STUDENT SUMMER INTERNSHIP TECHNICAL REPORT

Coding a Weather Model

DOE-FIU SCIENCE & TECHNOLOGY
WORKFORCE DEVELOPMENT PROGRAM

Date submitted:

October 13, 2017

Principal Investigators:

Andres Cremisini, DOE Fellow Student
Florida International University

Kristopher Klingler, Manager
Jon Bradley, Mentor
Sandia National Laboratories

Florida International University Program Director:

Leonel Lagos Ph.D., PMP®

Submitted to:

U.S. Department of Energy
Office of Environmental Management
Under Cooperative Agreement # DE-EM0000598

FIU | Applied Research Center
FLORIDA INTERNATIONAL UNIVERSITY

ABSTRACT

DOE Fellow, Andres Cremisini, completed a 10-week internship with Sandia National Laboratories (SNL) in Albuquerque, New Mexico. Under the management of Kristopher Klingler and the mentorship of Jon Bradley, he was tasked with conceiving and coding a realistic weather model for use in physical security applications. The objective was to make a weather model that could use real data to accurately predict wind and precipitation conditions at any location of interest on the globe at any user-determined time. The intern received guidance on software design, the C++ programming language and clear communication of project goals and ongoing progress. In addition, Mr. Cremisini was given license to structure the program however he best saw fit, an experience that will benefit ongoing research endeavors.

TABLE OF CONTENTS

LIST OF FIGURES

1. INTRODUCTION

The objective of this internship was to develop a fast and realistic wind and precipitation model and integrate the code into a larger physical security system using real data. The intern identified (National Centers for Environmental Prediction (NCEP) as a suitable data source, as it is a government agency and offers historical and present-day data at suitable grid fidelity (.25 degrees of latitude/longitude). The model required two primary features: the ability to report weather and location at any time and place, and the ability to alter ("tweak") data in order to simulate and test the results of certain weather conditions such as gusts or pockets of precipitation on the physical security system. In order to deliver the first feature, the intern identified a statistical method of spatial interpolation called "ordinary kriging," which is a widely used method in geographic information systems (GIS) and predicts values at un-sampled locations by building a model of the spatial covariance present in a data set (in this case, the NCEP data). Basic ordinary kriging does not have a temporal component, so the intern introduced a temporal variable in the kriging algorithm. The second feature was implemented probabilistically, allowing the user to choose any number of points in space or time and a range of values from which to choose a random scalar for any weather feature (wind speed or direction, amount of rain, etc.). The structure of the system, conceptualized and approved by Sandia mentors before the intern began coding, is shown in Figure 1.

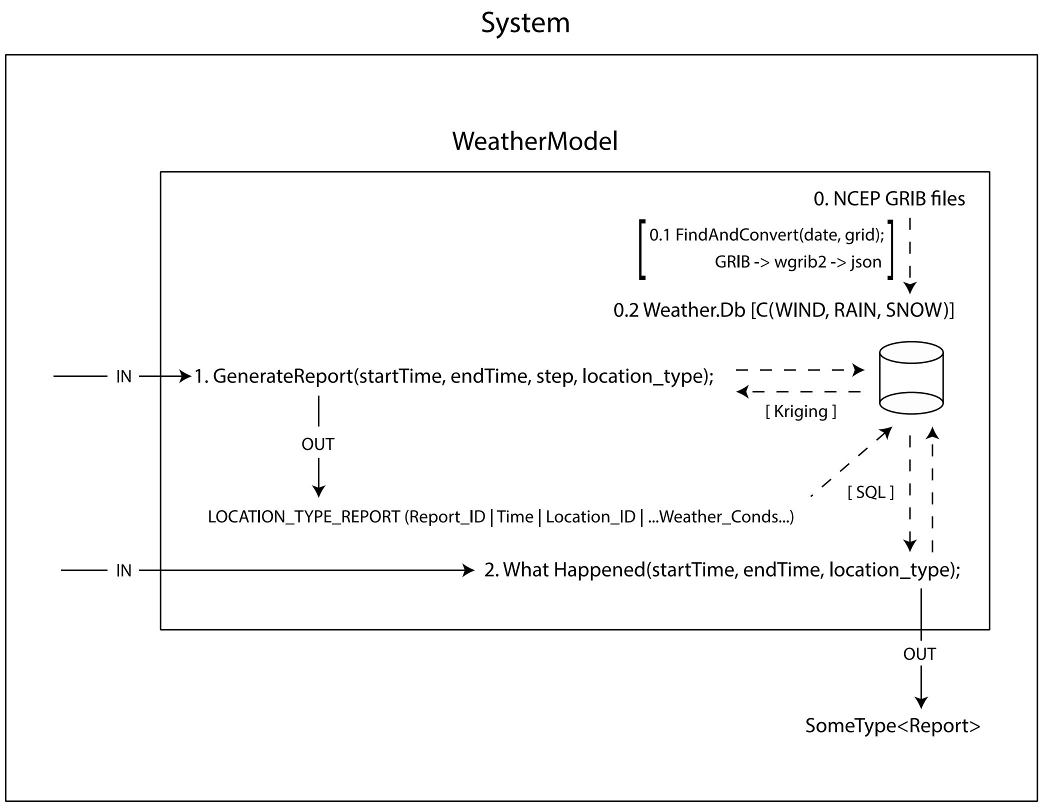
System
WeatherModel
0. NCEP GRIB files
0.1 FindAndConvert(date, grid);
GRIB -> wgrib2 -> json
0.2 Weather.Db [C(WIND, RAIN, SNOW)]
IN
1. GenerateReport(startTime, endTime, step, location_type);
[Kriging]
OUT
[SQL]
LOCATION_TYPE_REPORT (Report_ID | Time | Location_ID | ...Weather_Conds...)
IN
2. What Happened(startTime, endTime, location_type);
OUT
SomeType<Report>

Figure 1. Diagram of software architecture.

2. EXECUTIVE SUMMARY

This research work has been supported by the DOE-FIU Science & Technology Workforce Initiative, an innovative program developed by the US Department of Energy's Environmental Management (DOE-EM) and Florida International University's Applied Research Center (FIU-ARC). During the summer of 2017, a DOE Fellow intern Andres Cremisini spent 10 weeks doing a summer internship at Sandia National Laboratories under the supervision and guidance of Kristopher Klingler and Jon Bradley. The intern's project was initiated on June 5, 2017, and continued through August 11, 2017 with the objective of coding a time-based wind and precipitation model with the capability to predict weather conditions and any given point in time and space using the C++ programming language.

3. RESEARCH DESCRIPTION

Objectives

The goal of this research is to develop a realistic wind and precipitation model for use in physical security systems. This entails developing a method of spatiotemporal interpolation and designing an efficient, easy to understand software architecture to implement the model.

Descriptions

Kriging

Kriging is a method of spatial interpolation that models the values at a location v in a system Z as the sum of a local mean μ_l and a residual $R(v)$:

$$Z(v) = \mu_l + R(v)$$

Equation 1: Kriging Value

Where v is a location in a kriging system and $Z(v)$ is the value at v. All values in Z, known or unknown, are modeled in this fashion. μ_l is the simple arithmetic mean of all values in a given neighborhood $N(v)$, thus:

$$\mu_l = \frac{Z(N(v)_1) + \cdots + Z(N(v)_k)}{k}$$

Equation 2: Kriging Mean

With $Z(N(v)_i)$ the value of the *i*th neighbor of location v. The *k* neighbors may be chosen in any reasonable fashion. The model developed by the intern allows the user to specify any *k* (up to the total number of sample points), performing a *k-nearest neighbor* algorithm to find the *k* spatially nearest neighbors for a given v. In the present work, the primary concern in choosing k is efficiency, and with preliminary testing (formal testing was beyond the scope of the task), *k = 15* gave realistic results in an acceptable time.

Since $N(v)_i$ are always sampled locations and thus $Z(N(v)_i)$ is known, the only unknown in the kriging value model (Equation 1*)* is $R(v)$. Kriging models this value as a weighted sum of all $R(N(v)_i)$:

$$R(v) = \sum_{i=1}^{k} w_i \, R(N(v)_i)$$

Equation 3. Residual Value

The purpose of this vector w ($\in \mathbb{R}^{k \times 1}$) is to minimize the mean squared prediction error at un-sampled locations. This is called the best linear unbiased predictor (BLUP), and it is found by adding the constraint:

$$E\left[\hat{Z}(v)\right] = E[Z(v)] = \mu_l$$

Equation 4: Unbiasedness Constraint

This means that the expected ("kriged") value at an unknown location v should be the value at v if it were sampled, which should equal the local mean. After employing a series of somewhat lengthy statistical derivations (which are not included here for brevity), this generates the following constraints (Equations 5, 8):

$$\sum_{j=1}^{k} w_j \, Cov\left[Z(v_i), Z\left(v_j\right)\right] + \lambda = Cov[Z(v_i), Z(v_0)] \; \forall i \in \{1, \dots, N\}$$

Equation 5: Covariance Constraint [1]

Where *k* is the number of neighbors at v_0, v_0 is an un-sampled location and v_i, v_j are all in $N(v_0)$. λ is the Lagrange multiplier given by $L(w_i, \lambda)$, and *Cov[…]* is given by the chosen model of the semivariogram of the system. The semivariogram γ is a plot of the average mean squared difference of the values at locations at all distances *h* in the sample data:

$$\gamma\left(h_j\right) = \frac{1}{2M\left(h_j\right)} \sum_{i=1}^{M\left(h_j\right)} [Z(v)_i - Z(u)_i]^2 \;\; \forall j$$

Equation 6: Semivariogram Function [1]

Where *location(v)* != *location(u)*, $M\left(h_j\right)$ is the number of sampled locations at distance h_j, and $\{(Z(v), Z(u))_1, \dots, \left(Z(v), Z(u)\right)_{M(h)}\}$ is the set of all sampled locations v, u at distance h_j. $\{h_1, \dots, h_n\}$ is the set of all the distances present in the sample data. Figure 2 shows the semivariogram plot and corresponding model fit for the wind direction given by NCEP on July 14, 2017, with all h_j and $\gamma\left(h_j\right)$ normalized. The semivariogram is fit using a standard Gaussian function.

In order to perform kriging for unknown locations v_0 at a possible distance $h_0 \notin \{h_1, \dots, h_n\}$, γ is modeled with a covariance function (continuous, positive definite), giving:

$$Cov\left[Z(v_i), Z\left(v_j\right)\right] = max[\gamma(h)] - \gamma\left[dist\left(Z(v_i), Z\left(v_j\right)\right)\right]$$

Equation 7: Covariance Function

Where $max[\gamma(h)]$ is the largest mean squared difference (this is commonly called the *sill*).

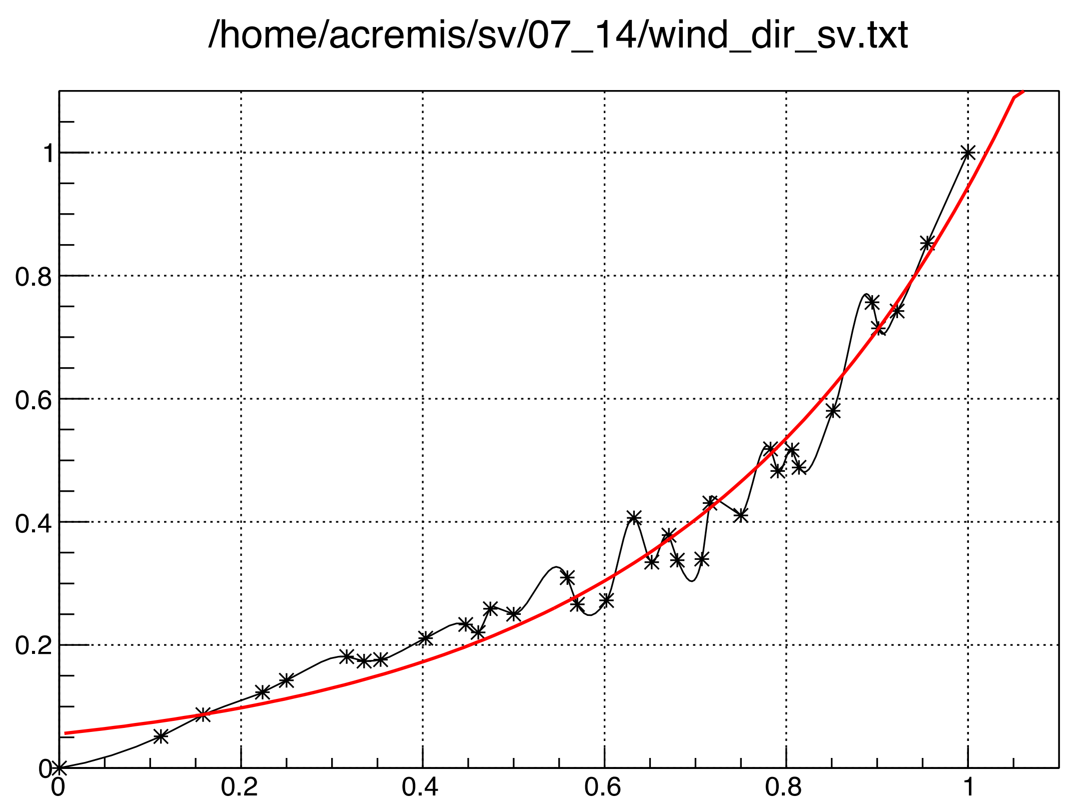

Figure 2. Semivariogram plot and fit.

The second constraint is:

$$\sum_{i=1}^{k} w_i = 1$$

Equation 8: Weight Constraint [1]

Both of these constraints (Equations 5, 8) can be satisfied by solving the Kriging equation:

$$Cw = c$$

Equation 9: Kriging Equation

Where C ($\in \mathbb{R}^{k+1\ x\ k+1}$) is the covariance matrix given by $C_{i,j} = Cov\left[Z(N(v_i)), Z\left(N(v_j)\right)\right]\ \forall i,j\ \in \{1, \dots, k\}$, $C_{k+1} = [1 \quad \dots \quad 1 \qquad 0]$, w is the weight vector in Figure 2, and c ($\in \mathbb{R}^{k+1\ x\ 1}$) is the covariance vector given by $c_i = Cov[Z(N(v_i)), Z(v_0)]\ \forall i\ \in \{1, \dots, k\}$, $c_{k+1} = \lambda$. Thus:

$$w = C^{-1}c$$

Equation 10: Kriging Solution

In practice, however, the matrix C is not inverted due to the computational cost of matrix inversion. Instead, matrix decomposition is used to solve the equation. Therefore, Equation 3 (reprinted below for the reader's convenience):

$$R(v) = \sum_{i=1}^{k} w_i \, R(N(v)_i)$$

is solved, and the value $Z(v_0)$ at any un-sampled location v_0 in the general neighborhood of the sample data can be estimated by Equation 1 (reprinted below):

$$Z(v) = \mu_l + R(v)$$

There are several versions of kriging, of which the most common are *simple, ordinary,* and *kriging with trend.* The procedure outlined in this section and used by the intern is known as *ordinary kriging.* While describing all of these procedures in detail is outside the scope of this document, the reader is referred to the references [1, 2, 3, and 4] for a more detailed discussion.

Software Architecture

The reader is referred to Figure 1 for a visual representation of the software architecture and data flow:

1. NCEP Data (Acquisition and Conversion)
   The first step in building the weather model entails requesting a sample grid size and a day from the user then using this information to send a http request to NCEP and downloading the corresponding data. NCEP delivers this data in the GRIB2 format, which is decoded using the wgrib2 command line application (open source, courtesy of NCEP) [5]. The GRIB2 data is converted to .csv format and stored on the disk. Wgrib2 was integrated into the C++ application and is called automatically from the same.

2. Semivariogram Calculation
   Every weather attribute downloaded from NCEP requires a separate semivariogram calculation. Each semivariogram is encoded as a simple map data structure and given the name of the weather attribute (wind direction, wind speed, etc.). The user is given the option to store this data in a .csv file and/or plot the values as displayed in Figure 2.

3. Semivariogram Fitting
   Each semivariogram data structure is fitted with a covariance function. The model uses the ROOT C++ data analysis library [6] to fit each semivariogram (open source, courtesy of CERN), choosing the function (either exponential or Gaussian) with the least chi-squared error. Each function is stored in a corresponding function object type and tied to its corresponding weather attribute.

4. Kriging
   Given an unsampled location, time interval and interpolation frequency, ordinary kriging is used to estimate values for every weather attribute in the system. Since NCEP reports hourly sample data and the user might wish to interpolate at a particular second that does not coincide with an NCEP sample time, the following algorithm was implemented (presented here in pseudocode):

```
If requested time t is un-sampled:
    For every weather attribute:
        For every $N(v)_i$:
            Find the nearest time in the past, $t_0$, and the nearest in the future, $t_1$
            Find the value at $v_0$ at $t_0$ and $v_1$ at $t_1$
            Make a linear function f using $v_0$ and $v_1$
            Assign $Z[N(v)_i]$ to f(t)
        End for
    End for
Perform Kriging normally
```

5. Weather Tweaker
   The intern was tasked with implementing weather “knobs,” which allows the user to artificially introduce certain weather conditions such as gusts. The user is asked for a tweaking grid, tweaking time, and a value range from which to randomly scale the values at all affected locations. This effectively allows the user to “make the weather,” using this feature to test the effect of a north-westerly gust on some section of the physical security system, for example.

6. Location Status Reports
   The intern was tasked with generating reports for locations pertinent to the physical security system (alarm locations, for example) as well as locations in the neighborhood of the security system but that do not form part of it. The system asks the user for a list of alarm locations in .json format, and if during the interpolation stage a requested un-sampled location coincides with one of the alarm locations, an “Alarm Report” is generated which probabilistically determines an alarm state (triggered/un-triggered) given known behaviors of alarm types under certain weather conditions. If an un-sampled location is not an alarm location, a “Location Report” is generated with only the interpolated values for all weather attributes.

7. Database Integration
   At the beginning of the interpolation process, all of the data generated in step 1 is stored in a database. In addition, the user is given the option to store all the interpolated data in a database as well as with the option to query the database from the C++ runtime. This feature can be used for offline data analysis given the Alarm Reports generated during runtime.

4. RESULTS AND ANALYSIS

Although model testing for prediction accuracy was outside the scope of the intern's assigned task, the intern's managers and mentors have determined that the interpolated values generated by the model are realistic, and that the features coded in the model meet the specifications given to the intern at the beginning of the summer.

5. CONCLUSION

2D ordinary kriging was determined to be a good method for modeling weather data, sufficient for the task of generating believable conditions in the model of a physical security system.

6. REFERENCES

1. Franklin, Meredith (2014). Solution to Ordinary and Universal Kriging Equations.

2. Oliver, M.A. (1990). Kriring: a method of interpolation for geographical information systems. *Geographical Information Systems,* Vol. 4, No.3, 313-332.

3. How Kriging Works. Available at: http://pro.arcgis.com/en/pro-app/tool-reference/spatial-analyst/how-kriging-works.htm. (Accessed: 19th August 2017)

4. KRIGING: Available at: http://people.ku.edu/~gbohling/cpe940/Kriging.pdf. (Accessed: 19th August 2017)

5. Climate Prediction Center - wgrib2: grib2 utility. Available at: http://www.cpc.ncep.noaa.gov. (Accessed: 12th August 2017)

6. ROOT: A Data Analysis Framework. Available at: http://www. https://root.cern.ch. (Accessed: 12th August 2017)