Stony Brook University

A Novel Methodology for Stochastic Formulation of Short Term Cloud Cover Forecasts

A Dissertation Presented

by

Ya-Ting Huang

to

The Graduate School

in Partial Fulfillment of the

Requirements

for the Degree of

Doctor of Philosophy

in

Applied Mathematics and Statistics

Stony Brook University

August 2015

Stony Brook University

The Graduate School

Ya-Ting Huang

We, the dissertation committee for the above candidate for the
Doctor of Philosophy degree, hereby recommend
acceptance of this dissertation.

James Glimm - Dissertation Advisor
Professor, Department of Applied Mathematics and Statistics

Roman Samulyak - Chairperson of Defense
Professor, Department of Applied Mathematics and Statistics

Song Wu - Member
Assistant Professor, Department of Applied Mathematics and Statistics

Minghua Zhang - Outside Member
Professor, School of Marine and Atmospheric Sciences

This dissertation is accepted by the Graduate School.

Charles Taber
Dean of the Graduate School

Abstract of the Dissertation

A Novel Methodology for Stochastic Formulation of Short Term Cloud Cover Forecasts

by

Ya-Ting Huang

Doctor of Philosophy

in

Applied Mathematics and Statistics

Stony Brook University

2015

Following the chaos theory proposed by Lorenz, probabilistic approaches have been widely used in numerical weather prediction research. This paper introduces an innate methodology to measure the uncertainty of stochastic cloud boundary forecast. A stochastic partial differential equation is inserted into a numerical weather prediction model, and backtested to validate the probabilistic results of the model. This methodology can be applied to a variety of topics in numerical weather prediction research.

The proposed method is applied to the short term forecast of cloud cover. A two parameter model based on physical principles of wind velocity disper-

sion and surface evaporation rate drives the stochastic model. They are used to couple a stochastic partial differential equation with a standard weather model (WRF) and satellite data to yield a probabilistic prediction of cloud cover. Results show good predictive capability of the model in forecasting cloud boundary for one half hour, with a gradual loss of predictive power over the following hour.

Key Words: numerical weather prediction, probabilistic model, front tracking, Fokker-Planck equation.

To family and friends

Table of Contents

List of Figures

List of Tables

Acknowledgements

First and foremost, I am deeply grateful to my advisor, Professor James Glimm, without whom none of this thesis would have been possible. It has been my honor to work under the guidance of Professor James Glimm. In addition to clarifying the doubts and passing on the knowledge, his action conveys the virtues to his students what a great scientist should have – self-discipline, enthusiasm, and integrity.

It is also my privilege to thank Professor Minghua Zhang for the patience, guidance and time he devoted to me at all stages of the research process. I would also like to thank Professor Roman Samulyak and Professor Song Wu for their continued help and support being my dissertation committee. My work was generously supported over the past years by Army Research Office W911NF0910306.

I have benefitted greatly from my fellow graduate students for their friendship and encouragement . In particular, I would like to mention Ryan Kaufman, HyungKyung Lim, Tulin Kaman, Ying Xu, and Wenlin Hu. I would like to specially thank Jeremy Melvin and Pooja Rao. Their kindness and generosity have helped me to get through those tough moments.

Finally, I wish to thank my family: my beloved husband and my best

friend Chen-Hung Wu, my parents Hua-Thai Huang and Chi-Fang Chang, my brother, Wei-Jay Huang, and my lifelong friend Yi-Hsuan Tsai, for the unconditional support and love. This dissertation would never have come to be without them. My dissertation is dedicated to them.

Chapter 1

Introduction

Numerical weather predictions are inherently statistical, in view of the large number of detailed physical phenomena incompletely modeled in simulations. In spite of this fact, prediction methods are primarily deterministic, with an overlay of statistics in the form of a limited ensemble of predicted scenarios. The purpose of this paper is to introduce a more intrinsically stochastic methodology. Because of the possible wider interest in the methodology developed here, we formulate some of our results in a generality which goes beyond the present context. Our work has the following components:

1. a probabilistic forecasting model with a physical basis,
2. parameter estimation based on observational data,
3. insertion of stochastic equations into a standard weather model (WRF),
4. backtesting to validate the predictions of the probabilities generated.

This program is far too ambitious to carry out in general, and we study a limited context in which the program can be completed. This context, short term prediction of cloud cover, is itself of practical interest in its relation

(among other possible applications) to solar energy generation and the utilization of standby generators, which have a range of start up times. We consider only the question of cloud cover as recorded in satellite images, and do not address the relation of cloud cover to radiation received by a solar panel.

Beyond the prediction applications addressed here, the methodology developed may have value in assessing NWP cloud cover subgrid models. Our calibration of the wind turbulence modeled diffusion constant could be of interest to the study of atmospheric chemistry, which also requires a turbulent related diffusion constant, [12].

Optimizing the photovoltaic system has been a popular topic, with many approaches developed to improve the accuracy of forecast related to solar power generation. The methodology we introduce in this paper improves upon the cloud cover forecasts of prior work in both numerics and statistics in regard to points 1-4 above. As an important factor in power production, the forecast of cloud cover is directly forecasted in our study. We assimilate the observed data (the satellite images) and the numerical weather forecast data to increase the accuracy of the prediction.

[33] use regression analysis to parameterize the related humidity, precipitation and three level cloud cover coming from a Numerical Weather Prediction (NWP) model.The authors observe the grid point value (GPV) of irradiance from five weather stations and they correlate the hourly averaged radiance to the weather variables with a correlation coefficient of 0.9. [9] use support vector machines (SVM) to classify and categorize the forecast power production. These authors use NWP model data, temperature, relative humidity,

and three-level cloud cover level, to provide hourly forecasts of the insolation averaged over a approximately 6x7 km^2 grid cells. The forecasts are compared to the measurement of power production from solar power plants. The annual root mean square error has been reduced by nearly half through their use of SVM. Errors are given as 10% of peak power production. In contrast to their studies, we forecast five minute cloud cover, we predict probabilities of cloud cover on a 1x1 km^2 grid, and we implement the prediction as a stochastic Fokker-Planck equation into NWP.

Probabilistic approaches have been proposed in cloud screening, see [27], [13], where the cloud cover detection was improved by use of a Bayesian scheme, using prior numerical weather information, including sea surface temperature level, as the prior information. The authors detect but do not predict the cloud cover. [36] introduced the Fokker-Planck equation to simulate hydrological models, including the cloud cover field. This paper solves the Fokker-Planck equation to model the transport process to diffuse the cloud boundary concentration rather than the probability diffusion as we consider. Our diffusion of probabilities, rather than the diffusion of the cloud boundaries, appears to be a more satisfactory physics model for cloud boundaries, which are certainly sharp (discontinuous) on a 1 km^2 grid scale.

Ensemble based statistical predictions, known as the Monte Carlo (MC) method, [8], are slow to converge, so that a generous 10% error in the prediction would require an unrealistically large number of 100 scenarios. Moreover, the generation of the ensemble has to represent the trproportional to aue probabilities of unknowns in the simulations. For uncertain initial conditions,

there seems to be no statistically tested method to achieve this goal. For uncertainties in the parameters defining the simulation model, the problem is deeper, in that the uncertainties may not translate into testable probabilities for the observations. In place of such MC methods, we base our work on a partial differential equation (the Fokker-Plank equation) for the evolution of probabilities.

Chapter 2

Numerical Models

2.1 Model Input Definition

2.1.1 The Satellite Images

For observation data, we use satellite images provided by College of DuPage Weather Lab. The images are from the Geostationary Operational Environmental Satellite system (GOES), operated by the United States National Environmental Satellite, Data, and Information Services (NESDIS). The satellite image dataset covers the North America with different resolutions. Here we use the finest grid offered, 1 km resolution, see Fig. 2.1 We analyze images for Long Island, with an update frequency of 15 minutes.

To meet satellite image processing needs, Laboratoire d'Optique Atmospherique (LOA) built an efficient and intuitive image analysis and display system, named Satellite Process Handling Images uNder XWindow. (SPHINX), see [11] [23].

Msphinx has been instrumental in serving as both a daily image process-

ing tool at LOA, and research tool at various weather research communities. In this thesis, we use the Sphinx as a pre-processing tool to detect the cloud boundaries from the raw satellite images.

By the color-filling function in Msphinx, we choose the display level carefully to have the best effect. The display level here is mainly decided by the brightness level, but not limited to this. The algorithm in the color filling function first recognizes the geographical condition of the image, then based on the brightness level, the cloud area is identified. To automate the image process without the graphic interface, we define a macro module of Msphinx to serve as the pre-processing tool.

With this visual analysis tool, we the identify the brightness level of each pixel, which is then matched to a color with pixels which is recognized as cloud by its brightness level. If the level of brightness exceeds a certain level, which is relatively decided by the overall brightness, then the pixel will be filled red (RGB code [254, 0, 0]) and identified as cloud. By contouring the boundary of the color mask, we can generate a cloud boundary level set function for initialization.

2.1.2 Meteorological Data

WRF accepts various types of different gridded data; the input we use here is real-time data set and reanalysis from National Centers for Environmental Prediction (NCEP). NCEP receives data sets from Numerical Weather Prediction (NWP) models in real-time and then updates the gridded data set, which represents the state of the atmosphere, and incorporating observations.

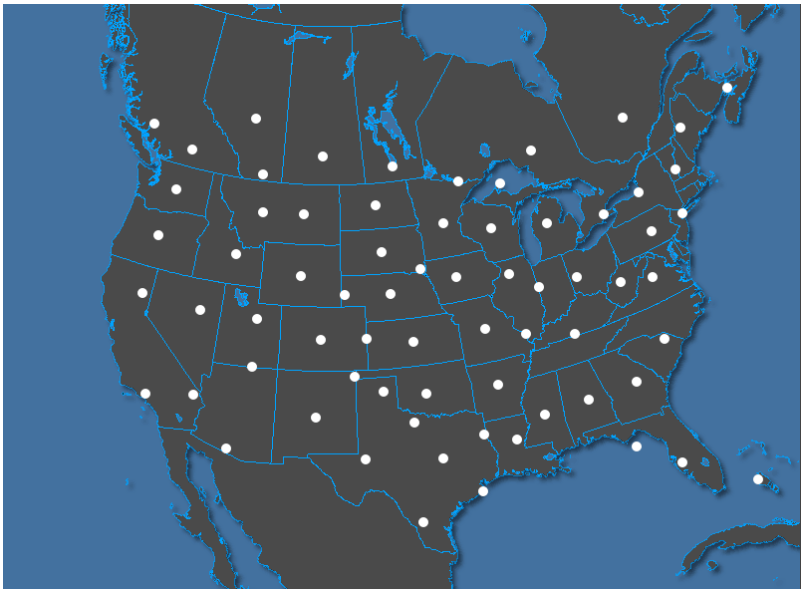

Figure 2.1: The domain of satellite images with 1 km resolution provided by Geostationary Operational Environmental Satellite system (GOES).

The models include the Global Forecast System(GFS), the ETA models, and the Rapid Update Cycle (RUC) models. The data sets are stored in GRIB format, and they contain gridded model output. We use the output of this model as the WRF input. The output contains analysis fields and forecast hours for multiple parameters and levels. The domain of our input covers North America. Table 1 provides detailed information of the gridded data. The dataset comes from the Continental United States (CONUS), 12 km resolution, gridded data, which is developed by North American Mesoscale Forecast System (NAM), supported by National Operational Model Archive and Distribution System (NOMADS). The NAM data contains dozens of different weather parameters and its domain covers the whole north America. To match the meteorological domain with the computational domain chosen, the WRF

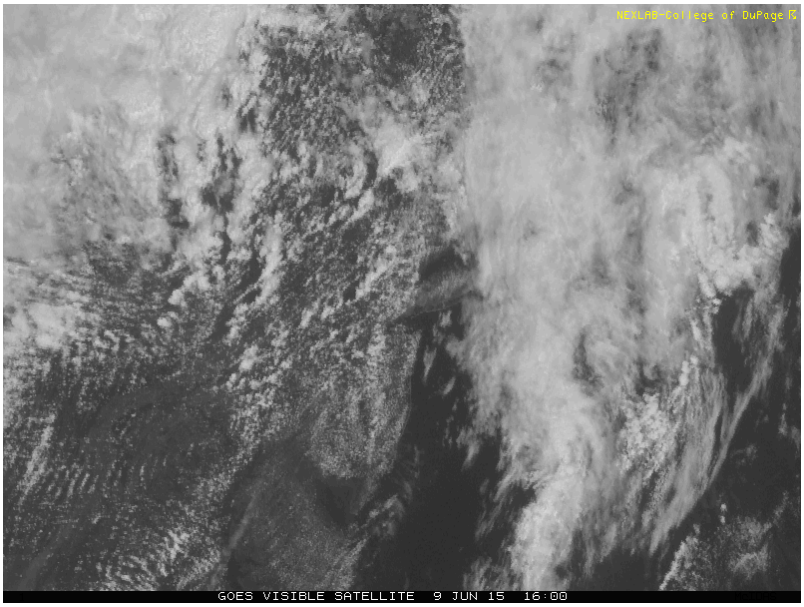

Figure 2.2: Satellite images are taken at 12pm UTC June, 9th, 2015, with the center point located at 40°48'00.0"N 73°18'00.0"W. Shown is 745X548 pixels, each 1 km^2. Raw satellite image provided by the Geostationary Operational Environmental Satellite system (GOES).

Preprocessing System (WPS) serves to define the model domains and interpolate the geographical data to the computational grids. The computational domain is centered at 40°48'00.0"N 73°18'00.0"W, with domain size 745x548 km^2

2.2 WRF

The Weather Research and Forecasting Model (WRF) is a mesoscale numerical weather prediction system [34], which features multiple dynamical cores, a 3-dimensional variational (3DVAR) data assimilation system, and a

software architecture allowing for computational parallelism and system extensibility. It provides a flexible and efficient model which is designed to serve both operational forecasting and atmospheric research needs. By its advances in physics, numerics, and data assimilation, WRF allows researchers to conduct simulations with either real data (real.exe) or idealized configurations, and it is suitable for a broad spectrum of applications across scales ranging from meters to thousands of kilometers, from local to global simulations. Its spectrum of physics and dynamics options reflects the experience and input of the broad scientific community. The WRF-Var variational data assimilation system accepts a host of observation types in pursuit of optimal initial conditions, while its WRF- Chem model provides the capability of air chemistry modeling. The WRF effort has been collaborative among the National Center for Atmospheric Research's (NCAR) Mesoscale and Micro-scale Meteorology (MMM) Division, the National Oceanic and Atmospheric Administration's (NOAA) National Centers for Environmental Prediction (NCEP) and Earth System Research Laboratory (ESRL), the Department of Defense's Air ForceWeather Agency (AFWA) and Naval Research Laboratory (NRL), the Center for Analysis and Prediction of Storms (CAPS) at the University of Oklahoma, and the Federal Aviation Administration (FAA), with the participation of university scientists. Nowadays WRF broadly serves real-time NWP, data assimilation development and studies, parameterized-physics research, regional climate simulations, air quality modeling, atmosphere-ocean coupling, and idealized simulations [35] [40].

2.2.1 WPS

The WRF Preprocessing System (WPS) is a set of three programs. Geogrid, Metgrid, and Ungrid, see Fig. 2.4. The collective roles for these programs are to prepare intermediate files to the real program for real-data simulations. Each of the programs performs one stage of the preparation:

The first one, Geogrid, specifies the simulation domains, and interpolates various terrestrial data sets to the model grids. The simulation domain is defined using information in the input file of the WPS. There are three choices of map projections provided in WPS: polar stereographic, Lambert-conformal, and Mercator. In this study, we choose the Lambert-conformal in our prediction as in Fig. 2.5 . By default, and in addition to computing latitude and longitudes for every grid point, Geogrid will interpolate soil categories, land use category, terrain height, annual mean deep soil temperature, monthly vegetation fraction, monthly albedo, maximum snow albedo, and slope category to the model grids. Output from Geogrid is written in the WRF I/O API format, and thus, by selecting the NetCDF I/O format, Geogrid can be made to write its output in NetCDF for easy visualization using external software packages. The Geogrid module is performed only once when the computational domain is chosen.

Second, the Ungrib program decodes the external analysis and forecast data from the GRIB format into intermediate formats to WRF. The GRIB formats contain time-varying meteorological fields. They are typically from another regional or global model, such as NCEP's NAM or GFS models. It supports both GRIB1 and GRIB2 formats, also native and hybrid grid data.

Ungrib writes intermediate data files in three user-selectable formats, which is not limited to be used by WRF [3].

Third, the Metgrid horizontally interpolates the intermediate-format meteorological data from Ungrid in multi simulation domain by the Geogrid program and it also interpolates meteorological fields to WRF *eta* levels. Unlike Geogrid, Metgrid is performed in the initialization step. Output from Metgrid is written in the WRF I/O API format, and thus, by selecting the NetCDF I/O format, Metgrid can be made to write its output in NetCDF for easy visualization using external software packages, including the new version of RIP4.

2.2.2 ARW driver

The ARW (Advance Research WRF) model is a fully compressible, non-hydrostatic model (with a hydrostatic option). It is built up by the ARW dynamics solver together with other compatible components in the WRF system as in Fig. 2.7. The vertical coordinate is a terrain-following hydrostatic pressure coordinate, which we will introduce later. The grid staggering is the Arakawa C-grid, and the model uses higher-order numerics, which includes the Runge-Kutta 2nd- and 3rd-order time integration schemes, and 2nd- to 6th-order advection schemes in both horizontal and vertical directions. It uses a time-split small step for acoustic and gravity-wave modes, and the dynamics conserves scalar variables [28] [37].

Vertical Coordinate and Variables

The ARW equations are formulated using a terrain-following hydrostatic pressure vertical coordinate denoted by η and defined as

$$\eta = (p_h - p_{ht})/\mu \tag{2.1}$$

where

$$\mu = p_{hs} - p_{ht} \tag{2.2}$$

p_h is the hydrostatic component of the pressure, and p_{hs} and p_{ht} denotes values along the surface and top boundaries. The coordinate definition, proposed by Laprise (1992) [21], is the traditional η coordinate used in many hydrostatic atmospheric models. The range of p is from 1 at the surface to 0 at the upper boundary of the model domain, as in Fig. 2.6. This vertical coordinate is also called a mass vertical coordinate. The computational domain we use in this study consists of 74x46 horizontal grid cells, of 10 km resolution, with 27 mass vertical coordinates.

2.3 FronTier

2.3.1 Front Tracking Method

In compressible fluid dynamics, many applications need to simulate fluid flows with sharp fronts. These problems can be handled by solving the governing equations in integral form. Numerically, it can be solved by conservative finite difference schemes. However, these conservative scheme can be low order

when a sharp interface occurs. For instance, The first order Godunov method could result in excessive numerical diffusion and destroy the sharpness of the front. Consider to this difficulty, van Leer [22] and Colella conceived the second- order scheme of the Godunov method. The extension generates sharp discontinuities without overshoots and oscillations. In the Godunov method, Riemann problems are still the building blocks in resolving the jumps at cell boundaries, yet characteristic information is provided to maintain the high order of accuracy. However, in the extended scheme, the shock is spread over several meshes. The physics may be inaccurately represented by this extended scheme. The difficulty is in applying finite difference across a discontinuity, while most schemes deal with smooth regions of flow well. Richtmyer and Morton [31] proposed a scheme in 1967 which addresses the problem, called the front tracking method. The outline of the Front tracking method is as follows:

Initialize the geometry of the interface and physical states on the fix grid and front points.

Propagate points on the discontinuity, giving both the updated states and the new location of the tracked front. The jumps in the state variables across the fronts are handled based on the Rankine-Hugoniot equations, combined with the differential equations in characteristic form.

Regular grid points away from the discontinuities are updated using a finite difference scheme. No differences are allowed across a tracked wave.

Points which locate nearby fronts are updated through interpolation of states both on the front and neighboring interior states.

Points with neighbors lying on the opposite side of the front, are updated by computing an artificial neighbor lying on the discontinuity. This point is simply the intersection of the grid line with the front, and its states are computed by interpolation of the nearby front states. This artificial point is then used in a modified form of the difference equations which account for variable mesh spacing.

The front tracking method were first implement by James Glimm and his group [10] numerically by 1985. Since then the front tracking technique has been broadly used in different research fields, such as shock refractions, shock accelerated interface mixing, and the motion of saturation fronts in porous medium.

2.3.2 Front Tracking Interface (FTI)

Another important component of this the numerical model is the Front Tracking code FronTier, whose role is to move the cloud boundary dynamically, see [1] and [2]. We use the Application Programming Interface (API) of FronTier called Front Tracking Interface (FTI) from [17] and [16] (http://www.ams.sunysb.edu/fti), to access the passive tracking method and the interface data structure in FronTier. FTI provides an interface to connect WRF and FronTier. The cloud boundary level set function for initialization from satellite data and the cloud velocity field for every time step from WRF are assigned to FTI as client routines. With these client routines, FTI initializes and propagates the cloud boundaries. For this purpose, we set the average wind velocity within the η level from 5 to 10 out of 27 total η levels [26].

Table 2.1: The physical schemes applied in the WRF simulation. The schemes include microphysics (mp_physics), longwave radiation (ra_lw_physics), shortwave physics (ra_sw_physics), surface input source, and planetary boundary layer (bl_pbl_physics), and the cloud effects (iCloud).

mp_physics	WSM 3-class scheme
ra_lw_physics	RRTM scheme
ra_sw_physics	MM5 shortwave (Dudhia)
surface_input_source	SI/gridgen
bl_pbl_physics	YSU PBL scheme
iCloud	1 (turn on the cloud effect)

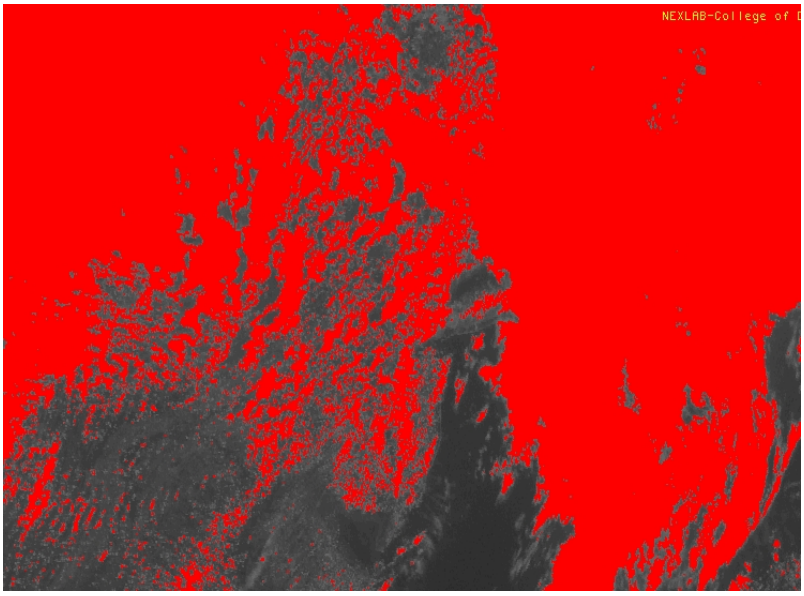

Figure 2.3: The processed satellite image with the Fig. 2.2. by Msphix, a satellite image processing and analysis library. By the satellite-image specific algorithm, the library recognizes the geographical terrain on the image, and then calculates the brightness level of each pixel. If the level of brightness exceeds a certain level, which is relatively decided by the overall brightness, then the pixel will be filled red (RGB code [254, 0, 0]) and identified as cloud. As a result, the raw satellite image will be masked with a red cover. The cloud cover boundary is the boundary of the red color mask.

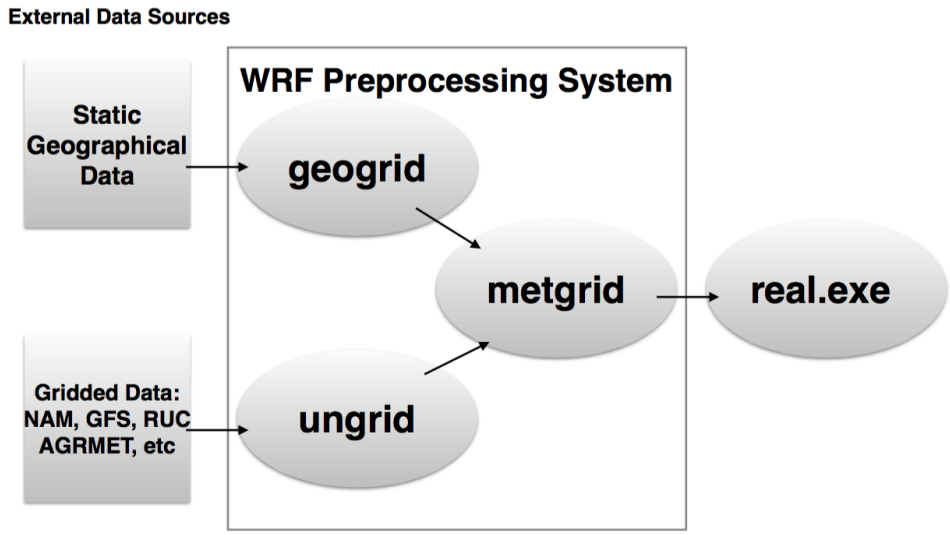

Figure 2.4: The program procedure of WRF preprocess system (WPS). It serves to analyze and interpolate the external meteorological data to formats WRF can interpret. It is a set of three programs to prepare the data for real-data simulations.

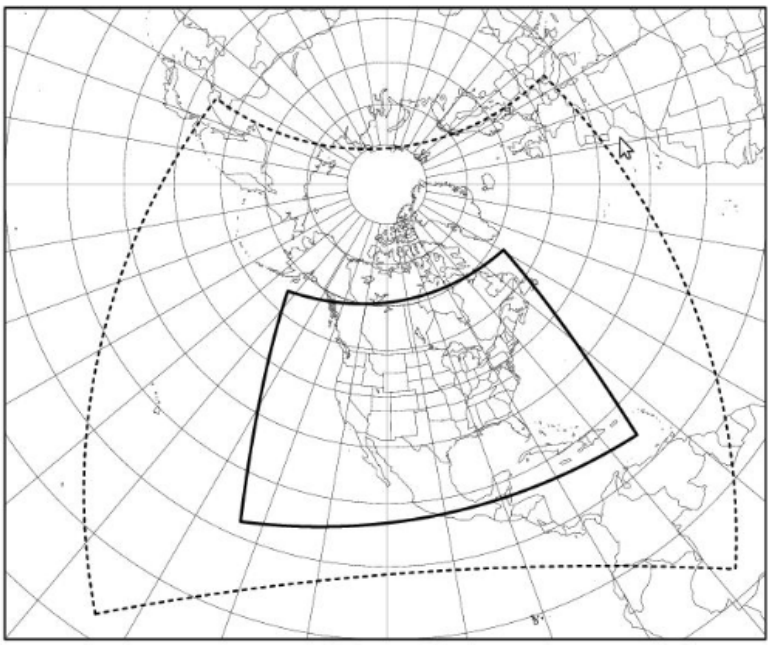

Figure 2.5: The grid domain using Lambert-conformal projection in WRF

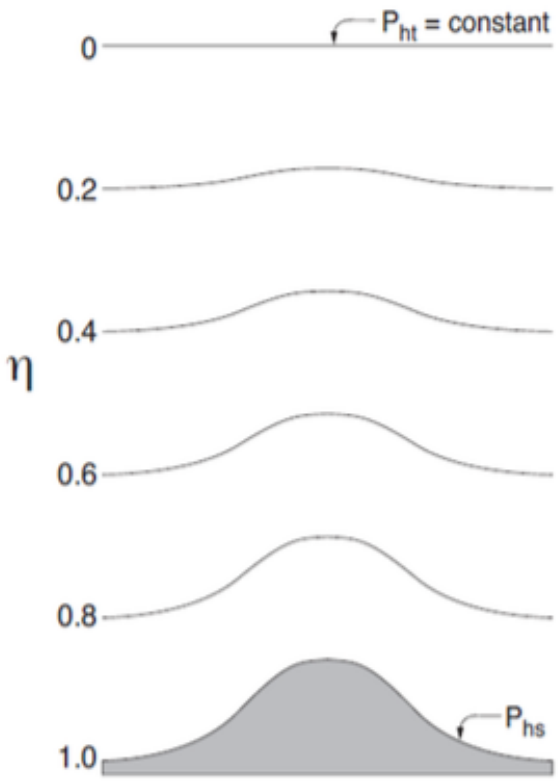

Figure 2.6: The traditional η coordinate used in atmospheric models, also called mass vertical coordinate.

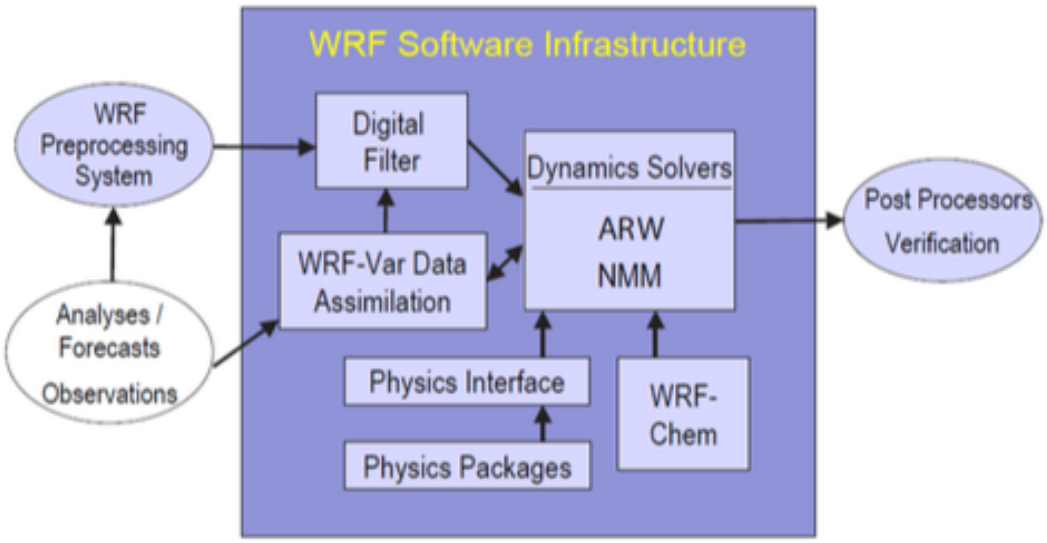

Figure 2.7: The main components in WRF System. The ARW solver is chosen in our simulation.

Chapter 3

Probabilistic Forecasting Model

3.1 Program Flow

In this study, we combine several programs to build our stochastic forecast model, including Msphinx, WRF and FronTier. To connect each program as module and to automate huge amount of data downloading/processing, we write high-level drivers in shell scripts and python scripts to control the whole program flow. The program flow is shown in Fig. 3.1. We also use crontab to automate the daily routine of downloading/ preprocessing the data.

3.2 The Error Probability Function

We build a probabilistic model for forecasting based on two types of errors, those proportional to the distance from the predicted cloud boundaries and those independent of this distance, each with a distinct physical basis. Wind speed errors introduce position error in the predicted cloud boundary, with the probability of error proportional to the distance from the boundary.

The appearance of new cloud boundaries associated with new clouds, on the other hand, is an error source independent of the distance to the predicted cloud boundary. In principle, there should also be an error source associated with new cloud boundaries coming from newly formed holes in the cloud cover, but these do not seem to play a significant role in the data we analyze. The error probability function, on which an error probability is based, is the result of the correlation of a prediction to a subsequent observation.

To build the probability model, we use satellite images from time 0 and t, the first for model initialization and the second for model validation and error modeling. FTI initializes both images as an FT interface. Both define the observed cloud boundaries. FTI propagates the time 0 FT interface for t time steps by the wind velocity field generated from WRF to obtain the predicted cloud boundaries. Then we call an FTI routine in each grid cell to identify it as sun or cloud. This is done for both the predicted domain and the observed domain. In each case, we assign components (sun = 0, cloud = 1) to each grid block.

We identify the occurrence of an error if at a specific time period and grid block base the predicted and observed components do not agree. Cut cells, these cut by an FT interface, are for simplicity assigned to have half area as sun and half as cloud. We next define the error area, Error Area$(d, d+\Delta d, t)$, as the error in which an error occurs, while lying within a distance $[d, d+\Delta d]$ of the predicted cloud boundary. Similarly we define the total area, Total Area$(d, d+\Delta d, t)$, also associated with the distance interval $[d, d+\Delta d]$ to the boundary. The cumulative areas defined above result not only from a

sum over space but also over multiple time events. Later, we will sort this integral according to values of physical parameters, to establish correlation to the error. Using these definitions, we define an error probability function

$$err = f(d|\ t) = \frac{\sum(\text{Error Area}(d,\ d+\Delta d,\ t))}{\sum(\text{Total Area}(d,\ d+\Delta d,\ t))}\ , \tag{3.1}$$

where d represents the distance to the boundary. The distance d is given a sign, positive for locations within predicted sun and negative within locations of predicted cloud. $\Delta d = 1$ pixels (1 km) represents the bin size of the distance interval, and t represents the given forecast period, i.e, the time elapsed between the initializing data and the predicted data. The sums run over grid cells and multiple prediction events.

The data assimilation improves the cloud cover prediction [20]. The cloud fraction (CLDFRA) variable in WRF is indirectly calculated by temperature, relative humidity and other related variables [5]. Assimilated satellite data improved the prediction (reduce the probability of error). In prediction mode, we determine a WRF-error probability for cloud cover, as a function of the signed distance to the WRF-FTI predicted cloud boundary. See Fig. 3.2. In this function, we use 1 km resolution prediction while using 5 km resolution for probability assessment.

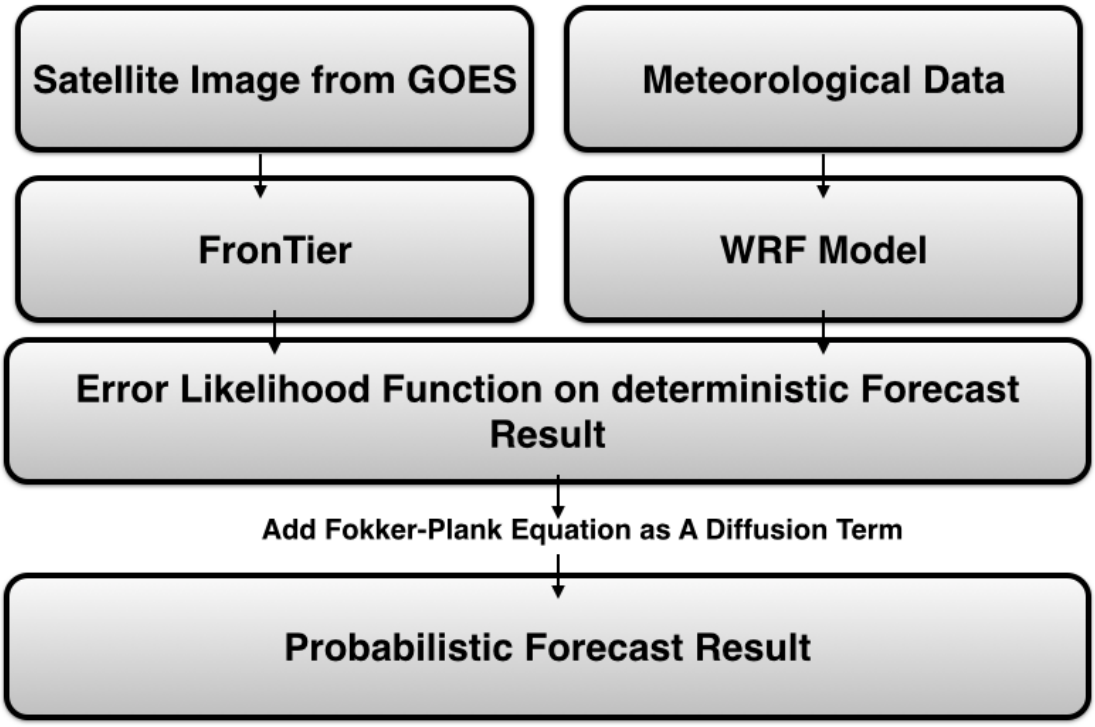

Figure 3.1: The program flow. We combine several programs to build our stochastic forecast model, including Msphinx, WRF and FronTier. Two types of data are needed for initialization. Observed one from satellite images and the meteorological data from NCEP. Then we run the preprocessing system WPS in WRF to build the computational domain we need. Next, FronTier is called to initialize the FT-Interface. Every time step in the forecast period, the velocity will generate from WRF to propagate the FT-Interface from FronTier. We build the error probability function for every forecast period to obtain the variance of diffusion. The next stage of our model is advancing the probability by Fokker-Planck equation. Using the diffusion rate and the source term, we develop the probability map to observe the evolution of probability.

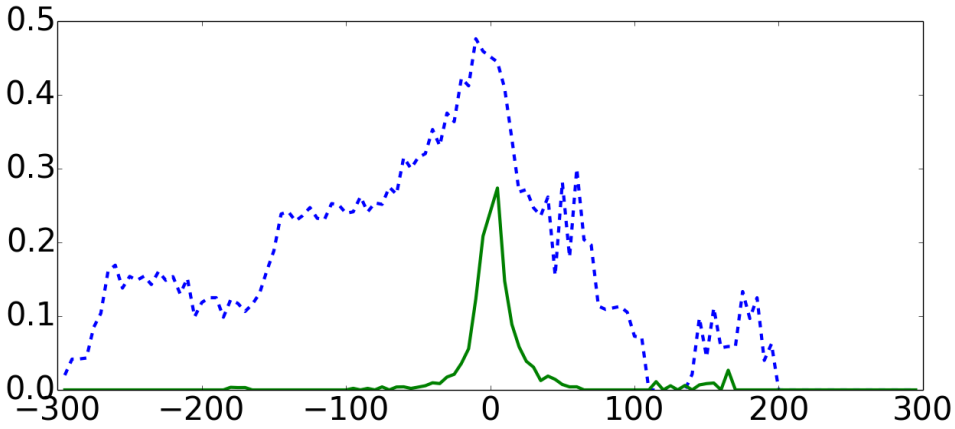

Figure 3.2: A plot of the error probability ratio (error area/ total area) vs. the signed distance (km) of 30 minutes forecast period. The green solid line represents the error probability function built based on the data assimilated forecast model; the blue dotted line represents the error probability function built with the cloud fraction result from NWP model. The variable of cloud fraction in WRF is calculated by other variables.

Chapter 4

Parameter Estimation

4.1 The Two Parameter Probability Error Model

4.1.1 Non-Linear Least Squares Method

The non-linear least squares method is used to fit a set of m observations with a n parameters model, using non-linear regression. The basis of the method is the use of successive iterations of linear regressions to refine the fitting parameters [25]. Consider observations $\mathbf{x}$ with a sample size of m, and a model function $\mathbf{y} = f(\mathbf{x}, \beta)$. β is a vector of size n. In the non-linear least squares method, we minimize the sum of the residuals,

$$S = \sum_{i=1}^{m} r_i^2, \tag{4.1}$$

where

$$r_i = y_i - f(x_i,\ \beta) \tag{4.2}$$

where r_i is the residual of the i^{th} out of m observations. Then we solve for the fitting parameter vector by the Gauss-Newton algorithm. We use a python module in Scipy called Statsmodel [32] to solve the non-linear curve fitting for us. Statsmodel provides classes and functions for the estimation of various different statistical models, statistical tests, and statistical data exploration. The results are tested against other existing statistical packages to ensure the correctness of the fitting.

4.1.2 Curve-Fitting

We observe that the distribution of error in prediction is a combination from two different sources, which can be observed from Fig. 7.1. As a result, a probabilistic model is built based on two physical principles: (a) the error in the wind speed, and (b) the spontaneous appearance of cloud boundaries.

We fit (3.1) to the model distribution

$$f(x|C,\overline{D}) = \frac{1}{2}\, e^{-\frac{x^2}{2\overline{D}}} + C \cdot H(x). \tag{4.3}$$

Here the signed distance x to the cloud boundary is positive in a region of predicted sun and negative in a region of predicted cloud. From the distribution (4.3) fit to observed data, the propagation error is parameterized as a sum of proportional to a normal distribution with a variance $\overline{D}$, plus a formation

error with proportional to a Heaviside function,

$$C \cdot H(x) = \begin{cases} 0, & x < 0, \\ C, & x > 0. \end{cases} \tag{4.4}$$

A non-linear least squares regression analysis fits these two error probability function parameters.

The variance $\overline{D}$ grows approximately linearly with time, and from this fact, we define the diffusion rate $D = \overline{D}/t$, where $t = 1$ minute, the smallest time interval considered here. Note that D is a numerical approximation to $\partial\overline{D}/\partial t$ and the average of those time periods. We find a relationship between the diffusion rate D with the average wind velocity. The horizontal wind velocity $\vec{U}$ is extracted and averaged spatially from the wind velocity field of WRF. We collect those time period with similar mean wind velocity into bins, and then apply the curve-fitting method to every bin in order to obtain the diffusion rate D under different wind velocity condition. In this way, we obtain a series of diffusion rates related to wind velocity.

A similar analysis relates the constant C to the surface evaporation rate (SFCVP) extracted from WRF. Now the probabilistic model is fully developed with two parameters, proportional to a normal distribution with the variance $\overline{D}$ and a step function $C \cdot H(x)$.

Chapter 5

Fokker-Planck Equation Implementation

5.1 Fokker-Planck Equation

The Fokker-Planck equation, also known as the Kolmogorov forward equation, is named after Adriaan Fokker and Max Planck. The Fokker-Planck equation is a partial differential equation which describes the evolution of a probability density function, [39] [7], due to transport, diffusion, and as presented here the spontaneous generation of probabilities [15].

Consider an Ito process driven by the standard Wiener process W_t. It can be described as a stochastic differential equation

$$dX_t = \vec{U}dt + \sqrt{D(X_t, t)}dW_t \tag{5.1}$$

where $\vec{U}$ represents the drift, and the diffusion rate $D(X_t, t)$. From (5.1), the evolution of the probability density $p(x, t)$ of the random variable X_t can be derived,

$$\frac{\partial}{\partial t}p(x,t) + \frac{\partial}{\partial x}\left[\vec{U}p(x,t)\right] = \frac{\partial^2}{\partial x^2}\left[\frac{D(x,t)}{2} \cdot p(x,t)\right] . \tag{5.2}$$

5.2 Stochastic Cloud Cover Forecast

$$\frac{\partial p(\mathbf{x},t)}{\partial t} + \nabla \cdot \vec{U} p(\mathbf{x},t) = \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \frac{\partial^2}{\partial x_i \, \partial x_j} \left[\frac{D}{2} \cdot p(\mathbf{x},t) \right] + C \cdot H(x) \ , \quad (5.3)$$

where $N = 2$ and D as the diffusion rate, to model the dynamic evolution of the probability density.

In (5.3), $p(x,t)$ is the probability of cloud existence at a given time and location. The left hand side displays the dynamic change in the probability density and the advection term. The wind velocity $\vec{U}$ governs the advection term, which describes cloud boundary dynamics as driven by the wind. The right hand side displays two terms: the probabilistic diffusion and the source term. The diffusion rate D governs the growth of the error through time. C is determined by a parameterized physical factor, the surface evaporation rate, from WRF.

To solve the Fokker-Planck equation, we discretize the equation using the finite-volume method [18] [19] in space and a temporal upwind scheme to reduce the spatial oscillations, [29], [24] . We initialize the domain by assigning two components 0 and 1, to represent sun and cloud respectively. We then solve the Fokker-Planck equation with the WRF choice of 1 minute time steps, to create a probability map. On the map, each cell is assigned with a value which represents the probability of cloud cover presence at each given time step.

Chapter 6

Backtesting

6.1 Model Validation

To validate our dynamic probability model, we access mismatches between the observed probability and predictions. In the probabilistic model, every grid cell on the domain is assigned a probability of cloud existence. We collect data points within common probability intervals in bins of probability events. For points inside each bin, we calculate the observed probability. The model generated probability and observed probability are plotted against each other in Fig 7.8. A point (x, y) on the plot corresponds to model generated probability (y-coordinate) against the observed probability (x-coordinate). The 45-degree line represents the perfect fit of these two probabilities.

6.2 The Statistical Tests

As model validation, we compare the finite sample from the observation dataset with the model-generated probability. First, we form a null hypothesis that the observed probability is equal to the model-generated probability,

$$H_0 : p_{observed} = p_{model} \ . \tag{6.1}$$

A statistical error in the sample mean is generated by the finite size of the observation data. The finite sample mean deviates from the true (infinite sample) mean by a χ^2 distribution. We investigate the sampling error to differentiate the statistical error due to the finite sample size from the error caused by the model itself. We classify our events into 10 probability intervals. Then the null hypothesis for each bin $[p_i, p_{i+1}]$ can be rewritten as

$$\begin{cases} H_0 : & p = (p_i + p_{i+1})/2 \\ H_1 : & p \neq (p_i + p_{i+1})/2 \end{cases} \tag{6.2}$$

To test the hypothesis, we conduct the χ^2 test and calculate the non-rejection intervals with the 95% confidence level for each probability interval. We draw the non-rejection interval as error bars on the probability plot. The model error can be distinguished from the error due to finite size statistical data analysis accordingly.

From the result in Fig. 7.9, we define the model error as the difference of

predicted probability and observed probability in each bin.

$$err = prob_{predicted} - prob_{observed} \tag{6.3}$$

In this case, we form another null hypothesis that the error rate of our probability model is lower then 0.2,

$$H_0 : err < 0.2 \tag{6.4}$$

The region of rejection is on only one side of the the sampling distribution. We use the one-tailed test to examine the null hypothesis.

6.3 The χ^2 Test

To test the fraction of exceptions for the probabilistic model, a recommended technique in backtesting is the χ^2 Test. To conduct this test, we define the violation process I as the prediction of

$$\begin{cases} 0, & \text{if the cloud cover component is equal to 1} \\ 1, & \text{if the cloud cover component is equal to 0} \end{cases} \tag{6.5}$$

The failure rate can be defined as x/n, where n is the number of trials. Thus the violation x follows a binomial probability distribution

$$f(x) = \binom{n}{x} p^x (1-p)^{n-x} \tag{6.6}$$

where the number of violations, x, the total number of observations, n, and $1-p$ equals to the confidence level. To check the observed mean violation rates, we apply a χ^2 test. χ^2 defines the statistics of an observed (sample) mean as defined by a finite sample drawn from an infinite source of sample. It is a measure of the finite size effect on the sample mean.

To calculate the degree of freedom of χ^2 distribution, we apply the formula of the degree of freedom (DF),

$$DF = (r-1) * (c-1) \tag{6.7}$$

where r is the number of categories, and c is the number of levels of categorical variables. Since we have both observed and predicted events, as well as binary states for both events, we can calculate our degree of freedom as

$$(2-1) * (2-1) = 1. \tag{6.8}$$

With the large size of observations, the binomial distribution can be well-approximated by the normal distribution. Thus we have

$$z = \frac{x - np}{(np(1-p))^{1/2}} \approx N(0,1) \tag{6.9}$$

where np is the mean, and $(np(1-p))^{1/2}$ is the standard deviation. In this case, the model will be rejected if

$$x \notin [np - z \cdot [np(1-p)]^{1/2}, np + z \cdot [np(1-p)]^{1/2}] \tag{6.10}$$

Chapter 7

Numerical Results

7.1 Probabilistic Forecasting Model

With the observation from satellite images and the meteorological initialization data for WRF, we show the predicted and observed cloud boundaries for prediction periods of 15, 30, 45, and 60 minutes in Fig 7.1. The predicted cloud boundaries track the observed ones well in the 30 minutes forecast period. However, the predictive power of the cloud boundary model is decreasing with time. To better analyze the error, we build the error probability function as in Fig. 7.2 to establish the relationship between error and the distance to boundaries.

7.2 Parameter Estimation

We apply the curve fitting analysis to all 15, 30, 45 and 60 minutes forecast periods, with the standard deviation of 7.51, 12.89, 18.57 and 20.99 km respectively. We observe that the variance grows approximately linearly

in time. The results of fitting the error probability function to the model distribution are shown in Fig. 7.2. The error probability functions in four different forecast periods basically retain the same shape which is consist with the sum of proportional to a normal distribution and proportional to a Heaviside function as designed. However, the peak error value is increasing when the forecast period becomes longer. The highest value, which occurs near the cloud boundary, lies in the middle of the error probability function.

To examine the goodness of fit of the two-parameter model and the observed data as given by the probability from the error probability function, a quantile-quantile plot (Q-Q plot) is presented. The Q-Q plot compares two probability distributions graphically, with the two sets of quantiles plotted against each other. If two distributions agree after linear transformation, the points in the Q-Q plot will show a straight line. In Fig. 7.3, we observe that the model distribution successfully models the errors, not only in the center part dominated by the proportional to normal distribution, but also in the tails of the distribution.

We apply a second degree polynomial to describe the relationship between the constant C and surface evaporation rate. The goodness of fit to the data is assessed from covariance matrix of the parameters, with the diagonal used to calculate the variance. We similarly analyze the relationship between D and wind speed. The upper and lower curves with a 95% confidence level are plotted in Fig 7.5.

7.3 Stochastic Cloud Cover Forecast

With the parameters $\vec{U}$, D, and C, we solve Fokker-Planck equation numerically to build the probability map as shown in Fig. 7.6 and Fig. 7.7. The white area represents cloud, and the blue area is open sky. At time step 0, as shown in the upper-left, the map is initialized with sharp boundaries. Over time, the boundaries become fuzzier due to the effects of diffusion. At the cloud edge, the blue gradually grows faint with time, which means that the uncertainty is increasing along cloud boundaries. While we lose the forecast power near the boundary, we retain a strong forecast power further from the boundaries.

7.4 Backtesting

Fig. 7.8 shows the comparison of observed and the model-generated probability, with the sampling error caused by estimating the model by a finite number of observations. While the Q-Q plots show the goodness of the model fit, the probability plots in Fig. 7.8 show the predictive power of the probabilistic model. The plots show good predictive capability beyond 30 minutes, and reasonable prediction results up to one hour. The predictive power starts to decrease first in the sunny region. As expected, the null hypothesis is rejected. This means that the model is not perfect and that the observed errors in prediction, while small, are model limitations more than statistical and data analysis limitations. With this knowledge, we assess the size of the model related errors, to conclude that the model gives good accuracy for up to 30

minutes and probably useful accuracy for an hour.

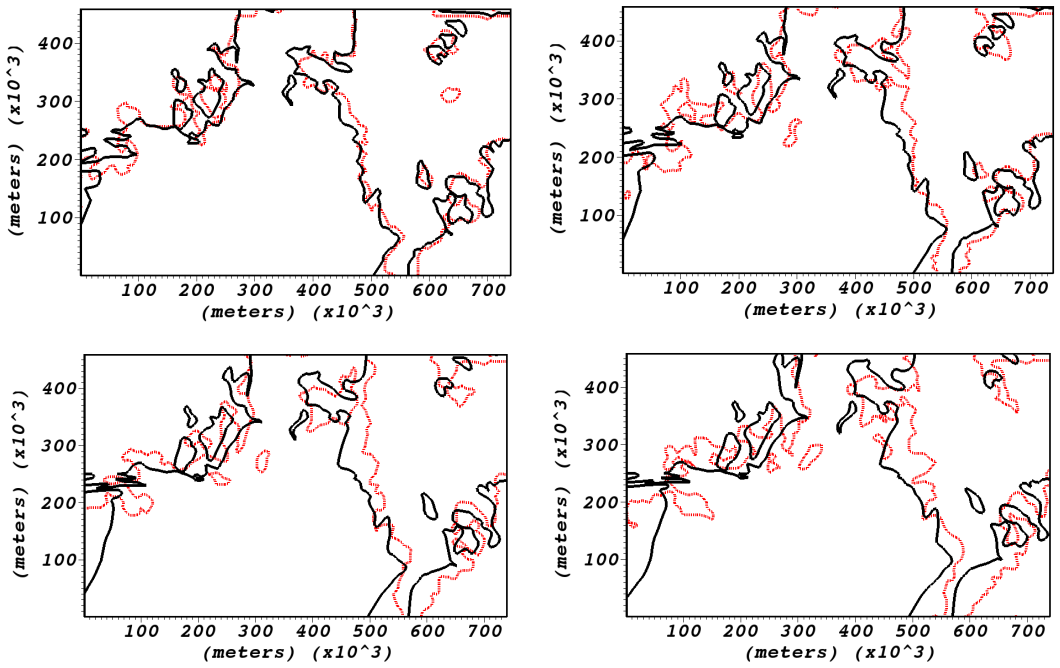

Figure 7.1: These plots are generated from the same date and same location as in Fig. 2.3. They show the comparison between observation (red dotted curves) and forecast (black solid curves) after 15, 30, 45, and 60 minutes (upper-left, upper-right, lower-left, lower-right). (Color figures available online and in [14]) In these plots we can recognize two different kinds of error as a mismatch between observed and predicted cloud boundaries due to a) cloud propagation error. b) the appearance of new cloud boundaries (i.e., the dotted circle in the upper-right corner of the 15 minutes plot) due to the formation of new clouds.

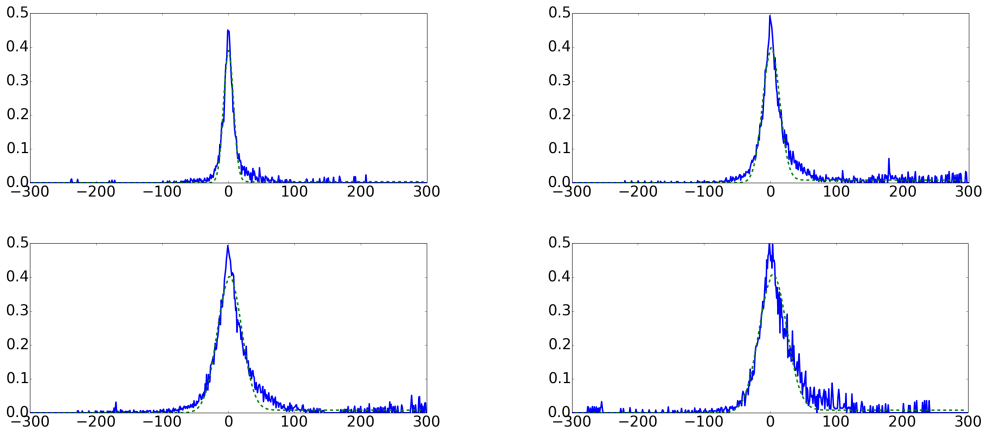

Figure 7.2: A plot of the error probability (error area/ total area) vs. the signed distance (km) to the predicted cloud boundaries and a two parameter fit of the probabilistic distribution of observed data for 15, 30, 45, and 60 minutes period (upper-left, upper-right, lower-left, lower-right). Positive distances represent predicted sun while negative distances represent the predicted clouds. The solid line is the observed error probability function; the dotted line is the model distribution from the proportional to a normal distribution and a step function. Note the asymmetry of the plots, with the deviation from the proportional to normal distribution occurring in the sunny (positive) portion of the data only.

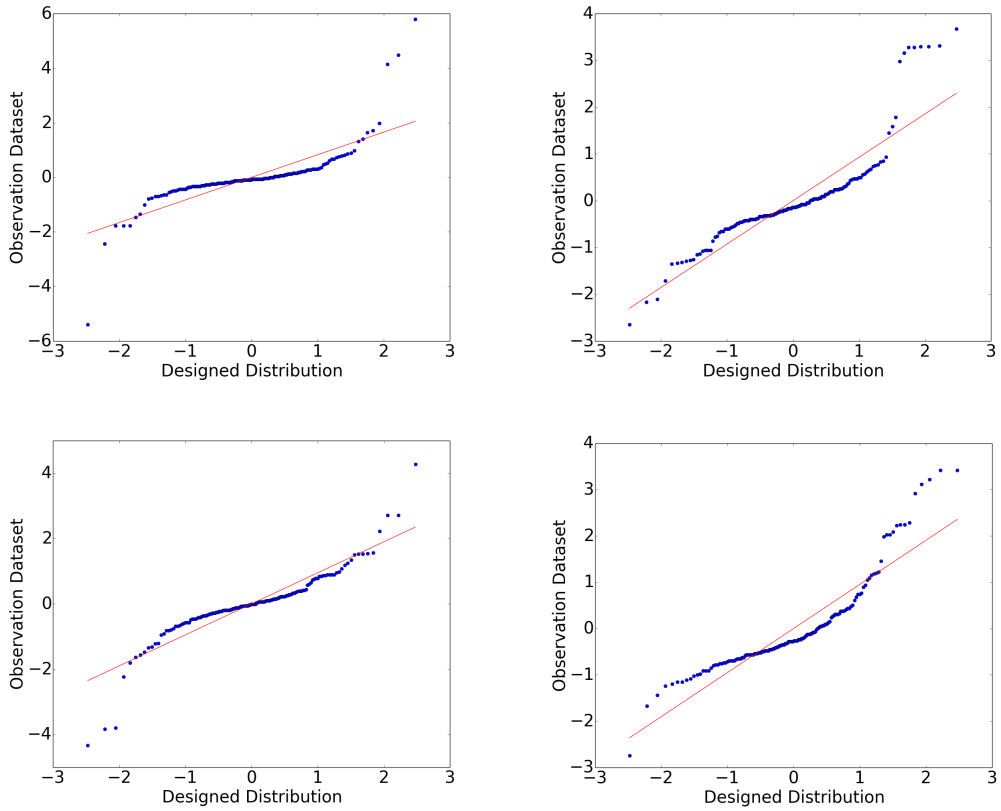

Figure 7.3: The Q-Q plots represent the goodness of fit of the model to data. We show the comparisons between the observed error probability function and the fitting distribution for periods of 15, 30, 45, and 60 minutes (upper-left, upper-right, lower-left, lower-right). The straight line represents an ideal goodness of fit line. The dots are the quantile distribution of the datasets. In all plots, the data fits the model distribution well, except for some outliners. We conclude that the two parameter model fits observed data over a 1 hour period.

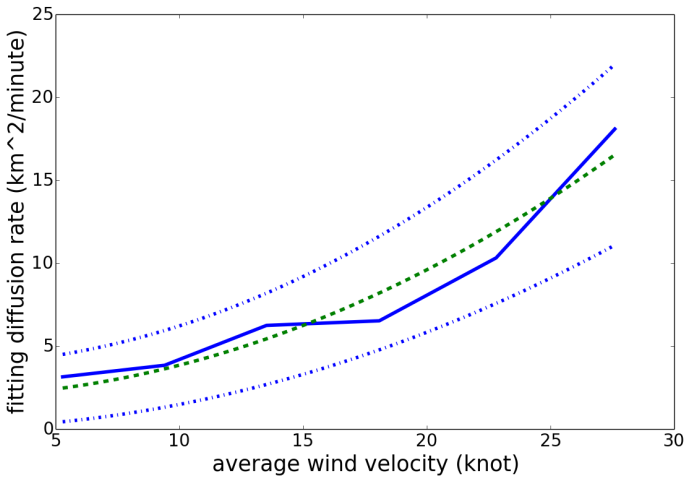

Figure 7.4: The diffusion rate D (km^2/min) plotted vs. the average wind velocity (knot); To find the relationship, we obtain the observed variables for each time step. We then sort the numerical results for the cloud boundary by their level into bins. We conduct a regression analysis on the numerical cloud boundary results in each bin. As a result, we can obtain the estimated parameters. The solid line plots the observed relationship between the numerical results and the fitting parameters. A polynomial regression fit yields the dotted line. To investigate the polynomial fit quality, the covariance matrix of the fitting parameters is calculated. Using the diagonal of the covariance matrix, the variance is calculated to draw the upper curve and lower curve with 95% confidence level as the dash-dot line. In this figure, we can observe that the fitted curve does not exceed the tolerance interval with a 95% confidence level.

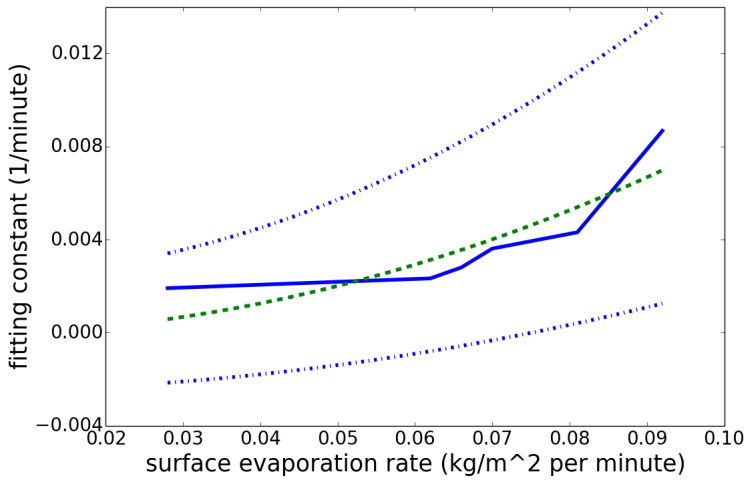

Figure 7.5: The RHS constant error term from the $H(x)$ vs. surface evaporation rate (kg/m^2 per minute); To find the relationship, we obtain the observed variables for each time step. We then sort the numerical results for the cloud boundary by their level into bins. We conduct a regression analysis on the numerical cloud boundary results in each bin. As a result, we can obtain the estimated parameters. The solid line plots the observed relationship between the numerical results and the fitting parameters. A polynomial regression fit yields the dotted line. To investigate the polynomial fit quality, the covariance matrix of the fitting parameters is calculated. Using the diagonal of the covariance matrix, the variance is calculated to draw the upper curve and lower curve with 95% confidence level as the dash-dot line. In this figure, we can observe that the fitted curve does not exceed the tolerance interval with a 95% confidence level.

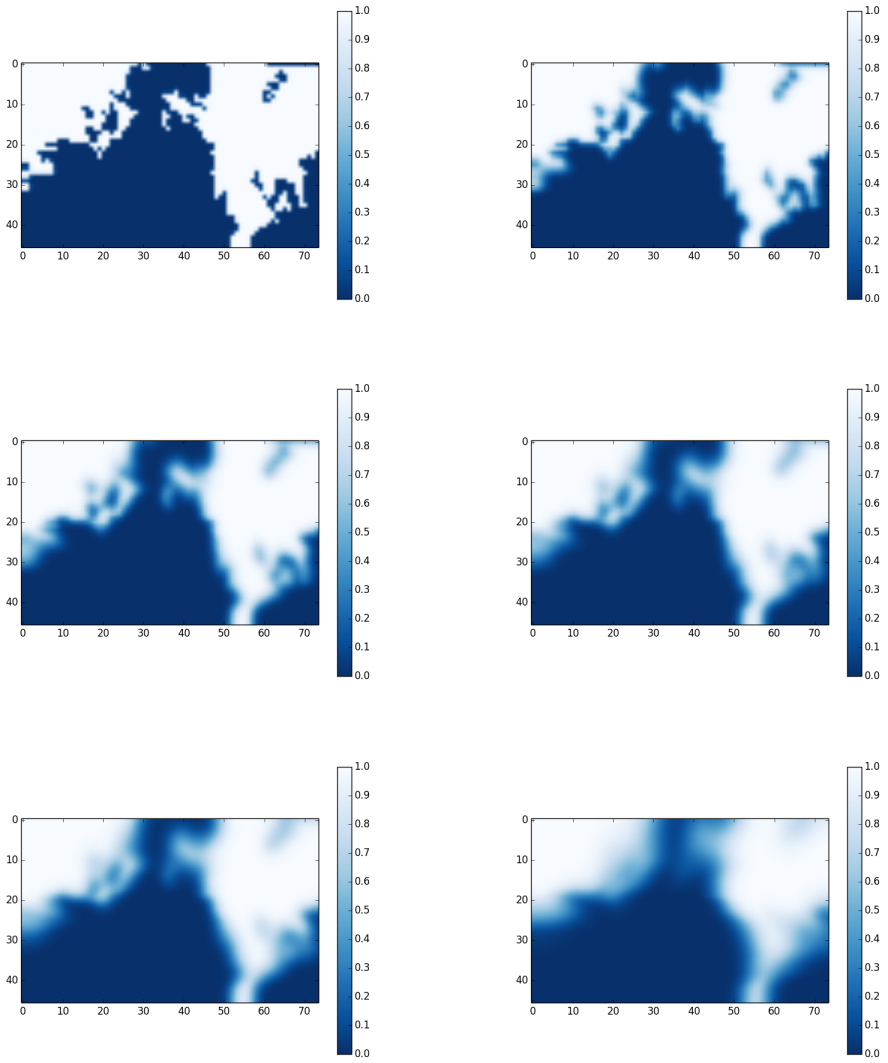

Figure 7.6: Propagation of the probability map. Initialization, and after 15, 30, 60, 120, and 180 minutes (upper-left, upper-right, mid-left, mid-right, lower-left, lower-right). The initial probability map is generated from the satellite image from the same date, location, and resolution as in Fig. 2.3. The average wind velocity is 15.28 knots.

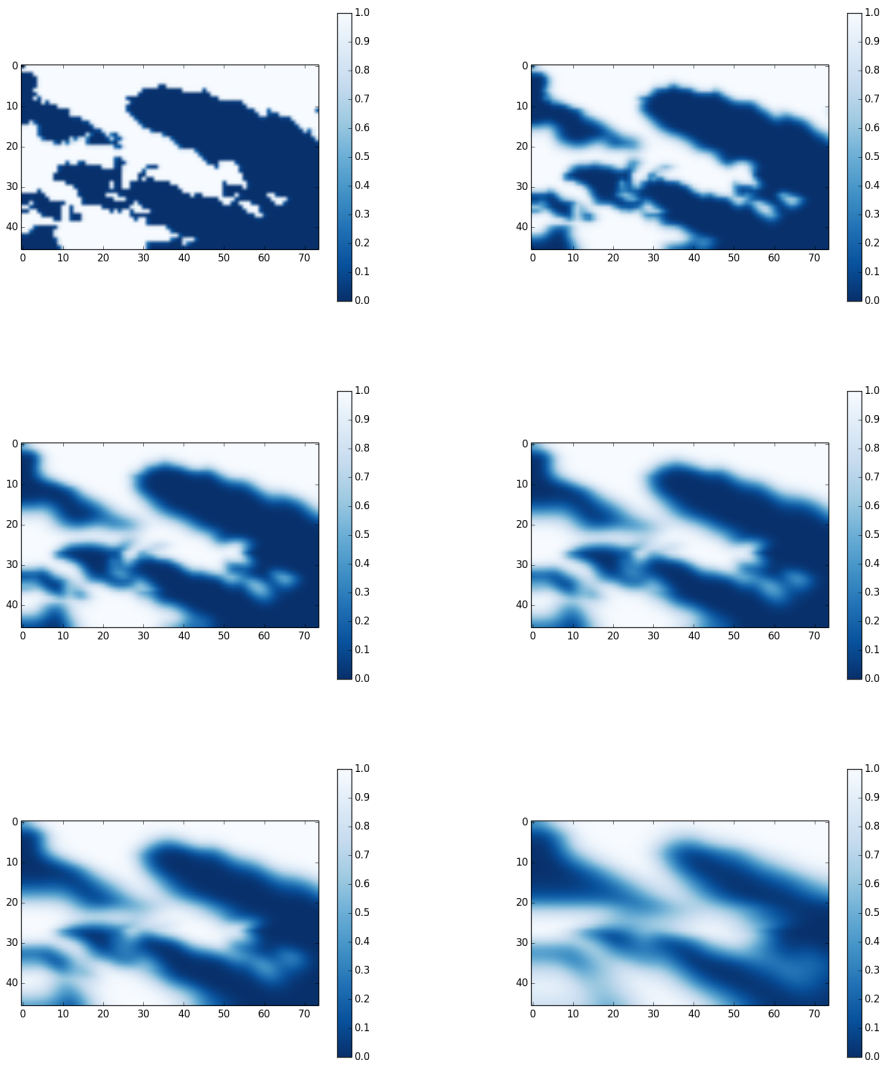

Figure 7.7: Propagation of the probability map. Initialization, and after 15, 30, 60, 120, and 180 minutes (upper-left, upper-right, mid-left, mid-right, lower-left, lower-right). The initial probability map is generated from the satellite image from at 12pm UTC March, 11th, 2015, with the center point located at 40°48'00.0"N 73°18'00.0"W. The average wind velocity is 26.23 knots. Compared to Fig. 7.6, the cloud boundaries are becoming blurred more rapidly, indicating a lose of model predictive power due to the higher wind velocity and turbulence.

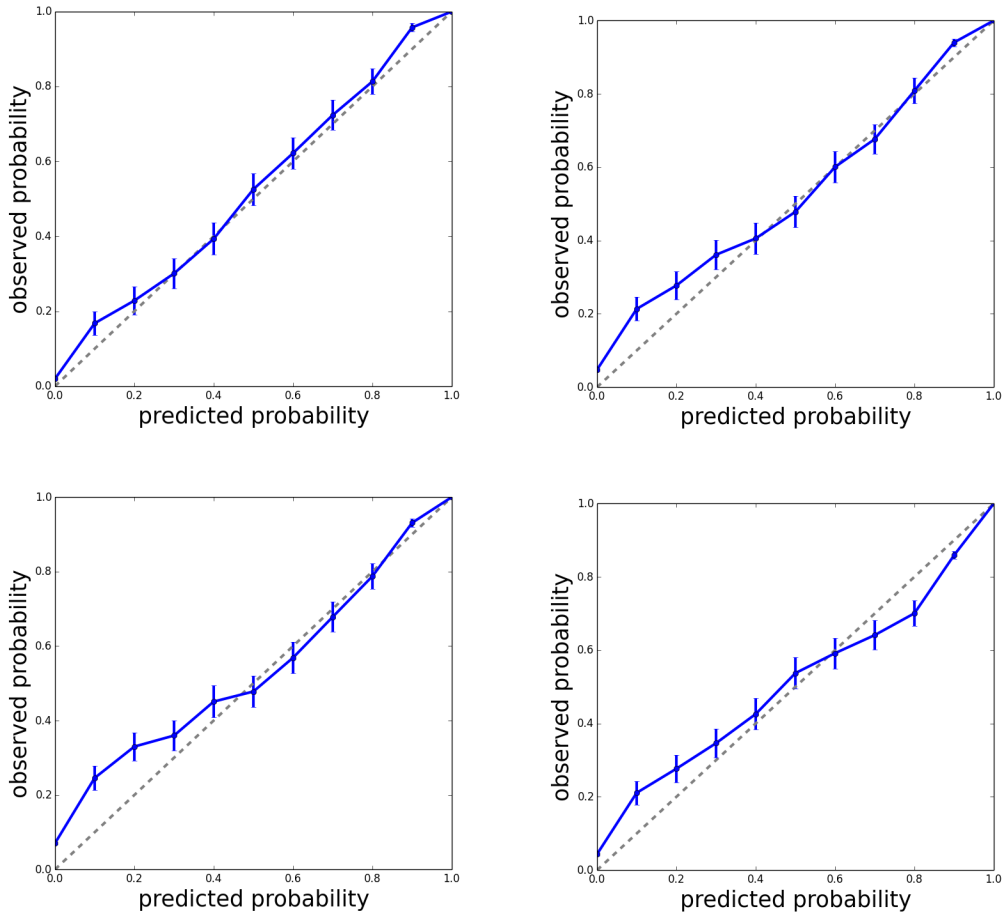

Figure 7.8: We plot the observed probabilities (y-axis) vs. the model generated probabilities (x-axis) for 15, 30, 45, and 60 minutes periods (upper-left, upper-right, lower-left, lower-right). The error bars show the statistical error of the observational finite sample. The observed probability for 15 minutes and 30 minutes fits well, while the 45 minutes and 1 hour forecast show some loss of the predictive power of the model.

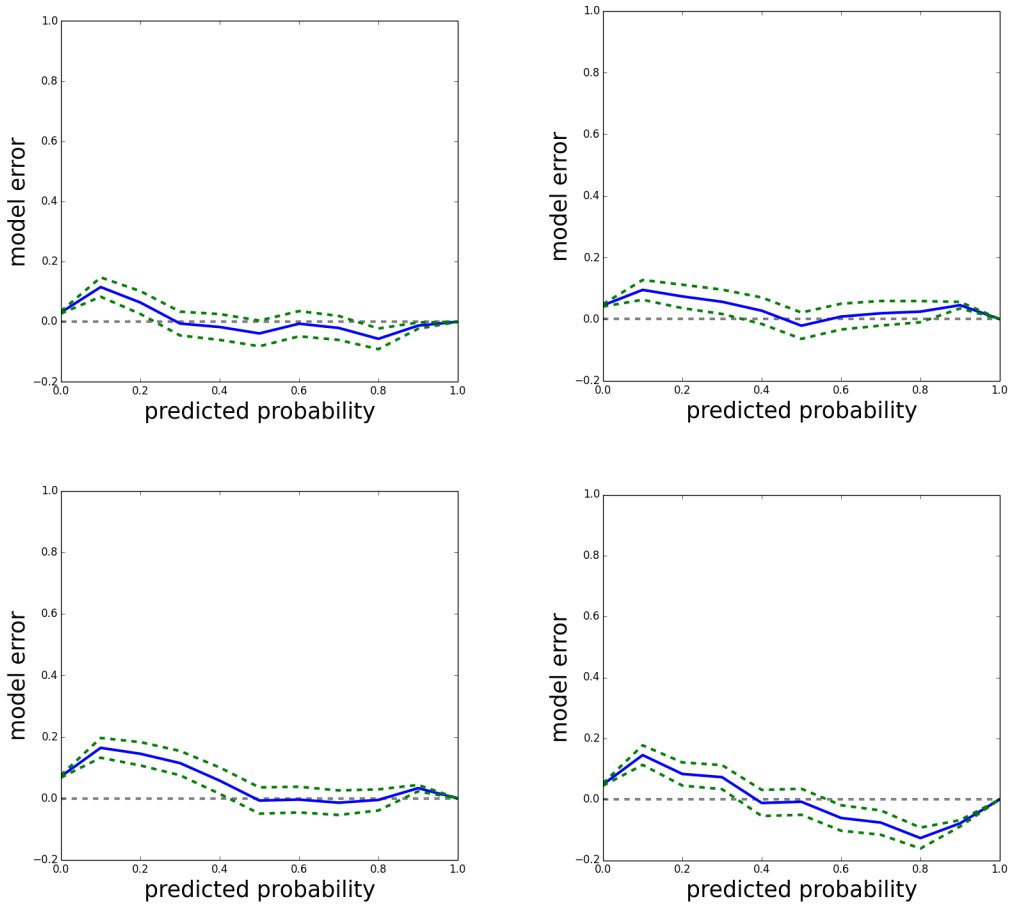

Figure 7.9: We plot the model error as y-axis vs. the model generated probabilities as x-axis for a 15, 30, 45, and 60 minutes period (upper-left, upper-right, lower-left, lower-right). The observed probability for 15 minutes and 30 minutes fits well, while the 45 minutes and 1 hour forecast show some loss of the predictive power of the model.

Chapter 8

Conclusions

We have developed a probabilistic methodology for weather forecasting. The key steps are a) A physics based probabilistic model. b) Parameter estimation. c) A stochastic equation for insertion into numerical weather prediction (NWP) models. d) Backtesting.

While the methodology appears to be applicable for broader contexts, the key feature of the short term cloud forecast problem chosen that allows the analysis to move forward is the availability of sufficient data. The methodology has been tested for the short term cloud cover forecast problem. The key data used are a) satellite image data and b) WRF meteorological data, including initialization protocols. The physics based model for the generation and motion of probabilities is based on two processes: velocity dispersion (small scale turbulence) in the wind field and surface evaporation phenomena for the generation of new cloud boundaries. We build a two parameter probability model for prediction based on the physics model, one parameter depending on the distance to the cloud boundaries and the other independent of this

distance.

As our third step, we insert the probability propagation model as a Fokker-Planck equation into WRF. The implementation is an independent module, which can be coupled not only with WRF but also with other numerical weather models. The module also serves as a stand alone post-processing tool. It generates probabilistic results based on the deterministic variables from WRF. In the fourth part of this study, we validate the prediction of the probabilities generated. Overall, we get good accuracy for 30 minutes and probably useful accuracies for 1 hour and more. The model has a tendency to overestimate the probability of cloud in a sunny region.

Bibliography

[1] W. Bo, B. Fix, J. Glimm, X. L. Li, X. T. Liu, R. Samulyak, and L. L. Wu. Frontier and applications to scientific and engineering problems. *Proceedings of International Congress of Industrial and Applied Mathematics*, 7:1024507–1024508, 2008.

[2] W. Bo, X. Liu, J. Glimm, and X. Li. A robust front tracking method: Verification and application to simulation of the primary breakup of a liquid jet. *SIAM J. Sci. Comput.*, 33, 2011.

[3] Melissa S. Bukovsky and David J. Karoly. Precipitation Simulations Using WRF as a Nested Regional Climate Model. *J. Appl. Meteor. Climatol.*, 48(10):2152–2159, October 2009.

[4] S. D. Campbell. A review of backtesting and backtesting procedures. In *Finance and Economics Discussion Series Divisions of Research Statistics and Monetary Affairs*, 2005.

[5] Fu-Lung Chang and James A. Coakley. Estimating errors in fractional cloud cover obtained with infrared threshold methods. *Journal of Geophysical Research: Atmospheres*, 98(D5):8825–8839, 1993.

[6] Jaime Daniels, Wayne Bresky, Steve Wanzong, Chris Velden, and Howard Berger. *GOES-R Advanced Baseline Imager (ABI) Algorithm Theoretical Basis Document for Derived Motion Winds*. NOAA NESDIS CENTER for SATELLITE APPLICATIONS and RESEARCH, USA, 7 2012.

[7] Niannian Fan, Deyu Zhong, Baosheng Wu, Efi Foufoula-Georgiou, and Michele Guala. A mechanistic-stochastic formulation of bed load particle motions: From individual particle forces to the fokker-planck equation under low transport rates. *Journal of Geophysical Research: Earth Surface*, 119(3):464–482, 2014.

[8] T. Fauchez, C. Cornet, F. Szczap, and P. Dubuisson. Assessment of cloud heterogeneities effects on brightness temperatures simulated with

a 3d monte carlo code in the thermal infrared. In *Radiation Process in the Atmosphere and Ocean*, volume 1531, pages 75–78, Corfu, Greece, 1 2011. AIP Publishing.

[9] J. Fonseca, T. Oozeki, T. Takashima, G. Koshimizu, Y. Uchida, and K. Ogimoto. Use of support vector regression and numerically predicted cloudiness to forecast power output of aphotovoltaic power plant in kitakyushu, japan. *Prog. Photovolt: Res. Appl.*, 20, 2012.

[10] J. Glimm, C. Klingenberg, O. Mcbryan, B. Plohr, D. Sharp, and S. Yaniv. Front tracking and two-dimensional riemann problems. *Advances in Applied Mathematics*, 6, 1985.

[11] Louis Gonzalez and Christine Deroo. The mgraph package was developed at loa by louis gonzalez and christine deroo, 2008.

[12] Harshvardhan, S. E. Schwartz, C. M. Benkovitz, and G. Guo. Aerosol influence on cloud microphysics examined by satellite measurements and chemical transport modeling. *Journal of the Atmospheric Sciences*, 59, 2002.

[13] Andrew K. Heidinger. A naive bayesian cloud-detection scheme derived from calipso and applied within patmos-x. *Journal of Applied Meteorology and Climatology*, 51, 2012.

[14] Ya-Ting Huang. *A stochastic formulation of short term cloud cover forecasts*. PhD thesis, Stony Brook University, Stony Brook University, NY11790, 2015.

[15] Richard Jordan, David Kinderlehrer, and Felix Otto. The variational formulation of the fokker-planck equation. *SIAM J. Math. Anal*, 29:1–17, 1999.

[16] R. Kaufman, H. Lim, and J. Glimm. Conservative front tracking: the algorithm, the rationale and the api. preprint, 2015.

[17] Ryan Kaufman. *Software Tools for Stochastic Simulations of Turbulence*. PhD thesis, Stony Brook University, Stony Brook University, NY11790, 2014.

[18] D. Kim and H. Choi. A second-order time-accurate finite volume method for unsteady incompressible flow on hybrid unstructured grids. *Journal of Computational Physics*, 162, 2000.

[19] J. Kim and P. Moin. Application of a fractional-step method to incompressible navier-stokes equations. *Journal of Computational Physics*, 59, 1985.

[20] Andrzej Z. Kotarba. Estimation of fractional cloud cover for moderate resolution imaging spectroradiometer/terra cloud mask classes with high-resolution over ocean aster observations. *Journal of Geophysical Research: Atmospheres*, 115(D22):n/a–n/a, 2010. D22210.

[21] R. Laprise. The euler equations of motion with hydrostatic pressure as an independent variable. *Monthly Weather Review*, 120, 1992.

[22] B. V. Leer. Towards the ultimate conservative difference scheme. v. a second-order sequel to godunov's method. *Journal of Computational Physics*, 32, 1979.

[23] M. Legrand, A. Plana-Fattori, and C. N'doum. Satellite detection of dust using the ir imagery of meteosat: 1. infrared difference dust index. *Journal of Geophysical Research: Atmospheres*, 106(D16):18251–18274, 2001.

[24] F. Liu, V. Anh, and I. Turner. Numerical solution of the space fractional FokkerPlanck equation. *Journal of Computational and Applied Mathematics*, 166, 2004.

[25] Kaj Madsen, Hans Bruun, and Ole Tingleff. Methods for non-linear least squares problems, 1999.

[26] Shayesteh E. Mahani, Xiaogang Gao, Soroosh Sorooshian, and Bisher Imam. Estimating cloud top height and spatial displacement from scan-synchronous goes images using simplified ir-based stereoscopic analysis. *Journal of Geophysical Research: Atmospheres*, 105(D12):15597–15608, 2000.

[27] C. J. Merchant, A. R. Harris, E. Maturi, and S. Maccallum. Probabilistic physically based cloud screening of satellite infrared imagery for operational sea surface temperature retrieval. *Quarterly Journal of the Royal Meteorological Society*, 131:2735–2755, 2005.

[28] Krishna Osuri, U. Mohanty, A. Routray, Makarand Kulkarni, and M. Mohapatra. Customization of WRF-ARW model with physical parameterization schemes for the simulation of tropical cyclones over North Indian Ocean. *Natural Hazards*, pages 1–23, June 2011.

[29] L. Pichler, A. Masud, and L.A. Bergman. Numerical solution of the Fokker-Planck equation by finite difference and finite element methods - a comparative study. In V. Plevris M. Papadrakakis, M. Fragiadakis, editor, *Computational Methods in Stochastics Dynamics*, volume 26, pages 25–28, Corfu, Greece, 7 2011. The organization, Springer.

[30] R. T. Pinker, I. Laszlo, J. D. Tarpley, and K. Mitchell. Geostationary satellite parameters for surface energy balance. *Advances in Space Research*, 30(11):2427–2432, November 2002.

[31] R. D. Richtmyer and K. W. Morton. *Difference Methods for Initial-Value Problems*. Wiley, New York, 2 edition, 1967.

[32] J.S. Seabold and J. Perktold. Statsmodels: Econometric and statistical modeling with python. In *Proceedings of the 9th Python in Science Conference*, 2010.

[33] A. S. Bin Mohd Shah, H. Yokohama, and N. Kakimoto. High-precision forecasting model of solar irradiance based on grid point value data analysis for an efficient photovoltaic system. *Sustanable Energy, IEEE Transactions on*, 6:1949–3029, 2014.

[34] W. C. Skamarock, J. B. Klemp, J. Dudhia, D. O. Gill, D. M. Barker, M. G. Duda, Huang X-Y, W. Wang, and J. G. Powers. *A Description of the Advanced Research WRF Version 3*. Mesoscale and Microscale Meteorology Division, National Center for Atmospheric Research, Boulder, Colorado, USA, 6 2008.

[35] W. C. Skamarock, J. B. Klemp, J. Dudhia, D. O. Gill, M. Barker, K. G. Duda, X. Y Huang, W. Wang, and J. G. Powers. A description of the Advanced Research WRF Version 3. Technical report, National Center for Atmospheric Research, 2008.

[36] Ninghu Su. Generalisation of various hydrological and environmental transport models using the FokkerPlanck equation. *Environmental Modeling Software*, 19, 2004.

[37] N. Veltishchev and V. Zhupanov. Experiments on numerical modeling of intense convection. *Russian Meteorology and Hydrology*, 33(9):560–569, 2008.

[38] N. K. Virdi. A review of backtesting methods for evaluating value-atrisk. *International Review of Business Research Papers*, 7, 2011.

[39] K. Wang. The Fokker-Planck Equation for power system stability probability density function evolution. *Power Systems, IEEE Transactions on*, 28, 2013.

[40] George Zittis, Panos Hadjinicolaou, and Jos Lelieveld. Comparison of WRF Model Physics Parameterizations over the MENA-CORDEX Domain. *American Journal of Climate Change*, 03(05):490–511, 2014.