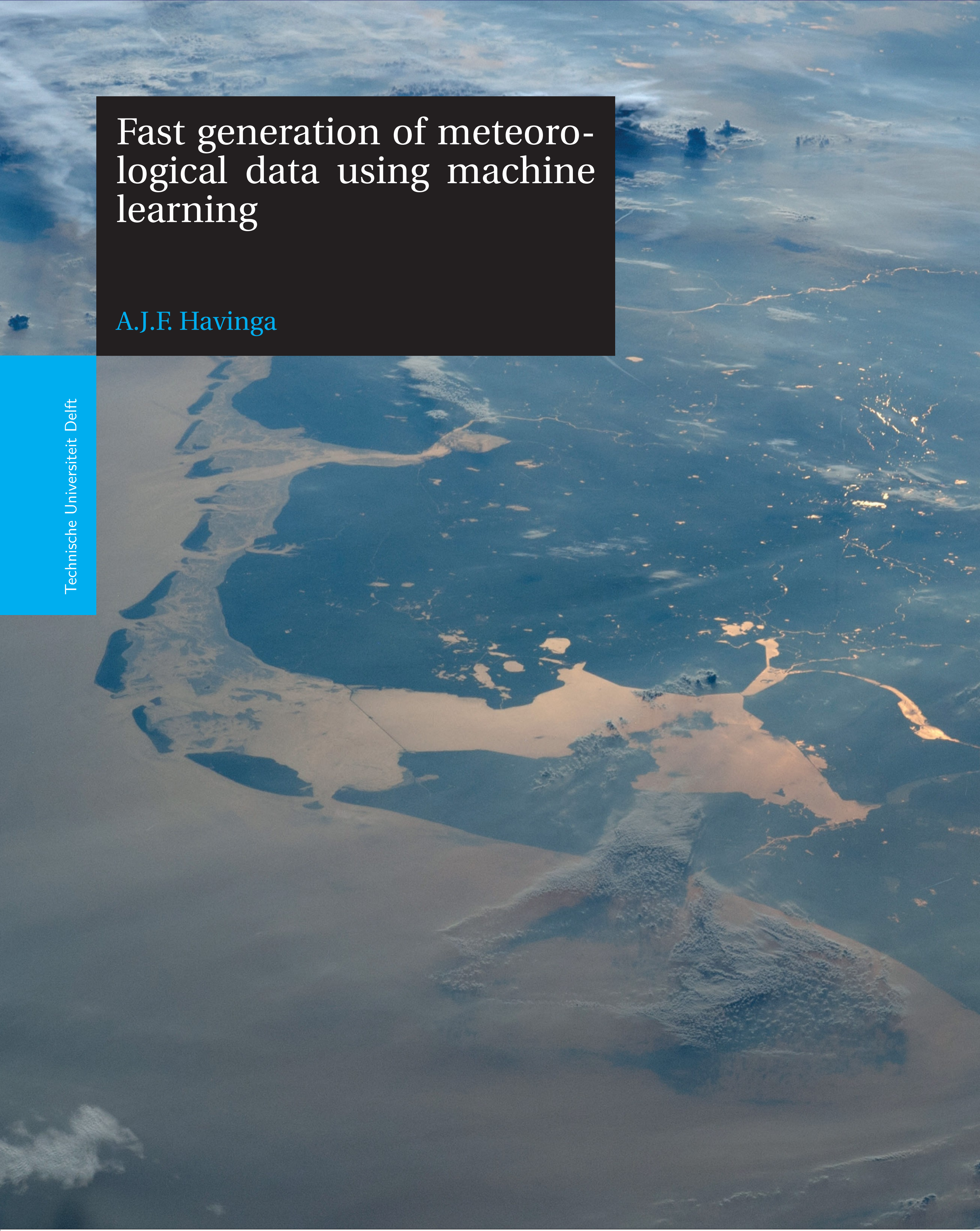


Fast generation of meteorological data using machine learning

A.J.F. Havinga

Technische Universiteit Delft

TU Delft Delft University of Technology

Challenge the future

Fast generation of meteorological data using machine learning

by

A.J.F. Havinga

in partial fulfillment of the requirements for the degree of

Master of Science
in Aerospace Engineering

at the Delft University of Technology,
to be defended publicly on Friday, December 5, 2025 at 13:30.

Supervisors:	Dr. V. Meijer,	TU Delft
	Dr. Rer. Nat. M. Mertens,	TU Delft
	Dr. Ir. A. J. Segers,	TNO
Chair:	Dr. C. Falsetti,	TU Delft

An electronic version of this thesis is available at http://repository.tudelft.nl/.

Delft
University of
Technology

Preface

This thesis marks the end of my time as a student. For most of my life I have been interested in vehicle design and climate. My time as a student started at the University of Twente, doing a bachelor in Advanced Technology. During my Minor in Luleå I came to the realization that I wanted to pursue a master Aerospace Engineering in Delft. During my master I was intrigued by courses in emissions and climate effects, after which I decided to follow up on this interest by carrying out my thesis in air quality.

This study was performed as a Master thesis at TNO. It has been particularly interesting to combine air quality with AI. During this thesis I have learned a lot about methane, its climate impact and its mechanisms, as well as the workings of AI. It has been great to be part of this project and am delighted to see that the work will be continued.

Firstly, I would like to thank my supervisors from TU Delft, Vincent Meijer and Mariano Mertens for the supervision. I am grateful for the time they spent guiding me through my thesis and it has been a pleasure working with both of you. Particularly the help in the finishing stages of the thesis has been very helpful in improving the quality of the report, allowing for a better structure and visualization of the process.

Secondly, I would like to thank my supervisor from TNO, Arjo Segers for the guidance during the project. It has been great working with you, discussing the progress on a weekly basis while learning about air quality modelling has been very interesting. In addition I would like to thank the team at TNO. Everyone has been very welcoming during the entirety of the thesis, making it a pleasure to come in the office and carry out the work.

Finally, I would like to thank my friends and family. Throughout the years I have travelled, worked and lived in various places and established many lasting friendships along the way, with some relationships even going back to my early years in primary school. Similarly my family has always been important to me, I have had their continuous encouragement over the years, for which I am very grateful. A special thanks goes out to my father, Paul Havinga, who unfortunately passed away in 2024. I have always had his unwavering support and he consistently motivated me to get the best out of myself. His influence has been instrumental in helping be become who I am today.

A.J.F. Havinga
Delft, November 2025

CONTENTS

1

Introduction

In recent years, awareness on the impact of greenhouse gases on the environment has increased significantly. One important greenhouse gas is methane (CH_4), it is after CO_2 the second-most important greenhouse gas. In a period ranging from 1750 to 2011 methane concentrations have increased by a factor of 2.5 (722 ppb to 1803 ppb). Methane abundance is a factor 200 lower than that of CO_2, however it is responsible for 25% of the anthropogenic radiative forcing [1]. Its combination of a short lifetime and high radiative efficiency therefore means that reducing methane emissions is an essential step for limiting climate change [2].

In order to understand the transport of greenhouse gases in the atmosphere, atmospheric Chemical Transport Models (CTM) have been developed. One such model is TM5 (Tracer Model 5) [3], a community model that is used at TNO (Toegepast Natuurwetenschappelijk Onderzoek). TM5 has many uses, one of which is to describe the presence of methane in the atmosphere.

Chemistry transport models such as TM5 require a vast amount of data as input. One of the inputs required for TM5 is meteorological data on a global scale. Traditionally meteorological data is acquired from ECMWF (European Center for Medium-Range Weather Forecasts), which maintains a large database containing global meteorological data dating back to 1940. This meteorological data is downloaded from and postprocessed to a format suitable for TM5. Unfortunately, the time required for downloading and processing this data is a major bottleneck in the time required to do analysis runs using TM5.

The recent surge in development of AI (Artificial Intelligence) models, combined with the availability of large datasets of meteorologic data led to the development of several AI forecasting models that can compete with state-of-the-art numerical models on accuracy [4], while being several orders of magnitude faster. In recent years many models have been developed that can accurately generate a weather forecast using AI. The emergence of these models has led to the question whether they can reduce the reliance on existing databases to speed up the acquisition of meteorological data.

This study will therefore focus on answering the aforementioned question and will investigate the possibility of generating meteorological data using these AI models, and replacing the current data with AI generated data. In this report, a literature study will be carried out in which the current state of AI applied to meteorology will be investigated. This analysis will be continued by the definition of performance criteria important for TM5, which will then be accompanied with a decision of the most suitable forecasting model for this project, along with an elaboration of the chosen forecasting model.

Once a forecasting model has been chosen, a new pipeline will be defined in order to integrate the AI model into TM5. The new pipeline will mainly focus on the calculations required to match the meteorological output from the AI model to the input of TM5, along with an error propagation study, in which the error induced by the new pipeline will be estimated. The next step will be to integrate the new pipeline in TM5 and run it in various configurations. Particular attention will be paid to verification of the new pipeline, impact of each forecasting model output variable and effect of forecasting length on the output of TM5. The final step is to interpret the results, draw conclusions and discuss the results and possible improvements.

2

Problem Definition

This report will focus on the atmospheric Chemical Transport Model (CTM) TM5. A CTM is a model that can simulate the distribution of atmospheric trace gases in space and time. At TNO TM5 is used for this. In order to run TM5, data needs to be downloaded, preprocessed and stored locally before it can be used. This pipeline has proven to be a major bottleneck, causing TM5 to slow down significantly. This has led to the desire for an alternative to downloading large datasets from the meteorological database. Using a numerical forecasting system for generating a weather forecast is not an option. Numerical forecasting systems are well developed and have proven to be reliable, however these models are computationally expensive, thus still requiring a lot of time to generate data.

Recent advances in machine learning have led to the development of weather forecasting AI models that have shown to produce accurate weather forecasts while being orders of magnitude faster than traditional numerical models [5]. Generating meteorological data using AI could therefore allow for a significant speed up in the pipeline required for running TM5. This report will therefore focus on replacing meteorological data downloaded from meteorological data archives with machine learning generated data, leading to the main research question of this report.

Is it possible to use machine learning generated meteorological data to replace data downloaded from meteorological data archives for running chemical transport models, while maintaining the transport model accuracy on methane mixing ratios?

In order to answer the research question, three subquestions have been defined that break up the research question. The first part in answering the research question is to determine what the current state of AI is in meteorology, which will be done based upon a literature study. For this the following subquestion has been defined:

Can meteorological data suitable for TM5 be generated using AI?

Once the first subquestion has been answered, a suitable AI meteorological model can be selected and a pipeline to integrate the model into TM5 can be defined. Once the new pipeline has been defined, the sensitivity of TM5's methane mixing ratios on AI generated meteorological data can be gauged by altering the amount of meteorological variables and forecasting lead times. This results in the second research question:

To what extent are simulated methane mixing ratios in TM5 affected by a switch to AI generated meteorological data?

The final subquestion is aimed at at the amount of time that can be saved through the use of AI generated meteorological data. For this the reduction in data that needs to be downloaded will be investigated, after which the following research subquestion can be answered:

How much time can be saved in TM5 through the use of machine learning for generating meteorological data?

3

Atmospheric simulation models

This chapter will introduce global atmospheric simulation models, starting with an elaboration on ECMWF's meteorological simulation models in Section 3.1. For this study the meteorological data will be used in atmospheric chemistry model TM5, which will be elaborated further in Section 3.2.

3.1. Global meterological models

Meteorology plays a large role in society, with uses ranging from people choosing what to wear to climate monitoring. For this reason databases containing meteorological data are maintained by a variety of organizations. One of these databases is ERA5 (ECMWF Re-Analysis v5), which is generally considered to be the highest quality meteorological database in existence. This section will start off with a short introduction of ERA5. The second part of this section will discuss the vertical discretization methods commonly used for numerical systems.

3.1.1. ERA5 meteorological database

Meteorological data is one of the key inputs for TM5. TNO currently downloads meteorological data from the ERA5 reanalysis database [6] which is an atmospheric reanalysis database created by ECMWF. ERA5 contains hourly or 3 hourly estimates of many atmospheric and surface variables, such as pressures, temperature, and wind velocities, for the period from January 1940 until now. ERA5 generates its results using 4D-Var data assimilation, which is a process where a model forecast is combined with observations to arrive at a combination between forecast and observations [7]. This database is generally considered to give the most accurate depiction of the weather conditions in the Earth's atmosphere [6]. ECMWF stores its data into a database called MARS (Meteorological Archival and Retrieval System). The MARS archive contains 2D and 3D meteorological data such as daily operational forecasts and ERA5 reanalysis [8]. TM5 specifically uses reanalysis data from ERA5, which will be considered the ground truth for meteorological data in this study.

3.1.2. Vertical discretization

For the vertical discretization of meteorological models in a continuous physical system, ERA5 makes use of model levels. The vertical layers that ECMWF uses for its models are denoted as 'hybrid sigma pressure layers'. This denotes that two approaches are combined: for the troposphere (up to about 10 km), 'sigma' layers are used that follow the surface pressure, while for the stratosphere pure pressure layers are used. This has the advantage that near the surface the layers are terrain following, thus aligned with elevation in mountainous regions. In the stratosphere however, layers are simply aligned with pressure, which reflects the 'layered' structure of the stratosphere. The hybrid layers are defined by the pressure at their interfaces $P_{i,j,k}$, which is defined using the surface pressure $P_{s_{i,j}}$ and two lists of coefficients named a_k and b_k. Indices i and j denote the latitudinal and longitudinal location. k is used to represent the vertical layers. The pressure $P_{i,j,k}$ at the hybrid layer interfaces can be calculated using the following equation [9]:

$$P_{i,j,k} = a_k + b_k \cdot P_{s_{i,j}} \tag{3.1}$$

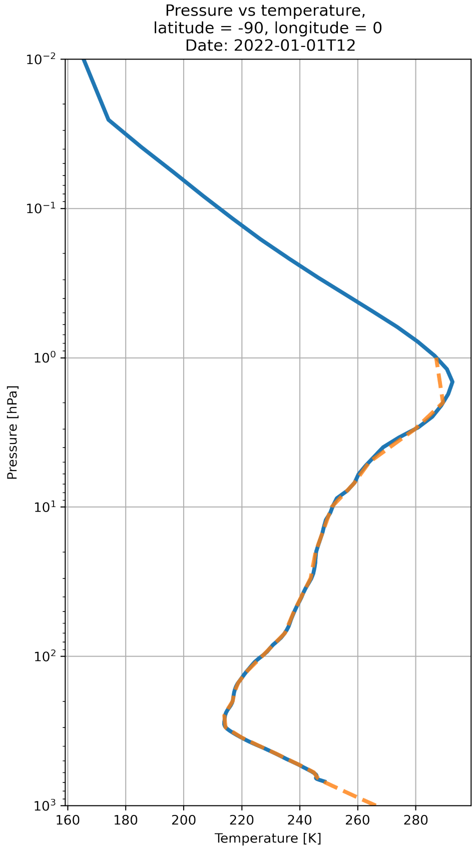


Figure 3.1: Temperature data comparing pressure levels to model levels on the south pole.

In order to calculate the pressure at the model levels, the following equation can be used [9]:

$$P_k = \frac{P_{k+1/2} + P_{k-1/2}}{2} \tag{3.2}$$

An alternative commonly applied method for the vertical discretization of meteorological data is through the use of pressure levels. Pressure level data is not terrain following, but is given at predefined pressures, independent of the surface pressure. The usage of pressure levels causes problems particularly when the actual surface pressure drops below 1000 hPa. ECMWF stores data at all 37 pressure levels, even if that pressure is higher than the actual surface pressure. For pressures higher than the surface pressure, data is extrapolated to the required pressure levels, despite this data being invalid. Meteorological AI models are commonly trained on this data and therefore also produce data at all pressure levels. An example can be seen in Figure 3.1, where the pressure is plotted against temperature on the South Pole. In the figure it can be seen how the model level data stops at a pressure of roughly 700 hPa (surface pressure), whereas the pressure level dataset continues down to 1000 hPa.

Numerical forecasting models, such as the ones available in the MARS archive are generated using model levels, after which data is interpolated to pressure levels and stored. This data is commonly either stored in MARS, or in the CDS (Climate Data Storage). The CDS is a different database for meteorological data, containing only the most commonly used variables and formats. The CDS makes use of disk storage, making it significantly faster than the MARS archive, which stores its data on tape drives.

3.2. Global atmospheric tracer model TM5

Atmospheric chemistry models, such as TM5, have been developed to understand the distribution of atmospheric trace gases. TM5 is a model that can determine the composition of the atmosphere through time and space. It has many applications, such as assessing the impact and potential consequences of emissions. TM5 starts off by discretising the earth's atmosphere into a 3D grid. Secondly, meteorological data is incorporated

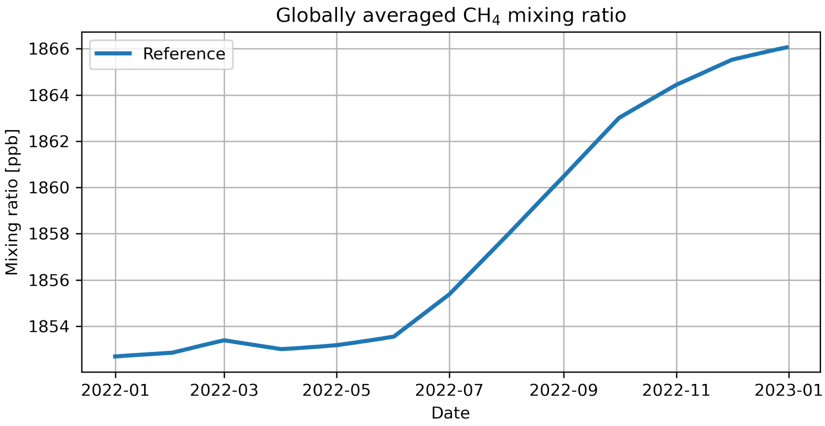


Figure 3.2: Global mean mixing ratio CH_4 in 2022

into this grid. As a third step the various operators governing the transport and chemistry are split into a time stepping procedure. The benefit of using such an operating splitting algorithm is that it improves accuracy without the requirement of small time steps. These operators are advection in three directions, vertical mixing, chemistry and sources/sinks. As a result the model provides the atmospheric mixing ratios of various trace gases, such as CO, NO_2 and CH_4 among others [3].

From this archive, meteorological data is downloaded at 137 model levels and a horizontal resolution of roughly 30 km. A multitude of variables are imported for TM5. This data comprises of surface pressure, vorticity, divergence, temperature, humidity and detrainment rates. Apart from the model level data, multiple surface field datasets are required as well. The focus for this study will be on replacing the model level data, since these datasets are the largest in size. A table with all required meteorological variables can be seen in A.1. This meteorological data is then coarsened to a 1° by 1° horizontal resolution; this is roughly 100x100 km at the equator, but in east-west direction the cells become smaller towards the north pole. The 137 model levels are merged into 34 model levels for the reference version of TM5 used in this study. For this study a version of TM5 configurated for 34 model levels was used, however multiple versions of TM5 exist, configured to simulate using various horizontal and vertical resolutions. After the processing step, temperature, humidity, horizontal and vertical massfluxes in model levels will be used for TM5, along with a variety of surface level variables.

This study will use a version of TM5 that is configured to simulate methane mixing ratios. TM5 is capable of determining the mixing ratio of methane over the globe for the entire time period that is simulated. These results can be processed for further analysis. An example of one of the outputs that can be retrieved from TM5 is the global averaged methane mixing ratio, which can be seen in Figure 3.2 for the entirety of 2022. The mixing ratio starts at 1854 ppb and remains relatively constant for the first half of the year, which is due to the balance between emissions and breakdown of CH_4. In the second half of the year the mixing ratio increases, after which it reaches 1866 ppb by January first 2023. This seasonal dependency can be attributed to an increase in emissions on the northern hemisphere (wetlands, rice fields and forest fires) in the summer period [10].

Apart from time series, TM5's results can also be used to investigate more specific phenomena, such as spatial patterns in the mixing ratios. Figure 3.3 shows snapshots of the mixing ratio for model levels 34 and 23 levels on December the 30th, after a one year simulation. The given model levels are the closest representation to 1000 and 500 hPa respectively for the 34 model level configuration of TM5. On both plots a mean difference of ± 140 ppb exists between the northern and southern hemisphere, being denoted as a north/south gradient. The reason for this gradient is due to most of the CH_4 sources being present in the northern hemisphere, and mixing towards other latitudes being slower than mixing along the longitudes [11].

Model level 34 is the surface level and can be seen on the left plot. The impact of surface sources are very visible in the mixing ratios on model level 34. It can be seen how particularly in China and India the mix-

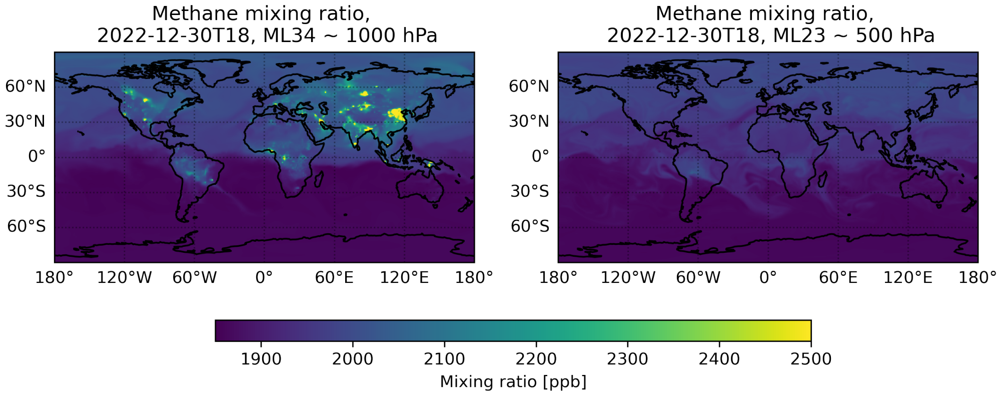


Figure 3.3: Methane mixing ratio at model levels 34 an 23, which are at ≈ 1000 and ≈ 500 hPa respectively.

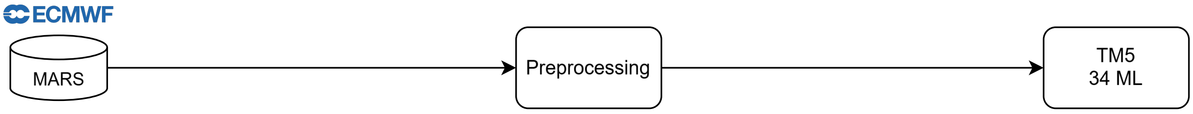


Figure 3.4: Existing pipeline of TM5. On the left hand side meteorologic data is retrieved from the MARS archive, which is coarsened to the correct format, after which it is used in TM5.

ing ratio is significantly higher than elsewhere. The pollution in North America is lower already, followed by Europe having even lower mixing ratio's. On the southern hemisphere, neither South-America, nor Africa or Australia appear to contribute as significantly to the CH_4 mixing ratio. The mixing ratios in model level 23 (right plot), roughly 1.5 km above sea level, show how the regions containing large amounts of CH_4 are more uniformly distributed, caused by the dispersion of local high concentrations of methane, however the north/south gradient is still visible.

In Figure 3.4 the current pipeline from TM5 can be seen. The current pipeline is straightforward and only consists of three steps, starting with data being downloaded from the MARS archive on the left hand side of the figure. The second step is the preprocessing step, in this step the meteorological data is coarsened to the required format for TM5 and massfluxes are calculated from the vorticity and divergence. After the data has been processed it can be used in TM5. Later in this report, a new pipeline will be defined, in which the AI model will be included in the process. The new pipeline will be added to Figure 3.4 later in this report.

4

AI AND METEOROLOGY

This chapter will elaborate on what AI is, give a brief introduction on how it works and its current state in Section 4.1. Section 4.2 will then elaborate on the various AI meteorological models that are currently in existence. A set of performance metrics on which the various AI models can be compared will be defined in Section 4.3, after which a decision will be made on the most suitable forecasting model for this study. Finally in Section 4.4 the chosen meteorological model will be further elaborated.

4.1. CURRENT STATE OF AI

Artificial intelligence (AI) is a broad and multifaceted field that intersects various disciplines, such as computer science, neuroscience, and psychology. Although a universally accepted definition of AI has yet to be established, existing attempts to define a definition tend to share several common elements. Aspects that are often mentioned are the capability of a system to show signs of intelligence, by means of learning, reasoning and decision making in a similar fashion as human beings [12]. Machine learning is a subset of AI that aims to improve a system's performance through learning from data instead of computational methods [13]. Despite AI being a topic of research for several decades, AI models have not seen widespread use until recent years [12]. Improvements in computing power and the availability of big data has allowed for these AI models to become possible [13]. The popularity of AI can largely be attributed to the possibility to utilize big data [13], as well as the many sectors in which AI can be applied. Nowadays many applications of AI have been developed, for example in computer science, healthcare, education and the transport sector [12]. The field of AI is still developing at a rapid pace, with many companies exploring the possibilities of AI.

This study will focus on a specific aspect of machine learning, called neural networks. A neural network is a type of machine learning model inspired by the structure and functioning of the human brain [12]. It has become a broad research field and is the basis of what many AI models are built upon nowadays. Neural networks in its basic form contain a large number of interconnected neurons. These neurons are simple elements that convert an input to an output. The output it creates depends on a variety of factors, such as the input from other neurons and the strength of its connections [14]. Figure 4.1 showcases an example of an M-P (McCulloch-Pitts) neuron. The neuron receives several inputs (x_i) from its preceding neurons. Each neuron input is multiplied by a weight (w_i), which was determined during the training procedure. The neuron threshold (θ) is subtracted from the sum of inputs and weights, after which the activation function (f) calculates the output (y). The activation function is a non-linear function which dictates the output based on the input.

A single neuron on its own is not particularly powerful. However when connecting a large amount of neurons in a layered structure, a neural network is created. An example of a neural network can be seen in Figure 4.2. A neural network is capable of learning and modelling complex relations. The amount of neurons in a neural network can vary significantly, large models nowadays can easily consist of billions of neurons. The example in Figure 4.2 showcases a neural network with three hidden layers. The input layer on the left hand side transmits its data into the first hidden layer. Within this layer, each neuron independently processes the received input and forwards the result to the subsequent layer. Neurons within the same layer do not interact

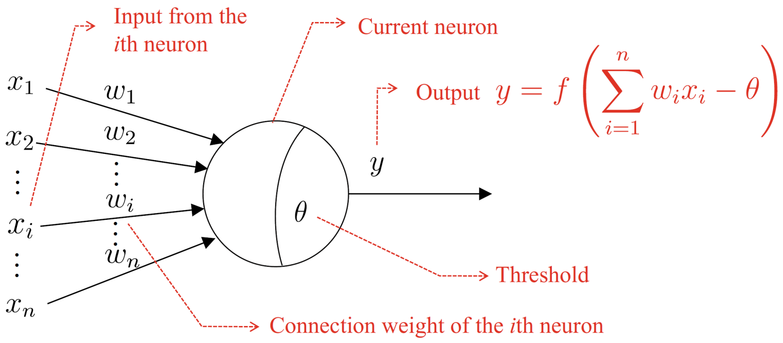


Figure 4.1: M-P neuron model [13]

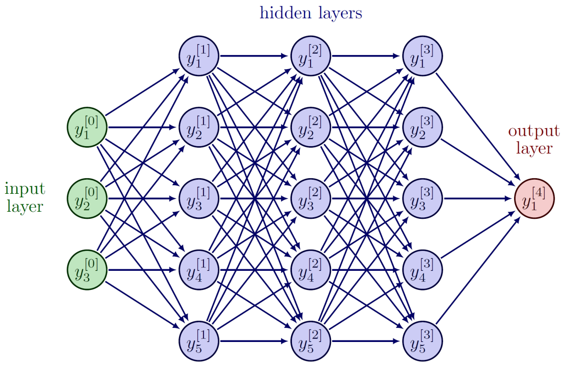


Figure 4.2: A neural network with three hidden layers [14].

with one another; communication occurs exclusively between adjacent layers. Once all hidden layers have processed the data, everything is combined in the final layer where the output is produced. These neural network machine learning models are also what the modern AI meteorological forecasting models are built upon that will be discussed in section 4.2.

4.2. METEOROLOGICAL MODELS

Meteorology is an application of AI that still undergoes frequent developments, with new models still being developed and updated regularly. The first AI models capable of competing with numerical forecasting models started appearing in 2022 [15]. Ever since the first models were developed, many followed suit. Nowadays a multitude of AI weather forecasting models exist, such as FourCastNet [16], PanguWeather [5], GraphCast [17], Fuxi [18], and FengWu [19]. These models each have their strengths and weaknesses. In Table 4.1 a comparison has been made between these models. PanguWeather [5], GraphCast [17], and FourCastNet [16] are older models that have established themselves in recent years as reliable models that in some aspects can outperform state of the art numerical weather forecasts. FuXi [18] is a newer model that distinguishes itself by focusing on variables impacting human activities, such as aviation and wind/solar energy. FengWu [19] is another relatively new model that is capable of generating even higher quality forecasts than the aforementioned models.

Research shows that many of the models developed in recent years show good performance compared to state of the art numerical weather prediction models such as ECMWF's IFS HRES (Integrated Forecasting System High Resolution) for fundamental performance metrics, such as RMSE (Root Mean Square Error) of temperature, geopotential and wind speeds at various altitudes [4]. These forecasts can be generated in a matter of seconds on GPU nodes, while numerical models take minutes to hours of running on platforms with thousands of conventional CPU nodes. In order to compare the various models with each other, an open framework called Weatherbench [20] has been set up by Google. Weatherbench can be used to rate various models on their deterministic and probabilistic scores on a variety of performance metrics. These metrics can then be compared for a variety of output variables and forecasting lengths. An example can be

Table 4.1: A comparison of five AI forecasting models on their year of introduction, architecture, vertical resolution, timesteps and number of output variables.

Model	Introduction	Architecture	Vertical resolution	Timesteps	N. variables
PanguWeather	2022	3D transformer	13 PL	1 hour	9
FourCastNet	2022	Adaptive Fourier Neural Operator	13 PL	6 hour	13
GraphCast	2023	Graph Neural Network	37 PL	6 hour	11
FuXi 2.0	2024	U-Transformer	13 PL	1 hour	28
FengWu	2023	Cross-modal fusion Transformer	37 PL	6 hour	9
AIFS	2024	Graph Neural Network	13 PL	6 hour	13

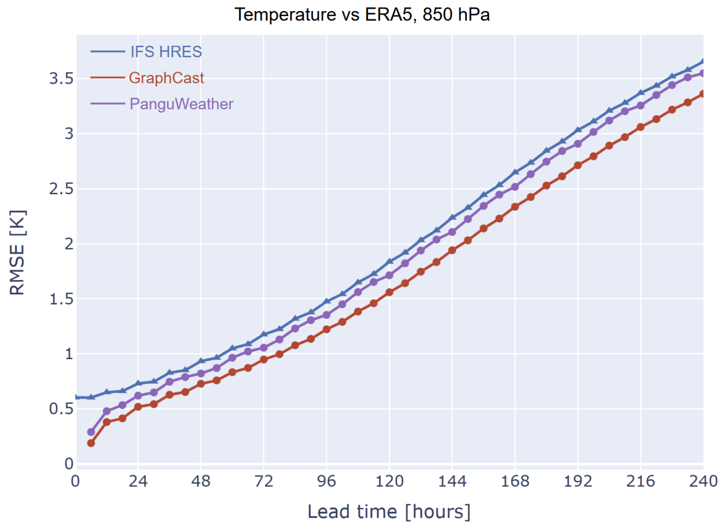


Figure 4.3: Temperature RMSE comparison at 850 hPa for IFS HRES, GraphCast and PanguWeather, using ERA5 as reference [20].

seen in Figure 4.3, where GraphCast, PanguWeather and IFS are compared on temperature using RMSE as performance metric with ERA5 as reference. As can be seen from the figure, the temperature RMSE of IFS, GraphCast and PanguWeather has been tracked over a 240 hour forecast lead time. As expected, the RMSE rises over time, as the forecast accuracy reduces for longer forecasting times. Another thing to note is the fact that the RMSE of GraphCast as well as PanguWeather is lower than the RMSE of IFS HRES for the entire forecast, indicating that these AI models outperform IFS HRES on this metric over the entire forecasting period.

Despite the impressive performance of these models in performance metrics such as the one mentioned in Figure 4.3, AI models do have weaknesses, typically caused by the manner in which they have been trained. AI models have generally been aimed at minimizing the RMSE of the predicted output, compared to the target output. These models therefore perform well when comparing RMSE, however they tend to be less accurate in other metrics, such as forecasting extreme weather events and cyclone tracking [21]. An example can be seen in figure 4.4 where all AI models underestimate the high wind speeds encountered during the storm by roughly 5 m/s.

Another downside of AI meteorological models is the appearance of a smoothening effect on a large scale. Figure 4.5 shows a temperature forecast of Europe for a 6 day lead time, where PanguWeather is compared against IFS. What can be seen from the figure is that on a global scale PanguWeather and IFS generate a comparable forecast. When looking closer however, IFS portrays a lot more detail, while PanguWeather has smoothened its forecast, taking away small scale structures. The reason for this smoothing effect and poor forecasting ability of extreme events is due to the focus on minimizing RMSE in the training process. Training on minimizing RMSE penalizes distinctive forecasts with large activity, resulting in poor forecasting ability of extreme events and causes smoothing effects for long forecasting times [4].

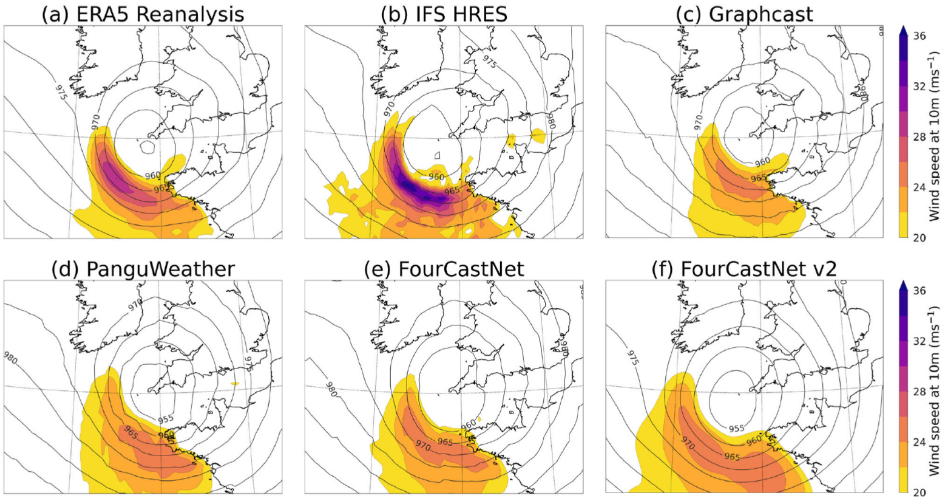


Figure 4.4: A comparison of four established AI models to IFS and ERA5 on wind speed. The top left figure shows ERA5 reanalysis, which is compared to IFS HRES in the middle of the top row. Top right, bottom left, bottom middle and bottom right show the forecast of GraphCast, PanguWeather, FourCastNet, and FourCastNet V2 respectively. All AI models underestimate wind speeds, while IFS overestimates winds speeds encountered during storm Ciaran [21].

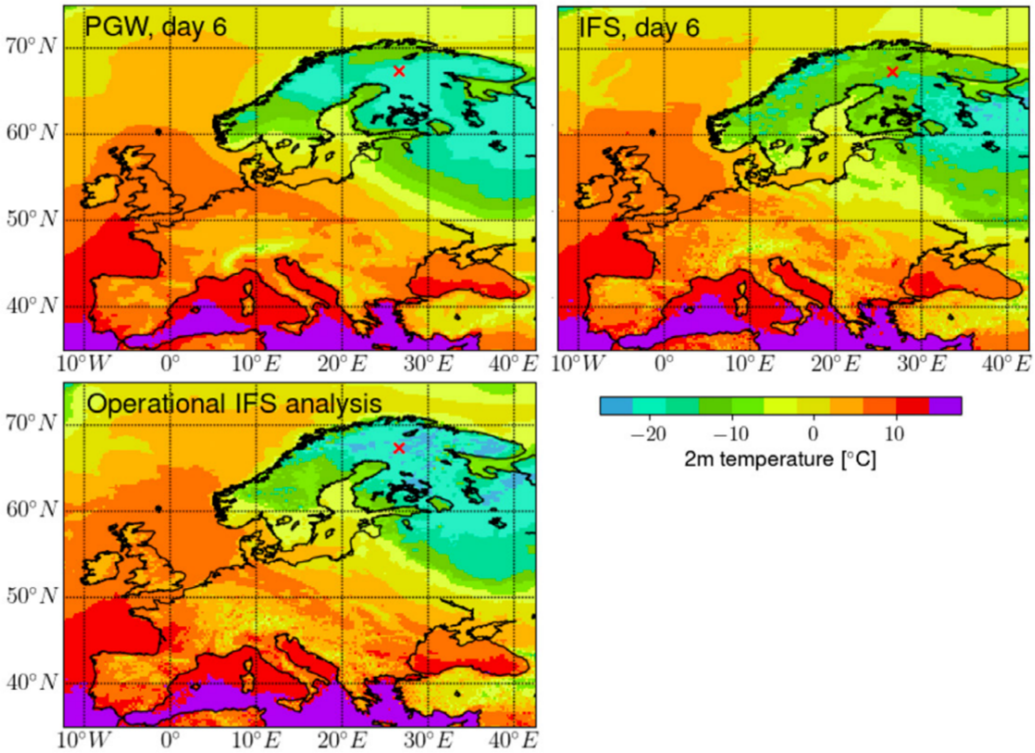


Figure 4.5: Three figures of the 2m temperature after a 6 day forecast, with top left Pangu-Weather (PGW), top right IFS and bottom left operational IFS analysis. From the figure it can be seen how Pangu-Weather shows limited detail in its forecast, whereas IFS and operational IFS do resolve small scale details [4].

4.3. MODEL EVALUATION FRAMEWORK AND SELECTION

In order to determine which meteorological AI model is most suitable for this study, several evaluation metrics have been established. A variety of meteorological AI models have been evaluated on these metrics, after which a comparison will be made at the end of this section. For this comparison five metrics have been defined.

Feasibility: The first metric is based upon the possibility of getting the model functioning. Due to the many unknowns in the process, such as the complexity of the models, use of a HPC (High-Performance Computing) cluster and not having worked with these models before, an emphasis was placed upon choosing a model that can be set up with the least amount of difficulties. Support and documentation varies per model. As such, an emphasis was placed upon choosing a more established model that is supported by ECMWF. A grade of 3 will be given for models that well documented and used by ECMWF. Well documented models without the support of ECMWF will be rewarded with a 2. Finally new models with little documentation that are not supported by ECMWF will be given a score of 1.

Vertical resolution: Another important aspect of the model is the vertical resolution of the output, as a reduction in vertical resolution is likely to reduce the accuracy of TM5. The TM5 version that will be used as reference for this project is configured to an input of 34 model levels, covering a vertical range of roughly 0-80 km. For this reason a model should be sought that forecasts at a similar vertical resolution. The score for vertical resolution will be based upon its availability of model levels and the amount of vertical levels. A model providing an output at 37 model levels will receive a score of 3, whereas 37 pressure levels will receive a score of 2. Finally a score of 1 will be given for 13 pressure levels.

Meteorological output variables: The output of the meteorological AI model serves as input for TM5. TM5 requires temperature, humidity, horizontal, and vertical massfluxes as an input. The vast majority of AI models forecast the temperature, humidity and wind in u and v direction. Vertical velocity is a less common forecasting variable that only some models provide. Massfluxes can be calculated from the velocities, though ideally the forecasting model would directly forecasts massfluxes. A score of 1 was given to forecasting models that generates at least temperature, humidity and horizontal winds. Some models also forecast the vertical velocity, which will be rewarded with a 2. A model generating horizontal and vertical massfluxes would simplify post-processing and will therefore be rewarded with a 3. No models are currently available that output massflux, hence models were only rewarded with either a 2 or 1.

Performance: The performance of the model was the fourth metric. Less emphasis was placed upon the performance of the forecasting model as difference between the various forecasting models is small for short forecasting times, along with the fact that the other metrics were deemed more important for this project. The performance of the model was based on the normalised RMSE, derived from Weatherbench [20], as well as a literature research. The RMSE of various output variables was calculated and normalised to ERA5 for various forecasting lengths. A comparison of the most prominent models in Weatherbench can be seen in Figure 4.6. From the figures is can be seen how Pangu-weather is the worst contender in most metrics, while GenCast and GraphCast generally have the lowest normalised RMSE. A score of 1 was given to the models that perform on par, or slightly better than IFS. A significant improvement to IFS was awarded with a 2. Finally the current state-of-the-art models, which even outperform the models awarded a 2 will be given a score of 3.

Future proofing: The final metric is based upon a likely switch in the future to the meteorological AI model developed by ECMWF, called AIFS. AIFS is still in an early stage of development, and therefore not yet suitable for this study. Since ECMWF is the default source for meteorological data for TM5, and because TM5 is also employed under projects lead by ECMWF, it would make sense to use AIFS for TM5 in the future. Using AIFS would also allow for technical support, usage of the ECMWF computing facilities, as well as the possibility of requesting changes to make AIFS more suitable to TM5. Choosing a model that is built upon a similar architecture as AIFS, being a Graph Neural Network (GNN), will therefore likely result in a more straightforward switch. The scoring of this metric was based upon the usage of a Graph Neural Network. Models using such an architecture were therefore given a score of 2, whereas a score of 1 was given otherwise.

The time required to generate a forecast was not used as a metric, despite speeding up the model's pipeline being an important aspect of the study. The reason for not taking the speed of the forecasting model into ac-

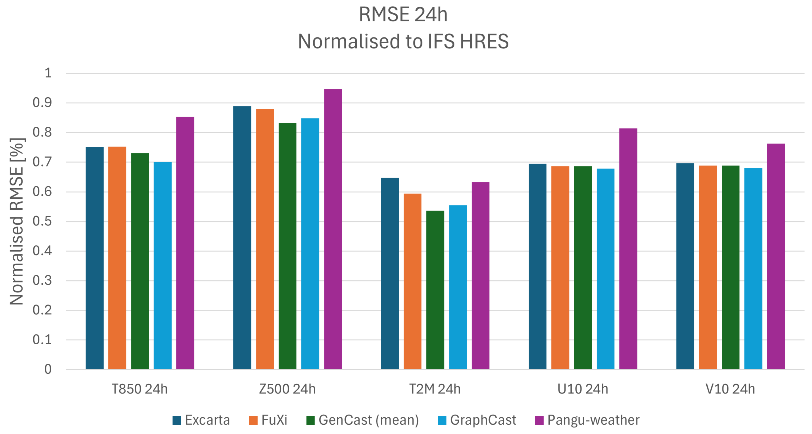


Figure 4.6: A comparison of five AI models on normalised RMSE. Four variables have been compared, being the temperature at two altitudes, geopotential, and horizontal wind speeds. The RMSE has been calculated for a forecasting time of 24 hours for all models.

count is due to the fact that practically all AI forecasting models claim to be orders of magnitude faster than numerical models, and should be able to roll out a multiple day forecast in a matter of seconds, provided that suitable hardware is used. At these speeds generating meteorological data will no longer be the bottleneck and the difference between the individual models will be negligible when the entire pipeline is taken into account.

A variety of models have been evaluated and three suitable models are shown in the spider plot in Figure 4.7. The figure shows a comparison between FourCastNet, GraphCast and FengWu. FourCastNet is an older model that scores high on feasibility, however it does not perform well on most other metrics. A newer model such as FengWu on the other hand has an excellent performance, however it does not suit this study on most other metrics. GraphCast performs reasonably well on all metrics. It is a well supported model with good documentation, provides output at 37 pressure levels, models vertical velocity, is built upon a GNN and has a reasonably good performance. These aspects make GraphCast a good all-rounder, resulting in it being chosen as the most suitable model.

4.4. Architecture of GraphCast

GraphCast is a forecasting model developed by Google Deepmind [17]. GraphCast was published in 2023 and has proven to be a high quality forecasting model. It is capable of producing a 10-day forecast at higher accuracy than the current state-of-the-art numerical forecasting systems. GraphCast generates meteorological variables at a horizontal resolution of 0.25 °, consisting of 5 surface variables and 6 atmospheric variables on 37 pressure levels. These variables are summarized in Table A.2, the pressure levels can be found in Table A.3.

The architecture of GraphCast is a Graph Neural Network, based on an ”encode - process - decode” configuration [22]. Earlier research has shown that a GNN can be very effective at modeling complex physical systems, such as meteorology [23], as well as other mesh-based simulations [24]. Another advantage of a GNN is that it scales well with size due to the possibility for learned message passing, allowing for spatial interactions over any range [17]. To illustrate the workings of GraphCast, a schematic figure of the processing behind GraphCast is shown in Figure 4.9. The figure shows several aspects of the processing steps, starting with steps *a), b)* and *c)*, which represent the manner in which a forecast is rolled out. Meteorological data at t = 0 and t = -6 hours is used as an input to generate a forecast at t = 6 hours. This output can then be used as input to generate a forecast for t = 12 hours. Repeating these steps allows for rolling out up to a 10-day forecast. This forecasting procedure has been represented in Figure 4.8.

Steps *d), e)* and *f)* show the ”encode - process - decode” configuration. In step *d)*, the encoding step is highlighted. In this step, the latitude-longitude grid, is translated into a pre-defined mesh from which GraphCast can generate its forecast. Data from the surface variables, along with all pressure levels, over various grid cells is translated into a single node on the multi mesh, resulting in a large amount of data per node. Step *e)* shows

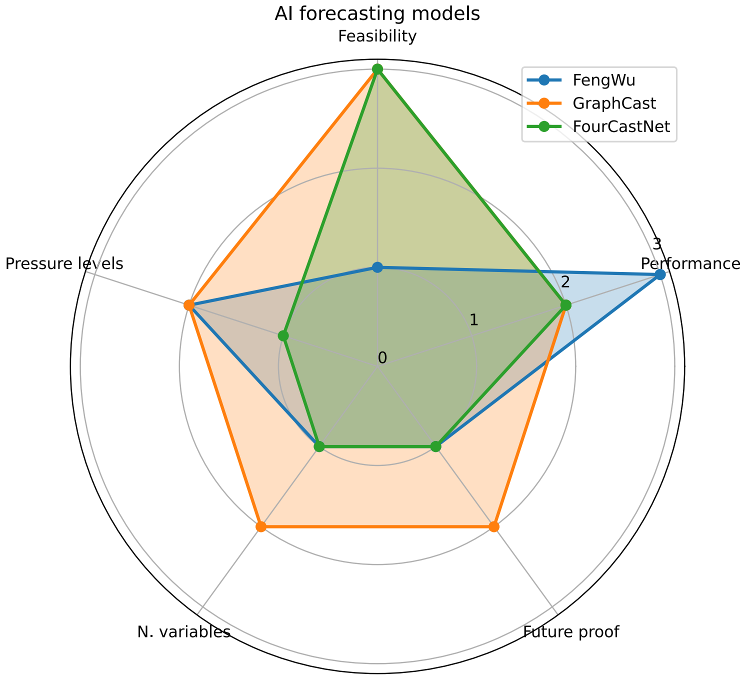


Figure 4.7: A spider plot illustrating the suitability of three forecasting models to TM5. Five legs can be seen where the metrics feasibility, performance, future proofing, number of variables and pressure levels can be seen.

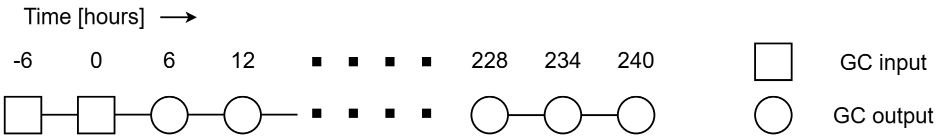


Figure 4.8: Timestepping procedure of GraphCast, datasets at t = -6 and 0 hours as input (squares), after which 40 forecasting timesteps (circles) can be generated until t = 240 hours.

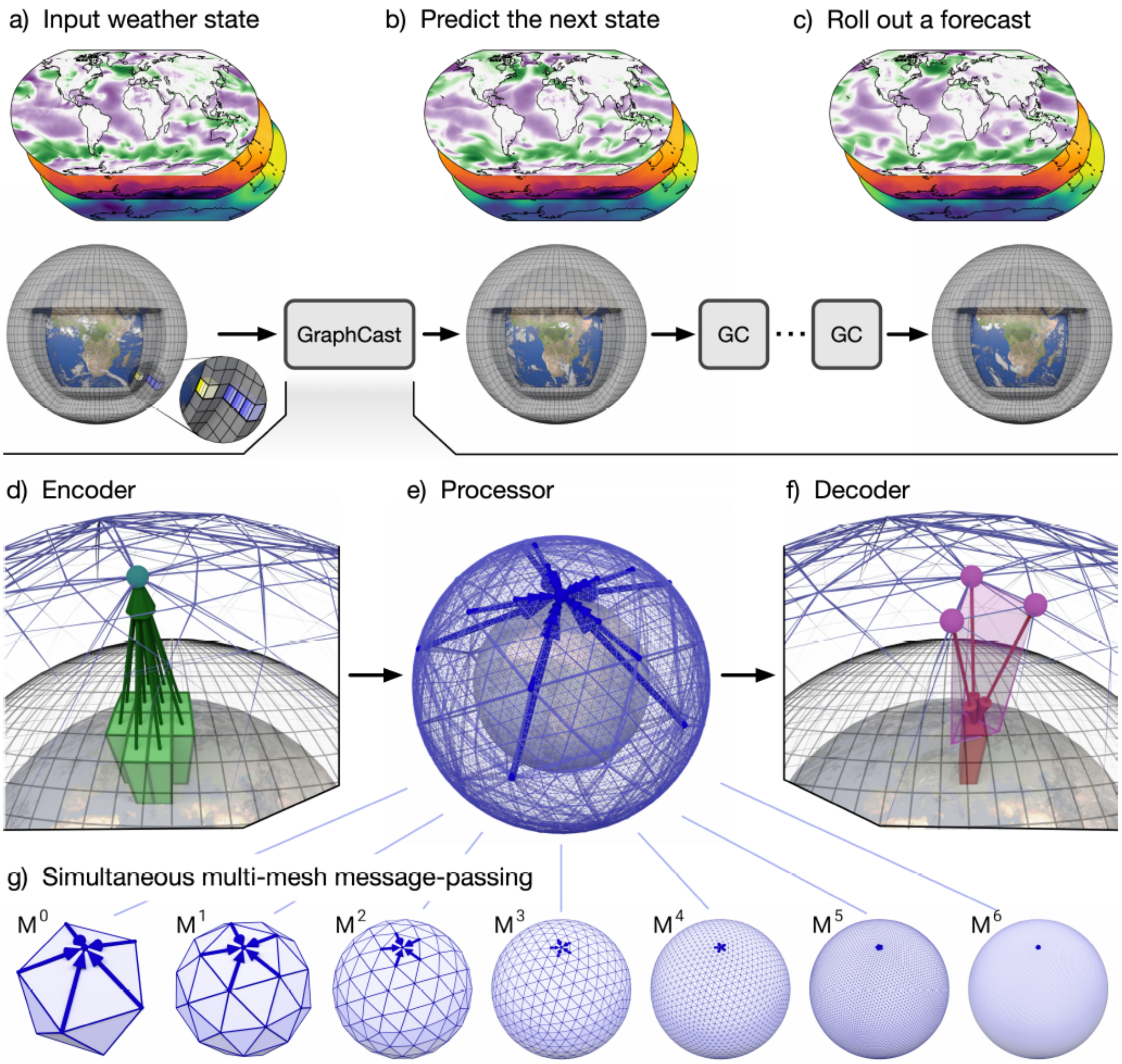


Figure 4.9: A schematic overview of GraphCast [17]. Steps *a), b)* and *c)* show the stepping procedure for rolling out a forecast. Steps *d), e)* and *f)* showcases the ”encode - process - decode” procedure. In *g)* the multi-mesh can be seen.

the processing step, with the multi-mesh. The blue arrows show the learned message passing paths which will be used by the processor. The processor then uses 16 unshared GNN layers in order to generate the next state of the forecast. Step *f)* denotes the final step. Once the processor is done, the decoder will then translate the multi-mesh back into a latitude-longitude grid.

Step *g)* shows the multi-mesh as discussed earlier. The multi-mesh is essentially a refined icosahedral mesh, as seen in M^0, containing 10 nodes. This mesh has been refined six times, multiplying the amount of nodes by 4 for each refinement step, increasing the amount of nodes from 12 to 40962. During this refining process, the message-passing is incorporated, as the connection to its neighboring node remain known for every refinement step [17].

5

Definition of new TM5 pipeline

This chapter will describe the steps required to prepare meteorological data for TM5 using GraphCast. For this a new pipeline has been defined, which will be discussed in Section 5.1. Section 5.2 will discuss the process of downloading data from the CDS and generating a forecast with GraphCast. Preprocessing the data to achieve a correct format for TM5 will be discussed in Section 5.3. The final step is running TM5, which will be discussed in Chapter 6.

5.1. Processing pipelines

A new pipeline has been defined and can be seen in Figure 5.1. The top of the figure shows the original pipeline as discussed in Figure 3.4. The bottom part of the figure shows the newly defined pipeline. The first step of the pipeline is to download data from the CDS, which is the archive containing meteorological data for GraphCast. The next step is to generate a forecast using GraphCast. The output of GraphCast is not directly suitable for TM5, hence a preprocessing step is required, which is shown in blue. As a final step TM5 can be used, which is shown in pink.

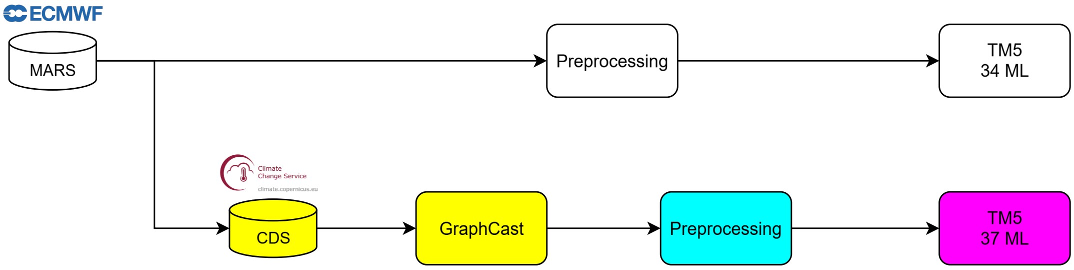


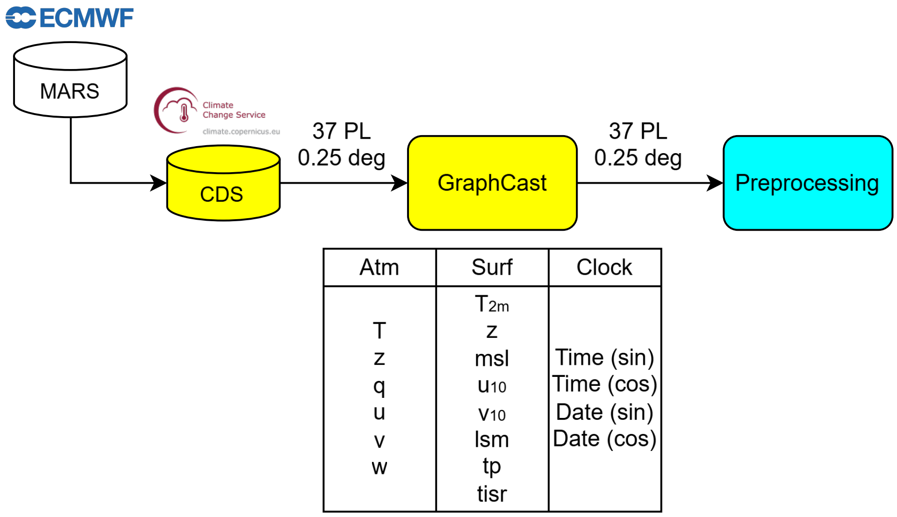
Figure 5.1: A figure representing the original pipeline for TM5 on top, along with the new pipeline used for including GraphCast generated metorological data into TM5 at the bottom. The new pipeline downloads an intial timestep of meteorological data from the CDS to generate a GraphCast forecast from. The forecast from GraphCast will subsequently be processed after which it can by used by TM5. The colors represent the different steps that will be discussed in Chapter 5 and 6.

5.2. GraphCast

This section will discuss the yellow steps in Figure 5.2.1, which consists of downloading meteorological data from the CDS and running GraphCast. This section will be split in two parts, Subsection 5.2.1 discusses the details required for running GraphCast. Subsection 5.2.2 then discusses the forecasts generated by GraphCast, along with its strengths and weaknesses.

5.2.1. Running GraphCast

GraphCast has been made available to the public, along with three pre-trained models varying in resolution. Required files containing the model weights for these models can be downloaded from the Google Cloud Bucket [25], [26]. Meteorological data should be downloaded from the CDS at a horizontal resolution of 0.25°



Atm	Surf	Clock
	T_{2m}	
T	z	
z	msl	Time (sin)
q	u_{10}	Time (cos)
u	v_{10}	Date (sin)
v	lsm	Date (cos)
w	tp	
	tisr	

Figure 5.2: A figure showing the datastream of GraphCast. Data is downloaded from the CDS after which it is used for GraphCast. The required variables are listed below GraphCast. After generating a simulation, GraphCast puts out the same variables in the same format.

and 37 pressure levels for atmospheric variables [27]. The required variables can be seen in Table A.2. The data should be downloaded at t = -6 and 0 hours, after which a 240 hour forecast can be rolled out.

Downloaded data requires little processing, apart from the total precipitation. GraphCast requires per timestep a total precipitation over those 6 hours, while the CDS archive provides the precipitation per hour. The precipitation data should therefore be downloaded in timesteps of 1 hour, after which 6 records should be summed together to get the total precipitation over a 6 hour timespan.

Once the data has been downloaded, the next step is to sort the data into datasets that can be used by GraphCast. GraphCast requires three datasets: an input, forcings, and a target dataset. The input dataset is the dataset that contains all information required for generating a new timestep. It is the largest, requires a combination of meteorological and non-meteorological data and can be seen in Table A.2. The time and date should be represented as a function of sine and cosine and can be generated using the scripts provided with GraphCast. The steps described in the two paragraphs above can also be seen in Figure 5.2

The forcings and target datasets are required for the timesteps that GraphCast will be forecasting, e.g. for a 24 hour forecast it will contain t = 6, 12, 18 and 24 hours. The forcings dataset consists of all non-meteorological variables, being time, date and Top-Of-Atmosphere (TOA) incident solar radiation. The target dataset is the dataset that GraphCast will write its forecast to. This dataset is an empty dataset required for storing the predictions after the run, set up in a format that contains all meteorological variables of Table A.2.

Once all data has been prepared, GraphCast can be started. The hardware requirements for running GraphCast are reasonably high and cannot be ran on an ordinary desktop. For this project, GraphCast ran on the TNO HPC (High Performance Computing), using CPU's. The TNO HPC contains various work nodes; in this study, most simulations were done on a HPC node using a Intel Xeon Gold 6338N CPU. On these CPU's it took roughly 20 minutes per timestep using a single core, while requiring 64GB of RAM to run.

GraphCast can also run on GPU's, allowing for a forecast to be generated significantly faster. A segmentation error prevented GraphCast to be ran on a TNO GPU, requiring the use of CPUs instead. Based on an experiment with a cloud-based GPU, a decrease of a factor 100 in model runtime has been estimated. Assuming that a similar speed up will be possible using the TNO hardware, it would suggest that generating a timestep on the high resolution model used for this project can be done in roughly 10 seconds, instead of the 20 minutes required when using CPU's.

5.2.2. Comparison of GraphCast with ERA5

In order to evaluate the error of GraphCast compared to ERA5, various comparisons will be made between variables computed for a variety of forecasting times ranging from 6 to 72 hours. GraphCast can generate a forecast of up to 240 hours, however the assumption was made that past 72 hours the quality of the forecast would have reduced to an extent that TM5 would have been significantly affected. The error of GraphCast will be evaluated by comparing the outcome of GraphCast to ERA5 meteorological data valid for the same time as available from the CDS archive, which serves as the ground truth here. In this section the temperature will be compared at a surface pressure of 1000 hPa, as well as through a comparison over the zonal mean. A final comparison will be made by comparing the RMSE of all atmospheric variables generated by GraphCast at pressure levels 1000, 850 and 500 hPa to the ERA5 for forecasting lengths ranging from 0 to 72 hours. The chosen pressure levels are generally considered to be the most influential pressure levels for weather phenomena.

The RMSE is a common metric used for checking a forecast quality as it provides a single number that represents the forecast quality over the entire spatial field that is tested [28]. Another advantage of using the RMSE is that it allows for a comparison to literature, particularly Weatherbench [20]. The RMSE can be calculated using the following equation:

$$RMSE = \sqrt{\frac{1}{M}\sum_{i=1}^{M}(\hat{x}_i - x_i)^2} \tag{5.1}$$

In this equation, M is the number of points on the grid. $\hat{x}$ is the forecast value, where x is the reference value from ERA5. The index i represents the location of the datapoint that is evaluated [28].

For this study a variety of metrics for GraphCast have been evaluated, of which a selection has been made of the most interesting comparisons. Starting off with a comparison of the temperature, Figure 5.3 shows the temperature at 1000 hPa on 2022-01-03, 18.00 UTC. This comparison shows a snapshot of one point in time, it is important to note that weather conditions, and therefore forecasting quality, varies with time. The comparison made aims to show general trends in the forecasting error, though details will vary based upon specific weather phenomena. The top left plot shows the absolute temperature, with top right, bottom left and bottom right showing the difference between ERA5 and a forecast lead time of 6, 24 and 72 hours respectively. As can be seen in Figure 5.3, GraphCast forecasts the temperature reasonably well, particularly for short forecasting times. At a forecasting time of 6 hours, GraphCast's forecast shows a strong resemblance to ERA5, with differences at a maximum of ± 1 degree Kelvin over the entire plot. For longer forecasting times the differences starts becoming more apparent. For a 72 hour forecast the errors become more noticeable, with differences exceeding 5 degrees, particularly on the North and South Pole. These differences past ± 80 degrees latitudinally are thought to be due to distorted cells caused by the latitudinal/longitudinal grid.

A comparison has also been made based upon the mean of the longitude and plotted as pressure vs. latitude, these results can be seen in Figure 5.4. In this figure another comparison is made for temperature. Once again a snapshot has been made for one point in time, therefore the details may vary based upon the weather conditions. Highest temperatures are reached at the surface in the tropics and high up in the stratosphere. From the difference plots it can be seen how for a pressure ranging from 1000 to 100 hPa the forecast quality is good, with differences ranging between ± 1 degree even for a 72 hour forecast. For pressures below 100 hPa however the forecast quality is significantly reduced, with 1 hPa being the most extreme case, where differences of up to 5 degrees can be seen for a 6 hour forecast. This difference between GraphCast and ERA5 is due to GraphCast being trained to prioritize the forecast quality close to the surface over the stratosphere [17].

Another comparison that can be made is through the RMSE at various pressure levels. A comparison has been made by calculating the normalised RMSE of 12 timesteps over a 12 month period, using ERA5 data as reference. This combination of 12 timesteps was done to reduce errors caused by specific weather phenomena. The RMSE has been normalised by dividing the RMSE by the mean of reference data from ERA5. The data was normalised to allow for a better comparison. For this comparison all atmospheric variables from GraphCast have been included in the comparison, starting off with a more detailed analysis for the temperature to discuss the general trends. This can be seen in Figure 5.5.

From the top left plot it can be seen that the normalised RMSE starts at 0 for a forecasting time of 0 hours.

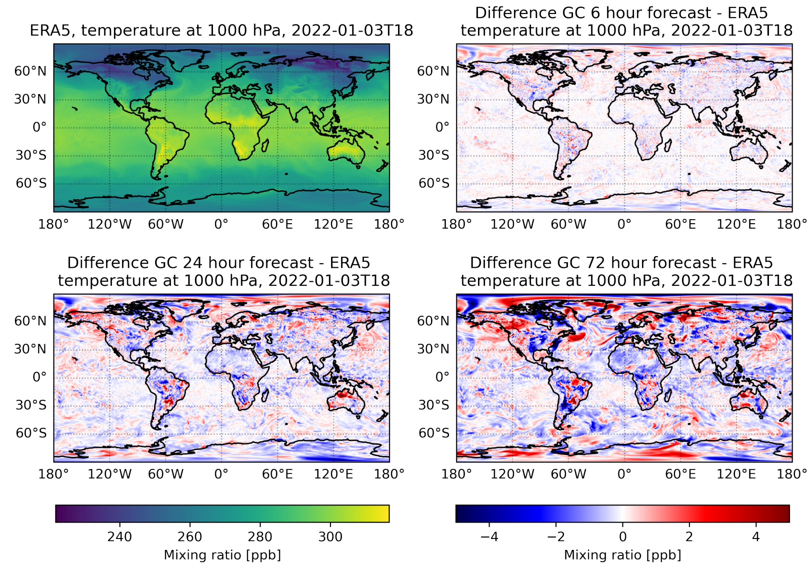

ERA5, temperature at 1000 hPa, 2022-01-03T18
Difference GC 6 hour forecast - ERA5
temperature at 1000 hPa, 2022-01-03T18
Difference GC 24 hour forecast - ERA5
temperature at 1000 hPa, 2022-01-03T18
Difference GC 72 hour forecast - ERA5
temperature at 1000 hPa, 2022-01-03T18
60°N
30°N
0°
30°S
60°S
180°
120°W
60°W
0°
60°E
120°E
180°
240
260
280
300
Mixing ratio [ppb]
−4
−2
0
2
4
Mixing ratio [ppb]

Figure 5.3: A comparison of the temperature at 1000 hPa at 2022-01-03T18, for forecasting lengths of 6, 24 and 72 hours.

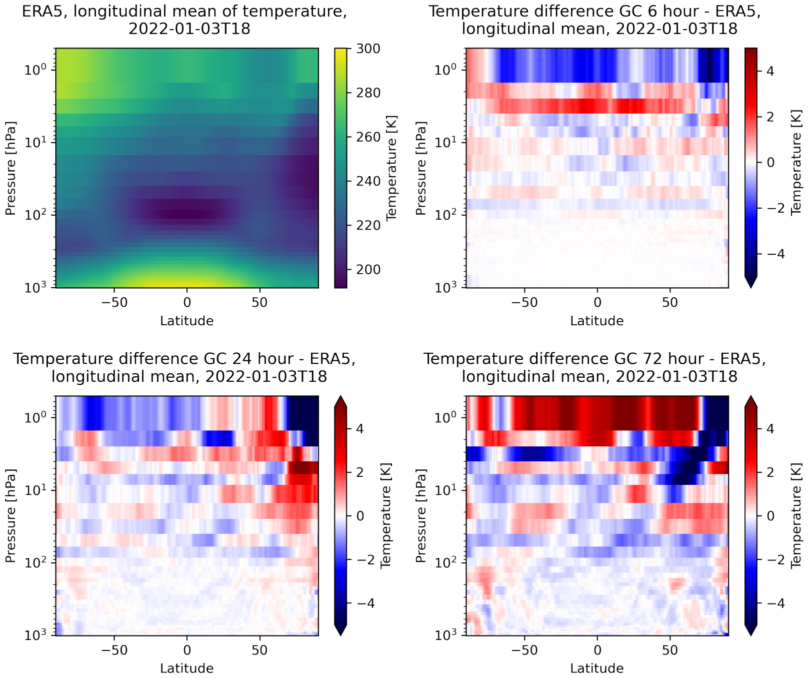


Figure 5.4: Comparison of longitudinal mean of temperature, latitude vs. pressure on 2022-01-03T18. Absolute temperature from ERA5 is shown in the top left graph, followed by the difference to a 6, 24 and 72 hour GraphCast forecast top right, bottom left and bottom right graphs respectively.

For the first forecasting step the normalised RMSE rapidly increases to a value ranging from 0.08% - 0.12%, depending on the pressure level. After this initial increase, the normalised RMSE increases at a reduced rate, ending with an RMSE ranging from 0.4% to 0.5% for a 72 hour forecast. Another peculiarity that can be seen is the stepping pattern. A similar trend could be seen in Figure 4.3, this pattern is an artifact of training on ERA5 data, which uses a 12 hour assimilation window [20].

Based upon GraphCast's prioritization for lower altitudes, a correlation between pressure and RMSE was expected. However this correlation is not apparent in Figure 5.5. For the vertical velocity and specific humidity the highest RMSE can be seen at the lowest pressures, however for the temperature and wind speeds this relation is not seen. Part of the reason could be due to the lowest pressure in this comparison still being relatively high. In Figure 5.4 it can be seen that the quality of the forecast starts to reduce for altitudes above 100 hPa, whereas the highest altitude in this comparison is at 500 hPa. Another possible contribution to the absence of correlation to pressure could be that weather phenomena occurring at specific pressure levels exhibit varying degrees of forecast difficulty for GraphCast, which may contribute to the observed reduction in dependency on pressure levels.

For usage in TM5, the horizontal and vertical velocities are thought to be the most influential variables. A comparison has therefore been made to the error margin of ERA5 in order to determine how GraphCast's error relates to the inherent error of ERA5. For the horizontal velocities it can be seen in [6] and [29] that the RMSE of horizontal velocities ranges from 1.5 to 2 m/s in ERA5, the RMSE of GraphCast ranges from 1 to 3.5 m/s, which is still considered close enough to ERA5's error. The vertical velocity is a less reliable variable. Studies indicate an error margin in ERA5 ranging from 10% to 50% of its mean depending on conditions [29]. From the bottom row in the figure it can be seen that the RMSE is over 50% for all conditions, even going up to 150% for a 72 hour forecast at 500 hPa.

5.3. TM5 DATA PREPARATION

TM5 requires meteorological data in a different form than is generated by GraphCast. Processing steps are required to adapt GraphCast's output to TM5's input, which is shown in blue in Figure 5.1. In this section the processing steps will be broken down in three, where each step will be elaborated further in its respective subsection. Figure 5.1 has been updated into Figure 5.6 where the preprocessing steps has been divided into three steps, being coarsening, model level conversion, and massflux calculation.

5.3.1. DATA COARSENING

From Figure 5.6 it can be seen that the first preprocessing step is coarsening the data. By starting off with coarsening the dataset, the size of the dataset is shrunk by a factor 16, significantly speeding up the remaining calculations. A downside of starting with the coarsening step is that data is lost, resulting in a slight reduction in data quality for remaining calculations. A schematic overview of the coarsening can be seen in Figure 5.7. The gridpoints are represented as green crosses. The data is averaged over 0, 0.25, 0.5 and 0.75 degrees in the longitudinal and latitudinal direction to create a 1 degree horizontal resolution required for TM5. The center of the new coarsened gridpoint then lies at 0.375, 0.375 degrees. This grid coarsening will be done in the same manner for all output variables.

After the grid coarsening, a translating step is performed for compatibility to TM5. GraphCast's grid is defined from 0 to 359.75 degrees longitudinally, and 90 to -90 latitudinally. TM5 on the other hand is defined as -180 to 180 degrees longitudinally and -90 to 90 latitudinally. This requires an extra step in which GraphCast's output is re-arranged to fit TM5's grid definition.

5.3.2. CONVERSION TO MODEL LEVELS

The second step in the preprocessing procedure is to convert the pressure levels from GraphCast to model levels. GraphCast provides its output in pressure levels, while TM5 uses model levels for its vertical discretization, hence the pressure levels should be converted to model levels. New model levels have been defined that closely match the 37 pressure levels from GraphCast based upon the 137 model levels defined by ECMWF [30]. For defining model levels, the pressure of all 138 model hybrid interface layers has been evaluated using a standard pressure of 1000 hPa, and the a and b coefficients from the ECMWF model level documentation in equation 3.1 [30]. Once the pressure at all hybrid layers is known, suitable hybrid layers should be selected,

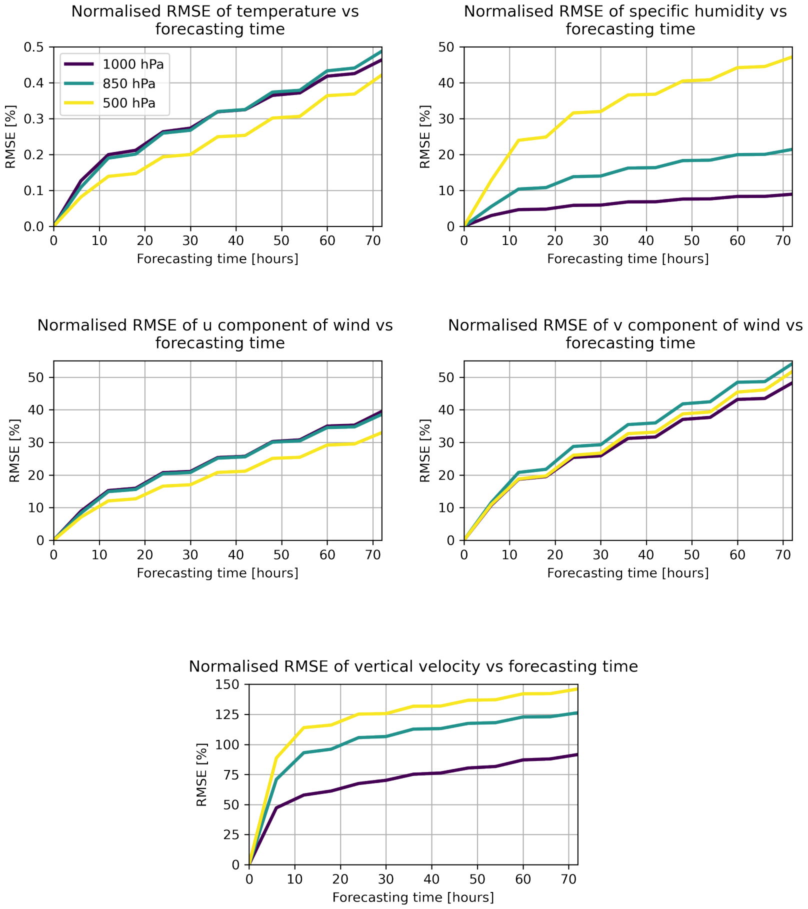


Figure 5.5: Normalised RMSE vs. forecasting time of GraphCast's output atmospheric variables. The RMSE has been determined through combining the 3rd of every month 18.00 UTC for 12 timesteps.

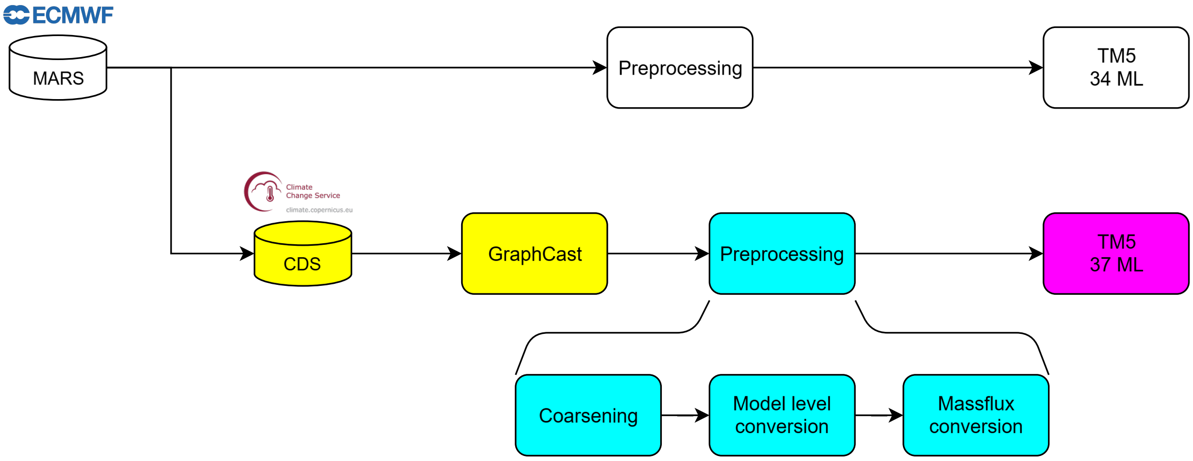


Figure 5.6: An updated representation of Figure 5.1 dividing the preprocessing step in the three steps that consists of coarsening, model level conversion, and massflux conversion.

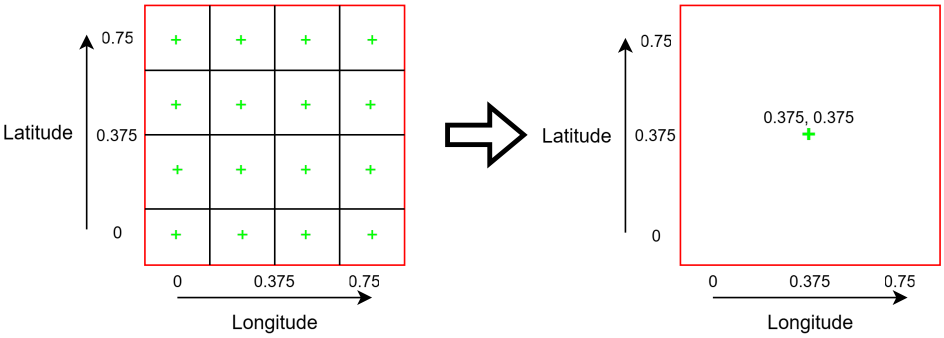


Figure 5.7: Horizontal coarsening , with 16 grid points (green) on the left hand of the figure, coarsened to one grid point on the right part of the figure.

which was done by selecting the hybrid layers to be in between the pressure levels. E.g. for pressure levels 950 and 1000 hPa, the hybrid layer closest to 975 hPa would be sought. This was done for all pressure levels, after which the series of a and b coefficients seen in table A.4 has been derived. The pressures of the model levels can then be calculated using equation 3.2.

The newly defined model levels match the pressure levels from GraphCast reasonably well at a surface pressure of 1000 hPa. However when the surface pressure changes, the model levels will no longer match the pressure levels, making the initial definition merely a baseline. In order to ensure that the pressure levels translate well to the newly defined model levels for all surface pressures, pressure level data will be linearly interpolated to the new model levels. The pressure of the model levels is determined over the entire spatial grid using equations 3.1 and 3.2, taking the actual surface pressure, along with the earlier defined a and b coefficients.

Close to the surface, magnitudes of the variables to be interpolated can change rapidly. This rapid change is not always captured by the pressure levels, causing a mismatch close to the surface as can be seen on the top two figures in Figure 5.8. On the top left plot of the figure the difference between interpolated data at model level 37 and ERA5 data at model level 135 (which is at a similar altitude as converted model level 37) can be seen. From the figure it can be seen how there are large deviations of up to 5 degrees Kelvin on the oceans, as well as a difference of up to 10 degrees Kelvin on the North Pole. The plot on the top right shows the temperature vs. pressure for the original ERA5 model levels and the interpolated model levels for a latitude of 78 degrees and longitude of 165 degrees, this point is highlighted as the green cross on the left plots. This location was chosen based on its large error of 10 degrees Kelvin at the surface. From the figure it can be seen how for a pressure higher than 1000 hPa the converted model level data is not able to match the rapid temperature drop observed in the original ERA5 model levels.

The error has been reduced by including surface level data to the interpolation, allowing for an extra data point to capture strong vertical gradients close to the surface. Suitable surface variables are ’2m temperature’, ’10m u-component of wind’ and ’10m v-component of wind’. This addition has been done by defining the pressure of the surface variable to be the highest of either the surface pressure or 1001 hPa, to ensure that the surface variable will be above the 1000 hPa pressure level at all times. With the surface variables included, all variables can now be interpolated to the model level pressures. When looking at the bottom two graphs of Figure 5.8 it can be seen how the difference between ERA5 and the converted model levels to 137 model levels has improved significantly. The mismatch on the oceans, as well as the 10 degree offset on the North Pole has been reduced to less than a degree. When looking at the pressure vs. temperature plot at the selected location, it can be seen how the converted model levels now better capture the rapid temperature drop near the surface. The conversion is still not perfect, temperature differences can still be seen in regions of elevation, such as Greenland and the South Pole. This error is likely caused by the inclusion of surface level data at the bottom of the dataset, instead of including it at its respective pressure. A possible improvement could therefore be to discard all data at pressures levels higher than the surface pressure and include the surface variables at the actual surface pressure, this option is left for future work.

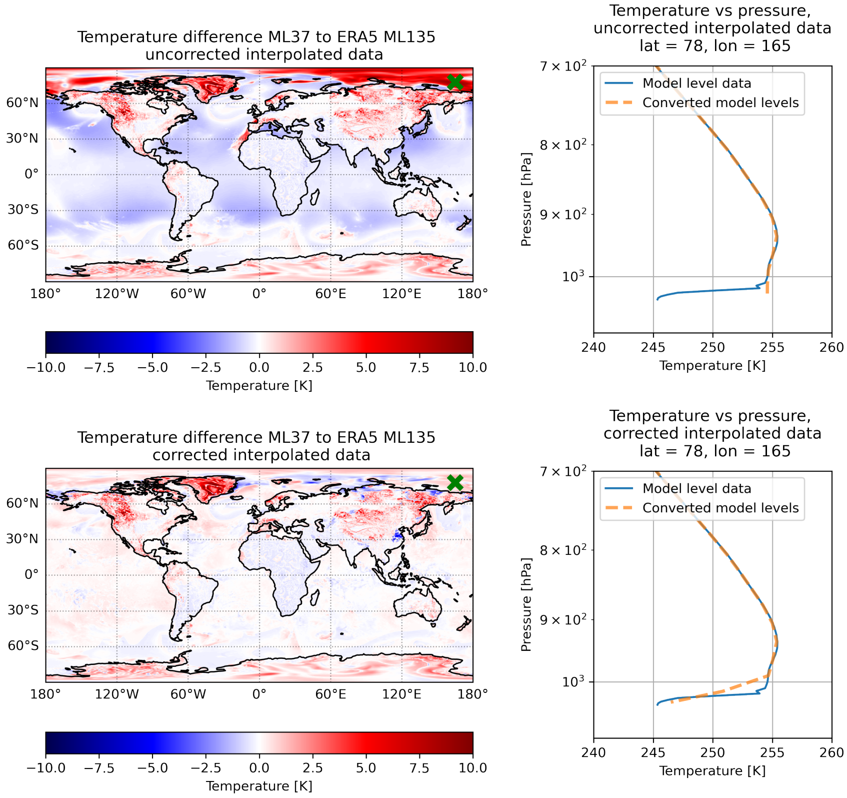


Figure 5.8: Temperature near surface for converted model levels. Top graphs without the use of surface variables, bottom graphs with the use of surface variables. The green cross in the left figures indicates the location used in the right plots.

5.3.3. Model level conversion error

In order to determine how the converted model levels correlate to the original 137 model levels, a comparison has been made between the two by the converted variables against pressure. This comparison has been made at a latitude and longitude of 0 degrees for a single point in time. Different locations or time will change the results, though the general trends will still apply. In Figure 5.9 two plots can be seen showing the converted model levels for temperature and the u-component of wind compared against the original 137 model levels. From the figure it can be seen how the converted model levels closely match the model levels, particularly from 1000 to 100 hPa. This strong correlation is due to the GraphCast having a high vertical resolution in this region, allowing for sufficient datapoints for the interpolation.

From 1 to 100 hPa it can be seen how the correlation decreases with reducing pressure. Particularly on the right plot the converted data shows reduced correlation. At 25, 9 and 1 hPa it can be seen how the converted levels do not match the velocity peaks of the model levels, with differences of up to 10 m/s. This difference is due to a combination of the large variability in the data, as well as the reduced vertical resolution of GraphCast above 100 hPa. Below 1 hPa no converted model levels exists anymore due to 1 hPa being the lowest pressure level of GraphCast. Finally as discussed earlier in this section, in regions of strong vertical gradients close to the surface, a mismatch can be seen that has been partly corrected with the usage of surface level data, though a miscorrelation still exists.

Another point that should be taken into account is that the current TM5 configuration used for the emission methane inversions is a 34-layer coarsening of the original 137 meteorological levels. Thus a similar error is likely to be present already. This means that the error found here is unlikely to significantly affect the

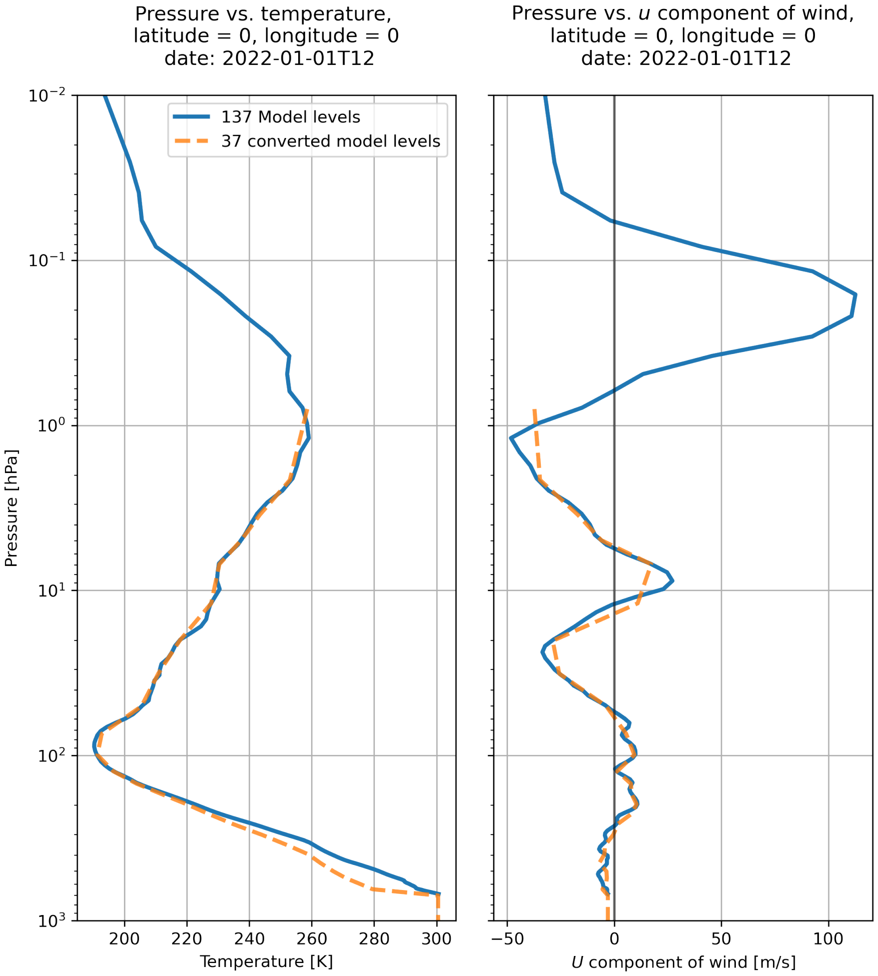


Figure 5.9: A comparison of model level data to converted model levels for temperature and wind velocity in u direction.

output of TM5. Despite the reduced correlation at pressures below 100 hPa, the correlation was still deemed sufficient, hence the error was accepted for this study.

5.3.4. Massflux calculation

With the coarsening and model level conversion completed, the third step in the preprocessing phase is to convert GraphCast's wind velocities to massfluxes. TM5 is defined on grid boxes, with each grid box filled with air. Air is here the mixture of N_2, O_2 and H_2O, among with "traces" of trace-gasses such as CO_2, CH_4, CO, NO_2, O_3, etc. The trace gasses are transported with the rest of the air-mixture between boxes, in the direction of the wind. The transport is implemented with a finite-volume scheme, that requires the air massfluxes (in kg/s) through the surfaces of the boxes. GraphCast generates horizontal wind velocities in east- and northern direction in m/s, as well as the vertical wind velocity in Pa/s, hence the need for the conversion of wind velocities to massfluxes.

In order to convert the wind velocities to massfluxes, the wind velocities should be integrated over the cell boundary. The grid cells are represented using the latitude (ϕ), longitude (λ), and vertical coordinate (η). This integration should be performed for both horizontal, as well as vertical velocities. The horizontal velocities are generated by GraphCast as wind speeds in m/s, which can be converted using the following equation

[31]:

$$\Phi = \frac{1}{g} \iint \mathbf{n} \cdot \mathbf{v} \frac{\partial p}{\partial \eta} d\eta \, dl \tag{5.2}$$

In here **n** represents the normal vector of the surface that is integrated over, **v** is the wind velocity at the surface, η represents the vertical coordinate, with $\frac{\partial p}{\partial \eta}$ the pressure change over the cell height. dl is used to represent the horizontal coordinate of the cell that should be integrated. Finally g is the gravitational constant. Due to the use of model levels the vertical position is linked to the pressure. For this reason the vertical integration can be represented using the following equation [31]:

$$\int_{\eta_{k-1/2}}^{\eta_{k+1/2}} \frac{\partial p}{\partial \eta} d\eta \approx \Delta P_k \tag{5.3}$$

In this equation Δp represents the pressure change over the height of the cell. $\eta_{k-1/2}$ and $\eta_{k+1/2}$ are the interface points between the vertical grid cells and can be determined using the hybrid pressure levels. Model levels are defined from the top of the atmosphere to the Earth's surface, such that pressure increases with increasing model levels. For the east/west massflux, the length over which the wind velocity should be integrated is the latitudinal displacement, which can be represented by $R\Delta\phi$, R is the radius of the earth, with $\Delta\phi$ representing the width of the cell in radians. This brings the final equation for the east/west massflux to:

$$\Phi^u = \frac{R}{g} \int u(\lambda, \phi, \eta) \Delta P \, d\phi \tag{5.4}$$

For the north/south massfluxes, the equation needs to be slightly altered due to the increase of cell density when approaching the north and south poles. Due to this increase in cell density, the size of the gridbox through which the air mass flows reduces proportionally. For this reason a multiplication with the cosine of the latitude is required, resulting in the following equation:

$$\Phi^v = \frac{R}{g} \int v(\lambda, \phi, \eta) \Delta P \, d\lambda \cos\phi \tag{5.5}$$

With the horizontal massfluxes known, one additional translational step is required for TM5. The velocities are defined at the mid points, however for TM5 these points need to be translated to the interface points, which can be done by calculating the mean of two interfacing grid cells. A slight error is introduced by using the coarsened data for translation to the interface points. Performing this translation before the coarsening would improve the quality of the data, though this has not been implemented yet. On the grid boundaries an alternative should be found. For the 0 and 360 degrees longitudinal grid, the neighboring cells are 360 and 0 degrees respectively. On the latitudinal grid, two new rows of grid cells should be defined on -91 and 91 degrees with a massflux in v of 0 kg/s. The reason for these cells not passing any massflux is due to the cells at 90 and -90 degrees becoming infinitely thin at top and bottom respectively.

The calculation of the vertical massflux is more complicated, it requires an integration of the vertical velocity over the surface area of the grid box. For the ECMWF meteorological model, the vertical flux is defined as [9]:

$$\dot{\eta} \frac{\partial p}{\partial \eta} = -\frac{\partial p}{\partial t} - \int_0^{\eta} \nabla \cdot \left(\mathbf{v_H} \frac{\partial p}{\partial \eta} \right) \tag{5.6}$$

GraphCast does not put out the vertical flux directly, however it does put out the pressure coordinate vertical velocity, which in turn is given as [9]:

$$\omega = -\int_0^{\eta} \nabla \cdot \left(\mathbf{v_H} \frac{\partial p}{\partial \eta} \right) + \mathbf{v_H} \cdot \nabla P \tag{5.7}$$

Substituting equation 5.7 into equation 5.6 results in the following equation:

$$\dot{\eta} \frac{\partial p}{\partial \eta} = -\frac{\partial p}{\partial t} + \omega - \mathbf{v_H} \cdot \nabla P \tag{5.8}$$

Equation 5.8 can then be integrated over its grid surface to arrive at the final equation for the massflux:

$$\Phi^w = \frac{R^2}{g} \iint_{A_{ij}} \left(-\frac{\partial p}{\partial t} + \omega - \mathbf{v_H} \cdot \nabla P \right) d\phi \, d\lambda \cos\phi \tag{5.9}$$

Equation 5.9 contains three important elements: vertical velocity ω, pressure change with time $\frac{\partial p}{\partial t}$, and massflux due to cell curvature $\mathbf{v_H} \cdot \nabla P$, all in Pa s^{-1}. The vertical velocity is a direct output of Graphcast, however the remaining two terms should be calculated. The pressure change with time can be calculated using equation 3.1, which defines the pressure at the given model levels. The velocity in Pa s^{-1} can then be derived through the differentiation of the pressure change with time. Out of the given equations, the surface pressure is the only part that is a function of time, hence the term with coefficient a can be neglected. This results in the following equation:

$$\frac{\partial p}{\partial t} = \frac{\frac{\partial p_s}{\partial t}\left(b_{k+1/2} + b_{k-1/2}\right)}{2} \tag{5.10}$$

The final part of equation 5.2 that needs to be solved is the $\mathbf{v_H} \cdot \nabla P$ term. This term requires the horizontal pressure gradient, which should be multiplied with the velocity in the direction of the gradient. For level k, this leads to the following equation:

$$V_k \nabla P_k = u_k \frac{\partial P}{\partial R\lambda} + v_k \frac{\partial P}{\partial R\phi} \tag{5.11}$$

Similar to the horizontal massflux, the vertical massflux should also be translated to its interface points. The level boundaries can be found by calculating the mean of two interfacing model levels. Fluxes through the surface and top of the model are 0.

5.3.5. MASSFLUX CONVERSION ERROR

To assess the validity of the conversion, a comparison has been made between the converted massfluxes to the massflux data on model levels as retrieved from ERA5. This comparison will be performed based upon the two datasets for various model levels, as well as a comparison to pressure for various locations. The massflow calculation will be based upon the vertically interpolated data, since this will be the dataset that most closely represents the dataset that will be used for the project. A downside of using this dataset is that the errors caused by the coarsening and vertical interpolation will both be present.

A comparison to 137 model levels cannot be done directly due to the dependency between massflux and model layer thickness. In order to allow for a direct comparison, the 137 model layers have been merged to 37 model layers. Figure 5.10 shows on the left hand side the absolute massflux of the reference data, and the difference between both datasets on the right. The top graphs show the massflux for model level 37 (closest to the surface), being ML37, whereas the bottom row shows the massflux higher up in the atmosphere, at ML 18. As can be seen in the figure, the postprocessed data matches the reference data pretty well. The error is fairly evenly spread over the entire surface. This strong resemblance can be further seen when looking at the difference in magnitude between the two datasets, where the magnitude of the difference is roughly a factor 10 smaller than the absolute massflux for both model levels shown.

Another comparison can be made when plotting the massflux vs. altitude, which was done for latitude and longitude both at 0 degrees, as is seen in Figure 5.11. From the figures it can be seen how the postprocessed massfluxes correlate well to the reference massfluxes for most pressures. Differences can be seen in levels with strong gradients in the vertical coordinate, such as can be seen at a pressure of 500 hPa for the massflux in u, where the postprocessed massflux is roughly 1.5e8 kg/s higher than the reference massflux. This difference is likely caused by mismatch introduced in the coarsening step. Overall the calculation from horizontal velocities to massfluxes can be considered to be rather accurate.

The final parameter of interest is the vertical velocity, which once again proves to be the most difficult variable. Figure 5.12 shows a comparison of the vertical massflux through the interface between model levels 36 and 37, here named model level 36.5 for the three leftmost figures. The three figures on the right hand side show a comparison for model level 20.5. As can be seen from the graphs, the correlation between the two datasets is not great. Particularly for model level 36.5, as seen on the left hand side of the figure, a completely different pattern is visible. These differences appear to be particularly strong in regions of elevation, as can be seen in the Alps and the Andes. Such a dependency on altitude gradient suggests an incorrect correction for the vertical velocity due to model levels following the rapidly changing altitude of the terrain, although this has not been verified. Another reason could be due to the different way in which the vertical massflux has been calculated, which is partly described in [31]. The accumulation of these differences could significantly alter the data, causing a large mismatch. When moving higher into the atmosphere, for example as can

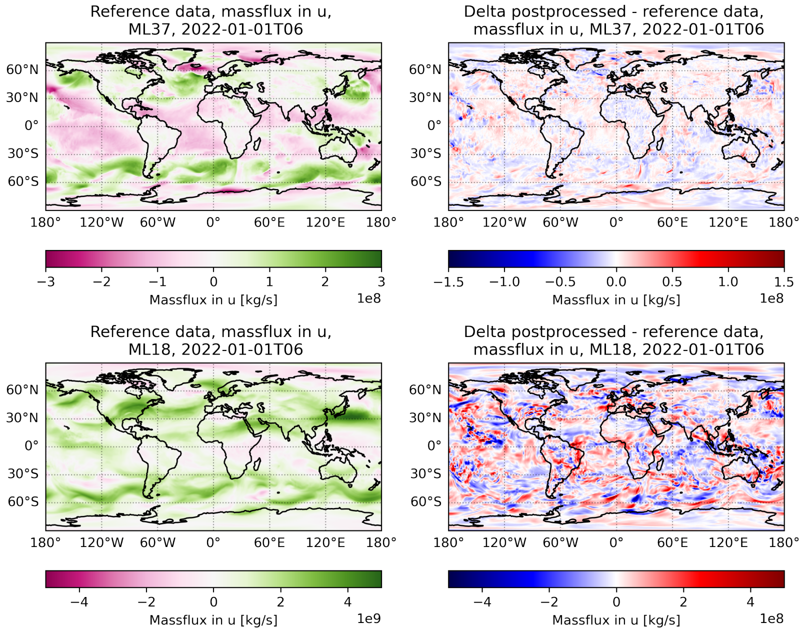

Reference data, massflux in u, ML37, 2022-01-01T06
Delta postprocessed - reference data, massflux in u, ML37, 2022-01-01T06
Reference data, massflux in u, ML18, 2022-01-01T06
Delta postprocessed - reference data, massflux in u, ML18, 2022-01-01T06
60°N
30°N
0°
30°S
60°S
180°
120°W
60°W
0°
60°E
120°E
180°
Massflux in u [kg/s]
1e8
1e9
−3
−2
−1
0
1
2
3
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
−4
−2
0
2
4

Figure 5.10: Comparison between reference data and postprocessed data for massflux in u direction. The two left plots show the absolute massflux for model levels 37 and 18 from the reference data. The two figures show the difference between reference data and postprocessed data for model levels 37 and 18.

be seen from the model level 20.5 figures, the differences reduce and a better correlation can be seen. The data still does not correlate as good as the horizontal massflux though, with deviations in the same order of magnitude as the absolute dataset.

Another comparison can be made by plotting the vertical velocity against altitude in Figure 5.13. The vertical massflux again does not correlate well. When looking at the figure, in some instance the postprocessed vertical massflux appears to follow a similar trend as the reference data, such as in the region of 1000 to 100 hPa in the right plot. In other instances however the postprocessed data follows a completely different trend, such as in the range of 10 to 300 hPa on the left plot. Whether this large difference will cause a significant error in simulating CH_4 transport should be derived from a test in TM5. A comparison in TM5 with and without the usage of the vertical velocity will reveal whether this difference is acceptable.

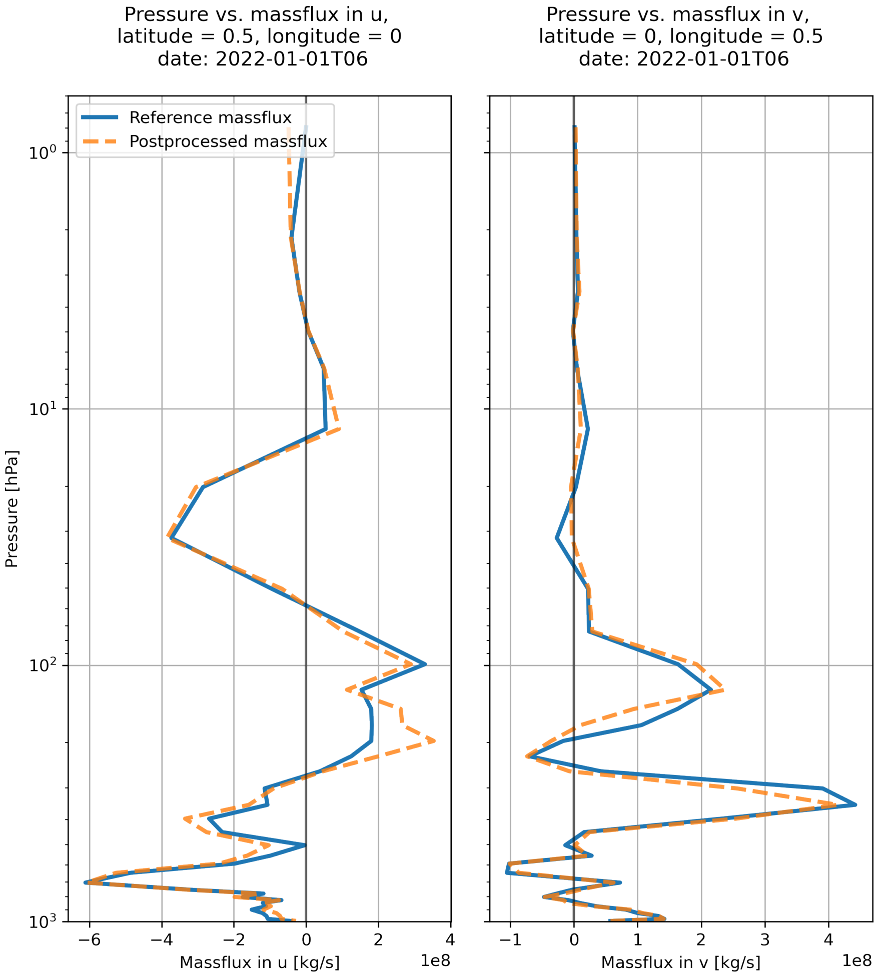


Figure 5.11: The left plot shows the massflux in u vs. pressure, whereas the right figure shows the massflux in v vs. pressure. The figures show the massflux at the western and southern boundary of the cell at a latitude and longitude of 0.5.

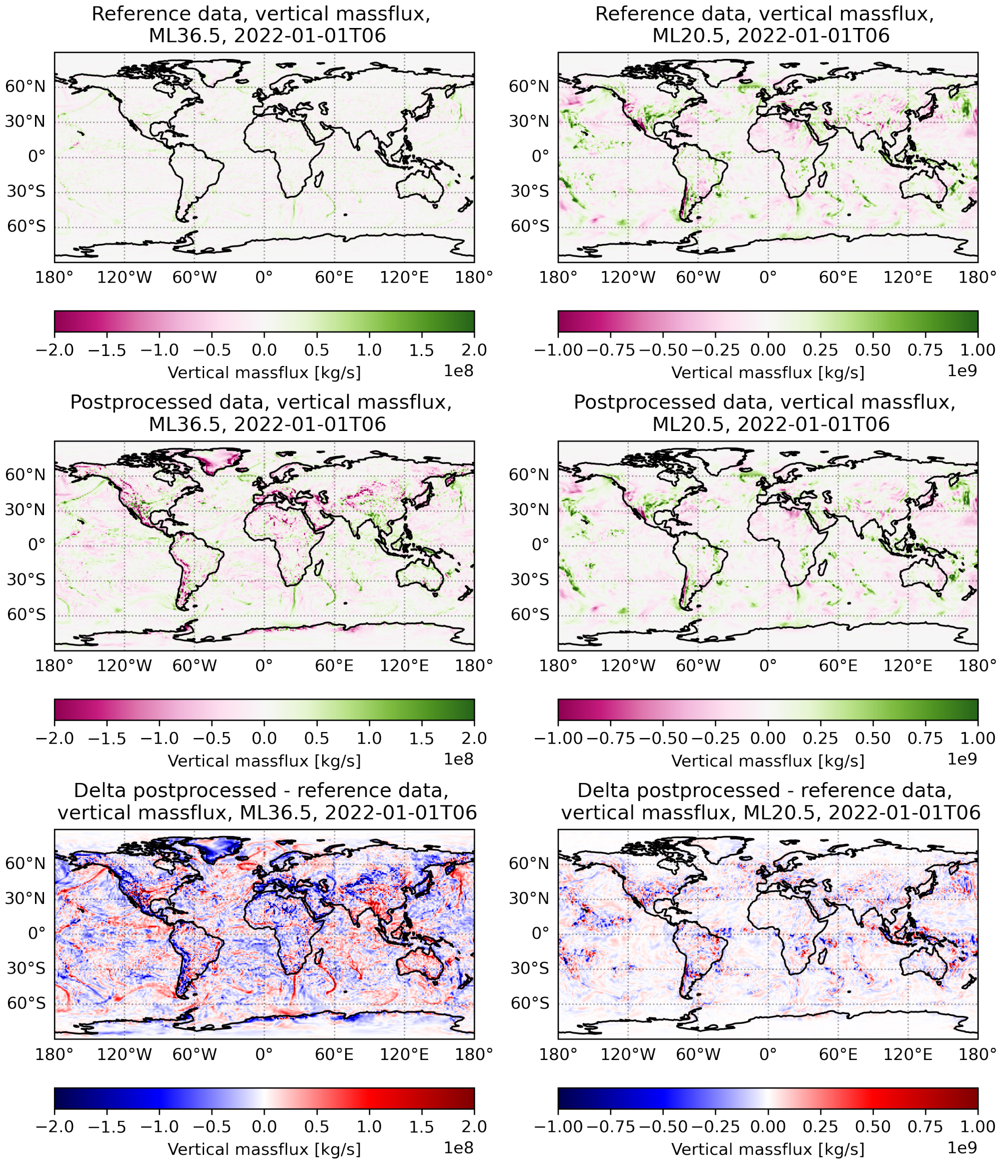

Reference data, vertical massflux, ML36.5, 2022-01-01T06
Reference data, vertical massflux, ML20.5, 2022-01-01T06
Postprocessed data, vertical massflux, ML36.5, 2022-01-01T06
Postprocessed data, vertical massflux, ML20.5, 2022-01-01T06
Delta postprocessed - reference data, vertical massflux, ML36.5, 2022-01-01T06
Delta postprocessed - reference data, vertical massflux, ML20.5, 2022-01-01T06
60°N
30°N
0°
30°S
60°S
180°
120°W
60°W
0°
60°E
120°E
180°
−2.0 −1.5 −1.0 −0.5 0.0 0.5 1.0 1.5 2.0
1e8
−1.00 −0.75 −0.50 −0.25 0.00 0.25 0.50 0.75 1.00
1e9
Vertical massflux [kg/s]

Figure 5.12: Comparison of reference data and postprocessed data for vertical massflux. The top two rows show the absolute values of vertical massflux for reference and postprocessed data. The bottom row shows the difference between reference and postprocessed data for model levels 36.5 and 20.5.

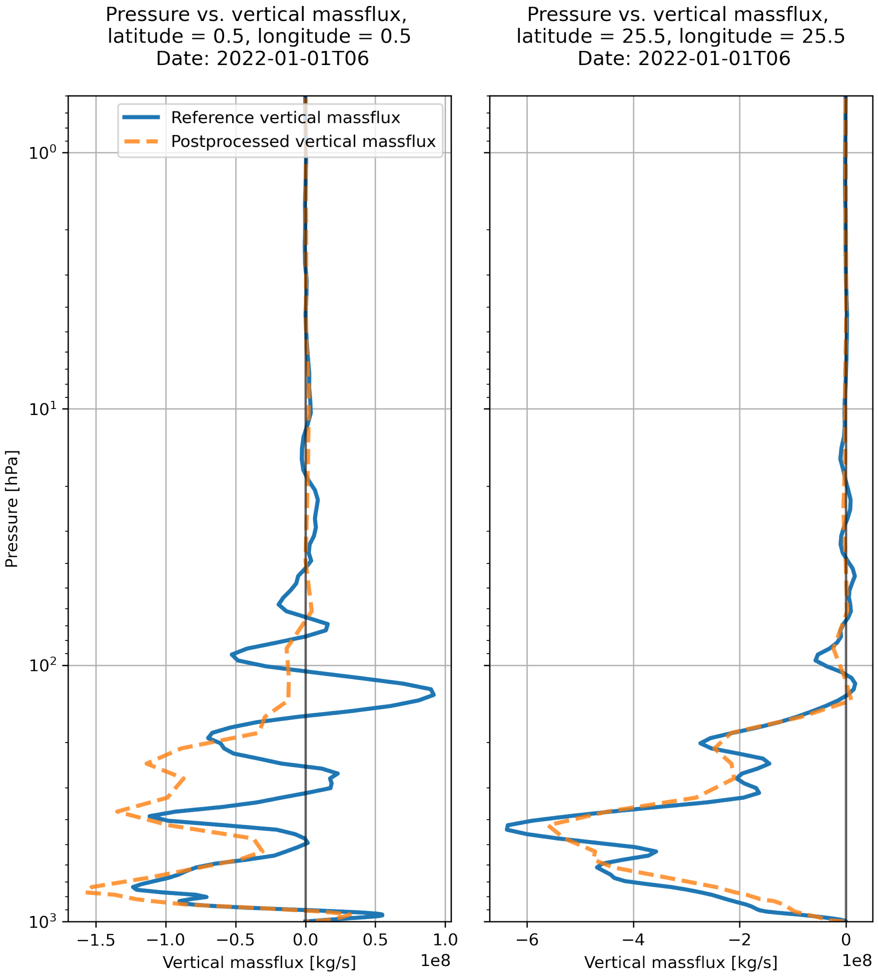


Figure 5.13: A comparison between reference data and postprocessed data for the vertical massflux. The left plot shows the comparison at a latitude and longitude of 0.5, whereas the right plot shows the comparison for a latitude and longitude of 25.5.

6

TM5 METHANE SIMULATIONS

This chapter will describe the final step in the process, which is to fully verify and execute the newly defined pipeline, and use AI generated meteorological data for TM5. In Section 6.1 the process verifying the new pipeline and integrating GraphCast into TM5 will be discussed. In Section 6.2 a reference simulation will be performed using the 34 model level version of TM5. Section 6.3 will discuss the impact of a switch to 37 model levels in TM5. The new preprocessing steps will be included in TM5 in Section 6.4. The final step will be to include GraphCast forecasts in TM5 and assess the impact on its accuracy, which will be done in Section 6.5.

6.1. Integration process of GraphCast in TM5

Before running any simulations in TM5, the process of verifying the new pipeline and assessing the results will be discussed in this section. The process will be split up in four parts to determine the impact of the new model levels, the processing step and GraphCast individually. This breakdown will be discussed in Subsection 6.1.1. Subsequently two quality metrics will be defined on which the pipelines will be defined in Subsection 6.1.2.

6.1.1. Process for verifying new pipeline and GraphCast

Four steps have been defined to verify the new pipeline, a graphical representation of these steps can be seen in Figure 6.1. The first step is to define a reference simulation based on the 34 model level version of TM5. This step can be seen in white. With the reference simulation defined, a new version of TM5 can be introduced using the newly defined 37 model levels, shown as orange. Both versions will acquire their data from the MARS archive on 137 model levels, which is subsequently mapped to the respective model levels required to run TM5. The third step, shown in green, is to introduce variables that were derived using the new pipeline. In this step data at pressure levels is downloaded from the CDS, which is converted to model levels and used in TM5.

Once all variables have been evaluated, a selection will be made of the variables that do not significantly alter TM5's output. This selection will serve as the new reference simulation for the GraphCast forecast runs. The final step is to introduce GraphCast into the pipeline, which can be seen in pink. TM5 will initially simulate a short forecasting time, after which the forecasting time can be progressively increased until three day forecasts are processed. The impact of the forecasting length can then be evaluated. A naming conventions has been defined for the various simulations, these are illustrated in the figure accompanying each pipeline.

6.1.2. Quality metric definition

The first metric is a time series of the globally averaged CH_4 mixing ratio. The second metric is a comparison at two measurement stations, where TM5's output can be compared to flask measurement data. The globally averaged CH_4 mixing ratio will be determined using the following equation:

$$MR = \frac{\sum^{Earth} (\chi_{\mathrm{CH}_4} \cdot M_{air})}{\sum^{Earth} M_{air}} \tag{6.1}$$

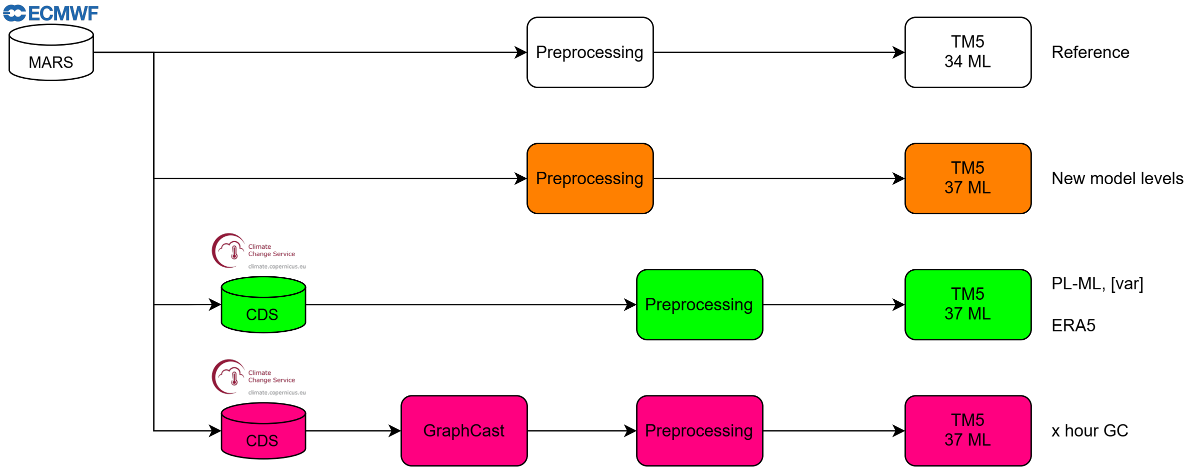


Figure 6.1: Four different TM5 pipelines have been defined to verify each aspect of the new process. The white pipeline shows the original pipeline on 34 model levels. The orange pipeline shows the new pipeline for a TM5 version on 37 model levels, still using meteorological data from the original MARS archive. The green pipeline shows the 37 model level version of TM5 using the new processing method. Finally in pink GraphCast is included in the pipeline.

In here, χ_{CH_4} is the mixing ratio of methane in parts per billion for each grid cell. M_{air} is the airmass for each gridcell. The multiplication of the airmass and methane concentration will be summed over every cell in the domain, which can subsequently be divided by the sum of the airmass to determine the global mixing ratio. The airmass can be calculated using the following equation:

$$M_{air} = A \cdot dP/g \tag{6.2}$$

In the equation above, A is the surface area of the gridcell, which is multiplied by the pressure difference between the top and bottom of the cell (dP). This is then divided by the gravitational constant to arrive at the airmass in kg. The mixing ratio will be determined for every timestep in a one year period. This metric is useful to compare two simulations on overall differences and trends. The trends should remain consistent and comparable across the different cases, without displaying irregular patterns or systematic drifts over time.

To verify the global growth and seasonal dependency of CH_4, the globally averaged mixing ratio can be compared to the left plot in Figure 6.2. The figure shows a monthly mean in CH_4 mixing ratio, this mixing ratio was determined based upon flask measurements taken at marine surface sites. Flask measurements are sample measurements where an air sample is stored in a flask and analysed on its chemical composition. These flask measurements are taken at various measurement stations around the globe, which are shown in Figure 6.2 on the right plot as the blue and red crosses. A selection of the flask measurement stations has been made based on samples that contain predominantly well-mixed marine boundary layer air, representative of a large volume of the atmosphere [32]. These locations are typically remote marine sea-level locations.

One important point of notice is that mixing ratio from Figure 6.2 is determined using marine surface sites, whereas the global mixing ratio is determined over the entire atmosphere, resulting in an indirect comparison. For this reason for the first comparison two figures will be shown. One figure showing the globally averaged mixing ratio averaged over all model levels, along with a figure showing mixing ratio averaged over model level 34, being the model level closest to the surface. Since the mixing ratio at model level 34 is acquired at a similar altitude as the measurement data, it should provide a better comparison. For this reason it will be looked into once for the validation between TM5 and measurement data. For the remainder of the study however, the averaged mixing ratio over all model levels will be used for verification, as it takes the entire domain into account.

The second verification step is to do a comparison between TM5 and flask measurement data. Two measurement stations have been chosen based upon the quality and availability of data, which can be seen on the right plot in Figure 6.2 as the red crosses. The two chosen stations are the Amundsen-Scott South Pole Station and Mt Waliguan, China. The measurement station on the South Pole is located far away from exterior methane sources that impact the mixing ratio, resulting in a clean signal with low variability and a clear

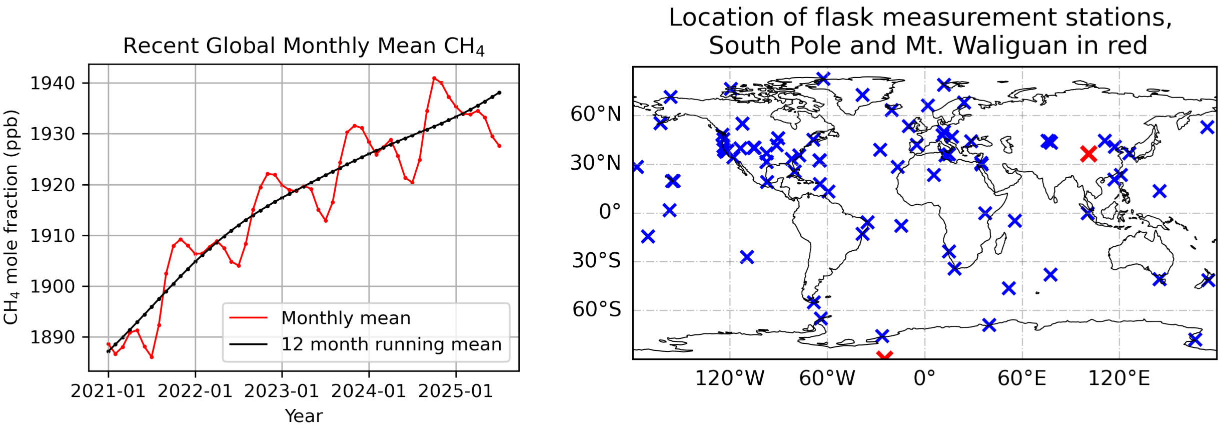


Figure 6.2: Two figures used for the flask measurements. The left figure shows the mean of the methane mixing ratio. In red the monthly mean can be seen, with a 12 month running mean in black [32]. The figure on the right shows the flask measurement stations in 2022, in red the two stations that have been chosen for further analysis can be seen.

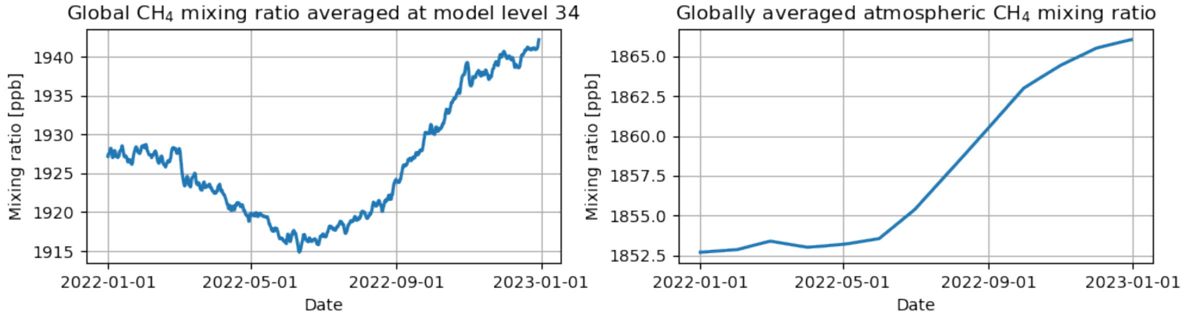


Figure 6.3: Two plots showing the global mean CH_4 mixing ratio for a one year simulation. The left plot shows the mixing ratio over model level 34 (close to the surface), the right plot shows the mixing ratio averaged over all model levels.

seasonal dependency. Mt. Waliguan on the other hand is located in South East Asia and therefore heavily influenced by exterior methane sources. The influence from these methane sources results in a significantly higher variability in the signal. TM5's can determine the CH_4 mixing ratio at these measurement station for every timestep, which can be used for comparison.

6.2. REFERENCE TO COMPARE SIMULATIONS

The globally averaged mixing ratio is presented in Figure 6.3. The left plot in the figure shows the mixing ratio for the model level 34, whereas the right plot shows the mean over all model levels. The values at model level 34 can be compared with the globally averaged monthly mean illustrated by the red curve in the left panel of Figure 6.2. This reference curve indicates a decline in the mixing ratio during the first half of the year, followed by a pronounced increase in the second half, amounting to an annual rise of approximately 13 ppb. This seasonal pattern corresponds well with the left plot in Figure 6.3. The mixing ratio begins at roughly 1927 ppb and decreases by approximately 10 ppb over the first six months. This decline is followed by a rapid increase during the remaining 6 months, until approximately 1940 ppb is reached by the end.

Since the right plot in Figure 6.3 is calculated over all model levels, its seasonal dependency is not as prevalent. The mixing ratio starts at 1853 ppb and remains stable for 6 months. After the sixth month the mixing ratio starts rising until it reaches 1866 ppb on December 31. As was discussed in Section 3.2, this pattern is due to the breakdown and emissions of CH_4 being in balance with each other during the first half of the year, after which an increase in emissions in the northern hemisphere causes the mixing ratio to increase [10].

Figure 6.4 shows the time series of observed and simulated methane mixing ratios at the two chosen measure-

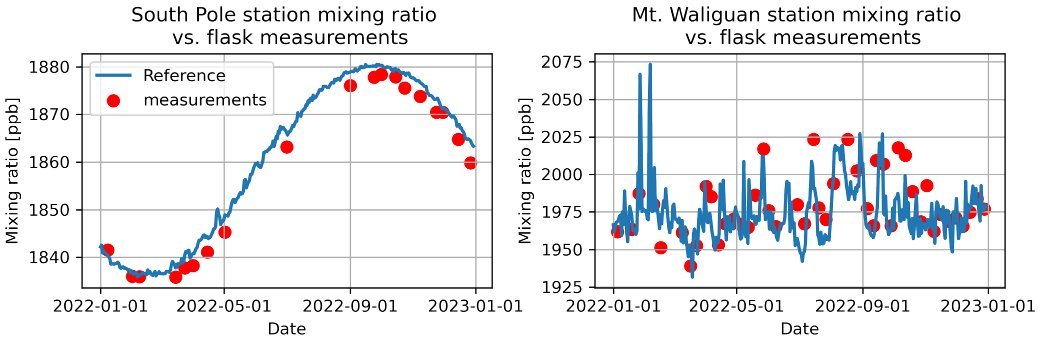


Figure 6.4: TM5 mixing ratio for the two chosen stations. On the left graph the mixing ratio on the South Pole can be seen, with the mixing ration on Mt. Waliguan on the graph on the right.

ment stations. In blue the model output from TM5 can be seen, the red dots show the sample measurements. On the South Pole 19 measurements are provided in 2022, whereas 45 measurements are present on Mt. Waliguan in 2022. A strong seasonal dependency can be seen on the South Pole, which can be seen on both the measurements as well as the simulation data. A good correlation between TM5 and the measurement data can be seen, since the difference between the two remains within 5 ppb on the South Pole. The figure on the right shows the dataset of Mt Waliguan, China, which contrary to the South Pole shows a significantly larger variability, along with a reduced seasonal dependency. This larger variability is mostly attributed to the proximity to strong methane sources in South East Asia. The correlation to measurement data has also reduced, with differences of over 20 ppb occuring occasionally.

6.3. IMPACT OF 37 MODEL LEVELS

Now that a reference run is defined, a new version of TM5 can be simulated on 37 model levels, which can be seen as the orange pipeline in Figure 6.1. The global mixing ratio has once again been determined and can be seen in Figure 6.5. In the figure the top plot shows the absolute global mixing ratio of the 34 model level reference in blue, along with the new 37 model level simulation in orange, while in the bottom plot the difference between the two simulations can be seen. The two datasets behave very similar, starting off with the same global mixing ratio, after which the mixing ratio slightly increases compared to the reference simulation, as can be seen from the increasing line on the difference plot. After a one year simulation a difference of 0.15 ppb exists between the two datasets, pointing to a longer lifetime of CH_4. A reason for this difference could be due to the change in vertical resolution. The 34 model level of TM5 has its model levels more densely distributed in the stratosphere, whereas the new model level variant has a higher resolution at altitudes below 100 hPa. This reduction in vertical resolution in the stratosphere could lead to transport mechanisms such as the Brewer-Dobson circulation to be poorly modelled [33]. A difference of 0.15 ppb however is significantly smaller than the offset to measurements that was encountered in the previous section. For longer simulation periods this difference could become problematic. Inversion studies typically have simulation periods of up to 40 years, in which a consistent drift of 0.15 ppb a year could result in a significant difference, this has not been investigated in this study.

The mixing ratio at the two chosen stations is plotted in Figure 6.6 in the top two plots, together with the difference to the 34 model level version of TM5 in the bottom plots. The new model level version of TM5 shows good correlation to the reference simulation. On the South Pole station, the difference between the two measurements stays within 1 ppb, which is significantly closer than the offset to the measurements. The difference on the South Pole appears to show a seasonal dependency, due to the increase in mixing ratio during the first four months of the year, which is followed by a reduction in mixing ratio in the remainder of the year. This difference on the South Pole is possibly caused by the reduced vertical resolution in the stratosphere. Methane concentrations on the South Pole are partially governed by stratospheric weather phenomena [33], hence the reduction in vertical resolution is likely to impact how well this mechanism is modelled. On Mt. Waliguan fairly good correlation can also be seen. The difference between the two datasets has increased compared to the South Pole, with differences varying between ± 5 ppb for most of the dataset, though due to the large variability of the dataset on Mt. Waliguan, this is to be expected. Once again the difference between the two datasets is significantly smaller than the difference between the measurement data. The impact of the

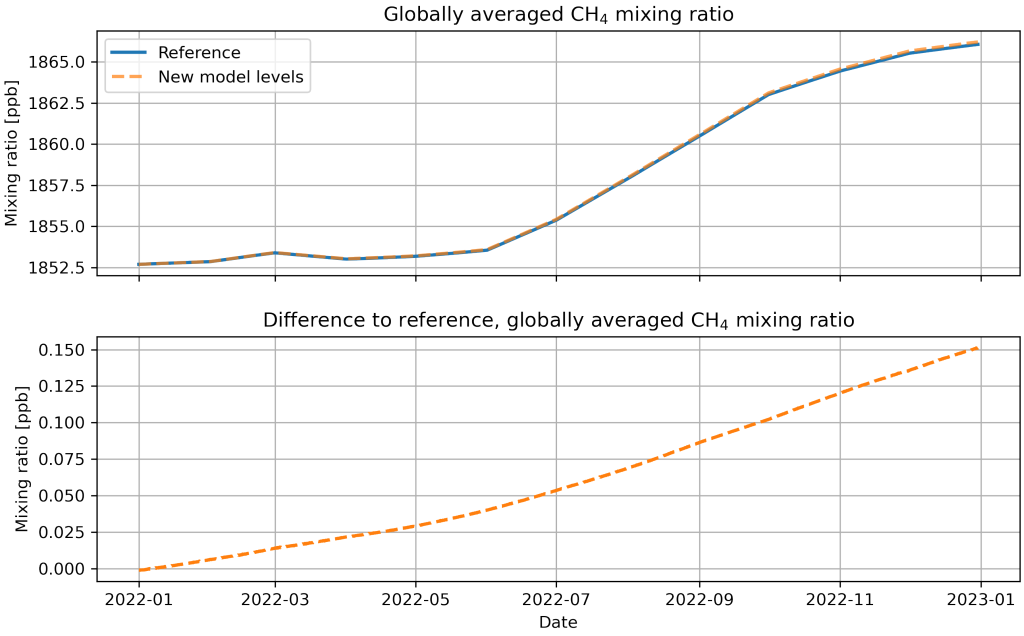


Figure 6.5: Global mean atmospheric mixing ratio of the 37 and 34 model level versions of TM5. The absolute mixing ratio can be seen in the top figure. The bottom figure shows the difference between the two versions.

change from 34 to 37 model layers is therefore considered to be of minor impact for the methane simulations.

6.4. IMPACT OF NEW METEOROLOGICAL VARIABLES

With the new version of TM5 on 37 model levels defined, the next step is to look into the green pipeline on Figure 6.1. In this step the different variables on pressure levels as downloaded from the CDS are converted and will be introduced in TM5. For the introduction of the new variables, the 37 model level version of TM5 will be used as a reference. In total the following four experiments have been conducted in this step of the process:

- PL-ML [T]
- PL-ML [q]
- PL-ML [mfuv]
- PL-ML [mfw]

PL-ML is used to denote that the meteorological data has been acquired on pressure levels and converted to model levels. The variable in between brackets is used to denote which variable is retrieved from the CDS and processed. T stands for temperature, q for humidity, and mfuv represents horizontal massflux, and mfw is used to denote vertical massflux. Other atmospheric data fields required for TM5 are not considered here, which are the convective and diffusive fields. These can be calculated from the aforementioned 3D datasets, however this was not yet included in this study. The variables will be introduced one-by-one, while the remaining variables are still acquired from the MARS archive on model levels. Separating these steps allows for evaluating the impact of each variable separately.

After all simulations have been performed, the global mixing ratio, along with the mixing ratio at two stations had been determined. Figure 6.7 shows the absolute global mixing ratio of the various simulations in the top plot. The difference to the 37 model level version of TM5 can be seen in the bottom plot. From the absolute mixing ratio it can be seen how most variables show good correlation to the new model level simulation

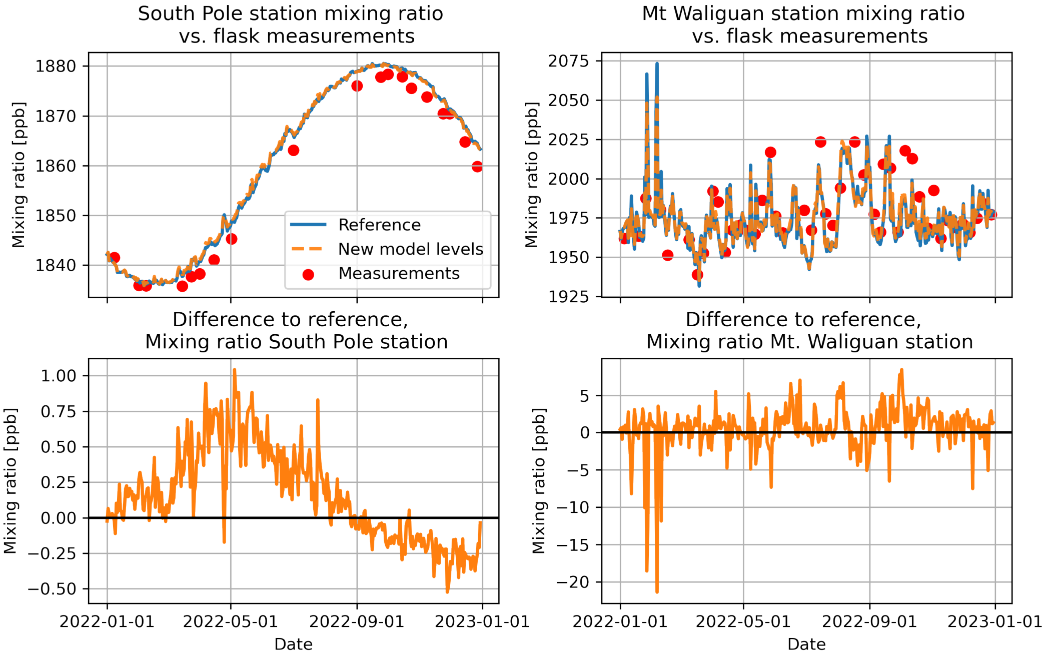


Figure 6.6: A comparison of the reference run to the new model level simulation. Top left shows the absolute mixing ratio at the south pole, with the difference to the reference run on the bottom left. The right plots show the absolute and difference to reference on Mt. Waliguan.

of TM5, apart from the vertical massflux. When looking at the difference plot on the bottom the difference remains close to 0 for the entire duration of the simulation for all variables, except vertical massflux, which rises to a difference of 1.5 ppb by the end of the simulation. This difference is consistent with expectations, as a large error was observed in the massflux calculations in Section 5.3.4. Another thing to note is that the simulation does not continue after the 14th of November, this is due to a negative airmass being calculated in the simulation, causing TM5 to terminate. This is another indication of how the newly defined vertical massflux is problematic.

The mixing ratio at the two stations can be seen in Figure 6.8. It can once again be seen how the temperature, humidity and horizontal massflux simulations of TM5 all follow the 37 model level simulation closely. Small deviations can be seen in the difference plots, although for most of the simulation these stay within ± 1 ppb on the south pole and ± 5 ppb on Mt. Waliguan. When comparing these deviations to the measurement data it can be seen that these differences are small in comparison, which is why these are deemed acceptable.

The simulation including the new vertical massflux on the other hand is once again troublesome. The vertical massflux simulation overestimates the mixing ratio on the South Pole by over 5 ppb for the first 4 months, after which it underestimates the mixing ratio by roughly 2.5 ppb from May on. On Mt. Waliguan it can be seen that a completely different pattern exists, with little correlation to the reference simulation. The vertical massflux therefore significantly alters the results, and causes TM5 to terminate prematurely. For this reasons the conclusion was drawn that the vertical massflux should not be used in TM5 in its current form.

With all variables now simulated in TM5, a selection of the new variables has been made. From the results it can be seen that the temperature, humidity and horizontal massflux all perform well and can be used in TM5. The new reference simulation will therefore consist of 37 model levels, taking the temperature, humidity and horizontal massfluxes from the new pipeline.

The new combination of variables has run in TM5, after which the global mixing ratio has once again been calculated and compared to the reference and 37 model level run, as seen in Figure 6.9. In the figure a slight offset in the mixing ratio can be seen, ending at an offset of roughly 0.17 ppb after one year, which was to be

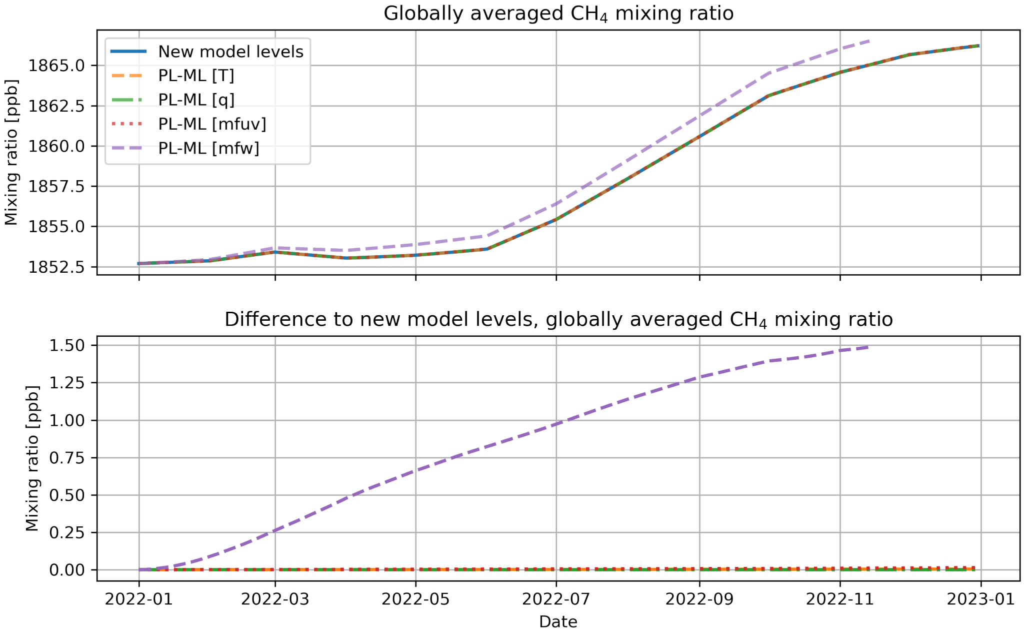


Figure 6.7: Global mean atmospheric mixing ratio of four simulations with different variables from the CDS, the bottom figure shows the difference to the 37 model level version of TM5.

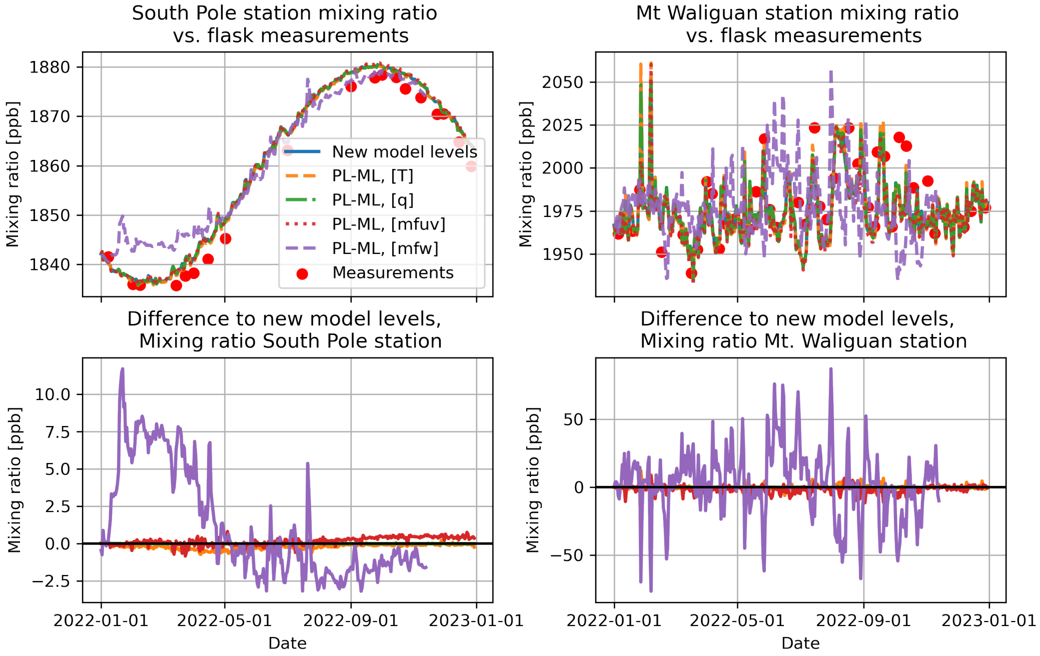


Figure 6.8: A comparison of the simulations using different variables to the new model level simulation. Top left shows the absolute mixing ratio at the south pole, with the difference to the new model levels can be seen on the bottom left. The right plots show the absolute and difference to reference on Mt. Waliguan.

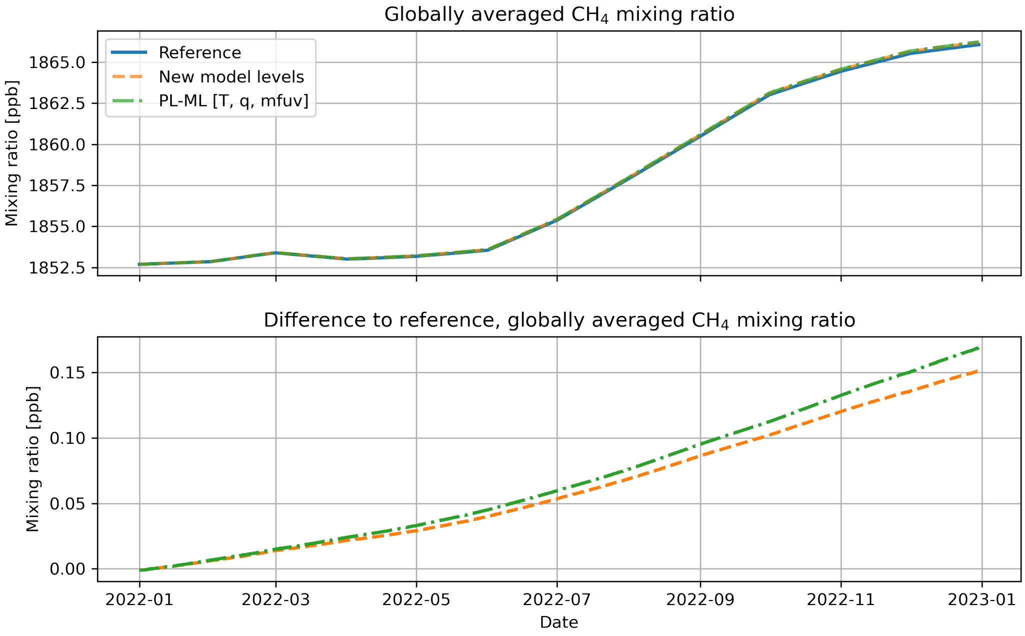


Figure 6.9: A Comparison of the combination of the new variables and the new model levels to the reference simulation on 34 model levels. The top plot shows the absolute globally averaged mixing ration, with the difference to the reference simulation shown on the bottom figure.

expected based upon the verification. The largest offset of 0.150 ppb is caused by the switch to the 37 model layer version, with the variables from the new pipeline only increasing the difference of the mixing ratio by another 0.02 ppb. The 0.02 ppb difference is thought to be caused by the errors introduced in the processing steps to convert GraphCast's output format to TM5's input format.

In Figure 6.10 the new combination of variables is compared to the 37 model level version and the reference simulation at the two selected measurement stations. From the time series in the top plots in the figure it can be seen how the new simulation follows the previous TM5 simulations very well. When looking at the difference figures it can once again be seen how the difference between the various options are small. On the south pole the difference remains within ± 1 ppb. From the difference plot it does appear like a seasonal dependency exists, the reason for this is unknown, however since the magnitude is roughly a factor 50 smaller than the seasonal dependency on absolute mixing ratio, it was considered to be negligible. On Mt. Waliguan the difference remains for the most part within ± 10 ppb. These differences are considered to be small with regards to the offset to the measurements and are therefore considered good enough for proceeding.

The new simulation using 37 model levels and three processed variables looks good according to the tests shown in this section. This new version will therefore be used as the new reference for GraphCast's forecast simulations. For these simulations, this reference simulation will be referred to as ERA5 from now on.

6.5. Impact of GraphCast forecasts

In earlier sections, the various steps of the pipeline have been validated. The final step is to include GraphCast forecasts into the pipeline, as can be seen in pink in Figure 6.1. The impact of the forecast on TM5 can then be determined. The new simulations will be compared against the ERA5 reference simulation that was defined in the previous section. A forecast can be started using two sets of meteorological data of time steps 6 hours apart. A TM5 simulation requires 8 data sets per day that together form a 3 hourly timeseries, whereas GraphCast can only generate its forecast in 6 hour timesteps. This discrepancy can be solved by preparing two datasets to start at $t = 0$ and 3 hours and use them alternately. A risk of this method is that it could result in a discrepancy between the two datasets, particularly for longer forecasting times.

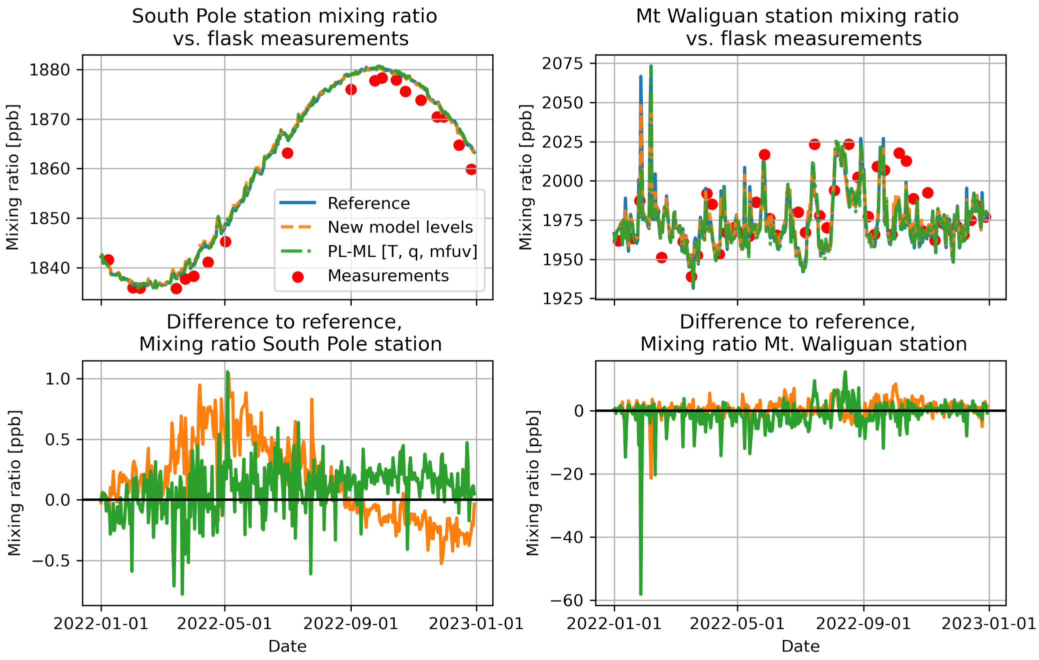


Figure 6.10: A comparison of the combined variable run to the new model level and reference simulations. Top left shows the absolute mixing ratio at the south pole, with the difference to the reference run on the bottom left. The right plots show the absolute and difference to reference on Mt. Waliguan.

A time series of meteorological data sets in time steps of 3 hours has been obtained from the CDS archive, and by selecting pairs that are 6 hours apart from each other, a series of 72 hours forecasts has been made. The result is that for every timestep, 13 different sets of meteorological data are available. A graphical representation of how the various datasets have been composed based upon the forecasting length can be seen in Figure 6.11. Six datasets arranged top to bottom can be seen, each containing 2 input timesteps, followed by 12 forecasted timesteps, separated by 6 hour intervals. Two options with different maximum forecasting lengths are shown with the red and blue colors. A 12 hour forecast uses the first two forecasted timesteps of a dataset pair, after which it skips two datasets. The 36 hour forecast uses the first six timesteps, after which it skips six datasets. In total seven forecasting lengths have been tested, being a 6, 12, 24, 36, 48, 60 and 72 hour forecast.

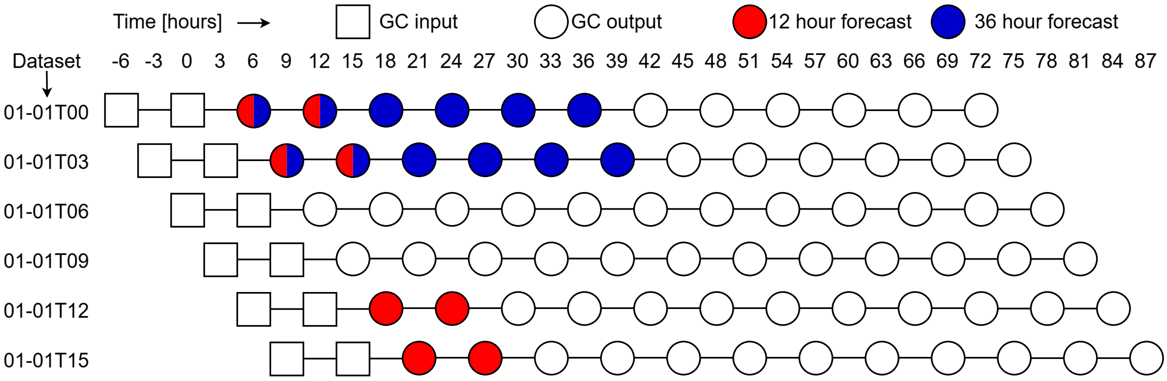


Figure 6.11: Illustration of 6 GraphCast datasets for various timesteps. Time records for a 12 hour forecast can be seen in red, with time records for a 36 hour forecast in blue. Red/blue time records are used by both 12 and 36 hour forecasts.

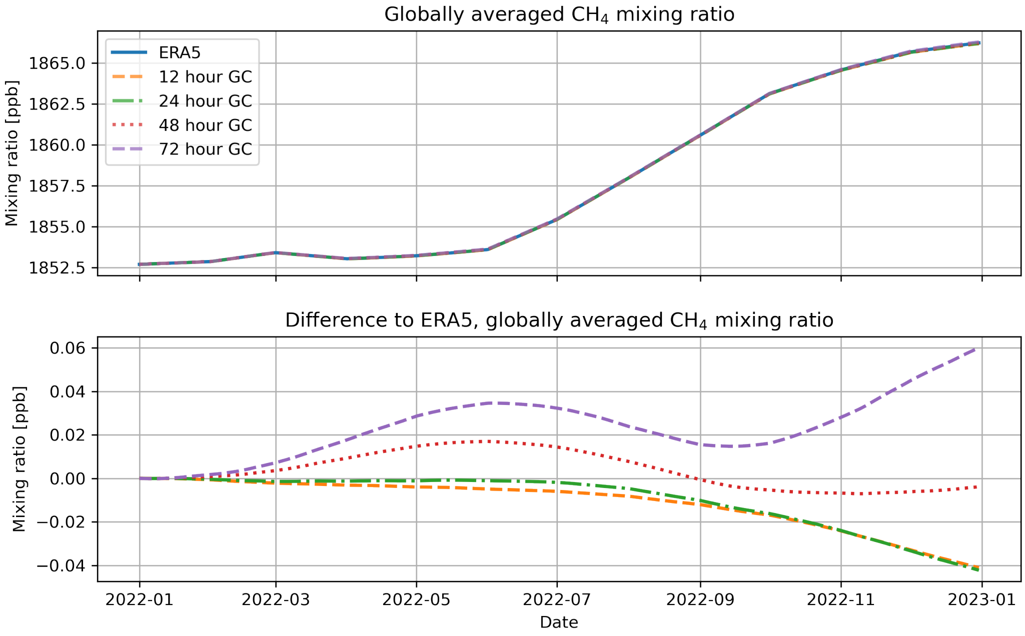


Figure 6.12: Comparison various forecasting lengths in TM5, global absolute mixing ratio top, delta global mixing ratio below.

6.5.1. GLOBAL MIXING RATIO

TM5 simulations based on each of the 7 forecast selections have been compared to the reference defined in Section 6.4. Four forecasting times can be seen in Figure 6.12. The global mixing ratio's appear very similar throughout the year. However, it can be seen how the different datasets start diverging slightly from each other at the end of the year. When looking at the difference plots of the mixing ratio, it can be seen how the various forecasting times compare to the ERA5 reference simulation. The trends look different from what was seen in earlier comparisons. In the verification process, often a small but constant drift could be seen, while in the forecasting runs a more irregular pattern can be seen. For the first 6 months, the differences in the various datasets slowly drift apart from each other, with a longer forecasting length resulting in a larger drift. After the first six months, the differences converge over the subsequent three months, before diverging again during the final three months as the mixing ratios begin to drift apart. The reason for this irregular drift is thought to be due to the forecasting error being variable with the seasons. The differences in the mixing ratio's remain fairly small throughout the year, with the difference of the 72 hour forecast ending at a mixing ratio of roughly 0.045 ppb. When comparing this offset to the previously seen offsets in the verification process, together with the offset to the measurement data, even a 72 hour forecasting dataset can still be considered to be performing well.

6.5.2. STATION MIXING RATIO

Figure 6.13 zooms in on the two selected stations. For this comparison, only a 24 and 72 hour forecast have been compared to ERA5 to provide a cleaner figure. For the forecasting times not shown, similar trends were seen as were found in the 24 and 72 hour forecast runs. On first sight, the results look good, both forecasting times still follow the same trend as the ERA5 simulation on both stations. When looking to the south pole station, a slight drift can be seen near the end of the year compared to the reference. This drift appears to become larger for longer forecasting times, with the 72 hour forecast being roughly 2 ppb below the reference on the 31th of December, with lower forecasting times giving a smaller offset. This larger offset is to be expected, with the longer forecasting times moving further away from the original reference dataset. The underlying cause for the drift is believed to be due to the reduction in forecasting quality of GraphCast in the stratosphere. As was previously discussed, mixing ratio's at the South Pole are partly governed by stratospheric weather phenomena, which are likely to have been affected by the poor forecast quality of GraphCast.

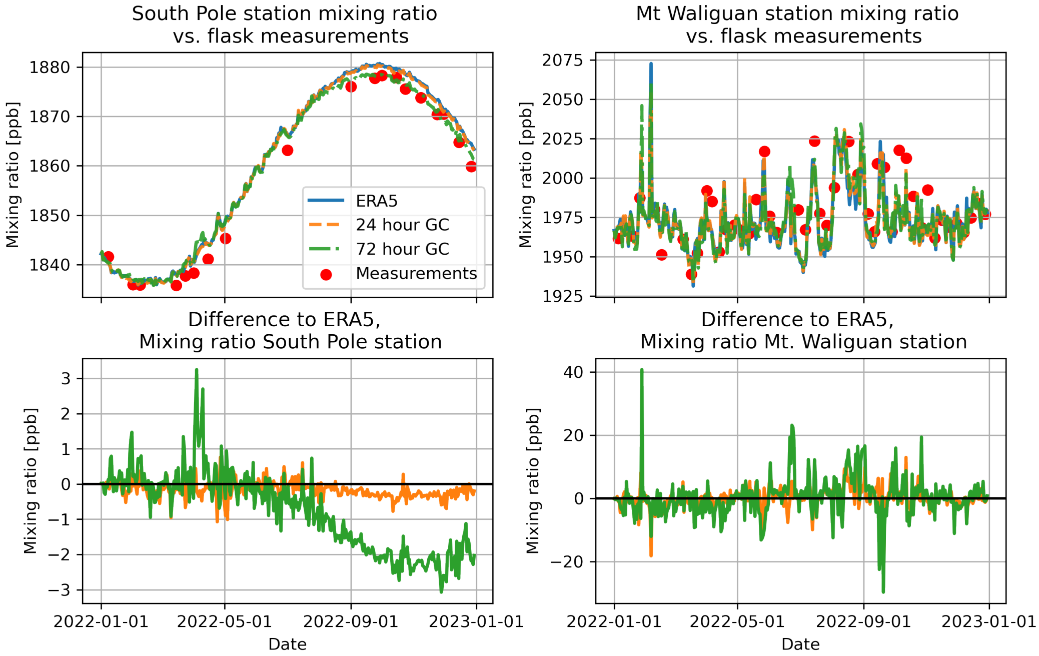


Figure 6.13: A comparison of a 24 and 72 hour forecasting time to ERA5 data. Top left shows the absolute mixing ratio at the south pole, with the difference to the reference run on the bottom left. The right plots show the absolute and difference to reference on Mt. Waliguan.

One interesting point is that for a longer forecasting time, the difference to the measurements on the south pole is reduced. In general this improvement in correlation would be considered to be a positive development, though in this case the reasons for this improvement is not likely to be generic, since the error in the meteorological data is known to increase. For this reason the improved correlation on the South Pole is considered to be a matter of lower quality in meteorological data coincidentally decreasing the offset between measurements and model output, instead of a genuine improvement to the model.

When looking at Mt. Waliguan, both simulations based on 24 and 72 hour forecasts follow the simulation using ERA5 reasonably well. When looking more closely, large differences between the various simulations can be seen on some occasions, particularly sudden peaks in the dataset that are not as well represented. Examples are the peaks at the end of January and September, with these peaks being the most extreme for the 72 hour forecasting time. Regardless of these peaks, both simulations remain within a ±10 ppb difference compared to a simulation based on ERA5 data for the majority of the simulation, these differences are still smaller than what differences that were seen in earlier verification steps, as well as smaller than the offset compared to the measurement data.

Interestingly no significant drift can be seen between the two datasets on Mt. Waliguan. The reason for this is thought to be due to external factors being the main driver for the variability in the dataset. The south pole is located far away from external sources and is therefore mainly driven by meteorological phenomena, which is why a change in meteorological data can be seen more clearly here. Mt. Waliguan on the other hand is located in China and is therefore more affected by its surroundings. A change in meteorological data will therefore be less apparent on Mt. Waliguan than on the south pole.

In order to get a more global view of the effect of forecast length on the simulation of methane mixing ratios at the surface, a comparison can be made between all flask measurements to the various TM5 simulations. The RMSE over all flask measurements for all stations has been compared to their respective datapoints for the 34 model level simulation, as well as ERA5 and all GraphCast simulations, which can be seen in Figure 6.14. The figure contains two datapoints for the 34 and 37 model level simulations using ERA5 data at a forecasting time of 0 hours, followed by seven datapoints for the forecast simulations in TM5. The RMSE of all

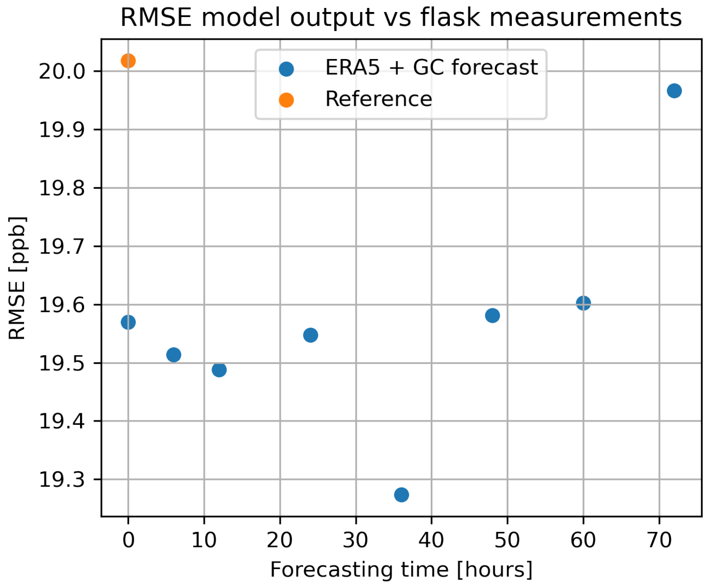


Figure 6.14: RMSE of flask measurements over all measurement stations in 2022 for various TM5 simulations.

datapoints ranges from 19.4 to 20.1 ppb. Most datapoints from TM5 are close to 19.55 ppb, apart from the 36 and 72 hour forecast, which are off by roughly 0.3 ppb. Another point of notice is that the 34 model level reference simulation has a slightly higher RMSE of 20.05, which could be due to the finer vertical resolution at low altitudes that the new 37 model level version has over the 34 model level version of TM5. Overall though, all results are within 5% of each other which is still considered small compared to the difference in RMSE to the flask measurements. For this reason it appears that the forecast length has little effect on the quality of TM5 close to the surface.

6.5.3. Evaluation of vertical profiles

In Section 5.2 it was investigated how GraphCast meteorological simulations evolve with forecast time for different pressure levels. In order to determine how this error manifests itself in TM5, the RMSE of the mixing ratio has been determined on model levels representative of the earlier used pressure levels in Figure 5.5, using the ERA5 simulation as a reference. The RMSE was then normalised by dividing the RMSE by the mean of the mixing ratio of its respective model level.

Figure 6.15 shows the RMSE of the CH_4 mixing ratio for model levels 37, 31 and 22, which are the levels closest to 1000, 850 and 500 hPa respectively. For the calculation, a simulation based on ERA5 data was used as the reference, hence an RMSE of 0 at a forecasting time of 0 hours. From the figure it can be seen how the RMSE starts off at 0 ppb, then quickly increases to 0.3 % for model level 37 and ± 0.1 % for model levels 31 and 22, for the shortest forecasting time of 6 hours. For longer forecasting times, the RMSE at all three model levels rises linearly until the longest forecasting time is reached at 72 hours. The pattern seen in Figure 6.15 is very similar to the pattern seen in Figure 5.5, where the RMSE quickly increased for the first forecasts, after which it would linearly increase with a significantly reduced rate.

The RMSE at model level 37 is significantly higher than the RMSE at model levels 31 and 22. The reason of this is thought to be due to the limited mixing at model level 37, which can also be seen in Figure 3.3. On the left plot small concentrated regions are visible where the mixing ratio of CH_4 is significantly higher than its surroundings. For these regions, a small shift to another location, caused by different meteorological data, will result in a relatively high RMSE, despite the results being very similar. Methane is generally better mixed at higher altitudes, as can also be seen in Figure 3.3, resulting in an overall lower RMSE in higher altitudes.

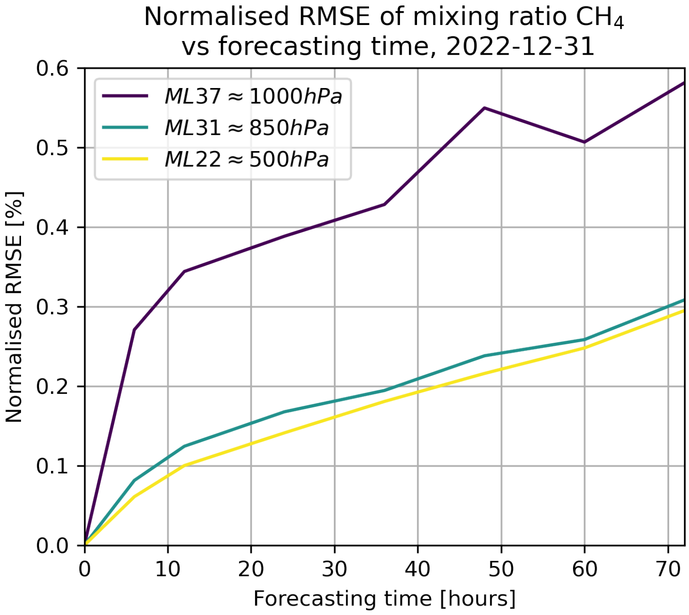


Figure 6.15: Normalised RMSE of methane mixing ratio vs. forecast lengths for model levels representative of 500, 850 and 1000 hPa.

6.5.4. Zonal mean

Another point that was highlighted in Section 5.2 was the rapid reduction of forecasting quality high up in the stratosphere, as was illustrated in Figure 5.4. To analyze this, the longitudinal mean of the mixing ratios was calculated, in a similar fashion as shown in Figure 5.4, and the results are shown in Figure 6.16. The figure shows the absolute mixing ratio of the reference simulation using ERA5 on the top left, followed by the difference between three forecast lengths. The differences are shown for a 24, 48 and 72 hour forecast, which can be seen top right, bottom left and bottom right respectively. The first thing that becomes apparent from the difference figures is the large errors that can be seen in altitudes above 100 hPa. These differences are over 150 ppb on several occasions, which is significantly higher than the error of less than 5 ppb that is seen at the surface. This error pattern is very similar to the pattern that was seen in Figure 5.4, which is why GraphCast's error is thought to be the main contributor to this error. The low RMSE at lower altitudes explains the strong correlation that was seen earlier in the comparison to flask measurements.

This error in higher altitudes could be a potential reason for the drift seen on the South Pole measuring station in Figure 6.13. The stratospheric Brewer-Dobson circulation is an important process for hemispheric transport [33]. Methane in the northern hemisphere rises up to the stratosphere, after which it will travel towards the southern hemisphere. Since the upper troposphere is poorly modelled by GraphCast, this error will therefore likely translate into the error seen on the South Pole in Figure 6.13. Apart from the Brewer-Dobson circulation, changes in temperature affect rate of chemical destruction of CH_4 [34]. Changes in temperature will therefore affect the methane mixing ratio, which is partly transported to the surface as well. These effects are further emphasized by the absence of surface level sources at the South Pole, making transport from the stratosphere relatively important.

The second observation to be made is how the error grows with longer forecasting times. For a 12 hour forecasting time the difference stays below 150 ppb. For the 72 hour forecast, the difference frequently exceeds 150 ppb at altitudes higher than 100 hPa. Similarly in the remaining model levels it can be seen how the difference to a simulation using ERA5 slightly increases with a longer forecasting time. This increase is to be expected, as a longer forecasting time means a larger error compared to ERA5 data.

6.5.5. Comparison with satellite observations

A final comparison to measurement data has been made using satellite observations from the TROPOMI instrument. TROPOMI is an instrument on board of a polar-orbiting satellite that plays a key role in gathering data on atmospheric trace gases. TROPOMI measures backscatter of solar radiation, from which mixing ratios

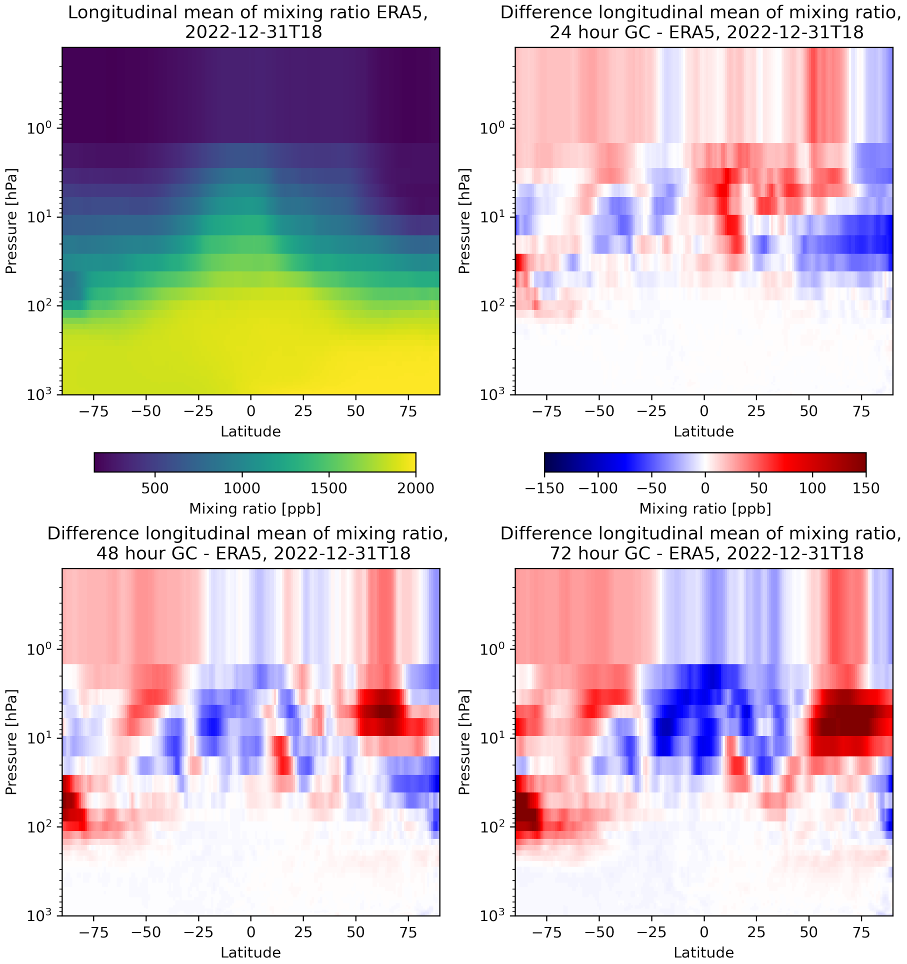


Figure 6.16: The top left plot shows the zonal mean of the absolute CH_4 mixing ratio for the ML37 reference simulation. The top right, bottom left and bottom right show the difference to the ML37 reference simulation for a 24, 48 and 72 hour forecast time respectively. The difference figures all share the same colorbar.

of various atmospheric trace gases could be retrieved [35]. The methane product retrieved from TROPOMI provides a column mixing ratio in ppb, which represents the average mixing ratio over the entire atmosphere at a given location. The same quantity can be simulated from TM5. The available retrievals and simulations have been averaged over the year on a 1° by 1° grid. Figure 6.17 shows the absolute column mixing ratio in the top left plot.

The first thing to note is that there is no full coverage over earth's surface. The reason for this lack of coverage is due to the satellite data being dependent on a good reflection of sunlight in particular wavelengths. Water for example absorbs this wavelength, making it difficult to gather data over large bodies of water; only in the case of sun-glint on the water towards the satellite, the amount of reflected light is large enough to make a retrieval of the CH_4 column. The absolute mixing ratio in the measured locations varies from 1850 to 2075 ppb. The largest mixing ratio can be seen at a latitude of roughly 20 degrees, while a significantly lower mixing ratio can be seen on the south pole, illustrating the north/south gradient of methane. This observation is in accordance with earlier observations, such as the comparison seen in Figure 3.3.

When looking to the two plots on the top right and bottom left of Figure 6.17 a direct comparison has been made between TROPOMI and simulations based on ERA5 and based on a 72 hour GraphCast forecast to bring the difference between the two versions of TM5 into context to the difference to measurements. The two plots look very similar to each other, though differences are visible. It can be seen how near the equator above water, TM5 has a tendency to underestimate the CH_4 mixing ratio compared to the TROPOMI measurements. On land TM5 appears to tend more towards a higher mixing ratio than TROPOMI, with large differences visi-

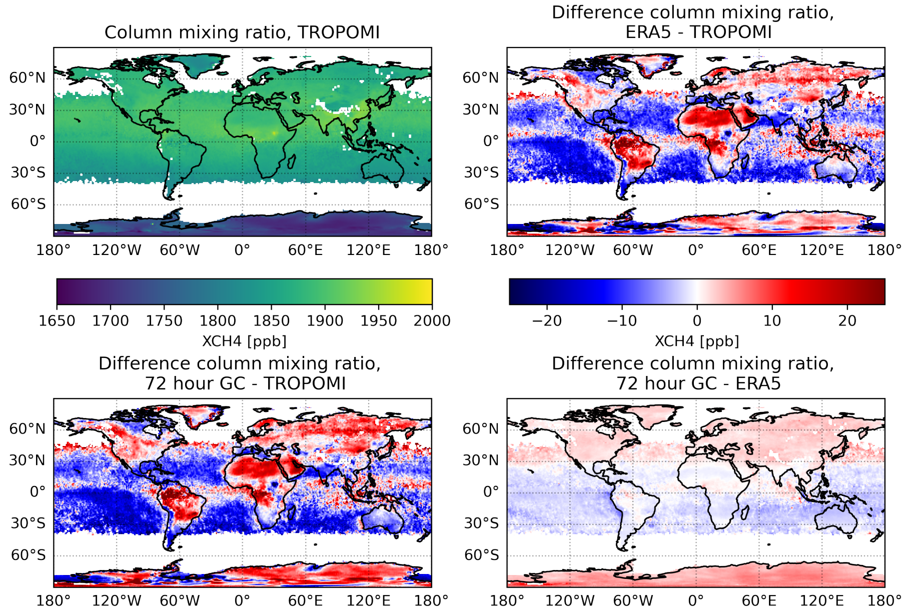

Column mixing ratio, TROPOMI
Difference column mixing ratio, ERA5 - TROPOMI
60°N
30°N
0°
30°S
60°S
180°
120°W
60°W
0°
60°E
120°E
180°
1650
1700
1750
1800
1850
1900
1950
2000
XCH4 [ppb]
−20
−10
0
10
20
XCH4 [ppb]
Difference column mixing ratio, 72 hour GC - TROPOMI
Difference column mixing ratio, 72 hour GC - ERA5

Figure 6.17: The top left plot shows the absolute column mixing ratio as determined by TROPOMI over a 1 year period. The top right figure shows the difference in column mixing ratio between TROPOMI and the ML37 reference simulation. Bottom left figure shows the difference between TROPOMI and the 72 hour forecast simulation. In the bottom right figure, the difference in column mixing ratio between the ML37 reference and 72 hour forecast simulations can be seen.

ble in the northern Africa, as well as near the Amazon. TROPOMI retrievals however are known to be biased low over bright surfaces such as deserts, partly explaining the large differences as seen in northern Africa [36].

When looking at the bottom right plot, it can be seen how the forecasting data has lowered the mixing ratio near the equator, with a difference of roughly 6 ppb being visible between a latitude of -30 to 30 degrees. At northern and southern latitudes an increase in mixing ratio can be seen, with the south pole showing an increase of nearly 10 ppb over most of the south pole. A comparison can be made to Figure 6.16, where it can be seen that on the North and South Poles, at altitudes higher than 100 hPa, a large increase in mixing ratio can be found, whereas a significantly lower mixing ratio can be seen near the equator for those pressures. A similar pattern can be seen in the lower right panel that shows the difference between the differences, a significantly higher mixing ratio can be seen on the north and south pole, where a reduction in mixing ratio can be seen near the equator again. Overall the magnitude of the error between simulations based on 72 hour GraphCast forecasts and based on ERA5 is roughly 30% of the error between simulations based on ERA5 and the TROPOMI measurements, which is still considered acceptable.

6.5.6. Evaluation of simulation times

In this section the error of TM5's output due to the use of GraphCast forecasts will be compared to the cost saved through the use of GraphCast generated data. The quality of TM5's output will be derived from the methane mass, which has been calculated by multiplying the methane mixing ratio with the air mass, according to the following equation:

$$M_{\mathrm{CH}_4} = \chi_{\mathrm{CH}_4} \cdot A \cdot dP/g \tag{6.3}$$

The methane mass has been chosen as a metric since it encompasses the mixing ratio for the entire atmosphere for the calculation of the RMSE, while also taking into account how air mass changes with altitude. The RMSE will subsequently be normalised by dividing the RMSE by the average methane mass per cell. The methane mass has been calculated for the final timestep in the simulations, being the 30th of December, 2022, and can be seen in Figure 6.18. The x-axis shows the data fraction, this data fraction represents the ratio between the amount of data that has been downloaded for a simulation using forecast data and the amount of data that would have been downloaded if no forecast was done. The data fraction is used rather than a time period, as the time needed to generate a forecast depends on the hardware specifications. Using the data fraction provides a more general and comparable metric. Another advantage is that it highlights the re-

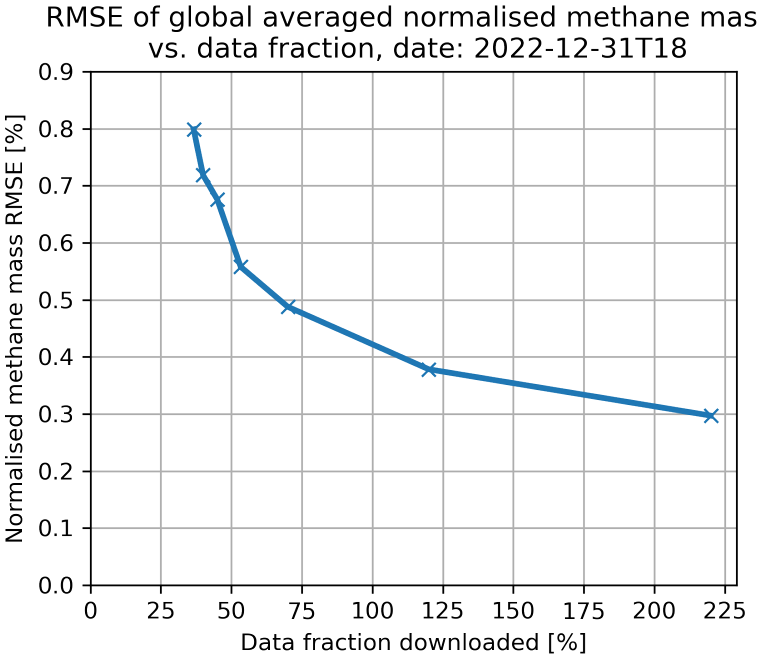


Figure 6.18: Methane atmospheric mass RMSE vs. data fraction downloaded

duction in required local storage space.

In order to calculate the data fraction, the amount of data that had to be downloaded for the run is divided by the amount of data that would otherwise have been downloaded. For example, for a 24 hour forecast, two timesteps have to be downloaded, after which four timesteps will be generated. Each forecasted timestep still requires vertical massflux data to be downloaded, meaning that forecast timesteps still requires 20% of the data to be downloaded per generated timestep. As such, for a 24 hour forecast, $2 \cdot 100\% + 4 \cdot 20\% = 280\%$ of data should be downloaded. Since four timesteps will be generated, this 280% will be divided by $4 \cdot 100\%$, meaning that for a 24 hour forecast, a data fraction of $\frac{2.8}{4}$ will be achieved. The data required for surface level variables has been neglected in this comparison, since the size of the surface level datasets is significantly smaller than the model level datasets. Including this data in the comparison would slightly increase the data fraction.

One important point of notice is that the data fraction has a maximum value of 220%, where a maximum of 100% would be expected. This peculiarity is due to the manner in which forecast data has been used in TM5. As can be seen in Figure 6.11, the dataset used for TM5 contains only forecast data, whereas the two preceding timesteps of data are required to generate a forecast have not been used. This means that for a 6 hour forecast, two timesteps should be downloaded, and another 20% is required to include the vertical massflux. For follow up research, it would be advised to include the downloaded data in the dataset used for TM5 to ensure that the data fraction will be below 100% for all forecast lengths. Another benefit of this combination is that the quality of the dataset will improve, which will result in a lower overall RMSE, particularly for simulations with a short forecasting time.

The methane mass has been calculated for all forecasting times that have been tested. The RMSE was subsequently calculated for all forecasting times, comparing to the ERA5 simulation. The results have been plotted in Figure 6.18. In the figure the relation between the RMSE of the global average CH_4 mass and the relative amount of data to be downloaded can be seen. An inversely proportional relation can be seen. The RMSE rapidly increases when approaching the asymptote at 20%. This asymptote is caused by the vertical massflux still being acquired through the original pipeline, resulting in a maximum of 80% of atmospheric data that can be substituted by GraphCast. The figure also shows how the RMSE slowly decreases for a larger data fraction, with an RMSE of 0.8% at a fraction of 35%, reducing to an RMSE of 0.3% found at a fraction of 220%.

7

Conclusions

In this study the possibility of generating meteorological data using AI for atmospheric chemistry models has been investigated. This consisted of choosing a suitable meteorological AI model, integrating this model in a pipeline for the TM5 model, after which the new pipeline was verified and its sensitivity to the forecast length investigated.

In this study the current state of AI, as well as meteorologic machine learning models were investigated in order to answer the first research sub question, being: **Can meteorological data suitable for TM5 be generated using AI?** From the literature research and application of GraphCast it can be deduced that it is indeed possible to generate meteorological data using AI. Current state of the art AI models capable of accurately generating a weather forecast are abundant and nowadays even capable of outperforming numerical forecasting models in some metrics, while being orders of magnitude faster.

Graphcast has been chosen as the most suitable AI model for this study based upon a variety of metrics, such as feasibility and output format. GraphCast provides a good vertical resolution, documentation and sufficient output variables, along with being an accurate model built upon a GNN. A pipeline has been defined which converts GraphCast's format to a format suitable for TM5. Verification steps showed that the new pipeline works good for all variables, apart from the vertical massflux. The largest error is introduced by the change in vertical resolution for the new model levels, resulting in an annual drift of 0.15 ppb in the globally averaged methane mixing ratio.

Meteorological data is used to drive the atmospheric transport model TM5. During this study it turned out that including the vertical massflux derived from the new pipeline resulted in instability problems, a large difference in the global mixing ratio and poor correlation to flask measurements. These problems were expected as the conversion from vertical velocity to vertical massflux has shown to be problematic. It did however result in the vertical massflux being excluded from the new pipeline. As a result, the new pipeline included only the temperature, humidity and horizontal massflux.

The second research sub question was: **How much accuracy is lost in TM5 through the use of machine learning for generating meteorological data?** Seven different forecasting lengths have been tested and the results investigated. The overall performance of the new model configuration was good. Increased forecasting times resulting in larger errors, though for most metrics the errors remained small. The globally averaged mixed ratio had good results, with a 72 hour forecasting period resulting in a maximum difference of 0.06 ppb compared to an ERA5 reference simulation.

In a comparison with surface flask measurements, a small drift could be seen on the south pole, once again with the larger forecasting times leading to the largest drift. On Mt. Waliguan in China, little difference in error could be seen between the various forecasting times, likely due to interference from surrounding methane sources outweighing the effect of meteorological data. Interestingly when the RMSE was calculated between all flask measurements and TM5's output, no significant increase in RMSE could be seen. The fact that little change could be seen in comparison to measurement data indicates that the influence of the generated forecast data at low altitudes is small compared to the error margin of TM5.

When looking further into the data, peculiarities can be found. It was shown how GraphCast performs poorly in high altitudes, which carries over in TM5. A difference of over 150 ppb between the ERA5 reference simulations and the longest forecasting length simulations can be seen in the stratosphere. The largest loss of accuracy is for altitudes above 100 hPa, while the accuracy for altitudes below 100 hPa remained good. The reason for this is thought to be due to the poor capability of GraphCast to forecast meteorological variables at high altitudes. Another example of GraphCast's errors carrying over in TM5 was seen from the RMSE in mixing ratio for various model levels. The RMSE was calcualted and plotted against forecasting times, where it could be seen how the RMSE increased for longer forecasting times, similar to how the RMSE of generated meteorological variables increased for longer forecasting times.

The final research sub question was: **How much time can be saved in TM5 through the use of machine learning for generating meteorological data?** To answer this question a comparison was made between the RMSE of the mass corrected global mixing ratio and the reduction on data volume required for the TM5 simulations. A strong inversely proportional relation could be seen between the forecasting time and the data fraction. For longer forecasting times, the RMSE increased quickly, whereas the RMSE reduced for shorter forecasting times. Some peculiarities were seen in the RMSE and data fraction, which were relatively high. These peculiarities were caused by not having included ERA5 data in the forecast dataset. Regardless of these anomalies, it can still clearly be seen how the accuracy of TM5 decreases for longer forecasting times.

With the three sub questions answered, the main research question can be answered, which was: **Is it possible to use machine learning generated meteorological data to replace data downloaded from meteorological data archives for running chemical transport models, while maintaining the transport model accuracy on methane mixing ratios?** Machine learning generated meteorological data can be used to replace ERA5 data for atmospheric chemical transport models. From the results it can be seen how the output of TM5 remains largely of good quality, despite slowly reducing for longer forecasting times. For longer forecasting times it was shown how particularly the error in the stratosphere increases rapidly. The error in TM5 has been compared to the reduction in data that needs to be downloaded for TM5. Overall it can be concluded that machine learning generated meteorological data can be used to replace ERA5 data while maintaining efficiency for short forecasting times, however improvements particularly on the vertical massflux and forecasting quality in the stratosphere can lead to significant improvements.

8

OUTLOOK

This section will be split into two sections, Section 8.1 focuses on the chosen AI model, where possible changes to the AI model will be discussed that can streamline the process of generating data and preparing it for TM5. Section 8.2 discusses the improvements that can be made to the processes defined in this report that can improve the results.

8.1. AI METEOROLOGY

In this study various forecasting models have been compared, after which GraphCast was decided to be the most suitable model for this project. During the project it became apparent that various changes could be made to the forecasting model that can speed up the process and reduce the error caused by the generated data. One common problem became apparent on multiple occasions during this study was the inaccuracy of the vertical velocity. During the validation of the forecast it already became apparent that the error of the vertical velocity as generated by GraphCast was of a similar order of magnitude as the data itself. This error was later amplified by the conversion from vertical velocity to vertical massflux. It is for this reason that the vertical velocity dataset from GraphCast was excluded from the pipeline and still retrieved from the original TM5 pipeline. This problem can be solved by changing to a model that either generates a vertical massflux at high accuracy, alternatively a model can be sought that generates the vorticity and divergence.

Another point that was encountered during this study is GraphCast's lack of forecasting quality in the stratosphere. From the literature research, as well as during the verification steps of GraphCast it became apparent that the quality of Graphcast's forecast drops rapidly in the stratosphere. This error propagates into TM5, which can be seen on the longitudinal mean, and is likely a cause for the drift in mixing ratio seen on the South Pole. This reduced forecasting quality is caused by GraphCast's training, which aims at maintaining forecasting quality at low altitudes. Using a model that places a greater emphasis on the forecasting quality in stratosphere could therefore reduce these errors. One risk with using a model that prioritizes higher altitudes could be that it comes at the cost of performance in the lower altitudes, which would not be ideal.

The third point that resulted in an error in TM5 was the reduced vertical resolution of GraphCast in the stratosphere. The total resolution has increased from 34 to 37 model layers, however the distribution has changed to increase the layer density in the troposphere. The amount of layers in the stratosphere has reduced compared to the 34 model level version of TM5. As a result of this redistribution, stratospheric phenomena are likely not as well captured anymore. As a result a drift in the globally averaged mixing ratio, as well as a seasonal dependency on the South Pole was found. For this reason having a forecasting model with an increased layer density in the stratosphere is likely to improve the results.

In order to prepare GraphCast's output for TM5, several processing steps are required, which induce several errors and increase processing time. GraphCast generates its results on 37 pressure levels on a 0.25 degree horizontal grid, while TM5 is currently often configured and run on 34 model levels and a 1 degree horizontal grid. An AI model that generates its results on 37 model levels would allow for skipping the conversion from pressure levels to model levels. Similarly by using a model that generates its output on a 1 degree horizontal grid would negate the need for a coarsening step. Both steps would reduce the complexity of the preprocessing steps.

Another improvement that can be made on the format of GraphCast's output is the size of the timesteps, currently GraphCast works in 6 hour timesteps, which requires the need for loading in alternating datasets. Having a model that provides output at 3 hourly frequency would simplify the process further.

8.2. DATA PROCESSING

Apart from the forecasting model, problems were also encountered related to the processing steps. The most obvious problem that has been apparent throughout this project is the inaccurate conversion of vertical velocity to vertical massflux. In the previous section it was already discussed how a forecasting model generating vertical massflux could be a viable solution. An alternative option would be to investigate why the conversion was not successful and adapt the calculations. The benefit of revising the calculations would be that it opens up the option to many more forecasting models, while still being able to include all five variables in TM5, though it will still come with the complexity required for the calculations.

Another point that was not investigated in this report, but has the potential to improve the accuracy of TM5 is a study on how to best generate and include ERA5 data in the forecasting dataset. In this report, the data that was required to start GraphCast was not included in the forecasting dataset, while this could have been included to improve the quality of the dataset. For this reason a data fraction higher than 1 was found, while the RMSE of TM5's output was also higher than it could have been if the starting data was included in the dataset. On that same note, different optimization strategies for generating meteorological data have not been investigated. The potential gain from such an optimization strategy would be better meteorological data or longer forecast lengths.

During this study, several forecast lengths have been investigated, with the maximum forecast length being a 72 hour forecast, despite GraphCast being able to do a 10 day forecast. At the start of this project it was assumed that after 72 hours the quality of the meteorological data would have deteriorated to such an extent that it would not make sense to do any longer forecasts. During the simulations in TM5 it became apparent that even for a 72 hour forecast TM5 still performed reasonably well. For this reason it would be advisable to include longer forecasting times for follow up research, to better capture the effects of longer forecasting times.

For TM5 not all input data has been acquired from GraphCast. Surface level variables, as well as fields for convection and diffusion are still downloaded from the MARS archive. As a consequence a relatively large amount of data still needs to be downloaded. GraphCast does generate various surface level variables, and the convective/diffusive fields can be calculated from available 3D data, however this has not been investigated yet. Including these variables in the dataset acquired from GraphCast could allow for a reduction in data that needs to be downloaded.

One final remark is the usage of GPUs for running GraphCast. This has been briefly investigated and it was shown how using GPUs could significantly increase the speed at which meteorological data will be generated. During the duration of this project it has not been possible to run GraphCast on GPUs available for this project. When this would be possible however, it has the potential result a huge decrease in processing time when applied.

Bibliography

[1] R. Pachauri and L. Meyer, *Climate Change 2014: Synthesis Report*, Technical Report 1 (IPCC, 2014).

[2] P. Bergamaschi, A. Segers, D. Brunner, J. Haussaire, S. Henne, M. Ramonet, T. Arnold, T. Biermann, H. Chen, S. Conil, M. Delmotte, G. Forster, A. Frumau, D. kubistin, X. Lan, M. Leuenberger, M. Lindauer, M. Lopez, G. Manca, J. Müller-Williams, S. O'Doherty, B. Scheeren, M. Steinbacher, P. Trisolino, G. Vítková, and C. Kwok, *High-resolution inverse modelling of european ch4 emissions using the novel flexpart-cosmo tm5 4dvar inverse modelling system,* Atmospheric Chemistry Physics **22**, 13242 (2022).

[3] V. Huijnen, J. Williams, M. van Weele, T. van Noije, M. Krol, F. Dentener, A. Segers, S. Houweling, and W. Peters, *The global chemistry transport model tm5: description and evaluation of the tropospheric chemistry version 3.0,* Geoscientific Model Development **3**, 445 (2010).

[4] Z. B. Bouallègue, M. C. A. Clare, E. G. Linus Magnusson, M. Maier-Gerber, M. Janoušek, M. Rodwell, F. Pinault, J. S. Dramsch, S. T. K. Lang, B. Raoult, F. Rabier, M. Chevallier, I. Sandu, P. Dueben, M. Chantry, and F. Pappenberger, *The rise of data-driven weather forecasting,* BAMS Bulletin of the American Meteorological Society **105**, E864 (2024).

[5] K. Bi, L. Xie, H. Zhang, X. C. X. Gu, and Q. Tian, *Accurate medium-range global weather forecasting with 3d neural networks,* Nature **619**, 533 (2023).

[6] H. Hersbach, B. Bell, P. Berrisford, S. Hirahara, A. Horányi, J. Muñoz-Sabater, J. Nicolas, C. Peubey, R. Radu, D. Schepers, A. Simmons, C. Soci, S. Abdalla, X. Abellan, G. Balsamo, P. Bechtold, G. Biavati, J. Bidlot, M. Bonavita, G. De Chiara, P. Dahlgren, D. Dee, M. Diamantakis, R. Dragani, J. Flemming, R. Forbes, M. Fuentes, A. Geer, L. Haimberger, S. Healy, R. Hogan, E. Hólm, M. Janisková, S. Keeley, P. Laloyaux, P. Lopez, C. Lupu, G. Radnoti, P. de Rosnay, I. Rozum, F. Vamborg, S. Villaume, and J.-N. Thépaut, *The era5 global reanalysis,* Quarterly Journal of the Royal Meteorological Society , 1999 (2020).

[7] ECMWF, *Era5: Data documentation,* (2025).

[8] ECMWF, *Mars user documentation,* (2023).

[9] ECMWF, *Ifs documentation cy49r1 - part iii: Dynamics and numerical procedures,* (ECMWF, 2024) Chap. 3.

[10] J. East, D. J. Jacob, N. Balasus, A. A. Bloom, L. Bruhwiler, Z. Chen, J. O. Kaplan, L. J. Mickley, T. A. Mooring, E. Penn, B. Poulter, M. P. Solprizio, J. R. Worden, R. M. Yantosca, and Z. Zhang, *Interpreting the seasonality of atmospheric methane,* Geophysical Research Letters **51** (2024), https://doi.org/10.1029/2024GL108494.

[11] E. Dlugokencky, L. P. Steele, P. lang, and K. Masarie, *The growth rate and distribution of atmospheric methane,* Geophysical Research **99**, 17021 (1994).

[12] R. S. Lee, *Artificial intelligence in daily life*, 1st ed. (Springer, 2020).

[13] Z.-H. Zhou, *Machine Learning*, 2nd ed. (Springer, 2023).

[14] E. Herberg, *Lecture notes: Neural network architectures,* ArXiv (2023), https://doi.org/10.48550/arXiv.2304.05133.

[15] C.-C. Liu, K. Hsu, M. S.Peng, D.-S. Chen, P.-L. Chang, L.-F. Hsiao, C.-T. Fong, J.-S. Hong, C.-P. Cheng, and H.-C. K. Kuo-Chen Lu, Chia-Rong Chen, *Evaluation of five global ai models for predicting weather in eastern asia and western pacific,* npj Climate and Atmosphere Science **7** (2024), https://doi.org/10.1038/s41612-024-00769-0.

[16] B. Bonev, T. Kurth, C. Hundt, J. Pathak, M. Baust, K. Kashinath, and A. Anandkumar, *Spherical fourier neural operators: Learning stable dynamics on the sphere,* ICML'23: International Conference on Machine Learning , 2806 (2023).

[17] R. Lam, A. Sanchez-Gonzalez, M. Willson, P. Wirnsberger, M. Fortunato, F. Alet, S. Ravuri, T. Ewalds, Z. Eaton-Rosen, W. Hu, A. Merose, S. Hoyer, G. Holland, O. Vinyals, J. Stott, A. Pritzel, S. Mohammed, and P. Battaglia, *Graphcast: Learning skillful medium-range global weather forecasting,* Atmospheric and Oceanic Physics (2023), https://doi.org/10.48550/arXiv.2212.12794.

[18] L. Chen, X. Zhong, F. Zhang, Y. Cheng, Y. Xu, Y. Qi, and H. Li, *Fuxi: a cascade machine learning forecasting system for 15-day global weather forecast,* npj climate and atmospheric science **6** (2023), https://doi.org/10.1038/s41612-023-00512-1.

[19] K. Chen, T. Han, J. Gong, L. Bai, F. Ling, J.-J. Luo, X. Chen, L. Ma, T. Zhang, R. Su, Y. Ci, B. Li, X. Yang, and W. Ouyang, *Fengwu: Pushing the skillful global medium-range weather forecast beyond 10 days lead,* ArXiv (2023), https://doi.org/10.48550/arXiv.2304.02948.

[20] S. Rasp, S. Hoyer, A. Merose, I. Langmore, P. Battaglia, T. Russel, A. Sanchez-Gonzalez, V. Yang, R. Carver, S. Agrawal, M. Chantry, Z. B. Bouallegue, P. Dueben, C. Bromberg, J. Sisk, L. Barrington, A. Bell, and F. Sha, *Weatherbench 2: A benchmark for the next generation of data-driven global weather models,* ArXiv (2023), https://doi.org/10.48550/arXiv.2308.15560.

[21] A. J. Charlton-Perez, H. F. Dacre, S. Driscoll, S. L. Gray, B. Harvey, N. J. Harvey, K. M. R. Hunt, R. Lee, R. Swaminathan, R. Vandaele, and A. Volonté, *Do ai models produce better weather forecasts than physics-based models? a quantitative evaluation case study of storm ciarán,* Climate and atmospheric science **7** (2024), https://doi.org/10.1038/s41612-024-00638-w.

[22] P. W. Battaglia, J. B. Hamrick, V. Bapst, A. Sanchez-Gonzalez, V. Zambaldi, M. Malinowski, A. Tacchetti, D. Raposo, A. Santoro, R. Faulkner, C. Gulcehre, F. Song, A. Ballard, J. Gilmer, G. Dahl, A. Vaswani, K. Allen, C. Nash, V. Langston, C. Dyer, N. Heess, D. Wierstra, P. Kohli, M. Botvinick, O. Vinyals, Y. Li, and R. Pascanu, *Relational inductive biases, deep learning, and graph networks,* Arxiv (2018), https://arxiv.org/abs/1806.01261.

[23] R. Keisler, *Forecasting globalweather with graph neural networks,* ArXiv (2022), https://arxiv.org/abs/2202.07575.

[24] T. Pfaff, M. Fortunato, A. Sanchez-Gonzalez, and P. W. Battaglia, *Learning mesh-based simulation with graph networks,* ArXiv (2020), https://arxiv.org/abs/2010.03409.

[25] G. Deepmind, *Google deepmind graphcast and gencast,* (2024).

[26] G. Deepmind, *Google cloud bucket,* (2024).

[27] ECMWF, *Era5: What is the spatial reference,* (2025).

[28] I. T. Jolliffe and D. B. Stephenson, *Forecast Verification : A Practitioner's Guide in Atmospheric Scienc*, 2nd ed. (John Wiley & Sons, Incorporated, 2011).

[29] G. Gualtieri, *Analysing the uncertainties of reanalysis data used for wind resource assessment: A critical review,* Renewable and Sustainable Energy Reviews **167** (2022), https://doi.org/10.1016/j.rser.2022.112741.

[30] ECMWF, *L137 model level definitions,* (2024).

[31] A. Segers, A. Bregman, P. van Velthoven, M. Krol, and E. Meijer, *Mass conservative computation of advective fluxes for tracer transport models,* (2005).

[32] G. monitoring Laboratory, *Trends in atmospheric methane (ch4),* (2025).

[33] D. A. Belikov, N. Saitoh, and P. K. Patra, *An analysis of interhemispheric transport pathways based on three-dimensional methane data by gosat observations and model simulations,* JGR Atmosphere **127** (2022), https://doi.org/10.1029/2021JD035688.

[34] Y. Zhao, M. Saunois, P. Bousquet, X. Lin, A. Berchet, M. Hegglin, G. Canadell, R. Jackson, D. Hauglustaine, S. Szopa, A. Stavert, N. Abraham, A. Archibald, S. Bekki, M. Deushi, P. Jöckel, B. Josse, D. Kinnison, O. Kirner, V. Maréval, F. O'Connor, D. A. Plummer, L. Revell, E. Rozanov, A. Stenke, S. Strode, S. Tilmes, E. Dlugokencky, and B. Zheng, *Inter-model comparison of global hydroxyl radical (oh) distributions and their impact on atmospheric methane over the 2000–2016 period,* Atmospheric Chemistry and Physics **19**, 13701 (2019).

[35] A. Lorente, T. Borsdorff, A. Butz, O. Hasekamp, J. aan de Brugh, A. Schneider, L. Wu, F. Hase, R. Kivi, D. Wunch, D. F. Pollard, K. Shiomi, N. M. Deutscher, V. A. Velazco, C. M. Roehl, P. O. Wennberg, T. Warneke, and J. Landgraf, *Methane retrieved from tropomi: improvement of the data product and validation of the first 2 years of measurements,* Atmospheric Measurement Techniques **14**, 665 (2021).

[36] A. Segers, R. Nanni, S. Houweling, and J. van Peet, *Evaluation and Quality Control document for observation-based CH4 flux estimates for the period 1983 – 2023*, Tech. Rep. D2.4.1-2024R2 (Copernicus, 2024).

A

APPENDIX

A.1. TM5 INPUT PARAMETERS

Meteorological input variables, along with their possible alternatives can be seen in Table A.1. The table shows whether something is a model level or surface level variable, along with the name of the variable. Finally the use in TM5 is can be seen in the final column.

Table A.1: List of all meteorological variables required for TM5, along with a potential alternative that could replace it.

Variable	**Alternative**	**Level**	**Name [units]**	**TM5 usage**
lnsp		SFC	log-surface-pressure [-]	Pressure levels
	sp	SFC	Surface pressure [Pa]	Pressure levels
sr		SFC	Surface roughness [m]	Vertical diffusion
blh		SFC	Boundary layer height [m]	Vertical diffusion
ewss		SFC	East-west surface stress [N/m2/s]	Vertical diffusion
nsss		SFC	North-south surface stress [N/m2/s]	Vertical diffusion
slhf		SFC	Surface latent heat flux [W/m2]	Vertical diffusion
sshf		SFC	Surface sensible heat flux [W/m2]	Vertical diffusion
u10m		SFC	10m wind u component [m/s]	Vertical diffusion
v10m		SFC	10m wind v component [m/s]	Vertical diffusion
vo		ML	Vorticity [1/s]	Air massfluxes in x/y/z
d		ML	Divergence [1/s]	Air massfluxes in x/y/z
	u	ML	u component of wind [m/s]	Air massfluxes in x/y
	v	ML	v component of wind [m/s]	Air massfluxes in x/y
	etadot	ML	Eta coordinate vertical velocity [1/s]	Air massfluxes in z
	w	ML	Vertical velocity [Pa/s]	Air massfluxes in z
t		ML	Temperature [K]	Temperature
q		ML	Specific humidity [$\text{kg}_{water}/\text{kg}_{air}$]	Humidity
	mumf	ML	Mean updraught mass flux [kg/m2/s]	Convection
	mdmf	ML	Mean downdraught mass flux [kg/m2/s]	Convection
	mudr	ML	Mean updraught detrainment rate [kg/m2/s]	Convection
	mddr	ML	mean downdraught detrainment rate [kg/m2/s]	Convection

A.2. GRAPHCAST INPUT VARIABLES

GraphCast requires a variety of inputs, ranging from meteorological data on surface and model levels, to several time measurements. All required inputs variables for GraphCast can be seen in Table A.2.

Table A.2: Surface and atmospheric variables and their respective level type, units and coordinates, required for starting GraphCast

Variable	**Acronym**	**Level type**	**Units**	**Coordinates**
2m temperature	T_{2m}	Surface	K	Time, lat, lon
Temperature	T	Atmospheric	K	Time, level, lat, lon
Geopotential	z	Atmospheric	$m^2\,m^{-2}$	Time, level, lat, lon
Geopotential (Surface)	z	Surface	$m^2\,m^{-2}$	Lat, lon
Mean sea level pressure	msl	Surface	Pa	Time, lat, lon
Specific humidity	q	Atmospheric	$kg\,kg^{-1}$	Time, level, lat, lon
10m u component of wind	u_{10}	Surface	$m\,s^{-1}$	Time, lat, lon
10m v component of wind	v_{10}	Surface	$m\,s^{-1}$	Time, lat, lon
u component of wind	u	Atmospheric	$m\,s^{-1}$	Time, level, lat, lon
v component of wind	v	Atmospheric	$m\,s^{-1}$	Time, level, lat, lon
Vertical velocity	w	Atmospheric	$Pa\,s^{-1}$	Time, level, lat, lon
Land sea mask	lsm	Surface	-	Lat, lon
Total precipitation	tp	Surface	m	Time, lat, lon
TOA incident solar radiation	tisr	Surface	$W\,M^{-2}$	Time, lat, lon
Local time of day (sin)	Time (sin)	Clock	n/a	Time
Local time of day (cos)	Time (cos)	Clock	n/a	Time
Elapsed year progress (sin)	Date (sin)	Clock	n/a	Time
Elapsed year progress (cos)	Date (cos)	Clock	n/a	Time

A.3. GRAPHCAST PRESSURE LEVELS

GraphCast puts out its meteorological data on 37 pressure levels, ranging from 1 to 1000 hPa. These pressure levels are listed in Table A.3.

Table A.3: GraphCast pressure levels as in- and output from GraphCast

Pressure levels (37) [hPa]
1, 2, 3, 5, 7, 10, 20, 30, 50, 70, 100, 125, 150, 175, 200, 225, 250, 300, 350, 400, 450, 500, 550, 600, 650, 700, 750, 775, 800, 825, 850, 875, 900, 925, 950, 975, 1000

A.4. GraphCast Model level definition

New model level *a* and *b* coefficients have been defined, which can be seen in Table A.4.

Table A.4: Defined model half-levels with corresponding coefficients.

Level	A coefficient	B coefficient
0	0	0
1	159.2794	0
2	269.5396	0
3	427.5925	0
4	564.4135	0
5	823.9678	0
6	1571.623	0
7	2465.771	0
8	3911.49	0
9	6156.074	0
10	8608.525	0.000059
11	11116.66	0.00089
12	13324.67	0.003971
13	14975.62	0.009261
14	16527.32	0.018318
15	17901.62	0.032176
16	19031.29	0.051773
17	19859.39	0.077958
18	20412.31	0.124448
19	20361.82	0.16891
20	19874.03	0.222333
21	18917.46	0.285354
22	17471.84	0.358254
23	16262.05	0.411125
24	14898.45	0.466003
25	12668.26	0.549301
26	10370.18	0.630036
27	8880.453	0.680643
28	8163.375	0.704669
29	6804.422	0.749797
30	6168.531	0.770798
31	4993.797	0.809536
32	3955.961	0.843881
33	2659.141	0.887408
34	1659.477	0.922096
35	734.9922	0.95655
36	62.78125	0.9885
37	0	1