Calibration of Hydrologic Models Using Distributed Surrogate Model Optimization Techniques: A WATCLASS Case Study

by

Mahtab Kamali

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Systems Design Engineering

Waterloo, Ontario, Canada, 2009

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners. I understand that my thesis may be made electronically available to the public.

Abstract

This thesis presents a new approach to the calibration of hydrologic models using distributed computing framework. Distributed hydrologic models are known to be very computationally intensive and difficult to calibrate. To cope with the high computational cost of the process a Surrogate Model Optimization(SMO) technique built for distributed computing facilities is proposed. The proposed method along with two analogous SMO methods are employed to calibrate the WATCLASS hydrologic model. This model has been developed at the University of Waterloo and is now a part of the Environment Canada MESH (Environment Canada community environmental modeling system called Modèlisation Environmentale Communautaire (MEC) for Surface Hydrology (SH)) systems. SMO has the advantage of being less sensitive to the “curse of dimensionality” and it is very efficient for large scale and computationally expensive models. In this technique, a mathematical model is constructed based on a small set of simulated data from the original expensive model. SMO technique follows an iterative strategy which in each iteration the SM map the region of optimum more precisely.

A new comprehensive method based on a smooth regression model is proposed for the calibration of WATCLASS. This method has at least two advantages over the previously proposed methods: it does not require a large number of training data and it does not have many model parameters and therefore its construction and validation process is not demanding.

To evaluate the performance of the proposed SMO method, it has been applied to five well-known test functions and the results are compared to two other analogous SMO methods. Since the performance of all SMOs are promising, two instances of WATCLASS modeling Smoky River watershed are calibrated using these three adopted SMOs and the resultant Nash numbers are reported.

Acknowledgements

I am deeply grateful to my supervisors, professors K. Ponnamabalam and E.D. Soulis. Dr. Ponnambalam taught me how to find solutions for challenging real world problems and Dr. Soulis taught me to be patient, hard working and optimistic, even in the worst situations.

I would like to thank my thesis committee members, Dr. F. Karray, Dr. H. Tizhoosh, Dr. D. Burn and Dr. V. Van Nguyen for their time and useful comments.

I would like to thank my colleague Frank Seglenieks for his help throughout this research.

My especial thanks go to my friends Alaleh Dara, Faezeh Jahanmiri, Bita Roshanaie and Homeyra Pourmohammad Ali for their emotional support, camaraderie, entertainment, and the caring they provided.

I also owe thanks to my best friend and sister, Morvarid for her unconditional love as well as to my brothers, Matin and Moeen for their support.

I am forever indebted to my parents, Mahin Moezzi and Mohammad Mehdi Kamali. There is no suitable word that can describe their love and support throughout my life. They raised me, supported me, taught me, and loved me. This thesis is dedicated to them as a token of my gratitude.

I also wish to thank my little Soroush for being such a patient son.

Last but not least, I would like to express my gratefulness to my beloved husband, Ali Ghodsi Boushehri. I owe thanks to him not only for his patience and support but also for his enlightening scientific insights, which always helped me to arrive at better solutions.

Contents

List of Tables **ix**

List of Figures **x**

1 Introduction **1**

1.1 Problem Definition and Objectives 1

1.2 Organization of the Dissertation 5

2 Literature Review **6**

2.1 Introduction 6

2.2 Calibration 6

2.2.1 Introduction 6

2.2.2 Manual Calibration 7

2.2.3 Automatic Calibration Strategies 8

2.2.4 Global Optimization Methods 11

2.2.5 Surrogate Model Optimization (SMO) 14

2.3 Sensitivity Analysis 26

2.3.1 Introduction 26

2.3.2 Local Sensitivity Analysis 28

2.3.3 Global Sensitivity Analysis 28

2.3.4 Gradient based methods 29

2.3.5 Variable Ranking (Parameter Screening) 32

2.3.6 Variance based methods 37

2.3.7 Association based methods 38

2.4 Summary 40

3 Methodology **41**

3.1 Introduction 41

3.2 Calibration 42

3.2.1 Regularized Function Map (RFM) 42

3.2.2 Comparison of RBF, DACE and RFM SMO Methods on test functions 48

3.2.3 Adapted Surrogate Function for Distributed Computing (ASFDC) . 53

3.3 Sensitivity Analysis for Calibration Purposes 60

3.3.1 Introduction 60

3.3.2 PCA for Sensitivity Analysis Application 64

3.4 Summary 64

4 Case Study **66**

4.1 Smoky River Watershed 66

4.1.1 Smoky landcover 66

4.1.2 Parameters definition 69

4.2 WATCLASS 71

4.2.1 WATCLASS Gridding 76

4.2.2 Different Versions of WATCLASS 77

5 Results **81**

5.1 Introduction 81

5.2 Calibration 82
5.2.1 Sampling Strategy 83
5.2.2 Calibration of WATCLASS with sixteen parameters 84
5.2.3 Discussion of the WATCLASS Calibration Results-Part I 88
5.2.4 Calibration of WATCLASS with forty-four parameters 89
5.2.5 Discussion of the WATCLASS Calibration Results-Part II 92
5.3 Results of Sensitivity Analysis 94
5.3.1 Morris Method 94
5.3.2 Standard Regression Coefficients (SRC) 96
5.3.3 Kolmogorov-Smirnov 97
5.3.4 Results of entropy tests for grid-based sampling 99
5.3.5 Discussion of the first part of Sensitivity Analysis 100
5.3.6 Sensitivity Analysis using SMO 101
5.3.7 Results of WATCLASS calibration using PCA 102
5.4 Summary 105

6 Conclusion 106

6.1 Summary and Conclusion 106
6.2 Contribution 107
6.3 Future Work 108

APPENDICES 110

.1 Taylor Expansion 111
.2 Model Validation 114
.3 Abbreviation 116

Bibliography 117

List of Tables

3.1 The Dixon-Szegő test functions 51

3.2 Comparison of DACE and RBF and RFM on Dixon -Szegö 51

3.3 Comparison of ARBF and ADACE and ARFM on Dixon-Szegö 59

4.1 CLASS.ini Parameter description 70

4.2 CLASS.ini Parameter description 72

4.3 Calibration results for CLASS.ini Parameters 80

5.1 Statistics of 16 parameters WATCLASS 84

5.2 Statistics of 44 parameters WATCLASS 84

5.3 The results of ADACE on WATCLASS with 16 parameters 86

5.4 The results of ARBF on WATCLASS with 16 parameters 87

5.5 The results of ARFM on WATCLASS with 16 parameters 87

5.6 The results of ADACE on WATCLASS with 44 parameters 91

5.7 The results of ARBF on WATCLASS with 44 parameters 91

5.8 The results of ARFM on WATCLASS with 44 parameters 92

5.9 Morris 95

5.10 Standard Regression Coefficients (SRC) 97

5.11 Kolmogorov-Smirnov test 98

5.12 Entropy test 100

5.13 PCA and ARFM on WATCLASS with 44 parameters 103

5.14 Table of comparison 104

List of Figures

3.1 Schematic approach of RFM method 50

3.2 The proposed framework for performing automatic calibration 58

4.1 Smoky River location in Alberta 67

4.2 Smoky River different land covers 68

4.3 WATCLASS hydrograph (NS= -0.3) 78

4.4 WATCLASS hydrograph (NS= 0.72) 79

Chapter 1

Introduction

1.1 Problem Definition and Objectives

The idea of employing hydrologic models for a better understanding of water circulation in the environment, and in particular for predicting streamflow, has a four decade history. These models require some numerical preprocessing such as parameter calibration and sensitivity analysis to conduct their task properly. The early models were compatible with the computational facilities of the time and thus they were mostly simple, conceptual models. With improvements in computational resources, scientists found the ability to develop more detailed distributed models, which are more computationally intensive. In addition, depending on location and domain, the hydrologic model simulation might run for a couple of hours or even days. These problems are not specific to hydrology and exist in many research areas such as the aerospace industry that deal with complicated phenomena.

This thesis presents some numerical investigations, such as a calibration and sensitivity analysis of the WATCLASS hydrologic model (Soulis et al., 2000). The limitations/conditions of the WATCLASS model are:

- this model is a physically distributed hydrologic model with high computational costs,
- a large number of model parameters leads to the need for a large number of model simulations for conducting parameter estimation of the model,
- this frequent model updates causes differences in two consecutive model versions and hence frequent model reassessment.

The mentioned conditions and limitations are valid for any large scale evolving model that requires calibration.

The series of simulations performed during this study are aimed to help WATCLASS modelers in the model validation, sensitivity analysis and calibration procedure. One of the main issues of this study was dealing with frequent updates of the WATCLASS model. Due to these updates, it was possible to obtain different outputs in two consecutive versions. For the sake of comparison, some experiments shown in this document have been done on the newer version (WATCLASS 3.3) and some on the older version (WATCLASS 2.7). The results clearly show the improvement of the WATCLASS model throughout the course of this research. During this research we realized that these model updates are unavoidable and necessary. Over time the model is improved through field work yet, it delivers unexpected results. At the beginning of this study, all of the simulations are more for diagnostic purposes than for sensitivity analysis or calibration. Subsequently, when the model produced reliable results and became validated on different watersheds, we began to conduct sensitivity analysis and calibration experiments.

To cope with the inevitable WATCLASS model updates, a flexible algorithm for model calibration is suggested and implemented to serve the needs of the researchers in the field. The proposed algorithm complies with model limitations.

These three facts and the potential of distributed CPUs propelled us to develop a robust calibration strategy suitable for the calibration of computationally expensive models using several CPUs.

Until recently, model calibration on a single computer using the most efficient evolutionary or surrogate model optimization techniques was standard. Although these methods were fast compared to their previous versions, running more than two thousand instances of WATCLASS using one single computer was not practical.

The potential of access to a distributed computing facility was a strong motivation for the development of a more efficient distributed calibration framework. At the beginning of this research, Flexor, a 12-CPU computational facility, provided a reliable source of multiple computers. Subsequently SHARCNET, a multi-institutional high performance computer network with hundreds of CPUs makes it possible to perform hundreds of simulations simultaneously. The mentioned possibilities and limitations were the main factors for shaping the present framework.

New research and improvements in soft modeling tools and machine intelligence provided new directions for model calibration. Instead of searching for the best parameter set using the original model, a surrogate model is built based on a limited number of points from the original model. Once this surrogate model met the model validation criteria it can work as the tool for finding the best set of parameters of the original model.

The other concern of this research was to discover a practical method for recognition of the most important WATCLASS parameters. The proposed sensitivity analysis aims to guide modelers to avoid the common process of overparameterization of the model.

There are different types of sensitivity analysis methods each with different approaches. This document offers an overview of the most important ones and presents the results of the ones that are computationally feasible.

The main goal of this work is to find an acceptable results in the least possible time with the available facilities. The length of computation time spent to find a good set of parameters is critical, as the final target of any hydrologic model is its applicability to real time decision-making environments. WATCLASS was mainly developed to work as a part of the Environment Canada weather prediction model. For real time modeling, finding a good set of parameters in a short time is more critical than finding the best (if possible at all) set of parameters in a long time. The physically-based hydrologic models are very good approximation of nature, yet they contain many simplifications and process estimations. These models are most likely nonlinear, nonconvex and non-monotonic, therefore searching for their global optimum is theoretically impossible. Even if the global optimum was found it is not rational to spend a huge amount of time and computational cost for finding the optimum of an approximate model.

This document suggests an approximate modeling based optimization framework covering these requirements. The model approximation process starts with a coarse mesh of the space with a small number of model simulations and it becomes finer in the regions of interest. The approximate model shrinks as the mesh becomes finer to cover a smaller area with a higher precision. Once a reasonable result is achieved the optimization process is terminated. This process begins with a global search and as the process continues it focuses more on the local search.

The target of this research is to develop a calibration technique that provides appropriate model parameters without intensive computational efforts and is compatible with distributed computing facilities.

1.2 Organization of the Dissertation

Chapter 2 reviews existing techniques for the calibration and sensitivity analysis of hydrologic models. Two related but different subjects, calibration and sensitivity analysis of hydrologic models, are the main subjects of this research. This chapter is divided into two main sections. The first section reviews relevant optimization techniques designed for the optimization of large scale expensive models. This review is followed by the details of the calibration methods (calibration is searching for the parameters that minimizes model error) that will be used and modified. The second section provides a description of different sensitivity analysis methods and highlights the most important ones in each category.

The first part of Chapter 3 explains the proposed method for calibration, and presents experimental results that compare the performance of the proposed method with competing methods. Afterward, the necessary modification to the proposed method and two other competing methods for making them compatible with distributed computing facilities is explained. The second part of this chapter investigates the suggested method for performing sensitivity analysis.

Chapter 4 describes a case study. The case study is to calibrate the WATCLASS hydrologic model and analyze the model sensitivity with respect to its parameters. The first part of this chapter explains the Smoky Watershed and the second part explains WATCLASS.

Chapter 5 presents the results of calibrating WATCLASS. The comparison of the proposed calibration method and two other surrogate model optimization methods are presented first. This is followed by calibration of WATCLASS using the suggested sensitivity analysis methods.

Chapter 6 presents the summary of the conducted research and the obtained results.

Chapter 2

Literature Review

2.1 Introduction

This chapter presents the calibration and sensitivity analysis of physically-based hydrologic models applied on large catchments. Calibration and sensitivity analysis are two related subjects that have different natures and roots. To serve these differences, this chapter is divided into two parts. Calibration is explained in section 2.2 and sensitivity analysis is explained in section 2.3. Each section reviews different aspects of each subject with a detailed review of the methods that have been employed in this dissertation.

2.2 Calibration

2.2.1 Introduction

Calibration is a general term that refers to the process of finding acceptable values for parameters of a physical model. Any physically based model including physically distributed hydrologic models is built from an assembly of mathematical equations that approximate the processes happening in nature.

Approximation of nature exists in almost every single equation. The claim behind each one of them is the best approximation and none of them claims to model nature perfectly. Often, these equations are made up of a combination of a mathematical equations and parameters and coefficients. Calibration of a model involves tuning these parameters and coefficients to have the best matching output to the observed values.

In this document, a physically distributed hydrologic model called WATCLASS, which is constructed to model the mechanism of water movement in a watershed and predict the streamflow, is calibrated. Details of this model are explained in section 4.2. To model a catchment using WATCLASS, the basin is divided into smaller piece (grids) and the collected results of these pieces build the output results.

For calibration, a goodness of fit measure between the model output and the observed value is selected to choose the corresponding best parameter set. A set of model parameters is accepted if the model output is similar to the observed value and rejected if model prediction is very different from observation.

In general, the calibration process is similar to an optimization process. This process is complicated since the model space is not convex, the model computational cost is high and the model dimensionality is high.

There are two different approaches for the calibration of hydrologic models: manual calibration and automatic calibration. A brief explanation of these techniques is presented next.

2.2.2 Manual Calibration

Manual calibration is generally the process of finding an acceptable parameter set by trial and error. In this process, experts are directly involved and use their expertise to search the parameter space. This close interaction makes the manual calibration process extremely laborious and expert-dependent yet very reliable, informative and precise (Madsen, 2000). This process helps the modeler to better understand the model and to track the problems in the algorithm. However, manual calibration loses its utility as the model becomes more reliable and complete. The benefit of gaining knowledge through intimate interaction with the model is offset by the time-consuming nature of the manual calibration.

Manual calibration used to be a common practice for calibration of watershed models in past decades. The development of computer resources and the availability of optimization algorithms has made this intensive human-model interactive process obsolete (Boyle et al., 2000).

The new improvements in computational resources have motivated hydrologists to employ or develop automatic calibration algorithm for hydrologic models (Duan et al., 1992).

2.2.3 Automatic Calibration Strategies

Automatic calibration refers to a calibration process in which an algorithm searches the parameter space and finds the best parameter set. This technique has dominated manual calibration since it is not labor intensive, it is not biased by the expert knowledge and it is robust and adjustable with model improvements (Madsen, 2003).

In general, an automatic calibration process has three main parts (Gupta et al., 1998):

- *A criterion for comparison of model simulated results with observed values: An error measure of performance is necessary to evaluate the model and compares its output with observed values.* This measure reflects the goodness of fit between model output and observed values. Two common error measurements are the mean squared error (MSE) and the sum of squared errors (SSE). Though these criteria have been used in the hydrology community these criteria have the tendency to be sensitive to outliers. Another more common error criterion in hydrology is Nash-Sutcliffe (Nash & Sutcliffe, 1970), a normalized SSE, which is defined as:

$$NS(x) = 1 - \frac{\sum_{i=1}^{n}(Y_{observed} - Y_{simulated})^2}{\sum_{i=1}^{n}(Y_{observed} - E(Y_{observed}))^2} \tag{2.1}$$

In this equation $Y_{simulated}$ is the model predicted value (streamflow in WATCLASS) and $E(Y_{observed})$ is the expected value of observed value or $Y_{observed}$. Equation 2.1 is defined for time interval from 1 to n, depending on the model time interval.

NS ranges from $-\infty$ to 1, where negative values of NS mean that the prediction of the model is very different from the observed values and NS values close to 1 mean that model output is close to the observed values. NS measure compares model prediction and performance of longterm averages. In this study NS numbers less than 0.2 are considered poor model prediction and NS numbers larger than 0.8 are considered near perfect prediction. The other error measure for evaluation of model output is the daily root mean square (DRMS) which is defined as:

$$DRMS = \sqrt{\frac{1}{n}\sum_{i=1}^{n}[Y_{simulated} - Y_{observed}]^2} \tag{2.2}$$

This measure is a variance estimator which works under the assumption that i^{th} data point $error(i) = Y^{i}_{simulated} - Y^{i}_{observed}$ is normally distributed with a mean of zero and a constant variance. As this measure is less common than NS, NS was used for the WATCLASS case study.

- *Important model parameters: Sensitive parameters along with their distribution information should be recognized.* The second part of an automatic calibration is the investigation of the sensitivity of parameters and the influence of one/couple of parameters on the model output. There are different perspectives for conducting sensitivity analysis. Each sensitivity analysis method is developed to serve a certain goal. For instance, in the early stages of model development, recognizing the abrupt change in model output due to small perturbations in an unimportant model parameter might lead to a complete model investigation. An entire chapter of this document is devoted to sensitivity analysis methodology.

- *An optimization algorithm: The third part of an automatic calibration method is the choice of an optimization algorithm.* This part is considered to be the most important part among these three. The most common method is to seek the best parameter set of the model to achieve the best fit of the model output to the observed values (or the minimum error) by a global optimization algorithm such as genetic algorithm ((Holland, 1975) and (Fraser & Burnell, 1970)), simulated annealing (Metropolis et al., 1953) and DDS (Tolson & Shoemaker, 2007b). Usually, this process starts from a point and proceeds to the next acceptable point and continues until the algorithm reaches to the optimum. The corresponding parameter set is the best parameter set.

This was the general approach to automatic calibration, however it is possible to adapt each step of this procedure for each particular application.

In the next section, a brief explanation of different algorithms for calibration is reviewed. The algorithms are divided into two main categories: Global Optimization Methods and Surrogate Model Optimization Methods. The first category is a broad range of algorithms that covers both gradient-based and non-gradient-based algorithms. Though these algorithms have been applied to many applications and showed a high potential in solving many optimization problems, their computational cost is generally high.

The second category, Surrogate Model Optimization method, has found many applications in optimization of large scale models. The main body of this chapter discusses the second category. For completeness, some of the most common Global Optimization algorithms are reviewed and WATCLASS has been optimized using one that is shown to outperform most of these methods (Tolson & Shoemaker, 2007b).

2.2.4 Global Optimization Methods

Global optimization methods refer to methods that search model space globally. These methods are commonly used when the model is nonlinear and non-convex with many local minima. Global Optimization methods are divided into two main categories: Multistart local optimization methods and Evolutionary methods.

Multistart Local Optimization

Derivative-based Optimization methods When an accurate derivative is obtainable, these types of methods are the most efficient among local optimization methods. However, in simulation based problems where the analytical derivative of the model requires a substantial computational effort, partial derivatives using finite difference, or automatic differentiation (Nocedal & Wright, 1999) can be used. Automatic differentiation attaches to the source code of the objective function to produce derivatives yet the results are sometimes inaccurate. To compute the partial derivative of the model at a point, it is required to simulate the model k (model dimension) times in the vicinity of that point to compute the gradient of the model for that point. As this process needs substantial computational efforts, it is not recommended for high dimensional models.

Derivative-free trust region methods Trust-region methods are derivative-based local optimization methods. Powell (Powell, 2002) introduced a derivative-free trust-region method that uses a quadratic approximation of the objective function instead of the original expensive model. This approximate model is only valid in the trust-region, where we can trust the approximate model within that region. By the replacement of the approximate model in lieu of the original objective function, the computational burden of the optimization process will be decreased. This process converges to the local minimum and there is no proof that the global minimum is found.

Pattern Search Pattern Search was initially introduced by Torczon (Torczon, 1997). This method works with a mesh of points that shrinks or expands with respect to the value of the objective function. The mesh structure is called a pattern. This optimization process is a good candidate for cases where a derivative is not available, as it works regardless of the objective function derivative. Therefore, this method is a good choice for nonsmooth and noisy objective functions. This method is computationally expensive, as it needs many function evaluations for construction of the mesh.

Evolutionary Optimization

In this section, global optimization methods that start from a randomly selected point are investigated. These methods usually are evolutionary-based algorithms and heuristically search the space of objective function.

The most well-known methods in this category are genetic algorithm (GA) (Fraser &

Burnell, 1970), simulated annealing (SA) (Metropolis et al., 1953) and their extensions. Though, these methods are quite successful in finding near global optima, there is no mathematical proof for any of them. The one exception is SA if the annealing process cools down very slowly. The main disadvantages of these methods are that they are computationally intensive. Duan (Duan, 2004) has extensively reviewed global optimization methods applied to hydrologic model calibration. He has shown that the Shuffle Complex Evolution (SCE) algorithm is the most efficient method for calibration. Tolson (Tolson & Shoemaker, 2007a) has mentioned that the SCE algorithm ((Duan et al., 1992) and (Duan et al., 1993)) has been the most cited method for the past 10 years in this field. He has introduced a new optimization algorithm called a Dynamically Dimensioned Search (DDS) algorithm (Tolson & Shoemaker, 2007b) and he has extensively compared this method with SCE. These comparisons show that DDS outperforms SCE with about 80% fewer model evaluations. In this study, DDS has been applied to WATCLASS calibration.

Dynamically Dimensioned Search DDS is a greedy-type optimization algorithm that always keeps and starts from the best points identified so far. The dimension of the search in the neighborhood varies but the variance of perturbation stays constant and the number of varying decision variables decreases as the number of function evaluations approaches its maximum. This method is based on the idea that all of the variables do not have similar impact on the objective function, so it randomly selects the set of perturbing variables. The detailed discussion of this method along with its algorithm is available in (Tolson & Shoemaker, 2007b) and the code can be downloaded from

`http://www.civil.uwaterloo.ca/btolson/software.htm.`

2.2.5 Surrogate Model Optimization (SMO)

In general, most of Global Optimization Algorithms tend to be computationally expensive, sensitive to the initial point and not easily parallelized. For this reason, the direct optimization of computationally expensive models using Global Optimization Algorithms is impractical. As a result, researchers become interested in a less computationally intensive strategy to optimize expensive models. A less expensive strategy is an optimization strategy that optimizes a cheap ”surrogate” model which is constructed on a limited number of original-expensive-model simulations. This optimization technique is called Surrogate Model Optimization (SMO). SMO refers to optimizing a mathematical/soft model called a surrogate model instead of the original costly model. For instance, a neural-network or fuzzy model identification can be used to construct a model based on simulated points (Kamali et al., 2005). This technique is considered to be a practical strategy for optimizing computationally expensive, black-box models.

SMO is based on the assumption that all or part of the so-called expensive model, which is produced by simulation, can be generated by a less computationally expensive surrogate model. Since the optimization process relies on the surrogate model, finding the optimum does not involve a high computational cost.

The SMO process is usually iterative and in each iteration the surrogate model becomes more complete and closer to the simulated model. This surrogate model scouts the region of simulation for a minimum. This search is biased by surrogate model and is different from doing blind simulation. Choice of a suitable surrogate model is an important step, which usually needs field expert knowledge.

The general procedure for the construction of a surrogate model is explained in algorithm 1. Surrogate model optimization can be done in many different ways, yet with the same bases.

Algorithm 1 Surrogate Optimization general procedure

```
1: Randomly generate samples of the parameters.
2: Find the corresponding model output for the sample.
3: Construct the surrogate model based on known data points.
4: Find the optimum of the surrogate model.
5: Find the original model results corresponding to surrogate model predicted
   optimum.
6: Compare surrogate model results and the original model results. If they are close
   enough the optimum of the surrogate model is the original model optimum and the
   surrogate model is complete. Terminate the process, otherwise continue.
```

SMO search can focus either on local search or global search. In local SMO, the model is constructed around the best point in the region of interest (Myers & Montgomery, 1995). These approximations are usually only valid in a small neighborhood of the best point. Global SMO focuses on the trend and behavior of the model in the entire domain. This approach is very similar to curve fitting, where a model with only a small number of original model calls is constructed.

The use of a surrogate model for optimization dates back to 1964 (Kushner, 1964). Different approaches of SMO have been introduced in literature. Chen (Chen et al., 2006) has offered a broad review of SMO and its applications. Some of the recent applications are: evaluation of circuit performance with CAD/CAM simulators with consideration of manufacturing uncertainty by (Koziel & Bandler, 2007), optimal design of electromagnetic actuators by (Encica et al., 2007), 2D turbomachinery flow by (Peter et al., 2007) and shape optimization by (Marsden et al., 2004). SMO technique has recently been introduced in hydrology. Some of the SMO applications in hydrology are: calibration of groundwater bioremediation models by Mugunthan *et al.* (Mugunthan et al., 2005), assessment of parameter uncertainty in groundwater models (Mugunthan & Shoemaker, 2006), stochastic management of pump-and-treat by (Bau & Mayer, 2006) and calibration of WATCLASS hydrologic model using Fuzzy-TSK surrogate model (Kamali et al., 2005). All these papers endorse the efficiency of using SMO in optimization of expensive models.

SMO popularity comes with its various approaches. Some of the most common types of SMOs are: the multivariate polynomials developed by ((Box & Draper, 1987) and (Myers & Montgomery, 1995)), the radial basis function by ((Buhmann, 2003), (Powell, 1992) and (Powell, 1999)), regression splines by (Friedman, 1991) and kriging models by ((Sacks et al., 1989) and (Cressie, 1993)). Some SMO approaches are more popular than others. For instance, using kriging in the optimization of expensive functions by((Schonlau et al., 1998) and (Sacks et al., 1989)) are considered to be among the most cited papers.

The first step in every SMO is an appropriate surrogate model (or response surface model). It is possible to involve more model information, such as derivatives (Stephen et al., 2004) in the construction of surrogate models, however in this thesis only model output (model response) is considered. A surrogate model is the main tool for discovering the model space and searching for the location of the optimum.

In this document, three different approaches for construction of surrogate models are discussed.

The first approach is based on the construction of a radial basis function regression model and a polynomial error function. This method was initially introduced by Gutman (Gutmann, 2001b) and is called Gutman-RBF. This method has been adopted for use in distributed computing frameworks and the adopted version has been employed for the calibration of WATCLASS. In the meantime, another version of RBF called stochastic RBF that shares some similar features with what we have previously proposed (Kamali et al., 2005) has been introduced by Regis *et. al.* (Regis & Shoemaker, 2007b). Regis stochastic

RBF, unlike our adopted version, has a distance measure for finding the optimum point.

In the second approach, a surrogate model is constructed based on the idea of correlated errors, using the kriging concept. This method has been developed by Sacks *et al.* (Sacks et al., 1989) and is called Design and Analysis of Computer Experiments (DACE). This method is one of the SMOs that has been employed for optimizing our test functions and has been modified to work with distributed computing facilities for the calibration of WATCLASS.

A novel approach to SMO is discussed and a new method is proposed in the third section. This method is developed to overcome the disadvantages of previously mentioned SMOs. This approach is constructed to serve our belief that this model exhibits a certain type of smoothness. Thus, a penalty function to regularize (or smooth) this model is defined and has been added to the least square problem. This new method is called Regularized Function Map (RFM) and is explained in section 3.2.1. It is expected that this penalty function enhances the prediction of the model and therefore decreases the computational cost of finding RFM optimum compared to DACE and RBF.

Radial Basis Function Approximation (RBF)

In this section, Radial Basis Functions (RBF) perform the function approximation task in the SMO technique. RBF is a general term that has been used in different disciplines with slightly different meaning. In the context of SMO, the term RBF stands for the RBF interpolant. In this document, when we use RBF we mean the RBF interpolant. A detailed description of this method can be found in (Gutmann, 2001a).

The RBF response surface model is the most commonly used surrogate model in the environmental modeling field. Application of this method for global optimization has been initially proposed by Gutmann (Gutmann, 2001b) and later on it was improved and applied to the environmental modeling field by Regis *et al.* in ((Regis & Shoemaker, 2007a) and (Regis & Shoemaker, 2005)). To understand the basis of this technique, a brief mathematical overview of this method follows.

Theoretically, the RBF algorithm is designed for a continuous function f on a compact space, which is by definition a subset of Euclidean space $\mathbb{R}^n$ that is closed and bounded. For n input vector data points $x^1, \cdots, x^n$ and their corresponding function values, the interpolant $f(x)$ is defined as:

$$f(x) = \sum_{i=1}^{n} \lambda^i \phi(\|x - x^i\|) + p(x) \tag{2.3}$$

where $\lambda^i \in \mathbb{R}$ and $p \in \prod_m^d$, $\prod_m^d$ is defined as the polynomials of degree at most m with d variables. $\|.\|$ denotes the Euclidean norm and ϕ is a basis function. A linear form of polynomial error function, $b^T x + a$, is employed in this document, therefore, the interpolant simplifies to:

$$f(x) = \sum_{i=1}^{n} \lambda_i \phi(\|x - x^i\|) + b^T x + a \tag{2.4}$$

A few options for basis function, ϕ, are as follows:

- linear, $\phi(r) = r$
- Thin plate spline, $\phi(r) = r^2 logr,\ r > 0$; and $\phi(0) = 0$
- Cubic, $\phi(r) = r^3$
- Multiquadratic, $\phi(r) = \sqrt{r^2 + \gamma^2}$
- Gaussian, $\phi(r) = e^{-\gamma r^2},\ r \geq 0\ and\ \gamma$ is a positive constant

To solve this interpolation problem, some definitions are necessary:

- Φ is a $n \times n$ matrix whose elements are $\Phi_{ij} := \phi(\|x^i - x^j\|)$ where $i, j = 1, \cdots, n$
- $F = (f(x^1), \cdots, f(x^n))^T$
- $\Lambda = (\lambda^1, \cdots, \lambda^n)^T \in \mathbb{R}^n$

The first requirement for finding proper coefficients of this interpolant is to have the right side of Eq. 2.4 equal to the function outputs or $y(x^i) = f_i$. However, this condition does not guarantee a unique solution (Gutmann, 2001b). To guarantee a unique solution, it is possible to take advantage of previously proven relations by Powell (Powell, 1992) and solve the following system:

$$\begin{cases} y(x^i) = f^i & i = 1, \cdots, n \\ \sum_{i=1}^{n} \lambda^i x^i = 0 \end{cases}$$

or

$$\begin{cases} F = \Phi\Lambda + B^T X + A \\ X^T \Lambda = 0 \end{cases}$$

where B is the first order coefficients of the polynomial, A is the zero order coefficients of the polynomial and $c = [B^T\ A]^T$ which is the vector of polynomial coefficients.

This system can be cast in the form of matrix relations, where $P = [X\ 1]$:

$$\begin{pmatrix} \Phi & P \\ P^T & 0 \end{pmatrix} \begin{pmatrix} \Lambda \\ c \end{pmatrix} = \begin{pmatrix} F \\ 0 \end{pmatrix}$$

The only necessary condition for the coefficient matrix to be invertible is to have $n \geq 1$. If this condition is satisfied at the beginning of the process, the coefficient matrix remains invertible when newer points are added to the previous points. If at the beginning of the process n data points are available then by proceeding further new sample points will be added to the set which makes the sample size exceed n.

There are few suggested strategies for finding the minimum of the RBF SMO. Since RBF is a parametric function, it is not possible to find its optimum through a closed form formula. Some previously proposed methods such as multi level single linkage (MLSL) algorithm by Kan *et al.*(Kan & Timmer, 1987) or Pattern Search algorithm by Torczon (Torczon, 1997) have been employed for optimizing RBF. Gutmann (Gutmann, 2001b) has also employed a utility function that has been originally proposed by Powell (Powell, 1992). This optimization approach is based on the minimization of a bumpiness criterion which is only applicable to thin plate splines, linear basis functions and cubic basis functions. In this method, Gutmann minimizes a predefined roughness penalty, or simply a measure of bumpiness. The details of this method is not of interest here and interested readers are referred to Schaback for more information and proofs on this measure (Schaback, 1993).

The performance of this utility function has been questioned by many researchers. For instance, it requires the choice of too many non-trivial parameter values or it converges slowly to the global optimum due to lack of a complete local search (Regis & Shoemaker, 2005). Regis has suggested a better global search strategy by restricting the algorithm in the subregion of current iteration by searching for a point with minimum function value, but not very close to the previously evaluated points (Regis & Shoemaker, 2005). Moreover, in the case of inadequate progress in each subregion a restart strategy is adopted (Regis & Shoemaker, 2007a). Examining these strategies still does not guarantee the convergence to the global optimum, however, the new adopted technique results in a better performance in some test problems (Regis & Shoemaker, 2007a). Regis *et al.* (Regis & Shoemaker, 2007b) have proposed a new stochastic optimization strategy which is less complicated than the Gutman-RBF suggested technique (Gutmann, 2001b). In this stochastic optimization strategy, the next optimum is found based on two criteria. One criterion relates to the distribution of the points in the neighborhood of the best point and the second criterion relates to their distances from that point. A weight factor between zero and one is assigned to each of these criterion.

To overcome the shortcomings of the suggested optimization processes of RBF, a new heuristic approach for optimization of RBF that is suitable for a distributed computing facilities is proposed and its performance is compared with Gutman-RBF on some test functions. The details for the proposed method are presented in chapter 3.

Surrogate Model Optimization using Kriging (DACE)

The Design and Analysis of Computer Experiments (DACE) approach ((Sacks et al., 1989) and (Schonlau et al., 1998)) suggests a framework for finding the optimum point of a computationally expensive model.

This model is based on the assumption that observations are coming from a model that contains two parts: regression part that explains the general behavior of the data; and error part that covers the small discrepancies between the data and model output. The formula of such model is:

$$y(x^i) = \sum_{h=1}^{k} \beta_h f_h(x^i) + \epsilon^i \qquad (i = 1, ..., n) \tag{2.5}$$

In this equation, n is the number of input data,

$f_h(x)$s are linear or nonlinear basis functions of x, for example, polynomial basis functions,

β_hs are the basis function corresponding coefficients such as polynomial coefficients,

and ϵ^is are errors which are assumed to be normally distributed variables. The errors are correlated and have zero mean and σ^2 variance. They are spatially related and their correlation is a function of their distances. The correlation between the error of two adjacent points is defined as a function of their distances (see Eq. 2.6). The errors in Eq. 2.5 are defined to follow a special form of a stationary process. In stationary process, the mean and variance of the variable are constant but they are spatially correlated. The difference between this modeling process and regression lies in the assumption of correlated errors which do not hold in ordinary regression. The main reason for assuming correlated errors in this process stems from the nature of errors in the computer codes, which are more related to modeling error than measurement error or noise (Schonlau et al., 1998).

The idea of estimating the value at a point using data in its neighborhood was initially proposed by Krige (Krige, 1951). In this process, it is assumed that the error of modeling is following Krige concept. It is supposed that the correlation between two errors, ϵ^i and ϵ^j, is a function of their distances but not necessarily Euclidean distance. The function formulation is:

$$Corr(\epsilon^i, \epsilon^j) = exp[-d(x^i, x^j)] \tag{2.6}$$

In equation 2.6, d is a distance function of two points, x^i and x^j, and is defined as:

$$d(x^i, x^j) = \sum_{h=1}^{k} \theta_h |x_h^i - x_h^j|^{p_h} \qquad (\theta_h \geq 0\ ,\ p_h \in [1, 2]) \tag{2.7}$$

In Eq. 2.7, θ_h and p_h are two sets of parameters that needed to be adjusted. θ_h is the parameter that controls the weight of the distances of two variables and is basically a scaling factor. p_h is the parameter that controls the smoothness of the function in h direction. p_h can be either 1 or 2. In $p_h = 1$ higher correlation between points far from each other is considered compared to $p_h = 2$ (see Eq. 2.7). For more discussion on effects of different values of θ_h and p_h on correlation refer to (Schonlau et al., 1998).

Considering these two parameters for each dimension, each model will have at least

$2k$ parameters that need to be tuned. In addition to that, there are basis functions and their coefficients (see Eq. 2.5) that add more parameters to this model.

One way to investigate if it is necessary to choose a model with too many parameters is starting with applying a Krige model only to the residual and if the results were not satisfactory then move to a more complicated system. In this type of model, the output is simply generated by a model with a mean of μ with correlated errors with variance of σ^2 (see Eq. 2.8). Experiments by Schonlau *et al.* (Schonlau et al., 1998) demonstrate that the treatment of errors in this way makes the model very flexible and capable of modeling very complicated functions .

Our preliminary experiments also suggested that using a simple model with Kriging on the residual is sufficiently fine and there is no need to use a more complicated model. Hence, the model that we employed is:

$$y(x^i) = \mu + \epsilon^i \qquad (i = 1, ..., n) \tag{2.8}$$

μ is the mean of the response at known sample points x and ϵ^i is the modeling error that are correlated as described in Eq. 2.6 and Eq. 2.7.

One way to estimate all model parameters, $\theta_1, \cdots, \theta_k, p_1, \cdots, p_k$, μ and σ^2, is using maximum likelihood idea. The likelihood function of y is presented in Eq. 2.9. Thus, in order to compute the statistical model parameters, μ and σ^2, the likelihood function indicated in Eq. 2.9 should be maximized with respect to μ and σ^2.

$$\frac{1}{(2\pi)^{n/2}(\sigma^2)^{n/2}|R|^{1/2}} exp[-\frac{(y - 1\hat{\mu})' R^{-1}(y - 1\hat{\mu})}{2\sigma^2}] \tag{2.9}$$

Where R is the covariance matrix and 1 stands for a unit vector of size n. The solution of this optimization is:

$$\hat{\mu} = \frac{1'R^{-1}y}{1'R^{-1}1} \tag{2.10}$$

and

$$\hat{\sigma}^2 = \frac{(y - 1\hat{\mu})'R^{-1}(y - 1\hat{\mu})}{n} \tag{2.11}$$

Eq. 2.10 and 2.11 provide necessary terms for computation of output prediction, $\hat{y}$. The prediction for the unseen point x^* is shown in Eq. 2.12:

$$y(x^*) = \hat{\mu} + r^{*'}R^{-1}(y - 1\hat{\mu}) \tag{2.12}$$

In Eq. 2.12, $\hat{\mu}$ is the estimated mean using known points. The second term contains the information about known points covariance matrix R times the error in the observed points. r denotes the vector of correlation between x^* and previously sampled data (one element of this vector, r_i^*, is defined as $r_i^* = Corr(x^*, x_i)$).

θ and p are the two parameters that explain how the objective function typically behaves. Although the DACE model is derived under the assumption that these parameters are known, in practice they are unknown and should be estimated. Sacks *et al.* have provided a complete explanation of computing these parameters using maximum likelihood and cross-validation (Sacks et al., 1989). Schonlau *et al.* have also mentioned that their so-called sleight of hand for parameter estimation did not have any serious impacts on the result (Schonlau et al., 1998). However, replacement of known parameters by estimated parameters might have two minor consequences. First, there is a possibility of underestimation for small size samples and second, after adding each new sample point all of the process parameters should be re-estimated (Schonlau et al., 1998).

In this thesis, some of DACE codes were downloaded from the DACE website at `http://www.stat.ohio-state.edu/~comp_exp/`. It provides the codes for computing all of these parameters. The only issue with this software was the definition of a lower and upper limits for θ which sometimes caused different values for θ when the limits were alternating, this phenomenon is been also mentioned by other researchers such as (Kleijnen, 2007). We fixed the parameter values based on the results of model validation. In terms of CPU time, the process of DACE parameter estimation was a bottle neck of DACE model construction where in higher dimensional cases it became prohibitively slow.

Here, we only presented the Kriging the residual model, which is considered to be one of the simplest in DACE model category. However, it is possible to construct an even more complex model by replacing μ with a basis function such as polynomial. Replacing a basis function in place of μ will result in having a more complicated model and hence estimating more model parameters.

The DACE model is a computer experimental design tool and one part of any experimental design is the strategy for sampling. The sampling strategy that was employed in DACE was Latin Hypercube Sampling(LHS) which has been proposed by (McKay et al., 1979). This method belongs to the category of Stratified Random Sampling where the samples are randomly taken from non-overlapping strata. LHS is the sampling strategy that has been employed in this thesis and is illustrated in section 5.2.1.

The main reason for describing the DACE model is to use it as a surrogate model whose optimum point conforms to the original simulations. Hence, the next process after sampling and model construction is finding the optimum of the DACE model. Schonlau *et al.* (Schonlau et al., 1998) have employed a strategy known as Expected Improvement (EI) strategy. EI criterion is basically a measure that balances the uncertainty embedded in the prediction of new sample points and the best predicted value. This algorithm searches the model space for a better objective function value considering the uncertainty involved in function predictions. We will not deal with the details of this strategy, however, interested readers can refer to (Schonlau et al., 1998) for more information.

The main drawback of EI that has been also pointed out by Huang is imposing a large computational burden on the process (Huang et al., 2006). The main reason for this high computational cost is calculation of the expected value for the function prediction. This calculation cost increases as the number of evaluated points increases. This process high computational cost is the main limitation of this model which has caused a limited number of applications of this model in the literature (Huang et al., 2006).

To overcome this limitation and to adopt DACE algorithm for distributed computing with reduced computational cost, the new heuristic search method for finding DACE optimum point is adopted for this technique and is explained in Chapter 3.

2.3 Sensitivity Analysis

2.3.1 Introduction

Sensitivity Analysis (SA) is another main process for exploration of a model and its interaction with its parameters. It is more a subjective issue than an objective one, in other words, the modeler based on her requirements from the model develops or chooses a specific SA approach.

There are different scopes and definitions for SA. SA can refer to the process of identifying contribution of each of the model parameters/inputs or combination of them on model output. It also might be employed for assessment of model predictions, model structure or parameters intervals.

In general, SA term refers to the process of discovering the importance of each/couple of the parameters in a model in order to serve a specific task, some of them mentioned previously and some of them are model understanding, pre-processes reducing and time and computational cost saving. Modelers conduct SA to discover facts about their models. Some of their questions that have been mentioned by Hamby (Hamby, 1994) are as following:

- Which parameter has no/little impact on model uncertainty/output?
- Which parameter contribution is significant on output variability/variance?
- Which parameters are correlated with each other?
- Which parameter/s are correlated with the output?

In addition to these questions, one might think of SA as a way to overcome the high computational cost of model optimization (calibration for our case study) where lower computational cost results in a more manageable model.

SA has become a very common practice in modeling since there are many potential benefits of it. Regarding the goal SA is serving, different SA methods have been developed which might deal with one aspect more than the other.

For a model in its validation stage, such as WATCLASS, it is necessary to use some SA techniques to discover the probable interaction of some parameters, impact of parameters on model output and changes of model output due to small parameters perturbations. Sometimes, unexpected interactions of two model parameters can reveal an influential coding problem. However, these are not all of the benefits of SA. In section 3.3.1, we will explain how by reducing the dimension of model the computational cost of calibration process was reduced.

Sensitivity Analysis methods can be categorized in many different ways. One way is dividing SAs in terms of their spatial approaches for exploring model space as: Local

Sensitivity Analysis methods and Global Sensitivity Analysis methods which are explained in section 2.3.2 and section 2.3.3. In addition to that, different categories of SA techniques are introduced and some of the methods in each category are presented.

2.3.2 Local Sensitivity Analysis

In Local Sensitivity Analysis (LSA) methods, sensitivity of the model is evaluated based on the output gradient with respect to the parameters in the vicinity of a nominal value. LSA is useful when sensitivity of the model with respect to slight changes in the parameter set is required. For instance, first-order and second-order derivatives are techniques that evaluate the sensitivity of a model to its input/parameters locally.

LSA is a common sensitivity test for cases such as material fatigue models (Castillo et al., 2007), mechanical models, the risk evaluation of a system (Borgonovo & Peccati, 2004), or cases that sensitivity of the model around output mean is of importance.

2.3.3 Global Sensitivity Analysis

In this category of SA methods, the entire region of parameters is explored and the variation/uncertainty of the output is investigated. Uncertainty analysis of the input data is not of concern in this document, so we will not deal with the effects of model input uncertainty in the model output.

In general, GSA results are more reliable when a linear or slightly non linear model is investigated. Although, there are GSA methods that tackle global model variance or global output uncertainty analysis, their high computational cost limits them to low dimensional models.

Some of the common GSA methods are: Correlation analysis such as partial correlation analysis (Draper & Smith, 1981), multiple regression analysis, two and several sample tests, graphical methods and variance based methods.

There are at least two obstacles for performing GSA: i) The computational costs of such techniques are comparatively high compared to LSAs; ii) These types of methods give best results where the sensitivity of parameters in the entire input space stays constant. Therefore, GSA results of highly nonlinear models with non consistent behavior in their interval (such as step functions) do not contain useful information about the regions with more sensitivity.

2.3.4 Gradient based methods

One of the most well-known method in category of LSA methods is gradient based method, which explores the sensitivity of model output with respect to each of its parameters individually using gradients of the model in the region of interest. This type of methods mainly investigates model sensitivity in a specific interval. For non-linear models, the obtained sensitivity information does not generalize the sensitivity of the entire model. There are many gradient based SA methods and some of them are listed here.

One-Factor-At-A-Time(OAT)

The most commonly used SA method is the One-Factor-At-A-Time (OAT) (Saltelii et al., 2006). In OAT, one parameter varies at a time while other ones are kept unchanged (Box, 1954). The relative variation of the output with respect to the variation of each of the parameters shows the sensitivity of the model to that parameter. This method is an LSA method, since only a specific location of the parameter space is explored (the other parameters are kept constant at a nominal value such as their mean). This SA technique is very common as its computational cost is very low and it is easy to compute and implement. The formulation of this method is as:

$$d(x_i) = \frac{\dfrac{f(x_1, x_2, ..., x_i + \Delta x_i, ..., x_k) - f(x_1, x_2, ..., x_i, ..., x_k)}{f(x_1, x_2, ..., x_k)}}{\dfrac{\Delta x_i}{x_i}} \tag{2.13}$$

$d(x_i)$ is the model sensitivity at the parameter x_i. Clearly, parameters scaling scheme do not have any influence on the result, as the results are normalized. The main limitation of OAT is its inability to provide information about the interaction of parameters of the model. Since this method is locally concentrated on a specific point, the space it explores is very tiny. For linear or approximately linear models, it is considered a useful tool of parameter screening. To come up with the shortcoming of this method, mainly not being applicable to non linear models, and to take advantage of its low computational cost, Morris (Morris, 1991) has suggested a method, which is an extension of OAT.

Morris method

Similar to OAT method, the Morris method numerically computes the partial differential of the model output f with respect to one of the parameters x_i, $\partial f/\partial x_i$, which shows the influence of x_i on f. Based on the Morris sensitivity test, the results of the interaction of x_i and f are categorized into the following:

- the effect of x_i on f is negligible if this partial value is zero for all values of x_i.
- the effect of x_i on f is linear and additive if this partial value is constant but not zero for all values of x_i.
- the effect of x_i on f is nonlinear if this partial value is a nonconstant function of x_i for all values of x_i.

In addition to that, if more than one parameter belongs to the third category, parameter interactions should be considered as well. Morris method's sensitivity measure, $EE(x_i)$, is a randomized OAT in which samples are randomly selected from p^k samples. Each of these p^k samples belongs to a space of k parameters, that has been divided into p levels. A term called the Elementary Effects(EE) for each parameter x_i is defined and computed as illustrated in Eq. 2.14:

$$EE(x_i) = \frac{f(x_1, x_2, ..., x_i + \Delta x_i, ..., x_k) - f(x_1, x_2, ..., x_k)}{\Delta x_i} \tag{2.14}$$

The Δx_i is randomly selected from a group of p^k samples. In order to compare EE_i of all parameters, x's should be normalized. EE method was later on improved by Campolongo (Campolongo, 2007), in which a different sampling strategy was adopted.

Once EE for different parameters and different parameter perturbations were obtained a distribution of EEs, F_i, can be obtained. The mean and standard deviation of F_i distribution corresponding to x_i, are determined from collected random samples. A large mean value for the distribution F_i shows high overall impact of parameter x_i on the output, while a large standard deviation shows parameters interactions or nonlinearity.

This method is among GSA methods which determine sensitivity of model parameters qualitatively by putting them in three categories:

- negligible
- linear and additive
- non-linear and involved in interactions with other factors.

This method needs $2rk$ runs, where k is the dimension of the parameter vectors and r is the size of the sample that F_i is constructed with. It is clear that for calculating each

EE, two model runs are needed. An EE sample of size r, needs $2r$ simulations for each dimension. Morris has suggested a more efficient sampling strategy in which only $r(k+1)$ runs are performed (Welch et al., 1992).

First-order and Second-order Taylor Expansion

Estimation of the mean and variance of the model output becomes more complicated when the model is nonlinear and non monotonic. Taylor Expansion is a common method for representing a function sensitivity by its derivatives at a single point. Using this expansion, it is possible to compute mean and variance of the model using its derivatives. If a model is smooth (which is the case in most physical models), Taylor expansion results allow modeler to evaluate the model at a point and compute model derivatives at that point and then generalizes the results for other region of the model space.

The details of First-order and Second-order Taylor Expansion is presented in Appendix .1. We have not used this method directly for sensitivity analysis, but the results that has been presented in chapter 3 show that by using our proposed method it is possible to compute mean and variance of the optimum implicitly.

2.3.5 Variable Ranking (Parameter Screening)

In this approach, parameters are ranked based on a specific criterion such as correlation of each or a couple of model parameters with the output. This method filters valuable parameters from not valuable parameters (sensitive from non-sensitive) and acts as a pre-processor which separates parameters under certain assumptions (e.g. orthogonality and independence). Simplicity and scalability are the main advantages of this method. This approach is very application dependent; if the main purpose of SA is reducing the variance, the parameter/parameters that have the least effect on the variance are omitted;

if a learning machine development is the goal, parameter/parameters that have least effects on modeling error will be omitted. In this approach, however, parameters interactions are not sufficiently analyzed and this is considered to be this method main drawback.

Kolmogorov-Smirnov test

The Kolmogorov-Smirnov (KS) test (Darling, 1957) is a statistical test for comparison of two underlying probability distributions without any assumption about the distributions. The KS test is among screening techniques that rank parameters based on a pre-defined criterion. This test is based on Monte-Carlo simulation and a goodness of fit criterion. Hence, this technique is sometimes categorized as a Monte-Carlo filtering method. Model outputs are classified based on a chosen criterion and two samples are generated, one set with members that satisfy the chosen criterion and the other one with members that do not. The difference of the empirical cumulative distribution functions of these samples is then calculated as:

$$D(X_i) = sup_{X_i} \|F_n(X_i) - F(X_i)\| \tag{2.15}$$

In Eq. 2.15, F is the cumulative distribution function and F_n is empirical distribution defined as $F_n(x) = \frac{1}{n}\sum_i^n I_{X_i \leq x}$ where $I_{X_i \leq x}$ is an indicator function. An indicator function is a binary function that takes only 0 and 1 values, $I_A(X) \rightarrow 0, 1$. This function is defined as:

$$I_A(X) = \begin{cases} 1 & X \in A \\ 0 & otherwise \end{cases}$$

If a sample becomes accepted with respect to the selected criterion it is put in the first distribution, otherwise it is put in the second distribution. D is a notion of distance that indicates the similarity of the distributions.

The goal is to investigate if two probability density functions are identical. The first hypothesis, H_0 hypothesis, is for a situation where two distributions are identical or: $F_n(X_i) = F(X_i)$. The other hypothesis, H_1 hypothesis, is for a situation where two distributions are not identical or: $F_n(X_i) \neq F(X_i)$.

Large values of D support the first hypothesis H_0 and suggests that X_i is a significant factor in the model output. Small values of D support the second hypothesis H_1 and implies that X_i is not an important factor. This Monte Carlo based method is a common SA technique in the environmental field. For instance, it has been applied to infiltration capacity model for investigating land surface parameters sensitivity by Demaria *et al.* (Demaria et al., 2007).

Regression-Based Methods

Regression based techniques are among screening methods. The most common regression based measure of sensitivity is Standard Regression Coefficients (SRC). These coefficients are calculated based on a linear regression of the model Monte Carlo simulation on the parameters. A parameter with a higher coefficient is ranked higher than one with a smaller coefficient.

The Pearson Correlation Coefficient is also a regression-based technique that measures the effect of model parameters on the model output by calculating the correlation between each of model parameter and model output. Another regression-based measure is Partial Correlation Coefficients (PCC), which are based on the linear regression model of all parameters except one and the output. The method measures the strength of association between model parameters and its output. SRC and PCC gave perfect results when model is linear with respect to its parameters. Moreover, the regression model should be a good representation of the model. The goodness of fit is measured by the model coefficient of determination represented in 2.17. If this coefficient is close to 1 the regression model is an acceptable representation of the parameters and the output, otherwise it is not reliable.

The basis of SRC is as follows:

The Monte Carlo sampling results are demonstrated in matrix M , which is $n \times k$ as following:

$$M = \begin{pmatrix} x_1^{(1)} & x_2^{(1)} & \dots & x_k^{(1)} \\ x_1^{(2)} & x_2^{(2)} & \dots & x_k^{(2)} \\ \dots & \dots & \dots & \dots \\ x_1^{(n)} & x_2^{(n)} & \dots & x_k^{(n)} \end{pmatrix}$$

In matrix M, each column is generated from its corresponding distribution. Here, it is assumed that all parameters are from zero-centered normal distribution where mean of the distribution is zero or $x_i \sim N(\bar{x_i}, \sigma_{x_i}) \quad \bar{x_i} = 0, \quad i = 1, 2, \cdots, k.$

Each of the rows of M matrix corresponds to one input parameters of one simulation. The output for this trial set is $y = (y^1, y^2, \dots, y^n)$. A regression model for the M matrix could be defined as:

$$y^m = b_0 + \sum_{i=1}^{k} b_i x_i^m + \varepsilon^m \tag{2.16}$$

In Equation 2.16, i is the selected parameter, m is the simulation row number which is 1≤m≤n, b_i is regression model coefficients and ε^m is the error term for the mth simulation. The coefficients are computed using least square technique and are then normalized.

SRCs are defined as: $\beta_{x_i} = b_i\sigma_{x_i}/\sigma_Y$ which indicates each factor contribution in model variance. As mentioned, coefficient of determination is a factor that evaluates the goodness of fit of a linear model. This coefficient is indicated by R_Y^2 and is defined as:

$$R_Y^2 = \frac{\sum_{i=1}^{n}(y_{reg}^i - \bar{y})^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{2.17}$$

This factor shows the fraction of the model's output variability that is accounted for in the regression model and is $0 \leq R_Y^2 \leq 1$. An R value that is close to 1 indicates the closeness of the real model with the regression model. Usually, a regression model prediction is not reliable if its coefficient of determination is less than 0.6 (Campolongo & Saltelli, 1997).

SRC is easy to implement, understandable and reliable, however, it is unlikely that a simulation could be modeled by a linear regression precisely. In the case of a nonlinear but monotonic function, where the relation between model input and output is strictly non-decreasing or non-increasing over the entire interval of the input space model, the rank transformation is applied to the data.

In rank transformation technique, 1 is assigned to the smallest output and N is assigned to the largest output. The linear regression is applied to the rank transformed data and the new SRC are called Standardized Rank Regression Coefficients (SRRC). Following the same procedure, it is possible to transform Partial Correlation Coefficients (PCC) to Partial Rank Correlation Coefficients (PCC).

In the case of a non-linear and non-monotonic model the regression model is not reliable and it fails to produce reliable sensitivity information.

2.3.6 Variance based methods

These types of methods analyze the variance of the output to identify the most influential parameters (or parameter sets) affecting output variance. In this method the conditional variance of output with respect to a subset of input space is evaluated. In this method, the output variance is decomposed into contribution of one or couple of model parameters. These types of methods produce reliable results for both nonlinear and non-monotonic models. However, the process of variance decomposition is usually associated with a high computational cost as it is done using Monte Carlo simulations.

A well-known method in this category is the Sobol method (Campolongo et al., 1993). This method is a model-free SA method which is designed to accommodate all types of model assumptions. Using this technique, it is possible to compute the first or higher order effect of one parameter or several interacting ones with good precision. However, the computational cost of computing Sobol sensitivity indices is very high and it rises with the increase in the model dimensionality. For instance, evaluation of all sensitivity indices for a model with k parameters, gives $2^k - 1$ different patterns of interaction and to evaluate each of this patterns N, which is equal or greater than 1000 simulations are required. Even though there are literatures about cost reduction of this process such as r-LHS, it is still considered one of the most computationally intensive methods. Sudret (Sudret, 2008) has offered a new method using polynomial chaos expansions which reduces the computational cost of Sobol indices. He has used 10,000 simulations for evaluation of Sobol sensitivity indices of a 8 dimensional model, which is claimed to be 2-3 order of magnitude less computation than original Sobol method computation. He recommends to conduct $(k + 2)N$ simulations for computation of Sobol indices (see (Saltelli et al., 2000) and (Saltelli et al., 2005)). Hence, for a model with 16 parameters (e.g., WATCLASS with 16 parameters) 18,000 simulations and for a model with 44 parameters (e.g., WATCLASS with 44 parameters) 46,000 simulations are required (roughly, if each model run takes 10 minutes 125 computational days for 16 parameter model and 319 computational days for 44 parameter model are required.)

This high computational cost of Sobol method is the main reason for not computing Sobol sensitivity indices for the computationally expensive models including our case study.

2.3.7 Association based methods

These types of methods allow the user to investigate the interaction of any number of model parameters with the output and this interaction is not limited to linear associations (i.e. correlation). The nonlinear interaction is mostly discovered using entropy measurement, a term introduced in information theory field, and finds the mutual information shares between two variable distributions.

In this section, one method in the category of linearly associated methods and one in the category of nonlinearly associated methods are discussed.

Correlation Coefficient

Correlation Coefficient (CC) is a measure that provides information about linear relationship between each model parameter and output. This coefficient changes between -1 to 1 and is defined as:

$$CC(x^r, Y) = \frac{\sum_{i=1}^{n}(x_i^r - \bar{x}^r)(y_i - \bar{y})}{[\sum_{i=1}^{n}(x_i^r - \bar{x}^r)^2]^{1/2}[\sum_{i=1}^{n}(y_i - \bar{y})^2]^{1/2}}$$

Positive values of CC show that generally x^r and Y increase and decrease together and negative values of CC indicate that they change in opposite directions. In addition, values in close vicinity of 1 or -1 indicate a well-defined linear relationship, whereas values that are close to zero show no linear relation.

CC gives information about the linear relationship between two variables and does not offer any information about probable nonlinear relationship of two parameters. Salltelli *et al.* (Saltelli et al., 2004) provides examples of linear/nonlinear relationships and demonstrate that a perfect nonlinear relationship between a model parameter and output may sometimes result in CC values close to zero.

Entropy tests for grid-based sampling

When the underlying relationship between input and output is nonlinear, correlation methods fail. Saltelli *et al.* (Saltelli et al., 2004) in his book " Sensitivity Analysis in Practice" has illustrated this failure by a simple example. This example shows the limitation of first order sensitivity analysis based on variance, however it is clear that in the calculation process of the second order this effect will be discovered.

In contrast, entropy-based sensitivity index is not limited to linear relation. Entropy test is a probabilistic method for measuring the information content of one variable or interaction of two variables. In general, this quantity is defined as:

$$H(X) = -\sum_{i=1}^{n} p(x_i)ln(p(x_i))$$

where H is entropy associated with X and p is probability mass function. For continuous variables, where p is not a discrete function, $H(X)$ can be calculated using a classification strategy. Hence,

$$H(Y) = -\sum_{i=1}^{nD}(nD_i/N)ln(nD_i/N)$$

where H is entropy associated with Y, N is number of samples, nD is number of classes that divide Y and nD_i is number of members of class i, where $i = 1, \cdots, nD$.

The entropy of each parameters equals:

$$H(X_i) = -\sum_{c=1}^{nI}(nI_c/N)ln(nI_c/N)$$

where nI is the number of classes that divide X_i, nIc is number of members of class c, where $c = 1, \cdots, nI$.

The entropy associated with X_i and Y equals to:

$$H(X_i, Y) = -\sum_{i=1}^{nD}\sum_{c=1}^{nI}(nO_{ic}/N)ln(nO_{ic}/N)$$

where nO_{ic} is the number of point in both class c and class i.

To measure the strongness of the associations between Y and X_i a normalized entropy based uncertainty coefficient is defined by (Helton et al., 2006) as:

$$U(Y, X_i) = 2[H(Y) + H(X_i) - H(Y, X_i)]/[H(Y) + H(X_i)]$$

By calculating entropy based uncertainty coefficients, it is possible to discover the interaction of model parameters with output. Moreover, this process does not require high computational efforts.

2.4 Summary

This chapter presented an overview of some of commonly known Calibration and Sensitivity Analysis techniques and their differences. In the first part, different calibration techniques and their advantages and disadvantages are explained. This is followed by our main motivations for employing a surrogate model optimization approach. Two important techniques of surrogate model optimization are explained and their drawbacks are pointed out.

In the second part of this chapter, sensitivity analysis and its different categories are presented.

Chapter 3

Methodology

3.1 Introduction

This Chapter contains two main parts. The first part of this chapter starts with the details of the proposed method and motivation for its development. The model is applied to selected test functions and the results are compared with those from two widely used SMO techniques, RBF and DACE. Then, the necessary modification on three methods to make them suitable for the distributed computing framework is explained.

In the second part of this chapter, a new approach for conducting sensitivity analysis is presented. This new approach facilitates calibration by reducing the dimension of the model and thus reducing the computational burden of the calibration process.

3.2 Calibration

3.2.1 Regularized Function Map (RFM)

The main advantage of DACE and RBF surrogate models is that they provide near global optimum with lower computational cost compared to global optimization methods. However, they have many model parameters that require tuning for achieving near optimum results. The process of model parameter adjustment is iterative. This process involves the computation of a validation error (is explained in Appendix .2) for each set of model parameters and find a set that yields least validation error. RBF coefficients are computed by matrix inversion, which is not considered costly, however, for constructing a RBF surrogate model with acceptable precision a large number of training data is required. DACE model, on the other hand, is not demanding in terms of the number of training data, but its model construction is very costly since it involves an iterative process for the estimation of model parameters.

In DACE, the optimum point is found by computing the expectation of a distribution of unknown points. The difficulties associated with this prediction are pointed out in (Huang et al., 2006). For RBF, not only a large number of training data is required but also the suggested optimization method by Gutmann (Gutmann, 2001b) is not applicable to a number of suggested basis functions (such as multiquadratic and Gaussian). As a result, DACE due to its slow iterative procedure and RBF for its demanding number of training data are not considered the desired SMO for this case study.

Our goal is to propose a surrogate model with fewer model parameters that requires smaller number of training data. Plus, if the proposed surrogate model can be cast into a convex form having the global optimum, the low computational cost would be guaranteed and its optimization process would be straightforward.

The proposed surrogate model is based on the least square minimization concept. This concept is a well known technique for curve fitting, but it is known to be very sensitive to outliers. To overcome this sensitivity, a smoothness constraint is added to the least squared problem. The smoothness constraint controls the “jumps” of the function caused by the existence of outliers or noise.

Thus, the least square minimization is combined with the smoothness constraint. This smoothness constraint is inspired from the work of (Belkin et al., 2003) on graph regression. The smoothness is in the form of coefficients that keeps the function values of a neighborhood similar.

To formalize this model, imagine $\hat{y}$ is a fit on a vector of the output y. Based on what has been mentioned, the two necessary conditions for $\hat{y}$ to be a smooth and the best linear unbiased fit over y are:

- The least square cost function should be minimized to guarantee that the fit function predicts the output as good as possible,
- A smoothness constraint is applied to impose a heavy penalty when function values of two adjacent points are very different from each other. An adjacency weight is assigned to each set of points in the same neighborhood.

These two conditions can be formalized as these optimization problems:

$$\min_{x} \quad \sum_{i}(y_i - \hat{y}_i)^2 \tag{3.1}$$

and

$$\min_{x} \quad \sum_{i,j} W_{ij}(\hat{y}_i - \hat{y}_j)^2 \tag{3.2}$$

In Eq. 3.1 and Eq. 3.2, y_i is ith output corresponding to $x^i = [x^i_1, ...x^i_k]$ vector of input variable, $\hat{y}_i$ is the model prediction for y_i, W is the weight matrix and its elements, w_{ij}, are functions of the Euclidean distances of two vectors, x^i and x^j. The weight matrix works as a bound between the outputs and their corresponding inputs, the closer the points the higher is the corresponding weight. The involvement of the weight into the optimization problem will cause a smoother model. This weight matrix imposes the condition that two points in the same neighborhood could not have very different function values.

A linear model over the target data is defined as $\hat{y} = X^T\beta$, where X is matrix of input variables and y is vector of output variables. As explained previously, to have the best smooth linear map, there must be two considerations. To minimize Eq. 3.1, the square errors between prediction and observation should be minimized as shown in Eq. 3.3:

$$\begin{aligned} \sum_{i=1}^{n}(y_i - X^{i^T}\beta)^2 &= (Y - X^T\beta)^T(Y - X^T\beta) \\ &= Y^TY - 2\beta^T XY + \beta^T XX^T\beta \end{aligned} \tag{3.3}$$

Similarly, to have a smooth fit, Eq. 3.2 should be minimized as shown in Eq. 3.4:

$$\begin{aligned} \sum_{i,j}(y_i - y_j)^2 W_{ij} &= \sum_{i,j}(y_i^2 + y_j^2 - 2y_iy_j)W_{ij} \\ &= \sum_{i,j}(y_i^2 W_{ij}) + 2\sum_{i,j}(y_j^2 W_{ij}) - 2\sum_{i,j}(y_iy_jW_{ij}) \\ &= \sum_{i}(y_i^2 D_{ii}) + \sum_{j}(y_j^2 D_{jj}) - 2\sum_{i,j}(y_iy_jW_{ij}) \\ &= 2\sum_{i}(y_i^2 D_{ii}) - 2\sum_{i,j}(y_iy_jW_{ij}) \\ &= 2Y^T(D - W)Y \\ &= 2Y^TLY \\ &= 2\beta^T XLX^T\beta \end{aligned} \tag{3.4}$$

where L is defined as $D - W$ and is called matrix Laplacian, D is a diagonal matrix defined as: $D_{ii} = \sum_j W_{ji}$ and $D_{ii} = 0$ if $i \neq j$ and $w_{i,j}$ is defined as $w_{i,j} = e^{-\|X^i - X^j\|}$.

Eq. 3.3 and 3.4 are combined a follows:

$$\phi = (1-\alpha)(Y^TY - 2\beta^TXY + \beta^TXX^T\beta) + \alpha 2\beta^TXLX^T\beta \tag{3.5}$$

The cost function ϕ has two competing terms. The first one ensures that our estimate $X^T\beta$ is as close as possible to the observation Y while the second term does not allow very different function value estimation for points in the same neighborhood and hence guarantees smoothness. α is a parameter that is $0 \leq \alpha \leq 1$ and controls the contribution of each term in the fixed estimation. $\alpha = 1$ puts all the stresses on the smoothness and ignores the first part of ϕ. $\alpha = 0$ ignores smoothness and minimizes the sum of square errors.

To minimize the cost function ϕ with respect to β, it is required to have $\frac{\partial\phi}{\partial\beta} = 0$. This derivative will lead to:

$$\begin{aligned} & -(1-\alpha)XY + (1-\alpha)XX^T\beta + 2\alpha XLX^T\beta = 0 \\ \Rightarrow\ & (1-\alpha)XY = (1-\alpha)XX^T\beta + 2\alpha XLX^T\beta \\ \Rightarrow\ & \beta = (1-\alpha)((1-\alpha)XX^T + 2\alpha XLX^T)^{-1}XY \end{aligned} \tag{3.6}$$

Therefore, the best linear smooth fit on the data based on α, X, L and Y is found. Thus, the only parameter that needed to be tuned in the model validation process is α. This method, however, has the potential to map more complicated data. Since number of model parameters (e.g α) is independent of model dimension, augmenting the data space dimension, k, into a higher dimension n will lead to take advantage of blessing of dimensionality and the nicer behavior of the data in the higher dimension (Gershenfeld, 1999). A candidate map can be a radial basis function, defined as:

$$\Phi = \beta^T e^{(-\|X^i - X^j\|_2/2\sigma^2)} \tag{3.7}$$

By the assumption that $\chi = e^{(-\|X^i - X^j\|_2/2\sigma^2)}$, then $\Phi = \beta^T\chi$. Thus, X has been mapped to χ. Performing a simple transformation makes it possible to use the previous derivations and to fit a radial basis function to the data by using equation 3.6 for computing β.

Now that the general form of RFM is explained, our approach for optimizing the special cases is presented. To evaluate the performance of proposed RFM methods, we primarily employ test functions known as Dixon-Szeg*ö* (Dixon & Szego, 1978). Subsequently, this method is applied to calibrate WATCLASS. The details of notation for these two case studies are explained in section 3.2.1.

Dixon-Szegő test function optimization: notation details

In cases where optimizing Dixon-Szeg*ö* test problems is the target, these functions are represented by Φ shown in Eq. 3.7. The minimum of the test function, X, is the same as minimum of Φ, since Φ approximately maps of the test function. Hence, each function input-output is modeled using Eq. 3.7, where Φ is substituted by the vector of output and X is substituted by the vector of input data. As a result, Φ represents the test function, once a proper number of inputs-outputs were fed to the Eq. 3.7. Therefore, the minimum of Φ is the same as the minimum of the test function. The lower and upper bounds of X are also considered in the optimization process.

WATCLASS calibration: notation details

In the case of WATCLASS calibration, Φ represents the Nash-Sutcliff (NS) criterion explained in Eq. 2.1. The target NS number is 1 which demonstrates a perfect model simulation. Hence, the corresponding parameter set, X^*, that yields a NS number equal to 1 is searched. However, in practice it is nearly impossible to reach to a NS equal to 1. In general, WATCLASS simulation result is considered acceptable if a NS number equal or greater than 0.7 is achieved. In this experiment, although we set the optimum to be 1 as shown in Eq. 3.8 but the optimization process was terminated once a result equal to or greater than 0.7 was obtained.

The rest of this section deals with the mathematical formalism for finding a set of parameters of WATCLASS, X^*, that yield near global optimum results. The constraints are WATCLASS constraints but the procedure is general and is easily adaptable for other optimization problems.

Before finding X^*, it is required to find χ^*. Furthermore, χ is an exponential function so, it adds two more constraints to the problem. These two constraints are $\chi > 0$ and $\chi \leq 1$. χ has to be valid with respect to these constraints. Thus, the problem is:

$$\Phi = \beta^T \chi = 1 \qquad and \qquad 0 < \chi \leq 1 \tag{3.8}$$

In Eq.3.8, the best χ^* that results in $\Phi = 1$ should be found. This is a linear programming problem and is solved using MATLAB linear programming tools. The answer of the optimization problem 3.8, χ^* , is a distance vector of size $1 \times n$ with the distances of all training points from the optimum point or

$$\chi^* = [e^{(-\|X^*-X^1\|_2/2\sigma^2)}, \cdots, e^{(-\|X^*-X^i\|_2/2\sigma^2)}, \cdots, e^{(-\|X^*-X^n\|_2/2\sigma^2)}]$$

The next step after finding χ^* is computing the corresponding X^* . This leads to Eq. 3.9:

$$\begin{gathered} \sum_{i=1}^{n} \frac{1}{\|X^* - X^i\|_2/2\sigma^2} = \frac{1}{ln\chi^*} \\ LB \leq X^i \leq UB \quad j = 1, \cdots, k \end{gathered} \tag{3.9}$$

where LB stands for X vector lower bound and UB stands for X vector upper bound.

To find X^* coordinates, a nonlinear optimization problem is solved, where $\|X^* - X^i\|_2 = \sqrt{(x_1^* - x_1^i)^2 + \cdots + (x_k^* - x_k^i)^2}$. Thus, X^* coordinates which are a vector of $1 \times k$ could be found using any common optimization technique. We found that an appropriate tool for this particular case study is the conjugate gradient method.

Alternatively, if for some reason the algorithm did not converge to the global optimum, the Multidimensional Scaling(MDS) method (Kruskal & Wish,) can be alternatively used. MDS is a classic dimensionality reduction approach that constructs a matrix of n points based on the information of their pairwise Euclidean distances. This technique was applied to the matrix of distances to find its minimum x^*. Interested readers are referred to (Kruskal & Wish,) for more details of this method.

In theory, this X^* is the vector that yields an NS value equal to one. This X should be inserted into WATCLASS and the real value of the NS number should be compared with the predicted one.

The main advantage of RFM over RBF and DACE is having a closed form formula. Once the approximation process completes, the optimum of the model could found in one iteration with very small amount of computational effort. Moreover, contrary to DACE and RBF, there is no assumption for output distribution in RFM.

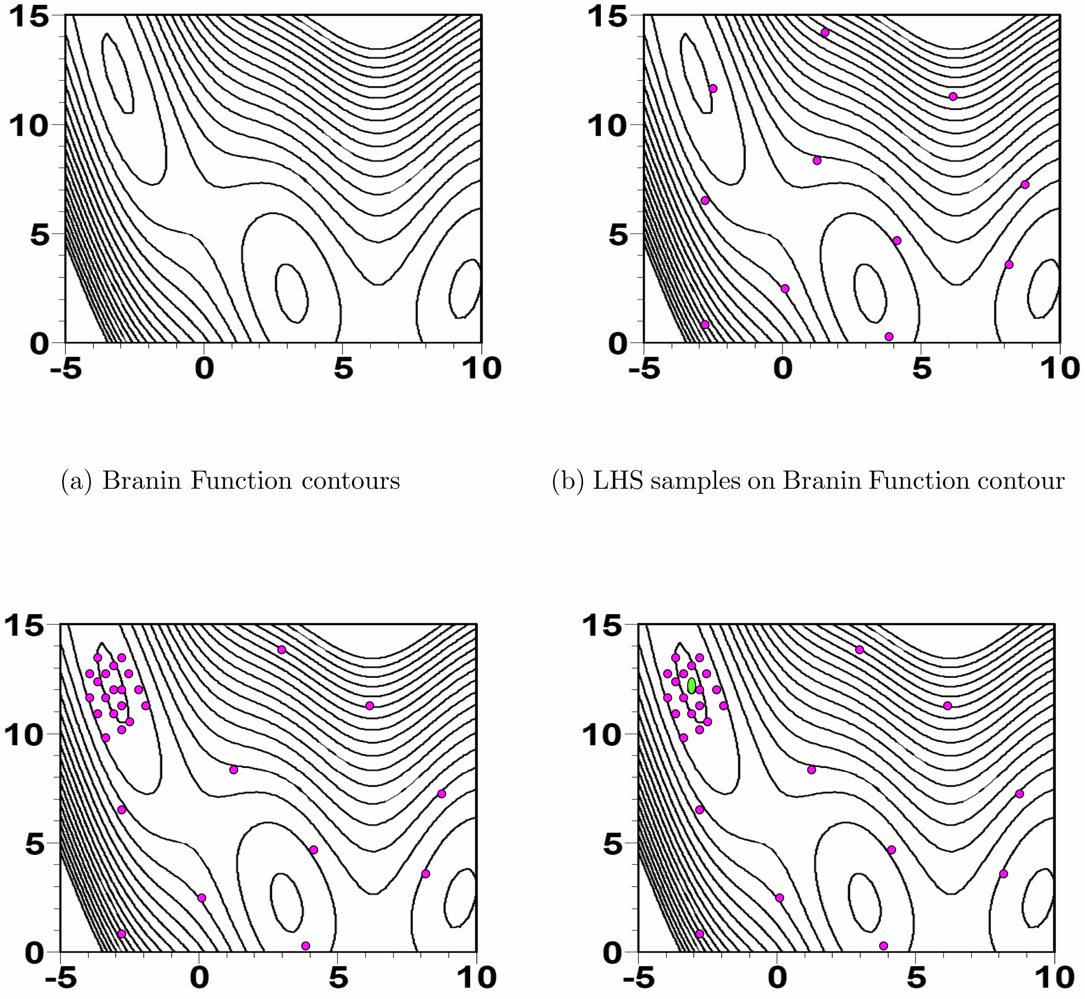
3.2.2 Comparison of RBF, DACE and RFM SMO Methods on test functions

A set of experiments was done to compare the performances of RFM, DACE and RBF. The proposed RFM method is tested on five Dixon-Szeg*ö* (Dixon & Szego, 1978) test functions presented in Table 3.1. The notation for optimizing these functions is explained in section 3.2.1. Each set of the optimization experiments was done in 30 trials and the mean of the number of function evaluations is reported. If one or more than one trials did not converge to the optimum after the maximum number of function evaluations was reached, the number of unsuccessful trials is reported as well. Table 3.2 compares the results of of SMO method for the mentioned test functions. A number of function evaluations for surrogate models is extracted from their corresponding publications ((Schonlau et al., 1998) and (Regis & Shoemaker, 2007a)).

RFM model construction started with 2 × k LHS generated training points. The number of training points is increased until an acceptable prediction error is obtained. The criterion for acceptable prediction error was the mean of the mean of prediction errors (the first mean is computed over one trial for one set of prediction errors and the second mean is computed for all 100 trials) for 100 different trials using 100 different sets of training data points. RFM parameters, α and σ, were adjusted with the same manner.

Figure 3.2.2 is schematic of the optimization procedure of RFM method. The contours of Branin function are shown in part (a). The results of LHS sampling is then shown on the Branin contours in part (b). RFM method will be constructed based on the results of this LHS samples and the minimum of the RFM is found. Part (c) illustrates the process of approaching minimum of the Branin Function in several iterations. As is shown, after couple of iterations, the SMO will concentrate on the region of optimum and searches for the optimum in a small region.

Table 3.2 shows that all of the three surrogate optimization techniques could efficiently find the optimum. DACE model needed comparably fewer function evaluations to converge, however its model construction process was more time consuming than the RFM and RBF. This slowness might stem from the process of the DACE model construction which is an iterative process compared to two other methods that only deal with the matrix inversion process. The other drawback of DACE method is the sensitivity to dimensionality as discussed in section 2.2.5. The DACE model construction process



(a) Branin Function contours

(b) LHS samples on Branin Function contour

(c) RFM method search focuses on one region

(d) The minimum of the Branin function is achieved

Figure 3.1: Schematic approach of RFM method

Table 3.1: The Dixon-Szegő test functions

Test function	Dim	Domain	No. of local min	Global minimum value
Branin	2	$[-5, 10] \times [0, 15]$	3	0.398
Goldstein-Price	2	$[-2, 2]^2$	4	3
Hartman3	3	$[0, 1]^3$	4	-3.86
Shekel5	4	$[0, 10]^4$	5	-10.1532
Hartman6	6	$[0, 1]^6$	4	-3.32

Table 3.2: Average number of function evaluations for a relative error of $< 1\%$ for comparison of DACE and RBF and RFM on Dixon-Szegö (Dixon & Szego, 1978) test functions.

Test function	DACE	RBF	RFM
Branin	28	40.58	43.6
Goldstein-Price	32	58.6	64
Hartman3	35	61.4	61.6
Hartman6	121	$*212.7(9)^1$	$*200(11)$
Shekel5	NA	$*451(26)$	$*350(20)$

becomes extremely slow when model dimension exceeds a certain limit. This point has been raised in the literature as well (Huang et al., 2006). Huang and *et al.* (Huang et al., 2006) have reported that the high computational burden of DACE model optimization for high dimensional models causes the modeler to avoid widely employing this method.

In addition, in all of the results that had been presented by Jones *et al.* (Schonlau et al., 1998), corner points are included in the training data. However, corner points in the training data is possible just for low dimensional models. For instance, for the WATCLASS model of dimension 16, 2^{16} corner points and their corresponding function evaluations are needed, which is impractical. Gutmann in (Gutmann, 2001b) has also carried out a set of experiments on the sample sets that includes corner points. This experiment results illustrate that inclusion of corner points will lead to up to 50% fewer number of function evaluations. In this experiment, however, we did not consider corner points in the training data since it is impractical to include them in our main case study, WATCLASS.

RBF, DACE and RFM can be compared in 4 different aspects: number of training data for model validation and model construction, CPU time and mean of number of function evaluations.

As can be seen in Table 3.2, the RBF model compared to DACE and RFM SMOs required a larger number of training sample points. It was also very sensitive to dimensionality and its efficiency was decreased with the growth of the dimension and complexity. For instance, for Shekel function optimization RBF needed 28% more data points than RFM method. DACE needed comparably less data points with consideration of corner points. The RBF validation and construction process was faster than the DACE model and slower than RFM. DACE model construction became overwhelmingly slow for a model with a dimension of 6 or more.

The RFM surrogate model shows more promising results for the complicated and higher dimensional models. For instance, a complicated unsmooth function such as Shekel RFM was more successful than RBF, in terms of both the number of function evaluations and the convergence rate. The number of training data for this model is similar to DACE and the process of model construction is similar to RBF, hence it has the advantages of both models. In addition to that, it requires two/three model parameters (α and β) or (α, β and σ^2) to be tuned compared to DACE that needed $2k + 2$ model parameters and RBF that needed a large number of model parameters depending on the choice of the basis functions and degree of error polynomial. Overall, RFM was more efficient method in terms of both CPU time and clock time.

This pilot experiment illustrates the potential of RFM in optimization of complicated functions. RFM with all of its advantages over DACE and RBF in terms of saving time is suppose to work more efficient for calibration of WATCLASS. Overall, all of these SMOs were successful in the optimization of complicated functions. However, none of these SMOs are suitable for distributed computing. To investigate the potential of these SMOs, an adapted version of these SMOs for distributed computing called Adapted Surrogate Function for Distributed Computing(ASFDC) is explained in section 3.2.3.

3.2.3 Adapted Surrogate Function for Distributed Computing (ASFDC)

In section 2.2.5, it was mentioned that finding a DACE model optimum would limit this method for large dimensional applications. The RBF model also requires a large number of training data which makes it impractical for similar cases. Although the RFM suggested model does not have any of these drawbacks, it proceeds by computing one point at a time. In general, none of the mentioned methods are suitable for distributed computing, since their optimization process is serial and the surrogate model results depend on the results of previous iterations. To overcome these issues, a heuristic sampling and search method is suggested. This method has been developed and applied to a Fuzzy-TSK surrogate model in (Kamali et al., 2005) and DACE surrogate model in (Kamali et al., 2007) and the results are shown to be very promising.

The proposed heuristic optimization strategy is based on the random generation of a large sample size. In order to cover the entire region of interest 10,000 (or more) sample points using a space filling sampling strategy such as Latin Hypercube Sampling were generated. The region of interest is chosen based on parameter values that produce acceptable function values based on the surrogate model prediction. For instance, if it is for a WATCLASS model calibration case, the parameters set that produce an NS number greater than 0.6 are selected. It is also possible to choose the parameters that yield the 50 best output results with respect to the surrogate model. The bound on the large LHS sample is the minimum and maximum of the parameters producing these results.

Function values of these samples with respect to the DACE/RBF/RFM surrogate models are computed. The sample points and their function values are then arranged in descending order with respect to their predicted function values and the first point in the column and its corresponding parameter set (i.e., the best point having best predicted function value in the column) is chosen. While distributed computing facilities or similar facilities is available, a set of best points instead of one best point is chosen and their function values with respect to the expensive model are computed.

Distributed computers provide the priority of calling the expensive model in parallel and therefore the time needed for one simulation will be the same as many. In the next iteration, the new simulated points are added to the sample collections and they will be ranked once again. The same process as before will be repeated but with a 10% increase in the minimum of NS number or a smaller number of the best acceptable results (e.g. 40 points instead of 50 are selected). This process makes the region of interest shrink and becomes smaller as the process proceeds. Hence, the search is performing in a smaller region and the chance for finding the optimum is higher.

Since this method is heuristic, it is possible to encounter the same results for two consecutive iterations. This is usually due to failure in complete local search. To avoid being confined in a small boundary in consecutive iterations a shrinkage strategy was adapted, where the LHS sample were generated in a small neighborhood around the best simulated point. This strategy improves the local search and helps to enrich the region in the close vicinity of the best simulated point.

The proposed search technique has the privilege of being readily adaptable to any distributed computing facility and also to our available supercomputer SHARCNET (for more information about this facility refer to `http://www.sharcnet.ca/index.php`). It is also adaptable to any future expansion of this facility, since it is easy to control the number of simultaneous simulations. The other advantages of this method is being understandable and not being specific to the type of surrogate model.

The details of Adapted Surrogate Function Optimization for Distributed Computing (ASFDC) are described in a flowchart in Fig. 3.2 and each of its seven stages are explained in the following. This is the general procedure of ASFDC for optimization of large scale models. However, in chapter 5 we have provided more details about applying this process to our case study which was calibration of WATCLASS model.

1. Stage 1: Sampling.

   A series of random samples of parameters is generated. Latin Hypercube Sampling (LHS) method is recommended specially if all of the parameters distributions are uniform. Thereafter all samples are transferred to SHARECNET (or any distributed computing service) where thousands of simulations could be done simultaneously.

2. Stage 2: Approximate modeling.

   Each of the surrogate models of SMOs, RBF and DACE and RFM, are trained and validated using training data set and validation data set. The SMO is constructed based on n best training data selected from the results obtained so far. n is chosen to give least prediction error. The surrogate model is employed to model the input of the model with respect to its output. The input data for the case study of this research is parameter values and their corresponding NS numbers. The NS numbers (see Eq. 2.1) and their corresponding parameters sets are modeled with the surrogate model. Thus, an approximate model is constructed to map parameter sets with respect to NS numbers. Since the procedure is the same for all SMOs, in the flowchart they are all in the same row.

3. Stage 3: Second round of sampling.

   Using LHS, a large sample of inputs in the region of surrogate model are generated (e.g. 10,000 or more). This way, the entire space of the input (in our case study space of the parameters) are covered with dense sample points. The upper and lower bound of the input samples depends on upper and lower bound on the inputs that were chosen in the previous stage. So, it is guaranteed that these sample points are in the region of the surrogate model.

4. Stage 4: Finding the optimum of the surrogate models.

   Based on the trained surrogate models, a set of input data that had the best results with respect to the surrogate model is selected. In our case study, (100 or 50) parameter sets with the best surrogate model function values are selected.

   To balance the local search and global search together, if after two consecutive iterations the original model output results have not been improved, the shrinkage strategy is employed. This procedure is designed to search the space more locally and so several sets of parameters are generated in the close vicinity of the best found point. In our case study, 100 (or 50) sample points using LHS in very close vicinity of parameter that generate the best NS number were generated.

5. Stage 5: Model simulation.

   The original expensive model call for the selected sample points. In this case study for 100/50 sample points the simulation results of WATCLASS were found using the distributed computing facilities.

6. Stage 6: Comparison.

   The output of the real model is compared with the approximate model output to evaluate the approximation process.

7. Stage 7: Final decision.

   If the predicted results of the surrogate model were close to the simulation results of the real model, the approximate model is reliable and the optimum point is the model optimum point. In our case study, if desirable results such as acceptable NS number was achieved the process was terminated. Otherwise these points are added to the points in stage 1 and the process is started from training and validation stages. Since the number of training data should stay constant, model is trained with respect to the best results. Thus, the interval that surrogate model mimic becomes smaller and hence, the modeling process gets more concentrated in the optimum region.

The performance of adapted SMOs were evaluated by optimization of Dixon-Szeg*ö* test functions. Table 3.3 illustrates the performance of adapted SMOs in the optimization of these test functions. These experiments are conducted in 30 trials and the mean number of function evaluations are reported. All the experiments started with the same number of training data as the previous experiments on original version of SMOs.

The results of Table 3.3 compare the performance of ARBF, ADACE and ARFM on five test functions. The reported performance of ARBF and ARFM on Shekel functions shows that the adapted version worked even better in terms of convergence, moreover, ARBF needed fewer function evaluations compared to Gutmann-RBF.

For all test functions the number of iterations is reduced, since the function evaluation

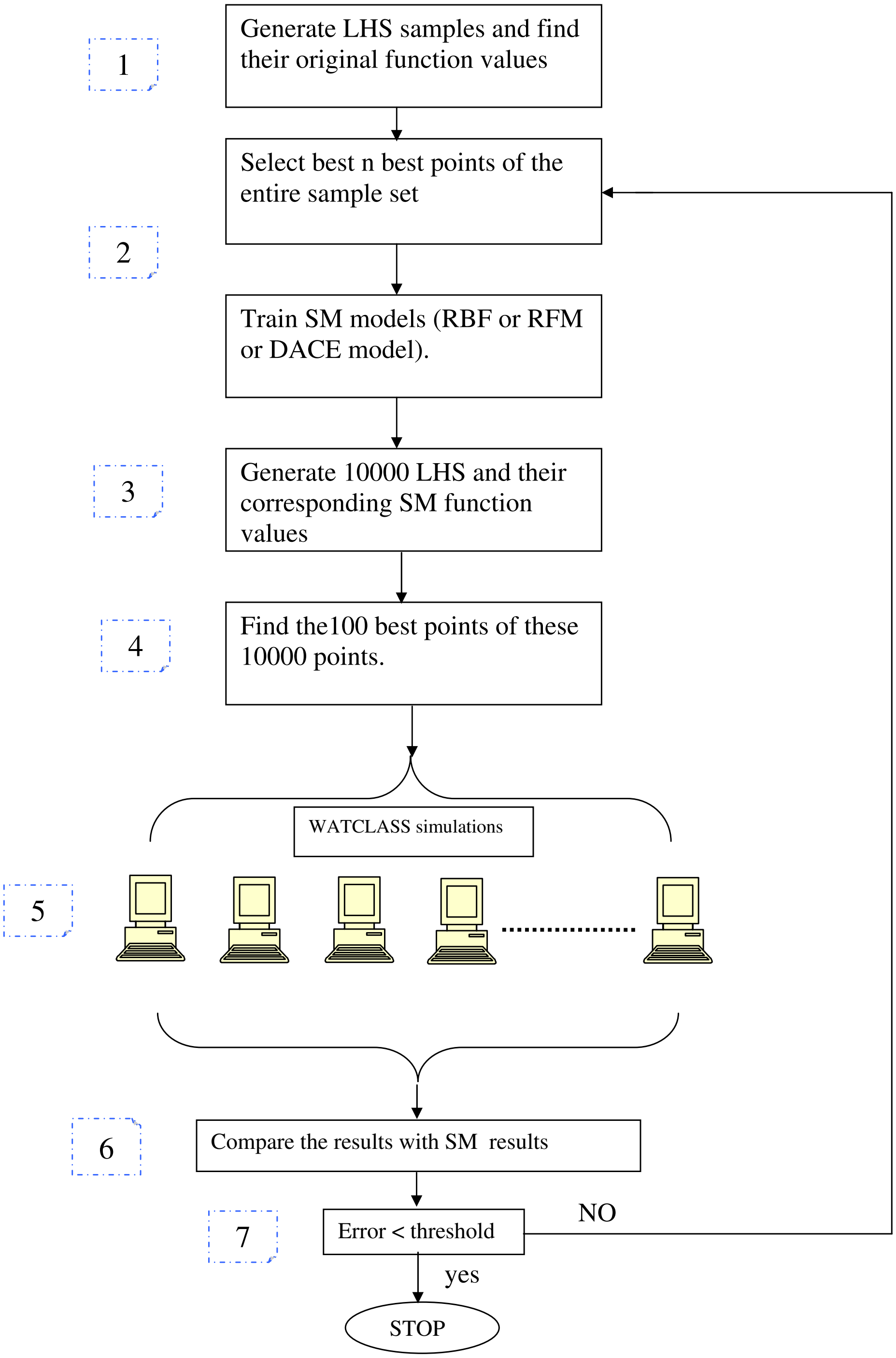

1
Generate LHS samples and find their original function values
Select best n best points of the entire sample set
2
Train SM models (RBF or RFM or DACE model).
3
Generate 10000 LHS and their corresponding SM function values
4
Find the100 best points of these 10000 points.
WATCLASS simulations
5
6
Compare the results with SM results
7
Error < threshold
NO
yes
STOP

Figure 3.2: The proposed framework for performing automatic calibration

Table 3.3: Average number of function evaluations for a relative error of < 1% for comparison of Distributed Computing Adaptive DACE and RBF and RFM on Dixon -Szegö (Dixon & Szego, 1978) test functions.

Test function	ADACE	Max and Min of ADACE	ARBF	Max and Min of ARBF	ARFM	Max and Min of ARFM
Branin	38	25-41	42.8	38-45	46	40-51
Goldstein-Price	48	22-58	61.8	31-84	65	62-71
Hartman3	51	33-61	66.1	43-89	65	58-74
Hartman6	210(4)	177-286	*233(8)[2]	118-250	*205(9)	193-223
Shekel5	398(13)	342-459	*425.7(14)	390-470	*361(11)	315-386

process became mostly parallel and not serial. Using distributed computers makes the number of iterations more important than number of function evaluations. ARFM performance specifically for more complicated functions such as Shekel5 and Hartman6 was noticeably better than ARBF. These results endorsed our assumption that the heuristic framework works well for optimizing complicated test functions. Encouraging by the results, we applied this strategy for calibration of WATCLASS.

ADACE works well with this new optimization framework. It performance was more satisfactory than ARBF for Shekel function. These results show the overall better performance of ADACE and ARFM compared to ARBF. This should stem from similar assumption exists in both ARFM and ADACE, that points in the same neighborhood should have similar outputs.

3.3 Sensitivity Analysis for Calibration Purposes

3.3.1 Introduction

Nearly all of the mentioned SA methods and their extensions were dealing with the role a parameter plays in the model output. But the issue of conducting sensitivity analysis for a model with high computational cost still exists. What if the main objective of Sensitivity Analysis is model calibration? And what if the strong relationship between the parameters does not allow the elimination of some of the parameters? Hence, if the main objective is to reduce model complexity to calibrate the model with less effort, the sensitivity assessment approach of the model could be a bit different from what has been mentioned so far. Furthermore, once computational cost becomes the most important issue, it would be ideal to conduct SA and to perform calibration on a model using the same set of data.

One way to reduce the computational burden and the complexity of the calibration process is reducing the dimension of the model that is optimized (or calibrated, for WATCLASS). In the SMO framework, the main medium for model optimization is the surrogate model. An intuitive way for the optimization of a lower dimensional model is to construct the surrogate model with a subset of model parameters that gives the least prediction error for the surrogate model. Thus, the parameters are screened for their prediction precisions of the surrogate model rather than reduction in output variance.

One strategy to find this subset is putting aside one parameter and constructing the surrogate model with the rest of the parameters. The least important parameter is the one that results in the least prediction error on the surrogate model. This parameter is then omitted from the set of parameters and the process continues until a subset of parameters that results in a surrogate model with acceptable prediction precision is selected. This technique is a famous technique in the Feature Selection field and is called backward elimination (Kohavi & John, 1997). This process is understandable, simple and useful, however, it is not a SA method appropriate for a model with correlated/associated parameters.

There are other ways of reducing the dimension of the surrogate model. For instance, it is possible to have a different set of parameters (preferably with lower dimension) such that each of them is a linear combination of the original (here surrogate) model parameters. This method exists and is called the Principal Component Analysis (PCA). This well-known method has been cited more than 5000 times.

PCA couple with RFM

The Principal Component Analysis method (PCA) is a statistical technique that has found applications in pattern recognition and in image processing. This technique dates back to 1901 which initially was brought up by K. Pearson (Pearson, 1901). PCA is a method for reducing the dimension of the space with minimum information lost. The main idea is to discover a smaller set of model parameters with less redundancy, where the measure of redundancy is a correlation between parameters. In this technique the data is mapped onto an orthogonal lower dimension. This technique was coupled with RFM for reducing the computational cost of WATCLASS model.

If X is the original matrix of parameters having k dimensions, in this technique a reduced matrix XX is found that has the useful information of X with lower dimension. So,

$$XX = W^T X \tag{3.10}$$

without loss of generality, two constraints could be added:

First, cov(XX) is a diagonal matrix or cov(XX)= $\begin{pmatrix} \lambda_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \lambda_k \end{pmatrix}$

Second,

$$W^{-1} = W^T \tag{3.11}$$

where W is a $k \times k$ orthonormal transformation matrix. Then,

$$\begin{aligned} cov(XX) &= E[(XX)(XX)^T] \\ &= E[(W^T X)(X^T W)] \\ &= W^T E[XX^T] W \\ &= W^T cov(X) W \end{aligned} \tag{3.12}$$

by rearranging it:

$$W cov(XX) = cov(X) W \tag{3.13}$$

Since $cov(XX)$ is a diagonal matrix and W is an orthonormal matrix, we arrive at the familiar linear algebra equation of eigenvalues and eigenvectors as following:

$$[\lambda_1.\vec{W}_1, \cdots, \lambda_k.\vec{W}_k] = cov(X)[\vec{W}_1, \cdots, \vec{W}_k]$$

where $W_1, \cdots, W_k$ are k eigenvectors of $cov(X)$. Identifying that W is the matrix of covariance of X eigenvector in Eq. 3.10, it is possible to estimate a lower dimensional matrix XX by only keeping the eigenvectors corresponding to the largest eigenvalues and shrinking the dimension of W by $l \ll k$.

Usually the amounts of information lost during the process of dimensionality reduction using PCA is calculated by:

$$InfoL = 1 - \frac{\sum_1^l \lambda_i}{\sum_1^k \lambda_i} \tag{3.14}$$

Since $\lambda_1 \ll \lambda_2 \ll ...\lambda_l \ll ... \ll \lambda_k$, it is clear that with keeping the eigenvector rows (in W matrix) corresponding to the larger eigenvalues and omitting the rest of rows corresponding to the smaller eigenvalues we have not lost a considerable amount of information.

Now that the input space has reduced, it is possible to find the best set of parameters in the reduced space. Having the RFM formulation (presented in section 3.2.1) it is possible to replace X which was input data by XX which is the reduced input data in Eq. 3.6 and then arrive at:

$$\beta = (1-\alpha)((1-\alpha)[XX][XX]^T + 2\alpha[XX]L[XX]^T)^{-1}[XX]Y \tag{3.15}$$

The dimension of XX is $l \ll k$, hence the dimension of β will be $1 \times l$ instead of $1 \times k$. This way, all of the computations will be performed in the lower space and so a more precise surrogate model with the same number of data is achieved.

It is also possible to augment the dimension of XX and benefit from the nice behavior of data in the higher dimension. As was mentioned in the previous section, $\chi\chi$ is computed as $\chi\chi = e^{(-\|[XX]^i - [XX]^j\|_2/2\sigma^2)}$. If XX in Eq. 3.15 is replaced by $\chi\chi$:

$$\beta = (1-\alpha)((1-\alpha)[\chi\chi][\chi\chi]^T + 2\alpha[\chi\chi]L[\chi\chi]^T)^{-1}[\chi\chi]Y \tag{3.16}$$

Eq. 3.16 and Eq. 3.15 demonstrate that with a small amount of computational efforts, we are able to perform all computations in a smaller space. This characteristic is unique to RFM, as DACE and RBF do not have the protectional of being cast in a closed form formula.

Having β, we are able to find the corresponding optimum of the model. The procedure is very similar to what explained in sections 3.2.1. Once one or couple of XX candidates that performed acceptable on the RFM surrogate model are found, they will transform to the original input space by $W.XX = X$, where X will be the desired input or sets of input that should be simulate using the original model.

3.3.2 PCA for Sensitivity Analysis Application

As it was pointed out in section 3.3.1, the main purpose of using PCA in sensitivity analysis is to facilitate the calibration process. Calibration is performed in a lower dimension using PCA to reduce computational effort. As a result, the process has less computational cost and it requires less training data. Furthermore, the model is more tractable when it is mapped on a lower dimension.

The general algorithm of the using PCA coupled with an SMO is shown in Algorithm 2. In this algorithm, PCA works as a preprocessor that refines input data and come up with a new coordinate for the data in the lower dimension. The rest of the optimization process is similar to previous processes except that it takes advantage of the lower dimension data. Once the optimum/near optimum points were obtained, they will be mapped on the upper dimension in order to be simulated with the original model.

In the first two and the last stages of Algorithm 2 the same procedure as the first two stages of Algorithm 1 are followed. However, in Algorithm 2 the surrogate model is constructed based on the reduced input data instead of original input data.

3.4 Summary

In this chapter the main contributions of this research are explained. In the the first part, the proposed calibration algorithm is detailed. The proposed framework are employed for optimization of some test functions. The promising performance of the proposed

Algorithm 2 Algorithm for the calibration process using the PCA

```
1: Randomly generate samples of the parameters.
2: Find the corresponding model output for the sample.
3: Reduce the space dimension using PCA and find the principal directions that contain 98% of the model information.
4: Construct the surrogate model based on the data on the reduced space.
5: Using the surrogate model, find the optimum/best points of the surrogate model in the reduced space.
6: Map the optimum point on the real space, find its corresponding point in the original space and find its corresponding original model simulation results.
7: Compare surrogate model results and the original model results, if they were close enough this point is the original model optimum point and the surrogate model is complete. Terminate the process, otherwise continue from first step.
```

framework lead us to optimize real world models using this framework. In the second part, a new sensitivity analysis approach using PCA dimensionality reduction technique for facilitating calibration is explained.

In the next chapter the case study that all of the mentioned methods are employed to calibrate is introduced.

Chapter 4

Case Study

The case study of this research is the WATCLASS hydrologic model that has been calibrated over Smoky River watershed. This watershed is chosen since it is comparably large and complex enough to challenge a physically based model. In addition to that a complete input data set was available to us which could lead to a more accurate process of calibration. This hydrologic model and watershed are described in this chapter.

4.1 Smoky River Watershed

4.1.1 Smoky landcover

Smoky River watershed is a part of Canada's largest watershed, the Mackenzie River Basin. This basin itself is a part of the Arctic Ocean drainage basin. The Smoky River is a tributary of the Peace River (see Fig. 4.1) and occupies $3840km^2$ drainage area in the foothills of Rocky Mountain. Fig. 4.1 depicts the schematic geographic location of the Peace River in Alberta. The Smoky flows into the Peace River. Figure 4.2 shows the map of dominant land cover. In this map, each area is coded with a color and

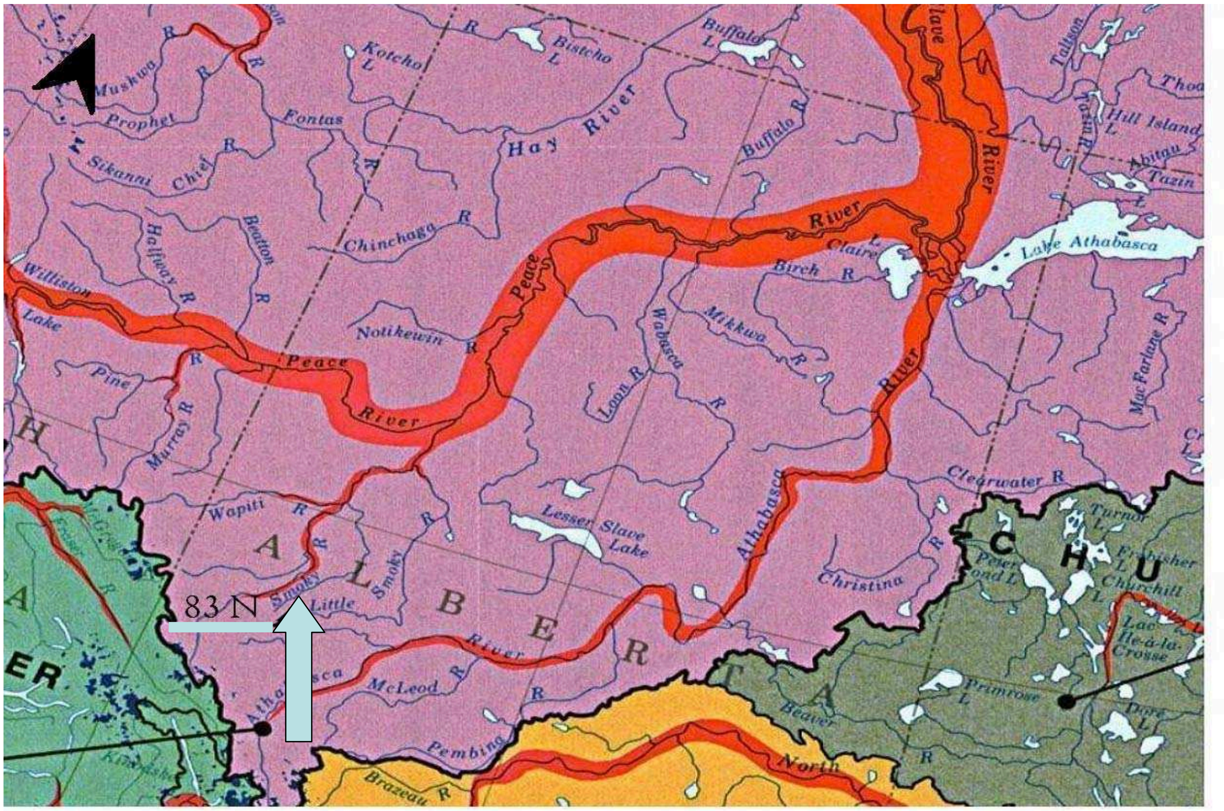


Figure 4.1: Smoky River Watershed is shown by an arrow it is located in south part of Alberta

with an abbreviations. The primary cover class which covers the majority of the land is shown with a color. For example, Forest is shown in green and farmland is shown in red. The secondary cover class, which stands second in occupying an area, is shown by an abbreviation. For instance, if for an area forests are the secondary coverage it is shown by AF. The same for Taiga, where it is the second land coverage it is shown as FV in the map. The map of entire area is accessible through the Natural Resources Canada web site at `http://atlas.nrcan.gc.ca/site/english/maps`

The calibration of this watershed was done over period of three years. Two different instances of WATCLASS over Smoky were calibrated and hence two different sets of parameters were used. In the first set, sixteen parameters corresponding to one land class

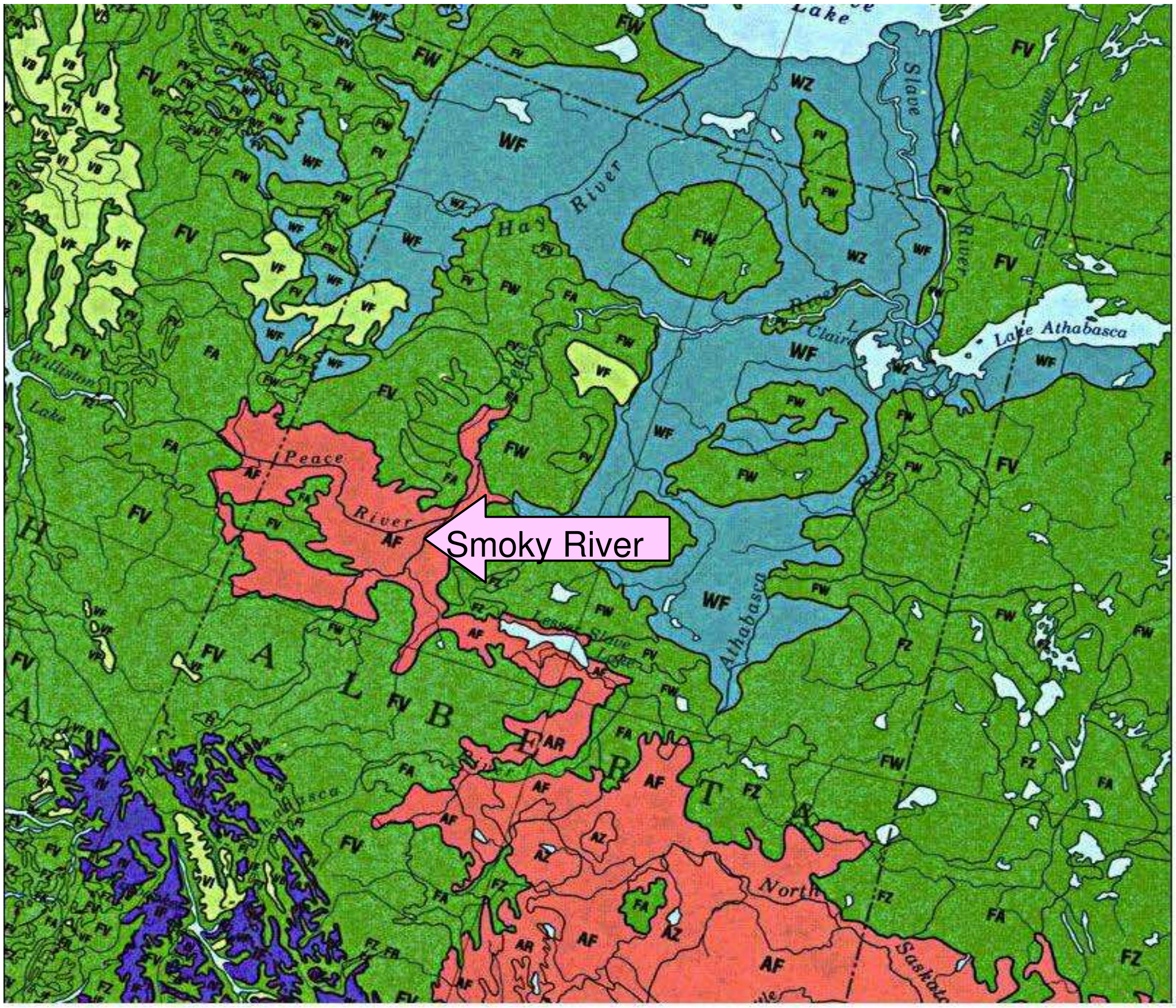


Figure 4.2: Different land covers of Smoky River Watershed in the map Each area is coded with a color and with some abbreviations. The primary cover class which covers the majority of the land is shown with a particular color. Forest is shown in green and farmland is shown in red. The secondary cover class, which stands second in occupying an area, is shown by the abbreviations. For instance, forests are shown as AF, Taiga is shown as FV and wetland is shown as FV.

for the entire watershed were assumed. The list of the parameters and their descriptions is in section 4.1.2. In this thesis we just performed model calibration and did not deal with model validation.

In the second set, two land classes which corresponded to two land classes were assumed. The geographic map of the basin confirms that approximating the entire watershed in one or two land classes is an appropriate approximation.

The upper bound and lower bound of the parameters were dictated by either physical characteristic of the watershed that mainly was coming from field experiments or extensive experiments with WATCLASS.

4.1.2 Parameters definition

First parameter set

The list and definition of the first set of parameters that was chosen to be calibrated is shown in Table 4.1. As was noted above, these parameters characterize one land class subsurface for Smoky watershed. In Fig. 4.2, one could see that Smoky is mainly covered by forest. In the first set of experiments, it is assumed that Smoky is covered by one land class and that WATCLASS has been calibrated over this land class.

Basically, the parameters that are presented in Table 4.1: ddenrow and xslprow are responsible for overland flow; sand, clay; wfcirow and organic are responsible for interflow; drnrow, grkfrow, wfsrow and sdeprow are responsible for baseflow.

Second parameter set

The definition of the second set of parameters are very similar to what is presented in shown in Tables 4.1, but since two land classes were assumed the same of parameters were assumed for the second land class. In this set, it is assumed that Smoky is mainly covered

Table 4.1: CLASS.ini Parameter description

Parameter	Description	Lower Bound	Upper Bound
ddenrow	Drainage index for layer 1 for landclass 1	0	1
wfcirow	interflow index (mean of KSAT [1])for layer 1 for landclass 1	0	20
xslprow	valley slope for landclass 1	0.001	0.1
sand11	volume of sand in layer1 landclass 1	0	100
clay11	volume of clay in layer1 landclass 1	0	100
orgm11	volume of organic soil in layer1 landclass 1	0	20
sand12	volume of sand in layer2 landclass 1	0	100
clay12	volume of clay in layer2 landclass 1	0	100
orgm12	volume of organic soil in layer2 landclass 1	0	20
sand13	volume of sand in layer3 landclass 1	0	100
clay13	volume of clay in layer3 landclass 1	0	100
orgm13	volume of organic soil in layer3 landclass 1	0	20
drnrow	vertical conductivity factor at bottom of layer of landclass 1	0	1
grkfrow	change in lateral conductivity at depth H_0 of landclass 1	0	1
wfsrow	Drainage index for layer 3 of landclass 1	0	1
sdeprow	depth of weathered soil of landclass 1	0.4	3

by forest but that the rest of the watershed is covered by farmland. WATCLASS over Smoky is calibrated using this set of parameters. In addition to land classes parameters, twelve parameters that describe river land classes are added to the previous parameter set. In this set of experiments, parameters that were involved in the process of snow melting are also calibrated. The list of these parameters can be seen in Table 4.2.

The parameters in Table 4.2 include the ones that are responsible for cold soil process (snowlimvs)and WATFLOOD hydrologic parameters (wf_r1s).

4.2 WATCLASS

WATCLASS is a meso-scale distributed hydrologic model that combines Canadian Land Surface Scheme (CLASS) ((Verseghy, 1993) and (Verseghy, 1991)) and WATFLOOD (Kouwen, 1988). WATCLASS was developed for the Mackenzie River Basin (MRB) as part of the Mackenzie GEWEX study (MAGS)(Soulis & Seglenieks, 2007). This model is specifically designed to close the energy and water balance for the Mackenzie River basin (MAGs) using the results of the atmospheric and hydrologic research. By definition a basin of a river contains all of the land area that their runoff contribute to that river. This definition applies to Mackenzie River basin as well. This river discharges to the Arctic Ocean and its basin occupies an area of $1.8 \times 10^6 km^2$. Because of the impact of this basin on global warming many researches from all over Canada have concentrated on different aspect of climate change in this basin.

WATCLASS model was initially developed to create the ability to model continental domain watersheds at short time scale (1 hour or so) and at small spatial scales (10 to 25 km). This project attempted to improve the assessment of Canada's water resources relating to change in the climate. Some processes such as snow accumulation and melt

Table 4.2: CLASS.ini Parameter description

Parameter	Description	Lower Bound	Upper Bound
wf_r1	river roughness for river class1	0	6
wf_r2	river roughness for river class2	0	6
wf_r3	river roughness for river class3	0	6
wf_r4	river roughness for river class4	0	6
wf_r5	river roughness for river class5	0	6
$snowlimv1$	depth for 100% coverage of aged snow class1	0.1	0.5
$snowlimv2$	depth for 100% coverage of aged snow class2	0.1	0.5
$snowlimv3$	depth for 100% coverage of aged snow class3	0.1	0.5
$snowlimv4$	depth for 100% coverage of aged snow class4	0.1	0.5
$snowlimv5$	depth for 100% coverage of aged snow class5	0.1	0.5
$snowlimv6$	depth for 100% coverage of aged snow class6	0.1	0.5
$gwscale$	log-fraction of storage groundwater released	0.1	0.5

and blowing snow were among physical processes that were not well understood (Marsh & Gyakum, 2003) and well modeled in other hydrologic models.

In order to predict streamflow, WATCLASS requires 7 inputs, which are precipitation, temperature, humidity, pressure, wind speed, short wave and long wave radiations. These inputs are generated by Environment Canada global atmospheric forecast model called Canada Global Environmental Multiscale (GEM) model that has been in operation since Oct. 2006. GEM uses mathematical equations for modeling the physics and dynamics of the atmosphere to predicts these inputs. GEM input is usually coming from weather satellites stations.

A detailed explanation of CLASS and WATFLOOD are in (Barlett et al., 2006), (Barlett et al., 2003) and (Stadnyk et al., 2005). The motivation behind the development of WATCLASS was to improve the soil-water budget in CLASS. The additions include lateral flow and improvement of the cold water process in the CLASS model. These improvements have affected overland flow, interflow, baseflow and cold-soil process algorithms, which are briefly explained in the following subsections. However, WATCLASS modification for iced soil process is beyond the scope of this document and can be found in (Soulis & Seglenieks, 2007).

Overland Flow

In general, the overland flow (surface runoff) is the flow on a sloping surface. This phenomenon occurs either when the surface is saturated from above such as the case of impermeable soils and is called Hortonian overland flow, or when the surface is saturated from below, which usually happens when break-out of ground water happens.

In CLASS version 3.3, when the depth of ponded water becomes more than 10 cm, overland flow will be generated. In WATCLASS this process has been improved. An overview of overland flow treatment of WATCLASS follows.

Combining continuity, momentum and energy equations for turbulent in open channel flow conditions gives

$$q = \frac{\sqrt{s}}{n} h^{5/3} \tag{4.1}$$

where q is the unit flow rate, h is the surface flow depth, s is the slope and n is the Manning surface roughness coefficient. In order to quantify q, change in storage S with respect to time $\frac{dS}{dt}$ should be considered. The variation in the storage is measured by $\frac{dS}{dt} = -A.S_0^{5/3}$, where $A = \frac{\sqrt{slope}}{L_s^{5/3}.L_v^{2/3}.n}$, $S_0 = de_0.L_v.L_s$, where de_0 is the initial water depth above ponding limit, L_v is the stream length and L_s is the effective hillslope length, then $\frac{dS}{dt} = \frac{d(de.L_v.L_s)}{dt}$, or $\frac{d(de.L_v.L_s)}{dt} = \frac{d(de)}{dt}.L_v.L_s$ where the variation in water depth above ponding limit or $\frac{d(de)}{dt}$ is overland flow. So that,

$$-A.S_0^{5/3} = \frac{d(de)}{dt}.L_v.L_s$$

and

$$\frac{d(de)}{dt} = -A.S_0^{5/3}/L_v.L_s$$

The detailed explanation of formulas and implementation can be found in (Soulis & Seglenieks, 2007).

Interflow

Interflow is the term that stands for near surface lateral flow. Interflow in CLASS 3.3 has been mainly modeled by Richard's equation which is a combination of Darcy's law for vertical unsaturated flow with continuity equation.

$$\frac{\partial \theta}{\partial t} = \frac{\partial}{\partial x}(k\frac{\partial \psi}{\partial x}) + \frac{\partial}{\partial y}(k\frac{\partial \psi}{\partial y}) + \frac{\partial}{\partial z}(k\frac{\partial \psi}{\partial z}) - \frac{\partial k}{\partial z} \tag{4.2}$$

Richard's equation computes the change in the volumetric soil water content with respect to time. In Eq. 4.2, θ is the volumetric soil water content , ψ is soil potential, t is time and k is hydraulic water conductivity.

In CLASS 3.3, only the vertical movement which involves gravity and capillary forces is considered (Soulis & Seglenieks, 2007). In WATCLASS, Richard's equation is solved under the assumption of long shallow aquifer. Thus, the horizontal flow and the vertical flow are considered together.

Interflow is calculated based on the following formula:

$$q_{int} = 2D_D.K_0(\theta_S.S^f.\epsilon H.\Lambda_I) \tag{4.3}$$

In 4.3, D_D is the drainage density, θ_S is the porosity, K_0 is the saturated hydraulic conductivity, S is the effective saturation, f is a coefficient related to soil properties, Λ_I is the aquifer slope and H is the aquifer thickness and ϵ is the aquifer efficiency and is defined as: $\epsilon = \frac{(1-e^{-\lambda H})}{\lambda H}$ and it ranges from 0 to 1 where λ is a conductivity decay coefficient (Soulis et al., 2000). The detailed formulation can be seen in (Soulis & Seglenieks, 2007) also.

Base Flow

Base flow is the water in the streams supplied mostly from ground-water. Base flow maintains stream flow between water-input events such as rainfall or snow melt. In CLASS, Richards equation(Eq. 4.2) was solved assuming that the soil moisture gradient is negligible therefore the base flow is a Darcian flow or $q_{baseflow} = k_v(\theta)$ in which k_v is vertical hydraulic conductivity and θ is the soil moisture in the deepest layer of the soil element. Since the base flow in this formula is very sensitive to soil moisture and soil moisture changes vertically, it does not vary much. In WATCLASS, base flow is modeled very similar to interflow, furthermore base flow relies on the ratio of the horizontal conductivity at the bottom of the layer to the horizontal conductivity at the top of the layer and field capacity, instead of soil moisture at saturation. These gives WATCLASS more flexibility in modeling the base flow.

Cold Soil Process

Low temperature or snow changes the properties of soil. The capability of CLASS3.3 in modeling frozen soil is limited. Pore space is the only variable that changes with respect to ice content and other soil properties remain the same. The WATCLASS model accommodates variation of conductivity and porosity with respect to the ice content. The resultant porosity is called effective porosity and is calculated as: $\phi_{effective} = max(\theta_{liq} - \theta_{min}, \phi - \theta_{ice} - \theta_{min,0})$

where θ_{liq} is volumetric content of liquid water, θ_{min} is the minimum volumetric water content (hydroscopic) and θ_{ice} is the volumetric ice content (hydroscopic).

4.2.1 WATCLASS Gridding

The WATCLASS model, similar to any distributed hydrologic model, is constructed to generate stream flow for a particular watershed. The watershed is divided into grids. The WATFLOOD model is based on a discretization approach called group response unit. In this approach each grid is divided into classes that share similar land covers. Each land cover contributions to the response is determined with regards to its contribution to the grid area. Each grid can theoretically have a large number of classes, however, six land classes have been found to be sufficient. Each of these land classes contains many parameters that need to be optimized. The precision of the modeling process is very much limited to the available computational resources. In the most common scenario, a watershed is divided into 10 km by 10 km grids where each of this grid has 6 land cover classes. WATCLASS is able to model grids from 10 km by 10 km to 50 km by 50 km.

4.2.2 Different Versions of WATCLASS

Throughout the course of this study, WATCLASS was updated at least three times. Class 2.7 various extensions codes were experimented as the time of study. Later on, most of the enhancements became part of CLASS 3.3. This version of CLASS plus the lateral flow and streamflow routines codes from WATCLASS have become a part of Environment Canada Mesh(Modèlisation Environmentale Communautaire Surface and Hydrology MEC-H) and System. The experiments in this thesis are mainly produced with the newer version, however, in one case sensitivity and calibration of the model was done simultaneously (shown in section 5.3.7) using an older version of WATCLASS. The results of this experiment show the better performance of the recent version.

To demonstrate the range of the results that WATCLASS can produce, two hydrographs are presented. Figure 4.3 shows the first state of the model that we started with and Figure 4.4 shows the state when we achieved desirable results.

In order to give the hydrologist an idea about the real values of WATCLASS model parameters, Table 4.3 presents a set of typical parameter values that was resulted in acceptable NS results. The neighborhood radius where similar NS results are achievable is also provided.

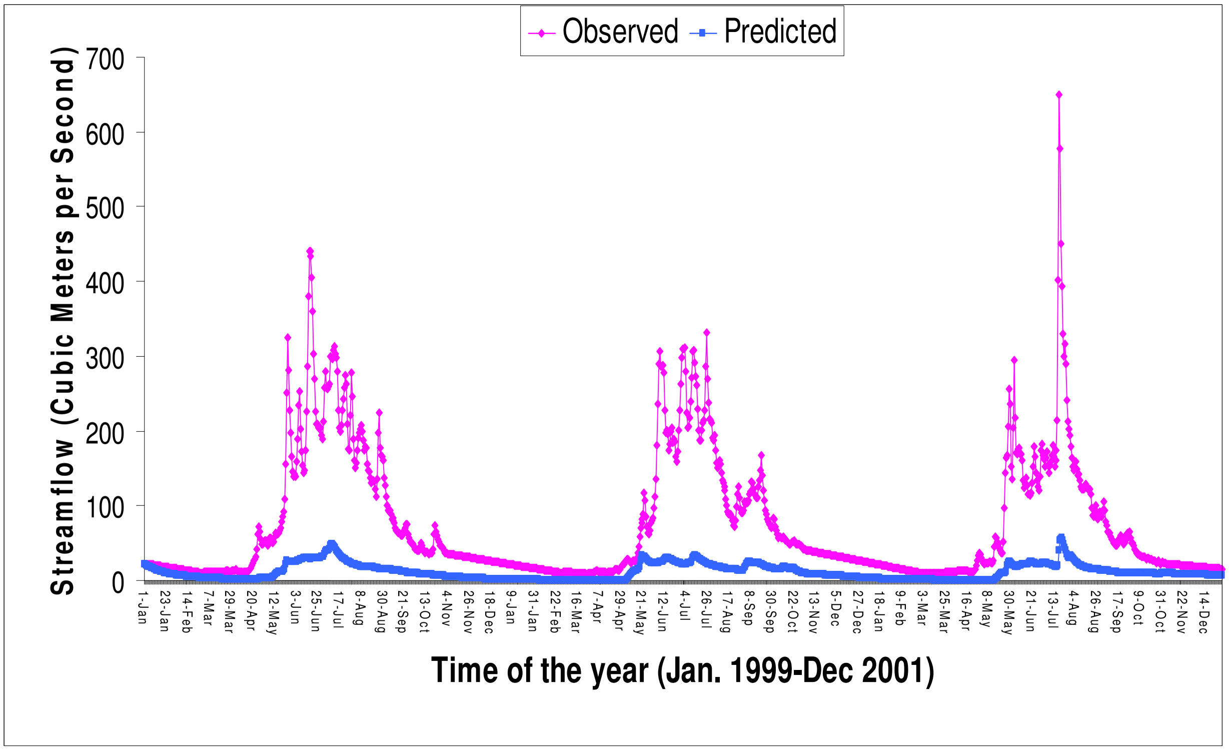


Figure 4.3: Hydrograph of WATCLASS prediction and achievement of ”not acceptable” results (NS= -0.3)

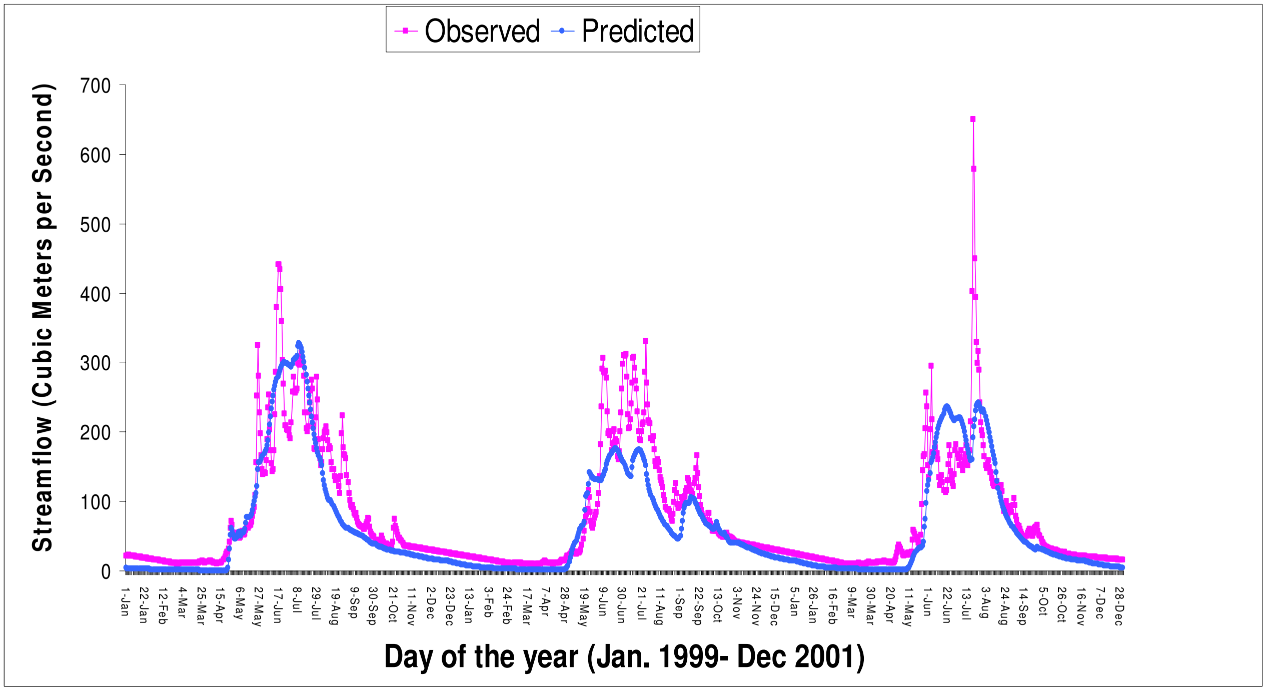


Figure 4.4: Hydrograph of WATCLASS prediction and achievement of ”acceptable” results (NS= 0.72)

Table 4.3: Calibration results for CLASS.ini Parameters

Parameter name	Parameter value	neighborhood radius
ddenrow	0.5	+0.1
wfcirow	5.258	0.8
xslprow	0.01	0.005
sand11	42.2	12.8
clay11	51.3	10
orgm11	6.5	1
sand12	64.4	10.1
clay12	31.7	5
orgm12	3.9	1
sand13	24.7	21
clay13	75	7
orgm13	0.3	2
drnrow	0.43	0.05
grkfrow	0.2	0.18
wfsrow	0.5	0.12
sdeprow	1.8	0.1.2

Chapter 5

Results

5.1 Introduction

This chapter has two parts. The first part of this chapter presents the results of calibration of WATCLASS over Smoky River using three described surrogate model optimization methods. The calibration results for two different parameter sets are investigated and presented. The results are compared based on five criteria: maximum obtained Nash-Sutcliffe numbers; minimum obtained Nash-Sutcliff numbers; mean and variance of obtained Nash-Sutcliff numbers for last iteration over all trials and mean of number of function evaluations for all trial to achieve desirable results. The results serve the first purpose of this experiment, calibration, and they also evaluate the sensitivity of the surrogate model around the optimum results compared to WATCLASS.

In the second part of this chapter, Sensitivity Analysis results of WATCLASS using some of the described sensitivity methods are presented. Afterward, calibration results of WATCLASS when the dimension of the model is reduced by PCA is also presented.

5.2 Calibration

This section is focused on the WATCLASS calibration results. In this part, the ASFDC algorithms are applied to two different WATCLASS case studies. In the first case study, sixteen model parameters and in the second case study forty-four parameters were calibrated. The description of these parameters can be seen in sections 4.1.2.

The criterion for the calibration of WATCLASS is Nash-Sutcliffe criterion as defined in Eq. 2.1.

The surrogate models are employed to model WATCLASS model performance (NS numbers) with respect to the WATCLASS parameters. The surrogate model is then employed to predict the values of NS on some unseen parameter sets. The parameter sets that correspond to acceptable NS are selected and afterward the original model simulation is performed (e.g.,in this set of experiments, for each iteration 50 best points for sixteen parameters WATCLASS or 100 best points for forty-four parameters WATCLASS are chosen and their corresponding WATCLASS values are found using SHARCNET distributed computing facility).

NS coefficients of all simulated results are obtained and if one or more than one of the points among this set met the criterion on NS (which was $NS \geq 0.69$), the optimization process is terminated, otherwise the process is repeated with the best new points. Calibration of WATCLASS is conveyed using adopted SMOs and the heuristic optimization technique.

Applying SMOs without using distributed computers was infeasible, since each WATCLASS model evaluation took 10 to 15 minutes on the Pentium 4 computer. It is assumed that at least 1000 function evaluations were required to achieve the acceptable NS number. This number is chosen based on the calibration of WATCLASS using a very efficient global optimization method called DDS (Tolson & Shoemaker, 2007b) which on average requires 1000 function evaluations.

5.2.1 Sampling Strategy

All of the surrogate models are built on samples that have been generated using Latin Hypercube Sampling (LHS) strategy. LHS is a sampling strategy that has been proposed by McKay (McKay et al., 1979). In this technique, the interval of each parameter is divided into m divisions and a random value from each interval is drawn. Later these random values are coupled together randomly to yield N random samples that enjoy a uniform coverage of the parameter intervals. The process of dividing the parameter interval into m divisions enhances the spread of the random samples and assures that samples are taken from the entire interval.

N sample points based on each parameter distribution are generated but these N points are uniformly distributed in each of m divisions. These values are then randomly coupled together and make a vector of parameters (or one sample unit). This sampling strategy has the advantage of being uniformly distributed even with a small number of samples which is unlikely to happen in Monte Carlo sampling.

For the process of calibration of sixteen parameter WATCLASS (first case study) 2000 LHS were generated. In each trial, a certain number of samples that was dictated by model validation stage was randomly drawn from the set of 2000 sample points. The statistical characteristic of the initial sample is presented in Table 5.1.

A similar process was performed for WATCLASS with 44 parameters. 3000 sample points were generated and in each trial a certain number of samples was randomly drawn from this set. The general statistical characteristic of this set is reported in Table 5.2.

Table 5.1: Statistical parameters of LHS samples for 16 parameters WATCLASS

Maximum value of NS[1]	0.67
Minimum value of NS	-0.5
Mean of NS[2]	0.51
Variance of NS	0.0093

Table 5.2: Statistical parameters of LHS samples for 44 parameters WATCLASS

Maximum value of NS[3]	0.64
Minimum value of NS	-0.31
Mean of NS[4]	0.54
Variance of NS	0.0052

5.2.2 Calibration of WATCLASS with sixteen parameters

Adapted Distributed Computational DACE (ADACE)

In this set of experiments, the DACE surrogate model was employed to model WATCLASS parameters with respect to NS numbers. In order to save computational time, each set of training data was randomly selected from the 2000 LHS samples and their simulation results. The first step of this experiment was surrogate model validation. Using the k-fold cross-validation method, the number of training data and other surrogate model parameters was set at a minimum validation error. After the validation stage, the surrogate model was used to conduct the calibration process.

In each iteration, 10000 LHS samples were generated to approximately cover the entire space of the parameters. Then their corresponding function values using DACE model were found. The best 50 sample points out of 10000 points were selected and their

WATCLASS simulated results were achieved using SHARCNET distributed services.

Therefore, in each iteration, 50 data points were added to the collection of the data and the DACE model was reconstructed based on the best simulated results selected from all of the obtained results throughout the previous trials. This policy assured that in each iteration the algorithm becomes more concentrated on the regions with acceptable results. In addition, if after two iterations a better result was not obtained, the shrinkage strategy was employed. In this strategy 50 randomly generated points were produced in the neighborhood of the best obtained point.

The entire procedure was repeated for 15 independent trials. In the DACE technique, the user has to choose the degree of polynomial basis functions and error functions. These choices make the model more flexible on the one side but demanding more time and computational efforts on the other side. The most difficult part of this process, however, was tuning the parameters of DACE correlation function which was computationally expensive. Based on the results of the validation error, the best error function for this surrogate model was cubic spline correlation function and for the regression function it was the hyperline.

The DACE surrogate model had the minimum validation error with 180 training data. In the first iteration, based on the prediction of DACE, 50 points were selected and simulated using SHARCNET. In the second iteration the same procedure as the first one was followed. If the best results of any two consecutive iterations for any of the trials stayed the same 50 points were randomly generated around the best point and were simulated. However, all of the trials reached a NS number greater than 0.68 after 2 iterations and the shrinkage strategy was not employed. Table 5.3 reports the results of these 15 trials. The first and second rows report the maximum and minimum NS that were obtained in the last iterations of all trials. The third and fourth rows report the mean and variance of the maximums of all 15 trials. The last row presents the mean of number of function evaluations.

Table 5.3: The results of ADACE on WATCLASS with 16 parameters

Maximum value of NS[5]	0.706
Minimum value of NS[6]	0.68
Mean of NS for 15 trials[7]	0.692
Variance of NS for 15 trials[8]	3.2425e-005
Mean of No. function evaluations to reach NS=0.68	247

ARBF

The general procedure of calibration of WATCLASS using ARBF was similar to the ADACE procedure explained in section 5.2.2. The lowest validation error occurred when the Gaussian basis function and a first degree polynomial error function were involved. Furthermore, best prediction error results achieved with 280 training data points. The surrogate model training stage took time as many combinations of basis and error functions were tried together.

All trials reached to a desirable NS number in two trials with 380 function evaluations on average. In Table 5.4, the results for adoptive RBF (ARBF) are presented. Although the RBF surrogate model required more training data than DACE, its training process was fast. ARBF achieved satisfactory results after 2 iterations, so the shrinkage strategy was not used at all.

Table 5.4: The results of ARBF on WATCLASS with 16 parameters

Maximum value of NS	0.696
Minimum value of NS	0.679
Mean of NS for 15 trials	0.69
Variance of NS for 15 trials	6.26e-006
Mean of No. function evaluations to reach NS=0.68	380

ARFM

The number of training data for Adaptive Regularized Function Map (ARFM) was similar to ADACE, 180, however, in all trials the surrogate model found satisfactory results after one iteration.

RFM model parameters, α, β and σ^2, were tuned based on the results of model validation error. Unlike ADACE this process was very straightforward and fast. The set of parameters that yield the least validation error on validation data sets were chosen. When compared to previous SMOs, the training and validating stage needed less computational effort. The results are presented in Table 5.5.

Table 5.5: The results of ARFM on WATCLASS with 16 parameters

Maximum value of NS	0.694
Minimum value of NS	0.68
Mean of NS for 15 trials	0.687
Variance of NS for 15 trials	1.3778e-005
Mean of No. function evaluations to reach NS=0.687	230

5.2.3 Discussion of the WATCLASS Calibration Results-Part I

In this part, WATCLASS was calibrated using three SMOs, ADACE and ARBF and ARFM. In general, using SMO approach largely improved the calibration process. For calibrating 16 parameters, an average of 380 function evaluations were required at most. This is less than using a global optimization methods such as DDS (Tolson & Shoemaker, 2007b), which required 400 WATCLASS simulations on average. Furthermore, SMOs results do not depend on initial points, which largely affect the performance of global optimization methods such as DDS.

The other advantage of SMO methods over global optimization methods such as DDS is the potential of spreading the simulations. A simple example may be employed. Imagine that the SHARCNET system has 200 CPUs available for the entire experiments and that each WATCLASS simulation takes about 10 minutes on each of these CPUs. Hence, to calibrate WATCLASS using DDS 400 simulations of 10 minutes or 4000 minutes are required. With these 200 CPUs, we are able to perform all 15 trials simultaneously. 15 CPUs are needed and 185 other CPUs are idle. In other words, a calibration experiment using one of the global optimization methods, will take at least 4000 minutes.

For a similar calibration procedure using SMO methods, the 400 simulations can be spread between 200 CPUs. Since the entire calibration was done in only 3 iterations each trial takes about 30 minutes and all 15 trials take 300 minutes. Assuming that both methods, DDS and SMO, are as efficient, this is about one order of magnitude less than what a good global optimization method is taking.

The other benefit of SMO methods are the thorough search of the region with acceptable results. This search provides useful information about the sensitivity of the model in that region as well. In section 5.3.6, we will discuss the usefulness of these results for performing sensitivity analysis.

In general, in terms of number of function evaluations ARFM performs slightly better than ADACE and considerably better than ARFM. The mean of obtained desirable NS in all 15 trials was around 0.69. This is considered a very good result for calibration of WATCLASS model over Smoky River basin.

5.2.4 Calibration of WATCLASS with forty-four parameters

In this set of experiments, the performance of adopted SMOs for the calibration of WATCLASS with forty-four parameters are examined. Since the dimension of the problem is almost triple of the pervious experiment, it is expected to encounter a much more complex model. Usually the effect of dimension growth on the number of training data is exponential (e.g. if the dimension is 3 and it is doubled, the number of training data that needed is $6^2 = 36$ not $6 \times 2 = 12$). So model optimization becomes more difficult when the dimension of the model increases. This is a very well known issue and is known as the curse of dimensionality.

ADACE

ADACE was conducted in 15 trials and each trial was constructed with 400 training data points randomly selected from a set of 3000 generated LHS samples. The stopping criterion of this algorithm was to reach to NS number of 0.69 ± 0.01. Since the points were very coarsely grided, the shrinkage strategy was employed in the first iteration. That is, in all trials 100 points were randomly generated and simulated in the close vicinity of the best achieved point in the primary LHS sample. This strategy helped the surrogate model to model points that are more concentrated in a region with high NSs. Otherwise, it was required to do a large number of simulations to explore the entire WATCLASS space and this process needed an overwhelming number of function evaluations. The best 400 points from the new set of all points (400 +100) were used for the construction of DACE model. Based on the validation error results, the best error and a regression function for this experiment were the same as for the sixteen parameters. This fact can be used toward future calibration of WATCLASS. WATCLASS performance regardless of the number of parameters is following a general trend. Hence, these basis functions are most probably staying the same for different choices of land classes and parameters for Smoky watershed.

The computational cost of constructing and training of DACE were significantly high. In each iteration $10,000$ points were randomly generated using LHS and their function values based on the surrogate model were computed. Then, 100 best points among them were selected and simulated with WATCLASS. In the shrinkage strategy, 100 points around the best point were randomly generated and simulated as well.

Most trials could not achieve satisfactory results after 2 iterations and the process continued for the third iteration, so that the shrinkage strategy was employed. On average with 673 function evaluations an acceptable NS result was achieved that was almost 3 times more than what was needed for calibrating WATCLASS with 16 parameters. These results show that ADACE model is not very sensitive to the model dimensions and expect to perform well with high dimensional model. Other than its very demanding model construction procedure, it performed satisfactorily in the higher dimensions.

Table 5.6 presents the results for this set of experiments.

ARBF

The calibration process using ARBF started with 15 trials of 900 training data. The stopping criterion was to achieve an NS number of about 0.69 ± 0.01 as previous experiment. After performing the first 100 simulations around the best point, a matrix of input-output data was sorted and 900 parameters sets with highest NS numbers were selected. A RBF

Table 5.6: The results of ADACE on WATCLASS with 44 parameters

Maximum value of NS	0.71
Minimum value of NS	0.68
Mean of NS for 15 trials	0.6869
Variance of NS for 15 trials	6.6771e-005
Mean of No. function evaluations to reach NS=0.68	673

surrogate model was constructed based on this new set of data. 10, 000 samples using LHS were generated and the best 100 points of these 10,000 with regards to the RBF surrogate model was selected and simulated using WATCLASS. If the best point met the error criterion, the process terminated otherwise continued until an acceptable results were obtained. Since the process of 100 WATCLASS simulations on SHARCNET were performing simultaneously, there was a possibility of obtaining multiple good results at the same time. This process was inevitable and happens in few cases.

On average the ARBF surrogate model achieved a satisfactory result after 1090 simulations. Most of the trials achieved a NS number greater than 0.68 after 2 iterations. The results are shown in Table 5.7.

Table 5.7: The results of ARBF on WATCLASS with 44 parameters

Maximum value of NS[9]	0.7
Minimum value of NS[10]	0.687
Mean of NS for 15 trials	0.691
Variance of NS for 15 trials	8.1e-006
Mean of No. function evaluations to reach NS=0.68	1090

As it was expected, ARBF required more function evaluations compared to ADACE.

However, its model training and validating procedure was very simple and fast. Overall, this model is preferred over ADACE in cases that large number of initial points are available. Since it is expected that on average after 190 function evaluations 2 iterations the satisfactory results are achieved. However, in the case of ADACE case on average 273 function evaluations 3 iterations plus a comparably huge computational effort for model validation should be put to achieve desirable results.

ARFM

The number of training points for ARFM with the optimum validation error were 400. The procedure of this set of experiments was very similar to the ADACE, except that the training time was comparably lower than ADACE model. Most trials converged after 2 iterations and the shrinkage strategy was only applied to 2 trials out of 15 . The results are shown in Table 5.8.

Table 5.8: The results of ARFM on WATCLASS with 44 parameters

Maximum value of NS	0.69
Minimum value of NS	0.679
Mean of NS for 15 trials	0.683
Variance of NS for 15 trials	1.9286e-005
Mean of No. function evaluations to reach NS=0.685	613

5.2.5 Discussion of the WATCLASS Calibration Results-Part II

In this part of the experiments, WATCLASS with 44 parameters was calibrated. Compared to global optimization methods such as DDS (Tolson & Shoemaker, 2007b), which needed 1000 function evaluations on average (we did a set of 15 trials experiment calibrating WATCLASS over Smoky using DDS), SMO performs really well. All 3 SMOs performed well for calibration of WATCLASS over the Smoky basin.

ARBF as shown in Table 5.7 was not computationally as efficient as two other SMOs. Its comparably large number of function evaluations stem from its weakness in modeling higher dimensional data. The number of training data required for validating RBF was almost double of what was needed for other two SMOs.

ADACE and ARFM share a similar concept for smoothness of the model which is totally different from ARBF approach. The promising performance of ADACE method endorses the assumption of previous modelers such as Schonlau *et al.*(Schonlau et al., 1998) about the modeling errors. It is very likely that errors in such physical models are the result of modeling error and not measurement error. Thus, errors in their nature are not white noise and are mostly correlated. ADACE reached a NS number of 0.71 which was the highest achieved NS among all these three SMOs.

For ARFM, only 613 function evaluations were needed to calibrate WATCLASS with 44 parameters which was the minimum among these three SMOs. Moreover, its validation process was very straightforward and compared to complicated construction and validation process of ADACE. Contrary to ADACE, number of ARFM model parameters were very small which made the process of model construction and validation very fast and robust.

If we follow the same assumptions as section 5.2.3, with 200 CPUs where each model simulation takes about 15 minutes. 15 trials of WATCLASS calibration using ARFM need 720 CPU minutes, where an outstanding global optimization method such as DDS needs 150000 CPU minutes. This is more than two orders of magnitudes more efficient.

ADACE needed 673 function evaluations to achieve reasonable results. However, the ADACE training time for a higher dimension (which was 44 here) was five times more than the previous example (16 parameters) is a main drawback of this method.

In general, these experiments using SMOs show the potential of SMO methods and its high compatibility with available computational tools.

5.3 Results of Sensitivity Analysis

This section presents the sensitivity analysis of WATCLASS over the Smoky River watershed using couple of mentioned methods. The first part of this section presents the sensitivity analysis results of sixteen parameter WATCLASS model. The main reason for choosing only one set of parameters and assessing their sensitivity was the high computational cost of the process. In addition to that, this set can nearly represent the other set of 44 parameters. In case that couple of the SA methods yielded interesting results it is possible to apply these SA methods to the higher dimensional model. Therefore, for the preliminary SA tests, the focus is only on the parameters of the first WATCLASS model.

In the second part, the results of applying the new strategy for reducing computational cost of calibration are compared with the case that this strategy is not employed.

5.3.1 Morris Method

In this set of experiments, the Morris method was applied to evaluate first-order parameter sensitivities. For each set of runs, all of the parameters were kept constant except one. Each parameter interval was divided into eight levels; i realization of parameter x, x_i, is computed as following:

$$x_i = LB_i + \frac{(UB_i - LB_i) \times p}{P} \tag{5.1}$$

where LB_i and UB_i are lower limit and upper limit of ith parameter value, p is an integer $0 \leq p \leq P$ which is randomly changing in the interval. P as is shown is upper limit of p. About 600 simulations for each parameter are done and 300 Elementary Effect for each parameter computed (for each Elementary Effect, two function evaluations are required). The chosen criterion for function evaluation is the NS coefficient and the variation of the NS coefficient with respect to each parameter is calculated. So, the Elementary Effect is calculated based on these NS coefficients. Mean and variance of the Elementary Effect corresponding to each parameter is reported.

The main issue for calculation of mean in the Morris method is when it deals with non-monotonic models. In such models, one expects to have negative Elementary Effect which could be canceled out with the positive ones. To avoid this from occurring, it is possible to compute mean and variance for absolute values, which is called G distribution.

Table 5.9: Morris

parameter name	F distribution Mean	F distribution Variance	G distribution mean	G distribution Variance
drnrow	-0.0012	2.1946e-005	0.0020	1.9203e-005
sdeprow	-.0013	1.0568e-004	0.0035	9.5180e-005
ddenrow	0.0614	0.0233	0.0615	0.0233
xslprow	0.0300	0.0037	0.0303	0.0036
grkfrow	0.0034	3.9734e-005	0.0035	3.8620e-005
wfsrow	-0.0010	1.8088e-004	0.0010	1.8088e-004
wfcirow	0.0495	0.0066	0.0497	0.0066
sand1	0.0267	0.0096	0.0308	0.0094
clay1	-0.01777	7.6981e-004	0.0182	7.5180e-004
orgm1	-1.3651e-004	1.2174e-006	6.0429e-004	8.6970e-007
sand2	0.0088	0.0013	0.0211	9.0789e-004
clay2	-0.0051	5.2016e-004	0.0149	3.2372e-004
orgm2	-1.3767e-004	3.2271e-006	7.5189e-004	2.6789e-006
sand3	-0.0162	4.7701e-004	0.0186	3.9276e-004
clay3	0.0108	4.1514e-004	0.0122	3.8403e-004
orgm3	-4.6111e-004	4.4935e-006	9.3545e-004	3.8288e-006

Based on the results of Table 5.9, parameters related to the organic part of the soil layer had the smallest mean and least variance in the set of sixteen parameters. This means that they have the least impact on the output among other factors. Based on expert knowledge, this conclusion is valid since the volume of sand and clay have more impact on circulation of water in a soil column than organic soil.

Based on this test, the two most influential parameters are drainage density and saturated hydraulic conductivity at the surface. Their higher means with respect to other parameter means demonstrate the influence of these two parameters on the output and their high variances suggest high interaction with other parameters. The difference is not very significant and does not suggest omitting other parameters for calibration process. These results agree well with our understanding of the process.

This experiment demonstrates that, on average, which parameter is more important than the other. This result is very informative for the modeler who is interested to see if his understanding of his model matches well with the results. Since this technique is a global sensitivity analysis method, it is not possible to discover parameter regions with more sensitivity which is usually the case in nonlinear model using this method.

5.3.2 Standard Regression Coefficients (SRC)

This test uses the results of 4500 simulations using LHS. The entire space of parameters-NS were modeled using a linear regression model and the model goodness was evaluated using R^2 or coefficient of determination. Since R^2 value was around 0.58, applying regression for SA is not unreasonable, though R^2 is on border.

This test, similar to Morris method gives the overall importance of each of the parameters in the output. This method gives very good results for linear models. Border R^2 of this results endorses the assumption of previous results that NS numbers are expected to behave nonlinearly with respect to its parameters.

Table 5.10: Standard Regression Coefficients (SRC)

Rank of importance	parameter name	β coefficient
1	ddenrow	0.5353
2	wfcirow	0.3890
3	xslprow	0.3433
4	clay3	0.1802
5	grkfrow	0.1434
6	sand1	0.1336
7	sdeprow	0.1109
8	sand2	0.0911
9	orgm1	0.0713
10	wfsrow	0.0609
11	orgm2	0.0589
12	orgm3	0.0513
13	drnrow	0.0481
14	clay2	0.0443
15	sand3	-0.0377
16	clay1	-0.0509

Both Morris and SRC demonstrate that drainage density is shown to be the most influential parameter and this agrees well with expert expectation. Furthermore, organic soil is shown to have small impact on the model has a low rank in both methods respectively.

This technique is understandable, easy to implement and useful as preliminary inspection.

5.3.3 Kolmogorov-Smirnov

For this experiment, a Kolmogorov-Smirnov (KS) test was conducted for the results of Monte Carlo simulations with 7500 samples. The empirical distributions of factors that generate high Nash coefficients ≥ 0.65 was compared with its complement set[11]. Table

[11] complement set is the set of samples that generate NS less than 0.65

5.11 presents a comparison of the level of significance. These ranks can be interpreted as an indication of the impact of each factor on the output results.

Table 5.11: Kolmogorov-Smirnov test

parameter name	rank
ddenrow	1
wfcirow	2
xslprow	3
clay1	4
clay3	5
clay2	6
sand1	7
sand3	8
grkfrow	9
sdeprow	10
drnrow	11
sand2	12
orgm2	13
orgm1	14
orgm3	15
wfsrow	16

Based on values of d, three parameters, drainage density, lateral saturated hydraulic conductivity at the surface and percent of clay volume in first layer, are slightly more sensitive than other parameters of the model. More precisely, the original distribution of the sample for these three parameters differs a little from the distribution that generates well-behaved model output. This test identifies the difference between two distributions, but it does not give any more information about the place where the difference lies, whether it is in the middle or in the tail.

KS test results suggest that the sensitivity of the last seven parameters of the model is not considerably high. Overall, even the distribution of the first three parameters was just slightly different from their original distributions. This might be due to almost similar sensitivity of all model parameters to the output.

5.3.4 Results of entropy tests for grid-based sampling

In order to investigate the association of each of the parameters with the output a grid-based entropy test was performed. To find the association of the output with each of the parameters, a Monte-Carlo experiments for the sixteen parameters WATCLASS was performed. The normalized entropy based uncertainty coefficient of all parameters were in the same range, the results are shown in Table 5.12. To ensure the correctness of the results, this test was repeated 20 times on randomly selected samples. The sensitivity of none of the parameters was noticeable. Uncertainty coefficients of parameters for acceptable outputs and output having a NS number greater than 0.2 are obtained as well. This time the results did not change either, so again no significant sensitivity to any of the parameters was discovered. This test endorsed the results of previous association-based or rank-based experiments. The model did not show significant sensitivity to any of the parameters, so it does not seem possible to omit any of the parameters from the list. None of the parameters have hugely more association with the model output than the others and the distribution of none of the parameters for acceptable results was significantly different from its original distribution.

Table 5.12: Entropy test

Parameter name	Uncertainty Coefficient	Uncertainty Coefficient NS > 0.2	Uncertainty Coefficient NS > 0.6
drnrow	0.5047	0.4851	0.5668
sdeprow	0.5088	0.4859	0.5674
ddenrow	0.5060	0.4884	0.5631
xslprow	0.5070	0.4883	0.5623
grkfrow	0.5071	0.4917	0.5774
wfsrow	0.5065	0.4897	0.5668
wfcirow	0.5037	0.4856	0.5608
sand1	0.5086	0.4963	0.5699
clay1	0.5072	0.4990	0.5850
orgm1	0.5024	0.4861	0.5634
sand2	0.5068	0.4881	0.5631
clay2	0.5299	0.5045	0.5869
orgm2	0.5046	0.4891	0.5691
sand3	0.5050	0.4869	0.5613
clay3	0.5071	0.4877	0.5671
orgm3	0.5044	0.4863	0.5633

5.3.5 Discussion of the first part of Sensitivity Analysis

Based on this experiment, it is concluded that the WATCLASS model does not suffer from the so-called parameterization problem that exists in most physical models. This might stem from cross-correlation/association of the parameters together.

Based on the results of previous methods, drainage density is the most important parameter in this model. WATCLASS modelers also acknowledge this importance. This information is very helpful for the sake of scientific insight of the model development but it is not considered to be very helpful in terms of reducing computational cost of calibration. As pointed out before, one of the main purposes of SA in this document is the reduction of the calibration computational cost. It is clear that knowing the importance of one parameter from 16/44 is not very helpful.

The following section discusses a strategy for reducing computational cost of calibration process and presents the calibration results using these strategies. In addition to that, a new technique based on previously developed RFM calibration method is tailored for conducting model sensitivity around the optimum.

5.3.6 Sensitivity Analysis using SMO

The results that have been presented in Tables 5.3 to 5.8 not only show the promising performance of SMO in calibration of WATCLASS, but also indicate the sensitivity of WATCLASS around the optimum.

As demonstrated in the Flowchart in Figure 3.2, as the process reaches its termination the surrogate model becomes more concentrated in the optimum region. SMO results, as presented in mentioned tables, provide very helpful information about the region of optimum. These results not only show the maximum and minimum of the achieved NS in the close vicinity of the optimum but also provide the variance of the results in that region. These results, similar to Taylor sensitivity index, helps the modeler to evaluate model stability in the region of the optimum. The advantage of SMO approach is that it provides reliable sensitivity information without additional computational cost.

Using SMO as calibration demonstrates that model performs stable in the neighborhood of optimum and slight changes in the parameter values will not have an impact on NS. So, there comes another advantage of SMO approach, which is using Taylor series expansions for calculating sensitivity of the model around a nominal value. Using derivative of SM and substitute it in the Taylor formula (see Appendix .1) can work as simulation model derivatives but with extra computational cost.

5.3.7 Results of WATCLASS calibration using PCA

The results of PCA coupled with RFM

The results of ADACE combined with PCA have been presented in (Kamali et al., 2007). In this paper, the previous version of WATCLASS (for more information refer to section 4.2.2) with 14 parameters has been calibrated using PCA. This paper compares the performance of LHS, LHS with DACE and LHS with DACE and PCA when all of these three approaches are used to find an acceptable results. These results show that PCA has improved the performance of ADACE.

However, since DACE did not have a closed form formula, it was not possible to provide a simple formula for coupling DACE and PCA. A DACE surrogate model was constructed in the lower dimension and its minimum was found. Then that minimum was mapped to the higher dimension and its corresponding WATCLASS result was found. The main advantage of using DACE in the lower dimension was the facilitation of DACE model construction and validation. DACE correlation function requires 2k parameters to be estimated which in the lower dimension the number of parameters will be largely reduced are is 2l.

The coupling of DACE and RFM was more straightforward. As was shown in Eq. 3.16, the input in the lower dimension substitutes the original model input. For conducting this set of experiments, the threshold of NS number was set to 0.68 ± 0.01. This experiment was repeated 15 times. The process started with 300 training data and after two iterations all trials achieved an acceptable results.

Table 5.13 compares the results of using RFM alone and when it is coupled with PCA. These results demonstrate that conducting calibration in the lower dimension hugely reduced the computational cost as the number of function evaluation is almost half in the case that PCA is coupled with RFM. But the the lower computational cost comes with an expense and it is the higher variance and lower mean of NS. That is, this time surrogate model map a larger space of the parameters and hence, the points around the best point might have not acceptable NS numbers.

Table 5.13: The comparison of the results of RFM and PCA coupled with ARFM on WATCLASS with 44 parameters

	ARFM and PCA	ARFM
Maximum value of NS	0.682	0.69
Minimum value of NS	0.12	0.679
Mean of NS for 15 trials	0.60	0.683
Variance of NS for 15 trials	2e-002	1.9286e-005
Mean of No. function evaluations to reach NS=0.679	385	613

Discussion

This set of experiments illustrates the promising results of surrogate model optimization methods for calibration of large-scale computationally expensive models. Not only did the performance of SMO for higher dimension improve but also better results were achieved. Using SHARCNET facilities, we were able to conduct thousands of parallel simulations simultaneously. This facility offers the possibility of accompanying model calibration with fast pace model updates.

The comparison of the performance of the three SMO methods are briefly presented in Table 5.14.

In general, in terms of number of function evaluations and number of iterations, ARFM works slightly better than ADACE and ARBF. ADACE performance is very similar to

Table 5.14: The comparison of DACE, RBF and RFM SMOS

	Advantages	Disadvantages
DACE SMO	Small number of training data	Large number of model parameters, time consuming process of model construction and model validation
RBF SMO	Straightforward process of model validation and model construction	Large number of training data and large number of model parameters
RFM SMO	Small number of training data and few model parameters	Pseudo inverse might be required in some cases

ARFM, except the model construction and validation process which is a difficult, time consuming process. RBF is very a very straight-forward SMO, once its basis functions and error functions are known. However, finding a proper basis function and error function is very challenging and time consuming. The overall results show the potential of adapted SMOs for 16 and 44 parameters WATCLASS and possibility of obtaining similar results for WATCLASS with higher number of parameters. This calibration results could be used as starting point for calibration of watersheds similar to Smoky and save the modelers lots of effort.

In the second part of experiments PCA is coupled with ARFM. The results show the effectiveness of this combination in improvement of number of function evaluations. This is the most efficient way of coming up with acceptable results.

5.4 Summary

This chapter presents the results of the calibration of WATCLASS using the proposed methods. One SMO that was developed and is called RFM and two that were adapted for distributed computing framework. In total two instances of WATCLASS were calibrated one with sixteen parameters in the first part and the other one with forty-four parameters in the second part.

In the second part of this chapter the results of several sensitivity analysis methods were presented and discussed. This was followed by a new approach for calibration of WATCLASS with even less computational effort. In this approach, RFM has been coupled with principal component analysis and its results on calibration of WATCLASS model are presented.

Chapter 6

Conclusion

6.1 Summary and Conclusion

The goal of this research was to develop a computationally efficient calibration framework compatible with the available distributed computational resources such as SHARCNET [1]. Moreover, this framework had to be flexible, robust and compatible with the future updates of the model that is calibrated. A calibration framework based on the Surrogate Model Optimization (SMO) approach suitable for distributed computers is proposed. The main case study of this research was WATCLASS hydrologic model.

The proposed calibration method contains all the above characteristics and is easy to understand and implement. This framework was mainly distributed surrogate model optimization in which two widely used SMO, DACE and RBF, were employed. A new SMO method was also developed to overcome the shortcomings of DACE and RBF. The general framework of this new method is based on the least squares regression with an addition of a smoothness constraint that complies with the expected smoothness of physically based hydrologic models. This proposed method is called Regularized Function Map

[1] `http://www.sharcnet.ca/index.php`

(RFM) and has the advantage of having a closed form optimum. To evaluate the performance of RFM for optimization, a set of preliminary experiments on five well-known test functions were conducted. The results of using DACE, RBF and RFM for optimization of these five test functions were promising and encouraged us to adapt all of them for distributed computing facilities. The adapted versions called ADACE, ARBF and ARFM were also applied to optimize those five test functions. The results of employing them for optimizing the test functions indicated that these techniques have the potential of being used in calibration of large scale models. Hence, two instances of WATCLASS model were calibrated using ADACE, ARBF abd ARFM. The distributed calibration process was conducted using SHARCNET distributed computing facility.

In general, the calibration process using the proposed SMOs and their extensions for distributed computers showed to be more efficient than previously employed global optimization techniques. By using the SMO approach and tailoring three SMO techniques for distributed computing facilities, we reduce the calibration time to 10% of what a global optimization method takes on average. In general, calibration of WATCLASS over Smoky with 16 parameters that took 45 days in the beginning of this study now takes 5 hours only, and calibration of WATCLASS over Smoky with 44 parameters that took 100 days at first now takes only 12 hours.

6.2 Contribution

The main contributions of this thesis can be summarized as following:

- Calibration and sensitivity analysis of WATCLASS: The first time that numerical investigations are done on WATCLASS hydrologic model. Throughout the course of this study, the results of sensitivity analysis and calibration have helped the modelers to capture and fix some of WATCLASS shortcomings. Hence, those preliminary results were vital for the development and improvement process of WATCLASS.

- Development of Regularized Function Map (RFM) surrogate model optimization technique; After some preliminary experiments with two widely used SMOs, RBF and DACE, we realized that none of these models are designed for large scale high dimensional models. Hence, we came up with a new approach that can handle models with these two characteristics. This new technique is based on the idea of regularized regression and is called Regularized Function Map (RFM).

- Adapting RFM and DACE and RBF surrogate models for distributed computing services; A general framework is offered that suits every SMO. Specifically, we modified DACE, RBF and RFM.

- PCA coupled with RFM; A new calibration technique for performing calibration on the lower dimension was performed by coupling Principal Component Analysis (PCA) and RFM.

6.3 Future Work

This research presents a new perspective for problem of hydrologic model calibration and opens the doors for conducting future research:

- **WATCLASS** During this research we helped the modelers to find imperfections in WATCLASS hydrologic model by conducting some statistical tests. We believe that like all physically based models, WATCLASS algorithms can improve to generate even better results. Moreover, modeling other watersheds using WATCLASS will help us to better discover this model unknown aspects.

- **Sensitivity Analysis** It is possible to merge Taylor Expansion with Surrogate Models to achieve more accurate results with lower computational cost. It is also possible to modify the SMO codes for generating sensitivity of the model while performing calibration.

- **Optimization** The only expensive model that was calibrated using the proposed framework was WATCLASS. There are many other potential applications of this technique in other engineering fields.

APPENDICES

.1 Taylor Expansion

If $Y = F(x)$, Taylor expansion of $F(x)$ with respect to the $F(a)$ is as follows:

$$F(x) = F(a) + \frac{F^{'}(a)}{(1)}(x-a) + \frac{F^{''}(a)}{(2)}(x-a)^2 + \cdots + \frac{F^{n}(a)}{(n)}(x-a)^n. \tag{1}$$

The main virtue of Taylor expansion is its precise estimation of function with a couple of phrases. In most cases, computation of up to three terms are adequate, this kind of estimation is called second-order Taylor estimation. In many cases only two terms are computed which is first-order Taylor estimation.

With Taylor expansion, it is possible to estimate mean and variance of the process using input information. First-order approximation of the expectation is $E(Y) \approx F(\mu_x)$ and second- order approximation of the expectation is as follows:

$$E(Y) \approx F(\mu_x) + \frac{F^{''}(\mu_x)}{2!}\sigma_x^2. \tag{2}$$

Taylor series expansion for multidimensional data is defined as:

$$F(X) = F(a) + \delta f(a)^T(X-a) + \frac{1}{2}\delta^2 f(a)(X-a) + \cdots \tag{3}$$

.

Where $X = (x_1, x_2, \cdots, x_k)$, $a = (a_1, a_2, \cdots, a_k)$ and X and a are two vectors in parameter space. $\delta f(a)$ is the gradient vector defined as : $\delta f(a) = (\frac{\partial f}{a_1}, \frac{\partial f}{a_2}, \cdots, \frac{\partial f}{a_k})$. The second order derivative,$\delta^2(x)$ is called Hessian matrix and is defined as:

$$H(f) = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f}{\partial x_1 \partial x_2} & \frac{\partial^2 f}{\partial x_2^2} & \cdots & \frac{\partial^2 f}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \frac{\partial^2 f}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_n^2} \end{bmatrix} \tag{4}$$

First-order approximation of the variance is:

$$Var(Y) \approx F^{'}(EX)^2 Var(X) \tag{5}$$

and second-order approximation of the variance is :

$$Var(Y) \approx F^{''}(EX)^2 Var(X^2) - 4a^2(F^{''}(EX)^2 Var(X). \tag{6}$$

When the target is the statistical properties of a simulation and direct derivative is not available, it is possible to calculate derivatives using finite difference method. There are three common types of finite difference:

- Forward difference: $\delta F(x) = F(x + \delta x) - F(x)$,
- Backward difference: $\delta F(x) = F(x - \delta x) - F(x)$,
- Central difference: $\delta F(x) = F(x + \frac{1}{2}\delta x) - F(x - \frac{1}{2}\delta x)$.

Finite difference method offers a useful tool for computation of the derivatives. Skipping details, using central difference technique; $F\prime(x) = \frac{F(x+\frac{1}{2}\delta x) - F(x-\frac{1}{2}\delta x)}{(\delta x)}$. $F\prime(x)$ value will be very close to the real value, if δx be in close vicinity of x. For second derivative, using central difference;

$$F''(x) = \frac{F(x + \delta x) - 2F(x) + F(x - \delta x)}{(\delta x)^2} \tag{7}$$

.2 Model Validation

An essential part of surrogate model optimization is the mathematical model construction. A good model produces reasonable error on new, unseen data, but the question is how many data points are needed to avoid overfitting or underfitting? Overfitting and underfitting are two terms that characterize the behavior of models. Overfitting is fitting a model with high complexity (a large number of model parameters) over data when a model with lower complexity is good enough. This phenomenon disables the generalization property of the model. One of the main indications used for detection of overfitting, is small error on training data but large prediction error. Underfitting, on the other hand, is where a model with low complexity is fitted over the data. In general, a model that exhibits a balance between prediction error and training is considered good.

All models regardless of the category they belong to have some parameters that need to be tuned for each application. For example in radial basis functions, a basis function is selected and then the center of the radial basis function or other model coefficients needs to be tuned. In addition, the number of training data that result in a good model should be found.

Hence, the main part of model construction becomes model validation, or the performance of the constructed model on new data points. One of the common and accurate model validation methods is cross validation ((Stone, 1974) and (Kohavi, 1995)). In this method, the accuracy of the model is assessed without further sampling and function evaluations. The simplest form of cross validation is the leave-one-data-out method. In this method, one data point is left aside and the model is constructed using other $n-1$ data points. Model prediction on the left data point is compared to its real value and the squared error is evaluated. This process continues for n times and the mean square error is computed. This error should be less than the threshold, in order to validate the model.

Since all model parameters are going to be estimated n times, the computational cost of this process is high. To avoid this computational cost, the k-fold cross validation method proposed by (Breiman et al., 1984) was used with $k \ll n$. In this technique, training data is divided into k partitions and the model is constructed based on the $k-1$ partitions and is tried on the left out partition. This process is continued k times and the average error over k partitions is measured.

Since k-fold model validation was less computationally intensive, this technique was employed to evaluate the goodness of each surrogate model.

.3 Abbreviation

SMO	Surrogate Model Optimization
SM	Surrogate Model
RBF	Radial Basis Function
DACE	Design and Analysis of Computer Experiment
RFM	Regularized Function Map
ADACE	Adoptive Design and Analysis of Computer Experiment
ARBF	Adoptive Radial Basis Function
ARFM	Adoptive Regularized Function Map
NS number	Nash-Sutcliff number
PCA	Principal Component Analysis
SA	Sensitivity Analysis
Corr	correlation
LHS	Latin Hypercube Sampling
GSA	Global Sensitivity Analysis
LSA	Local Sensitivity Analysis
OAT	One-Factor-At-A-Time
ASFDC	Adopted Surrogate Function for Distributed Computing

Bibliography

Barlett, P., MacKay, M., & Verseghy, D. (2006). Modified snow algorithms in the canadian land surface scheme: Model runs and sensitivity analysis at three boreal forest stands. *Atmosphere - Ocean*, *44*, 207–222.

Barlett, P., McCaughey, J., Lafleur, P., & Verseghy, D. (2003). Modelling evapotranspiration at three boreal forest stands using the class: Tests of parameterizations for canopy conductance and soil evaporation. *International Journal of Climatology*, *23*, 427–451.

Bau, D., & Mayer, A. (2006). Stochastic management of pump-and-treat strategies using surrogate functions. *Advances in Water Resources*, *29*, 1901–1917.

Belkin, M., Matveeva, I., & Niyogi, P. (2003). *Regression and regularization on large graphs* (Technical Report). University of Chicago, Department of Mathematics.

Borgonovo, E., & Peccati, L. (2004). Sensitivity analysis in investment project evaluation. *International Journal of Production Economics*, *90*, 17–25.

Box, G. (1954). The exploration and exploitation of response surfaces: some general considerations and examples. *Biometrics*, *10*.

Box, G., & Draper, N. (1987). *Empirical model-building and response surfaces.* John Wiley and Sons, Inc, New York.

Boyle, D., Gupta, H., & Sorooshian, S. (2000). Toward improved calibration of hydrologic models: combining the strength of manual and automatic methods. *Water Rresources Research*, *36*, 3663–74.

Breiman, L., Friedman, J., Olshen, R., & Stone, C. (1984). *Classification and regression trees.* Wadsworth International Group, Belmont, California.

Buhmann, M. (2003). *Radial basis functions.* Cambridge University Press, U.K.

Campolongo, F. (2007). An effective screening design for sensitivity analysis of large models. *Environmental Modelling and Software*, *22*, 1509–1518.

Campolongo, F., & Saltelli, A. (1997). Sensitivity analysis of an environmental model: an application of different analysis methods. *Reliability Engineering and System Safety*, *57*, 49–69.

Campolongo, F., Saltelli, A., Jensen, N., Wilson, J., & Hjorth, J. (1993). Sensitivity analysis for nonlinear mathematical models. *Mathematical Modeling and Computational Experiment*, *1*, 407–414.

Castillo, E., Fernandez-Cantelt, A., Hadi, A., & Lopez-Aenlle, M. (2007). A fatigue model with local sensitivity analysis. *Fracture of Engineering Materials*, *30*, 149–168(20).

Chen, V., Tsui, K., Barton, R., & Meckesheimer, M. (2006). A review on design, modeling and applications of computer experiments. *IIE Transactions*, *38*, 273 291.

Cressie, N. (1993). *Statistics for spatial data.* John Wiley and Sons, Inc, New York.

Darling, D. (1957). The kolmogorov-smirnov, cramer-von mises tests. *The Annals of Mathematical Statistics*, *28*, 823–838.

Demaria, E., Nijssen, B., & Wagener, T. (2007). Monte carlo sensitivity analysis of land surface parameters using the variable infiltration capacity model. *Journal of Geophysical Research*, *112*.

Dixon, L., & Szego, G. (1978). *The global optimization problem: An introduction, toward global optimization 2*. North-Holland, New York.

Draper, N., & Smith, H. (1981). *Applied Regression Analysis*. John Wiley and Sons, New York.

Duan, Q. (2004). *Global optimization for watershed model calibration*. Water Science and Applications Series.

Duan, Q., Gupta, V., & Sorooshian, S. (1993). A shuffled complex evolution approach for effective and efficient optimization. *Journal of Optimization Theory and Applications*, *76*, 501–521.

Duan, Q., Sorooshian, S., & Gupta, V. (1992). Effective and efficient global optimization for conceptual rainfall-runoff models. *Water Rresources Research*, *4*, 1015–31.

Encica, L., Echeverra, D., Lomonova, E. A., Vandenput, A. J. A., Hemker, P. W., & Lahaye, D. (2007). Efficient optimal design of electromagnetic actuators using space mapping. *Structural and Multidisciplinary Optimization*, *33*, 481–491.

Fraser, A., & Burnell, D. (1970). *Computer models in genetics*. McGraw-Hill, New York.

Friedman, J. (1991). Multivariate adaptive regression splines. *Annals of Statistics*, *19*, 1141.

Gershenfeld, N. A. (1999). *The nature of mathematical modeling*, chapter 13, 166. Cambridge University Press.

Gupta, H., Sorooshian, S., & Yapo, P. (1998). Toward improved calibration of hydrologic models: multiple and noncommensurable measures of information. *Water Rresources Research*, *4*, 751–763.

Gutmann, H. (2001a). On the semi-norm of radial basis function interpolants. *Journal of Approximation Theory*, *111*, 315–328.

Gutmann, H. (2001b). A radial basis function method for global optimization. *Journal of Global Optimization*, *19*, 201–227.

Hamby, D. (1994). A review of techniques for parameter sensitivity analysis of environmental models. *Environmental Monitoring and Assessment*, *32*, 1573–2959.

Helton, J., Johnson, J., Sallaberry, C., & Storlie, C. (2006). Survey of sampling-based methods for uncertainty and sensitivity analysis. *Reliability Engineering and System Safety*, *91*, 1175–1209.

Holland, J. (1975). *Adaptation in natural and artificial system*. The University of Michigan Press, Ann Arbor.

Huang, D., Allen, T., Notz, W., & Zeng, N. (2006). Global optimization of stochastic black-box systems via sequential kriging meta-models. *Journal of Global Optimization*, *34*, 441–466.

Kamali, M., Ponnambalam, K., & Soulis, E. (2005). Hydrologic model calibration using fuzzy tsk surrogate model. *Fuzzy Information Processing Society, 2005. NAFIPS 2005. Annual Meeting of the North American*, 799–803.

Kamali, M., Ponnambalam, K., & Soulis, E. (2007). Integration of surrogate optimization and pca for calibration of hydrologic models, a watclass case study. *Systems, Man and Cybernetics, 2007. ISIC. IEEE International Conference*, 2733–2737.

Kan, H., & Timmer, G. (1987). Stochastic global optimization methods part i: Clustering methods. *Mathematical Programming*, *39*, 27–56.

Kleijnen, J. (2007). Kriging metamodeling in simulation: A review. *European Journal of Operational Research*.

Kohavi, R. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection (pp. 1137–1143.). Morgan Kaufmann.

Kohavi, R., & John, G. (1997). Wrappers for feature subset selection. *Artificial Intelligence*, *1-2*, 273–324.

Kouwen, N. (1988). Watflood: A micro-computer based flood forecasting system based on real-time weather radar. *Canadian Water Resources Journal*, *13*, 62–77.

Koziel, S., & Bandler, J. (2007). Interpolated coarse models for microwave design optimization with space mapping. *IEEE Transactions on Microwave Theory and Techniques*, *55*, 1739–1746.

Krige, D. (1951). A statistical approach to some mine valuations and allied problems at the witwatersrand. *Canadian Water Resources Journal*, *13*, 62–77.

Kruskal, J., & Wish, M.

Kushner, H. (1964). A new method of locating the maximum point of an arbitrary multipeak curve in the presence of noise. *Transactions of the ASME. series D Journal of Basic Engineering*, *86*, 97106.

Madsen, H. (2000). Automatic calibration of a conceptual rainfall-runoff model using multiple objectives. *Journal of Hydrology*, *235*, 276–88.

Madsen, H. (2003). Parameter estimation in distributed hydrological catchment modeling using automatic calibration with multiple objectives. *Advances in Water Resources*, *26*, 205–216.

Marsden, A. L., Wang, M., Jr., J. D., & Moin, P. (2004). Optimal aeroacoustic shape design using the surrogate management framework. *Optimization and Engineering*, *5*, 235–262.

Marsh, P., & Gyakum, J. R. (2003). The hydrometeorology of the mackenzie river basin during the 199899 water year. *Journal of Hydrometeorology*, *4*, 645648.

McKay, M., Conover, W., & Beckman, R. J. (1979). A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, *21*, 239–245.

Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, A., & Teller, E. (1953). Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, *21*, 1087–1092.

Morris, M. (1991). Factorial sampling plans for preliminary computational experiments. *Technometrics*, *33*, 161–162.

Mugunthan, P., & Shoemaker, C. (2006). Assessing the impacts of parameter uncertainty for computationally expensive groundwater models. *Water Resources Research*, *42*.

Mugunthan, P., Shoemaker, C., & Regis, R. (2005). Comparison of function approximation, heuristic, and derivative-based methods for automatic calibration of computationally expensive groundwater bioremediation models. *Water Resources Research*, *41*, 1–7.

Myers, R., & Montgomery, D. (1995). *Response surface methodology: Process and product optimization using designed experiments.* John Wiley and Sons, New York.

Nash, J., & Sutcliffe, J. V. (1970). River flow forecasting through conceptual models part i a discussion of principles. *Journal of Hydrology*, *10*, 282290.

Nocedal, J., & Wright, S. (1999). *Numerical optimization.* Springer.

Pearson, K. (1901). On lines and planes of closest fit to systems of points in space. *Philosophical Magazine*, *1*, 559–572.

Peter, J., Marcelet, M., Burguburu, S., & Pediroda, V. (2007). Comparison of surrogate models for the actual global optimization of a 2d turbomachinery flow. *Proceedings of the 7th WSEAS International Conference on Simulation, Modelling and Optimization* (pp. 46–51).

Powell, M. (1992). *The theory of radial basis function approximation in 1990*, vol. 2 of *Wavelets, Subdivision Algorithms and Radial Basis Functions.* Oxford University Press.

Powell, M. (1999). *Recent research at cambridge on radial basis functions*, vol. 132 of *New Developments in Approximation Theory, International Series of Numerical Mathematics.* Birkhauser Verlag, Basel.

Powell, M. (2002). Uobyqa: Unconstrainted optimization by quadratic approximation. *Mathematical Programming*, *92*, 555–582.

Regis, R., & Shoemaker, C. (2005). Constrained global optimization using radial basis functions. *Journal of Global Optimization*, *31*, 153–171.

Regis, R., & Shoemaker, C. (2007a). Improved strategies for radial basis function methods for global optimization. *Journal of Global Optimization*, *37*, 113–135.

Regis, R., & Shoemaker, C. (2007b). A stochastic radial basis function method for the global optimization of expensive functions. *Informs Journal on Computing*, *19*, 497–509.

Sacks, J., Welch, W., Mitchell, T., & Wynn, H. (1989). Design and analysis of computer experiments. *Statistical Science*, *4*, 409–423.

Saltelii, A., Ratto, M., Tarantola, S., & Campolongo, F. (2006). Sensitivity analysis practices: Strategies for model-based inference. *Reliability Engineering and System Safety*, *91*, 1109–1125.

Saltelli, A., Ratto, M., Tarantola, S., & Campolongo, F. (2005). Sensitivity analysis for chemical models. *Chemical Reviews*, *105*, 2811–2827.

Saltelli, A., Tarantola, S., & Campolongo, F. (2000). Sensitivity analysis as an ingredient of modelling. *Statistical Science*, *15*, 377–395.

Saltelli, A., Tarantola, S., Campolongo, F., & Ratto, M. (2004). *Sensitivity Analysis in practice. A guide to assessing scientific models.* New York, Wiley.

Schaback, R. (1993). Comparison of radial basis function interpolants. *In Multivariate Approximation. From CAGD to Wavelets* (pp. 293–305). World Scientific.

Schonlau, M., Jones, D., & Welch, W. (1998). Efficient global optimization of expensive black-box functions. *Journal of Global Optimization*, *13*, 455–492.

Soulis, E., & Seglenieks, F. (2007). The mags integrated modeling system. In M. Woo (Ed.), *Cold region atmospheric and hydrologic studies, the mackenzie gewex experience*, vol. 2, 445–475. Springer.

Soulis, E., Snelgrove, K., Kouwen, N., Seglenieks, F., & Verseghy, D. (2000). Toward closing the vertical water balance in canadian atmospheric models:coupling of the land surface scheme class with the distributed hydrological model watflood. *Atmosphere-Ocean*, *38*, 251–269.

Stadnyk, T., Amour, N., Kouwen, N., Edwards, T., Pietroniro, A., & Gibson, A. (2005). A groundwater separation study in boreal wetland terrain: The watflood hydrological model compared with stable isotope tracers. *Isotopes in Environmental and Health Studies*, *41*, 49–68.

Stephen, J., Bhaskar, A., & Keane, A. (2004). A derivative based surrogate model for approximating and optimizing the output of an expensive computer simulation. *Journal of Global Optimization*, *30*, 39–58.

Stone, M. (1974). Cross validatory choice and assessment of statistical predictions. *Journal of the Royal Statistical Society*, *36*, 117–147.

Sudret, B. (2008). Global sensitivity analysis using polynomial chaos expansions. *Reliability Engineering and System Safety*, *93*, 964–79.

Tolson, B., & Shoemaker, C. (2007a). Cannonsville reservoir watershed swat2000 model development, calibration and validation. *Journal of Hydrology*, *337*, 68–86.

Tolson, B., & Shoemaker, C. (2007b). Dynamically dimensioned search algorithm for computationally efficient watershed model calibration. *Water Resources Research*, *43*.

Torczon, V. (1997). On the convergence of pattern search algorithms. *SIAM Journal of Optimization*, *7*, 1–25.

Verseghy, D. L. (1991). Class-a canadian land surface scheme for gcms, i. soil model. *International journal of climatology*, *11*, 111–133.

Verseghy, D. L. (1993). Class-a canadian land surface scheme for gcms, ii vegetation model and coupled runs. *International journal of climatology*, *13*, 347–370.

Welch, W., Buck, R., Sacks, J., Wynn, H., Mitchell, T., & Morris, M. (1992). Screening, predicting, and computer experiments. *Technometrics*, *34*, 15–25.