2011

Parallel surrogate detection in large-scale simulations

Lei Jiang
Louisiana State University and Agricultural and Mechanical College, lionelchange@gmail.com

PARALLEL SURROGATE DETECTION IN LARGE-SCALE SIMULATIONS

A Thesis

Submitted to the Graduate Faculty of the
Louisiana State University and
Agricultural and Mechanical College
in partial fulfillment of the
requirements for the degree of
Master of Science in Systems Science

in

The Department of Computer Science

by
Lei Jiang
B.E., University of Science and Technology of China, 2005
M.E., Wuhan University, China, 2007
May 2011

Acknowledgments

I would like to thank my supervisor Dr. Gabrielle Allen. The research work in the thesis is enabled through her funded projects. She envisioned the multidisciplinary research with a high level of application significance. She also has been helpful with my study and life here at Louisiana State University (LSU) during the three years since my arrival. I am also thankful to the other two members in my graduate advisory committee: Dr. Q. Jim Chen in Department of Civil and Environmental Engineering and Dr. Jian Zhang in Department of Computer Science: they provided lots of helpful comments and suggestions in the course of thesis writing.

Thanks to the fellow students and other nice professors I met at LSU. The atmosphere has made me become more motivated in investigations and discussions with regard to academic research. The help from LSU technical staff for high-performance computing resources is appreciated as well.

Lastly, thanks to my parents for their infinite love and support.

Table of Contents

List of Figures

List of Tables

Abstract

Simulation has become a useful approach in scientific computing and engineering for its ability to model real natural or human systems. In particular, for complex systems such as hurricanes, wildfire disasters, and real-time road traffic, simulation methods are able to provide researchers, engineers and decision makers predicted values in order to help them to take appropriate actions. For large-scale problems, the simulations usually take a lot of time on supercomputers, thus making real-time predictions more difficult. Approximation models that mimic the behavior of simulation models but are computationally cheaper, namely "surrogate models", are desired in such scenarios. In the thesis, a framework for scalable surrogate detection in large-scale simulations is presented with the basic idea of "using functions to represent functions". The following issues are discussed in the thesis: **i)** the data mining approaches to detecting and optimizing the surrogate models; **ii)** the scalable and automated workflow of constructing surrogate models from large-scale simulations; and **iii)** the system design and implementation with the application of storm surge simulations in the occurrence of hurricanes.

Chapter 1

Introduction

The increase in capacity of computational resources, data storage, integrated experimental and observational devices and connecting networks available to scientists and engineers is enabling new paradigms and methodologies for scientific discovery. Simulations, especially for those of complex systems, are greatly enabled on large computers. With many researchers now having easy access to supercomputers, domain scientists, such as physicists, biologists and engineers, are able to develop and run simulations that model the natural processes in various areas in a distributed and collaborative environment. The capability of mimicking system behavior can help decision makers to predict the future situation. In the scenario of large-scale simulations which are targeted in a spatial domain rather than a single point, one or many simulations are performed to find out the predicted values at points of interest for forecasting purposes. Such applications in scientific computing and engineering, including storm surge forecasting in hurricanes [1, 2], the prediction of methane inflow rates during mining [3], road traffic prediction [4], forecasting capabilities related to ecological and biological systems [5, 6], provide good case studies as complex system modeling based on simulations.

However, the increase in capacity of computational resources does not lead directly to

a rapid improvement of the simulations themselves. Instead, it brings a new challenge that motivates scientists to fully utilize the huge amount of simulation data created in supercomputers, thus fostering advanced scientific research. Driven by the urgent need for in-depth investigations in Louisiana coastal areas, especially during hurricane seasons, a data center, which provides research communities with scientific data resources on demand, is imperative.

1.1 Motivation

Machine learning and data mining techniques have been widely used in various fields since the 1990s. The functions exist to exploit hidden information and latent relationships from data and verifying them. While common data mining tasks, such as classification, regression and clustering, can identify the value of target variable with multiple attributes known, a simulation can also regarded as one or many functions mapping from a set of input parameters to the target variable in the area of interest. So it enables researchers to apply data mining methods in constructing surrogate models. In particular, as a typical simulation outputs time series on every node in the computational grid. Combining the parameter space of simulation input with its output and running multiple such simulations, there exist the following patterns: *i)* the relationship between the input and the simulated response as an objective function at a single point; *ii)* the spatio-temporal correlation between points of interest and *iii)* the clustering of points of interest based on the level of response. Along with describing the data mining models for detecting surrogate models from simulation data, we also present a scalable and automated workflow in this paper that facilitates the high-

performance data mining.

1.2 An Application Scenario: Storm Surge Prediction in Hurricanes

In this thesis, the author mainly focuses on the application of storm surge prediction in hurricanes. Storm surge prediction is essential is estimating the effect brought by the hurricane. The Louisiana Coastal Area presents an array of rich and urgent scientific problems that require new computational approaches: tropical cyclones, especially Hurricane Katrina in 2005 and Hurricane Gustav in 2008 have caused severe loss of life and property damage. Now dynamic storm surge prediction, mainly based on physics-based simulations ,is being operationally used as accurate and timely predictions are essential for decision makers to deploy appropriate evacuation plans. However, the dynamic and multi-physics nature of this problem, as well as the expensiveness of computational resource consumption, present new challenges for related research in coastal science and engineering.

Running a storm surge simulation, as typically using ADCIRC (Coastal Circulation and Storm Surge model) model [7], can take more than 2,200 CPU hours for a hurricane of 5 days. This means that even with 64 computing nodes in a cluster, more than 30 hours is needed for completing the simulation. While it is large-scale complex problem, the surrogate models in the thesis are based on extracted time series from 15 locations, corresponding to the tide gages in Louisiana, for each simulation. On the other hand, while a hurricane track can be a trajectory with an arbitrary order across the Gulf of Mexico, we use the

hypothetical tracks in the experiments to evaluate the model response to parameter space.

The thesis is organized as follows: in Chapter 2, the surrogate modeling approaches are presented along with the criteria used for model validation. Then in Chapter 3, the automated and scalable workflow of the data-oriented modeling is discussed in a view of framework development. The implementation and results are illustrated in Chapter 4, and finally the thesis is concluded in Chapter 5 along with the recommendations for future work.

Chapter 2

Data-Oriented Methods for Detecting Surrogate Models

2.1 Background and Problem Setting

Mining simulation data is receiving more and more attention in recent years. Besides the simulation-based prediction, optimal design is also an objective as it is common in many engineering applications [8]. As large computing resources become more accessible for researchers over years, the work in building data warehouses and mining large-scale simulation data spreads from area to area in order to detect features, such as [9] for computational fluid dynamics and [10] for protein unfolding simulations. In data mining related research in hurricane events, the significance of application has prompted the interdisciplinary research in recent years, such as the framework for querying and retrieving moving sensor data in hurricane events [11] and a real time storm surge forecasting system [12]. Due to the very limited availability of observational storm surge data [13], physics-based simulations become the primary method for predictions.

As the aim of our work is to detect surrogate models [14], which provide rapid approximations of more expensive models, at or between multiple points of interest (POI) in the

simulation domain, we denote a simulation as $\mathcal{F}(x)$ ($x \in \mathbb{R}^n$) and multiple points of interest as $p_1, p_2, ...p_s$. Thus, the value of each POI can be extracted from the simulation and is represented by $\mathcal{F}(x, p_i)$ or $\mathcal{F}(x, p_i, t)$ for a specific time t if time series are available at POI. Along with the simulations, there is a parameter space X^n (as $x \in X^n$) that represent the entire range of the input set for the simulation. All the factors can affect the target value at POIs. Furthermore, the representation of the parameter space is not definite for each factor: taking storm surge simulation as an example, scalar inputs such as central pressure and the pressure radius can be exactly represented by one value and all the locations are corresponding to a 2D coordinate as latitude and longitude; but the track file is organized by a set of coordinates as the location of hurricane center with time, associate with the wind velocity at each time point, but in order to find out the patterns with regard to the variation in parameter space, the representation can be re-defined to explore the space in a structural manner. This is an important part in the problem domain as it is desired to be able to quickly make predictions for an arbitrary input.

A typical applicable surrogate model, defined by domain scientists [15], is a response function that suggests a continuous surface for maximum surge (ζ) at a given location:

$$\zeta(x, y) = \phi_{km}([x_o, y_o], [c_p, R_p], [x, y]) \tag{2.1}$$

where the target variable ζ in the function is the response to the hurricane central pressure c_p, the hurricane pressure radius R_p, landfall location $[x_o, y_o]$ with a specified track angle k and forward speed m.

2.2 Basic Forms of Surrogate Models

2.2.1 Response Function Model

With the basic idea of "using a set of functions to represent a function", a typical tool, basis functions, is to be introduced. Basis functions become the base for all the surrogate models in the thesis as they provide the fundamental representational power [18].

2.2.1.1 Basis Functions

Behaving as universal approximators, basis functions play the essential role in the construction of response function. A basis functions is an element of a particular basis of a function space. Common basis includes exponential, radial basis, Fourier, B-spline and polynomial. While basis functions are usually not individually used, a set of basis functions always coexist in the model, namely *basis function system*. Denote a set of functional building blocks as ϕ_k, $k = 1,...,K$ and then a function $x(t)$ defined in this way is expressed as

$$x(t) = \sum_{k=1}^{K} c_k \phi_k(t) = C\phi(t) \tag{2.2}$$

Such a *basis function expansion* incorporates parameters c_1,c_2, ...,c_K as coefficients of the expansion. In computation, the matrix expression in the last term of Eq. 2.2 uses C to stand for the vector of K coefficients and ϕ to denote a vector of length K containing the basis functions.

In [18], it is strictly proved that a radial-basis-function network with one hidden layer is capable to represent any continuous functions as universal approximator. Actually, the proof regards the radial-basis-function network as a linear combination, so the proof shows the family of radial basis function is dense enough to cover a continuous domain. There exist other previous works showing the representational power of other basis functions [19, 20]. In the thesis, it is in a more practical view that a specific basis is chosen based on the effectiveness in the target problem solving.

2.2.1.2 Modeling Simulation Response using Basis Functions

As mentioned in Section 2.1, a response function can represent the output variation in response to the parameter space of simulation input. Functional data analysis [17] is used to construct such models. That is, multiple sets of basic functions are chosen as universal approximators for regression and each element in the parameter space contribute to one set. We denote y as the target variable, extracted from the simulation as $y = \mathcal{F}(x, p_i)$

$$y_i = \alpha + \sum_{j=1}^{n} x_j \beta_j + \epsilon_i \tag{2.3}$$

where α is the intercept, $\beta_j(t)$ (the same as $x(t)$ in Eq. 2.2) represents a basis coefficient expansion and ϵ_i is the residual. Then, x_j is one element from the parameter space, while n is the total number of parameters.

To fit this model, it would be converted to a form that least-square regression can be finally used. The optimization criterion become

$$\{\tilde{\alpha}, \tilde{\beta}\}_\lambda = \arg\min_{\alpha,\beta}\{\sum_{i=1}^{M}[y_i - \alpha - \sum_{j=1}^{n} x_j\beta_j] + \lambda\sum_{j=1}^{n}(\beta_j')^2\} \tag{2.4}$$

where λ is the regularization parameter that restricts the value range of β_j and thus avoids excessive local fluctuation in the estimated function, and M is the number of data points, equivalent to the number of simulations.

From the perspective of regression analysis, the basis function system has the following differences in comparison with ordinary curve fitting (e.g. using a polynomial): *i)* to perform curve fitting for a given function, the solving procedures always include least-squares regression in the end, which requires to solve a linear system. As for the linear system, it is desired that the number of equations should be no less than the number of known variables; and *ii)* It is computationally expensive to treat every independent variable equally in the design function. For example, when using a second-order polynomial to fit a function $y = f(x_1, x_2)$, it is not known a priori that which of x_1 and x_2 affect y in the first order or second order. Then, the design function is

$$y_i = a_1x_{i1}^2 + a_2x_{i2}^2 + a_3x_{i1} + a_4x_{i2} + a_5$$

Thus, at least 5 sampling points of $\{(x_{i1}, x_{i2}, y_i)\}$ are needed although there are only 2 independent variables.

In the surrogate model, the application can be described as: Suppose that it is needed to find a surrogate model to fit the scalar response. The input of the simulation model includes

n scalar variables $\{x_1, x_2, ..., x_n\}$ and the simulation output is just a scalar value y for the given input set. Each simulation is regarded as a *record*, so to use ordinary least-square fitting, it is required to have at least $n + 1$ records for a linear model and much more for higher-oder models. Otherwise, no solution can be found.

Therefore, with the limitation in representational power and computation using ordinary models, basis function system has the advantage of using a hierarchical structure to embed the coefficients to the model, while in essence it is still a linear model: in Eq. 2.3, both x_j and β_j are represented using basis functions. And here x_j itself can be independently represented as a function by the basis function system. So, it substantially addresses the issue of model expressiveness as long as the input is functional. And this is real in many applications or can always be designed to be functional.

In the above description, the response function is compared with ordinary least-square regression while both are supposed in the form of $y = f(X^n)$, where both y and each $x_k \in X^n$ are scalar values. This category of surrogate model is targeted to a single simulation output as (e.g. the maximum value at a specific location in the storm surge simulation) as the *scalar response* to the input. On the other hand, many simulations are performed with time evolution, meaning that the output at a POI is time series instead of a single value. While time series provide more information and the analysis of time series become more sophisticated, response functions can be more advantageous here. Thereby, surrogate models that reflect *functional response* are desired.

To illustrate the use of basis function system for modeling functional response, the form

of surrogate model is re-defined as

$$y_i(t) = \sum_{j=1}^{n} x_{ij}\beta_j(t_j) + \epsilon_i \tag{2.5}$$

With the target variable y becoming a function of time t, the corresponding problem setting is slightly different. While in Eq. 2.3 the goal is to find out the coefficients that well fit the scalar response, Eq. 2.5 embeds such coefficients in term $beta_j$ as the representational power determines that there much be such solutions. However, with regard to any of $y_1, y_2, ...y_M$, optimal coefficients are the target of this type of model given the simulation input. While these input parameters $x_1, x_2, ...x_n$ are pre-defined and don't vary with time in the setting of simulation models, they are not functional at this stage, although in the a general model it can also be a function of time t as $x_j(t)$.

Similar to scalar response, the solution of the surrogate model for functional response can also be converted into the form of an optimization problem:

$$\tilde{\beta} = \arg\min_{\beta}\{\sum_{i=1}^{M}\int [y_i(t) - \sum_{j=1}^{n} x_{ij}\beta_j(t)]^2\, dt\} \tag{2.6}$$

Although response functions demonstrate their representational power as universal approximators, domain knowledge is helpful to make sure that the model doesn't include misleading relationships, in order to avoid data dredging.

Experiments related to this part are illustrated in 4.2.2 and 4.2.3.

2.2.2 Correlation-Involved Surrogate Model

While response function can directly explore the response to simulation input, it is also interesting to discover the spatio-temporal correlation from a point of interest to another. In domain science, there also exists the theoretical foundation of the correlation: as discussed in [2], the coastal-basin geometry has a profound effect to the storm surge. While a simulation output doesn't include only a single value of the target variable, such correlation relationships between locations or variables can be helpful as a new type or a component of surrogate models.

In the context of surrogate model for simulations, while the correlated links between variables or locations may not be known a priori, the target is to find such causal links that tend to be invariants, with the examination of the variance across different simulations. Thus, statistical hypothesis testing is important to find out the links with a certain confidence level before proceeding to model construction. To specify the correlation across locations in the simulation output, y is again denoted as time series extracted from the simulation at location p_i: $y(p_i, t) = \mathcal{F}(x, p_i, t)$.

2.2.2.1 Granger Testing for Correlation Detection

Granger causality test [21] is performed to find out the specific links in the given data. The POIs $p_1, p_2, ...p_s$ in one simulation form the search space, while any directional pair $\{p_i \rightarrow p_j\}$ can be a link using a bivariate test and a multivariate test involves more points

such as $\{p_{i_1}, p_{i_2}, ...p_{i_k} \rightarrow p_j\}$. The result of the test, F-ratio and the corresponding P-value, is calculated based on the comparison between predicting the time series at p_j using only its own values and using time series from auxiliary locations $p_{i_1}, p_{i_2}...p_{i_k}$ along with its own. Then,

$$SSQE_R = \sum_{i=l}^{t} [y(p_j, i) - f(y(p_j, i-l), ..., y(p_j, i-1)]^2 \tag{2.7}$$

$$SSQE_U = \sum_{i=l}^{t} [y(p_j, i) - f(y(p_j, i-l)...y(p_j, i-1), y(p_{i_1}, i-l), y(p_{i_1}, i-l+1)...y(p_{i_k}, i-1)]^2 \tag{2.8}$$

$$F_{granger} = \frac{(SSQE_R - SSQE_U)/l}{SSQE/[t - l(1+k) - 1]} \tag{2.9}$$

In the above three equations, $SSQE_R$ is the sum of square error using the restricted method, which only involves the values at the same location and $SSQE_U$ is that using the unrestricted method, which is elaborated using values from locations $p_{i_1}, ...p_{i_k}$. Thus, the F-ratio is calculated based on the sum of square error with the specified degree of freedom in the problem setting. l represents the lag in prediction. The P-value is then easily obtained according to the value of $F_{granger}$, suggesting the probability that the null hypothesis (the values of the given auxiliary locations can improve the time series prediction) can hold.

In practice, the number of total locations would not be a large number. In storm surge

simulations, 15 locations are selected corresponding to the 15 tide gages in Louisiana **??**.Therefore, the search for the correlation relationships between locations would not result in costly computation.

2.2.2.2 Spatial-Temporal Causal Modeling

Spatio-temporal causal modeling method [23], proposed by Lozano *et al.*, is used for applying Granger causality to modeling the climate change attribution. The *spatio-temporal causal link*, which represents the causality between s set of time series, is defined by the following regression with regard to the time series at one location and its neighbors:

$$y(p_i, t) = \sum_{k=1}^{u} \sum_{l=1}^{s} \alpha_{k,l} y(p_k, t-l) \tag{2.10}$$

Also if more variables other than y are also involved in the model, then

$$y(p_i, t) = \sum_{k=1}^{u} \sum_{l=1}^{s} \alpha_{k,l} y(p_k, t-l) + \sum_{k=1}^{u} \sum_{l=1}^{s} \beta_{k,l} x(p_k, t-l) \tag{2.11}$$

where k represents a relative locations while l is the lag; the above $\alpha_{k,l}$ and $\beta_{k,l}$ are coefficients to be solved. The residual term is omitted here.

The solution of the coefficients in correlation-involved models is straightforward using least-square regression. However, a careful selection of the auxiliary locations is important. If the links can't result in stable helpfulness in prediction, it would be better to discard the link in the model. In Section 4.2.4, the parametric performance is illustrated.

2.3 Hybrid Surrogate Model

To consider a general model that works for more than one applications, it is worth incorporating the factors as mentioned in previous sections into one resultant model that can better represent the characteristics of the simulation data. Thereby, a hybrid surrogate model is taken in to account.

2.3.1 Procedural Surrogate Modeling

While large-scale simulations mimic the behavior of complex systems, it is assumed that both the response function and correlation-involved model can reflect a part of the characteristics of data. Then, it is necessary to revisit the problem setting. Based on the description in Section 2.2, one simulation is considered to be a function

$$\{y(p_1,t), y(p_2,t), ..., y(p_s,t)\} \leftarrow \mathcal{F}(X^n) \tag{2.12}$$

When we only focus on one single point of interest p_j, the response function model can be constructed with a basis function system,

$$y(p_j,t) = f(X^n, C, \phi) + \epsilon_0 \tag{2.13}$$

As a simulation model is considered to be noise free and thereby can be representable by a specific form of function system, ϵ_0 is not regarded as random effects but can be further

interpreted. Thus, it is thereby assumed that ϵ_0 results from: *i)* insufficient information is obtained from the selected point of interest p_j; and *ii)* the error from selected basis function system (error in computation or coefficients). While *ii)* is solely attributed to the response function modeling, *i)* can be alleviated by adding more information to the same model. So it is necessary to embed both the response function and correlation-involved effects into the same model. Because the response function is constructed independently from point to point, it is regarded as the *primary effect* and the spatio-temporal correlation between points of interest becomes the *secondary effect*.

Then, according to 2.2.2, spatio-temporal causal modeling can be applied on ϵ_0

$$\epsilon_0(p_j, t) = \sum_{k=1}^{u} \sum_{l=1}^{s} \epsilon_0(p_k, t - l) \tag{2.14}$$

or

$$\epsilon_0(p_j, t) = \sum_{k=1}^{u} \epsilon_0(p_k, t) \tag{2.15}$$

Eq. 2.14 suggests a typical model in the form of time series prediction as no concurrent values are used in the predictors, while Eq. 2.15 sets the lag l as 0 as concurrent values. Eq. 2.14 extends the capability of surrogate model as it can use real-time values from neighbors to improve the quality of prediction. Different from response functions, the model that Eq. 2.15 suggests would utilize historical data as a part of the model input, as offline simulations must be archived and remain available for the surrogate model. In contrast, for

response functions, the model coefficients are obtained through training with historical data and in the scenario of prediction, the surrogate can function with the model itself without any additional data.

Putting the two procedures together, the resultant model is

$$y_i(p_j) = \mathcal{F}_i(X^n, p_j) = f_1(X^n) + f_2(y(p_1), ..., y(p_u)) + \epsilon' \tag{2.16}$$

where f_1 and f_2 are referred to as the response function and correlation-involved model respectively. The denotation of time, t, as discussed with different cases, is omitted to keep the generality of the model.

This method is called *procedural surrogate modeling* as it specifies the primary and secondary and models the effects in such an order.

2.3.2 Combining multiple results

The hybrid surrogate model is constructed with the two steps as described above, by differentiating response function from correlation-involved model as primary and secondary effects. However, there always exist multiple results based on the type of model, the model parameter setting, and the data used for model training. In this subsection, a general approach is presented.

Suppose that there is multiple models m_1, m_2, ... ,m_r ($m_k = f_k(X^n, y_{p_{1...j}})$, for $k \in 1, 2, ..., r$) are trained using data sets from a pool of training data $\mathcal{D}_\mathcal{T}$. Another set

of validation data $\mathcal{D}_{\mathcal{V}}$ is available then for combination these results. The training and validation data sets are substantially the same using the same simulation model (statistically, they come from the same distribution), but may be localized to different regions of the general distribution. While the test data, as the real prediction task, may have more similarity with the validation data, it would lead to a combination in Eq. 2.17

$$\bar{f}(X^n, y_{p_{1\ldots u}}) = \sum_{k=1}^{r} b_k m_k \tag{2.17}$$

where b_k is the combining coefficient associated with the model m_k. So, the optimal $b_1, ..., b_r$ can be obtained using least-square regression:

$$\{b_1, ..., b_r\}_{\mathcal{D}_{\mathcal{V}}} = \arg\min_{b_1 \ldots b_r} \{\sum_{k=1}^{M_V} [\bar{f}(X^n, y_{p_{1\ldots u}}) - \mathcal{F}(X^n)]^2\} \tag{2.18}$$

where M_V is the size of validation data set (the number of simulations for validation).

2.4 Surrogate Detection with Statistical Inference

The approach to surrogate model construction from simulations is applicable for almost all the simulations that output time series in a domain. However, a good surrogate model, defined by the fitness and the variance across simulation samples, may not be found for every simulation model, especially for those chaotic or the models that comprise random effects themselves. In the statistical context, the task is to detect all the possible surrogate models with a certain confidence level. Statistical inference based on surrogate modeling is

discussed in the section.

2.4.1 Model Validation

A set of criteria and methods are described in order to validate the model. Other than root mean square error or mean absolute error, *coefficient of determination*(R^2) is used as a measure of how well future outcomes are likely to be predicted the model, represented by

$$R^2 = 1 - \frac{\sum_i^M (y_i - f_i)^2}{\sum_i^M (y_i - \bar{y})^2} \tag{2.19}$$

Another factor that can impact model performance is model input representation. The representation of parameter space $\{x_1, x_2, ...x_n\}$ is converted from the original input of the simulation model, and it is worth noting that domain knowledge can help with setting up a better representation to capture the characteristics of simulations. It would be important in experiments that the designed simulations can cover a problem domain, although it may not be clear a priori.

One way to test whether the designed simulations, as the training data of surrogate models, cover the problem domain is to use cross-validation: randomly dividing the data into several equivalent sets, and use most of them for training while leaving the rest for testing. A good design, as well as a good model, should not be expected to have much variance among different combinations of the data sets.

2.4.2 Hypotheses and Reliability

The statistical hypothesis testing is embedded in the surrogate modeling process. As a regression problem, the baseline hypothesis is that "whether or not the surrogate model can well fit the given simulation data" and then it comes to "whether model A is better than model B with statistical significance". There are a few notes with regard to making comparisons for model selection.

The value of R^2 is not deterministic in judging the quality of model for the following reasons: *i)* the value of R^2 depends on the application. For instance, in some applications, $R^2 = 0.9$ is not good enough while in some cases, 0.7 is acceptable; *ii)* In the situation of prediction, new data are fed to the model so there exists uncertainty in the performance. That is also why it is desired to combine multiple results. However, when modeling using the same approach, such criteria, including root mean square error, F ratio (as in the procedures of granger test), and coefficient of determination, can provide model comparison.

For simulation models, another source of uncertainty comes from the error of simulation model itself compared to the observational values. In terms of surrogate modeling, it is assumed that a simulation is noise free and has a continuous domain in response to its parameter space. However, surrogate modeling inducts a domain-specific simulation model into a general framework of statistical modeling, so more approaches to data mining are prospective in the future development.

Chapter 3

Scalable and Automated Workflow for Data-Oriented Modeling

In this chapter we illustrate our approach to addressing the following issues: *i)* how we impose parallelism in the platform as the scale of the problem increases; *ii)* how we can establish a generic framework that always facilitates the new types of surrogate models or patterns without structural change in the code or algorithm, as well as the continuous increase in the amount of data.

While the search space is rather large in two aspects: the range of geographic locations and the representation of parameter space, the task-level parallelism also depends on the workflow.

3.1 Scalability with Task-level Parallelism

3.1.1 Scalability: Task assembling

While a set of individual data mining tasks can be defined, such tasks, including the input data required, are assembled before the algorithm execution. For example, for a given

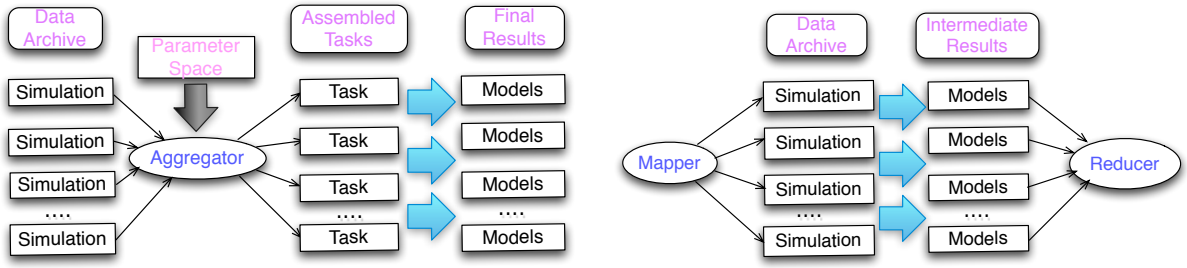

Figure 3.1: Scalability in mining hurricane simulation data. Left: Task assembling; Right: Map-Reduce

parameter space representation, a response function is to be constructed for each point of interest. So the time series of such locations are extracted and then aggregated. Thus, after the data are prepared for each task, one task can be executed by a computing unit. This scheme is suitable when independent tasks can be clearly defined (such as response function construction).

3.1.2 Scalability: Map-Reduce

While MapReduce [24] is very efficient as a programming model in Hadoop distributed file systems and other equivalents, the parallel scheme is general in processing large-scale data. For our data mining tasks such as spatio-temporal causal modeling, the discovery of causal links in one simulation is individually performed as the stage "map", while all the found links are to be validated in the rest of simulations as the stage "reduce".Figure 3.1 illustrates the execution under the two modes.

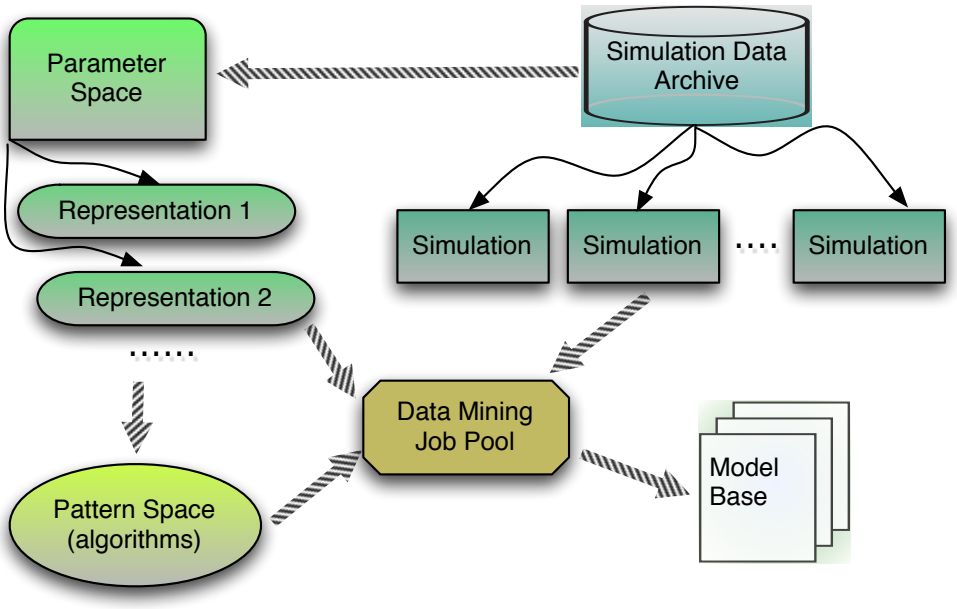

Figure 3.2: The framework structure for mining severe-storm simulation data (shadowed arrows indicate dependency)

3.2 Design Principle

The automated workflow includes three components involved in the entire process of data-driven modeling: parallel simulation, distributed data archive and core data mining. Parallel simulations provide the data source. However, a typical storm simulation using ADCIRC outputs more than 20 GBytes of data (the computational grid used consists of more than 1 million nodes) for one individual simulation, which brings challenges to data management [29] as simulations accumulate. So a distributed data archive can provide larger capacity for data with the interoperability to the original simulation environment.

There are several challenges with regard to the implementation of the workflow. First, the hierarchical parallelism is applied for better scalability: while the processing of each

POI under one simulation is the unit of a task, there exist multiple POIs in one simulation and the number of simulations keeps increasing. When the number of available computing units is larger than the number of simulations and that of POIs, a task partitioner is used to optimize the parallel performance; Second, the seamlessness between the data archive and the platform of data processing is important. If the simulation bunch has to be simply downloaded from the archive for analysis, that can slow down the workflow to a considerable extent. PetaShare [30] under Louisiana Optical Network Initiative (LONI) provides the capability of mounting and also allows users to attach programs on remote data centers [31].

3.3 Operational Automation and Workflow Extensibility

As for an *automated* workflow, we aim to ease the data mining operational procedures, especially for writing additional codes or modifying parameters to incorporate the new data. Moreover, data mining can be continuously performed in a distributed environment. The user scenario can be described as follows:

A user continuously submits simulation jobs on a cluster with the simulation output data archived to a distributed data archive right after the simulation is completed. Then, since she wants to use the prepared data mining algorithms to construct the updated models with new data involved, as well as validating the existing models stored in her working space, she submits another job with the path of the desired new data in distributed archive listed. Then, by mounting the archive (or getting the data) from the archive, the data mining tasks can be performed with the information in the model base.

Besides, extensibility is a criterion regarding the quality of the framework implementation here in order to continuously perform data mining with the increase of data and the development of more models. Especially, an updated validation is important for existing models after new data become available. At the same time, new models can be built from the enlarged database. The framework structure is presented in Figure 3.2, in which each module is extensible and thereby a check is easily performed based on the dependency between modules.

In the implementation, a simulation marker file is used to determine the addresses in storage for the simulations that are to be used for modeling. When the simulation marker files is read by the framework, the addresses are checked for the availability of data and whether or not the data have been preprocessed. For new simulations that only raw data are available in the addresses, the preprocessing is performed. Otherwise, the program directly uses the processed file, which is friendly to data analysis codes or software. This means that users only need to prepare a simple file to specify the data used for analysis, and then the framework is able to handle the procedure of scalable data processing and then prepare the data for user. In practice, the data preprocessing, such as extracting the time series of points of interest from the raw simulation output file, is the most time consuming part. Once the preprocessing has been done, user can construct any data set for modeling or analysis purposes, with the automated functionality provided by the workflow.

3.4 Challenges

In the scenario of surrogate modeling from storm surge simulation data, the huge amount of simulation output presents the most significant challenge. The ADCIRC model outputs in ASCII format and one simulation can take up more than 20 GBytes in storage as previously mentioned, which makes the time taken for data preprocessing rather long. As the data output itself is not that structural for database management, it is not easy to enhance the efficiency, especially when the storage itself becomes an issue. When more and more simulations are run, some of the raw data have to be removed in order to avoid excessive hard drive usage. It is desired to use indexing or other techniques to speed up data access and can be a part of the future work in this field.

The other challenge can be the cross-platform support of the framework. As the scalable data processing is needed, the prerequisite is that the programming language for data processing itself supports scalable computing. In the experiments of the thesis, the surrogate modeling part is written using R [25] for its abundant statistical libraries and ease of functional programming, but parallel programming in R is awkward due to the limitation of the R package itself. The improvement in framework implementation, with better compatibility, can result from hybrid programming cross platforms. Thus, a better platform should be used or designed with good scalability in terms of data-intensive operations.

In addition, from the general perspective of scalable data processing, many such jobs have common or similar workflows. It is also desired to extract a generic workflow design and result in a more robust and applicable framework.

Chapter 4

Framework Implementation and Results

In this chapter, the implementation issues, as well as the experiment results are presented. Starting from platform-specific enabling technologies, the design principle and workflow performance are discussed with regard to the workflow. Then, a series of experiments with the approaches to surrogate model construction are illustrated respectively.

4.1 Enabling Technologies

4.1.1 Louisiana Optical Network Initiative

As a resource provider of TeraGrid [26], the Louisiana Optical Network Initiative (LONI) [27] is a state-of-the-art, fiber optics network connecting large scale computer servers across Louisiana. LONI connects Louisiana and Mississippi research universities to one another as well as to the National Lambda Rail and Internet2. Usually, we request 32-128 processors to run each simulation job. When 128 processors are allocated, it takes about 15 hours to finish a simulation of a 5-day hurricane activity. In fact, it is usually the case that we need to submit more than 10 jobs simultaneously for the purpose of analysis.

4.1.2 PetaShare

PetaShare [28] is a distributed data archival, analysis and visualization cyberinfrastructure for data-intensive collaborative research. More specifically, PetaShare provides a data center that stores all the simulation data with interfaces for users to access and perform operations on the data. PetaShare is currently deployed at seven Louisiana campuses. Users can perform basic data operations, such as uploading and downloading, via either a set of commands or a web portal.

Each hurricane simulation usually produces useful data with more than 20 GBytes of disk space. Some applications however can utilize the simulation values at a few locations instead of downloading all the data. PetaShare enables this operation by mounting its available resources to the users' machines via Petafs commands. After mounting, users can run their own analytical models on LONI machines with PetaShare data. In particular, since each simulation covers the entire domain of the Gulf of Mexico, we frequently extract data for a few geographic locations from the simulations stored in PetaShare in order to perform analyses.

4.1.3 Workflow Performance

As mentioned above, extracting multiple points of interest out of millions of nodes in the mesh is time consuming, so a task partitioner can match the distribution of subtasks with the available computing resources. We present a preliminary strong scaling analysis for our

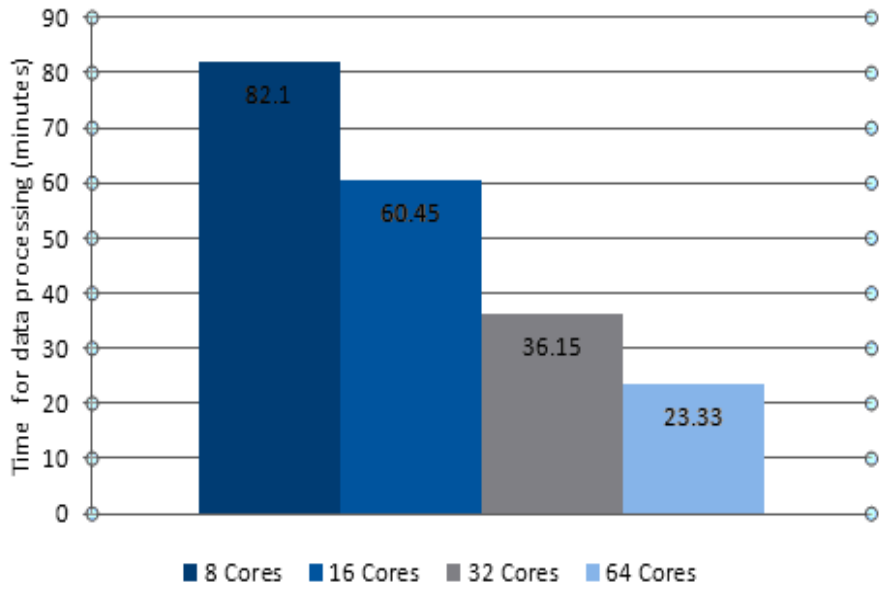

Figure 4.1: the strong scaling analysis for data processing

automated and scalable processing pipeline in Figure 4.3, with 4 selected simulations and 15 POIs in each corresponding to the tide gage stations in Louisiana. Thus, there are 60 sub-tasks for scaling.

4.2 Experiment Results

We present our current results with regard to mining severe-storm simulation data and the scalability in high-performance data processing.

4.2.1 Simulation Experiment Design

: In the scenario of storm surge prediction, we use ADCIRC model and both the wind and surge profiles are available through the storm surge simulation outputs. It is straightforward

Figure 4.2: Two hurricane track representations in simulation design (using Google Earth)

that the wind velocity at one point of interest has strong correlation to the level of surge height. However, it is not known how strong the correlation is between these two variables, as well as whether the storm surge height, which would directly impact decision makers regarding evacuation in hurricanes.

We use 25 storm surge simulations of hypothetical hurricane tracks. There are 5 landfall locations with 5 different track angles for each location. And then, the storm surge time series are extracted from the 15 POIs in Louisiana, corresponding to 15 tide gage stations. The 25 simulations cover almost all the possibilities of the track variation given the designed trajectory (currently as straightline). The track variation at one location is shown in the left of Figure 4.2. In Section 4.2.3, the right in the figure is used.

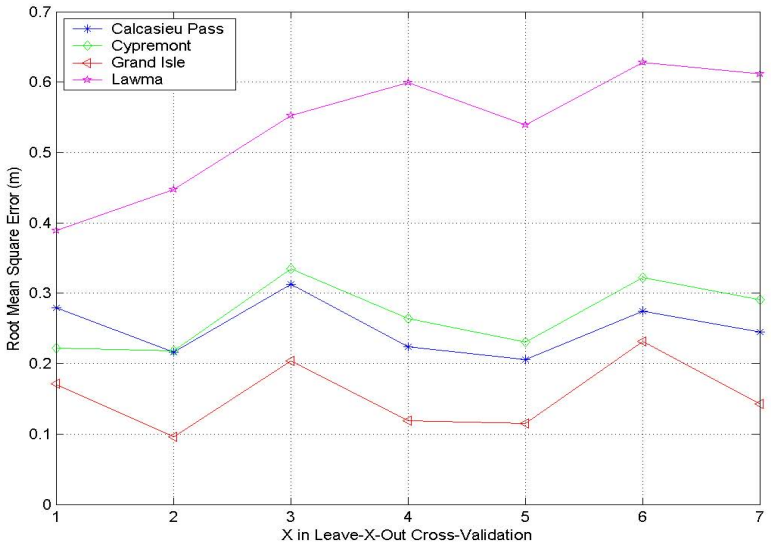

Figure 4.3: Maximum storm surge prediction as scalar response using neural networks

4.2.2 Surrogate models for scalar response

While the task is to predict the maximum storm surge height at a specific location with response to the simulation input, neural networks are used to make the prediction for each location respectively. In the spirit of cross-validation, 20 random runs are performed for a given number of simulations as training data while the rest is used for prediction. The number of simulations for prediction ranges from 1 to 7. The results are shown in Figure 4.3

It is clear to see the spatial heterogeneity, which means that at different locations the surge response would be different. In Grand Isle, the storm surge height is relatively easy to predict. The cause is due to the geometry and the direction of hurricane wind field.

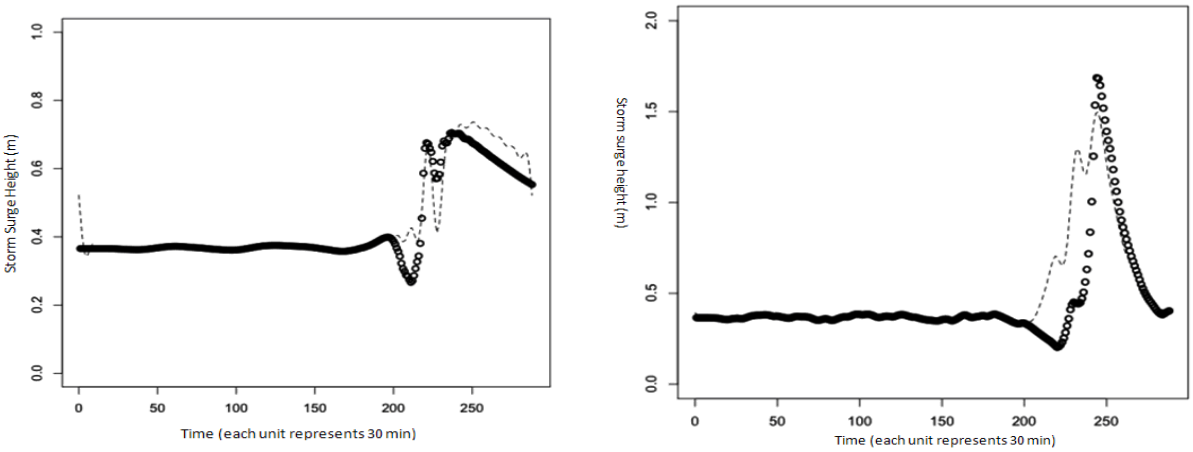

Figure 4.4: Storm surge time-series analysis as functional response (circled: simulation; dotted: surrogate)

4.2.3 Surrogate models for functional response: time series analysis for storm surge

Then, time series analysis is performed with the storm surge simulation data as shown in Figure. 4.4. The illustrated two figures are picked from all the 15 locations to show the simulated storm surge profile and the output of the surrogate model.

4.2.4 Correlation-Involved Models

4.2.4.1 Correlation-Based Scalar Response

With the additional data of wind velocity profile from simulation outputs, the surrogate model is improved for each location to get the maximum storm surge height. Table 4.1 shows the coefficient of determination (R2) and F-ratio at selected locations.

Table 4.1: Using wind-surge correlation to model the scalar response

Location	R2	F-ratio(18,9)
Carrollton	0.9661	14.272
New Canal	0.9279	6.441
Grand Isle	0.9679	15.116
Cypremont	0.9937	12.038

With a significant correlation between maximum storm surge height and wind velocity profile, the model can be potentially used for the storm surge forecasting with the input of wind field in the occurrence of a new hurricane, as the wind model is computationally much cheaper than the surge model.

4.2.4.2 Spatio-Temporal Causal Link

The spatio-temporal causal links over the 25 storm surge simulations, with a confidence level above 85%, are shown in Figure 4.5.

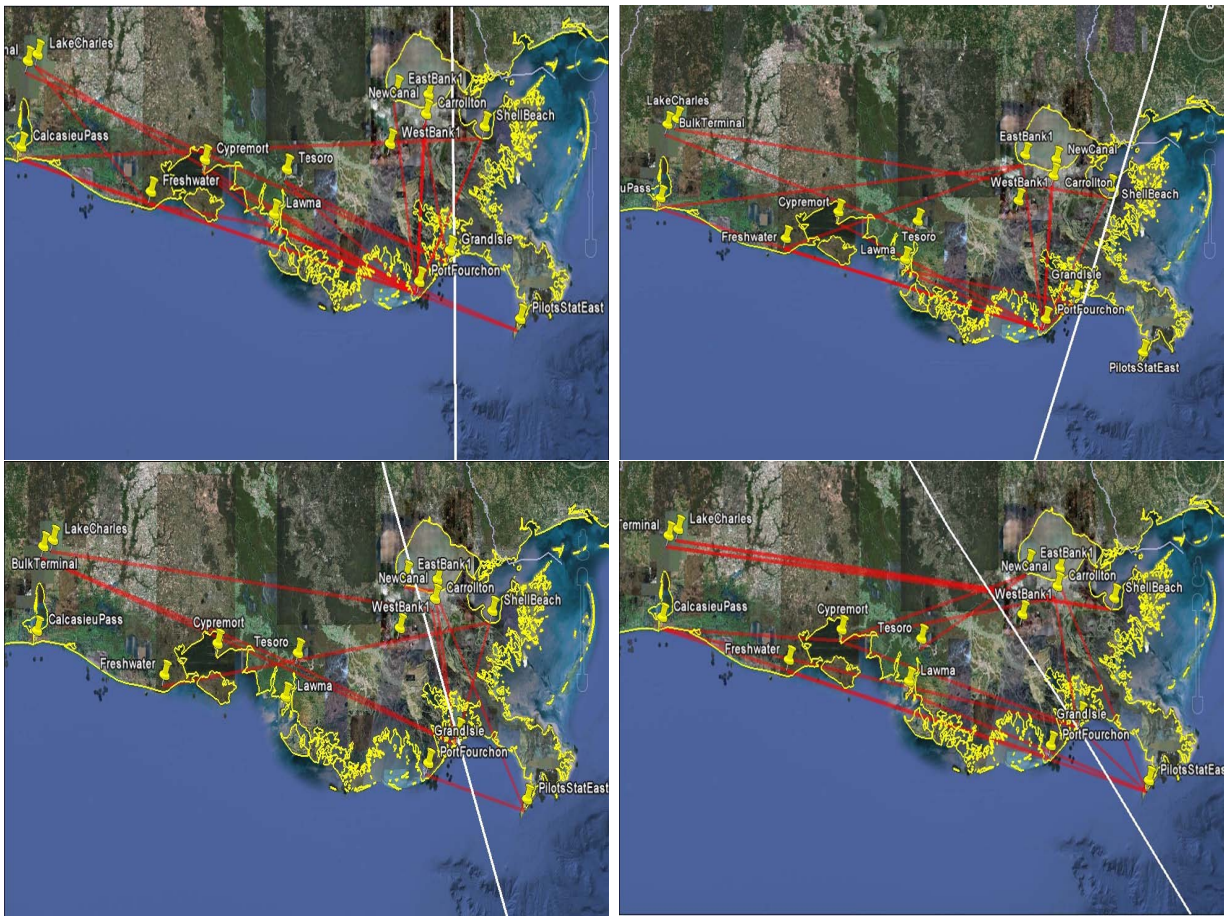

Figure 4.5: Spatial-temporal causal links in the simulations (with Google Earth)

Chapter 5

Summary, Conclusions and Recommendations

In this thesis, we present the methodology of surrogate model detection and construction as well as a scalable and automated workflow along with the methods that we use at the current stage to mining severe-storm simulation data. The workflow efficiency and the experiment results using the surrogate modeling approach have been shown with the application of storm surge prediction during hurricanes. Different types of data-oriented models reflect multiple implicit factors hidden in the simulation data and a statistical framework is proposed to validate such models.

Based on the surrogate modeling, an automated and scalable workflow is proposed in order to efficiently process the simulation data. Considering the simulation data involved in many applications are with large scale, the task-level parallelism in data processing is described and implemented in the framework. The framework structure is presented with user scenarios. Some considerations with regard to system design principle and framework implementations are also discussed as a general guideline for data-intensive computing systems and applications [32].

Based on the experiment results shown in the thesis, the following conclusions can be drawn: *i)* Lightweight surrogate models based on the scalar and functional response of simulations to its parameter space can help with making predictions in a significantly shorter time than running simulations themselves; *ii)* In large-scale simulations with the value of multiple points of interest available, the spatio-temporal correlation can lead to correlation-involved models that alleviate the error in response functions due to the dynamics of physical model and such models can be used along with response functions as a hybrid model. For the future development on both the modeling and framework sides, a set of recommendations are listed.

In an engineering point of view, the data mining process goes with the size of the simulation data archive keeping enlarged. So the problem scale, or the search space, would tremendously increase for a better confidence level in prediction, making scalable computing and convenient manipulation necessary. For example, when hundreds and thousands of simulations become available, it would cover tens of parameter space representations and thereby possibly hundreds of data mining subtasks at the same time. We have given a careful consideration to the extensibility and capacity for operations on a distributed environment. Besides, another direction is to construct a general workflow system with regard to scalable data processing and analytics.

As for the modeling approach itself, there exists space for refining the existing models and reducing the uncertainty. Especially, when it proceeds to the real-time forecasting with the input of observational data, more data mining techniques are to be incorporated in the

model. The current results have been shown potential for the application in engineering and decision making.

In specifics, the following points can be followed as future work.

- A new programming model, or design pattern, can be used for the framework for better scalability, including incorporating the R statistical programming into C++ for robustness and using Hadoop for parallel processing.
- A general abstract interface can be proposed for more types of simulations and it can result in a generic workflow or framework. Then the surrogate modeling methods can also be used in a broader context.
- More recent achievements in machine learning and data mining, such as Gaussian processes, can be used to elaborate the surrogate modeling approaches.

Furthermore, as the goal of surrogate modeling also includes bridging the gap between simulation and sensor data in terms of real-time forecasting. New models are desired to boost simulations to make them more accurate. While simulations are still used as the primary way in prediction in various real-world applications, there exist lots of potential in this direction.

Bibliography

[1] C. Mattocks and C. Forbes, "A real-time, event-triggered storm surge forecasting system for the state of North Carolina," in *Ocean Modeling*, vol. 25, pp. 95-119, 2008.

[2] D.Resio and J. Westerink, "Modeling the physics of storm surges," *Physics Today*, vol. 61, iss. 9, pp. 33-38, 2008.

[3] C. Karacan, "Evaluation of the relative importance of coalbed reservoir parameters for prediction of methane inflow rates during mining of longwall development entries," *Computer and Geosciences*, vol. 34, iss. 9, pp. 1093-1114, 2008.

[4] DynaMIT: a simulation-based system for traffic prediction. URL: http://web.mit.edu/its/dynamit.html

[5] A. Uhrmacher, A. Seitz, "Case-based simulation of ecological and biological systems," *Systems Analysis Modeling Simulation*, vol. 39, iss. 2, pp. 215-234, 2000.

[6] T. Hengl, H. Sierdsema, A. Radovic and A. Dilo, "Spatial prediction of species distribution from occurrence-only combining point pattern analysis, ENFA and regression-krigging," *Ecological Modeling*, vol. 220, pp. 3499-3511, 2009.

[7] ADCIRC Coastal Circulation and Storm Surge Model. URL: http://adcirc.org/

[8] T.F. Brady and E. Yellig, "Simulation data mining: a new form of computer simulation output," in *Proc. of the 2005 Winter Simulation Conference*, pp. 285-289, 2005.

[9] M. Jiang et al., "Feature mining paradigms for scientific data," in *Proc. of SIAM International Conference in Data Mining (SDM'03)*, pp. 13-24, 2003.

[10] D. Berrar et al., "Towards data warehousing and mining of protein unfolding simulation data," *Journal of Clinical Monitoring and Computing*, vol. 19, pp. 307-317, 2005.

[11] S.-S. Ho, W. Tang, W.T. Liu and M. Schneider, "A framework for moving sensor data query and retrieval of dynamic atmospheric events," in *Proc. of the 22nd International Conference on Scientific and Statistical Database Management (SSDBM'2010)*, pp. 96-113, 2010.

[12] J.G. Fleming, C.W. Fulcher, R.A. Luettich, B.D. Estrade, G.D. Allen, and H. S. Winer, A real time storm surge forecasting system using ADCIRC, *Estuarine and Coastal Modeling X*, M. Spaulding [ed], American Society of Civil Engineers, 2008.

[13] D.T. Resio, J. Irish and M. Cialone, "A surge response function approach to coastal hazard assessment - Part 1: Basic concepts," *Natural Hazards*, vol. 51, no. 1, pp. 163-182, 2009.

[14] Z. Qian, C.C. Seepersad and V.R. Joseph, "Building surrogate models based on detailed and approximate simulations," *Journal of Mechanical Design*, vol. 128, iss. 4, pp. 668-677, 2006.

[15] J. Irish, D.T. Resio, and M. Cialone, "A surge response function approach to coastal hazard assessment - Part 2: Quantification of spatial attributes of response functions," *Natural Hazards*, vol. 51, no. 1, pp. 163-182, 2009.

[16] D. Abramson et al., "Parameter space exploration using scientific workflows," in *Proc. of International Conference of Computational Science (ICCS'09)*, pp. 104-113, 2009.

[17] J. Ramsay and B. W. Silverman, *Functional Data Analysis (2nd ed.)*, published by *Springer*, pp. 261-277, 2005.

[18] J. Park and I.W. Sandberg, "Universal approximation using radial-basis-function networks," *Neural Computation*, vol. 3, no. 2, pg. 246-257, 1991.

[19] L.-X. Wang and J.M. Mendel, "Fuzzy basis functions, universal approximations and orthogonal least-squares learning," *IEEE Transactions on Neural Networks*, vol. 3, no. 5, pp. 807-814. 1992.

[20] A.R. Barron, "Universal approximation bounds for superpositions of a sigmoidal function," *IEEE Transactions on Information Theory*, vol. 39, no. 3, pp. 930-945, 1993.

[21] C.W.J. Granger, "Investigating causal relations by econometric models and cross-spectral methods," *Econometrica*, vol. 37, no. 3, pp. 424-438, 1969.

[22] NOAA (National Oceanic and Atmospheric Administration) Tides and Currents: http://tidesandcurrents.noaa.gov.

[23] A. Lozano, H. Li and et al., "Spatial-temporal causal modeling for climate change attribution," in *Proc. of the 16th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD'09)*, 2009.

[24] J. Dean and S. Ghemawat, "MapReduce: simplified data processing on large clusters," in *Proc. of USENIX Symposium on Operation Systems Design and Implementation (OSDI'04)*, 2004.

[25] R Development Core Team, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, ISBN 3-900051-07-0, 2011.

[26] TeraGrid. URL: http://teragrid.org/

[27] Gabrielle Allen and Charles McMahon and Edward Seidel and Tom Tierney, *The 2003 Louisiana Optical Network Initiative (LONI) Concept Paper*, CCT Technical Report Series, Louisiana State University, 2009.

[28] PetaShare: A Distributed Data Archival, Analysis and Visualization Cyberinfrastructure for Data-Intensive Collaborative Research. URL: http://petashare.org/

[29] S. Tummala and T. Kosar, "Data management challenges in coastal applications," *Journal of Coastal Research*, special issue no. 50, pp. 1188-1193, 2007.

[30] PetaShare: A Distributed Data Archival, Analysis and Visualization Cyberinfrastructure for Data-Intensive Collaborative Research: http://www.petashare.org/

[31] H. Bhagawaty, L. Jiang and et al., "Design, implementation and use of a simulation data archive for coastal science, in *Proc. of International ACM Symposium on High-Performance Parallel and Distributed Computing (HPDC'10)*, pp. 651-657, 2010.

[32] L. Jiang, G. Allen and Q. Chen, "Scalable and automated workflow for large-scale severe-storm simulations", to appear in *Proc. of the 23rd International Conference of Scientific and Statistical Database Management (SSDBM'11)*, 2011.

Vita

Lei Jiang was born in Wuhan, China, in 1986. He grew up in Wuhan and was exceptionally recruited by the Special Class for the Gifted Young, University of Science and Technology of China, when he was fifteen, right after the compeletion of only one-year study in senior high school (which is normally for three years). Then he obtained a Bachelor of Engineering in Computer Science from University of Science and Technology of China in 2005 and a Master of Engineering in Applied Computer Science from Wuhan University in 2007, respectively. After a short period in Canada, Lei transferred to Louisiana State University in July of 2008. Currently he is still a doctoral student at LSU.