata, 2019) or `libjasper` (Adams & Ward, 2004)), and (c) link the TEB library to WRF through a thin coupling interface (`module_sf_teb.F`; Figure 5.2). Then, to minimize complexity and overheads involved in publishing the TEB library in package managers (e.g., Ubuntu package repositories; (Canonical Ltd., 2019)), TEB is downloaded and compiled from source during the general WRF build process by relying on CMake's `ExternalProject` module (Kitware Inc., 2019b).

Once coupled in this way, any improvements, bug fixes, or other changes to the ULSM are inherited from the model source code repository (e.g., on GitHub) and included in WRF with a new commit (or version) identifier that is downloaded during the build process. This provides a central location for “issues,” maintains the community around the model, and facilitates the creation of offline tests (e.g., on units or individual components, or end-to-end integration tests of the whole model).

From this, a natural step is to make use of freely available Continuous Integration (CI) services to automate the execution of such tests on new commits on different operating systems and using multiple compiler versions and options, thus providing a stronger sense of reproducibility and trust. Where an appropriate testing methodology exists, CI can also enable model developers to accept code contributions from the community with more confidence.

## 5.4.2 Clarity

WRF/urban models (e.g., SLUCM, BEM/BEP) often reuse a subset of variables, parameters, and parts of the coupling “glue” code. Problems can arise as WRF allows each model to declare which state variables should be allocated in memory (via “schemes” in its registry). As the glue code for all models of one type is typically in a single large Fortran file (e.g., `module_sf_noahdrv.F`) with many conditional branches (e.g., `IF(SF_URBAN_PHYSICS == 1) THEN ...`) that is run irrespective of which model is activated, a model code contributor must understand most of the glue code to not introduce unintended memory issues. In the best case, using a variable that is not allocated leads to program crashes. In the worst case, it may lead to reading from, or writing to, other variables that are nearby in memory. In such case, the program may not fail but simply change results, possibly without the user being aware.

Here, we separate the configuration and coupling of TEB as much as possible from other models to improve clarity and reduce the time needed to understand the coupling. The prefix “TEB\_” is added to state variables, array dimension names, and parameters used by TEB, while as much glue code as possible is moved into the TEB coupling module (`module_sf_teb.F`; Figure 5.2). Without shared variables and parameters both the coupling and model can be understood in isolation and evolve independently from other models, hopefully encouraging community contributions. Despite duplicating state variables and parameters between urban models, we believe the benefits outweigh the disadvantages.

## 5.4.3 Reliability

WRF checks the validity of some, but not all, user inputs. Unfortunately, when an urban parameter is unspecified, unexpected simulation results can occur as uninitialized memory values are used instead of an error being raised. These errors may be hard to trace or can go undetected.

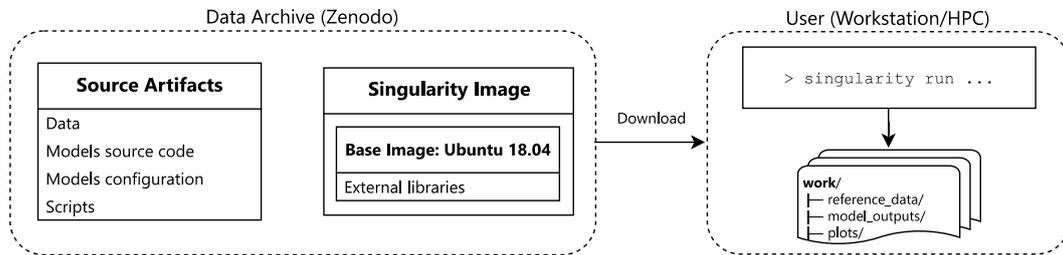
To solve this we separate parameter reading (in `module_sf_urban.F`) into three phases: (a) initialize all parameters with a known out-of-bounds value, (b) read the user-supplied parameters (from `URBPARM.TBL`), and (c) check that parameter values are not equal to the out-of-bounds value. Missing parameters cause an error message with the parameter name provided and stop WRF, thus saving time by early detection. Additional improvements could include checking value ranges for each parameter.

## 5.5 Scientific Reproducibility

Several issues with the lack of scientific reproducibility have been noted by several authors in various disciplines (Atmanspacher & Maasen, 2016; X. Chen et al., 2019; Cohen-Boulakia et al., 2017; Grüning et al., 2018; Redish et al., 2018; Sochat et al., 2017; Van Bavel et al., 2016). Although scientific reproducibility may be interpreted solely as the ability to reproduce a method and/or algorithm to reproduce the findings of the work, it can also be intended as the ability to re-run code on different platforms to obtain comparable results to those outlined in the research. Thus, to achieve the latter level of scientific reproducibility, several aspects must be considered. In the current context we identify: CPU (micro) architecture, operating system (OS), compiler vendor, compiler version, compiler options, external library versions, source and version of data sets, pre-processing and postprocessing steps of data, model version and configuration, and plotting routines.

One way to achieve a reasonable level of scientific reproducibility is to automate the generation of results included in this article by using a combination of Shell and Python scripts run through a Singularity container (Kurtzer et al., 2017). This approach remains architecture dependent but provides full control over OS (i.e., Ubuntu 18.04 in our case), compiler vendor (i.e., GNU in our case), source of data sets, processing steps for data, model

version and configuration, and plotting routines. Although this level of reproducibility may be deemed sufficient, the container would still rely on the Ubuntu package repository to install the compiler and external libraries



**Figure 5.3 | The scientific reproducibility approach taken in this paper.** The data archive (Meyer, 2020) contains all data and code used to evaluate the models. Users wishing to reproduce our results can download the data archive and run Singularity on their local or remote systems. As small differences in outputs may be possible because of the different hardware used, model outputs are also provided as reference.

without defining exact versions, or to download the “latest” (un-versioned) WPS high-resolution geographical data set (from the UCAR website). This can therefore lead to downloading newer compiler, library, or data set versions when re-building the Singularity image and ultimately alter results included in this paper.

Here, we achieve a reasonable level of scientific reproducibility by archiving all tools, data (including our results for reference, as equal results can only be guaranteed on the same hardware), and software together with a Singularity image containing OS and external libraries to Zenodo (Meyer, 2020; Figure 5.3). By doing this we remove the need for duplicating configuration settings in tables or appendices, thus reducing accidental errors and allowing reproducibility on local or high-performance computing (HPC) systems. Users wishing to reproduce the results described in this paper can download the data archive (Meyer, 2020) and run Singularity on their local or remote systems (Figure 5.3).

## 5.6 Model Evaluation

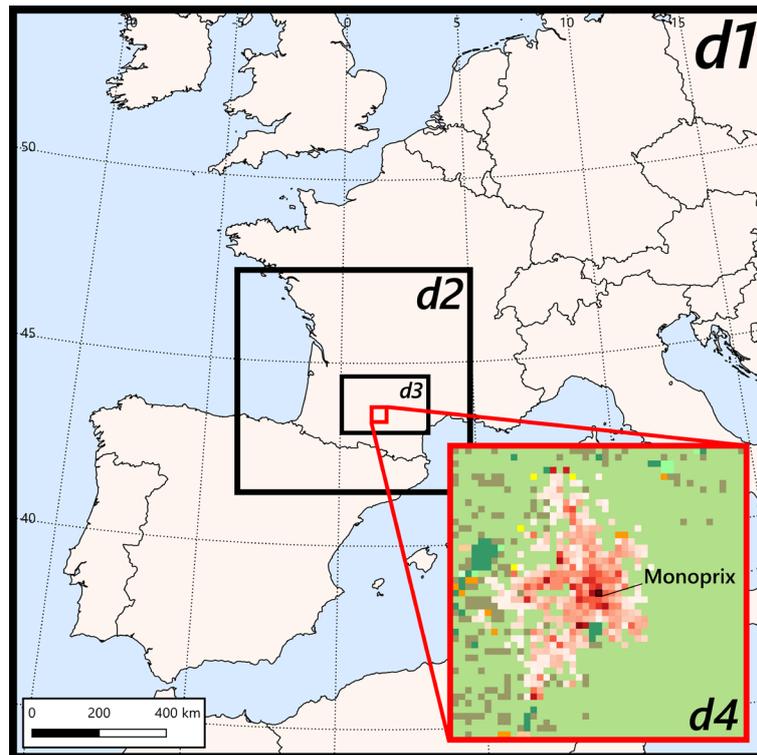
A fundamental aspect of any software development is testing. Although neither WRF nor TEB have been developed with testing in mind, in this section we outline tests for: the coupling (integration test, Section 5.6.3) and meteorological evaluation (Section 5.6.4). The former assesses the coupling technically, while the latter is used to explore the scientific benefit of the coupling. Both tests use similar model configurations (Section 5.6.2) with meteorological and geographical data for Toulouse, France (Section 5.6.1).

### 5.6.1 Site and Observational Data

Toulouse (Local Climate Zone 2; Hidalgo et al., 2019), the fourth largest city in France (475 438 inhabitants; INSEE, 2016), is located on a plain, 80 km north of the Pyrenees mountains (Figure 5.4). It has mild wet winters and dry hot summers (Joly et al., 2010). Central Toulouse (1°26'4"E, 43°36'15"N Monoprix; Figure 5.5) has homogeneous dense midrise buildings (see Figure 2 in Pigeon et al., 2007). Buildings (average height 15m) cover 55 % of the plan area, whereas vegetation covers 10 % (see Table 2 in Pigeon et al., 2007).

During the CAPITOUL (Canopy and Aerosol Particles Interactions in TOulouse Urban Layer; Masson et al., 2008) campaign, a Gill HS 50 sonic anemometer for eddy covariance (EC), and other meteorological sensors, were mounted on a tower at 48 m to 27 m a.g.l. (above ground level; depending on local wind conditions). EC sensors should be above the roughness sublayer to observe local-scale rather than microscale (e.g., individual urban obstacles) fluxes (Grimmond et al., 2004; Roth, 2000). The Goret et al. (2019) analysis of observed momentum fluxes confirms that they are located in the inertial sublayer (constant flux layer).

EC measurements (sampled at 50 Hz) postprocessing includes double-rotation (azimuth and pitch correction), recursive filtering according to McMillen (1988) with filter parameter set to 200 s, prior to 30 min covariance flux calculation (Pigeon et al., 2007). The EC flux footprint, calculated using the Kljun et al. (2015) model for each 30 min interval, identifies that the probable mean 80 % fetch extends to around 500 m in all wind directions, except for southerlies



**Figure 5.4 | The four WRF nested domains (d1–d4) used in online simulations.** These are shown on a base map from Natural Earth (2019). The innermost domain (d4, 1 km horizontal grid spacing) uses WPS MODIS 30 arc-sec land cover/use (UCAR, 2019). Grid cells for the Garonne river are replaced with urban land use and 1 km interpolated MApUCE urban fraction (Bocher et al., 2018; see Section 6.2). Higher urban fraction shown with darker shades of red. Manually assigned urban fraction (0.15) shown in yellow. See namelists in `configs/wrf/capitoul` in Meyer (2020) for the complete list of options used.

where it extends to 1 km (Goret et al., 2019). Given the homogeneous characteristics within 500 m radius of the tower and areas further south, we assume that the observed turbulent fluxes are comparable to the modelled turbulent fluxes (horizontal grid spacing 1 km; Figure 5.4, d4).

Radiation fluxes observed with a Kipp and Zonen CNR1 radiometer (sample rate 0.1 Hz) mounted at the tower top are averaged to 1 min (used to force the

offline TEB) and to 30 min (used in the meteorological evaluation). Air temperature and relative humidity measured with a Vaisala HMP233 Thermo-hygrometer (sample rate 0.1 Hz) at 43.3, 34.2, and 25 m a.g.l. when the mast was in the high, medium, and lowest position (respectively) are used as 1- and 30-min averages. The atmospheric pressure measured with a Vaisala PTB220 class A barometer (sample rate 0.1 Hz) at 20 m a.g.l., is used as 1 min average.



**Figure 5.5 | View of Toulouse downtown roofs from the terrace of the central site.** *Credit: V. Masson.*

Missing data are gap-filled every 1 min using observations from the routine observation station at Toulouse-Blagnac airport (7 km northwest of the tower) or a station operating at the site of Météo-France (flat grassland 6.5 km west southwest of the tower). The temperature and relative humidity values measured at these stations (2 m a.g.l.) are corrected by the average daily cycle of the differences between the values measured at the mast and at these sites. Wind speed measured at these stations (10 m a.g.l.) is corrected to the height of the mast assuming a logarithmic wind profile and neutral stratification. The values of aerodynamical roughness length and displacement height are 1.5

and 10.5 m, respectively. The evaluation is undertaken between 2 and 5 July 2004 when the tower was at 48 m a.g.l. (28 m above roof height). A strong urban canopy layer heat island was present before sunrise on 4 July 2004 (Hidalgo et al., 2008). An offline TEB simulation is forced with the required meteorological data and the surface morphological parameters averaged for the area within in a 500 m radius from the tower (see Figure 1 in Goret et al., 2019 and data in Meyer, 2020).

## 5.6.2 Model Setup

Four nested domains (Figure 5.4) are set up using GIS4WRF version 0.14.2 (Meyer & Riechert, 2019a, 2019b) with the innermost domain (Figure 5.4, d4) centred on the EC tower. The grid spacing is set to 1km for the horizontal and to 66 m (increasing with height and in pressure ( $\eta$ ) level equivalent) for the vertical, thus allowing equal comparison with observations (i.e., under single layer UCM assumptions (Section 5.3). The 48 m a.g.l. EC tower is represented at 33 m a.g.l., because the vertical extent of the buildings (mean building height 15 m) is not represented in WRF (see single-layer UCM assumptions in Section 5.3). The WPS MODIS land use (UCAR, 2019) and WRF urban fraction for the innermost domain are modified using GIS4WRF: (a) lake (MODIS class 21) used to indicate the River Garonne within central Toulouse is reassigned to cropland (MODIS class 12) as the river would otherwise be over represented in the 1 km grid (the river is  $\approx$  200 m wide), and (b) the urban fraction (i.e., constant for the whole grid for pixels defined urban; MODIS class 13, i.e., built) is replaced with that from the MApUCE (100 m resolution) data (Bocher et al., 2018) linearly interpolated to 1 km to provide spatial variability within the domain. Grid cells classified as urban in the MODIS 30 arcsec data set but not present in MApUCE (Figure 5.4, d4 yellow) are given a default fraction of 0.15 to be representative of the area. For the initial and boundary conditions we used ECMWF Cycle 28r2 analysis (ECMWF, 2004) with native horizontal grid spacing TL511 ( $\approx$  40km), gridded to 0.36 arc-degree.

**Table 5.3 | General WPS/WRF configuration settings used in integration test and meteorological evaluation.** Integration test values deviating from the common shared configuration are enclosed in parenthesis. The model time step is indicated for each domain (d1–d4). Time Standard (TS), Coordinated Universal Time (UTC), Lambert Conformal Conic (LCC). European Centre for Medium-Range Weather Forecasts (ECMWF). †Vertical grid spacing increasing with height ( $h$ ) and first level (L1) set to 66 m a.g.l. See namelists in `configs/wrf/capitoul` in Meyer (2020) for the complete list of options used

Option	Value	TS/Unit	Reference
<u>a) General</u>			
Start Time	1 July 2004 00:00	UTC	-
End Time	5 July 2004 08:18	UTC	-
Spin-up	(0)1	day	-
Timestep length	(180)108, (60)36, 12, 4	s	-
<u>b) Grid</u>			
Map Projection	LCC	-	-
Horizontal Spacing	27, 9, 3, 1	km	-
Vertical Spacing	$f(h)$ with L1 = 66 <sup>†</sup>	m	-
Vertical Levels	61	-	-
Nests and Grid Ratio	(2)4 and 1:3	-	-
Nesting Approach	1-way	-	-
Urban Classes	1	-	-
<u>c) Initial and Boundary Conditions</u>			
Data Set Name	ECMWF Cycle 28r2 analysis	-	ECMWF (2004)
Horizontal Spacing	TL511 ( $\approx$ 40 km)	-	-
Vertical Levels	61	-	-
Time Interval	6	h	-
<u>d) Physical Parametrization</u>			
Shortwave Radiation	Dudhia	-	Dudhia (1989)
Longwave Radiation	RRTM	-	Mlawer et al. (1997)
Microphysics	Single-moment 3-class	-	Hong et al. (2004)
PBL	YSU	-	Hong et al. (2006)
Surface layer	Revised MM5	-	Jiménez et al. (2012)
LSM	Noah-LSM	-	Chen & Dudhia (2001)
ULSM	TEB version 4.0.1	-	Meyer et al. (2020) and Masson et al. (2020)

**Table 5.4 | TEB configuration differences between the integration test and the meteorological evaluation.** Bowen ratio (B), Noah-LSM (N). See namelists in `configs/teb` in Meyer (2020) for the complete list of options used.

<b>Option</b>	<b>Integration test</b>	<b>Meteorological evaluation</b>
Vegetation Model	B	B/N
Air Conditioning	On	Off
Heating	On	Off
Green Roofs	On	Off
Solar Panels	On	Off

Domains are generated using WPS-CMake version 4.1.0 (Riechert & Meyer, 2019c). WPS/WRF are configured with parameters from Table 5.3 (see namelists in `configs/wrf/capitoul` in Meyer (2020) for the complete configuration). TEB (offline and online) surface characteristics are derived and adapted from Lemonsu et al. (2004) and Schoetter et al. (2017). To reduce the computational and storage cost during the integration test, we only run the outermost two domains with a longer simulation timestep (Table 5.3). Other differences in configuration settings between integration test and meteorological evaluation are reported in Table 5.4.

### 5.6.3 Integration Test

The model coupling is validated using an integration (i.e., end-to-end) test. Although neither WRF nor TEB carry out unit tests on their components, we assume that each on their own is working correctly. In this test, variables passed in the coupling (Table 5.2a) and TEB specific output variables (Tables 5.2b and 5.2c) are compared between offline (i.e., TEB) and online (i.e., WRF-TEB) models (Figure 5.6). As the inputs required to force TEB are not provided as standard WRF outputs, we introduce new variables in WRF's registry. To avoid permanently allocating memory for the additional 11 variables (Table 5.2) a new registry package (`teb_test`) is enabled through the namelist configuration option `teb_integration_test=1`. Thus, only the necessary variables are allocated without performance overheads when tests are not conducted, and further testing can be performed when new software releases

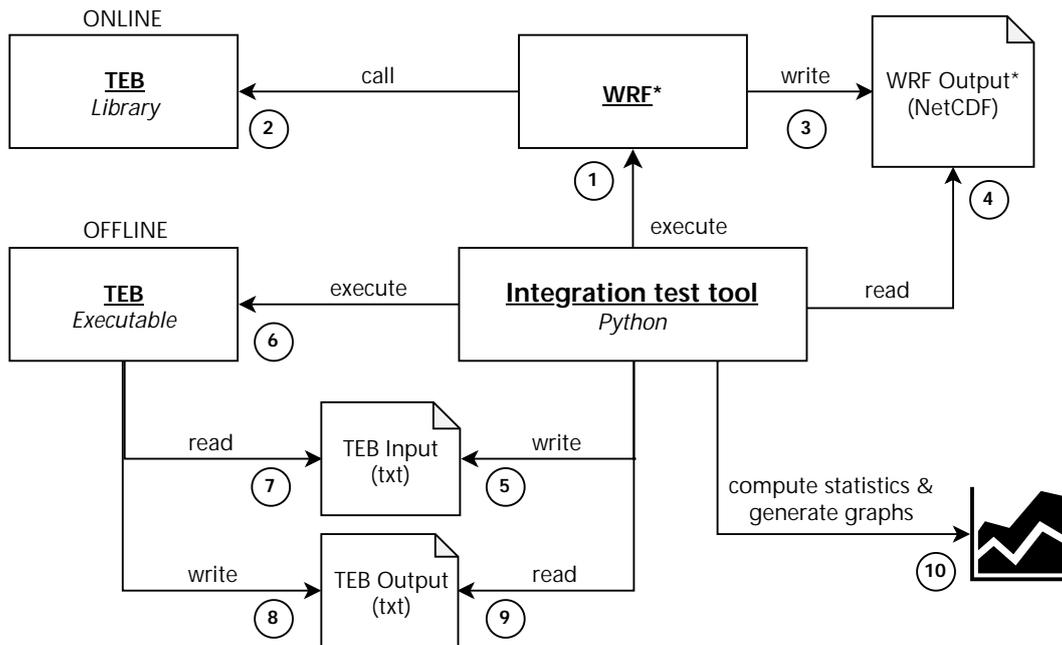
are available. Any difference larger than machine precision is attributed to coupling implementation errors. Results are evaluated graphically and statistically (Appendix B).

The implemented test detects errors that cause incorrect

1. loading of parameters from the urban parameter table file;
2. passing of parameters to TEB;
3. conversion of date/time from WRF to TEB conventions;
4. passing of geographical and date/time coordinates to TEB;
5. passing of TEB-internal state variables;
6. storing of TEB-specific output diagnostics;
7. updating of WRF state variables from TEB outputs;
8. grid cell looping; and
9. activation of TEB based on grid cell vegetation type and global scheme selector.

These errors can lead to software crashes or nonidentical results between TEB offline and WRF-TEB. With offline TEB forced with data from the WRF dynamical core quantities converted prior to use in TEB are not assessed. For example, as WRF uses mixing ratio whereas TEB uses specific humidity, it must be converted for TEB. Conversion errors would propagate in TEB and be evident in this comparison. Meteorological evaluation is undertaken separately (Section 5.6.4).

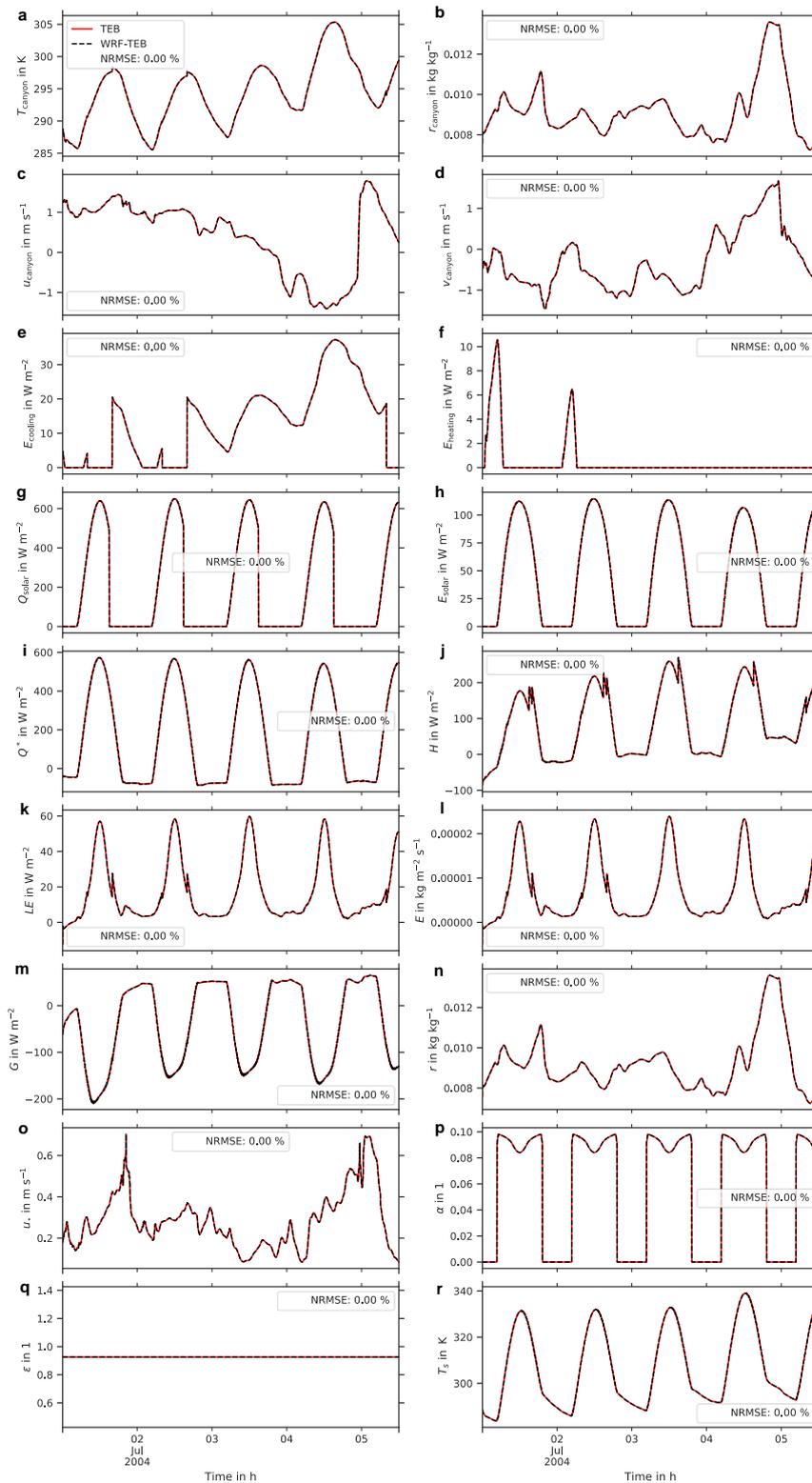
The regular WRF output are directly comparable to offline TEB outputs, as the relevant WRF quantities are not modified further. Modifications are prevented by setting the test grid urban fraction to 100 %. As this may be inappropriate for other submodels, the same method as used to obtain TEB forcing could be used. However, it may not be easy to test if the output quantities are correctly passed without using other techniques.



**Figure 5.6 | Integration test workflow.** (1) The integration test tool runs the integration test case in WRF, (2) the TEB library is called from WRF, and (3) WRF writes outputs (including TEB inputs) to NetCDF files. At the end of the WRF simulation, (4) TEB inputs are read and (5) TEB inputs are generated (6) for offline execution. (7) TEB reads the inputs, (8) simulates the same case, and stores outputs as text files. (9) Outputs are read and (10) compared to the WRF outputs using statistics and graphs. Asterisk (\*) indicates that WRF is run with `teb_integration_test=1` (see Section 5.6.3).

This testing approach (Figure 5.6) requires a common calling method of the offline model (TEB). As TEB can be compiled both as a library for online use and as an executable for offline use (Figure 5.6), the integration test tool (Figure 5.6) can run both TEB and WRF-TEB with the same source code. Thus, there is a strict testing of the coupling. The CCPP effort (Developmental Testbed Center, 2019) aims to organize models (/schemes) in a central location independent of a target framework. This may solve similar issues in the future. Unfortunately, it is not (yet) ready for use within WRF, or as an offline model and coupling testing tool.

Although a single configuration cannot represent all the degrees of freedom defined by the different model options and input parameters, we activate as many TEB options as possible (Table 5.4). Results show no visible differences between TEB and WRF-TEB (Figure 5.7). Similarly, no errors are detected using statistical metrics (NRMSE = 0 %, Figure 5.7).



**Figure 5.7 | Results from the integration test used to verify the coupling.** Line graph and normalized root-mean-square error (NRMSE) calculated from offline TEB and coupled WRF-TEB outputs for (a) dry-bulb air temperature, (b) mass mixing ratio of water vapor, and (c) zonal and (d) meridional component of wind velocity at half building height (a-d), buildings' power demand for (e) cooling and (f) heating, (g) thermal and (h) electric power production of solar panels on roofs, (i) net all-wave radiation, (j) turbulent sensible, and latent (k) flux density, (l) evaporation mass and (m) ground heat flux density, (n) mass mixing ratio of water vapor, and surface (o) shear (friction) velocity, (p) albedo, (q) emissivity, and (r) skin temperature

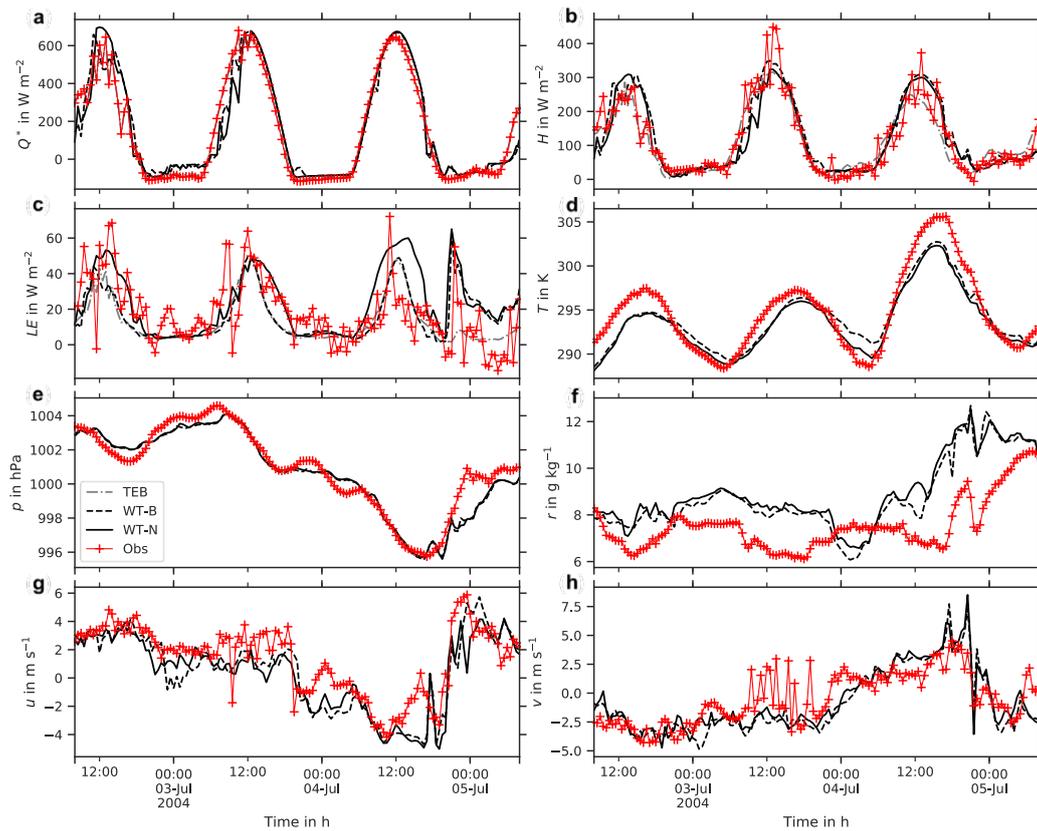
## 5.6.4 Meteorological Evaluation

With the coupling code verified (Section 5.6.3), a meteorological evaluation allows the scientific benefit of the coupling to be explored. Given the wide range of WRF options (Powers et al., 2017) the individual choices (e.g., radiation, microphysics, and boundary layer) may have a larger impact than the UCM selected. Here, the evaluation is focused on TEB.

For the evaluation period (2–5 July 2004) the net all-wave radiation flux density ( $Q^*$ ) is simulated well by TEB but only moderately well by WRF-TEB (Figure 5.8a; Table 5.5). Unsurprisingly, TEB forced with observations has a lower mean absolute error (MAE) for  $Q^*$  (MAE  $\approx 7.7 \text{ W m}^{-2}$ ; Figure 5.8a) than when forced with quantities simulated by WRF (MAE  $\approx 63 \text{ W m}^{-2}$ ; Figure 5.8a; Table 5.5). This difference is most likely caused by the cloud microphysics scheme, which simulates too much cloud overnight (2 and 3 July) and the next morning (3 July). This leads to an overestimation of  $Q^*$  during the night and an underestimation during the morning by WRF-TEB (Figure 5.8a). Turbulent sensible and latent heat flux densities ( $Q_H$  and  $Q_E$ ) are captured reasonably well by both TEB and WRF-TEB (MAE  $\approx 34$  and  $27 \text{ W m}^{-2}$  for  $Q_H$  and MAE  $\approx 13$  and  $11 \text{ W m}^{-2}$  for  $Q_E$  respectively; Figures 5.8b and 5.8c; Table 5.5).

**Table 5.5 | Meteorological evaluation of half-hourly values for 2–5 July 2004.** WRF-TEB with Noah-LSM (WT-N), WRF-TEB with Bowen ratio (WT-B), offline TEB (TEB), and arithmetic mean ( $\bar{x}$ ) of observation (OBS) values. Appendix B defines statistics. N = 145

Quantity	Unit	MAE			MBE			$\bar{x}$ OBS
		WT-N	WT-B	TEB	WT-N	WT-B	TEB	
$Q^*$	$\text{W m}^{-2}$	62.6	62.5	7.7	7.9	11.6	1.0	154.5
$Q_H$	$\text{W m}^{-2}$	34.9	34.1	27.7	-2.7	7.9	-4.9	120.0
$Q_E$	$\text{W m}^{-2}$	13.4	13.1	11.3	5.7	-1.1	-4.6	18.0
$T$	K	1.5	1.6	—	-1.2	-0.7	—	294.7
$p$	hPa	0.5	0.5	—	-0.2	-0.3	—	1000.9
$r$	$\text{g kg}^{-1}$	1.5	1.4	—	1.4	1.2	—	7.5
$u$	$\text{m s}^{-1}$	1.3	1.4	—	-1.0	-0.9	—	1.3
$v$	$\text{m s}^{-1}$	1.4	1.5	—	-0.2	-0.4	—	-0.5



**Figure 5.8 | Observations (Obs) and simulations results from the different models.** The results are shown separately for WRF-TEB with Noah-LSM (WT-N), WRF-TEB with Bowen ratio (WT-B), offline TEB (TEB) for (a) net all-wave radiation flux density, (b) turbulent sensible heat flux density, and (c) turbulent latent heat flux density. Results between WT-N, WT-B, and Obs are (d) dry-bulb air temperature, (e) atmospheric pressure, (f) mass mixing ratio of water vapor, (g) zonal component of wind velocity, and (h) meridional component of wind velocity. Observations from the Monoprix tower measured at (a-c, g, and h) 48 m a.g.l., (d, f) 43.3 m a.g.l., and (e) 20 m a.g.l. Atmospheric pressure corrected for height by linearly interpolating the pressure between the surface and the first model level to the measurement height at roof level because of the main assumption used in single-layer UCMs (see Section 5.3). All other quantities are uncorrected as changes would be minimal.

Dry-bulb air temperature ( $T$ ) at 48m a.g.l. (28m above roof level) is generally underestimated by WRF-TEB (mean bias error (MBE)  $\approx -1.2$  K; Figure 5.8d; Table 5.5) for the whole period and during the day, but slightly overestimated at night. Such underestimation requires further investigation but may be caused by other WRF processes (e.g., too low advected temperature). Mass mixing ratio of water vapor ( $r$ ) is overestimated (MBE  $\approx 1.4$  g  $\text{kg}^{-1}$ ; Figure 5.8f). Pressure ( $p$ ) and wind components ( $u$ ,  $v$ ) are simulated reasonably well (MAE  $\approx 0.5$  hPa and  $1.4$  m  $\text{s}^{-1}$ , respectively; Figures 5.8e, 5.8g, and 5.8h; Table 5.5), indicating that WRF has captured the general atmospheric dynamics.

Overall, the choice of vegetation scheme used (i.e., WRF-TEB with Noah-LSM (WT-N) or with Bowen ratio (WT-B)) results in similar simulation performance (Table 5.5, Figure 5.8).

## 5.7 Concluding Remarks

The coupled WRF-TEB model enables a wide range of urban climate processes to be analysed. In this paper, we describe techniques to help with the coupling approach, implementation, verification, and scientific reproducibility.

In implementing the coupling interface, we do not alter the current WRF framework but, instead, implement techniques to help with software modularity, clarity, and reliability, for example, treating TEB as an external library. We assess the software linkage with an integration test to ensure that the coupling is technically correct. The results of the integration test show no detectable differences with the offline TEB. The meteorological evaluation is used to confirm that the results are physically reasonable; these generally show reasonable agreement with net all-wave radiation and turbulent heat flux densities and other near-surface observations. Although improvements of surface fluxes and near-surface meteorological quantities may possibly be gained from using alternative parameters or parametrization schemes (e.g., microphysics, radiation) when configuring WRF, the interplay of these make attribution difficult. Furthermore, errors may arise from differences between observational source area (e.g., eddy covariance) and model grid length; parameter specification and uncertainties (e.g., from lack of availability, difficulty of “measurement” and theoretical understanding). This highlights the importance of undertaking both integration tests and meteorological evaluations.

Scientific reproducibility is addressed by providing model source code, configurations, data, and scripts with a Singularity image deposited on Zenodo. The coupled WRF-TEB model has been integrated into WRF and WRF-CMake and released as a free, open-source software on GitHub at <https://github.com/teb-model/wrf-teb>.

We encourage future versions of WRF to include the implementation of a flexible number (i.e., beyond three) of urban classes to allow for a greater heterogeneity of urban form and function to be represented.

*In the next chapter, I will use TEB and WRF-TEB to evaluate a machine learning emulator of urban land surface processes.*

## CHAPTER 6

---

# A Faster, More Accurate Urban Land Surface Model<sup>14</sup>

*Continuing from chapter 5, I conclude the research chapters by applying ML to offline and online urban land surface emulation. This investigation allows us to explore the emulation of radiation, as investigated in chapter 3, but with two other physical processes typical of parametrization schemes: conduction and convection.*

---

<sup>14</sup> This chapter is based on Meyer, D., Grimmond, S., Dueben, P., Hogan, R., & van Reeuwijk, M. (2022). Machine Learning Emulation of Urban Land Surface Processes. *Journal of Advances in Modeling Earth Systems*, 14(3). <https://doi.org/10.1029/2021MS002744>

# 6.1 Introduction

Land surface models (LSM) parameterize the energy exchange between the surface and the atmosphere, providing the lower boundary conditions (e.g., radiative and turbulent heat fluxes) to atmospheric models (Stensrud, 2007). For urban areas, ULSMs (urban LSMs) are currently employed in some operational numerical weather prediction (e.g., Bengtsson et al., 2017; Seity et al., 2011) and global climate models (e.g., Hertwig et al., 2021; Oleson et al., 2011) at the higher spatial resolution end but there is a growing need for broader adoption as they are fundamental to the delivery of integrated urban services (Baklanov et al., 2018; Grimmond et al., 2020). The complexity of ULSMs varies from simple assumptions (e.g., characterizing an impervious slab) to models that consider the 3D geometry of buildings with varying heights and material characteristics (Grimmond et al., 2009, 2010). This higher complexity, however, often comes at the cost of a greater number of site-specific input parameters and increased computational cost, which does not necessarily translate into improved results (Grimmond et al., 2011).

In recent years, machine learning (ML) techniques have shown potential in several areas of meteorology (e.g., Bolton & Zanna, 2019; Krasnopolsky et al., 2013; Nowack et al., 2018; Rasp et al., 2018; Rasp & Lerch, 2018). A key limitation of these techniques, however, is the need for large amounts of training data which, in urban meteorology, are often scarce.

One alternative to this is the creation of ML emulators (i.e., statistical surrogates of their physical counterparts) to improve the computational performance for a trade-off in accuracy (Meyer et al., 2021). Although emulators seek to improve the computational performance of current physical parametrizations, they offer no improvement in accuracy as surrogate models are, at best, as good as the data they are trained on. In urban land surface modelling, speed is, however, not such a limitation; unlike processes such as

radiative transfer where the fundamental processes are well understood but computational cost is the primary limiting factor (Meyer, Hogan, et al., 2022), most current ULSMs are reasonably fast but require several input parameters. Furthermore, previous comparisons (Grimmond et al., 2010, 2011) found that no individual ULSM is best at predicting all the main surface fluxes such as short- and longwave radiation, and turbulent sensible and latent heat fluxes. Although an obvious solution to this issue may be mitigated by running an ensemble of ULSMs coupled to a weather (or climate) model and use it to improve predictions, this is technically challenging to implement and hard to defend given the multi-fold increase in the computational cost resulting from running multiple ULSMs at once. Moreover, given the complexity of ULSMs, their availability, and the number of specific parameters needed to make realistic simulations, ULSMs often need a specialized team of people while an ML emulator may learn the behaviour of an ensemble mean and be a cheaper and easier alternative to run.

Here, we seek to develop an emulator of urban land surface processes and evaluate whether the strengths of multiple ULSMs can be combined to improve both accuracy and computational performance. The specific goals of this paper are:

1. To develop an ML emulator of urban land surface processes trained on the outputs of several ULSMs
2. To evaluate the emulator's accuracy and computational performance
3. To couple the developed ML emulator to a numerical weather model and to evaluate its accuracy and stability

To our knowledge, this is the first attempt to emulate a ULSM. In the following sections, we introduce the general problem of urban land surface modelling with details about data and methods used to develop the emulator (Section 6.2) and analyse the results (Section 6.3) before concluding with a summary and ideas for further work (Section 6.4).

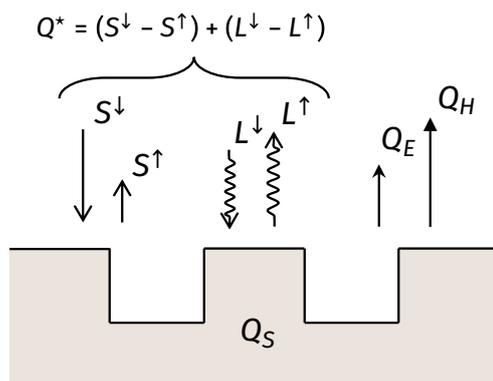
## 6.2 Methods

### 6.2.1 General Problem

At the core of ULSMs is the concept of surface energy balance (SEB), a general statement of energy conservation with applications to surfaces and volumes of all temporal scales (Oke et al., 2017). Physically, it describes the heating (or cooling) of a surface (Figure 6.1). Mathematically, it can be stated as:

$$\frac{dQ_s}{dt} = Q^* - Q_H - Q_E, \quad (6.1)$$

where  $dQ_s/dt$  is rate of change in thermal energy stored in a surface by conduction with  $Q_s$  the heat storage;  $Q^* = (S^\downarrow - S^\uparrow) + (L^\downarrow - L^\uparrow)$  is the surface net all-wave radiation flux density from downwelling ( $\downarrow$ ) and upwelling ( $\uparrow$ ) shortwave ( $S$ ) and longwave ( $L$ ) radiation; the convective heat flux densities are  $Q_H$  the turbulent sensible and  $Q_E$  the turbulent latent (or evaporative). Anthropogenic heat fluxes, the additional energy fluxes associated with human activities, if not simulated or prescribed, may be assumed to be zero or minimal in ULSMs (e.g., in low-density residential areas). The horizontal advection of heat and moisture is generally ignored or parameterized by

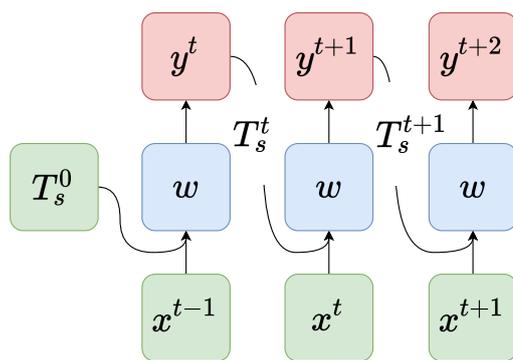


**Figure 6.1 | Conceptual sketch of surface energy balance exchanges (ignoring advection, vegetation, and anthropogenic heat flux) for a clear-sky day.** The arrows show the direction of fluxes relative to the surface. The shortwave flux is zero at night as the Sun's rays are below the horizon. The heat transfer processes are (left to right) radiative and convective. Other terms are defined in the text and Table 6.1.

ULSMs but implicitly included when coupled to weather models. ULSMs generally solve a prognostic equation in the form of Equation 1 to predict the evolution of upwelling short- and longwave radiation flux density, sensible and latent heat flux density, forced by downwelling short- and longwave radiation flux density, air temperature and humidity, atmospheric pressure, wind speed and direction and liquid (or solid) precipitation.

## 6.2.2 Urban Neural Network

The urban neural network (UNN) developed here is based on the multilayer perceptron (MLP; Bishop, 2006; Goodfellow et al., 2016), one of the simplest types of neural networks (NNs). The MLP-based UNN (Figure 6.2) is set up to predict upwelling short- and longwave radiation flux density and turbulent sensible and latent heat flux density (Table 6.1b) at time  $t + 1$  from inputs at time  $t$  of common meteorological variables such as dry-bulb air temperature and humidity, as well as the cosine of solar zenith angle  $\mu_0$  and model timestep length  $\Delta t$  (Table 6.1a). Both  $\mu_0$  and  $\Delta t$  are used to allow the UNN to run over different grid points at different spatial and temporal resolution when coupled to the weather model (Section 6.2.4). Specifically,  $\mu_0$  is used instead of latitude, longitude, and local time to reduce the number of features required by the UNN, and  $\Delta t$  to dynamically vary the timestep length, matching that used by the weather model. The surface temperature  $T_s$  is used rather than the upwelling longwave radiation  $L^\uparrow$  to mimic a physical system whereby  $T_s$  is used as a state between different timesteps and thus provide the initial condition at each new inference timestep (Figure 6.2). The UNN is implemented in TensorFlow (Abadi et al., 2015) version 2.6.2 (TensorFlow Developers, 2021c) and configured with two hidden layers, each having 256 neurons, rectified



**Figure 6.2 | Conceptual sketch of the urban neural network (UNN).** The trained UNN is used to predict outputs  $\mathbf{y}$  (Table 6.1b) at time  $t + 1$  from inputs  $\mathbf{x}$  (Table 6.1a) at time  $t$  using trained weights  $\mathbf{w}$ . The surface temperature  $T_s$  from a previous timestep provides the initial condition at each new inference.

linear unit (ReLU) activation function, and Adam optimiser (Kingma & Ba, 2015) with mean squared error as its optimisation function. This configuration is deemed optimal after conducting a hyperparameter optimisation of several configurations (Table S.1) and visually inspecting the results.

**Table 6.1 | Inputs and outputs from data sets and models used to conduct the simulations.** Multi-model Ensemble Mean (MEM). Depending on the model or data set used, data may be unavailable/not-applicable (-), available/outputted (✓), derived (D), constant (C), unmodified (U), state (S). <sup>†</sup>Additional inputs are given in namelists. <sup>a</sup>TEB requires direct and diffuse; these are computed using pvlib (Holmgren et al., 2018) version 0.9.0 (Holmgren et al., 2021) but given directly as-is in WRF. <sup>b</sup>Wind speed and direction are used instead of, and derived from the zonal and meridional components of wind velocity using MetPy version 1.1.0 (May et al., 2021). For computations between upwelling longwave radiation flux density  $L^\uparrow$  and surface temperature  $T_s$  a constant emissivity ( $\epsilon$ ) of 0.97 as reported in Coutts et al. (2007b) is used. The latent heat of vaporization  $\mathcal{L}_v$  is assumed to be constant with a value of 2.464 MJ kg<sup>-1</sup> which is applicable to 15 °C (Oke et al., 2017).

Symbol	Name	Unit	MEM	TEB <sup>†</sup>	WRF-TEB <sup>†</sup>	UNN	WRF-UNN	Derived as
<i>(a) Inputs</i>								
$T$	Dry-bulb air temperature	K	✓	✓	✓	✓	✓	-
$q$	Specific humidity	kg kg <sup>-1</sup>	✓	✓	✓	✓	✓	-
$p$	Atmospheric surface pressure	Pa	✓	✓	✓	✓	✓	-
$S^\downarrow$	Downwelling shortwave radiation flux density	W m <sup>-2</sup>	✓	✓ <sup>a</sup>	✓ <sup>a</sup>	✓	✓	-
$L^\downarrow$	Downwelling longwave radiation flux density	W m <sup>-2</sup>	✓	✓	✓	✓	✓	-
$u$	Zonal component of wind velocity	m s <sup>-1</sup>	✓ <sup>b</sup>	✓ <sup>b</sup>	✓	✓	✓	-
$v$	Meridional component of wind velocity	m s <sup>-1</sup>	✓ <sup>b</sup>	✓ <sup>b</sup>	✓	✓	✓	-
RR	Rainfall rate	kg m <sup>-2</sup> s <sup>-1</sup>	✓	✓	✓	✓	✓	-
$t_{local}$	Local time	s	✓	✓	✓	-	-	-
$\varphi$	Latitude	deg	✓	✓	✓	-	-	-
$\lambda$	Longitude	deg	✓	✓	✓	-	-	-
$\mu_0$	Cosine of solar zenith angle	rad	-	-	-	✓ <sup>a</sup>	✓	-
$\Delta t$	Timestep length	s	✓	✓	✓	✓	✓	-
<i>(b) Outputs</i>								
$S^\uparrow$	Upwelling shortwave radiation flux density	W m <sup>-2</sup>	✓	✓	-	✓	✓	-
$L^\uparrow$	Upwelling longwave radiation flux density	W m <sup>-2</sup>	✓	✓	-	-	D	$\epsilon\sigma T_s^4$
$T_s$	Surface (skin) temperature	K	-	✓	✓	S	S	$[L^\uparrow / (\epsilon\sigma)]^{1/4}$
$Q_H$	Turbulent sensible heat flux density	W m <sup>-2</sup>	✓	✓	✓	✓	✓	-
$Q_E$	Turbulent latent heat flux density	W m <sup>-2</sup>	✓	✓	✓	✓	✓	-
$E$	Evaporation mass flux density	kg m <sup>-2</sup> s <sup>-1</sup>	-	✓	✓	-	D	$Q_E / \mathcal{L}_v$
$Q_s$	Heat Storage	J m <sup>-2</sup>	-	✓	✓	-	D	Equation 6.1
$\alpha$	Surface albedo	1	C	C	C	-	D	$S^\uparrow / S^\downarrow$
$\epsilon$	Surface emissivity	1	C	C	C	-	-	-
$w_s$	Mass mixing ratio of water vapor	kg kg <sup>-1</sup>	-	✓	✓	-	U	-
$u_*$	Shear (friction) velocity	m s <sup>-1</sup>	-	✓	✓	-	U	-

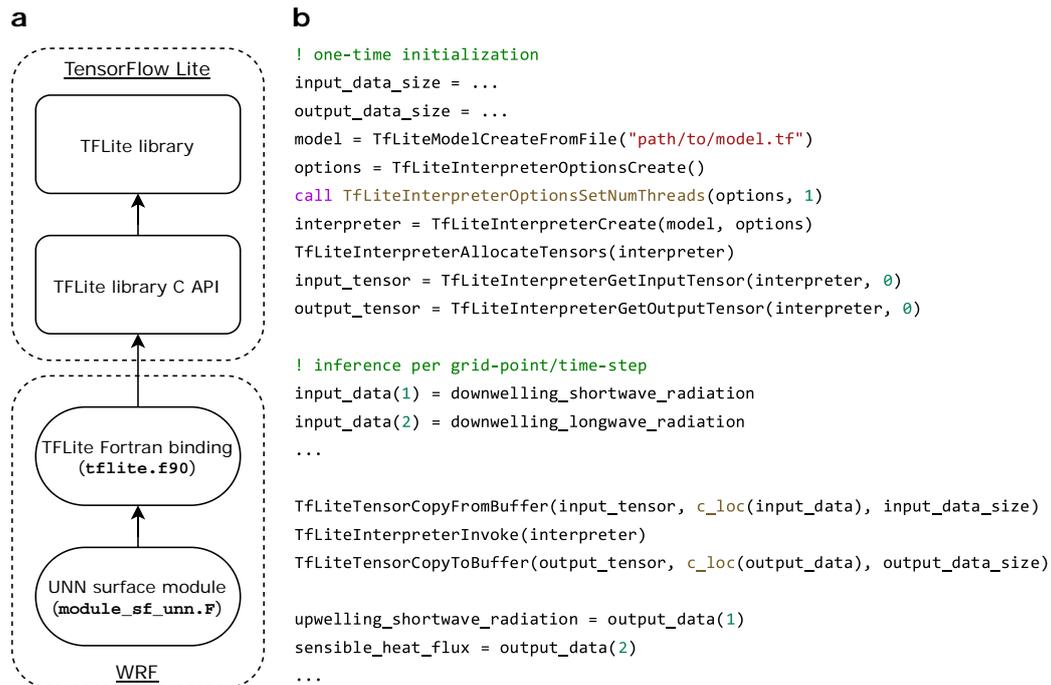
## 6.2.3 Town Energy Balance

To compare the UNN to a baseline, here we use the Town Energy Balance (TEB; Masson, 2000) model, a single-layer ULSM characterizing city areas based on building roofs, walls, roads, and vegetation, and assuming buildings create an infinite street canyon (Masson, 2000). TEB is chosen as it is a mature, widely used ULSM, extensively evaluated (e.g., Lemonsu et al., 2004; Masson et al., 2002; Pigeon et al., 2008), and available both offline (Meyer, Schoetter, Masson, et al., 2020) and online (i.e., coupled to weather models; e.g., Hamdi et al., 2012; Lemonsu & Masson, 2002; Meyer, Schoetter, Riechert, et al., 2020). Here we use the TEB software (Meyer, Schoetter, Masson, et al., 2020) version 4.1.2 (Masson et al., 2021) and refer to it as TEB. Similar to the UNN, TEB inputs

include typical meteorological variables at time  $t$  such as dry-bulb air temperature and humidity (Table 6.1a) to predict common surface energy balance variables at time  $t + 1$  (Table 6.1b).

## 6.2.4 Weather Research and Forecasting (WRF) Coupling

Both TEB and the UNN are coupled to the weather model WRF (Weather Research and Forecasting; Skamarock et al., 2019) in WRF-CMake (Riechert & Meyer, 2019b) version 4.2.2 (Riechert & Meyer, 2021) as it simplifies WRF-related development, configuration, and build-processes. Implementation details of WRF-TEB are provided in Meyer, Schoetter, Riechert, et al. (2020). Variables used in the WRF-UNN and WRF-TEB coupling are similar; however, as friction velocity  $u_*$  data are not provided in either observations or most ULSMs (Figure S.1), it is not part of the UNN (Section 6.2.2) and thus ignored in WRF-UNN. Porting the UNN to WRF is made seamless by relying on the available C application programming interface (API) provided with TensorFlow (TensorFlow Developers, 2021a). We choose the lightweight version of TensorFlow, TensorFlow Lite (TensorFlow Developers, 2021b) as: (a) it has CMake support, which makes the integration in WRF-CMake straightforward and allows sharing of project build options in WRF-CMake, and (b) it has a very succinct and accessible API (compared to TensorFlow library C API), that makes the Fortran binding used in the coupling easy to write. To perform the actual coupling, the UNN is exported to a TensorFlow Lite file from Python. To enable the UNN in WRF, the TFLite Fortran binding (`tf_lite.f90`; Figure 6.3a) and the UNN surface module (`module_sf_unn.F`; Figure 6.3) are written. The former is used to interface with the TFLite C API and the latter to initialize inputs (Table 6.1a) and run the UNN to generate outputs (Table 6.1b) which are passed to WRF for the next timestep.



**Figure 6.3 | Call to TensorFlow Lite (TFLite) C API from WRF as implemented for this study.** (a) TFLite integration: the Fortran interface module `tflite.f90` is used in WRF to bind to the TFLite C API, which is used by the UNN surface module `module_sf_unn.F` to initialize and run the neural network. (b) Example of TFLite Fortran binding (simplified from UNN surface module) use: first, a one-time initialization to load the TFLite model, configure its settings, and allocate input and output tensors is performed. Before each inference, input quantities (e.g., downwelling shortwave radiation) are stored in the input tensor in the expected order. Similarly, inference outputs are returned as output tensors from which individual quantities are accessed. Other pre- and post-processing steps such as normalization and error handling are omitted. See `models/wrf-unn/phys/module_sf_unn.F` & `models/wrf-unn/phys/module_sf_unn.F` in Meyer (2021) for the actual code.

## 6.2.5 Data and Model Setup

Grimmond et al. (2011)'s urban comparison study evaluated the accuracy of 32 ULSMs (or different configurations) from a wide range of international modelling groups (Table S.2) using directly observed fluxes in Preston, a suburban area of Melbourne (Australia). The 32 ULSMs vary in complexity and approach to characterising urban areas. These (Grimmond et al., 2010, 2011) include treating the surface as a single and homogenous surface slab (e.g. Noah land surface model; Chen et al., 2004), to resolving the surface energy balances for roofs, walls and ground between buildings assuming an infinite street canyon using in a single layer (TEB; Masson, 2000) or with multiple layers (Building effect parameterization—BEP; Martilli et al., 2002) or even resolving interactions in three dimensions (e.g. temperatures of urban facets—TUF—3D;

Krayenhoff & Voogt, 2007). Site information to configure the ULSMs were released in four Stages each giving more details (Table S.3): Stage 1—forcing data and minimal site information; Stage 2—basic surface cover fractions; Stage 3—urban morphology; and Stage 4—of urban materials characteristics. The main data sets, with local time stamps (UTC+10, i.e., 10-h ahead of Coordinated Universal Time), are:

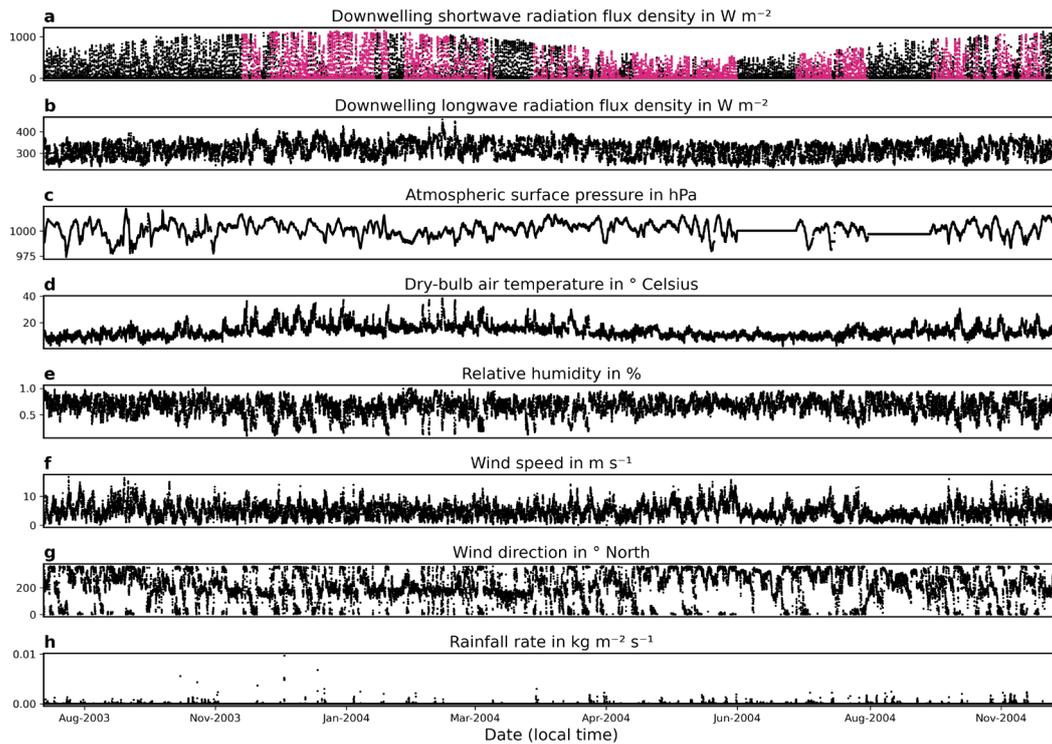
1. *Morphological parameters* (MP): provided in the four Stages, characterizing the surface around the observation site (Grimmond et al., 2011 Table 2, and our Table S.3).
2. *Meteorological Forcing* (MF; Grimmond et al., 2021): continuous gap-filled 30-min averages with a period ending timestamp (i.e., 10:30 is 10:01–10:30) of meteorological variables at 40 m a.g.l. between 12 August 2003 13:30 and 28 November 2004 23:00 (Figure 6.4; Table 6.1a).
3. *Multi-model output* (MO; Grimmond et al., 2013): continuous outputs, from 32 models or model-configurations between 13 August 2003 00:00 and 27 November 2004 23:30 reported in four separate Stages for upwelling short- and longwave radiation flux density, and turbulent sensible and latent heat flux density (Table 6.1b).
4. *Observations* (OBS; Grimmond et al., 2021): 30-min average fluxes with period ending timestamp measured at 40 m a.g.l. between 13 August 2003 00:00 and 27 November 2004 23:30. Methods to obtain observed fluxes are given in in Coutts et al. (2007a, 2007b). This data set has the same fluxes as the MO data set but with observational gaps (~39 % of MO; purple Figure 6.4a).

Here, a visual inspection of the MO data set for all Stages (Figure S.2) is used to remove ULSMs not simulating the turbulent latent heat flux density (8 ULSMs) or showing outliers (2 ULSMs), leaving 22 ULSMs (Figure S.3). The MO data set is used to compute the ensemble mean; hereafter referred to as the multi-model ensemble mean (MEM). To make evaluations consistent between MEM, UNN, and TEB, these are conducted for periods spanning 13 August 2003 00:00 and 27 November 2004 23:30 as defined in MEM totalling 22 704 30-min samples. For any given data set, the test fraction is the periods with OBS available (i.e., 39 % of MEM; number of samples  $N = 8\,866$ ; purple, Figure 6.4a) and the training fraction (used by the UNN; see point 2 below) for the remaining periods (i.e., 61 % of MEM;  $N = 13\,838$ ; black, Figure 6.4a). Evaluation metrics (Section 6.2.6) are calculated for the test fraction, with results (Section 6.3) using all samples except for the upwelling shortwave radiation flux density as this is zero at night-time (i.e., daytime:  $\text{OBS} > 2 \text{ W m}^{-2}$ ,  $N = 4\,272$ ).

Model-specific setups are as follows:

1. TEB uses morphological parameters from MP for the four Stages (Table S.3) forced with MF. TEB is run with 5-min (300-s) timesteps (from 13 August 2003 00:00 to 27 November 2004 23:30) after linear interpolation of the 30-min MF data set (e.g., 00:00, 00:05, ...) to predict the next 5-min (e.g., 00:05, 00:10, ...). The last 5-min sample of each 30-min period (e.g., 00:30) is used in analyses (Section 6.3). From the evaluation of TEB outputs at all Stages, Stage 4 is selected as it has the smallest errors (Appendix C).
2. The UNN is trained with MF as inputs and MEM from Stage 2 as outputs (Table 6.1) using the training fraction. Stage 2 is selected as it offers the 'best' trade-off between complexity (i.e., number of parameters used to configure the 22 ULSMs; Table S.3) and accuracy (Appendix C). Prior to training, the surface temperature  $T_s$  is derived from the MEM upwelling

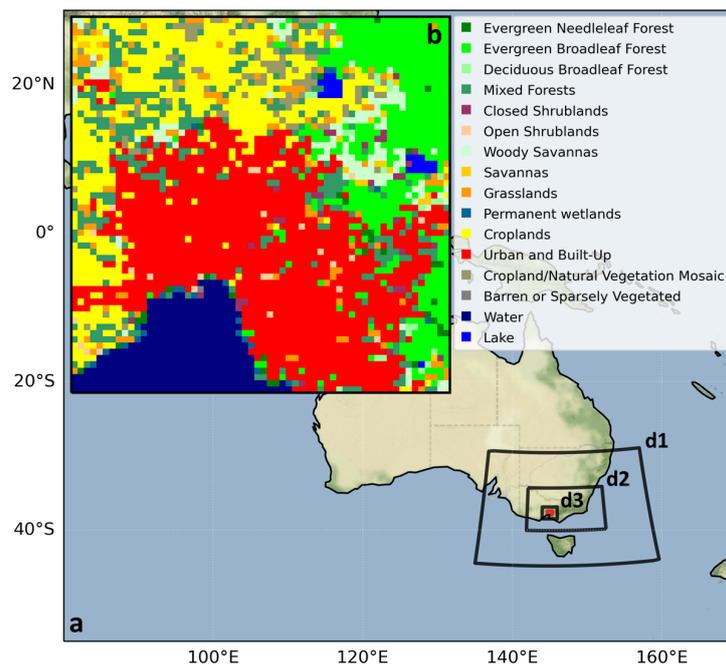
longwave radiation flux density  $L^\uparrow$  assuming a constant emissivity (Table 6.1b). To allow the UNN to be used with different timestep lengths, nine linearly interpolated copies of both inputs and outputs are made (with 1, 2, 5, 10, 20, 60, 120, 300, and 600-s timesteps), each derived from the 30-min data, and concatenated together with the original. A random subset corresponding to the same number of samples included with the 30-min data ( $N = 13\ 838$ ) is selected in each copy to keep the number of training samples across the linearly interpolated copies equal. Thus, the total number of samples used for training the UNN is 138 380 (i.e., ten times the original 30-min data). Of this, 25 % are randomly reserved for the early stopping mechanism. For inference, the UNN is forced with 5-min MF (12 August 2003 23:30 and 27 November 2004 23:30) derived by linearly interpolating the 30-min intervals to be consistent with TEB. As UNN outputs (Table 6.1b) include  $T_s$  rather than  $L^\uparrow$  used in evaluations, UNN outputs are postprocessed to derive  $L^\uparrow$  from  $T_s$  assuming a constant emissivity (Table 6.1b). The stochastic nature of the multilayer perceptrons is assessed by repeating the training (and inference) 100 times, each with a different random seed. At each iteration (a) for each UNN output variable ( $S^\uparrow, L^\uparrow, Q_H, Q_E$ ; Table 6.1b), the normalized mean absolute error (nMAE; Section 6.2.6) is computed using the ‘true’ MEM and UNN-predicted samples for the whole period (i.e., both train and test fractions) and (b) the mean nMAE ( $n\bar{MAE}$ ) is computed as  $0.25 \left( nMAE_{S^\uparrow} + nMAE_{L^\uparrow} + nMAE_{Q_H} + nMAE_{Q_E} \right)$ . The UNN with the median  $n\bar{MAE}$  from the 100 iterations (Figure S.4) is taken as the representative UNN and used in analyses (Section 6.3). Thus, all UNN-relevant metrics (Section 6.2.6) are computed using results from the UNN with the median  $n\bar{MAE}$ .



**Figure 6.4 | Meteorological data.** Observed meteorological forcing (MF) data (30-min, sources Section 6.2.5): **(a)** downwelling shortwave radiation flux density with period with evaluation observed fluxes (i.e., test fraction) shown (purple), **(b)** downwelling longwave radiation flux density, **(c)** atmospheric surface pressure, **(d)** dry-bulb air temperature, **(e)** relative humidity, **(f)** wind speed and **(g)** wind direction, and **(h)** rainfall rate. Wind speed and direction computed with MetPy version 1.1.0 (May et al., 2021) from the zonal and meridional components of wind velocity. Relative humidity computed with PsychroLib (Meyer & Thevenard, 2019) version 2.5.0 (Meyer & Thevenard, 2020) from dry-bulb air temperature, specific humidity, and atmospheric surface pressure. Local time is 10-h ahead of Coordinated Universal Time (UTC+10).

3. The coupled WRF-TEB and WRF-UNN simulations are set up with four nested domains (Figure 6.5), generated with GIS4WRF (Meyer & Riechert, 2019a) version 0.14.4 (Meyer & Riechert, 2020) and processed using WPS-CMake (WRF Preprocessing System) version 4.1.0 (Riechert & Meyer, 2019a). Both TEB and the UNN are run for the innermost domain (Figure 6.5b) with a 5-s timestep centered on the Preston measurement tower. The innermost domain with 1 km horizontal grid spacing has a 66 m vertical grid spacing close to the surface, increasing with height. As model buildings are assumed to be within the ground (Meyer, Schoetter, Riechert, et al., 2020) the flux tower sensors at 40 m a.g.l. is 33.6 m above the model surface (as mean building height = 6.4 m, Table S.3). WPS MODIS land use data (UCAR, 2019) are used as one urban class with the same urban and

vegetation fractions (i.e., all grid cells) for consistency between TEB and UNN simulations. Stage 4 parameters (Table S.3) are used in both WRF-TEB, and TEB-offline runs. The European Centre for Medium-Range Weather Forecasts (ECMWF) Cycle 28r2 analysis (ECMWF, 2004) are used to provide the initial and boundary conditions. Other parameters used to configure WRF and WPS are given in Table S.4. Simulations are run for summer (23 December 2003 10:00–27 December 2003 10:00) and winter (25 June 2004 10:00–31 June 2004 04:00) periods, with evaluations using 65 hr in summer (24 December 2003 14:30–27 December 2003 07:30) and 98.5 hr in winter (26 June 2004 21:00–30 June 2004 23:30) to allow some model spin-up; giving the longest continuous observation evaluation periods in the two seasons. Instantaneous WRF fluxes at each 5-min interval (e.g., 00:00, 00:05, ...) are averaged to 30-min time ending values for comparison with OBS.



**Figure 6.5 | Study location and domains.** The study area (a) within Australia and the four nested domains (d1-d3 and innermost, red) used in online simulations, and (b) innermost domain (1 km horizontal grid spacing) with WPS MODIS 30 arc-sec land cover/use (UCAR, 2019). The UNN and TEB are run for ‘urban’ (red) grid cells in WRF assuming land cover fractions of 0.445 building, 0.38 vegetation, and remainder non-building impervious. Sources: map features by Natural Earth Vector (Kelso & Patterson, 2009) are in the public domain. Map tiles by Stamen Design (2021), under Creative Commons Attribution 3.0 license (CC BY 3.0). Data from OpenStreetMap (OpenStreetMap contributors, 2017), under Open Data Commons Open Database License (ODbL).

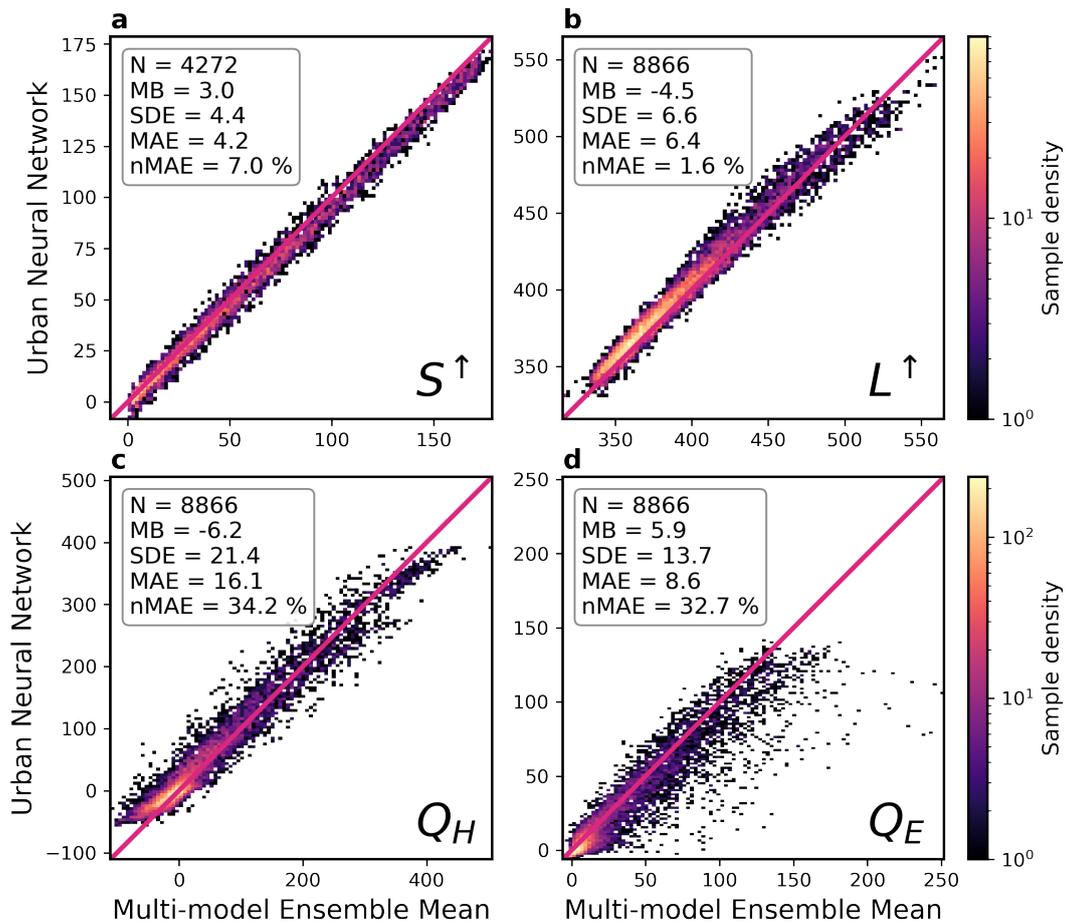
## 6.2.6 Evaluation Metrics

To assess the simulations, statistics are computed between ‘true’  $y^t$  and predicted  $\hat{y}^t$  samples at time  $t$  for  $N$  timesteps. The metrics used are: mean bias ( $MB = \frac{1}{N} \sum_{t=1}^N \hat{y}^t - y^t$ ), mean absolute error ( $MAE = \frac{1}{N} \sum_{t=1}^N |\hat{y}^t - y^t|$ ), mean absolute error normalized by mean absolute ‘true’ flux  $|\bar{y}|$  (hereafter referred to as the normalized mean absolute error,  $nMAE = 100 \% [MAE / |\bar{y}|]$ ) and standard deviation of the error ( $SDE = \sqrt{\frac{1}{N} \sum_{t=1}^N [(\hat{y}^t - y^t) - (\hat{y} - y)]^2}$ ). Depending on the evaluation type, ‘true’ samples  $y^t$  are from either OBS or MEM and predicted samples  $\hat{y}^t$  are from either MEM, UNN or TEB outputs (Section 6.2.5).

## 6.3 Results and Discussion

### 6.3.1 Multi-model Ensemble Mean

First, we assess the trained urban neural network (UNN) using the test fraction (Section 6.2.5) of the multi-model ensemble mean (MEM) data set to determine if the UNN captures the main processes in predicting the surface energy balance. A perfect emulator would have all points on the line  $x = y$  (Figure 6.6). The UNN has the highest skill for the daytime upwelling shortwave radiation flux density (Figure 6.6a): mean bias (MB) is  $3.0 \text{ W m}^{-2}$ , standard deviation of the error (SDE)  $4.4 \text{ W m}^{-2}$ , mean absolute error (MAE)  $4.2 \text{ W m}^{-2}$ , and normalized mean absolute error (nMAE) 7.0 %. The UNN-predicted longwave flux (Figure 6.6b) is slightly poorer (MB\SDE\MAE are  $-4.5\backslash6.6\backslash6.4 \text{ W m}^{-2}$ ) but with a lower nMAE (1.6 %) because of the larger absolute fluxes. The turbulent latent (Figure 6.6d) heat flux density is more accurately predicted (MB\SDE\MAE are  $5.9\backslash13.7\backslash8.6 \text{ W m}^{-2}$  and nMAE is 32.7 %) than the sensible (MB\SDE\MAE are  $-6.2\backslash21.4\backslash16.1 \text{ W m}^{-2}$  and nMAE is 34.2 %; Figure 6.6c) but with larger outliers. This relative ranking is consistent with the extensive ULSM evaluation literature.



**Figure 6.6 | Comparison of UNN to MEM.** 1:1 line (red), data density (color) and evaluation statistics (Section 6.2.6) shown, for 30-min flux densities of (a) daytime upwelling shortwave radiation ( $S^\uparrow$ ) and (b) 24 h upwelling longwave radiation ( $L^\uparrow$ ), turbulent (c) sensible ( $Q_H$ ), and (d) latent ( $Q_E$ ) heat. Units are  $W\ m^{-2}$  except for the nMAE percentage (%). Note that axes scales differ between plots.

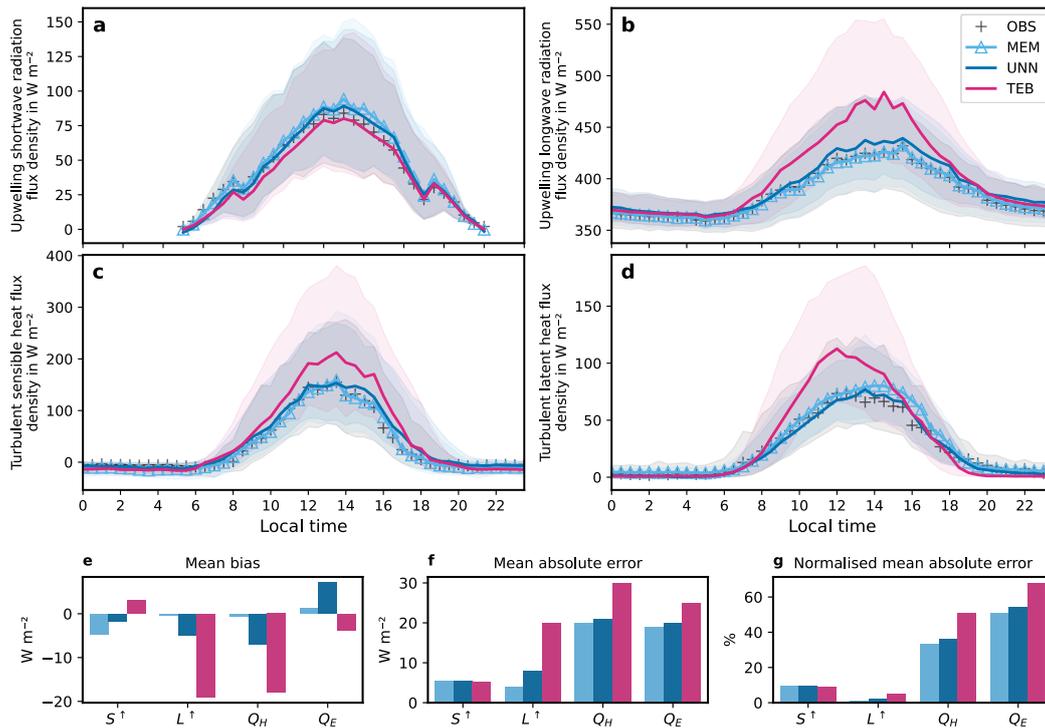
### 6.3.2 Offline Simulations

Second, we compare the UNN to observed fluxes (OBS) and a version of the widely used ULSM TEB (Section 6.2.3). Two evaluations are undertaken with each using different meteorological forcing data: (a) observed MF (offline) and (b) coupled to WRF (online). The offline results are analysed for both the test fraction and the short summer and winter periods (Section 6.2.5; Table 6.2). Online simulations (Section 6.3.3) are only evaluated for the latter two periods.

**Table 6.2 | Results summary.** Observed flux densities (upwelling shortwave  $S^\uparrow$  and longwave  $L^\uparrow$  radiation, and turbulent sensible  $Q_H$  and latent  $Q_E$  heat) and modeled (UNN, MEM, and TEB) evaluation metrics (Section 6.2.6) for N 30-min periods with the lowest nMAE (bold) per flux indicated per data cohort

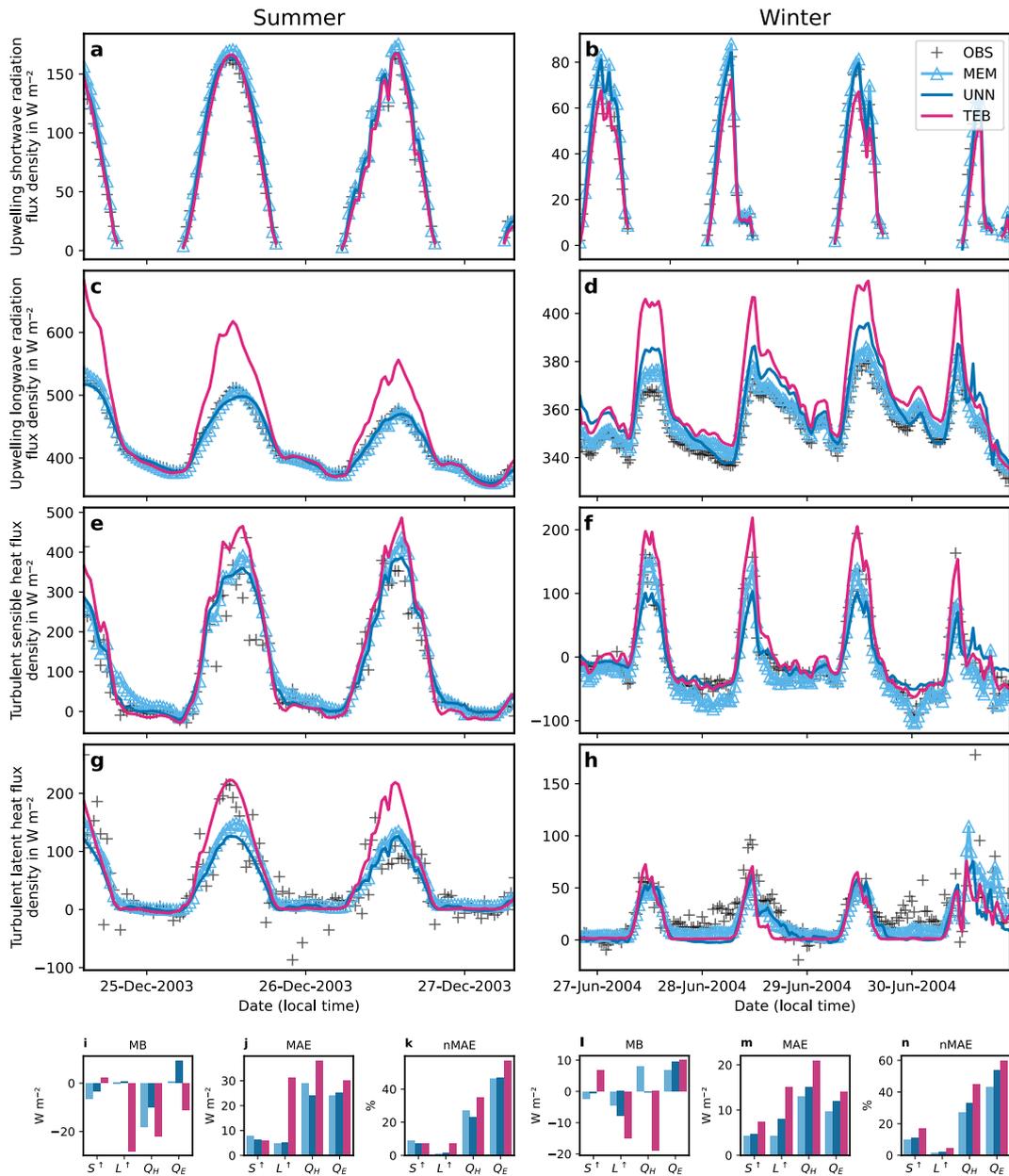
	$S^\uparrow$	$L^\uparrow$	$Q_H$	$Q_E$	$S^\uparrow$	$L^\uparrow$	$Q_H$	$Q_E$	$S^\uparrow$	$L^\uparrow$	$Q_H$	$Q_E$
<b>Observed</b>	<b>Mean (<math>W m^{-2}</math>)</b>											
<i>16 months</i>	58	390	40	34	-	-	-	-	-	-	-	-
<i>65.5 hours in summer</i>	87	420	100	46	-	-	-	-	-	-	-	-
<i>82 hours in winter</i>	41	350	-3.9	22	-	-	-	-	-	-	-	-
<b>Simulated</b>	<b>MB (<math>W m^{-2}</math>)</b>				<b>MAE (<math>W m^{-2}</math>)</b>				<b>nMAE (%)</b>			
<i>16 months (13 Aug. 2003 00:00 – 27 Nov. 2004 23:30; N = 8 866, except <math>S^\uparrow</math> N = 4 272)</i>												
MEM	-4.7	-0.39	-0.67	1.3	5.3	4.0	20	19	9.2	<b>1.0</b>	<b>33</b>	<b>51</b>
UNN	-1.7	-4.9	-6.9	7.1	5.5	7.8	21	20	9.4	2.0	36	54
TEB	3.0	-19	-18	-3.8	5.1	20	30	25	<b>8.7</b>	5.2	51	68
<i>65.5 hours in summer (24 Dec. 2003 14:30 – 27 Dec. 2003 07:30; N = 131, except <math>S^\uparrow</math> N = 73)</i>												
Offline: MEM	-6.6	0.4	-18	0.53	7.8	4.6	29	24	9.0	<b>1.1</b>	27	<b>46</b>
UNN	-3.2	0.65	-9.8	9.3	6.3	5	24	25	7.2	1.2	<b>23</b>	47
TEB	2.1	-28	-22	-11	6	31	38	30	<b>6.9</b>	7.3	35	57
Online: WRF-UNN	-5.8	14	15	-4.7	9.5	18	38	26	11	<b>4.2</b>	<b>35</b>	<b>49</b>
WRF-TEB	-2.1	-11	-41	-12	8.3	35	53	34	<b>9.6</b>	8.2	49	64
<i>82 hours in winter (26 June 2004 21:00 – 30 June 2004 23:30; N = 164, except <math>S^\uparrow</math> N = 56)</i>												
Offline: MEM	-2.4	-4.3	7.9	6.8	4.2	4.3	13	9.7	<b>10</b>	<b>1.2</b>	<b>27</b>	<b>43</b>
UNN	-0.43	-7.9	-0.24	9.4	4.7	8.0	15	12	11	2.3	33	54
TEB	6.7	-15	-19	10	7.3	15	21	14	17	4.2	45	60
Online: WRF-UNN	-3.3	-1.3	0.53	5.9	8.8	9.3	18	13	<b>21</b>	<b>2.6</b>	<b>38</b>	<b>56</b>
WRF-TEB	3.0	-3.5	-21	8.8	11	16	27	18	28	4.4	59	81

Relative to OBS, MEM has the lowest biases and errors for all but the upwelling shortwave radiation flux density, where it is outperformed by TEB (Figure 6.7; Table 6.2). Similarly, as the UNN captures the main surface processes modelled by the MEM (Section 6.3.1), its predictions outperform TEB's for all but the upwelling shortwave radiation. The MAE for the upwelling shortwave radiation (Figure 6.7a; Table 6.2) for all (MEM\UNN\TEB) is  $< 6 W m^{-2}$  (Figures 6.7e–6.7g; Table 6.2). The MAE for the longwave (Figure 6.7b) is larger than for the shortwave, with TEB ( $20 W m^{-2}$ ) having the largest between both MEM and UNN (MEM  $4 W m^{-2}$ ; UNN  $7.8 W m^{-2}$ ). Similarly, the MAE for the turbulent sensible (Figures 6.7c and 6.7f) and latent (Figures 6.7d and 6.7f) heat flux densities are larger for TEB ( $30$  and  $25 W m^{-2}$ , respectively) compared to MEM and UNN ( $\leq 21 W m^{-2}$ , Table 6.2). The nMAE for the latent is larger than sensible because of its smaller mean (Table 6.2; Figure 6.7g).



**Figure 6.7 | Offline (16-months,  $N = 8\,865$ ) simulated (MEM, UNN, TEB, see text) and observed (OBS) 30-min flux densities (lines) with interquartile range (shading). (a) upwelling short- ( $S^\dagger$ ) (daytime) and (b) long-wave ( $L^\dagger$ ) radiation, turbulent (c) sensible ( $Q_H$ ) and (d) latent ( $Q_E$ ) heat, with (e-g) their respective evaluation metrics (Table 6.2). Note that y-axes differ between plots.**

The short summer and winter focal periods are consistent with the 16-month results (Table 6.2, Figure 6.8), with generally higher biases and errors for TEB than for either MEM or UNN. However, both MEM and UNN outperform TEB in the winter, notably with better accuracy for upwelling shortwave radiation (Figures 6.8b and 6.8i–6.8n; Table 6.2).



**Figure 6.8 | Offline simulated (MEM, UNN and TEB) and observed (OBS) 30-min flux densities.** These are shown for (a,c,e,g, i-k) a summer and (b,d,f,h, l-n) winter period for (a-b) upwelling short- ( $S^{\uparrow}$ ) and (c-d) long-wave ( $L^{\uparrow}$ ) radiation, turbulent (e-f) sensible ( $Q_H$ ) and (g-h) latent ( $Q_E$ ) heat, with (i-n) evaluation metrics (Table 2). Note that y-axes differ between plots.

### 6.3.3 Online Simulations

The online coupled WRF-UNN and WRF-TEB simulation results for the grid cell centered on the measurement site are shown in Figure 6.9. The numerical stability of WRF-UNN (trained using MEM for Stage 2; Section 6.2.5) is demonstrated by executing thousands of iterations for hundreds of grid cells without numerical failure (i.e., 2-week, hundreds of domain grid points). The

WRF-UNN post-spin-up periods for both winter and summer (Figure 6.9) capture the general trends of observations. Despite being derived using data from simulations driven with Stage 2 parameters (Table S.3), WRF-UNN has better predictive skills than WRF-TEB (using Stage 4 parameters) for both summer and winter periods (Figure 6.9). WRF-UNN errors are generally lower than those of WRF-TEB in both seasons for all but the winter upwelling shortwave radiation flux density (Figure 6.9; Table 6.2) and generally consistent with the errors shown for offline simulations (Section 6.3.2).

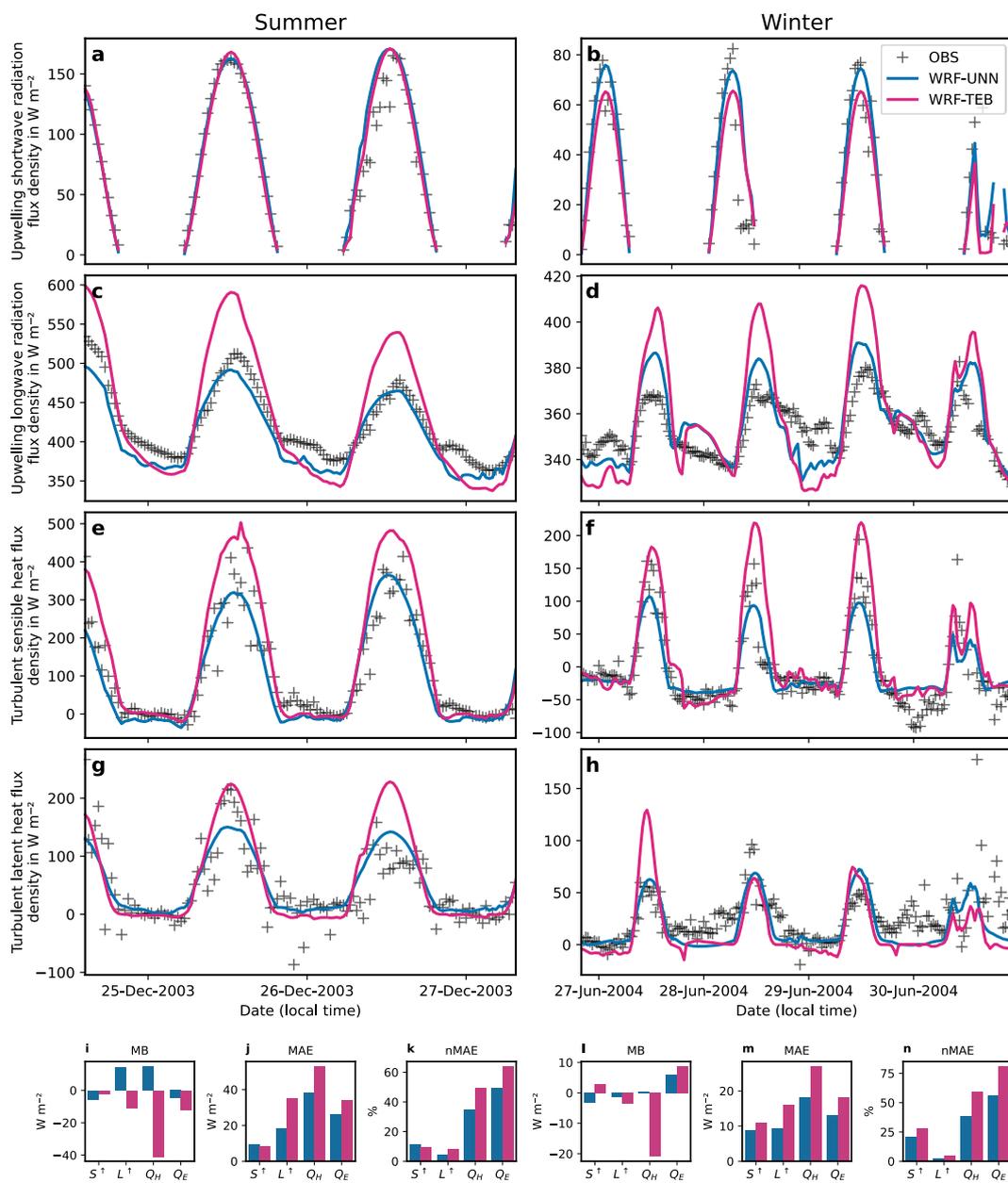
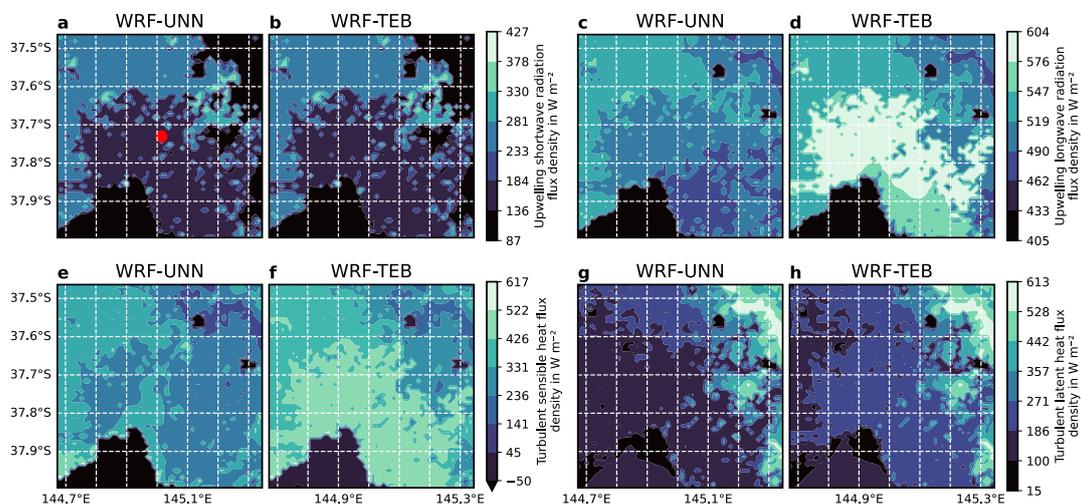


Figure 6.9 | As Figure 6.8, but for online simulations.

As online simulations are run for the entire domain, an additional, albeit qualitative, comparison can be made across the spatial domain. The inner domain surface cover is assigned the same land cover fractions (building 0.445, paved/road 0.175, vegetated 0.380) for all ‘urban’ grid cells (d4 red, Figure 6.5) in both WRF-TEB and WRF-UNN. Given the temporal difference between the two simulations is greatest around midday on 25 December 2003 (Figure 6.9c), we select this time for the spatial comparisons (Figure 6.10). Although the simulated upwelling shortwave radiation flux density (Figures 6.10a and 6.10b) has a similar pattern across the domain, the longwave (Figures 6.10c and 6.10d) in WRF-UNN has a smaller magnitude and spatial range across all the urban grid cells. As a result, the WRF-TEB upwelling longwave radiation flux density is overpredicted by about  $100 \text{ W m}^{-2}$  (Figure 6.9e). The warmer WRF-UNN surface temperature can explain the larger turbulent sensible heat flux density at the observational site (Figure 6.9e) and across the domain (Figures 6.10e and 6.10f).



**Figure 6.10 | Spatial comparison of online simulations.** The inner domain (d4, Figure 6.5) with observation tower (red) for the 30-min average period between 12:00 and 12:30 on 25 December 2003 with flux densities simulated using (a,c,e,g) WRF-UNN and (b,d,f,h) WRF-TEB for (a-b) upwelling shortwave (c-d) and longwave radiation, and turbulent (e-f) sensible and (g-h) latent heat.

### 6.3.4 Computational Performance

The runtime between offline UNN and TEB simulations is compared based on 100 repeats each for the 16-month period. UNN runs include data normalization and inference using TensorFlow in Python. All runs are conducted on a shared AMD EPYC 7742 CPU node with 32 cores and 124 GiB of system memory on a shared cluster. Both TEB and the UNN are configured to run fully single-threaded in a Singularity container running Ubuntu 20.04, GNU Fortran 9.3 compiler, and Anaconda Python 3.9. UNN ( $0.50 \pm 0.0053$  s) runs are over one order of magnitude faster than TEB runs ( $6.0 \pm 0.042$  s).

## 6.4 Conclusion

In this work, we successfully develop a neural network emulator of urban land surface processes (UNN) for offline and online applications. The UNN is trained on the multi-model ensemble mean (MEM) of 22 urban land surface models (ULSMs) for an area of Melbourne, Australia. The accuracy is assessed using flux observations and compared to a well-known ULSM (Town Energy Balance TEB) model. The MEM data are derived from a study with four Stages of increasing complexity (1–4; Appendix C). The UNN is trained using Stage 2 MEM, but compared to the Stage 4 TEB simulations, the latter using more site-specific information.

Compared to MEM, the UNN captures the general variability of surface energy balance fluxes. Relative to the observations, the UNN is more accurate than TEB—or than WRF-TEB when coupled to the Weather Research and Forecasting (WRF) model—while having reduced both computational demands (by over an order of magnitude) and model parameter requirements (i.e., trained using fewer site-specific parameters). Technically, the coupling to WRF is straightforward thanks to WRF-CMake and TensorFlow Lite C bindings.

As the first study to show the development and application of a machine learning (ML) emulator for urban land surface fluxes, we demonstrate its potential to improve the modelling of key surface energy balance fluxes: we combine the strengths of several ULSMs into one and show that such models can be successfully integrated into complex weather models, such as WRF. The development of (coupled) emulators such as WRF-UNN have other advantages compared to 'more-traditional' ULSMs such as code optimisation at the deployment stage, and integration into different codebases and hardware architectures through common high-level APIs.

Although the current evaluation did not assess, or assume, surface energy balance closure, which is essential for climate applications (Grimmond et al., 2010), further research is needed to assess this before UNNs are used in climate studies. Furthermore, with no variations in urban areas (e.g., land cover fractions, surface parameters, and climate) assessed because of the current lack of multi-site data sets, the natural progression to assess our findings more globally requires data sets currently being developed (Lipson et al., 2020) with or without data augmentation strategies as outlined by Meyer et al. (2021).

Indeed, if MEMs are found to be more accurate than any individual ULSM on a global scale, an ML emulator as described here could help improve both the speed and accuracy of current ULSMs. Aside from the apparent speed-up improvement typical of ML emulators and that of improved accuracy outlined here, ML approaches may also prove helpful in operational NWP models as the fewer site-specific parameters contained in MEM are easily retrievable and updatable globally using remote sensing techniques.

*In the next and concluding chapter, I will summarize the main conclusions, focusing on the general applicability of machine learning emulation in the context of weather modelling.*

# CHAPTER 7

---

## Conclusions

*In this final and concluding chapter, I summarise the main research findings, outline the lessons learnt, and briefly discuss what I think we will see in the next few years in the context of (operational) numerical weather prediction.*

# 7.1 Conclusions and Contributions

Skilful weather predictions and climate projections are needed to mitigate the effects of weather and climate (Bauer, Dueben, et al., 2021). The need for digital twins of the Earth system to investigate future policy scenarios poses a new challenge to developing weather and climate models and algorithms that run more efficiently on new hardware architectures (Bauer, Stevens, et al., 2021). Machine Learning (ML) techniques have been shown to be helpful in this transition by improving the performance of current algorithms with a trade-off in accuracy (emulation) that can be adapted to a vast range of architectures (e.g., Central Processing Units, or CPUs; Graphical Processing Units, or GPUs; and Tensor Processing Units, or TPUs). This PhD thesis finds that two widely used radiative transfer and urban land surface schemes can be sped up using ML emulators by about five and ten times, respectively, while keeping or improving their overall accuracy. The computational performance is evaluated by comparing the ML models' CPU runtime to that of traditional models. Moreover, because ML frameworks allow portability to different hardware architectures, an even greater speedup could be achieved when switching to GPUs or TPUs.

With higher resolution requirements, a more detailed representation of radiative transfer and land surface becomes critical in weather predictions and climate projections. In particular, the absence of 3D cloud radiative effects representation can lead to the underestimation of up to 30 % in the longwave (Heidinger and Cox 1996) and changes in instantaneous shortwave fluxes up to  $40 \text{ W m}^{-2}$  (Hogan et al. 2019). Similarly, a more detailed representation of urban processes in land surface schemes can provide more accurate lower boundary conditions (e.g., radiative and turbulent heat fluxes) to atmospheric models for urban areas (Grimmond et al., 2009, 2010). Here, the focus is on keeping or improving the accuracy of parametrizations when developing ML emulators. As such, two novel methods are developed:

1. Radiation schemes. Simulate the 3D effects as a correction term to a fast 1D radiation calculation rather than simulating the entire radiation problem in one go, as earlier studies have done, but at the cost of large errors. The correction term (the difference between ecRad's fast Tripleclouds solver that neglects 3D effects) is "learnt" between Tripleclouds and SPARTACUS. As clear-sky radiation is efficiently and accurately computed using the existing 1D radiation scheme (Tripleclouds), only tropospheric fluxes in cloudy conditions need to be changed, allowing this hybrid physical-ML approach to be more tolerant to errors and outliers. Although hybrid physical ML models have been used previously, they have been for slow components of a single complex scheme (Ukkonen et al., 2020; Veerman et al., 2021). Here the emulator is constructed from the difference between two physical schemes, which may not always be available.
2. Urban land surface schemes. Given that previous studies have found that no single model is 'best' at modelling all surface heat fluxes of cities (Grimmond et al., 2010), the proposed approach takes two steps: (a) to develop an ensemble approach that is demonstrated to be able to outperform a mature reference model; and (b) to emulate the ensemble which is shown to outperform the baseline reference model.

In addition to these novel contributions, the augmentation of datasets with synthetic samples to improve the accuracy of ML emulators is explored. Although larger datasets for training ML models are considered the 'best' way to make an ML model more generalizable (Goodfellow et al. 2016), the input data used in parametrization schemes can be challenging to source when it needs to be extracted from NWP models. For example, the knowledge and experience needed to set up NWP models or their high computational costs may be limiting factors. As such, improving ML models using data generation has already been outlined in previous studies for simple cases where no statistical dependence was present (e.g., Ukkonen et al., 2020). However, this

may be neither straightforward nor computationally cheap in the case of complex dependencies between variables. The method developed in this PhD thesis is applied to a simple toy model of downwelling radiation to show that the emulator's accuracy is improved by over 50 %.

With the current trend toward separating scientific code from hardware-dependent software layers by rewriting NWP models using domain-specific software frameworks, the goal is performance portability, whereby scientific code is run efficiently on different hardware, such as GPUs or FPGAs, without the need for adaptation. However, it has been recognised that specific components such as parametrization schemes and data assimilation are difficult to port, partly due to the reduced precision available in hardware such as GPUs. ML models, on the other hand, are a natural fit for hardware with low numerical precision while providing portability through the ecosystem of existing ML frameworks (Bauer, Dueben, et al., 2021). The emulators developed in chapters 3 and 6 for radiative transfer and urban land surface processes contribute to the growing set of problems that ML can successfully and efficiently solve.

## **7.2 Lessons Learnt**

By applying ML emulators to two fundamental atmospheric processes, I show how two parametrization schemes can be made more flexible and accurate while improving or keeping their computational costs at bay. As a result, they may be considered attractive for operational use.

From this research, the following lessons to develop reliable ML emulators for NWP applications are identified:

- Implement the simplest ML algorithms as described in the literature. If new ML algorithms are developed, compare the findings with previous results considering trade-offs between complexity, accuracy and the developed emulator's computational performance.
- Quantify and identify errors associated with an emulator offline before coupling it to an NWP model.
- Enforce standard coupling methods and protocols when coupling and evaluating parametrization schemes for online applications.
- Use easily understandable statistical metrics such as the absolute and relative errors wherever possible over more complex statistical metrics if the overall findings do not change.
- Use open-source software and data to allow the scientific community to reproduce and further develop the work.
- Avoid reinventing the wheel by writing custom code or software frameworks. Instead, reuse and contribute to existing general time-tested software frameworks and libraries developed by core communities. For example, this is shown in chapter 6 with TensorFlow Lite C bindings for coupling the urban neural network to the Weather Research and Forecasting model.
- Publish software and data. Although there may be cases where it is not practical for third parties to rerun experiments (e.g., large dataset or compute resources needed), researchers should nevertheless ensure that their results are reproducible. By doing so, both software creators and their users will benefit—with time, data and software may become unavailable, misplaced, or corrupted, and specific methods used in the research forgotten. Poorly or undocumented data dumps are unhelpful as they do not guarantee reproducibility of research experiments.

## 7.3 Outlook

Some scholars have discussed how ML could one day replace today's NWP models in their entirety (Figure 1.1c; Schultz et al., 2021). Although such monolithic models could prove helpful for gaining more predictive skills while reducing computational costs, I argue that this is unlikely to happen in the next decade within the research or operational context where the study and understanding of phenomena are paramount. Indeed, such a scenario may become a reality with the development of artificial general intelligence<sup>15</sup>, but the current consensus puts it another 30 years in the future (Grace et al., 2018; Müller & Bostrom, 2016). What I see likely to happen in the next ten years, however, is the development and use of highly sophisticated end-to-end ML models for commercial operational weather forecasting (e.g., The Weather Channel) where the main goal is to provide accurate short-to-medium term weather forecast products to their end-users without necessarily having to understand fundamental key patterns and behaviours. In all cases, in the most immediate future, the need to improve the computational speed and accuracy of any NWP models is likely to see a combination of traditional and ML approaches here referred to as hybrid approaches (Figure 1.1b). This will involve replacing parts or whole models for which ML may show a net advantage over more traditional approaches in use today, were it for speed, accuracy, or a mix of both. The private sector's efforts will be essential to develop or use current technologies operationally with fewer cost constraints than its public counterpart. For example, IBM GRAF is currently the first operational global NWP model to run at 3 km spatial resolution on GPUs (IBM, 2019), putting it several years ahead in terms of spatial resolution than leading international EU and US institutes.

---

<sup>15</sup> "the hypothetical ability of an intelligent agent to understand or learn any intellectual task that a human being can" (Hodson, 2019).

# APPENDICES

---

# A: Postprocessing Methods

As introduced in Section 3.2.2, it can be challenging to use NNs to predict flux and heating rate profiles that are both, physically consistent with each other, and with heating rate profiles free from excessive noise. Here we describe a method to obtain consistent profiles by postprocessing NN outputs. Rather than using NNs to predict the profiles of 3D effects on upward and downward fluxes, we use them to predict the profiles of 3D effects on scalar fluxes (equal to the downwelling plus upwelling) and 3D effects on heating rates. As the latter are proportional to the divergence of the 3D effect on the net flux (downwelling *minus* upwelling), the information content is the same, but it is expressed in variables that are closer to what we need, and it is easier for the NNs to predict. For the rest of this appendix we omit the term “3D effects on” prefix for describing fluxes and heating rates. As the postprocessing method is common across longwave and shortwave components, we explain the main method via the longwave and highlight differences in assumptions and processing separately at the end of the section.

The starting point is the output from the neural network: the scalar flux profile at half levels  $L^s = L^\downarrow + L^\uparrow$  (where  $L^\downarrow$  and  $L^\uparrow$  are the downwelling and upwelling fluxes) and the heating rate profile at full levels  $H = -\frac{c_p \Delta L^n}{g}$ , where  $L^n = L^\downarrow - L^\uparrow$  is the net flux,  $\Delta$  denotes the difference between the base and top of a layer so  $\Delta p$  is the pressure difference across a layer, and  $c_p$  and  $g$  are the specific heat of dry air and the gravitational acceleration. The postprocessing consists of the following steps:

1. Compute the total atmospheric flux divergence (i.e., total emission minus absorption, in  $\text{W m}^{-2}$ ) from a heating rate profile. Fundamentally the divergence is the difference in net flux between the bottom-of-atmosphere (BOA) and top-of-atmosphere (TOA), i.e.,  $D = L_{\text{BOA}}^n - L_{\text{TOA}}^n$ . To obtain this from the heating rate, we sum the profile of divergences of individual layers, that

is,  $D^H = \sum \Delta L^n$ , where the  $\Delta L^n$  profile is obtained from the heating rate by inverting the expression for  $H$  above.

2. Compute the total atmospheric flux divergence from the scalar fluxes. At TOA, the downwelling longwave flux is zero so  $L_{\text{TOA}}^n = -L^\uparrow = -L_{\text{TOA}}^s$  (in the shortwave the same formula can be applied because, even though the downwelling shortwave flux is not zero at TOA, the 3D effect on this part is). At BOA, the upwelling longwave flux is dominated by surface emission rather than reflection, so we can assume that the 3D effect is zero, leading to  $L_{\text{BOA}}^n = L^\downarrow = L_{\text{BOA}}^s$ . Therefore, the atmospheric divergence estimated from the scalar fluxes is  $D^s = L_{\text{BOA}}^s + L_{\text{TOA}}^s$ .
3. Rescale the heating rate profile so that its divergence equals that from the scalar flux. This is done by multiplying the heating rates by a scaling factor equal to  $D^s/D^H$ , and, if necessary, capping the scaling factor to lie in the range 0.5 to 2. If capping, the scalar fluxes are also scaled to ensure that they have the same divergence.
4. Use the rescaled heating rate (and hence  $\Delta L^n$ ) and scalar flux profiles to compute the profiles of upwelling and downwelling flux. First the  $L^n$  profile is computed by integrating  $\Delta L^n$  down from TOA from a start value of  $L_{\text{TOA}}^n = -L_{\text{TOA}}^s$ . Then the upwelling and downwelling components are computed from  $L^\uparrow = \frac{L^s - L^n}{2}$  and  $L^\downarrow = \frac{L^s + L^n}{2}$ .

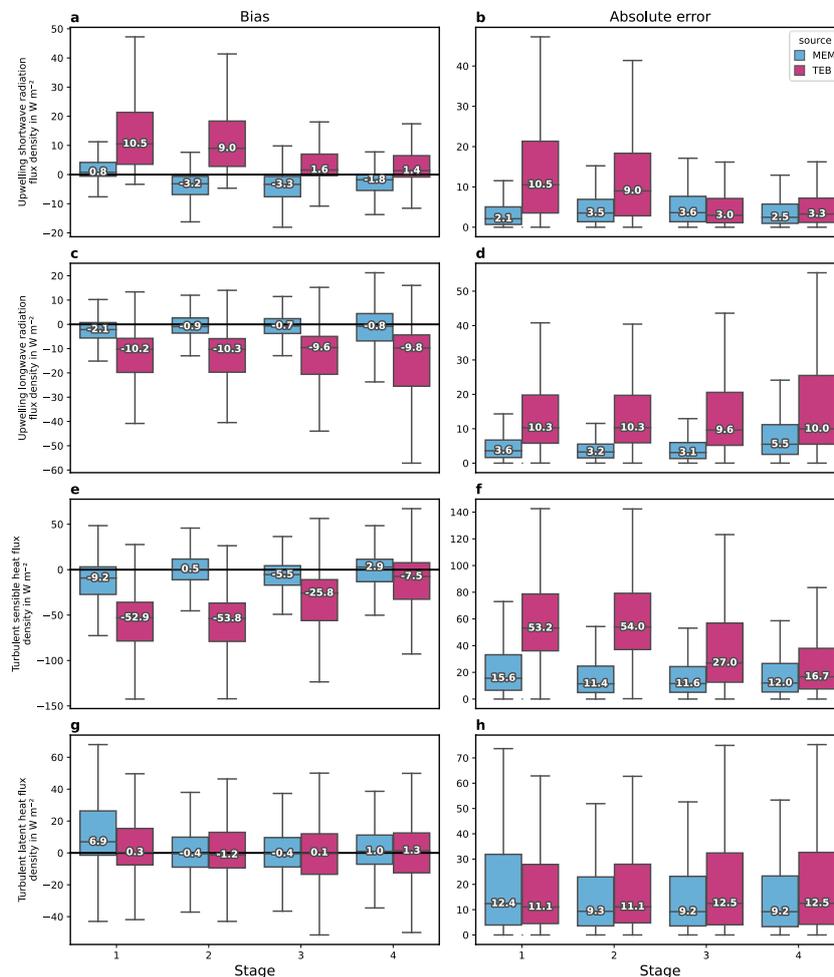
The calculation of shortwave components follows that of the longwave above, except for computing the BOA net flux from the scalar flux in step 2. The net shortwave flux is given by  $S^n = S^\downarrow - S^\uparrow$ , the scalar shortwave flux by  $S^s = S^\downarrow + S^\uparrow$ , and the albedo by  $\alpha = S_{\text{BOA}}^\uparrow / S_{\text{BOA}}^\downarrow$ , thus  $S_{\text{BOA}}^n = S_{\text{BOA}}^s [(1 - \alpha)/(1 + \alpha)]$ . As the total atmospheric flux divergence is the BOA net flux minus the TOA net flux, the total atmospheric flux divergence is computed from the shortwave scalar fluxes as  $D^s = S_{\text{BOA}}^s [(1 - \alpha)/(1 + \alpha)] + S_{\text{TOA}}^s$ . Other steps are identical to those for the longwave.

## B: Statistical Metrics

The vector of differences  $\mathbf{d} = (d_1, \dots, d_j)$  between two vectors  $\mathbf{x}_a$  and  $\mathbf{x}_b$  of paired quantities  $x_{a,i}$  and  $x_{b,i}$  is defined as  $\mathbf{d} = \mathbf{x}_a - \mathbf{x}_b$ . The mean bias error (MBE), mean-absolute error (MAE), and normalized root-mean-square error (NRMSE) for a time series of times  $1, \dots, N$  are defined as  $\text{MBE} = \frac{1}{N} \sum_{i=1}^N d_i$ ,  $\text{MAE} = \frac{1}{N} \sum_{i=1}^N |d_i|$ ,  $\text{NRMSE} = 100 \% \frac{\text{RMSE}}{\bar{x}_a}$  (where  $\bar{x}_a$  is the arithmetic mean of  $\mathbf{x}_a$ ), and the RMSE is defined as  $\sqrt{\frac{\sum_{i=1}^N d_i^2}{N}}$ .

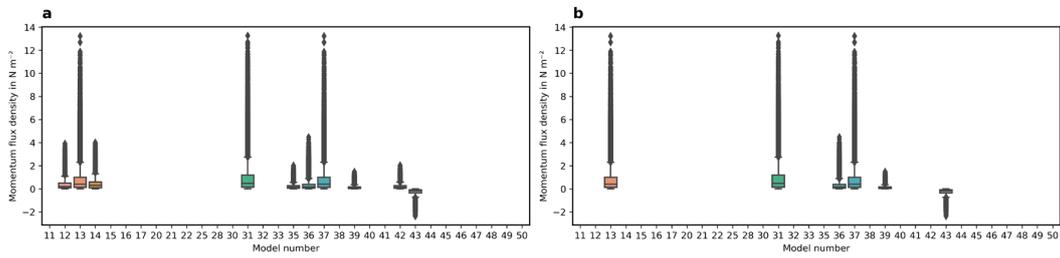
# C: Stage Selection

Stage 2 data (Table S.3) are selected for training the urban neural network (UNN; Section 6.2.2) and Stage 4 for the TEB to offer the best trade-off between complexity and accuracy and better model metrics (Figure C.1). Overall, TEB's mean bias and mean absolute error improve the more information is provided, notably when the site albedo is given in Stage 3 and 4 (Table S.3). MEM generally has the lowest overall mean bias and mean absolute error for Stage 2.

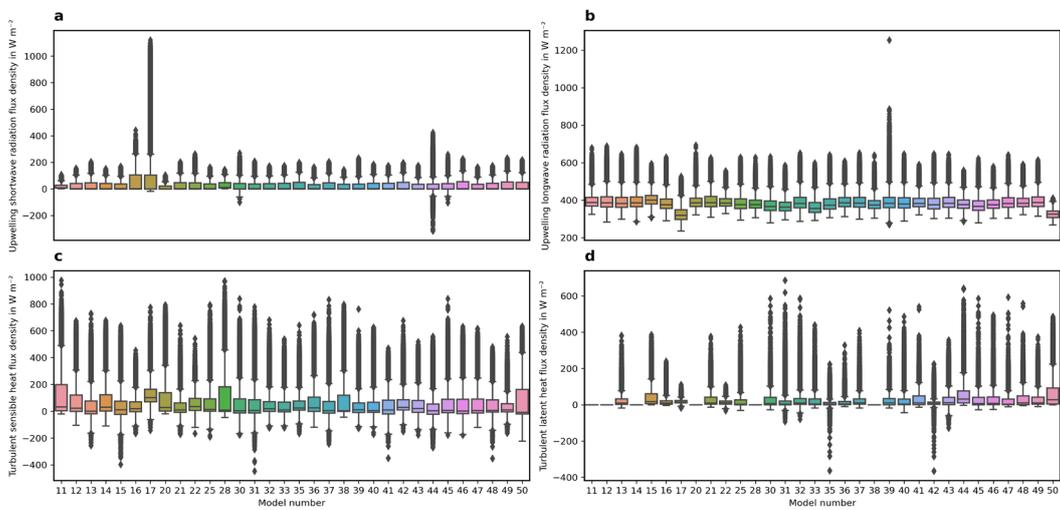


**Figure C.1 | Model metrics used in the stage selection.** Distribution (boxplot with median (labels) interquartile range (IQR) and whiskers: 1.5 IQR) of the 30-min biases and absolute errors (Section 6.2.6) relative to observations (OBS) for four Stages (Table S.3) of multi-model ensemble mean (MEM) and Town Energy Balance (TEB) calculated for the test fraction (Section 6.2.5) of 30-min fluxes of upwelling (a-b) short- and (c-d) long-wave radiation, and turbulent (e-f) sensible and (g-h) latent heat.

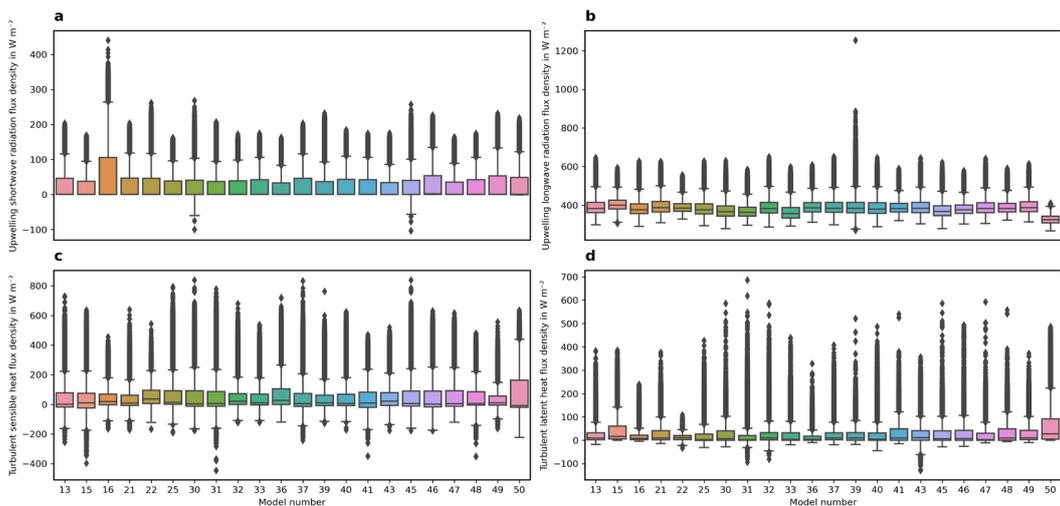
# SUPPLEMENTARY INFORMATION



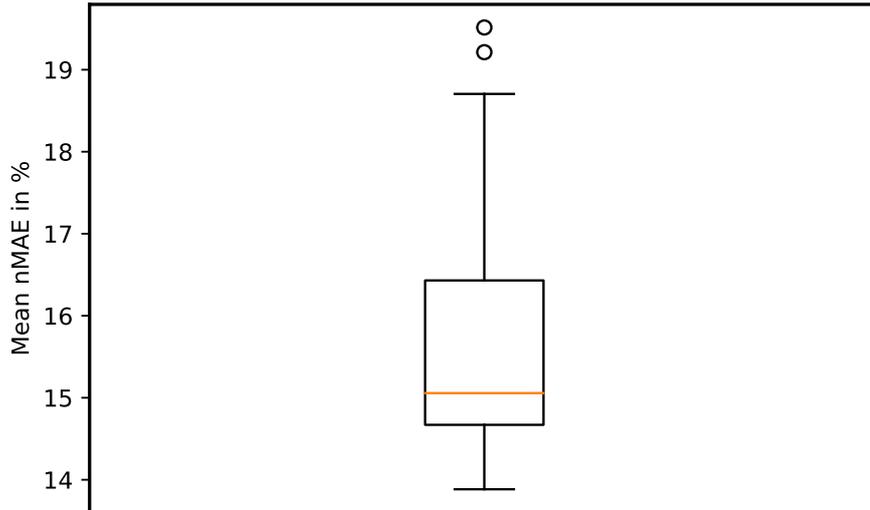
**Figure S.1 | Momentum flux density for the (a) original and (b) reduced number of models outputs data set.** As the original data set does not include the observations and only for six model, sit is not used in this study.



**Figure S.2 | Fluxes modelled by 32 urban land surface models or configurations for all four Stages (Grimmond et al. 2011) for four fluxes.** Upwelling (a) short- and (b) long-wave radiation, turbulent (c) sensible and (d) latent heat.



**Figure S.3 | As Figure S.2, but for the selected 22 urban land surface models.** Poor performers (17 and 44), or models/configurations not including latent heat flux (11, 12, 14, 20, 28, 35, 38, 42) are removed.



**Figure S.4 | Boxplot of 100 mean nMAE UNN iterations (Section 6.2.5).**

**Table S.1 | Configuration used for the grid search.** This uses KerasTuner (O'Malley et al., 2019) version 1.0.4 using Hyperband (Li et al., 2018). The Adam algorithm (Kingma & Ba, 2015) with mean squared error as its optimisation function is used. The epoch limit is set to 200 and the early stopping patience to 20 epochs using 25 % of the training fraction as defined in Section 6.2.5.

Hyperparameter	Values
Number of hidden layers	[1, 2, 3]
Number of Neurons	[16—512, every 32]
Activation function	[relu, tanh, sigmoid]
L2 Regularization	[ $1e^{-3}$ , $1e^{-2}$ , $1e^{-1}$ , $1e^0$ , $1e^1$ , $1e^2$ ]

**Table S.2 | Model participants in the Grimmond et al. (2011) study.**

<b>Model</b>	<b>Reference</b>
Building effect parameterization (BEP)	Martilli et al. (2002)
BEP coupled with building energy model	Martilli et al. (2002); Salamanca et al. (2009, 2010); Salamanca & Martilli (2010)
Community land model – urban (CLM-urban)	Oleson, Bonan, Feddema, & Vertenstein (2008); Oleson, Bonan, Feddema, Vertenstein, et al. (2008)
Institute of Industrial Science urban canopy model	Kawamoto & Ooka (2006, 2009a, 2009b)
Joint UK land environment simulator (JULES)	Best (2005); Best et al. (2006, 2011); Essery et al. (2003)
Local-scale urban meteorological parameterization scheme (LUMPS)	Grimmond & Oke (2002); Loridan et al.(2011); Offerle et al. (2003)
Met Office Reading urban surface exchange scheme (MORUSES)	Harman, Barlow, et al. (2004); Harman, Best, et al. (2004); Porson et al. (2010)
Multi-layer urban canopy model	Kondo et al. (2005); Kondo & Liu (1998)
Nanjing University urban canopy model-single layer	Kusaka et al. (2001); Masson (2000)
National and Kapodistrian University of Athens model	Dandou (2005)
Noah land surface model/single-layer urban canopy model	Chen et al. (2004); Kusaka et al. (2001); Loridan et al. (2010)
Seoul National University urban canopy model	Ryu et al. (2011)
Simple urban energy balance model for mesoscale simulation	Kanda, Kawai, Kanega, et al. (2005); Kanda, Kawai, & Nakagawa (2005); Kawai et al. (2007, 2009)
Single column Reading urban model tile version	Harman & Belcher (2006)
Slab urban energy balance model	Fortuniak (2003); Fortuniak et al. (2005)
Soil model for sub-meso scales (urbanised)	Dupont et al. (2006); Dupont & Mestayer (2006)
Temperatures of urban facets (TUF) 2D	Krayenhoff & Voogt (2007)
Temperatures of urban facets (TUF) 3D	Krayenhoff & Voogt (2007)
Town energy balance (TEB)	Lemonsu et al. (2004); Masson (2000); Masson et al. (2002); Pigeon et al. (2008)
Town energy balance (TEB) with multi-layer option	Hamdi & Masson (2008)
Vegetated urban canopy model	Lee & Park (2007)

**Table S.3 | Morphological parameters provided for the four different Stages (1-4) in Grimmond et al. (2011).** These parameters are used to configure TEB and WRF-TEB. For WRF-TEB the same parameters are used for all grid points classified as urban. Material characteristics provided in Stage 4 have information for four layers per facet (roof, wall, and road): composition/material, width ( $d$ , mm), volumetric heat capacity ( $c$ , MJ m<sup>-3</sup> K<sup>-1</sup>), and thermal conductivity ( $\lambda$ , W m<sup>-1</sup> K<sup>-1</sup>). Parameters not provided, but required to run TEB/WRF-TEB, are set to their default. <sup>a</sup>TEB/WRF-TEB: not used; <sup>b</sup>TEB/WRF-TEB: aggregated to a single vegetation value; <sup>c</sup>TEB/WRF-TEB: same value used for all wall/roof/road facets. \*In Grimmond et al. (2011) modelers were not given the exact latitude and longitude to keep the site anonymous; here these are given for TEB and WRF-TEB simulations.

Stage	Category	Data provided									
1	Forcing	See Table 6.2a									
	Site	*Lat. = -37.7306 °N, *Long. = 145.0145 °E; Measurement height = 30 m									
2	Area fraction	Pervious = 0.38; Impervious = 0.62									
3	Heights	Instrument height = 40 m; Roughness length for momentum = 0.4 m; <sup>a</sup> Max height of roughness elements = 12 m; Mean building height = 6.4 m; Height to width ratio = 0.42; <sup>a</sup> Mean wall to plan area ratio = 0.4									
	Area fraction	Building = 0.445; Concrete = 0.045; Road = 0.130; <sup>b</sup> Vegetation (not Grass) = 0.225; <sup>b</sup> Grass = 0.150; <sup>b</sup> Other (bare or pools) = 0.005									
	Other	Urban climate zone = 5; Population density = 415.78 inhabitants km <sup>-2</sup>									
4	Buildings	Wall									
		Roof									
		Road									
		Layer	$d$	$c$	$k$	$d$	$c$	$k$	$d$	$c$	$k$
		1	40.40	1.25	0.61	11.6	2.07	6.530	28.75	1.14	1.17
	2	54.00	1.40	0.430	50.00	0.0071	0.025	158.30	1.05	0.30	
3	42.00	0.0013	0.024	40.00	1.50	0.230	112.50	1.29	0.42		
4	12.50	0.67	0.160	12.50	0.67	0.160	650.45	1.43	3.72		
Surface		<sup>c</sup> Surface albedo = 0.15; <sup>c</sup> Surface emissivity = 0.97									

**Table S.4 | Main WPS/WRF configuration settings used with the model timestep for each domain (d1–d4).** Lambert Conformal Conic (LCC). <sup>†</sup>Vertical grid spacing increasing with height ( $h$ ) and first level (L1) set to 66 m a.g.l.

Option	Value	TS/Unit	Reference
<i>a) Time</i>			
Timestep length	135, 45, 15, 5	s	-
<i>b) Grid</i>			
Map Projection	LCC	-	-
Horizontal Spacing	27, 9, 3, 1	km	-
Vertical Spacing	$f(h)$ with L1 = 66 <sup>†</sup>	m	-
Vertical Levels	61	-	-
Nests and Grid Ratio	(2)4 and 1:3	-	-
Nesting Approach	1-way	-	-
Urban Classes	1	-	-
<i>c) Initial and Boundary Conditions</i>			
Data Set Name	ECMWF Cycle 28r2 analysis	-	ECMWF (2004)
Horizontal Spacing	TL511 ( $\approx$ 40 km)	-	-
Vertical Levels	61	-	-
Time Interval	6	h	-
<i>d) Physical Parametrization</i>			
Shortwave Radiation	RRTMG	-	Iacono et al. (2008)
Longwave Radiation	RRTMG	-	Iacono et al. (2008)
Microphysics	Single-moment 3-class	-	Hong & Lim (2006)
Cumulus	New Tiedtke Scheme	-	Zhang & Wang (2017)
PBL	YSU	-	Hong et al. (2006)
Surface layer	Revised MM5	-	Jiménez et al. (2012)
LSM	Noah-LSM	-	Chen & Dudhia (2001)
ULSM	TEB/UNN	-	Meyer, Schoetter, Riechert, et al., (2020)

# CODE AND DATA AVAILABILITY

---

All software and tools for the research chapters (Chapters 3-6) are archived separately with a Singularity (Kurtzer et al., 2017) image deposited on Zenodo as described in section 5.5. Users wishing to download (and reproduce) the results described in those chapters can download the respective data archives shown in Table D.1 and optionally run Singularity on their local or remote systems.

**Table D.1 | Links and references of permanent data archives used in this PhD thesis.**

\*Because of licensing restrictions, meteorological forcing (MF), observational (OBS), and multi-model output (MO) data sets cannot be bundled with the Meyer (2021b) data archive and need to be requested separately at <https://doi.org/10.5281/zenodo.4679279> (Grimmond et al., 2021) and <https://doi.org/10.5281/zenodo.4678387> (Grimmond et al., 2013), respectively.

Chapter	Digital Object Identifier	Reference
3	<a href="https://doi.org/10.5281/zenodo.5113055">10.5281/zenodo.5113055</a>	Meyer (2021c)
4	<a href="https://doi.org/10.5281/zenodo.5150327">10.5281/zenodo.5150327</a>	Meyer (2021a)
5	<a href="https://doi.org/10.5281/zenodo.3554517">10.5281/zenodo.3554517</a>	Meyer (2020)
6	<a href="https://doi.org/10.5281/zenodo.5142960">10.5281/zenodo.5142960</a>	*Meyer (2021b)

# ACKNOWLEDGEMENTS

---

I would like to thank my supervisors, Sue Grimmond and Robin Hogan, for their contributions and support throughout my PhD as well as Peter Dueben, Thomas Nagler, Maarten van Reeuwijk, and my other collaborators, for their contributions and feedback. I would also like to thank Robert Pincus for the interesting discussion that inspired chapter 3, Andrew Coutts, Jason Beringer, and the contributors to the urban model comparison project for providing the data used in chapter 6, as well as Brian Henn, Dev Niyogi, Stephen Rasp, and eight anonymous reviewers for reviewing the research papers which formed the bulk of chapters 3-7.

# REFERENCES

---

- Aas, K., Czado, C., Frigessi, A., & Bakken, H. (2009). Pair-copula constructions of multiple dependence. *Insurance: Mathematics and Economics*, 44(2), 182–198. <https://doi.org/10/bmnchh>
- Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., ... Zheng, X. (2015). *TensorFlow: Large-scale machine learning on heterogeneous systems*. <https://www.tensorflow.org/>
- Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., Kudlur, M., Levenberg, J., Monga, R., Moore, S., Murray, D. G., Steiner, B., Tucker, P., Vasudevan, V., Warden, P., ... Zheng, X. (2016). TensorFlow: A System for Large-Scale Machine Learning. *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)*, 265–283. <https://www.usenix.org/conference/osdi16/technical-sessions/presentation/abadi>
- Abbe, C. (1901). The physical basis of long-range weather forecasts. *Monthly Weather Review*, 29(12), 551–561. [https://doi.org/10.1175/1520-0493\(1901\)29\[551c:TPBOLW\]2.0.CO;2](https://doi.org/10.1175/1520-0493(1901)29[551c:TPBOLW]2.0.CO;2)
- Abbe, C. (1909). A chronological outline of the history of meteorology in the United States of north America. *Monthly Weather Review*, 37(4), 146–149. [https://doi.org/10.1175/1520-0493\(1909\)37\[146:ACOOH\]2.0.CO;2](https://doi.org/10.1175/1520-0493(1909)37[146:ACOOH]2.0.CO;2)
- Adams, M. D., & Ward, R. K. (2004). JasPer: A portable flexible open-source software tool kit for image coding/processing. *2004 IEEE International Conference on Acoustics, Speech, and Signal Processing*, 5, V–241. <https://doi.org/10.1109/ICASSP.2004.1327092>
- Allen, L., O'Connell, A., & Kiermer, V. (2019). How can we ensure visibility and diversity in research contributions? How the Contributor Role Taxonomy (CRediT) is helping the shift from authorship to contributorship. *Learned Publishing*, 32(1), 71–74. <https://doi.org/10.1002/leap.1210>
- Alley, R. B., Emanuel, K. A., & Zhang, F. (2019). Advances in weather prediction. *Science*, 363(6425), 342–344. <https://doi.org/10.1126/science.aav7274>
- American Meteorological Society. (2021a). *Climate—Glossary of Meteorology*. <https://glossary.ametsoc.org/wiki/Climate>
- American Meteorological Society. (2021b). *Weather—Glossary of Meteorology*. <https://glossary.ametsoc.org/wiki/Weather>
- Aon. (2021). *Weather, Climate & Catastrophe Insight—2020 Annual Report* (p. 81). Aon plc.
- Aristotle. (1952). *Meteorologica* (H. D. P. Lee, Trans.). Harvard University Press. <https://doi.org/10.4159/DLCL.aristotle-meteorologica.1952>
- Arnfield, A. J. (2003). Two decades of urban climate research: A review of turbulence, exchanges of energy and water, and the urban heat island. *International Journal of Climatology*, 23(1), 1–26. <https://doi.org/10.1002/joc.859>
- Atmanspacher, H., & Maasen, S. (2016). *Reproducibility: Principles, problems, practices, and prospects*. John Wiley & Sons.
- Baklanov, A., Grimmond, C. S. B., Carlson, D., Terblanche, D., Tang, X., Bouchet, V., Lee, B., Langendijk, G., Kolli, R. K., & Hovsepyan, A. (2018). From urban meteorology, climate and environment research to integrated city services. *Urban Climate*, 23, 330–341. <https://doi.org/10.1016/j.uclim.2017.05.004>

- Bauer, P., Dueben, P. D., Hoefler, T., Quintino, T., Schulthess, T. C., & Wedi, N. P. (2021). The digital revolution of Earth-system science. *Nature Computational Science*, 1(2), 104–113. <https://doi.org/10.1038/s43588-021-00023-0>
- Bauer, P., Stevens, B., & Hazeleger, W. (2021). A digital twin of Earth for the green transition. *Nature Climate Change*, 11(2), 80–83. <https://doi.org/10.1038/s41558-021-00986-y>
- Bauer, P., Thorpe, A., & Brunet, G. (2015). The quiet revolution of numerical weather prediction. *Nature*, 525(7567), 47–55. <https://doi.org/10.1038/nature14956>
- Bengtsson, L., Andrae, U., Aspelien, T., Batrak, Y., Calvo, J., de Rooy, W., Gleeson, E., Hansen-Sass, B., Homleid, M., Hortal, M., Ivarsson, K.-I., Lenderink, G., Niemelä, S., Nielsen, K. P., Onville, J., Rontu, L., Samuelsson, P., Muñoz, D. S., Subias, A., ... Køltzow, M. Ø. (2017). The HARMONIE-AROME Model Configuration in the ALADIN-HIRLAM NWP System. *Monthly Weather Review*, 145(5), 1919–1935. <https://doi.org/10.1175/MWR-D-16-0417.1>
- Benjamin, S. G., Brown, J. M., Brunet, G., Lynch, P., Saito, K., & Schlatter, T. W. (2019). 100 Years of Progress in Forecasting and NWP Applications. *Meteorological Monographs*, 59, 13.1–13.67. <https://doi.org/10.1175/AMSMONOGRAPHS-D-18-0020.1>
- Benjamin, S. G., Weygandt, S. S., Brown, J. M., Hu, M., Alexander, C. R., Smirnova, T. G., Olson, J. B., James, E. P., Dowell, D. C., Grell, G. A., Lin, H., Peckham, S. E., Smith, T. L., Moninger, W. R., Kenyon, J. S., & Manikin, G. S. (2016). A North American Hourly Assimilation and Model Forecast Cycle: The Rapid Refresh. *Monthly Weather Review*, 144(4), 1669–1694. <https://doi.org/10.1175/MWR-D-15-0242.1>
- Best, M. J. (2005). Representing urban areas within operational numerical weather prediction models. *Boundary-Layer Meteorology*, 114(1), 91–109. <https://doi.org/10.1007/s10546-004-4834-5>
- Best, M. J., Beljaars, A., Polcher, J., & Viterbo, P. (2004). A Proposed Structure for Coupling Tiled Surfaces with the Planetary Boundary Layer. *Journal of Hydrometeorology*, 5(6), 1271–1278. <https://doi.org/10.1175/JHM-382.1>
- Best, M. J., Grimmond, C. S. B., & Villani, M. G. (2006). Evaluation of the Urban Tile in MOSES using Surface Energy Balance Observations. *Boundary-Layer Meteorology*, 118(3), 503–525. <https://doi.org/10.1007/s10546-005-9025-5>
- Best, M. J., Pryor, M., Clark, D. B., Rooney, G. G., Essery, R. . L. H., Ménard, C. B., Edwards, J. M., Hendry, M. A., Porson, A., Gedney, N., Mercado, L. M., Sitch, S., Blyth, E., Boucher, O., Cox, P. M., Grimmond, C. S. B., & Harding, R. J. (2011). The Joint UK Land Environment Simulator (JULES), model description – Part 1: Energy and water fluxes. *Geoscientific Model Development*, 4(3), 677–699. <https://doi.org/10.5194/gmd-4-677-2011>
- Bishop, C. M. (2006). *Pattern recognition and machine learning*. Springer.
- Bjerknes, V. (1904). *Das problem der wettvorhersage, betrachtet vom standpunkte der mechanik und der physik*. Ed. Hölzel.
- Bjerknes, V. (2009). The problem of weather prediction, considered from the viewpoints of mechanics and physics (E. Volken & S. Brönnimann, Trans.). *Meteorologische Zeitschrift*, 18(6), 663–667. <https://doi.org/10.1127/0941-2948/2009/416>
- Bocher, E., Petit, G., Bernard, J., & Palominos, S. (2018). A geoprocessing framework to compute urban indicators: The MApUCE tools chain. *Urban Climate*, 24, 153–174. <https://doi.org/10.1016/j.uclim.2018.01.008>
- Bolton, T., & Zanna, L. (2019). Applications of Deep Learning to Ocean Data Inference and Subgrid Parameterization. *Journal of Advances in Modeling Earth Systems*, 11(1), 376–399. <https://doi.org/10.1029/2018MS001472>
- Bozzo, A., Benedetti, A., Flemming, J., Kipling, Z., & Rémy, S. (2020). An aerosol climatology for global models based on the tropospheric aerosol scheme in the Integrated

- Forecasting System of ECMWF. *Geoscientific Model Development*, 13(3), 1007–1034.  
<https://doi.org/10.5194/gmd-13-1007-2020>
- Brenowitz, N. D., & Bretherton, C. S. (2018). Prognostic Validation of a Neural Network Unified Physics Parameterization. *Geophysical Research Letters*, 45(12), 6289–6298.  
<https://doi.org/10.1029/2018GL078510>
- Brenowitz, N. D., & Bretherton, C. S. (2019). Spatially Extended Tests of a Neural Network Parametrization Trained by Coarse-Graining. *Journal of Advances in Modeling Earth Systems*, 11(8), 2728–2744. <https://doi.org/10.1029/2019MS001711>
- Bueno, B., Pigeon, G., Norford, L. K., Zibouche, K., & Marchadier, C. (2012). Development and evaluation of a building energy model integrated in the TEB scheme. *Geoscientific Model Development*, 5(2), 433–448. <https://doi.org/10.5194/gmd-5-433-2012>
- Burton, J. (1986). Robert FitzRoy and the Early History of the Meteorological Office. *The British Journal for the History of Science*, 19(2), 147–176.  
<https://doi.org/10.1017/S0007087400022949>
- Cahir, J. J. (2019, January 17). *Weather forecasting*. Encyclopedia Britannica.  
<https://www.britannica.com/science/weather-forecasting>
- Canonical Ltd. (2019). *Ubuntu – Ubuntu Packages Search*. <https://packages.ubuntu.com/>
- Chantry, M., Hatfield, S., Dueben, P., Polichtchouk, I., & Palmer, T. (2021). Machine Learning Emulation of Gravity Wave Drag in Numerical Weather Forecasting. *Journal of Advances in Modeling Earth Systems*, 13(7). <https://doi.org/10.1029/2021MS002477>
- Charney, J. G., Fjörtoft, R., & Neumann, J. (1950). Numerical Integration of the Barotropic Vorticity Equation. *Tellus*, 2(4), 237–254. <https://doi.org/10.1111/j.2153-3490.1950.tb00336.x>
- Chen, F., & Dudhia, J. (2001). Coupling an Advanced Land Surface–Hydrology Model with the Penn State–NCAR MM5 Modeling System. Part I: Model Implementation and Sensitivity. *Monthly Weather Review*, 129, 17. [https://doi.org/10.1175/1520-0493\(2001\)129<0569:CAALSH>2.0.CO;2](https://doi.org/10.1175/1520-0493(2001)129<0569:CAALSH>2.0.CO;2)
- Chen, F., Kusaka, H., Bornstein, R., Ching, J., Grimmond, C. S. B., Grossman-Clarke, S., Loridan, T., Manning, K. W., Martilli, A., Miao, S., Sailor, D., Salamanca, F. P., Taha, H., Tewari, M., Wang, X., Wyszogrodzki, A. A., & Zhang, C. (2011). The integrated WRF/urban modelling system: Development, evaluation, and applications to urban environmental problems. *International Journal of Climatology*, 31(2), 273–288.  
<https://doi.org/10.1002/joc.2158>
- Chen, F., Kusaka, H., Tewari, M., Bao, J., & Hirakuchi, H. (2004). Utilizing the coupled WRF/LSM/Urban modeling system with detailed urban classification to simulate the urban heat island phenomena over the Greater Houston area. *Fifth Symposium on the Urban Environment*, 25, 9–11.
- Chen, X., Dallmeier-Tiessen, S., Dasler, R., Feger, S., Fokianos, P., Gonzalez, J. B., Hirvonsalo, H., Kousidis, D., Lavasa, A., Mele, S., Rodriguez, D. R., Šimko, T., Smith, T., Trisovic, A., Trzcinska, A., Tsanaktsidis, I., Zimmermann, M., Cranmer, K., Heinrich, L., ... Neubert, S. (2019). Open is not enough. *Nature Physics*, 15(2), 113–119.  
<https://doi.org/10.1038/s41567-018-0342-2>
- Chen, Y., Jiang, W. M., Zhang, N., He, X. F., & Zhou, R. W. (2009). Numerical simulation of the anthropogenic heat effect on urban boundary layer structure. *Theoretical and Applied Climatology*, 97(1–2), 123–134. <https://doi.org/10.1007/s00704-008-0054-0>
- Cheruy, F., Chevallier, F., Morcrette, J.-J., Scott, N. A., & Chédin, A. (1996). Une méthode utilisant les techniques neuronales pour le calcul rapide de la distribution verticale du bilan radiatif thermique terrestre. *Comptes Rendus de l'Académie Des Sciences*, 322, 665–672.  
[hal-02954375](https://hal.archives-ouvertes.fr/hal-02954375). <https://hal.archives-ouvertes.fr/hal-02954375>

- Chevallier, F., Morcrette, J.-J., Chéruy, F., & Scott, N. A. (2000). Use of a neural-network-based long-wave radiative-transfer scheme in the ECMWF atmospheric model. *Quarterly Journal of the Royal Meteorological Society*, 126(563), 761–776. <https://doi.org/10.1002/qj.49712656318>
- Chevallier, F., Ruy, F. C., Scott, N. A., & Din, A. C. (1998). A Neural Network Approach for a Fast and Accurate Computation of a Longwave Radiative Budget. *Journal of Applied Meteorology*, 37, 13. [https://doi.org/10.1175/1520-0450\(1998\)037<1385:ANNAFA>2.0.CO;2](https://doi.org/10.1175/1520-0450(1998)037<1385:ANNAFA>2.0.CO;2)
- Clement, V., Ferrachat, S., Fuhrer, O., Lapillonne, X., Osuna, C. E., Pincus, R., Rood, J., & Sawyer, W. (2018). The CLAW DSL: Abstractions for Performance Portable Weather and Climate Models. *Proceedings of the Platform for Advanced Scientific Computing Conference on - PASC '18*, 1–10. <https://doi.org/10.1145/3218176.3218226>
- Cohen-Boulakia, S., Belhajjame, K., Collin, O., Chopard, J., Froidevaux, C., Gaignard, A., Hinsen, K., Larmande, P., Bras, Y. L., Lemoine, F., Mareuil, F., Ménager, H., Pradal, C., & Blanchet, C. (2017). Scientific workflows for computational reproducibility in the life sciences: Status, challenges and opportunities. *Future Generation Computer Systems*, 75, 284–298. <https://doi.org/10.1016/j.future.2017.01.012>
- Collins, M., Knutti, R., Arblaster, J., Dufresne, J.-L., Fichet, T., Friedlingstein, P., Gao, X., Gutowski, W. J., Johns, T., Krinner, G., Shongwe, M., Tebaldi, C., Weaver, A. J., & Wehner, M. (2013). Long-term climate change: Projections, commitments and irreversibility. In T. F. Stocker, D. Qin, G.-K. Plattner, M. Tignor, S. K. Allen, J. Doschung, A. Nauels, Y. Xia, V. Bex, & P. M. Midgley (Eds.), *Climate change 2013: The physical science basis. Contribution of working group I to the fifth assessment report of the intergovernmental panel on climate change* (pp. 1029–1136). Cambridge University Press. <https://doi.org/10.1017/CBO9781107415324.024>
- Coutts, A. M., Beringer, J., & Tapper, N. J. (2007a). Characteristics influencing the variability of urban CO<sub>2</sub> fluxes in Melbourne, Australia. *Atmospheric Environment*, 41(1), 51–62. <https://doi.org/10.1016/j.atmosenv.2006.08.030>
- Coutts, A. M., Beringer, J., & Tapper, N. J. (2007b). Impact of Increasing Urban Density on Local Climate: Spatial and Temporal Variations in the Surface Energy Balance in Melbourne, Australia. *Journal of Applied Meteorology and Climatology*, 46(4), 477–493. <https://doi.org/10.1175/JAM2462.1>
- Craft, E. D. (1999). Private Weather Organizations and the Founding of the United States Weather Bureau. *The Journal of Economic History*, 59(4), 1063–1071. <https://doi.org/10.1017/S0022050700024141>
- Craft, E. D. (2001). *Economic History of Weather Forecasting* (R. Whaples, Ed.). EH.Net Encyclopedia. <https://eh.net/encyclopedia/an-economic-history-of-weather-forecasting/>
- Czado, C. (2019). *Analyzing Dependent Data with Vine Copulas: A Practical Guide With R* (Vol. 222). Springer International Publishing. <https://doi.org/10.1007/978-3-030-13785-4>
- Dandou, A. (2005). Development and evaluation of an urban parameterization scheme in the Penn State/NCAR Mesoscale Model (MM5). *Journal of Geophysical Research*, 110(D10), D10102. <https://doi.org/10.1029/2004JD005192>
- de Munck, C., Pigeon, G., Masson, V., Meunier, F., Bousquet, P., Tréméac, B., Merchat, M., Poeuf, P., & Marchadier, C. (2013). How much can air conditioning increase air temperatures for a city like Paris, France? *International Journal of Climatology*, 33(1), 210–227. <https://doi.org/10.1002/joc.3415>
- Deisenroth, M. P., Faisal, A. A., & Ong, C. S. (2020). *Mathematics for Machine Learning*. Cambridge University Press.

- Developmental Testbed Center. (2018). *DTC | | WRF-NMM Users Page*. <https://dtcenter.org/wrf-nmm/users/>
- Developmental Testbed Center. (2019). *Common Community Physics Package (CCPP)*. <https://dtcenter.org/community-code/common-community-physics-package-ccpp>
- Dißmann, J., Brechmann, E. C., Czado, C., & Kurowicka, D. (2013). Selecting and estimating regular vine copulae and application to financial returns. *Computational Statistics & Data Analysis*, 59, 52–69. <https://doi.org/10/ghfn5n>
- Dudhia, J. (1989). Numerical Study of Convection Observed during the Winter Monsoon Experiment Using a Mesoscale Two-Dimensional Model. *Journal of the Atmospheric Sciences*. [https://doi.org/10.1175/1520-0469\(1989\)046<3077:NSOCOD>2.0.CO;2](https://doi.org/10.1175/1520-0469(1989)046<3077:NSOCOD>2.0.CO;2)
- Dueben, P., Chantry, M., Nipen, T., Denisenko, G., Ben-Nun, T., Gong, B., & Langguth, M. (2021). *D1.1 First version of datasets and cost functions to develop machine learning solutions for A1-A6* (version 1.0; MAchine Learning for Scalable MeTeoROlogy and Climate, p. 30). ECMWF. <https://www.maelstrom-eurohpc.eu/content/docs/uploads/doc6.pdf>
- Dupont, S., & Mestayer, P. G. (2006). Parameterization of the Urban Energy Budget with the Submesoscale Soil Model. *Journal of Applied Meteorology and Climatology*, 45(12), 1744–1765. <https://doi.org/10.1175/JAM2417.1>
- Dupont, S., Mestayer, P. G., Guilloteau, E., Berthier, E., & Andrieu, H. (2006). Parameterization of the Urban Water Budget with the Submesoscale Soil Model. *Journal of Applied Meteorology and Climatology*, 45(4), 624–648. <https://doi.org/10.1175/JAM2363.1>
- ECMWF. (2004). *Cycle 28r2 summary of changes* [Text]. ECMWF Documentation. <https://www.ecmwf.int/en/forecasts/documentation-and-support/evolution-ifs/cycle-archived/cycle-28r2-summary-changes>
- ECMWF. (2020). *EcRad* (1.3.0). <https://confluence.ecmwf.int/display/ECRAD>
- Ehrendorfer, M. (2011). *Spectral Numerical Weather Prediction Models*. Society for Industrial and Applied Mathematics. <https://doi.org/10.1137/1.9781611971996>
- Ek, M. B., Mitchell, K. E., Lin, Y., Rogers, E., Grunmann, P., Koren, V., Gayno, G., & Tarpley, J. D. (2003). Implementation of Noah land surface model advances in the National Centers for Environmental Prediction operational mesoscale Eta model. *Journal of Geophysical Research: Atmospheres*, 108(D22), 2002JD003296. <https://doi.org/10.1029/2002JD003296>
- Elsasser, W. M. (1942). *Heat transfer by infrared radiation in the atmosphere: Vol. no. 6*. Blue Hill Meteorological Observatory.
- Eresmaa, R., & McNally, A. P. (2014). *Diverse profile datasets from the ECMWF 137-level short-range forecasts* (p. 14). EUMETSAT Satellite Application Facility (NWP SAF). <https://nwpsaf.eu/site/download/documentation/rtm/nwpsaf-ec-tr-017.pdf>
- Essery, R. L. H., Best, M. J., Betts, R. A., Cox, P. M., & Taylor, C. M. (2003). Explicit Representation of Subgrid Heterogeneity in a GCM Land Surface Scheme. *Journal of Hydrometeorology*, 4(3), 530–543. [https://doi.org/10.1175/1525-7541\(2003\)004<0530:EROSHI>2.0.CO;2](https://doi.org/10.1175/1525-7541(2003)004<0530:EROSHI>2.0.CO;2)
- Fielding, M. D., Schäfer, S. A. K., Hogan, R. J., & Forbes, R. M. (2020). Parametrizing cloud geometry and its application in a subgrid cloud-edge erosion scheme. *Quarterly Journal of the Royal Meteorological Society*, 146(729), 1651–1667. <https://doi.org/10.1002/qj.3758>
- Forster, P., Ramaswamy, V., Artaxo, P., Berntsen, T., Betts, R., Fahey, D. W., Haywood, J., Lean, J., Lowe, D. C., Myhre, G., Nganga, J., Prinn, R., Raga, G., Schulz, M., & Van Dorland, R. (2007). *Changes in Atmospheric Constituents and in Radiative Forcing Chapter 2*. Cambridge University Press. [http://inis.iaea.org/search/search.aspx?orig\\_q=RN:39002468](http://inis.iaea.org/search/search.aspx?orig_q=RN:39002468)

- Fortuniak, K. (2003). A slab surface energy balance model (SUEB) and its application to the study on the role of roughness length in forming an urban heat island. *Acta Universitatis Wratislaviensis*, 2542, 368–377.
- Fortuniak, K., Offerle, B., & Grimmond, C. (2005). Application of a slab surface energy balance model to determine surface parameters for urban areas. *Lund Electronic Reports in Physical Geography*, 5, 90–91.
- Freitas, E. D., Rozoff, C. M., Cotton, W. R., & Dias, P. L. S. (2007). Interactions of an urban heat island and sea-breeze circulations during winter over the metropolitan area of São Paulo, Brazil. *Boundary-Layer Meteorology*, 122(1), 43–65. <https://doi.org/10.1007/s10546-006-9091-3>
- Frisinger, H. H. (1971). Meteorology Before Aristotle. *Bulletin of the American Meteorological Society*, 52(11), 1078–1080. [https://doi.org/10.1175/1520-0477\(1971\)052<1078:MBA>2.0.CO;2](https://doi.org/10.1175/1520-0477(1971)052<1078:MBA>2.0.CO;2)
- Frisinger, H. H. (1972). Aristotle and His "Meteorologica." *Bulletin of the American Meteorological Society*, 53(7), 634–638. [https://doi.org/10.1175/1520-0477\(1972\)053<0634:AAH>2.0.CO;2](https://doi.org/10.1175/1520-0477(1972)053<0634:AAH>2.0.CO;2)
- Fuhrer, O., Osuna, C., Lapillonne, X., Gysi, T., Cumming, B., Bianco, M., Arteaga, A., & Schulthess, T. C. (2014). Towards a performance portable, architecture agnostic implementation strategy for weather and climate models. *Supercomputing Frontiers and Innovations*, 1(1). <https://doi.org/10.14529/jsfi140103>
- Furley, D. J. (1954). Aristotle's Meteorologica - H. D. P. Lee: Aristotle, Meteorologica. With an English translation. (Loeb Classical Library.) Pp. xxx + 433; 2 maps, diagrams. London: Heinemann, 1952. Cloth, 15s. net. *The Classical Review*, 4(2), 117–119. <https://doi.org/10.1017/S0009840X00177856>
- Gentine, P., Pritchard, M., Rasp, S., Reinaudi, G., & Yacalis, G. (2018). Could Machine Learning Break the Convection Parameterization Deadlock? *Geophysical Research Letters*, 45(11), 5742–5751. <https://doi.org/10.1029/2018GL078202>
- Gettelman, A., & Rood, R. B. (2016). *Demystifying Climate Models* (Vol. 2). Springer Berlin Heidelberg. <https://doi.org/10.1007/978-3-662-48959-8>
- Gochis, D. J., Barlage, M., Dugger, A., FitzGerald, K., Karsten, L., McAllister, M., McCreight, J., Mills, J., RafieeiNasab, A., Read, L., Sampson, K., Yates, D., & Yu, W. (2018). *The WRF-Hydro modeling system technical description, (Version 5.0)* (p. 107). NCAR Technical Note. <https://ral.ucar.edu/sites/default/files/public/WRF-HydroV5TechnicalDescription.pdf>
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT Press.
- Goret, M., Masson, V., Schoetter, R., & Moine, M.-P. (2019). Inclusion of CO2 flux modelling in an urban canopy layer model and an evaluation over an old European city centre. *Atmospheric Environment: X*, 3, 100042. <https://doi.org/10.1016/j.aeaoa.2019.100042>
- Grace, K., Salvatier, J., Dafoe, A., Zhang, B., & Evans, O. (2018). Viewpoint: When Will AI Exceed Human Performance? Evidence from AI Experts. *Journal of Artificial Intelligence Research*, 62, 729–754. <https://doi.org/10.1613/jair.1.11222>
- Grell, G. A., Peckham, S. E., Schmitz, R., McKeen, S. A., Frost, G., Skamarock, W. C., & Eder, B. (2005). Fully coupled "online" chemistry within the WRF model. *Atmospheric Environment*, 39(37), 6957–6975. <https://doi.org/10.1016/j.atmosenv.2005.04.027>
- Greve, P., Warrach-Sagi, K., & Wulfmeyer, V. (2013). Evaluating Soil Water Content in a WRF-Noah Downscaling Experiment. *Journal of Applied Meteorology and Climatology*, 52(10), 2312–2327. <https://doi.org/10.1175/JAMC-D-12-0239.1>
- Grimmond, C. S. B., Best, M., Barlow, J., Arnfield, A. J., Baik, J.-J., Baklanov, A., Belcher, S., Bruse, M., Calmet, I., Chen, F., Clark, P., Dandou, A., Erell, E., Fortuniak, K., Hamdi, R., Kanda, M., Kawai, T., Kondo, H., Krayenhoff, S., ... Williamson, T. (2009). Urban Surface Energy

- Balance Models: Model Characteristics and Methodology for a Comparison Study. In A. Baklanov, G. Sue, M. Alexander, & M. Athanassiadou (Eds.), *Meteorological and Air Quality Models for Urban Areas* (pp. 97–123). Springer Berlin Heidelberg. [https://doi.org/10.1007/978-3-642-00298-4\\_11](https://doi.org/10.1007/978-3-642-00298-4_11)
- Grimmond, C. S. B., Blackett, M., Best, M., Coutts, A., Beringer, J., & Urban Model Comparison Team. (2021). *Phase 2 of the International urban energy balance comparison project—Forcing Data* (Original) [Data set]. Zenodo. <https://doi.org/10.5281/zenodo.4679279>
- Grimmond, C. S. B., Blackett, M., Best, M. J., Baik, J.-J., Belcher, S. E., Beringer, J., Bohnenstengel, S. I., Calmet, I., Chen, F., Coutts, A., Dandou, A., Fortuniak, K., Gouvea, M. L., Hamdi, R., Hendry, M., Kanda, M., Kawai, T., Kawamoto, Y., Kondo, H., ... Zhang, N. (2011). Initial results from Phase 2 of the international urban energy balance model comparison. *International Journal of Climatology*, *31*(2), 244–272. <https://doi.org/10.1002/joc.2227>
- Grimmond, C. S. B., Blackett, M., Best, M. J., Barlow, J., Baik, J.-J., Belcher, S. E., Bohnenstengel, S. I., Calmet, I., Chen, F., Dandou, A., Fortuniak, K., Gouvea, M. L., Hamdi, R., Hendry, M., Kawai, T., Kawamoto, Y., Kondo, H., Krayenhoff, E. S., Lee, S.-H., ... Zhang, N. (2010). The International Urban Energy Balance Models Comparison Project: First Results from Phase 1. *Journal of Applied Meteorology and Climatology*, *49*(6), 1268–1292. <https://doi.org/10.1175/2010JAMC2354.1>
- Grimmond, C. S. B., Blackett, M., Best, M., & Urban Model Comparison Team. (2013). *Phase 2 of the International urban energy balance comparison project—Data analysed—Anonymous* (Anonymous (10/4/21)) [Data set]. Zenodo. <https://doi.org/10.5281/zenodo.4678387>
- Grimmond, C. S. B., Bouchet, V., Molina, L. T., Baklanov, A., Tan, J., Schlünzen, K. H., Mills, G., Golding, B., Masson, V., Ren, C., Voogt, J., Miao, S., Lean, H., Heusinkveld, B., Hovespyan, A., Teruggi, G., Parrish, P., & Joe, P. (2020). Integrated urban hydrometeorological, climate and environmental services: Concept, methodology and key messages. *Urban Climate*, *33*, 100623. <https://doi.org/10.1016/j.uclim.2020.100623>
- Grimmond, C. S. B., & Oke, T. R. (2002). Turbulent Heat Fluxes in Urban Areas: Observations and a Local-Scale Urban Meteorological Parameterization Scheme (LUMPS). *Journal of Applied Meteorology*, *41*, 19. [https://doi.org/10.1175/1520-0450\(2002\)041<0792:THFIUA>2.0.CO;2](https://doi.org/10.1175/1520-0450(2002)041<0792:THFIUA>2.0.CO;2)
- Grimmond, C. S. B., Salmond, A. J., Oke, T. R., Offerle, B., & Lemonsu, A. (2004). Flux and turbulence measurements at a densely built-up site in Marseille: Heat, mass (water and carbon dioxide), and momentum. *Journal of Geophysical Research*, *109*(D24), D24101. <https://doi.org/10.1029/2004JD004936>
- Grüning, B., Chilton, J., Köster, J., Dale, R., Soranzo, N., van den Beek, M., Goecks, J., Backofen, R., Nekrutenko, A., & Taylor, J. (2018). Practical Computational Reproducibility in the Life Sciences. *Cell Systems*, *6*(6), 631–635. <https://doi.org/10.1016/j.cels.2018.03.014>
- Guerrette, J. J., & Henze, D. K. (2015). Development and application of the WRFPLUS-Chem online chemistry adjoint and WRFDA-Chem assimilation system. *Geoscientific Model Development*, *8*(6), 1857–1876. <https://doi.org/10.5194/gmd-8-1857-2015>
- Haiden, T., Sandu, I., Balsamo, G., Arduini, G., & Beljaars, A. (2018). Addressing biases in near-surface forecasts. *ECMWF Newsletter*, *Number 157*. <https://www.ecmwf.int/en/newsletter/157/meteorology/addressing-biases-near-surface-forecasts>
- Hamdi, R., Degrauwe, D., & Termonia, P. (2012). Coupling the Town Energy Balance (TEB) Scheme to an Operational Limited-Area NWP Model: Evaluation for a Highly Urbanized Area in Belgium. *Weather and Forecasting*, *27*(2), 323–344. <https://doi.org/10.1175/WAF-D-11-00064.1>
- Hamdi, R., & Masson, V. (2008). Inclusion of a Drag Approach in the Town Energy Balance (TEB) Scheme: Offline 1D Evaluation in a Street Canyon. *Journal of Applied Meteorology and Climatology*, *47*(10), 2627–2644. <https://doi.org/10.1175/2008JAMC1865.1>

- Hari Prasad, K. B. R. R., Venkata Srinivas, C., Venkateswara Naidu, C., Baskaran, R., & Venkatraman, B. (2016). Assessment of surface layer parameterizations in ARW using micro-meteorological observations from a tropical station: Surface layer parameterization. *Meteorological Applications*, 23(2), 191–208. <https://doi.org/10.1002/met.1545>
- Harman, I. N., Barlow, J. F., & Belcher, S. E. (2004). Scalar Fluxes from Urban Street Canyons Part II: Model. *Boundary-Layer Meteorology*, 113(3), 387–410. <https://doi.org/10.1007/s10546-004-6205-7>
- Harman, I. N., & Belcher, S. E. (2006). The surface energy balance and boundary layer over urban street canyons. *Quarterly Journal of the Royal Meteorological Society*, 132(621), 2749–2768. <https://doi.org/10.1256/qj.05.185>
- Harman, I. N., Best, M. J., & Belcher, S. E. (2004). Radiative Exchange in an Urban Street Canyon. *Boundary-Layer Meteorology*, 110(2), 301–316. <https://doi.org/10.1023/A:1026029822517>
- Hawkins, E., & Sutton, R. (2009). The Potential to Narrow Uncertainty in Regional Climate Predictions. *Bulletin of the American Meteorological Society*, 90(8), 1095–1108. <https://doi.org/10.1175/2009BAMS2607.1>
- Heidinger, A. K., & Cox, S. K. (1996). Finite-cloud effects in longwave radiative transfer. *Journal of the Atmospheric Sciences*, 53(7), 953–963. [https://doi.org/10.1175/1520-0469\(1996\)053<0953:fceilr>2.0.co;2](https://doi.org/10.1175/1520-0469(1996)053<0953:fceilr>2.0.co;2)
- Hellmann, G. (1908). The dawn of meteorology. *Quarterly Journal of the Royal Meteorological Society*, 34(148), 221–232. <https://doi.org/10.1002/qj.49703414802>
- Hersbach, H., Bell, B., Berrisford, P., Hirahara, S., Horányi, A., Muñoz-Sabater, J., Nicolas, J., Peubey, C., Radu, R., Schepers, D., Simmons, A., Soci, C., Abdalla, S., Abellan, X., Balsamo, G., Bechtold, P., Biavati, G., Bidlot, J., Bonavita, M., ... Thépaut, J. (2020). The ERA5 global reanalysis. *Quarterly Journal of the Royal Meteorological Society*, 146(730), 1999–2049. <https://doi.org/10.1002/qj.3803>
- Hertwig, D., Ng, M., Grimmond, S., Vidale, P. L., & McGuire, P. C. (2021). High-resolution global climate simulations: Representation of cities. *International Journal of Climatology*, 41(5), 3266–3285. <https://doi.org/10.1002/joc.7018>
- Hidalgo, J., Dumas, G., Masson, V., Petit, G., Bechtel, B., Bocher, E., Foley, M., Schoetter, R., & Mills, G. (2019). Comparison between local climate zones maps derived from administrative datasets and satellite observations. *Urban Climate*, 27, 64–89. <https://doi.org/10.1016/j.uclim.2018.10.004>
- Hidalgo, J., Masson, V., & Pigeon, G. (2008). Urban-breeze circulation during the CAPITOUL experiment: Numerical simulations. *Meteorology and Atmospheric Physics*, 102(3–4), 243–262. <https://doi.org/10.1007/s00703-008-0345-0>
- Hill, P. G., Manners, J., & Petch, J. C. (2011). Reducing noise associated with the Monte Carlo Independent Column Approximation for weather forecasting models: Reducing McICA Noise for Weather Forecasting. *Quarterly Journal of the Royal Meteorological Society*, 137(654), 219–228. <https://doi.org/10.1002/qj.732>
- Hocking, J., Vidot, J., Brunel, P., Roquet, P., Silveira, B., Turner, E., & Lupu, C. (2021). A new gas absorption optical depth parameterisation for RTTOV version 13. *Geoscientific Model Development*, 14(5), 2899–2915. <https://doi.org/10.5194/gmd-14-2899-2021>
- Hodson, H. (2019, March 1). DeepMind and Google: The battle to control artificial intelligence. *The Economist*. <https://www.economist.com/1843/2019/03/01/deepmind-and-google-the-battle-to-control-artificial-intelligence>
- Hogan, R. J., Ahlgrimm, M., Balsamo, G., Beljaars, A., Berrisford, P., Bozzo, A., Di Giuseppe, F., Di Giuseppe, F., Forbes, R., Haiden, T., Lang, S., Mayer, M., Polichtchouk, I., Sandu, I., Vitart, F., & Wedi, N. (2017). *Radiation in numerical weather prediction* (Research,

Forecast and Copernicus Departments, Paper to the 46th Science Advisory Committee, p. 51). ECMWF.  
<https://www.ecmwf.int/sites/default/files/elibrary/2017/17771-radiation-numerical-weather-prediction.pdf>

- Hogan, R. J., & Bozzo, A. (2018). A Flexible and Efficient Radiation Scheme for the ECMWF Model. *Journal of Advances in Modeling Earth Systems*, 10(8), 1990–2008.  
<https://doi.org/10.1029/2018MS001364>
- Hogan, R. J., Fielding, M. D., Barker, H. W., Villefranque, N., & Schäfer, S. A. K. (2019). Entrapment: An Important Mechanism to Explain the Shortwave 3D Radiative Effect of Clouds. *Journal of the Atmospheric Sciences*, 2019(1), 48–66. <https://doi.org/10.1175/JAS-D-18-0366.1>
- Hogan, R. J., & Matricardi, M. (2020). Evaluating and improving the treatment of gases in radiation schemes: The Correlated K-Distribution Model Intercomparison Project (CKDMIP). *Geoscientific Model Development*, 13(12). <https://doi.org/10.5194/gmd-13-6501-2020>
- Hogan, R. J., Schäfer, S. A. K., Klinger, C., Chiu, J. C., & Mayer, B. (2016). Representing 3-D cloud radiation effects in two-stream schemes: 2. Matrix formulation and broadband evaluation. *Journal of Geophysical Research: Atmospheres*, 121(14), 8583–8599.  
<https://doi.org/10.1002/2016JD024875>
- Hogan, R. J., & Shonk, J. K. P. (2013). Incorporating the Effects of 3D Radiative Transfer in the Presence of Clouds into Two-Stream Multilayer Radiation Schemes. *Journal of the Atmospheric Sciences*, 70(2), 708–724. <https://doi.org/10.1175/JAS-D-12-041.1>
- Holmgren, W. F., Calama-Consulting, Hansen, C., Anderson, K., Mikofski, M., Lorenzo, A., Krien, U., Bmu, Stark, C., DaCoEx, Driesse, A., Jensen, A. R., De León Peque, M. S., Konstant\_t, Mayudong, Heliolytics, Miller, E., Anoma, M. A., Guo, V., ... Dollinger, J. (2021). *pvlib/pvlib-python: V0.9.0* (v0.9.0). Zenodo. <https://doi.org/10.5281/ZENODO.5366883>
- Holmgren, W. F., Hansen, C. W., & Mikofski, M. A. (2018). pvlib python: A python package for modeling solar energy systems. *Journal of Open Source Software*, 3(29), 884.  
<https://doi.org/10.21105/joss.00884>
- Hong, S.-Y., Dudhia, J., & Chen, S.-H. (2004). A Revised Approach to Ice Microphysical Processes for the Bulk Parameterization of Clouds and Precipitation. *Monthly Weather Review*, 132, 18. [https://doi.org/10.1175/1520-0493\(2004\)132<0103:ARATIM>2.0.CO;2](https://doi.org/10.1175/1520-0493(2004)132<0103:ARATIM>2.0.CO;2)
- Hong, S.-Y., & Lim, J.-O. J. (2006). The WRF Single-Moment 6-Class Microphysics Scheme (WSM6). *Journal of the Korean Meteorological Society*, 42(2), 129–151.
- Hong, S.-Y., Noh, Y., & Dudhia, J. (2006). A New Vertical Diffusion Package with an Explicit Treatment of Entrainment Processes. *Monthly Weather Review*, 134(9), 2318–2341.  
<https://doi.org/10.1175/MWR3199.1>
- Hu, X.-M., Klein, P. M., & Xue, M. (2013). Evaluation of the updated YSU planetary boundary layer scheme within WRF for wind resource and air quality assessments. *Journal of Geophysical Research: Atmospheres*, 118(18), 10,490–10,505.  
<https://doi.org/10.1002/jgrd.50823>
- Hu, X.-M., Nielsen-Gammon, J. W., & Zhang, F. (2010). Evaluation of Three Planetary Boundary Layer Schemes in the WRF Model. *Journal of Applied Meteorology and Climatology*, 49(9), 1831–1844. <https://doi.org/10.1175/2010JAMC2432.1>
- Huang, X.-Y., Xiao, Q., Barker, D. M., Zhang, X., Michalakes, J., Huang, W., Henderson, T., Bray, J., Chen, Y., Ma, Z., Dudhia, J., Guo, Y., Zhang, X., Won, D.-J., Lin, H.-C., & Kuo, Y.-H. (2009). Four-Dimensional Variational Data Assimilation for WRF: Formulation and Preliminary Results. *Monthly Weather Review*, 137(1), 299–314.  
<https://doi.org/10.1175/2008MWR2577.1>

- Huntingford, C., Jeffers, E. S., Bonsall, M. B., Christensen, H. M., Lees, T., & Yang, H. (2019). Machine learning and artificial intelligence to aid climate change research and preparedness. *Environmental Research Letters*, *14*(12), 124007. <https://doi.org/10.1088/1748-9326/ab4e55>
- Iacono, M. J., Delamere, J. S., Mlawer, E. J., Shephard, M. W., Clough, S. A., & Collins, W. D. (2008). Radiative forcing by long-lived greenhouse gases: Calculations with the AER radiative transfer models. *Journal of Geophysical Research*, *113*(D13), D13103. <https://doi.org/10.1029/2008JD009944>
- IBM. (2021, June 18). *AI vs. Machine Learning vs. Deep Learning vs. Neural Networks: What's the Difference?* <https://www.ibm.com/cloud/blog/ai-vs-machine-learning-vs-deep-learning-vs-neural-networks>
- INSEE. (2016). *Populations légales 2016—Commune de Toulouse (31555)*. <https://www.insee.fr/fr/statistiques/3681328?geo=COM-31555>
- Jakub, F., & Mayer, B. (2015). A three-dimensional parallel radiative transfer model for atmospheric heating rates for use in cloud resolving models—The TenStream solver. *Journal of Quantitative Spectroscopy and Radiative Transfer*, *163*, 63–71. <https://doi.org/10.1016/j.jqsrt.2015.05.003>
- Janjic, Z. I. (2003). A nonhydrostatic model based on a new approach. *Meteorology and Atmospheric Physics*. <https://doi.org/10.1007/s00703-001-0587-6>
- Janjic, Z. I., Jr, J. P. G., & Nickovic, S. (2001). An Alternative Approach to Nonhydrostatic Modeling. *Monthly Weather Review*, *129*, 15. [https://doi.org/10.1175/1520-0493\(2001\)129<1164:AAATNM>2.0.CO;2](https://doi.org/10.1175/1520-0493(2001)129<1164:AAATNM>2.0.CO;2)
- Jiménez, P. A., Dudhia, J., González-Rouco, J. F., Navarro, J., Montávez, J. P., & García-Bustamante, E. (2012). A Revised Scheme for the WRF Surface Layer Formulation. *Monthly Weather Review*, *140*(3), 898–918. <https://doi.org/10.1175/MWR-D-11-00056.1>
- Joe, H. (2014). *Dependence Modeling with Copulas* (1st Edition). Chapman and Hall/CRC. <https://doi.org/10.1201/b17116>
- Joly, D., Brossard, T., Cardot, H., Cavailles, J., Hilal, M., & Wavresky, P. (2010). Les types de climats en France, une construction spatiale. *Cybergeo*. <https://doi.org/10.4000/cybergeo.23155>
- Kanda, M., Kawai, T., Kanega, M., Moriwaki, R., Narita, K., & Hagishima, A. (2005). A Simple Energy Balance Model for Regular Building Arrays. *Boundary-Layer Meteorology*, *116*(3), 423–443. <https://doi.org/10.1007/s10546-004-7956-x>
- Kanda, M., Kawai, T., & Nakagawa, K. (2005). A Simple Theoretical Radiation Scheme for Regular Building Arrays. *Boundary-Layer Meteorology*, *114*(1), 71–90. <https://doi.org/10.1007/s10546-004-8662-4>
- Katz, R. W., & Murphy, A. H. (Eds.). (1997). *Economic value of weather and climate forecasts* (1. publ). Cambridge Univ. Press.
- Kawai, T., Kanda, M., Narita, K., & Hagishima, A. (2007). Validation of a numerical model for urban energy-exchange using outdoor scale-model measurements. *International Journal of Climatology*, *27*(14), 1931–1942. <https://doi.org/10.1002/joc.1624>
- Kawai, T., Ridwan, M. K., & Kanda, M. (2009). Evaluation of the Simple Urban Energy Balance Model Using Selected Data from 1-yr Flux Observations at Two Cities. *Journal of Applied Meteorology and Climatology*, *48*(4), 693–715. <https://doi.org/10.1175/2008JAMC1891.1>
- Kawamoto, Y., & Ooka, R. (2006). Analysis of the radiation field at pedestrian level using a meso-scale meteorological model incorporating the urban canopy model. *Sixth International Conference on Urban Climate*, 446–449.

- Kawamoto, Y., & Ooka, R. (2009a). Accuracy validation of urban climate analysis model using MM5 incorporating a multi-layer urban canopy model. *Seventh International Conference on Urban Climate*, 4.
- Kawamoto, Y., & Ooka, R. (2009b). Incorporating an urban canopy model to represent the effects of buildings: Development of urban climate analysis model using MM5 Part 2. *Journal of Environmental Engineering (Transactions of AIJ)*, 74(642), 1009–1018. <https://doi.org/10.3130/aije.74.1009>
- Kelso, N. V., & Patterson, T. (2009). Natural Earth Vector. *Cartographic Perspectives*, 64, 45–50. <https://doi.org/10.14714/CP64.148>
- Kikegawa, Y., Tanaka, A., Ohashi, Y., Ihara, T., & Shigeta, Y. (2014). Observed and simulated sensitivities of summertime urban surface air temperatures to anthropogenic heat in downtown areas of two Japanese Major Cities, Tokyo and Osaka. *Theoretical and Applied Climatology*, 117(1–2), 175–193. <https://doi.org/10.1007/s00704-013-0996-8>
- Kingma, D. P., & Ba, J. (2015). *Adam: A Method for Stochastic Optimization*. 3rd International Conference on Learning Representations (ICLR), San Diego, CA, USA. <https://arxiv.org/abs/1412.6980>
- Kitware Inc. (2019a). *CMake*. <https://cmake.org/>
- Kitware Inc. (2019b). *ExternalProject—CMake 3.15.3 Documentation*. <https://cmake.org/help/latest/module/ExternalProject.html>
- Kljun, N., Calanca, P., Rotach, M. W., & Schmid, H. P. (2015). A simple two-dimensional parameterisation for Flux Footprint Prediction (FFP). *Geoscientific Model Development*, 8(11), 3695–3713. <https://doi.org/10.5194/gmd-8-3695-2015>
- Knutti, R., & Sedláček, J. (2013). Robustness and uncertainties in the new CMIP5 climate model projections. *Nature Climate Change*, 3(4), 369–373. <https://doi.org/10.1038/nclimate1716>
- Kondo, H., Genchi, Y., Kikegawa, Y., Ohashi, Y., Yoshikado, H., & Komiyama, H. (2005). Development of a Multi-Layer Urban Canopy Model for the Analysis of Energy Consumption in a Big City: Structure of the Urban Canopy Model and its Basic Performance. *Boundary-Layer Meteorology*, 116(3), 395–421. <https://doi.org/10.1007/s10546-005-0905-5>
- Kondo, H., & Liu, F.-H. (1998). A study on the urban thermal environment obtained through one-dimensional urban canopy model. *Journal of Japan Society for Atmospheric Environment*, 33(3), 179–192. [https://doi.org/10.11298/taiki1995.33.3\\_179](https://doi.org/10.11298/taiki1995.33.3_179)
- Krasnopolsky, V. M., Chalikov, D. V., & Tolman, H. L. (2002). A neural network technique to improve computational efficiency of numerical oceanic models. *Ocean Modelling*, 21. [https://doi.org/10.1016/S1463-5003\(02\)00010-0](https://doi.org/10.1016/S1463-5003(02)00010-0)
- Krasnopolsky, V. M., Fox-Rabinovitz, M. S., & Belochitski, A. A. (2008). Decadal Climate Simulations Using Accurate and Fast Neural Network Emulation of Full, Longwave and Shortwave, Radiation. *Monthly Weather Review*, 136(10), 3683–3695. <https://doi.org/10.1175/2008MWR2385.1>
- Krasnopolsky, V. M., Fox-Rabinovitz, M. S., & Belochitski, A. A. (2013). Using Ensemble of Neural Networks to Learn Stochastic Convection Parameterizations for Climate and Numerical Weather Prediction Models from Data Simulated by a Cloud Resolving Model. *Advances in Artificial Neural Systems*, 2013, 1–13. <https://doi.org/10.1155/2013/485913>
- Krasnopolsky, V. M., Fox-Rabinovitz, M. S., & Chalikov, D. V. (2005). New Approach to Calculation of Atmospheric Model Physics: Accurate and Fast Neural Network Emulation of Longwave Radiation in a Climate Model. *Monthly Weather Review*, 133(5), 1370–1383. <https://doi.org/10.1175/MWR2923.1>

- Krasnopolsky, V. M., & Lin, Y. (2012). A Neural Network Nonlinear Multimodel Ensemble to Improve Precipitation Forecasts over Continental US. *Advances in Meteorology*, 2012, 1–11. <https://doi.org/10.1155/2012/649450>
- Krayenhoff, E. S., & Voogt, J. A. (2007). A microscale three-dimensional urban energy balance model for studying surface temperatures. *Boundary-Layer Meteorology*, 123(3), 433–461. <https://doi.org/10.1007/s10546-006-9153-6>
- Kurtzer, G. M., Sochat, V., & Bauer, M. W. (2017). Singularity: Scientific containers for mobility of compute. *PLOS ONE*, 12(5), e0177459. <https://doi.org/10.1371/journal.pone.0177459>
- Kusaka, H., Kondo, H., Kikegawa, Y., & Kimura, F. (2001). A Simple Single-Layer Urban Canopy Model For Atmospheric Models: Comparison With Multi-Layer And Slab Models. *Boundary-Layer Meteorology*, 101(3), 329–358. <https://doi.org/10.1023/A:1019207923078>
- Lamb, H. H. (1995). *Climate, History and the Modern World* (0 ed.). Routledge. <https://doi.org/10.4324/9780203433652>
- Larraondo, P. R. (2019). *Application of machine learning techniques to weather forecasting* [Thesis, University of the Basque Country UPV/EHU]. <https://hdl.handle.net/20.500.11824/1141>
- Lawrence, B. N., Rezný, M., Budich, R., Bauer, P., Behrens, J., Carter, M., Deconinck, W., Ford, R., Maynard, C., Müllerworth, S., Osuna, C., Porter, A., Serradell, K., Valcke, S., Wedi, N., & Wilson, S. (2018). Crossing the chasm: How to develop weather and climate models for next generation computers? *Geoscientific Model Development*, 11(5), 1799–1821. <https://doi.org/10.5194/gmd-11-1799-2018>
- Le Moigne, P., Albergel, C., Boone, A., Belamari, S., Brun, E., Calvet, J.-C., Decharme, B., & Masson, V. (2018). *Surfex Scientific Documentation* (P. Le Moigne, Ed.). [http://www.umr-cnrm.fr/surfex/IMG/pdf/surfex\\_scidoc\\_v8.1.pdf](http://www.umr-cnrm.fr/surfex/IMG/pdf/surfex_scidoc_v8.1.pdf)
- Lee, S.-H., & Park, S.-U. (2007). A Vegetated Urban Canopy Model for Meteorological and Environmental Modelling. *Boundary-Layer Meteorology*, 126(1), 73–102. <https://doi.org/10.1007/s10546-007-9221-6>
- Lemonsu, A., Belair, S., & Mailhot, J. (2009). The New Canadian Urban Modelling System: Evaluation for Two Cases from the Joint Urban 2003 Oklahoma City Experiment. *Boundary-Layer Meteorology*, 133(1), 47–70. <https://doi.org/10.1007/s10546-009-9414-2>
- Lemonsu, A., Grimmond, C. S. B., & Masson, V. (2004). Modeling the Surface Energy Balance of the Core of an Old Mediterranean City: Marseille. *Journal of Applied Meteorology*, 43, 16. [https://doi.org/10.1175/1520-0450\(2004\)043<0312:MTSEBO>2.0.CO;2](https://doi.org/10.1175/1520-0450(2004)043<0312:MTSEBO>2.0.CO;2)
- Lemonsu, A., & Masson, V. (2002). Simulation of a Summer Urban Breeze Over Paris. *Boundary-Layer Meteorology*, 104(3), 463–490. <https://doi.org/10.1023/A:1016509614936>
- Lemonsu, A., Masson, V., Shashua-Bar, L., Erell, E., & Pearlmutter, D. (2012). Inclusion of vegetation in the Town Energy Balance model for modelling urban green areas. *Geoscientific Model Development*, 5(6), 1377–1393. <https://doi.org/10.5194/gmd-5-1377-2012>
- Li, L., Jamieson, K., DeSalvo, G., Rostamizadeh, A., & Talwalkar, A. (2018). Hyperband: A novel bandit-based approach to hyperparameter optimization. *Journal of Machine Learning Research*, 18(185), 1–52. <http://jmlr.org/papers/v18/16-558.html>
- Lin, C.-Y., Chen, F., Huang, J. C., Chen, W.-C., Liou, Y.-A., Chen, W.-N., & Liu, S.-C. (2008). Urban heat island effect and its impact on boundary layer development and land-sea circulation over northern Taiwan. *Atmospheric Environment*, 42(22), 5635–5649. <https://doi.org/10.1016/j.atmosenv.2008.03.015>
- Lipson, M., Grimmond, S., & Best, M. (2020). Calling for participants: A new multi-site evaluation project for modelling in urban areas. *Urban Climate News: Quarterly*

*Newsletter of the IAUC*, 75, 15–16. <https://www.urban-climate.org/wp-content/uploads/IAUC075.pdf>

- Liu, J., & Niyogi, D. (2019). Meta-analysis of urbanization impact on rainfall modification. *Scientific Reports*, 9(1), 7301. <https://doi.org/10.1038/s41598-019-42494-2>
- López-Pintado, S., & Romo, J. (2009). On the Concept of Depth for Functional Data. *Journal of the American Statistical Association*, 104(486), 718–734. <https://doi.org/10.1198/jasa.2009.0108>
- Lorenz, E. N. (2006). Reflections on the Conception, Birth, and Childhood of Numerical Weather Prediction. *Annual Review of Earth and Planetary Sciences*, 34(1), 37–45. <https://doi.org/10.1146/annurev.earth.34.083105.102317>
- Loridan, T., Grimmond, C. S. B., Grossman-Clarke, S., Chen, F., Tewari, M., Manning, K., Martilli, A., Kusaka, H., & Best, M. (2010). Trade-offs and responsiveness of the single-layer urban canopy parametrization in WRF: An offline evaluation using the MOSCEM optimization algorithm and field observations. *Quarterly Journal of the Royal Meteorological Society*, 136(649), 997–1019. <https://doi.org/10.1002/qj.614>
- Loridan, T., Grimmond, C. S. B., Offerle, B. D., Young, D. T., Smith, T. E. L., Järvi, L., & Lindberg, F. (2011). Local-Scale Urban Meteorological Parameterization Scheme (LUMPS): Longwave Radiation Parameterization and Seasonality-Related Developments. *Journal of Applied Meteorology and Climatology*, 50(1), 185–202. <https://doi.org/10.1175/2010JAMC2474.1>
- Lynch, P. (2014). *The emergence of numerical weather prediction: Richardson's dream* (1. paperback ed). Cambridge Univ. Press.
- Martilli, A. (2002). Numerical Study of Urban Impact on Boundary Layer Structure: Sensitivity to Wind Speed, Urban Morphology, and Rural Soil Moisture. *Journal of Applied Meteorology*, 41, 20. [https://doi.org/10.1175/1520-0450\(2002\)041<1247:NSOUIO>2.0.CO;2](https://doi.org/10.1175/1520-0450(2002)041<1247:NSOUIO>2.0.CO;2)
- Martilli, A., Clappier, A., & Rotach, M. W. (2002). An Urban Surface Exchange Parameterisation for Mesoscale Models. *Boundary-Layer Meteorology*, 104(2), 261–304. <https://doi.org/10.1023/A:1016099921195>
- Martin, G. M., Milton, S. F., Senior, C. A., Brooks, M. E., Ineson, S., Reichler, T., & Kim, J. (2010). Analysis and Reduction of Systematic Errors through a Seamless Approach to Modeling Weather and Climate. *Journal of Climate*, 23(22), 5933–5957. <https://doi.org/10.1175/2010JCLI3541.1>
- Masson, V. (2000). A Physically-Based Scheme For The Urban Energy Budget In Atmospheric Models. *Boundary-Layer Meteorology*, 94(3), 357–397. <https://doi.org/10.1023/A:1002463829265>
- Masson, V. (2006). Urban surface modeling and the meso-scale impact of cities. *Theoretical and Applied Climatology*, 84(1–3), 35–45. <https://doi.org/10.1007/s00704-005-0142-3>
- Masson, V., Bonhomme, M., Salagnac, J.-L., Briottet, X., & Lemonsu, A. (2014). Solar panels reduce both global warming and urban heat island. *Frontiers in Environmental Science*, 2. <https://doi.org/10.3389/fenvs.2014.00014>
- Masson, V., Gomes, L., Pigeon, G., Lioussé, C., Pont, V., Lagouarde, J.-P., Voogt, J., Salmond, J., Oke, T. R., Hidalgo, J., Legain, D., Garrouste, O., Lac, C., Connan, O., Briottet, X., Lachérade, S., & Tulet, P. (2008). The Canopy and Aerosol Particles Interactions in TOulouse Urban Layer (CAPITOU) experiment. *Meteorology and Atmospheric Physics*, 102(3–4), 135–157. <https://doi.org/10.1007/s00703-008-0289-4>
- Masson, V., Grimmond, C. S. B., & Oke, T. R. (2002). Evaluation of the Town Energy Balance (TEB) Scheme with Direct Measurements from Dry Districts in Two Cities. *Journal of Applied Meteorology*, 41, 16. [https://doi.org/10.1175/1520-0450\(2002\)041<1011:EOTTEB>2.0.CO;2](https://doi.org/10.1175/1520-0450(2002)041<1011:EOTTEB>2.0.CO;2)

- Masson, V., Lemonsu, A., Pigeon, G., Schoetter, R., de Munck, C., Bueno, B., Faroux, S., Goret, M., Redon, E., Chancibault, K., Stavropoulos-Laffaille, X., Leroyer, S., & Meyer, D. (2020). *The Town Energy Balance (TEB) model* (4.0.1). Zenodo. <https://doi.org/10.5281/zenodo.3894784>
- Masson, V., Lemonsu, A., Pigeon, G., Schoetter, R., de Munck, C., Bueno, B., Faroux, S., Goret, M., Redon, E., Chancibault, K., Stavropoulos-Laffaille, X., Leroyer, S., & Meyer, D. (2021). *The Town Energy Balance (TEB) model* (4.1.2). Zenodo. <https://doi.org/10.5281/zenodo.5775962>
- Maunder, W. J. (2019). *The Value of the Weather* (1st ed.). Routledge. <https://doi.org/10.4324/9780429344916>
- May, R. M., Arms, S. C., Marsh, P., Bruning, E., Leeman, J. R., Goebbert, K., Thielen, J. E., Bruick, Z. S., & Camron, M. Drew. (2021). *MetPy: A Python package for meteorological data* (1.1.0). <https://doi.org/10.5065/D6WW7G29>
- McMillen, R. T. (1988). An eddy correlation technique with extended applicability to non-simple terrain. *Boundary-Layer Meteorology*, 43(3), 231–245. <https://doi.org/10.1007/BF00128405>
- Meyer, D. (2020). *Data archive for paper “WRF-TEB: Implementation and Evaluation of the Coupled Weather Research and Forecasting (WRF) and Town Energy Balance (TEB) Model”* (1.0.0) [Data set]. Zenodo. <https://doi.org/10.5281/zenodo.3554517>
- Meyer, D. (2021a). *Data archive for paper “Copula-based synthetic data augmentation for machine learning-emulators”* (1.2.0) [Data set]. Zenodo. <https://doi.org/10.5281/zenodo.5150327>
- Meyer, D. (2021b). *Data archive for paper “Machine Learning Emulation of Urban Land Surface Processes”* (1.0.0) [Data set]. Zenodo. <https://doi.org/10.5281/zenodo.5142960>
- Meyer, D. (2021c). *Data archive for paper “Machine Learning Emulation of 3D Cloud Radiative Effects”* (1.1.0) [Data set]. Zenodo. <https://doi.org/10.5281/zenodo.5113055>
- Meyer, D., Grimmond, S., Dueben, P., Hogan, R., & van Reeuwijk, M. (2022). Machine Learning Emulation of Urban Land Surface Processes. *Journal of Advances in Modeling Earth Systems*, 14(3). <https://doi.org/10.1029/2021MS002744>
- Meyer, D., Hogan, R. J., Dueben, P. D., & Mason, S. L. (2022). Machine Learning Emulation of 3D Cloud Radiative Effects. *Journal of Advances in Modeling Earth Systems*, 14(3). <https://doi.org/10.1029/2021MS002550>
- Meyer, D., & Nagler, T. (2020). *Synthia: Multidimensional synthetic data generation in Python* (0.3.0). Zenodo. <https://doi.org/10.5281/zenodo.5150200>
- Meyer, D., & Nagler, T. (2021). Synthia: Multidimensional synthetic data generation in Python. *Journal of Open Source Software*, 6(65), 2863. <https://doi.org/10.21105/joss.02863>
- Meyer, D., Nagler, T., & Hogan, R. J. (2021). Copula-based synthetic data augmentation for machine-learning emulators. *Geoscientific Model Development*, 14(8), 5205–5215. <https://doi.org/10.5194/gmd-14-5205-2021>
- Meyer, D., & Raustad, R. (2020). MinimalDX. Zenodo. <https://doi.org/10.5281/zenodo.3892452>
- Meyer, D., & Riechert, M. (2019a). Open source QGIS toolkit for the Advanced Research WRF modelling system. *Environmental Modelling & Software*, 112, 166–178. <https://doi.org/10.1016/j.envsoft.2018.10.018>
- Meyer, D., & Riechert, M. (2019b). *The GIS4WRF Plugin* (0.14.2). Zenodo. <https://doi.org/10.5281/zenodo.3593935>
- Meyer, D., & Riechert, M. (2020). *The GIS4WRF Plugin* (0.14.4). Zenodo. <https://doi.org/10.5281/zenodo.4035877>

- Meyer, D., Schoetter, R., Masson, V., & Grimmond, S. (2020). Enhanced software and platform for the Town Energy Balance (TEB) model. *Journal of Open Source Software*, 5(50), 2008. <https://doi.org/10.21105/joss.02008>
- Meyer, D., Schoetter, R., Riechert, M., Verrelle, A., Tewari, M., Dudhia, J., Masson, V., van Reeuwijk, M., & Grimmond, S. (2020). WRF-TEB: Implementation and Evaluation of the Coupled Weather Research and Forecasting (WRF) and Town Energy Balance (TEB) Model. *Journal of Advances in Modeling Earth Systems*, 12(8). <https://doi.org/10.1029/2019MS001961>
- Meyer, D., & Thevenard, D. (2019). PsychroLib: A library of psychrometric functions to calculate thermodynamic properties of air. *Journal of Open Source Software*, 4(33), 1137. <https://doi.org/10.21105/joss.01137>
- Meyer, D., & Thevenard, D. (2020). *PsychroLib: A library of psychrometric functions to calculate thermodynamic properties of air* (2.5.0). Zenodo. <https://doi.org/10.5281/zenodo.3748874>
- Mills, G. (1997). An urban canopy-layer climate model. *Theoretical and Applied Climatology*, 57(3–4), 229–244. <https://doi.org/10.1007/BF00863615>
- Mitchell, T. M. (1997). *Machine Learning*. McGraw-Hill.
- Mlawer, E. J., Taubman, S. J., Brown, P. D., Iacono, M. J., & Clough, S. A. (1997). Radiative transfer for inhomogeneous atmospheres: RRTM, a validated correlated-k model for the longwave. *Journal of Geophysical Research: Atmospheres*, 102(D14), 16663–16682. <https://doi.org/10.1029/97JD00237>
- Monmonier, M. S. (1999). *Air apparent: How meteorologists learned to map, predict, and dramatize weather*. Univ. of Chicago Press.
- Moonen, P., Defraeye, T., Dorer, V., Blocken, B., & Carmeliet, J. (2012). Urban Physics: Effect of the micro-climate on comfort, health and energy demand. *Frontiers of Architectural Research*, 1(3), 197–228. <https://doi.org/10.1016/j.foar.2012.05.002>
- Morcrette, J.-J., Mozdzyński, G., & Leutbecher, M. (2008). A Reduced Radiation Grid for the ECMWF Integrated Forecasting System. *Monthly Weather Review*, 136, 13. <https://doi.org/10.1175/2008MWR2590.1>
- Mozdzyński, G., Hamrud, M., & Wedi, N. (2015). A Partitioned Global Address Space implementation of the European Centre for Medium Range Weather Forecasts Integrated Forecasting System. *The International Journal of High Performance Computing Applications*, 29(3), 261–273. <https://doi.org/10.1177/1094342015576773>
- Müller, V. C., & Bostrom, N. (2016). Future Progress in Artificial Intelligence: A Survey of Expert Opinion. In V. C. Müller (Ed.), *Fundamental Issues of Artificial Intelligence* (Vol. 376, pp. 555–572). Springer International Publishing. [https://doi.org/10.1007/978-3-319-26485-1\\_33](https://doi.org/10.1007/978-3-319-26485-1_33)
- Nagler, T., Schellhase, C., & Czado, C. (2017). Nonparametric estimation of simplified vine copula models: Comparison of methods. *Dependence Modeling*, 5(1), 99–120. <https://doi.org/10.ghfn5m>
- Nagler, T., & Vatter, T. (2020). *Pyvinecopulib (v0.5.5)* (v0.5.5). Zenodo. <https://doi.org/10.5281/zenodo.4288293>
- Natural Earth. (2019). *Natural Earth*. <https://www.naturalearthdata.com/>
- Noh, Y., Cheon, W. G., Hong, S. Y., & Raasch, S. (2003). Improvement of the K-profile Model for the Planetary Boundary Layer based on Large Eddy Simulation Data. *Boundary-Layer Meteorology*, 107(2), 401–427. <https://doi.org/10.1023/A:1022146015946>
- Noilhan, J., & Planton, S. (1989). A Simple Parameterization of Land Surface Processes for Meteorological Models. *Monthly Weather Review*. [https://doi.org/10.1175/1520-0493\(1989\)117<0536:ASPOLS>2.0.CO;2](https://doi.org/10.1175/1520-0493(1989)117<0536:ASPOLS>2.0.CO;2)

- Northwest Missouri State University. (2022). *Jean and the ENIAC | Jean Jennings Bartik Computing Museum | Northwest Missouri State University*.  
<https://www.nwmissouri.edu/archives/computing/albums/Jean-ENIACTEAM/index.htm>
- Nowack, P., Braesicke, P., Haigh, J., Abraham, N. L., Pyle, J., & Voulgarakis, A. (2018). Using machine learning to build temperature-based ozone parameterizations for climate sensitivity simulations. *Environmental Research Letters*, 13(10), 104016.  
<https://doi.org/10.1088/1748-9326/aae2be>
- Offerle, B., Grimmond, C. S. B., & Oke, T. R. (2003). Parameterization of Net All-Wave Radiation for Urban Areas. *Journal of Applied Meteorology*, 42, 17. [https://doi.org/10.1175/1520-0450\(2003\)042<1157:PONARF>2.0.CO;2](https://doi.org/10.1175/1520-0450(2003)042<1157:PONARF>2.0.CO;2)
- O’Gorman, P. A., & Dwyer, J. G. (2018). Using Machine Learning to Parameterize Moist Convection: Potential for Modeling of Climate, Climate Change, and Extreme Events. *Journal of Advances in Modeling Earth Systems*, 10(10), 2548–2563.  
<https://doi.org/10.1029/2018MS001351>
- Oke, T. R., Mills, G., Christen, A., & Voogt, J. A. (2017). *Urban Climates*. Cambridge University Press. <https://doi.org/10.1017/9781139016476>
- Oleson, K. W., Bonan, G. B., Feddema, J., & Jackson, T. (2011). An examination of urban heat island characteristics in a global climate model: HEAT ISLAND CHARACTERISTICS IN A GLOBAL CLIMATE MODEL. *International Journal of Climatology*, 31(12), 1848–1865.  
<https://doi.org/10.1002/joc.2201>
- Oleson, K. W., Bonan, G. B., Feddema, J., & Vertenstein, M. (2008). An Urban Parameterization for a Global Climate Model. Part II: Sensitivity to Input Parameters and the Simulated Urban Heat Island in Offline Simulations. *Journal of Applied Meteorology and Climatology*, 47(4), 1061–1076. <https://doi.org/10.1175/2007JAMC1598.1>
- Oleson, K. W., Bonan, G. B., Feddema, J., Vertenstein, M., & Grimmond, C. S. B. (2008). An Urban Parameterization for a Global Climate Model. Part I: Formulation and Evaluation for Two Cities. *Journal of Applied Meteorology and Climatology*, 47(4), 1038–1060.  
<https://doi.org/10.1175/2007JAMC1597.1>
- O’Malley, T., Bursztein, E., Long, J., Chollet, F., Jin, H., Invernizzi, L., & others. (2019). *KerasTuner*.  
<https://github.com/keras-team/keras-tuner>
- OpenStreetMap contributors. (2017). *Planet dump retrieved from https://planet.osm.org*.  
<https://www.openstreetmap.org>
- Pal, A., Mahajan, S., & Norman, M. R. (2019). Using Deep Neural Networks as Cost-Effective Surrogate Models for Super-Parameterized E3SM Radiative Transfer. *Geophysical Research Letters*, 46(11), 6069–6079. <https://doi.org/10.1029/2018GL081646>
- Patki, N., Wedge, R., & Veeramachaneni, K. (2016). The Synthetic Data Vault. 2016 *IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, 399–410.  
<https://doi.org/10.1109/DSAA.2016.49>
- Petty, G. W. (2006). *A first course in atmospheric radiation* (2nd ed). Sundog Pub.
- Pfister, C., Brázdil, R., Glaser, R., Bokwa, A., Holawe, F., Limanowka, D., Kotyza, O., Munzar, J., Rácz, L., Strömmer, E., & Schwarz-Zanetti, G. (1999). Daily Weather Observations in Sixteenth-Century Europe. *Climatic Change*, 43(1), 111–150.  
<https://doi.org/10.1023/A:1005505113244>
- Pigeon, G., Legain, D., Durand, P., & Masson, V. (2007). Anthropogenic heat release in an old European agglomeration (Toulouse, France). *International Journal of Climatology*, 27(14), 1969–1981. <https://doi.org/10.1002/joc.1530>
- Pigeon, G., Moscicki, M. A., Voogt, J. A., & Masson, V. (2008). Simulation of fall and winter surface energy balance over a dense urban area using the TEB scheme. *Meteorology*

and Atmospheric Physics, 102(3–4), 159–171. <https://doi.org/10.1007/s00703-008-0320-9>

- Pincus, R., Barker, H. W., & Morcrette, J.-J. (2003). A fast, flexible, approximate technique for computing radiative transfer in inhomogeneous cloud fields: FAST, FLEXIBLE, APPROXIMATE RADIATIVE TRANSFER. *Journal of Geophysical Research: Atmospheres*, 108(D13), n/a-n/a. <https://doi.org/10.1029/2002JD003322>
- Pincus, R., Hannay, C., & Evans, K. F. (2005). The Accuracy of Determining Three-Dimensional Radiative Transfer Effects in Cumulus Clouds Using Ground-Based Profiling Instruments. *Journal of the Atmospheric Sciences*, 62(7), 2284–2293. <https://doi.org/10.1175/JAS3464.1>
- Pincus, R., Mlawer, E. J., & Delamere, J. S. (2019). Balancing Accuracy, Efficiency, and Flexibility in Radiation Calculations for Dynamical Models. *Journal of Advances in Modeling Earth Systems*, 11(10), 3074–3089. <https://doi.org/10.1029/2019MS001621>
- Porson, A., Clark, P. A., Harman, I. N., Best, M. J., & Belcher, S. E. (2010). Implementation of a new urban energy budget scheme into MetUM. Part II: Validation against observations and model intercomparison. *Quarterly Journal of the Royal Meteorological Society*, 136(651), 1530–1542. <https://doi.org/10.1002/qj.572>
- Powers, J. G., Klemp, J. B., Skamarock, W. C., Davis, C. A., Dudhia, J., Gill, D. O., Coen, J. L., Gochis, D. J., Ahmadov, R., Peckham, S. E., Grell, G. A., Michalakes, J., Trahan, S., Benjamin, S. G., Alexander, C. R., Dimego, G. J., Wang, W., Schwartz, C. S., Romine, G. S., ... Duda, M. G. (2017). The Weather Research and Forecasting Model: Overview, System Efforts, and Future Directions. *Bulletin of the American Meteorological Society*, 98(8), 1717–1737. <https://doi.org/10.1175/BAMS-D-15-00308.1>
- QGIS Development Team. (2019). *QGIS Geographic Information System. Open Source Geospatial Foundation Project*. <https://qgis.osgeo.org/>
- Randall, D. A., Bitz, C. M., Danabasoglu, G., Denning, A. S., Gent, P. R., Gettelman, A., Griffies, S. M., Lynch, P., Morrison, H., Pincus, R., & Thuburn, J. (2019). 100 Years of Earth System Model Development. *Meteorological Monographs*, 59, 12.1-12.66. <https://doi.org/10.1175/AMSMONOGRAPHS-D-18-0018.1>
- Rasp, S. (2019). *Statistical methods and machine learning in weather and climate modeling* [Thesis, Ludwig-Maximilians-Universität]. <https://doi.org/10.5282/edoc.23867>
- Rasp, S., & Lerch, S. (2018). Neural Networks for Postprocessing Ensemble Weather Forecasts. *Monthly Weather Review*, 146(11), 3885–3900. <https://doi.org/10.1175/MWR-D-18-0187.1>
- Rasp, S., Pritchard, M. S., & Gentine, P. (2018). Deep learning to represent subgrid processes in climate models. *Proceedings of the National Academy of Sciences*, 115(39), 9684–9689. <https://doi.org/10.1073/pnas.1810286115>
- Redish, A. D., Kummerfeld, E., Morris, R. L., & Love, A. C. (2018). Opinion: Reproducibility failures are essential to scientific inquiry. *Proceedings of the National Academy of Sciences*, 115(20), 5042–5046. <https://doi.org/10.1073/pnas.1806370115>
- Reichstein, M., Camps-Valls, G., Stevens, B., Jung, M., Denzler, J., Carvalhais, N., & Prabhat. (2019). Deep learning and process understanding for data-driven Earth system science. *Nature*, 566(7743), 195–204. <https://doi.org/10.1038/s41586-019-0912-1>
- Richardson, L. F. (1922). *Weather prediction by numerical process*. Cambridge, The University press. <http://archive.org/details/weatherpredictio00richrich>
- Riechert, M., & Meyer, D. (2019a). *WPS-CMake (Version 4.1.0) (WPS-CMake-4.1.0)* [Computer software]. Zenodo. <https://doi.org/10.5281/zenodo.3407075>

- Riechert, M., & Meyer, D. (2019b). WRF-CMake: Integrating CMake support into the Advanced Research WRF (ARW) modelling system. *Journal of Open Source Software*, 4(41), 1468. <https://doi.org/10.21105/joss.01468>
- Riechert, M., & Meyer, D. (2019c). *WPS-CMake* (WPS-CMake-4.1.0). Zenodo. <https://doi.org/10.5281/zenodo.3407075>
- Riechert, M., & Meyer, D. (2020). *WRF-CMake: Integrating CMake support into the Advanced Research WRF (ARW) modelling system* (WRF-CMake-4.1.5). Zenodo. <https://doi.org/10.5281/zenodo.3733593>
- Riechert, M., & Meyer, D. (2021). *WRF-CMake: Integrating CMake support into the Advanced Research WRF (ARW) modelling system* (WRF-CMake-4.2.2). Zenodo. <https://doi.org/10.5281/zenodo.4449077>
- Roh, S., & Song, H. (2020). Evaluation of Neural Network Emulations for Radiation Parameterization in Cloud Resolving Model. *Geophysical Research Letters*, 47(21). <https://doi.org/10.1029/2020GL089444>
- Roth, M. (2000). Review of atmospheric turbulence over cities. *Quarterly Journal of the Royal Meteorological Society*, 126(564), 941–990. <https://doi.org/10.1002/qj.49712656409>
- Rozoff, C. M., Cotton, W. R., & Adegoke, J. O. (2003). Simulation of St. Louis, Missouri, Land Use Impacts on Thunderstorms. *JOURNAL OF APPLIED METEOROLOGY*, 42, 23. [https://doi.org/10.1175/1520-0450\(2003\)042<0716:SOSLML>2.0.CO;2](https://doi.org/10.1175/1520-0450(2003)042<0716:SOSLML>2.0.CO;2)
- Ryu, Y.-H., Baik, J.-J., & Lee, S.-H. (2011). A New Single-Layer Urban Canopy Model for Use in Mesoscale Atmospheric Models. *Journal of Applied Meteorology and Climatology*, 50(9), 1773–1794. <https://doi.org/10.1175/2011JAMC2665.1>
- Salamanca, F., Georgescu, M., Mahalov, A., Moustou, M., & Wang, M. (2014). Anthropogenic heating of the urban environment due to air conditioning: Anthropogenic Heating due to AC. *Journal of Geophysical Research: Atmospheres*, 119(10), 5949–5965. <https://doi.org/10.1002/2013JD021225>
- Salamanca, F., Georgescu, M., Mahalov, A., Moustou, M., Wang, M., & Svoma, B. M. (2013). Assessing summertime urban air conditioning consumption in a semi-arid environment. *Environmental Research Letters*, 8(3), 034022. <https://doi.org/10.1088/1748-9326/8/3/034022>
- Salamanca, F., Krayenhoff, E. S., & Martilli, A. (2009). On the Derivation of Material Thermal Properties Representative of Heterogeneous Urban Neighborhoods. *Journal of Applied Meteorology and Climatology*, 48(8), 1725–1732. <https://doi.org/10.1175/2009JAMC2176.1>
- Salamanca, F., Krpo, A., Martilli, A., & Clappier, A. (2010). A new building energy model coupled with an urban canopy parameterization for urban climate simulations—Part I. formulation, verification, and sensitivity analysis of the model. *Theoretical and Applied Climatology*, 99(3–4), 331–344. <https://doi.org/10.1007/s00704-009-0142-9>
- Salamanca, F., & Martilli, A. (2010). A new Building Energy Model coupled with an Urban Canopy Parameterization for urban climate simulations—Part II. Validation with one dimension off-line simulations. *Theoretical and Applied Climatology*, 99(3–4), 345–356. <https://doi.org/10.1007/s00704-009-0143-8>
- Santamouris, M., Papanikolaou, N., Livada, I., Koronakis, I., Georgakis, C., Argiriou, A., & Assimakopoulos, D. N. (2001). On the impact of urban climate on the energy consumption of buildings. *Solar Energy*, 70(3), 201–216. [https://doi.org/10.1016/S0038-092X\(00\)00095-5](https://doi.org/10.1016/S0038-092X(00)00095-5)
- Schäfer, S. A. K., Hogan, R. J., Klinger, C., Chiu, J. C., & Mayer, B. (2016). Representing 3-D cloud radiation effects in two-stream schemes: 1. Longwave considerations and effective cloud edge length: 3-D CLOUD-RADIATION EFFECTS: LONGWAVE. *Journal of*

- Geophysical Research: Atmospheres*, 121(14), 8567–8582.  
<https://doi.org/10.1002/2016JD024876>
- Schoetter, R., Masson, V., Bourgeois, A., Pellegrino, M., & Lévy, J.-P. (2017). Parametrisation of the variety of human behaviour related to building energy consumption in the Town Energy Balance (SURFEX-TEB v. 8.2). *Geoscientific Model Development*, 10(7), 2801–2831.  
<https://doi.org/10.5194/gmd-10-2801-2017>
- Schultz, M. G., Betancourt, C., Gong, B., Kleinert, F., Langguth, M., Leufen, L. H., Mozaffari, A., & Stadler, S. (2021). Can deep learning beat numerical weather prediction? *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 379(2194), 20200097. <https://doi.org/10.1098/rsta.2020.0097>
- Schuster, A. (1905). Radiation Through a Foggy Atmosphere. *The Astrophysical Journal*, 21, 1.  
<https://doi.org/10.1086/141186>
- Seitola, T., Mikkola, V., Silen, J., & Järvinen, H. (2014). Random projections in reducing the dimensionality of climate simulation data. *Tellus A: Dynamic Meteorology and Oceanography*, 66(1), 25274. <https://doi.org/10/ghfpd8>
- Seity, Y., Brousseau, P., Malardel, S., Hello, G., Bénard, P., Bouttier, F., Lac, C., & Masson, V. (2011). The AROME-France Convective-Scale Operational Model. *Monthly Weather Review*, 139(3), 976–991. <https://doi.org/10.1175/2010MWR3425.1>
- Shepherd, J. M. (2005). A Review of Current Investigations of Urban-Induced Rainfall and Recommendations for the Future. *Earth Interactions*, 9(12), 1–27.  
<https://doi.org/10.1175/EI156.1>
- Shin, H. H., & Dudhia, J. (2016). Evaluation of PBL Parameterizations in WRF at Subkilometer Grid Spacings: Turbulence Statistics in the Dry Convective Boundary Layer. *Monthly Weather Review*, 144(3), 1161–1177. <https://doi.org/10.1175/MWR-D-15-0208.1>
- Shonk, J. K. P., & Hogan, R. J. (2008). Tripleclouds: An Efficient Method for Representing Horizontal Cloud Inhomogeneity in 1D Radiation Schemes by Using Three Regions at Each Height. *Journal of Climate*, 21(11), 2352–2370.  
<https://doi.org/10.1175/2007JCLI1940.1>
- Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on Image Data Augmentation for Deep Learning. *Journal of Big Data*, 6(1), 60. <https://doi.org/10.1186/s40537-019-0197-0>
- Skamarock, W. C., Klemp, J. B., Dudhia, J., Gill, D. O., Barker, D. M., Duda, M. G., Huang, X.-Y., Wang, W., & Powers, J. G. (2008). *A Description of the Advanced Research WRF Version 3* (NCAR Technical Note NCAR/TN-475+STR).
- Skamarock, W. C., Klemp, J. B., Dudhia, J., Gill, D. O., Liu, Z., Berner, J., Wang, W., Powers, J. G., Duda, M. G., Barker, D. M., & Huang, X.-Y. (2019). A Description of the Advanced Research WRF Model Version 4. *NCAR Technical Note NCAR/TN-556+STR*, 145.  
<https://doi.org/10.5065/1dfh-6p97>
- Sklar, M. (1959). Fonctions de repartition an dimensions et leurs marges. *Publ. Inst. Statist. Univ. Paris*, 8, 229–231.
- Sochat, V. V., Prybol, C. J., & Kurtzer, G. M. (2017). Enhancing reproducibility in scientific computing: Metrics and registry for Singularity containers. *PLOS ONE*, 12(11), e0188511. <https://doi.org/10.1371/journal.pone.0188511>
- Soden, B. J., Collins, W. D., & Feldman, D. R. (2018). Reducing uncertainties in climate models. *Science*, 361(6400), 326–327. <https://doi.org/10.1126/science.aau1864>
- Stamen Design. (2021). *Stamen Maps*. <http://maps.stamen.com/>
- Stensrud, D. J. (2007). *Parameterization Schemes: Keys to Understanding Numerical Weather Prediction Models*. Cambridge University Press.  
<https://doi.org/10.1017/CBO9780511812590>

- Stephens, G. L. (1984). The parameterization of radiation for numerical weather prediction and climate models. *Monthly Weather Review*, 112(4), 826–867.  
[https://doi.org/10.1175/1520-0493\(1984\)112<0826:TPORFN>2.0.CO;2](https://doi.org/10.1175/1520-0493(1984)112<0826:TPORFN>2.0.CO;2)
- Tagasovska, N., Ackerer, D., & Vatter, T. (2019). Copulas as high-dimensional generative models: Vine copula autoencoders. In H. Wallach, H. Larochelle, A. Beygelzimer, F. dAlché-Buc, E. Fox, & R. Garnett (Eds.), *Advances in neural information processing systems* 32 (pp. 6528–6540). Curran Associates, Inc. <http://papers.nips.cc/paper/8880-copulas-as-high-dimensional-generative-models-vine-copula-autoencoders.pdf>
- Takane, Y., Kikegawa, Y., Hara, M., Ihara, T., Ohashi, Y., Adachi, S. A., Kondo, H., Yamaguchi, K., & Kaneyasu, N. (2017). A climatological validation of urban air temperature and electricity demand simulated by a regional climate model coupled with an urban canopy model and a building energy model in an Asian megacity. *International Journal of Climatology*, 37, 1035–1052. <https://doi.org/10.1002/joc.5056>
- TensorFlow Developers. (2021a). *Install TensorFlow for C*. TensorFlow.  
[https://www.tensorflow.org/install/lang\\_c](https://www.tensorflow.org/install/lang_c)
- TensorFlow Developers. (2021b). *TensorFlow Lite | ML for Mobile and Edge Devices*. TensorFlow.  
<https://www.tensorflow.org/lite>
- TensorFlow Developers. (2021c). *TensorFlow (v2.6.2)*. Zenodo.  
<https://doi.org/10.5281/zenodo.5645375>
- Thaler, F., Moosbrugger, S., Osuna, C., Bianco, M., Vogt, H., Afanasyev, A., Mosimann, L., Fuhrer, O., Schulthess, T. C., & Hoefler, T. (2019). Porting the COSMO Weather Model to Manycore CPUs. *Proceedings of the Platform for Advanced Scientific Computing Conference*, 1–11. <https://doi.org/10.1145/3324989.3325723>
- Thompson, P. D. (1983). A history of numerical weather prediction in the United States. *Bulletin of the American Meteorological Society*, 64(7), 755–769.  
<http://www.jstor.org/stable/26223121>
- Trivedi, P. K., & Zimmer, D. M. (2006). Copula Modeling: An Introduction for Practitioners. *Foundations and Trends® in Econometrics*, 1(1), 1–111. <https://doi.org/10/fwkzd3>
- UCAR. (2019). *WPS V4 Geographical Static Data*.  
[http://www2.mmm.ucar.edu/wrf/users/download/get\\_sources\\_wps\\_geog.html](http://www2.mmm.ucar.edu/wrf/users/download/get_sources_wps_geog.html)
- UCAR. (2020). *Wrf-model/WRF*. GitHub. <https://github.com/wrf-model/WRF>
- Ukkonen, P. (2021). *Exploring pathways to more accurate machine learning emulation of atmospheric radiative transfer* [Preprint]. Atmospheric Sciences.  
<https://doi.org/10.1002/essoar.10508510.1>
- Ukkonen, P., Pincus, R., Hogan, R. J., Nielsen, K. P., & Kaas, E. (2020). Accelerating radiation computations for dynamical models with targeted machine learning and code optimization. *Journal of Advances in Modeling Earth Systems*, 12(12).  
<https://doi.org/10.1029/2020ms002226>
- Unger, J. (1999). Comparisons of urban and rural bioclimatological conditions in the case of a Central-European city. *International Journal of Biometeorology*, 43(3), 139–144.  
<https://doi.org/10.1007/s004840050129>
- Unidata. (2019). *Unidata NetCDF [C]*. Unidata. <https://github.com/Unidata/netcdf-c>
- United Nations. (2019). *World Urbanization Prospects: The 2018 Revision (ST/ESA/SER.A/420)*. New York: United Nations. <https://population.un.org/wup/Publications/Files/WUP2018-Report.pdf>
- United States Army Research Laboratory. (2022). *Historic Computer Images*.  
<https://ftp.arl.army.mil/ftp/historic-computers/>

- Van Bavel, J. J., Mende-Siedlecki, P., Brady, W. J., & Reinero, D. A. (2016). Contextual sensitivity in scientific reproducibility. *Proceedings of the National Academy of Sciences*, 113(23), 6454–6459. <https://doi.org/10.1073/pnas.1521897113>
- Varnai, T., & Davies, R. (1999). Effects of Cloud Heterogeneities on Shortwave Radiation: Comparison of Cloud-Top Variability and Internal Heterogeneity. *Journal of the Atmospheric Sciences*, 56, 19. [https://doi.org/10.1175/1520-0469\(1999\)056<4206:EOCHOS>2.0.CO;2](https://doi.org/10.1175/1520-0469(1999)056<4206:EOCHOS>2.0.CO;2)
- Veerman, M. A., Pincus, R., Stoffer, R., van Leeuwen, C. M., Podareanu, D., & van Heerwaarden, C. C. (2021). Predicting atmospheric optical properties for radiative transfer computations using neural networks. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 379(2194), 20200095. <https://doi.org/10.1098/rsta.2020.0095>
- Wan, Z., Zhang, Y., & He, H. (2017). Variational autoencoder based synthetic data generation for imbalanced learning. *2017 IEEE Symposium Series on Computational Intelligence (SSCI)*, 1–7. <https://doi.org/10.1109/SSCI.2017.8285168>
- Warner, T. T. (2010). *Numerical Weather and Climate Prediction*. Cambridge University Press. <https://doi.org/10.1017/CBO9780511763243>
- Watson-Parris, D. (2021). Machine learning for weather and climate are worlds apart. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 379(2194), 20200098. <https://doi.org/10.1098/rsta.2020.0098>
- Wilcox, L. J., Liu, Z., Samset, B. H., Hawkins, E., Lund, M. T., Nordling, K., Undorf, S., Bollasina, M., Ekman, A. M. L., Krishnan, S., Merikanto, J., & Turner, A. G. (2020). Accelerated increases in global and Asian summer monsoon precipitation from future aerosol reductions. *Atmospheric Chemistry and Physics*, 20(20), 11955–11977. <https://doi.org/10.5194/acp-20-11955-2020>
- World Economic Forum. (2022). *The Global Risks Report 2022—17th Edition*. World Economic Forum; 978-2-940631-09-4. [https://www3.weforum.org/docs/WEF\\_The\\_Global\\_Risks\\_Report\\_2022.pdf](https://www3.weforum.org/docs/WEF_The_Global_Risks_Report_2022.pdf)
- Xie, B., Fung, J. C. H., Chan, A., & Lau, A. (2012). Evaluation of nonlocal and local planetary boundary layer schemes in the WRF model. *Journal of Geophysical Research: Atmospheres*, 117(D12). <https://doi.org/10.1029/2011JD017080>
- Xu, L., & Veeramachaneni, K. (2018). Synthesizing Tabular Data using Generative Adversarial Networks. *ArXiv:1811.11264 [Cs, Stat]*. <http://arxiv.org/abs/1811.11264>
- Zhang, C., & Wang, Y. (2017). Projected Future Changes of Tropical Cyclone Activity over the Western North and South Pacific in a 20-km-Mesh Regional Climate Model. *Journal of Climate*, 30(15), 5923–5941. <https://doi.org/10.1175/JCLI-D-16-0597.1>