A Reliable Web-based System for Hurricane Analysis and Simulation*

Shu-Ching Chen[1], Shahid Hamid[2], Sneh Gulati[3], Na Zhao[1], Min Chen[1], Chengcui Zhang[4], Paresh Gupta[1]
[1]School of Computer Science, Florida International University, Miami, FL 33199, USA
[2]Department of Finance, Florida International University, Miami, FL 33199, USA
[3]Department of Statistics, Florida International University, Miami, FL 33199, USA
[4]Department of Computer and Information Sciences, University of Alabama at Birmingham, Birmingham, AL 35294, USA

Abstract – *In this paper, we present our research and development efforts toward a reliable, large-scale, web-based system for hurricane analysis and simulation. The major contributions of this system lie in the following three aspects: 1) It supports the multidisciplinary research efforts in predicting and simulating the hurricane damages and insured losses; 2) It is a large-scale, distributed, web-based system, which allows both the professional and general users to conduct computational intensive on-line analysis simultaneously and to share and exchange the information; and 3) Database management techniques are employed in our system to manage the huge amount of hurricane-related historical and simulated data.*

Keywords: System modeling and integration, information fusion, data management.

1 Introduction

Hurricanes pose one of the greatest environmental threats to the human beings. It is in great needs of researches and tools, which can forecast and analyze hurricane phenomenon, support decision-making processes and prevent property damage and loss of life. The purpose of this paper is to present the interdisciplinary research efforts and system integration toward an interactive web application, which is intended as a reliable, scalable, and sharable system for hurricane analysis and simulation.

In the past few years, quite a few research efforts have been put into the area of forecasting and monitoring hurricanes and the associated hazard, which resulted in a number of techniques and systems developed to tackle this problem. Among those efforts, the Advanced Regional Prediction System (ARPS) [2] at University of Oklahoma, the Colorado State University's Regional Atmospheric Modeling System (RAMS) [9], and the HAZUS system [3] from the Federal Emergency Management Agent (FEMA) are the most significant work in the literature.

However, in most of these existing systems, neither database management nor warehouse techniques were used. Moreover, they are mostly standalone software applications which are difficult for the users to share and exchange the information, and thus increase the expenses to install, maintain, and upgrade the necessary software and hardware. In view of these issues, we aim to develop a web-based hurricane analysis and simulation system, which utilizes the database management techniques to handle the huge amount of hurricane data and has the capability to support both professional and general users in a very convenient and efficient way. In particular, in this study, a probabilistic model called Public Hurricane Risk and Insured Loss Projection Model (PHRLM) is developed to estimate the damages and insured losses due to the occurrence of hurricanes in Atlantic Basin, which provides the capability of predicting and estimating the full probabilistic distribution of the damage and loss for any significant storm event. The modeling methodology includes the following four major modules: 1) Storm Forecast Module; 2) Wind Field Module; 3) Damage Estimation Module; and 4) Loss Estimation Module.

In our previous work [1], the Storm Forecast Module has been studied and implemented, which consists of two major components, namely Annual Hurricane Occurrence Estimation (AHO), and Storm Genesis Time Estimation (SGT). In this paper, we further extend our work by investigating and developing the Wind Field Module, which is composed of Storm Track Generation, Wind Model Calculation and Wind Speed Correction. Furthermore, we refine the Storm Forecast Module by introducing the definition of the threat area, which will be discussed in Section 3.1. The system integration between the Storm Forecast Module and the Wind Field Module is discussed as well.

The rest of the paper is organized as follows. Section 2 introduces the system architecture for the PHRLM project. In Section 3, the system components (represented by use cases) are discussed in details and the streamlined processes for the hurricane simulation are described as well. In Section 4, the system performance is evaluated. Section 5 concludes this paper.

2 System architecture

In order to provide a flexible yet robust infrastructure to construct, maintain, and extend the hurricane simulation system, we utilize the three-tier architecture which naturally partitions the system components into different layers with respect to their functionalities. Therefore, different functionalities can be developed in a relatively independent manner. On the other hand, the interfaces between the different layers allow the seamless collaboration among them to perform the required tasks. Figure 1 gives an overview of the system architecture, which consists of three tiers, namely *database tier*, *application logic tier*, and *web presentation tier*.

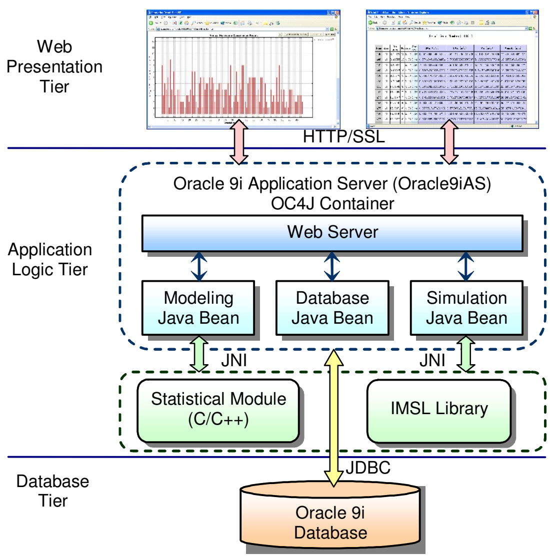

Figure 1. System architecture

- *The database tier* handles hurricane data management issues, such as data modeling, storage, access, and manipulation. This tier is indispensable to our system due to the large amount of historical data required for the hurricane analysis. In the meanwhile, a huge volume of simulated data will be produced during the hurricane simulation and projection process. Therefore, an Oracle9i database is deployed in our system to manage both the historical hurricane data and the generated hurricane simulation data. A database server is deployed at this tier.
- *The application logic tier* processes the user requests and generates the corresponding responses, hiding the technical details from the users' awareness. It also acts as the bridge between the database tier and the presentation tier. An Oracle9i Application Server is installed and operates in Linux environment to guarantee the running-time support. JavaBean technology is used to handle the data/control flow between the users and the underlying application logic and database tiers. The communication between the JavaBeans and the database are handled by JDBC [11]. In addition, for the sake of system performance, C/C++ language is used to perform the computation-intensive tasks and is integrated into the Java code through Java Native Interface (JNI). Meanwhile, IMSL [5] library CNL 5.0 is used since it provides a comprehensive set of mathematical and statistical functions which can be embedded into our applications.
- *The web presentation tier* manages various input/output data and their display, while the underlying application logic and back-end processing are hidden from the end users. The client communicates with the web server via HTTP and SSL protocol. JSP [6] web-scripting language is employed to generate the web page contents dynamically.

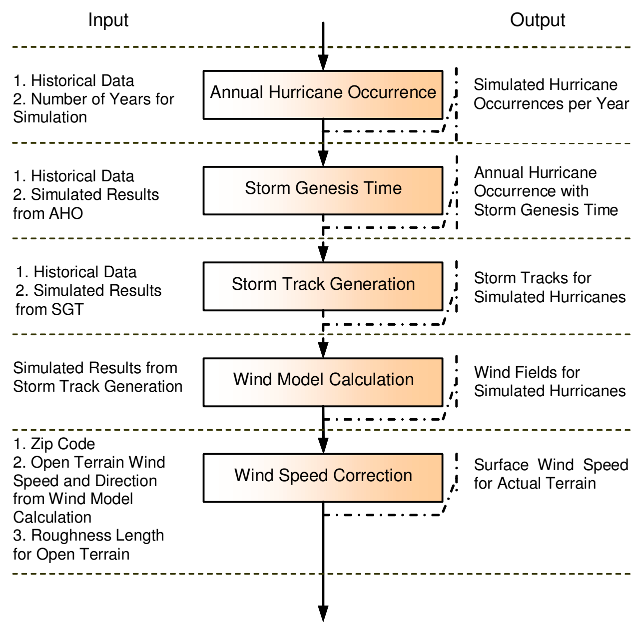

Figure 2. System use cases

3 Use cases

Our current system consists of five streamlined processes (use cases) as illustrated in Figure 2 to fulfill the hurricane simulation tasks based on a set of historical hurricane data stored in the database, which are imported from the North Atlantic "best track" HURDAT [4]. Currently, the historical data set contains totally 1,274 hurricanes occurred from 1851 to 2001. Each hurricane consists of a group of records, called storm fixes, which specify the locations and strength of that specific storm at every 6-hour interval.

3.1 Annual hurricane occurrence (AHO)

The goal of the AHO component is to estimate the number of hurricanes that occurs per year based on a hurricane occurrence probability distribution.

One meteorological fact is that the statistical properties of hurricanes vary with different year ranges, which in turn may considerably affect the probability distribution estimation and the final simulation results. Therefore, the historical hurricane data in our proposed system are categorized into five pre-defined data sets according to the climate cycles or qualifications of available data records, including *1851-2000*, *1900-2000*, *1944-2000*, *Multi-Decadal*, and *ENSO*. Here, "*Multi-Decadal*" set contains all warm (active) years and cold (quiet) years; and "*ENSO*" set stands for the EL Nino [7] and La Nina [8] years.

Based on the selection of the year range, a number of statistical models are applied against the target data set retrieved from the database. The statistical distribution that best characterizes the statistical property of that data set according to the domain knowledge in meteorology, is selected to perform the subsequent simulation. Table 1 shows some example outputs of this use case.

Table 1. Matrix of number of simulated storms per year

Year	Number of Tropical Cyclones
1	4
Y_i	H_i

In our previous work, we considered the hurricanes in the entire Atlantic Ocean Basin. In this work, we improve the system performance by focusing on the hurricanes fall in the threat area. Here, the threat area is defined as a radius of 900 km centered at 29.0 North (Latitude) and 83.0 East (Longitude), which is actually the region of the interest. The threat area surrounding Florida is shown in Figure 3. In other words, a hurricane will be considered only if it ever passed the threat area (for instance, hurricane Andrew 1992 [12] as shown in Figure 3).

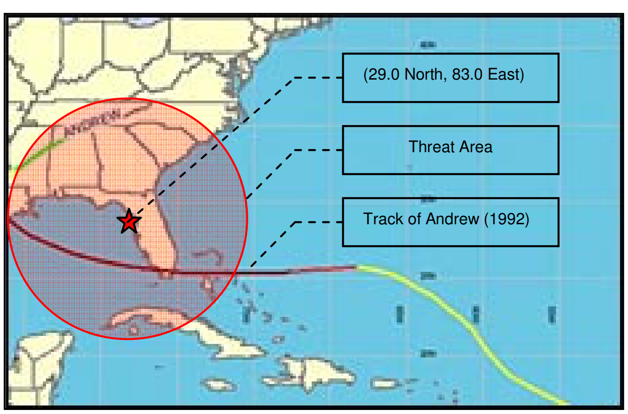

Figure 3. Threat area and the storm track of hurricane Andrew 1992

As mentioned earlier, there are totally 1274 historical hurricanes in the database. After the filtering process using the threat area definition, 598 of them are considered as the valid historical data. Figure 4 shows an example of the simulated results.

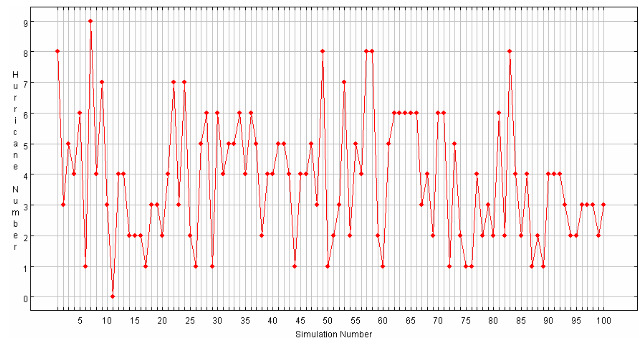

Figure 4. Simulated 100-year annual hurricane occurrence distribution

3.2 Storm genesis time (SGT)

SGT, stands for Storm Genesis Time, is the second use case of the PHRLM project. The major purpose of this component is to construct a probability distribution for storm genesis time and generate the genesis time for a series of simulated storms produced in the AHO component.

Storm genesis time is defined as the hour difference between the first fix data of the storm and the beginning of the hurricane season. The occurrence pattern of the intervals of the hurricane genesis time is identified and then used to predict the time intervals of simulated storms. The computation steps can be described as follows:

First, based on the user selected data set, the system queries the database and returns a set of storm fix data for the corresponding tropical cyclones. The fix data set consists of all the hurricane related information, such as: storm ID, storm name, genesis date, Julian date, genesis fix time, latitude, longitude, maximum wind speed, and central pressure.

Then, the system uses the retrieved fix data to calculate the number of hours between the genesis (HBG) of the storms (in 6 hour resolution). This information is further used to estimate the CDF (Cumulative Distribution Function) of HBG and to generate the genesis time of the simulated hurricanes based on the estimated CDF.

Table 2. Simulated storms genesis time

Year	Tropical Cyclones No.	Storm Genesis Time
1	1	2932
Y_i	H_i	T_i

IMSL libraries are utilized to calculate the storm genesis time of the simulated hurricanes based on the fitted distribution of HBG. Then, the system displays the generated genesis time and the corresponding simulated events to the user through a web interface. Table 2 shows an example of the simulated results of the SGT component.

Here a storm season starts at 0:00 Hours on May 01 and ends on April 30 of the following year. Each day the storm data is collected in one of the following intervals: I_1=[0:00, 6AM), I_2=[6AM, 12 Noon), I_3=[12Noon, 6PM), I_4=[6PM, midnight). For the sake of simplicity, each interval is represented by its starting point. For example, the storm with Storm ID 310 occurred in the interval I_3 on July 5, 1851, the SGT value for this storm is 24×(2397309 (Julian date of 5-Jul-1851) - 2397243 (Julian date of 1-May-1851)) + 12 (Starting point of I_3) = 1596. Therefore, based on the simulated storm genesis time, we can calculate the simulated starting date and time of the projected hurricanes.

3.3 Storm track generation

The Storm Track Generation component takes the output of SGT component as the input and generates the simulated storm tracks up to close of land for simulated storms. For each year range that the user requests, the system queries the database and returns the simulated storm data (*Year, Tropical Cyclones No., Storm Genesis Time*) generated from SGT. There are two major modules that comprise the storm track model. The first module computes the key hurricane statistics and generates the PDFs (Probability Distribution Function) based on a historical data set. The second module generates the stochastic storm tracks using the PDFs generated by the first module.

The steps for generating the simulated storm tracks up to close of land are listed as follows:

1. Calculating the latitude and longitude values in 100th degree.
2. Calculating the storm translation velocity for north, south, east, and west (meter/second, knots, miles/hour).
3. At close land, the system uses sampling from the Intensify Scaling Parameter to generate Rmax (Radius of the maximum winds).
4. Generating the fine scale tracks (latitude, longitude, UTC time, central pressure [in 10th of milibars], Rmax [in 10th of km], sea flags), etc.
5. Storing the data produced during previous steps in the database.

Table 3. The output of storm track generation

Number of records: 19								
Storm No.: storm00001 Date: 6/18/ Year: 1 Time: 04:00								
Category	Date	Utc(hr)	Latitude	Longitude	Central Pressure	Rmax	Holland B	Sea Flag
1	6/18	04:00	24.09	-83.00	994.32	47.46	1.37	1
1	6/18	05:00	24.18	-82.89	994.08	47.75	1.37	1
…	…	…	…	…	…	…	…	…

The storm track generation component is implemented using Fortran for the sake of efficiency. The example output of this component for the simulated storm No. 1 is listed in Table 3. At the current stage, we have generated 1,000 years of simulated storms, and the output will be passed as the input data onto the wind model calculation component.

3.4 Wind model calculation

After the simulated storm fixes are produced by the storm track generation component, wind model is calculated to produce the wind speed time series for each of the zip codes in Florida that would be affected by the hurricanes.

The wind field model code is implemented using The Interactive Data Language (IDL) [10]. Based on the input parameters, namely the storm category, latitude-longitude, radius of maximum winds, the central pressure and the Holland B parameter, radial profiles of the radial and tangential winds are calculated based on a stationary cyclone over open water to provide an "envelope" with which to set the size of the cyclone vortex. These radial and tangential winds produce the wind fields. The model takes the corresponding zip code information as the input. The resulting time series output for each of the affected zip code includes the landfall date (month/day/year) and time (hour: minute), zonal wind speeds (*zwind*, m/s), meridional wind speed (*mwind*, m/s), total wind speeds (*Wind*, m/s), and the wind direction in degrees (*dir*, deg) at discrete time intervals for the entire storm duration. Table 4 demonstrates the example output of the wind field model for zip code 32940 and the storm No. 918 in 1,000 years of simulated storms.

Table 4. Wind field model output format

Storm No.: storm00918 Date: 9/24/ Year: 226 Time: 13:00							
Zip code: 32940 longitude: -80.6925 latitude: 28.2205							
index	day	utc(hr)	zwind(m/s)	Mwind(m/s)	Wind(m/s)	dir(deg)
0	1	19.000	3.63371	-7.19740	8.06266	333
1	1	19.250	3.90878	-7.24202	8.22955	331
…	…	…	…	…	…	…

The results of the wind model calculations are further passed onto the wind speed correction component to take into account the actual terrain land cover data.

3.5 Wind speed correction

This component aims at refining open terrain wind speed produced by the wind model calculation component with respect to the actual terrain data (based on land use – land cover).

It is well known that wind speed is affected by the friction against the earth's surface in the lower levels of the atmosphere. The rougher the ground surface, the more the wind speed is slowed down. Generally speaking, large cities and forests will have much more influence on the wind speed as grass and shrubs will slow down the wind speed considerably, while the smooth water surface has little effect on wind speed. In this component, the term "Roughness Length" (*Zo*) is utilized to quantitatively describe the roughness characteristics of the ground obstacles. More specifically, the roughness length is defined as the distance above the ground level where the wind speed should theoretically be zero. A lower roughness length implies less exchange between the surface and the atmosphere. In our proposed model, the roughness length for the open terrain is set as a constant – 0.03m, while the actual roughness length is determined by the zip codes and wind directions. In brief, this component takes the following data as inputs:

- Zip code;
- Surface wind speed for open terrain produced by wind model calculation (m/s);
- Surface wind direction (degree from north) produced by wind model calculation; and
- Roughness length (m) for open terrain = 0.03m.

The system queries the database based on these inputs and returns a *Zoa* parameter, which corresponds to the actual roughness length based on FEMA HAZUS conversion table relating land use-land cover (LULC) to aerodynamic roughness (m). Then roughness is obtained as a weighted average of all roughness pixels within a 45 degree sector with the origin at the population-weighted centroid of the zip code and extending outward to 20 km from the centroid. The weighting function for averaging the roughness values is a Gaussian filter with a half power point at 3 km. The format of the lookup table is as follows:

Table 5. Roughness length record format

Zip	*Lon*	*Lat*	*Zo1*	*Zo2*	*Zo3*	*Zo4*	*Zo5*	*Zo6*	*Zo7*	*Zo8*

Here, *Zo1~Zo8* represent the actual roughness length for wind directions inclusive of (46º-90º), (1º-45º), (316º-0º & 360º), (271º-315º), (226º-270º), (181º-225º), (136º-180º) and (91º-135º), respectively.

Based on the inputs and the retrieved roughness value, the system computes and generates the appropriate value for actual terrain roughness. The outputs of this use case include:

- Open terrain friction velocity (m/s);
- Actual terrain friction velocity (m/s); and
- Surface wind speed for actual terrain (m/s).

4 System performance

The proposed system is able to provide efficient and reliable online hurricane occurrence simulation services by handling a high request rate and intense user accesses. By using the system response time as the performance measure, we have conducted sufficient experiments to evaluate the performance of this hurricane simulation system.

For the web based use cases, such as AHO, SGT, and Wind Model Calculation, we have designed the test cases to simulate a multi-user access scenario and to test the system performance of the system response time versus the number of users. In this test, the number of users is increased by five each time until 100. A set of test cases is generated, for example, the 1-user case, the 5-user case, and so on. All test cases are repeated 5 times, and the response time is averaged as the final result. We have tested all the cases under normal network traffic conditions using different parameter combinations, such as different historical hurricane datasets and different number of simulation years. The results are presented graphically by plotting the average system response time versus the number of users. The unit of system response time is millisecond. We have conducted 1,000 years of hurricane occurrence simulation based on all the historical records of 1274 hurricanes from the year 1851 to the year 2000. Figure 5 depicts the performance analysis for the AHO use case.

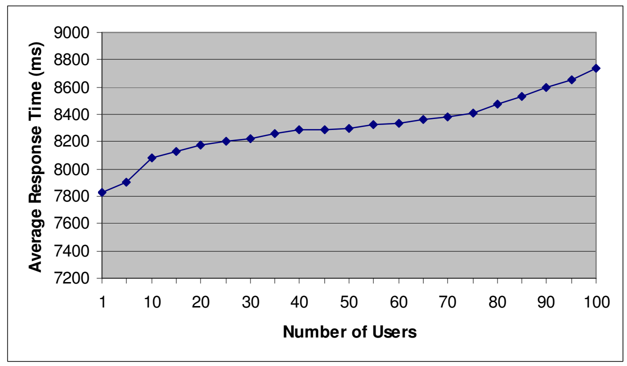

Figure 5. Performance analysis for AHO

As shown in Figure 5, as the number of users increases considerably, the average system response time does not increase significantly, which demonstrates the system scalability.

5 Conclusions

In this paper, we exemplified our efforts in modeling, designing, and implementing a large-scale, distributed, web-based hurricane analysis and simulation system. The well-established three-tier client-server architecture is adopted for the purpose of robustness, reusability, and scalability. We believe the multidisciplinary research efforts and the large-scale web-based system integration experience will benefit the future research and system development in related areas.

6 Acknowledgement

We gratefully acknowledge the on-going collaborative efforts with Hurricane Research Division at NOAA (Dr. Mark Powell and Dr. Soukup George), Cooperative Institute for Marine and Atmospheric Science, University of Miami (Nirva Morisseau-Leroy), and Florida State University (Dr. Steve Cocke) to provide the details of the meteorology components. This work is partially supported by Florida Department of Insurance (DOI) under

"Hurricane Risk and Insured Loss Projection Model" project. While the project is funded by Florida DOI, DOI is not responsible for this paper content.

References

[1] S.-C. Chen, et al., "A web-based distributed system for hurricane occurrence projection," *Software: Practice and Experience*, Vol. 34, No. 6, pp. 549-571, May 2003.

[2] M. Xue, K. K. Droegemeier, and V. Wong, "The advanced regional prediction system (ARPS) - a multiscale nonhydrostatic atmospheric simulation and prediction model," *Meteor. and Atmos. Physics*, Vol. 75, pp. 161-193, 2000.

[3] HAZUS home. http://www.fema.gov/hazus

[4] HURDAT data. http://www.aoml.noaa.gov/hrd/hurdat/Data_Storm.html

[5] IMSL Mathematical & Statistical Libraries. http://www.vni.com/products/imsl

[6] Javaserver Pages^{TM} Technology. http://java.sun.com/products/jsp

[7] NOAA EL Nino Page. http://www.elnino.noaa.gov/

[8] NOAA LA Nina Page. http://www.elnino.noaa.gov/lanina.html

[9] RAMS: Regional Atmospheric Modeling System, http://rams.atmos.colostate.edu/

[10] The Interactive Data Language. http://www.rsinc.com/idl/

[11] The JDBC API Universal Data Access for The Enterprise. http://java.sun.com/products/jdbc/overview.html

[12] Track of hurricane Andrew (1992) http://www.pbs.org/newshour/science/hurricane/facts.html (Source from NOVA)