DEVELOPING EFFICIENT STRATEGIES FOR GLOBAL SENSITIVITY ANALYSIS OF COMPLEX ENVIRONMENTAL SYSTEMS MODELS

A Thesis Submitted to the College of

Graduate and Postdoctoral Studies

In Partial Fulfillment of the Requirements

For the Degree of Doctor of Philosophy

In the School of Environment and Sustainability

University of Saskatchewan

Saskatoon

By

SEYED RAZI SHEIKHOLESLAMI

Permission to Use

In presenting this thesis/dissertation in partial fulfillment of the requirements for a Postgraduate degree from the University of Saskatchewan, I agree that the Libraries of this University may make it freely available for inspection. I further agree that permission for copying of this thesis/dissertation in any manner, in whole or in part, for scholarly purposes may be granted by the professor or professors who supervised my thesis/dissertation work or, in their absence, by the Head of the Department or the Dean of the College in which my thesis work was done. It is understood that any copying or publication or use of this thesis/dissertation or parts thereof for financial gain shall not be allowed without my written permission. It is also understood that due recognition shall be given to me and to the University of Saskatchewan in any scholarly use which may be made of any material in my thesis/dissertation.

Disclaimer

Reference in this thesis/dissertation to any specific commercial products, process, or service by trade name, trademark, manufacturer, or otherwise, does not constitute or imply its endorsement, recommendation, or favoring by the University of Saskatchewan. The views and opinions of the author expressed herein do not state or reflect those of the University of Saskatchewan and shall not be used for advertising or product endorsement purposes.

Requests for permission to copy or to make other uses of materials in this thesis/dissertation in whole or part should be addressed to:

Executive Director
School of Environment and Sustainability
University of Saskatchewan
Room 323, Kirk Hall
117 Science Place
Saskatoon, Saskatchewan S7N 5C6
Canada

OR

Dean
College of Graduate and Postdoctoral Studies
University of Saskatchewan
116 Thorvaldson Building, 110 Science Place
Saskatoon, Saskatchewan S7N 5C9
Canada

Abstract

Complex Environmental Systems Models (CESMs) have been developed and applied as vital tools to tackle the ecological, water, food, and energy crises that humanity faces, and have been used widely to support decision-making about management of the quality and quantity of Earth's resources. CESMs are often controlled by many interacting and uncertain parameters, and typically integrate data from multiple sources at different spatio-temporal scales, which make them highly complex. Global Sensitivity Analysis (GSA) techniques have proven to be promising for deepening our understanding of the model complexity and interactions between various parameters and providing helpful recommendations for further model development and data acquisition. Aside from the complexity issue, the computationally expensive nature of the CESMs precludes effective application of the existing GSA techniques in quantifying the global influence of each parameter on variability of the CESMs' outputs. This is because a comprehensive sensitivity analysis often requires performing a very large number of model runs. Therefore, there is a need to break down this barrier by the development of more efficient strategies for sensitivity analysis.

The research undertaken in this dissertation is mainly focused on alleviating the computational burden associated with GSA of the computationally expensive CESMs through developing efficiency-increasing strategies for robust sensitivity analysis. This is accomplished by: (1) proposing an efficient sequential sampling strategy for robust sampling-based analysis of CESMs; (2) developing an automated parameter grouping strategy of high-dimensional CESMs, (3) introducing a new robustness measure for convergence assessment of the GSA methods; and (4) investigating time-saving strategies for handling simulation failures/crashes during the sensitivity analysis of computationally expensive CESMs.

This dissertation provides a set of innovative numerical techniques that can be used in conjunction with any GSA algorithm and be integrated in model building and systems analysis procedures in any field where models are used. A range of analytical test functions and environmental models with varying complexity and dimensionality are utilized across this research to test the performance of the proposed methods. These methods, which are embedded in the VARS–TOOL software package, can also provide information useful for diagnostic testing, parameter identifiability analysis, model simplification, model calibration, and experimental design. They can be further applied to address a range of decision making-related problems such as characterizing the main causes of risk in the context of probabilistic risk assessment and exploring the CESMs' sensitivity to a wide range of plausible future changes (e.g., hydrometeorological conditions) in the context of scenario analysis.

Acknowledgements

I would like to take this opportunity to acknowledge a deep sense of gratitude to a number of colleagues and friends for contributing, in one form or another, to the research in this dissertation. First and foremost, I offer my sincere thanks to my supervisor, Dr. Saman Razavi, who introduced me to the subject with stimulating ideas on various aspects of sensitivity and uncertainty analysis of numerical models. His trust in me and open-mindedness to my research have made the past three and a half years a wonderful journey for me. He is more than just a supervisor, he is a friend.

I want to give special thanks to my advisory committee members for their invaluable comments and suggestions: Professor Howard Wheater, Founding Director of Global Institute for Water Security, who taught me how to see the general patterns, how to be rigorous, and how exciting it is to move between different disciplines. Regents' Professor Hoshin V Gupta, who has given me a lot of insightful thoughts and taught me valuable lessons on how to be an open-minded critical thinker. Professor Amin Elshorbagy for his extremely useful instructions and feedbacks on many aspects of this dissertation. Dr. Al Pietroniro, Executive Director of Environment and Climate Change Canada's National Hydrology Research Centre, for serving as my dissertation committee member; his insights and advice from practical and engineering viewpoints have been very useful for my research. I am also very grateful to Dr. Karl E Lindenschmidt for helpful comments and support as a chair of my dissertation committee.

My gratitude is extended to Dr. Amin Haghnegahdar, Dr. Shervan Gharari, and Mr. Fuad Yassin who shared their knowledge with me in various parts of my research. I would like to thank Professor Ali Kaveh, Dr. Siamak Talatahari, Professor Mahdi Zarghami, Professor Jeffrey McDonnell, and Professor Maureen Reed for being my sources of inspiration. My special thanks go to all my friends in the Razavi's Watershed Systems Analysis and Modelling Lab and to other members of the Global Institute for Water Security, School of Environment and Sustainability, and University of Saskatchewan. Also, I gratefully acknowledge the scholarship provided by the School of Environment and Sustainability through the Canada Excellence Research Chair in Water Security and the financial aid provided by the NSERC CREATE for Water Security program.

To my parents, Shahin Mousavian and Seyed Hossein Sheikholeslami, I owe great thanks for their encouragement, continual support, and endless efforts toward my education and for teaching me how to face things in the world and enjoy life. My warmest gratitude is due to my brother and his wife, my sister and her husband, my niece, Nafas, and my nephew, Mohammed-Taha, who helped me unconditionally throughout my entire life and for so many happy memories that we have.

وَالحَمدُ لله اولاً وَ آخراً

All praise be to God*

*Translation to English from Arabic. The calligraphy is a Persian calligraphy style known as Nastaʿlīq. For more details see https://en.wikipedia.org/wiki/Nasta%CA%BFl%C4%ABq

Dedication

This is dedicated to my parents.

In Loving Memory of My Beautiful Father…

Table of Contents

Permission to Use i

Abstract ii

Acknowledgements iii

Dedication iv

Table of Contents v

List of Figures x

List of Tables xiv

Chapter 1 Introduction 1

1.1 Background and Motivation 1

1.1.1 Challenges associated with the ever-increasing complexity of environmental models 1

1.1.2 Global sensitivity analysis to cope with outstanding challenges in CESMs 4

1.2 Objectives, Significance, and Research Questions 6

1.2.1 Objective (1): proposing a novel sequential sampling strategy 8

1.2.2 Objective (2): developing a new robustness measure and factor grouping strategy 8

1.2.3 Objective (3): crash handling in sensitivity analysis of computationally expensive models 9

1.3 Outline of the Dissertation 9

Chapter 2 Progressive Latin Hypercube Sampling for Robust sampling-based Analysis of Environmental Models 11

Synopsis 11

2.1 Introduction 12

2.2 Literature Review 15

2.2.1 What is Latin hypercube sampling? 15

2.2.2 Sampling improvements based on the notion of optimization 17

2.2.3 Input-oriented sequential sampling 19

2.3 Progressive Latin Hypercube Sampling 22

2.3.1 Definition 22

2.3.2 Mathematical formulation 22

2.3.3 Practical implementation 23
2.3.3.1 An iterative approach: doubling procedure (perfect-PLHS) 24
2.3.3.2 An optimization approach (quasi-PLHS) 25
2.3.3.3 Other criteria 27
2.4 Computational Experiments 29
2.4.1 Test problems 29
2.4.1.1 A 2-D analytical test function 29
2.4.1.2 A real-world problem 30
2.4.2 Design of experiments 31
2.4.2.1 Experiment I 31
2.4.2.2 Experiment II 31
2.4.2.3 Experiment III 32
2.4.2.4 Experiment IV 33
2.4.3 Setup for sampling strategies 34
2.5 Results and Discussion 35
2.5.1 Performance evaluation in the input space 35
2.5.2 Estimating the mean and variance of the 2-D problem 38
2.5.3 Comparison in a sensitivity analysis context 40
2.5.4 Comparison in an uncertainty analysis context 42
2.6 Conclusions 44
Author contributions 45

Chapter 3 Characterizing the Role of Internal Parameters on the Functioning of River Ice Model Using Global Sensitivity Analysis 46

Synopsis 46
3.1 Introduction 47
3.2 RIVICE Model Overview 50
3.3 Methodology 52
3.3.1 Regional sensitivity analysis (RSA) 52
3.3.2 Variogram analysis of response surfaces (VARS) 53
3.3.3 Bootstrapping strategy for reliability assessment 55
3.4 Study Site 56

3.5 Computational Experiments 58
3.6 Results and Discussion 60
3.6.1 Sensitivity analysis and parameter rankings 60
3.6.2 Assessment of confidence in GSA results by bootstrapping 64
3.7 Conclusions 65
Author contributions 66
Chapter 4 An Automated Factor Grouping Strategy for Robustness and Convergence Assessment of the Global Sensitivity Analysis Algorithms 68
Synopsis 68
4.1 Introduction 69
4.1.1 Motivation 69
4.1.2 Objectives and Scope 71
4.2 Review of the Literature 72
4.2.1 Difficulties associated with GSA of high-dimensional problems 72
4.2.2 Review of existing grouping strategies for GSA 76
4.3 The Proposed Factor Grouping Strategy 78
4.3.1 Generating the sensitivity matrix by bootstrapping 79
4.3.2 Transforming the distribution of sensitivity indices to be un-skewed 80
4.3.3 Factor grouping using agglomerative hierarchal clustering 82
4.3.4 A measure of robustness and convergence of GSA 83
4.3.5 Determining an optimal number of groups 85
4.3.5.1 An elbow method for finding optimal number of groups 85
4.3.5.2 Identifying optimal number of groups based on robustness assessment 86
4.3.6 The GSA method 87
4.4 Numerical Experiments 87
4.4.1 Illustration using the Sobol g-function 87
4.4.2 Modelling case study 89
4.4.2.1 Model description 89
4.4.2.2 Experimental setup 90
4.5 Results and Discussion 91
4.5.1 Factor grouping results 91

4.5.1.1 Results for the Sobol g-function ... 91
4.5.1.2 Effect of sampling variability on factor grouping results ... 93
4.5.1.3 Results for the MESH model ... 95
4.5.2 Assessment of the convergence and robustness of factor ranking results ... 97
4.5.2.1 Results for the Sobol g-function ... 97
4.5.2.2 Results for the MESH model ... 97
4.5.3 Comparison of the proposed strategies for finding optimal number of groups ... 100
4.6 Conclusions ... 103
Author contributions ... 104

Chapter 5 Efficient Strategies for Handling Simulation Model Crashes in Global Sensitivity Analysis ... 105

Synopsis ... 105
5.1 Introduction ... 106
5.1.1 Background and motivation ... 106
5.1.2 Existing Approaches to handling simulation crashes ... 108
5.1.3 Objective and scopes ... 110
5.2 Methodology ... 110
5.2.1 Problem statement ... 110
5.2.2 Proposed strategies for crash handling in GSA ... 111
5.2.2.1 Median substitution ... 111
5.2.2.2 Nearest neighbor substitution ... 112
5.2.2.3 Model emulation-based substitution ... 112
5.2.3 The utilized GSA framework ... 114
5.3 Case Studies ... 115
5.3.1 A conceptual rainfall-runoff model ... 115
5.3.2 A land surface-hydrology model ... 117
5.3.3 Experimental setup ... 118
5.4 Numerical Results ... 119
5.4.1 Results for the HBV-SASK model ... 119
5.4.2 Results for the MESH model ... 123
5.5 Discussion ... 129

5.5.1 Potential causes of failure in MESH 129
5.5.2 The role of sampling strategies in handling model crashes 132
5.6 Conclusions 133
Author contributions 135
Chapter 6 Conclusions and Future Directions 136
6.1 Summary of Dissertation Outcomes 136
6.2 Scope for Future Work 138
6.2.1 Thoughts and reflections on GSA of CESMs 138
6.2.2 Further research 141
6.3 Software Availability 141
Appendix 143
List of Publications 146
References 148

List of Figures

Figure 2-1 An illustration of the basic idea of Latin hypercube sampling: (a) A 4 by 4 example of Latin square - 4 different Latin characters are arranged in a way that no letter appears more than once in each row or column. (b) A 2-dimesional example of LHS with 4 sample points - There is only one point in each row and each column (the row and column taken by one of the sample points are darkened). 16

Figure 2-2 Illustrative examples of different configurations of Latin hypercube and/or space-filling designs. (a) An optimal sample with respect to both Latin hypercube and space-filling properties, (b) An optimal sample with respect to only space-filling properties, and (c) An LHS with very poor space-filling properties and highly correlated factors. 19

Figure 2-3 Example performances of existing sequential sampling strategies in a 100-dimensional space when projected onto a 2-dimensional sub-space. Performance of (a) Hammersley, (b) Halton, (c) Sobol', and improved (d) Halton and (e) Sobol' sequences (by leaping and scrambling) in a uniform distribution case. The plots in the bottom panel shows the same samples of plots above when transformed into a standard normal distribution space. All samples are projected onto dimensions 58 (horizontal axis) and 69 (vertical axis). 21

Figure 2-4 A doubling procedure of sample size for generating perfect-PLHS with n = 12 and p = 2: (a) An initial LHS with 3 sample points, (b) Dividing an initial sample domain into 6 intervals with equal marginal probability (c) The second slice with 6 sample points, (d) Dividing the second slice into 12 intervals with equal marginal probability, and (e) The third slice with 12 sample points. 25

Figure 2-5 The structure of the HYMOD rainfall-runoff model that consists of a soil moisture module (parameters: b_{exp} and C_{max}) and a routing module (parameters: *alpha*, R_s, and R_q). 30

Figure 2-6 The true cumulative distribution functions (CDFs) of the HYMOD model outputs: Nash-Sutcliffe metrics on streamflows (NS) and on the logarithm of streamflows (NS-log) 31

Figure 2-7 Comparison of different sampling algorithms in preserving one-dimensional projection properties as the sample size grows (the average of objective function defined by Eq. (4) over 100 replicates) – the objective function value of one indicates perfect performance – perfect-PLHS not shown as it remains on one for all slice numbers. 36

Figure 2-8 Comparison of different sampling strategies in uniformly spreading sample points based on the discrepancy metric (Eq. (7)) – The results are average of 100 replicates for the 100-dimenasional case – a lower discrepancy metric indicates a better dispersion of sample points. 37

Figure 2-9 Comparison of different sampling strategies in terms of maximum pairwise correlation between the factors (Eq. (8)) – The results are average of 100 replicates for the 100-dimenasional input space. .. 38

Figure 2-10 Case study 2: Comparison of different sampling strategies in estimating the mean (a & c) and variance (b & d) of the 2-dimensional problem. In (a) and (b), the deviations (errors) from the true mean and true variance were averaged over 100 replicates. In (c) and (d), the standard deviation of the estimated mean and variance (regardless of the true values) over the 100 replicates were calculated..... 39

Figure 2-11 Comparison of different sampling strategies in global sensitivity analysis of 2-D problem using the VARS method. The 5th and 95th percentiles of the 100 replicates are shown along with true values. Here, the $IVARS_{10}$ and VARS-TO (Total-Order effect) metrics were illustrated. 41

Figure 2-12 Comparison of different sampling strategies in global sensitivity analysis of NS criterion to HYMOD model parameters. The top panel is for the $IVARS_{50}$ metric and the bottom panel is for VARS-

TO (Total-Order effect) metric. The 5th and 95th percentiles of the 100 replicates are shown along with true values as the sample size grows. 41

Figure 2-13 Comparison of different sampling strategies in approximating the CDFs of the HYMOD model using K-S distance (top panel) and energy distance (bottom panel) metrics. Subplots (a) and (b) show the results for NS and NS-log, respectively. Subplot (c) shows the results for NS when of interest is to approximate the CDF of the model outputs with NS > 0.5 (only model outputs for behavioral parameter sets) to assess the sampling performance in a GLUE-type analysis – in each plot, the values were averaged over 100 replicates. 43

Figure 3-1 River ice processes simulated in RIVICE. 50

Figure 3-2 Catchment area of the Dauphin River consisting of the subbasins of the lakes: (a) Dauphin Lake; (b) Lake Winnipegosis; (c) Lake Manitoba; (d) Lake St. Martin; inset shows the Emergency Outlet Channel diverting water from Lake St. Martin into Buffalo Creek and onward into Lake Winnipeg [*adapted from Cold Regions Science and Technology, Vol. 82, Karl-Erich Lindenschmidt, Maurice Sydor, Richard W. Carson, "Modelling Ice Cover Formation of a Lake–River System with Exceptionally High Flows (Lake St. Martin and Dauphin River, Manitoba)," 36–48, 2012, with permission from Elsevier*]. 57

Figure 3-3 Evaluating the *RIVICE* parameters sensitivities using RSA; each subplot belongs to one model parameter, with its feasible range on the horizontal axis and the distribution of the *RIVICE* responses (cumulative RMSE distribution) on the vertical axis. 60

Figure 3-4 Evaluating *RIVICE* parameter sensitivities using VARS: (a) directional variograms; (b) integrated variograms (IVARS); the bottom plots (c and d) show a zoom-in of the top plots for very small values on the vertical axis (note that for variograms to remain meaningful, the distance between any two points within a given parameter range cannot exceed half of its range, i.e., $H_k \leq 50\%$). 61

Figure 3-5 Sensitivity indices for the *RIVICE* parameters using the RSA and VARS methods: (a) IVARS10; (b) IVARS30; (c) IVARS50; (d) K-S indices for RMSE measure; the numbers on the bars represent the parameter ranking obtained based on the sensitivity indices. 62

Figure 3-6 Ninety-five-percent confidence intervals (CIs) estimated based on the bootstrapping; subplots: (a) IVARS10; (b) IVARS30; (c) IVARS50 show the VARS-based metrics; subplot (d) shows the 95% CIs of K-S measure for RSA. 65

Figure 3-7 Reliability assessment of VARS and RSA for *RIVICE* model parameter ranking based on bootstrapping. 65

Figure 4-1 Cumulative distribution function for total number of input factors in GSA of environmental models. To make this plot we updated the data provided by Vanrolleghem et al. (2015) and Song et al. (2013). The Thomson Reuters Web of Science was used (August 2017) based on the search terms ''Global Sensitivity Analysis'' + "Environmental Model" + "Hydrology" 73

Figure 4-2 Flowchart of the factor grouping algorithm developed in this study 79

Figure 4-3 Conceptual distributions of (a) original sensitivity indices without normalization and (b) normalized sensitivity indices. Subplot (c) shows a zoom-in of the subplot (a) for small values on the vertical axis. When transforming the data in subplot (a), the differences between smaller sensitivity indices (moderately influential factors) should be expanded, whereas the differences between larger sensitivity indices (strongly influential factors) should be reduced. 81

Figure 4-4 Subplot (a) shows a typical plot of a distance metric (*y*-axis) for a cluster analysis versus the number of groups (*x*-axis). As can be seen, the distance metric decreases monotonically by increasing the number of clusters k, but from some $koptimal$ onwards it flattens significantly. Subplot (b) shows the corresponding clustering tree or dendrogram constructed by AHC. The height of each node in (b) represents the distance value between the right and left sub-branch clusters. The dashed line in (b) is the cutoff threshold for cutting the dendrogram into $koptimal$ groups. 86

Figure 4-5 Location of the Nottawasaga river basin in Canada 90

Figure 4-6 Factor grouping results for Sobol g-function based on (a) normalized sensitivity indices, and (b) sensitivity indices without normalization. The dashed lines show a zoom-in of the dendrogram for small values. Note that groups are labeled in order of importance (g_1 is the most important one). 92

Figure 4-7 Convergence plot for the sensitivity indices associated with $x1$ and $x2$ for the Sobol g-function estimated using an increasing number of model evaluations 93

Figure 4-8 Baker's index for comparing factor grouping results obtained from 40 runs of Sobol g-function. This measure varies between -1 to 1, with values close to 0 indicates that the two dendrograms are not statistically similar. The diagonal elements in this matrix is equal to 1 as they indicate the similarity of a dendrogram with itself. 95

Figure 4-9 Factor grouping results for MESH model. The dashed line shows a zoom-in of the dendrogram for small values. The *x*-axis labels correspond to model parameters. The groups are labeled in order of importance. 96

Figure 4-10 Comparison of the assessment of robustness based on (a) factor grouping and (b) individual factor ranking. In subplots (a) and (b) each line represents the evolution of robustness values associated with each factor of the Sobol g-function. 97

Figure 4-11 Comparison of the assessment of robustness based on (a) factor grouping and (b) individual factor ranking. Subplot (c) shows the median of the individual (red dashed line) and group-based (blue solid line) factor ranking results. In subplots (a) and (b) each line represents the evolution of robustness values associated with each parameter of the MESH model. 98

Figure 4-12 Trajectories of robustness values for each parameter of the MESH model in groups: (a) g_1, (b) g_2, (c) g_3, (d) g_4, (e) g_5, (f) g_6, and (g) g_7. Subplot (h) shows the evolution of median of the robustness values for each of the seven groups for the increasing number of function evaluations. 99

Figure 4-13 Top panel shows the evolution of the optimal number of groups with computational budget for (a) Sobol g-function and (b) MESH model. Bottom panel shows plots of the distance metric (*y*-axis) versus number of groups (*x*-axis) for (c) Sobol g-function and (d) MESH model when number of function evaluations is maximum. From $k = 4$ in subplot (c) and $k = 7$ in subplot (d) onwards the curves flatten notably 101

Figure 4-14 Maximum number of groups that is required to achieve minimum robustness values of (a) 0.90 for Sobol g-function and (b) 0.45 for MESH model parameters as the number of function evaluations grows. Bottom panel shows the histograms of the estimated robustness values for MESH model after (c) 40,000 and (d) 90,000 function evaluations. 102

Figure 5-1 Oldman river basin located in the Rocky Mountains in Alberta, Canada, flows into the Saskatchewan River Basin (adapted from Razavi et al., 2019). 117

Figure 5-2 Nottawasaga river basin in in Southern Ontario, Canada (adapted from Sheikholeslami et al., 2018). 118

Figure 5-3 Grouping of the 10 parameters of the HBV-SASK model when applied on the Oldman River Basin. The parameters are sorted from the most influential (to the left) to the least influential (to the right). 119

Figure 5-4 Comparison of the proposed crash handling strategies in sensitivity analysis of the HBV-SASK model using the STAR-VARS algorithm for different ratios of failures. The CDFs of the sensitivity indices for (a) strongly influential parameters {*FRAC*, *FC*, *C0*} (upper panel) and (b) moderately influential parameters {*C0*, *TT*, *alpha*, *K1*} (lower panel) are compared in this plot. The vertical line (solid black) on each subplot represents the corresponding ‘‘true’’ sensitivity index obtained when there were no failures. 121

Figure 5-5 Comparison of the proposed crash handling strategies in sensitivity analysis of the HBV-SASK model using the STAR-VARS algorithm for different ratios of failures. The CDFs of the sensitivity indices for weakly influential parameters {*LP*, *ETF*, *beta*, *K2*} are shown in this plot. The vertical line (solid black) on each subplot represents the corresponding ‘‘true’’ sensitivity index obtained when there were no failures. 121

Figure 5-6 Comparison of the crash handling strategies in estimating the parameter rankings for HBV-SASK model when (a)5%, (b)10%, (c)15%, and (d)20% of model runs were simulation crashes. The *y*-axis in each subplot shows the number of times out of 50 replicates that the rankings of the parameters are equal to the true ranking. 122

Figure 5-7 Scatter plots of the true NS values versus the imputed NS values when 20% of the HBV-SASK model simulations were deemed as model crashes. The accuracy of crash handling techniques is demonstrated in subplot (a) for the single NN method and in subplot (b) for the RBF method. These results belong to one replicate (arbitrarily chosen) out of 50 independent runs. 123

Figure 5-8 Grouping of the 111 parameters of the MESH model. The parameters are sorted from the most influential (to the left) to the least influential (to the right). This grouping is based on the results of the RBF method. 124

Figure 5-9 Sensitivity analysis results of the MESH model using different crash handling strategies for the most influential parameters. To better illustrate the results, the highly influential parameters in Group 1 are separately shown in two subplots. 125

Figure 5-10 Sensitivity analysis results of the MESH model using different crash handling strategies. To better illustrate the results, the moderately influential parameters in Group 2 are separately shown in three subplots. 126

Figure 5-11 Sensitivity analysis results of the MESH model using different crash handling strategies for weakly/non- influential parameters in Group 3. The bottom panel (c and d) shows a zoom-in of the top subplots for very small values on the vertical axis. 127

Figure 5-12 Plots comparing rankings of the MESH model parameters obtained by different crash handling strategies. Subplots (d), (e), and (f) (right column) show a zoom-in of the subplots (a), (b), and (c) (left column), respectively. The red line is the ideal (1:1) line. Note that a ranking of 1 represents the least sensitive and a ranking of 111 represents the most sensitive parameter 128

Figure 5-13 A 2-D projections of the MESH parameters for successful (blue dots) and crashed (red dots) simulations. Left column shows the threshold snow depth parameters *ZSNL* and right columns shows soil permeable depth (*SDEP*), maximum rooting depth (*ROOT*), and drainage index (*DRN*) for crop vegetation type. 131

List of Tables

Table 2-1 A review summary of studies for constructing optimal LHS 18
Table 2-2 The pseudo-code of the greedy search for generating quasi-PLHS 27
Table 2-3 Description of the HYMOD parameters 30
Table 2-4 Setup summary of computational experiments 34
Table 3-1 RIVICE parameters considered for the RSA and VARS sensitivity analysis, and their ranges of variation 58
Table 4-1 Recent studies that applied GSA to environmental models with 40 or more uncertain parameters 75
Table 4-2 The coefficients used in this study for 50-dimensional Sobol g-function 89
Table 5-1 HBV-SASK model parameters and their feasible ranges. 116
Table A-1 16 GRU types ranked by coverage area 144
Table A-2 MESH model parameters and their feasible ranges 144
Table A-3 Grouping of 111 parameters of the MESH 145

Chapter 1
Introduction

"Models are not right or wrong, but rather sound or unsound, as judged relative to how well they capture uncertainty and promote sensible inferences from the data."

Crane and Martin (2018)

1.1 Background and Motivation

1.1.1 Challenges associated with the ever-increasing complexity of environmental models

Environmental systems models are mainly built to simulate and predict the evolution of non-stationary, multivariate, and nonlinear behaviors that are often generated by cross-scale interactions and feedbacks among several environmental processes, including hydrological (e.g., soil moisture and evapotranspiration), geological (e.g., erosion and weathering), geochemical (e.g., neutralization of acidic solutions), atmospheric (e.g. cloud formation and radiative transfer), biological (e.g., microbial nutrition), and anthropogenic (e.g., land–use change and groundwater abstraction) processes. The complexity of these models has steadily grown in terms of *process complexity* and *process inclusivity*. While the former refers to the sophistication of modeled processes and answers the question of how and to what extent these processes are represented, which relates to our knowledge of the physics of the environmental processes, the latter refers to the number of processes included in the model and answers the question of which processes are represented. Thus, increasing model complexity means modifying model structure by adding new components into the model such as parameters, feedbacks, and boundary conditions which may operate on a range of spatiotemporal scales.

Building a *better* model is the desire that primarily derives the ever-increasing complexity of models, and thus it has been an important reason for decreased simplicity of models over time. It

is usually presumed that, in principle, the greater model complexity will lead to higher-fidelity models. Therefore, modelers have traditionally tended to add more details (based on Physics) to the models to rectify the discrepancies between model results and measured values – an indication of model's ability to describe some aspects of the underlying system. Although this approach is usually successful, it necessarily brings about an enormous increase in model complexity.

Apart from the added challenge of high computational cost associated with *Complex Environmental Systems Models* (CESMs), the inclusion of new components and process parameterizations (e.g., to better represent the spatial heterogeneity) increases the model's degrees of freedom and imposes a heavier burden on modelers for the estimation of model parameters. Importantly, high degrees of freedom, high degrees of parameter interactions, and lack of data, all together, can cause some parameters to be *non-identifiable*[1], and consequently can arise a situation that different combinations of parameter values may fit observed measurements equally well. As a result, unrealistic parameter values can generate a close match between the computed results and observed measurements. This ultimately leads to the "equifinality" issue that was originally introduced by Von Bertalanffy (1950a,b) and was formalized in the hydrologic community by Beven (1993).

Beck (1987) points out that as the model becomes more complex, it becomes more difficult to unambiguously falsify, test, or validate, due to many hypotheses involved in the CESMs. Beck (1987) states in his review that:

> *"Comprehensive models, which have become enormously complex assemblies of very many hypotheses, cannot be effectively falsified. This is partly a function of uncertainty in the field data, certainly a function of current limitations in the methods of system identification, and essentially a function, in the event of a significant mismatch between the model and observations, of being unable to distinguish which among the multitude of hypotheses have been falsified. In fact the detailed spatial patterns of water circulation and equally detailed*

[1] Model parameters are "identifiable" if there exists a unique parameter set that maximizes the likelihood function, i.e., parameters can be constrained by a given data set with a certain model structure (Sorooshian and Gupta, 1985).

differentiation of ecological behavior described by the more complex models would demand experimental observations that are simply not technically feasible."

Furthermore, Oreskes (2003) raises a "complexity dilemma": as the model complexity increases, the uncertainty in model predictions may also increase, and as such, one would put little confidence in the model predictions. This is contrary to the intuitive understanding that more complexity in models enhances the realism through adding more components to better simulate the underlying system. Oreskes (2003) asserts that:

"The closer a model comes to capturing the full range of processes and parameters in the system being modeled, the more difficult it is to ascertain whether or not the model faithfully represents that system. A complex model may be more realistic yet at the same time more uncertain."

Therefore, it has been suggested that model complexity and uncertainty exhibit a hypothetical U-shaped relationship such that as the model complexity increases, the uncertainty decreases due to adding a greater knowledge about the underlying system, but only up to a certain point, after which, as complexity continues to increase, so does the uncertainty due to a high level of interactions among various sources of uncertainty (see e.g., Hanna, 1993; Fisher et al., 2002; Snowling and Kramer, 2001; Perz et al., 2013). In other words, a new source of uncertainty will be introduced to the model (partially due to inevitable measurement errors or unavailability of observations) by adding a new component, which can be accumulated with other individual sources, and therefore amplifies the overall uncertainty. For example, Medici et al. (2012) investigated the circumstances in which the increasing complexity in three catchment-scale hydrology and nitrogen models may lead to acceptable model simulation or over-parameterization. Because of this U-shaped relationship, there may be an *optimal* complexity level that minimizes model uncertainty; however, as claimed by Oreskes (2003), it may <u>never</u> be possible to achieve that optimal state.

Finally, there is an additional challenge with more complex models, which is increased data requirements in the process of parameterization, calibration, and validation. CESMs often require a massive amount of distributed data collected from multiple data sources such as climatic data, soil properties, land use, topography, crop characteristics, groundwater levels, phosphorus and

nitrogen concentrations, etc. However, data availability is a significant problem, particularly for ungauged sites. Due to poor representation of the gauges over the study area, systematic/random errors in measurement equipment, and errors in data management, the impact of input data uncertainty on model outcome's accuracy can be significant in CESMs (McMillan et al., 2012; Zhang et al., 2016).

Overall, the foregoing discussion shows that the increased complexity in CESMs gives rise to three major concerns:

- CESMs are typically data-intensive and require more data-gathering effort, probably with new data collection techniques, and a more comprehensive dataset regarding different processes involved. This issue should indeed not be "covered up" by excluding some of the process from CESMs.
- Over-parameterization and the associated non-identifiability issue are common problems in CESMs. These models often can be tuned to calibration data, and accordingly can exhibit a high prediction fit. However, their validity and acceptability of the calibrated model can be doubtful when used for prediction.
- A careful attention is needed to be directed to proper treatment of the uncertainty in CESMs. This becomes more crucial when coming with severe uncertainties imposed by the anthropogenically induced socioeconomic and climatic changes.

1.1.2 Global sensitivity analysis to cope with outstanding challenges in CESMs

Overcoming the aforementioned challenges has greatly motivated researchers to develop advanced Global Sensitivity Analysis (GSA) methods. This is because GSA provides a way for systematically explaining how variations in different model input factors[1] and their interactions influence the model output variability, and thus a means of assessing which factor is more responsible for model output variations. Regarding the increased complexity of CESMs, GSA is

[1] To avoid distinctions between model parameters, boundary conditions, initial conditions, and forcings, all of these inputs to are referred to as "input factors" throughout this dissertation.

well recognized as being an essential aspect of the model analysis in addressing the outstanding challenges in CESMs (Razavi and Gupta, 2015), because:

- GSA determines the importance of input factors, and thus can be helpful for prioritizing data acquisition processes.
- GSA identifies non-influential factors; whose variations have little or no influence on the model output and can be fixed to reduce complexity of the CESMs.
- GSA attributes total uncertainty of the model outputs to multiple sources of uncertainty and their interactions.

Most well-known GSA algorithms are based on one of the following two schemes: (1) an analysis of derivatives, for example the method of Morris (Morris, 1991; Campolongo et al., 2007; Sobol and Kucherenko, 2009), and (2) an analysis of variance, for example the method of Sobol' (Sobol', 1993; Homma and Saltelli, 1996; Tarantola et al., 2006). However, it is not uncommon for methods based on these two schemes to give conflicting assessments, and Razavi and Gupta (2016a,b) showed that both the strategies are actually special cases of a more comprehensive, variogram-based approach that accounts for spatial structure of the model response surfaces. This variogram-based approach, known as *Variogram Analysis of Response Surfaces* (VARS), is a unifying framework that bridges across the derivative- and variance-based approaches by introducing the notion of "perturbation scale" (Haghnegahdar and Razavi, 2017).

Nevertheless, it is important to make an important remark concerning the uncertainty analysis. As categorized by Razavi (2017) and Razavi et al., (2019), uncertainty analysis techniques have been broadly developed under the two groups of forward uncertainty propagation (see, e.g., Helton et al., 2006; Rajabi et al., 2015; Williamson, 2015) and inverse uncertainty quantification (Bayesian inference) (see, e.g., Kuczera and Mroczkowski, 1998; Kavetski et al., 2006; Clancy et al., 2010). In the former, the uncertainty in input factors is propagated through the model to characterize the model outputs uncertainty (e.g., methods such as Monte-Carlo simulation and first-order second-moment techniques). However, the latter class of techniques attempts to improve estimates of model factors given *a priori* uncertainty and the discrepancy between model outputs and observations. On the other hand, GSA can be viewed as a third approach to

uncertainty analysis which can be employed to characterize the uncertainty of model outputs through attributing it to different uncertain model factors.

Considering the important features of GSA, sensitivity analysis needs to become a prerequisite in model building and system analysis in any field where CESMs are used. When CPU-demanding CESMs are involved, it is of vital importance to use clever techniques to efficiently conduct a comprehensive sensitivity analysis. Motivated by this challenge, the overarching objective of this dissertation is to develop efficiency-increasing strategies (as outlined in **Section 1.2**.) for robust sensitivity analysis of CESMs.

1.2 Objectives, Significance, and Research Questions

With the growth in complexity, the need for sensitivity analysis of CESMs has gained more recognition. However, two major inter-related challenges limit GSA's application to CESMs, namely (1) the curse of dimensionality, and (2) computational expense. The former refers to the fact that, as the number of uncertain factors increases, the volume of the factor space increases so rapidly that any attempt to explore the factor space in a statistically sound manner requires an exponentially-increasing number of model evaluations. The latter refers to the typically computationally intensive nature of CESMs, leading to long run-times that, together with the former, can make any meaningful sensitivity analysis of such models computationally prohibitive. Moreover, it is crucial to assess "robustness" of the GSA results for estimating the degree of insensitivity to sampling variation because the randomly drawn sample used to GSA is usually limited. This computational bottleneck in CESMs impedes effective implementation of the current-generation GSA methods. Three major factors can significantly affect the computational demand of the GSA techniques:

- Conducting GSA usually starts with generating a sample of input factors drawn from the feasible factor space, which will be then used to obtain CESMs responses at each sample point. Therefore, having an appropriate sampling strategy is crucial in reducing the computational cost. An appropriate sampling strategy should be capable of

generating a set of sample points with minimum sample size that are properly located in the factor space to ensure sufficient coverage of the output space.

- The sample size in GSA has been typically chosen based on available computational budget, without considerations/monitoring of the GSA's stability and convergence. Hence, the results are likely to have been highly sensitive to sampling variability. To avoid a possible lack of robustness in GSA result, it is common to use larger sample sizes, which will typically impose unnecessarily higher computational demand.
- Parameter-induced simulation crashes are a typical problem across most of the CESMs. When a computer code crashes, it is difficult to accomplish GSA, which can be very computationally costly for GSA algorithms because crashes can waste the rest of the model runs and prevent the completion of GSA. However, the existing GSA techniques, which require running CESMs for many configurations of input factors (some of which are prone to model crashes), are not equipped to effectively deal with model failures.

Given these gaps, this dissertation aimed to address the following key research questions:

1. How can we explore the input space of the high-dimensional CESMs efficiently and effectively to extract maximum amount of information from output space with minimum sample size?
2. How can we quantify the convergence rate and degree of robustness of GSA when applied to high-dimensional CESMs?
3. How can we handle simulation failures in GSA of computationally expensive CESMs?

There are three main objectives and contributions delivered by the research undertaken in this dissertation to address the above-mentioned questions, as highlighted in the next sub-sections.

1.2.1 Objective (1): proposing a novel sequential sampling strategy

Using an effective and efficient strategy for sampling high-dimensional factor spaces of CESMs is a cornerstone of sampling-based analyses such as optimization, surrogate modelling, uncertainty and sensitivity analysis. Performing these sampling-based analyses requires the computationally intensive codes to be executed on an appropriate distribution of sample points in factor space to extract maximum amount of information from the model response surface (i.e., output space).

One major drawback of traditional sampling strategies (e.g., Latin Hypercube sampling) is that they generate the entire sample set at once, a process that is known as "one-stage" or "one-shot" sampling. This requires users to specify the sample size prior to the associated sampling-based analysis. As a result, it is often the case that the user is not satisfied with the resulting sampling-based analysis (e.g., convergence criteria are not met), and needs to enlarge the sample size and resumes the sampling-based analysis with the updated/new samples.

The present dissertation addresses this gap (see **Chapter 2** and **3**) by introducing a novel strategy, called PLHS (Progressive Latin Hypercube Sampling), which sequentially generates sample points while progressively preserving the distributional properties of interest (Latin hypercube properties, space-filling, etc.), as the sample size grows.

1.2.2 Objective (2): developing a new robustness measure and factor grouping strategy

Characterizing and improving the "robustness" of the GSA results is an essential (but often neglected) part of any GSA method, particularly when applied to CESMs. Since GSA is a sampling-based technique, it is prone to statistical uncertainty, i.e., the results will be sensitive to randomness in the selection of the sample (due to sampling variability). Hence, robustness can be defined as the stability of the GSA results (i.e., the degree of insensitivity to sampling variation). In other words, lower variability of the results obtained over multiple trials of the algorithm (performed with different, identically distributed, sample sets) indicates a higher degree of robustness (see e.g., Montgomery (2008)).

The present research addresses this gap (see **Chapter 4**) by proposing a new measure of robustness for convergence assessment. This measure is based on a "factor grouping" strategy

that clusters sets of input factors into groups of similar properties based on their sensitivity. Given an ability to monitor the robustness of sensitivity analysis results, the user can improve efficiency by avoiding unnecessary model runs when the desired level of robustness is reached. In addition, the developed grouping capability is also useful when dealing with high-dimensional CESMs, where the user typically is not interested in the exact ranking for the many factors. Instead, it is beneficial to group factors into several distinct groups flagging "highly influential", influential", "moderately influential", "slightly influential", and "non-influential".

1.2.3 Objective (3): crash handling in sensitivity analysis of computationally expensive models

Sensitivity analysis of CESMs often requires running a model many times (hundreds or thousands of times). One of the main challenges during the GSA experiment is the failure/crash of the model simulations (computer code), particularly in CESMs with many interacting input factors. A simulation failure mainly happens due to the violation of the numerical stability conditions or mistakes made in the course of programming. These crashes can be very computationally costly for GSA because they can waste the model runs and prevent completion of sensitivity analysis. This problem has been commonly solved through reducing the feasible ranges for input factor(s) responsible for the failures in a hope to prevent them in the next experiment.

The present research addresses this gap (see **Chapter 5**) by exploring a series of automated strategies, suited to the majority of GSA methods, to deal with model crashes. These strategies allow users to cope with failed designs during GSA without knowing where they will take place and without re-running the entire experiment.

1.3 Outline of the Dissertation

This dissertation is a collection of published, accepted, or submitted papers from recognized peer reviewed Journals, as listed in the section of List of Publications within the dissertation. The titles of **Chapters 2** to **5** reflect the titles of these papers. These chapters begin with a synopsis of

the research motivation and findings followed by providing the paper that has been submitted, accepted, or published.

First, in **Chapter 1**, research questions are clearly defined, originality and significance of the research is highlighted, and it is explained that how this dissertation will add to, develop, or challenge existing literature in the field. To accomplish objective (1), **Chapter 2** introduces a new sequential sampling algorithm suited to any sampling-based analysis (e.g., optimization, uncertainty and sensitivity analysis), which has further been used in **Chapter 3** to evaluate the importance and identifiability of different parameters of a river ice model. **Chapter 4** answers explicitly the second research question to achieve objective (2) by presenting and testing a new measure of robustness to monitor and evaluate convergence of the GSA algorithms based on an automated factor grouping strategy. Objective (3) has been attained in **Chapter 5** through investigating and discussing the applicability of a series of alternative approaches in handling simulation failures during the GSA to circumvent the common need for re-running the entire experiment in such cases. Finally, **Chapter 6** brings together the findings of the research and recommends possible future extensions.

Chapter 2

Progressive Latin Hypercube Sampling for Robust sampling-based Analysis of Environmental Models

This chapter is a mirror of the following published article with minor changes to increase its consistency with the body of the dissertation. Changes were only made to avoid repeating the contents that have been presented more appropriately in other parts. References are unified at the end of the dissertation.

Sheikholeslami, R. and Razavi, S., 2017. Progressive Latin hypercube Sampling: An efficient approach for robust sampling-based analysis of environmental models. *Environmental Modelling & Software*, 93, 109–126. https://doi.org/10.1016/j.envsoft.2017.03.010

Synopsis

Efficient sampling strategies that comply with certain requirements concerning size of the problem, computational budget, and users' needs are essential for various sampling-based analyses, such as sensitivity and uncertainty analysis. In this chapter, we propose a new strategy, called Progressive Latin Hypercube Sampling (PLHS), which sequentially generates sample points while progressively preserving the distributional properties of interest (Latin hypercube properties, space-filling, etc.), as the sample size grows. Unlike Latin hypercube sampling, PLHS generates a series of smaller sub-sets (slices) such that (1) the first slice is Latin hypercube; (2) the progressive union of slices remains Latin hypercube and achieves maximum stratification in any one-dimensional projection; and as such (3) the entire sample set is Latin hypercube. The performance of PLHS is compared with benchmark sampling strategies across multiple case studies for Monte-Carlo simulation, sensitivity and uncertainty analysis. The results indicate that PLHS leads to improved efficiency, convergence, and robustness of sampling-based analyses.

2.1 Introduction

Simulation models have become an essential tool for environmental and water resources systems analysis and have been extensively employed to tackle various complex problems, including water supply systems design and operation, water quality and wastewater management, groundwater management, rainfall-runoff modelling, design optimization, risk assessment, and decision making (Castelletti et al., 2010). These models are typically characterized by: (1) highly non-linear/complex response surfaces, (2) large parameter/problem spaces (high-dimensional with large uncertainty at each dimension), and (3) high computational demand (long run times). These three characteristics challenge the use of sampling-based analyses such as uncertainty estimation (Liu and Gupta, 2007; Mugunthan and Shoemaker, 2006; Kuczera and Parent, 1998), sensitivity analysis (Sarrazin et al., 2016, Razavi and Gupta, 2015), surrogate modelling and optimization (Maier et al., 2014; Razavi et al., 2012a; Vrugt et al., 2006), and other Monte-Carlo type simulations (e.g., Rezaie et al., 2007; Linkov and Ramadan, 2004). The first characteristic above necessitates the collection of a sufficiently dense and properly distributed set of samples to adequately characterize the nonlinearity of the response surface, while the second requires a large sample size spreading across the entire high-dimensional space to ensure adequate exploration and coverage of the space. These, when coming with the third characteristic, impose significant computational burdens that may impede effective sampling-based analyses.

Sampling is a main building block of a range of algorithms designed for various types of environmental and water resources systems analysis. Depending on the type of analysis, sampling may be merely "input-oriented" (also called "model-free" here), where no adaptation is made based on resulting model outputs, or it may also be "output-oriented", where the sampling procedure is guided/adapted based on feedback received from the model outputs during sampling. Examples of the former, which is our focus in this chapter, include Monte Carlo simulation for uncertainty propagation, GLUE type methods for uncertainty analysis, sensitivity analysis, design of experiments, and some variations of surrogate modelling (e.g., response surface methodology and the other approaches categorized under "basic sequential framework" in Razavi et al. (2012b)). Examples of the latter include the surrogate modelling strategies where the response surface approximation evolves over time (i.e., the approaches categorized under

"Adaptive-Recursive Framework" and "Metamodel-Embedded Evolution Framework" in Razavi et al. (2012b)); generating perturbed hydrometeorological forcing data using the inverse approach (Guo et al., 2016); scenario generation and classification (Islam and Pruyt, 2016); and sensitivity analysis in multi-criteria decision analysis (Ganji et al., 2016).

The performance of a sampling strategy and the quality of its resulting samples directly controls the efficiency and robustness of any associated sampling-based analysis. A sample is deemed of quality if it possesses the intended distributional properties in parameter/problem space. The distributional properties of a sample are commonly assessed by one-dimensional projections for every dimension (i.e., marginal distributions), space-filling criteria, and correlation analysis. There is a wealth of literature over the past several decades on developing and improving various sampling strategies, including pseudo random sampling, stratified sampling, fractional and full factorial design (Box and Hunter, 1961), regular grid sampling, orthogonal design (Owen, 1992), Latin hypercube sampling (Mckay et al., 1979), and Sobol' sequences (Sobol', 1967). Latin Hypercube Sampling (LHS), pioneered by Mckay et al. (1979) and Iman and Conover (1980) and its variations such as orthogonal array-based LHS (Tang, 1993), orthogonal LHS (Ye, 1998), and symmetric LHS (Ye et al., 2000) are among the most commonly used sampling techniques for experiments with environmental and water resources systems models in a variety of application areas such as sensitivity and uncertainty analysis (e.g., Posselt et al., 2016; Gan et al., 2014; Zhan and Zhang, 2013), parameter calibration (e.g., Higdon et al., 2013), and surrogate modelling (e.g., Rajabi et al., 2015; Regis and Shoemaker, 2007). This may be mainly because of their (1) ease of use (comparable with random sampling), (2) insurance of one-dimensional projection properties ("Latin Hypercube" properties), and (3) ease in incorporating other criteria (e.g., orthogonality and symmetry) within sampling. Due to the second characteristic, LHS can be deemed a form of stratified sampling because it stratifies across the range of variables in accordance with the distributional properties of interest.

An effective sampling strategy needs to ensure the above properties, while being capable of preserving the distributional properties of the sampled points with any sample size. The "proper sample size", however, for a given simulation model and sampling-based analysis is not typically known a priori. The proper sample size here refers to a sufficiently large number of sample

points that will lead to convergence or robustness (i.e., degree of insensitivity to sampling variability) of the analysis results. On the other hand, the computational cost of a sampling-based analysis is linearly proportional to the sample size (i.e., number of model runs), assuming the computational demand of the sampling strategy is relatively negligible.

One major drawback of traditional LHS and many other sampling strategies is that they generate the entire sample points at once. This requires users to specify the sample size prior to the associated sampling-based analysis. Hence, users tend to utilize larger sample sizes, which possibly impose unnecessarily larger computational demand, to avoid "under-sampling" of the parameter/problem space. Also, it is often the case that the user is not satisfied with the resulting sampling-based analysis (e.g., convergence criteria are not met), and needs to enlarge the sample size and resumes the sampling-based analysis with the updated/new sample. In this case, the user will have a dilemma: either to generate a new sample by LHS with the size of interest and add it to the previously generated sample with the tradeoff that the union of the two samples will not be Latin hypercube, or to discard the previous sample at a computational cost and generate a new, larger sample to preserve the distributional properties of interest. Such needs warrant the development and application of "multi-stage" or "sequential" sampling, where sample size can grow progressively, while maintaining the desired distributional proprieties. This way, sequential sampling will allow the user to monitor the performance of the sampling-based analysis and assess the stopping criteria (e.g., convergence criteria) in an online manner.

In this chapter, we introduce a new and efficient sequential version of LHS, called Progressive Latin Hypercube Sampling (PLHS). As opposed to the traditional LHS approach that generates the entire sample set in one stage, the proposed PLHS will generate a series of smaller sub-sets while: (1) the first sub-set is Latin hypercube; (2) the progressive addition of sub-sets remains Latin hypercube and achieves maximum stratification in any one-dimensional projection; and thus (3) the entire sample set is Latin hypercube. In other words, PLHS will preserve the desired distributional properties while the sample size grows during the analysis. With several sampling-based numerical experiments for sensitivity and uncertainty analysis, we show that the proposed PLHS has multiple advantages over the one-stage sampling strategies, including improved convergence of the associated analysis and the robustness of the results to sampling variability.

This chapter is structured as follows. **Section 2.2** briefly explains the properties of original LHS, discusses different strategies for optimal sampling, and reviews existing sequential sampling methods. Next, in **Section 2.3**, we propose the PLHS approach, followed by a description of a heuristic algorithm for efficiently generating the optimal PLHS. **Section 2.4** describes case studies and experimental setup that are used to assess the PLHS. In **Section 2.5**, the experimental results and the analysis of the algorithm are provided and discussed. Finally, **Section 2.6** concludes the chapter.

2.2 Literature Review

2.2.1 What is Latin hypercube sampling?

Latin hypercube sampling (LHS) was inspired by the concept of "Latin square" from combinatorial mathematics, where an *n*-by-*n* matrix is filled with n different objects (i.e., numbers, characters, symbols, etc.) such that each object occurs exactly once in each row and exactly once in each column– see **Fig. 2-1a** for an example with 4 objects. The term "Latin" in Latin squares was inspired by the work of the famous mathematician, Leonhard Euler, who used Latin characters as the objects (Wallis and George, 2011). Like Latin squares, the basic idea of LHS for a 2-dimensional space and a sample size of n is partitioning each dimension into n disjoint intervals (levels) with equal marginal probability of 1/*n* and then randomly sampling once from each interval to ensure that there is only one point at each level. **Fig. 2-1b** shows a 2-dimensional example with 4 sample points uniformly distributed across each dimension by LHS. With no loss of generality, and for the sake of simplicity, the definitions and examples presented in this chapter are for the case of uniform distribution. For any other distribution (e.g., normal distribution), the uniformly distributed samples can be transformed by associated transformation functions.

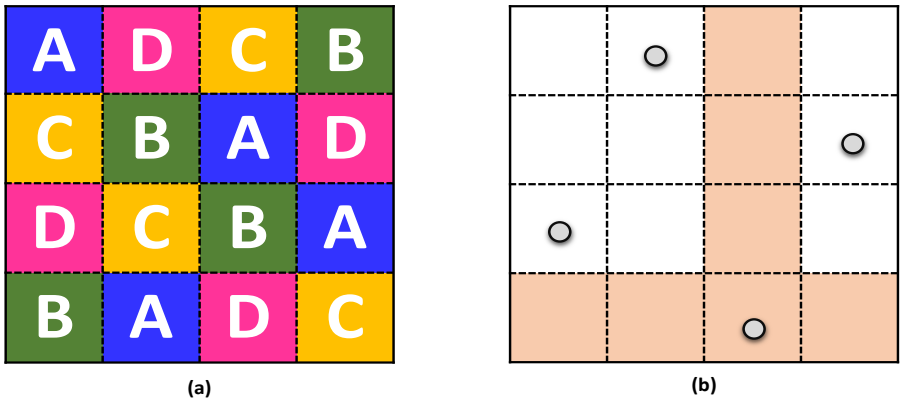


Figure 2-1 An illustration of the basic idea of Latin hypercube sampling: (a) A 4 by 4 example of Latin square - 4 different Latin characters are arranged in a way that no letter appears more than once in each row or column. (b) A 2-dimesional example of LHS with 4 sample points - There is only one point in each row and each column (the row and column taken by one of the sample points are darkened).

In the general case, consider a unit hypercube in a p-dimensional space, $C^p = [0,1]^p$, divided into n intervals (n is the sample size) with an equal length of $1/n$ along each axis – this creates n equally probable intervals indexed by $q = 1, \ldots, n$ corresponding to $[0 , 1/n)$, $[1/n , 2/n)$, $\cdots$, $[(n–1)/n , 1]$ for each dimension. LHS can be represented as n-by-p sample matrix $[x_{i,j}]$ ($i = 1, \ldots, n$; $j = 1, \ldots, p$), where $x_{i,j} \in [0,1]$ such that $x_{i,j}$ in the jth column belongs to only one interval. In other words, q is a random permutation of $\{1, 2, \ldots, n\}$ for each column, and each row of the matrix is a sample point. We denote this matrix by $\mathbf{LHS}(n, p)$. Original LHS ensures that the resulting sample possesses one-dimensional projection properties, indicating the projection of sample points in the p-dimensional space onto any dimension will follow the uniform distribution (or any other distribution of interest). Therefore, a sample is said to be "Latin hypercube" if and only if it possesses the one-dimensional projection properties. Such a sample, however, is only guaranteed to maximize the stratification in marginal distributions, while the multi-variate distributional properties (e.g., space-filling properties) in the p-dimensional space are not necessarily accounted for. There have been research efforts across a variety of fields to improve the performance of the original LHS, and several strategies were built on the original LHS, including Orthogonal Array-based LHS (Tang, 1993), Orthogonal LHS (Ye, 1998), and Symmetric LHS (Ye et al., 2000). For more detail, interested readers are referred to the reviews of the state of the art by Viana (2013) and Helton and Davis (2003).

2.2.2 Sampling improvements based on the notion of optimization

In addition to the methodological improvements such as orthogonal LHS and symmetric LHS that attempt to systematically generate Latin hypercube samples of better quality, there have been many studies that utilize the optimization theory to improve the performance of LHS (Pronzato and Müller, 2012; Xiong et al., 2009). Basically, the approach is to define secondary criteria (objective functions), in addition to being Latin hypercube, and formulate and solve an optimization problem to achieve (near) optimal Latin hypercube samples. This can be very effective, as in any variation of LHS, there may exist a huge number of configurations (sample points arrangements) that satisfy the associated LHS criteria but do poorly in terms of other criteria (e.g., space filling). The LHS algorithms typically randomly pick one of the many possible configurations, while optimization helps navigate through the myriad of choices and identify one that is (near) optimal in terms of secondary criteria.

The optimization problem for improving LHS belongs to the class of combinatorial optimization, with a total search space of $(n!)^{p}$ configurations for an exhaustive search (Viana et al., 2010). The computational efficiency of an optimization–assisted LHS algorithm depends on the size of the search space and the efficiency of the optimization algorithm used. In practice, therefore, the sampling procedure can become computationally demanding for larger values of n and p. A variety of optimization algorithms have been used in the literature to solve such combinatorial problems, including simulated annealing, genetic algorithms, and the branch-and bound-algorithm (see **Table 2-1** for a list of studies). However, in performing sampling-based analysis using LHS there are two notable remarks. First, once an optimal LHS is generated, it is independent of the considered application (model-free) and can be stored for future applications. Second, in most cases the computational cost of finding an optimal LHS is negligible in comparison with the time needed to run the computationally expensive computer simulations.

Table 2-1 A review summary of studies for constructing optimal LHS

Search algorithm	Criteria	Reference
Enhanced stochastic evolutionary algorithm	Max(*dist*)[1]	Husslage et al. (2011)
Translational propagation	Max(*dist*)	Viana et al., (2010)
Branch-and-bound	Max(*dist*)	van Dam et al. (2007)
Genetic algorithm	Max(*dist*)	Bates et al. (2004)
Simulated annealing	Max(*dist*)	Morris and Mitchell (1995)
Genetic algorithm	Min(L_2-*disc*)[2]	Rainville et al. (2012)
Simulated annealing	Min(L_2-*disc*), Max(*dist*)	Iooss et al. (2010)
Mixed integer linear programming	Min(*corr*) [3]	Hernandez (2008)
Florian's correlation reduction method	Min(*corr*)	Florian (1992)
Ranked Gram-Schmidt algorithm	Min(*corr*)	Owen (1994)
A heuristic algorithm	Min(*corr*), Min(L_2-*disc*)	Cioppa and Lucas (2007)
Exchange algorithm	Max(*ent*)[4]	Jourdan and Franco (2010)
Columnwise-pairwise	Max(*ent*), Max(*dist*)	Ye et al. (2000)
Enhanced stochastic evolutionary algorithm	Max(*ent*),Min(L_2-*disc*), Max(*dist*)	Jin et al. (2005)

[1]Max(*dist*) = Maximize the inter-point distance
[2]Min(L_2-*disc*) = Minimize L_2-discrepancy
[3]Max(*corr*) = Minimize correlation
[4]Max(*ent*) = Maximize entropy

Determination of the appropriate objective function(s) to be imposed on Latin hypercube properties (i.e., one-dimensional projection properties) via optimization has been a major topic of research. **Table 2-1** also reports different criteria that have been used in the literature as objective functions in different optimization-assisted LHS algorithms. These optimality criteria are mainly intended to improve space-filling properties to ensure that the sample points are uniformly scattered across the input space with minimal un-sampled regions. The two most commonly used objective functions are (1) maximizing the minimum inter-point distance among all possible pairs of sample points, and (2) minimizing the correlations (absolute value) between all pairs of columns of the sample matrix. To explain these, **Fig. 2-2** presents three example configurations when $n = 9$ and $p = 2$. The sample shown in **Fig. 2-2a** is optimal because it is Latin hypercube with strong space-filling properties, whereas the sample shown in **Fig. 2-2b** is not Latin hypercube, although it has strong space-filling properties. **Fig. 2-2c** shows a very poor Latin hypercube sample that, despite possessing one-dimensional projection properties, its sample points are poorly scattered in the factor space.

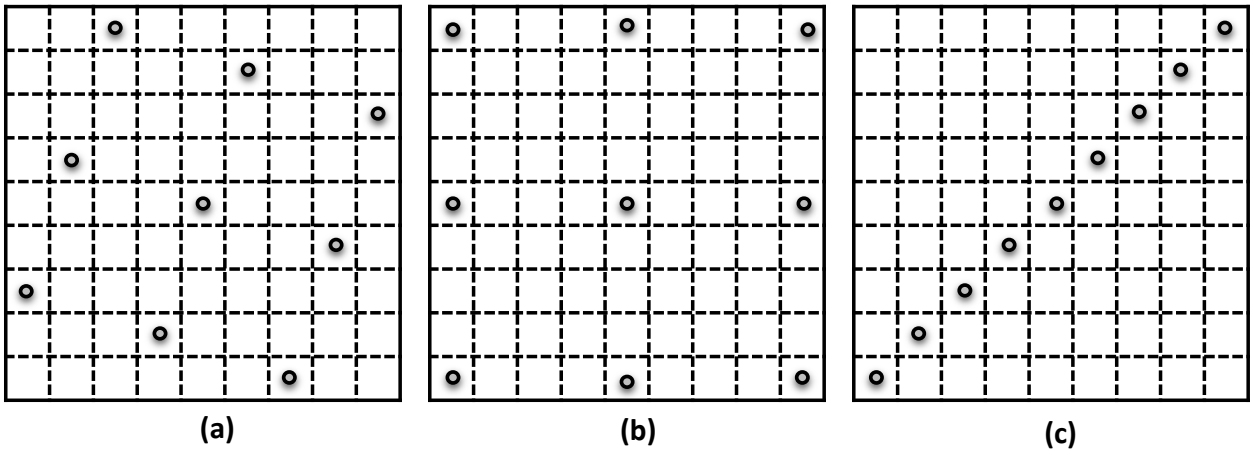


Figure 2-2 Illustrative examples of different configurations of Latin hypercube and/or space-filling designs. (a) An optimal sample with respect to both Latin hypercube and space-filling properties, (b) An optimal sample with respect to only space-filling properties, and (c) An LHS with very poor space-filling properties and highly correlated factors.

2.2.3 Input-oriented sequential sampling

The input-oriented (also called model-free) sequential strategies sample the input space iteratively without any feedback from any resulting output space (model response). At each iteration, they aim to sample the input space as uniformly as possible based on some pre-specified criteria. Many strategies for input-oriented sequential sampling (hereafter called "sequential sampling" for simplicity) have been developed, while they can be classified under two general families of stochastic and deterministic strategies. Stochastic strategies extend the one-shot strategies by iteratively searching for new points that satisfy or optimize pre-specified criteria, typically by means of optimization and randomization (e.g., Vořechovský, 2009; Xiong et al., 2009). Deterministic strategies, however, utilize deterministic routines designed for space filling (e.g., Schretter et al., 2012). These strategies are very computationally efficient, while having certain drawbacks, as explained below.

The so-called low-discrepancy sequences (also known as quasi-random sequences) are among the most well-known deterministic sequential sampling strategies. These include the Halton (Halton, 1960), Hammersley (Hammersley, 1960), and Sobol' (Sobol', 1967) sequences, most of which utilize prime numbers as bases to generate sample points for each dimension. These

sequences are only constrained by a low-discrepancy criterion to promote space-filling properties, with a caveat of possibly creating significant correlations between the factors, particularly in high-dimensional spaces (Loyola et al., 2016; Ong et al., 2012). The performance of low-discrepancy sequences has been extensively evaluated in the context of designs for computer experiments (e.g., Simpson et al., 2001; Kalagnanam and Diwekar, 1997).

The low-discrepancy sequences might have poor projection properties, particularly in high-dimensional spaces. Proper projection properties, also referred to as non-collapsing properties, are essential for effective sequential sampling (Pronzato and Müller, 2012; van Dam et al., 2007). A sample is said to possess projection properties when in any projection from its *p*-dimensional space to any lower dimensional sub-space, the sample points remain distinct from each other. This is vital in many sampling-based analyses. For illustration, **Fig. 2-3a**, **b**, and **c** depict a 2-dimensional projection of the first 1,000 points of 100-dimensional, 10,000-point samples sequentially generated by Hammersley (HM), Halton (HS), and Sobol' sequences. **Fig. 2-3f**, **g**, and **h** show the same samples when transformed onto a standard normal distribution. As can be seen, there is a strong correlation between the sample points of the two factors (58th and 69th), leaving large regions un-sampled. It should be noted that the correlation problems on samples projected onto lower dimensional sub-spaces are not reflected in discrepancy measurements (Loyola et al., 2016). Typically, for high dimensions (>~20), the sample size of low-discrepancy sequences should be quite large to ensure both space-filling and projection properties (Broad et al., 2015).

Various solutions have been proposed to resolve this issue, including using big prime numbers, leaping, and scrambling (see, e.g., Kocis et al., 1997). **Fig. 2-3d** and **i** show the improved performance of HS enabled with leaping and scrambling (HS-LS), where there is significant improvement in the dispersity of sample points but still there are some areas that remained un-sampled and some other areas with clusters of points. **Fig. 2-3e** and **j** show the superior performance of an enhanced Sobol' sequence by leaping and scrambling (Sobol'-LS), which has a high level of uniformity. Among these low-discrepancy samplings, we only used the Sobol'-LS as one of the benchmark strategies in this study.

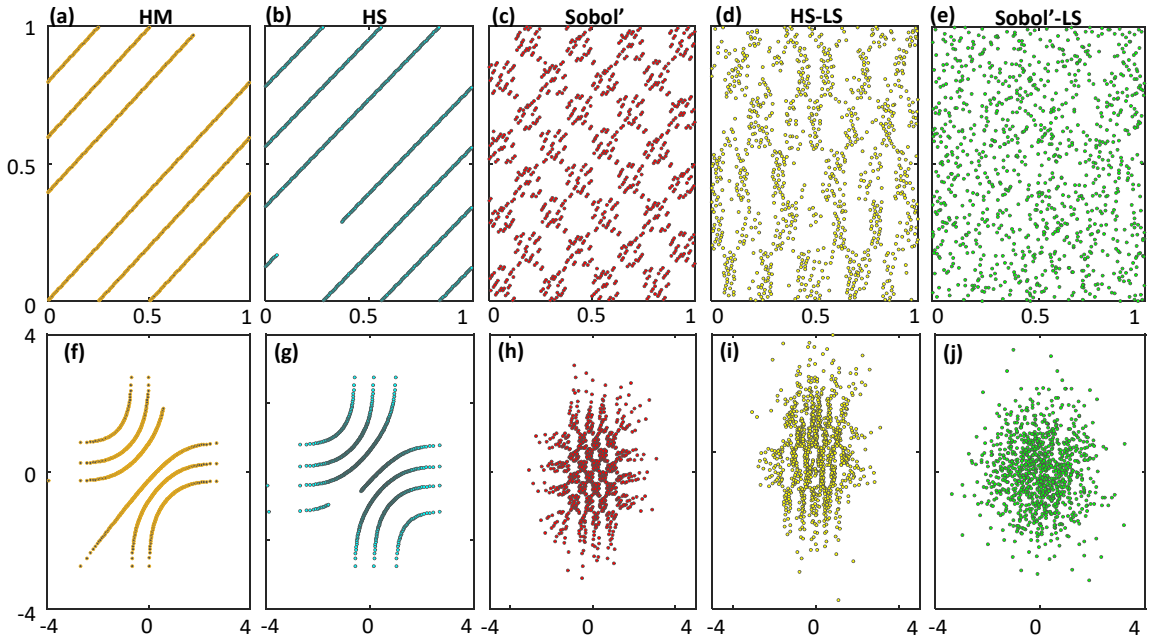


Figure 2-3 Example performances of existing sequential sampling strategies in a 100-dimensional space when projected onto a 2-dimensional sub-space. Performance of (a) Hammersley, (b) Halton, (c) Sobol', and improved (d) Halton and (e) Sobol' sequences (by leaping and scrambling) in a uniform distribution case. The plots in the bottom panel shows the same samples of plots above when transformed into a standard normal distribution space. All samples are projected onto dimensions 58 (horizontal axis) and 69 (vertical axis).

Crombecq et al. (2011) in a comprehensive study on the state-of-the-art sequential sampling strategies showed that the strategies with both space-filling and projective properties outperform the ones that only have the space-filling properties. Furthermore, Gong et al. (2016) showed that among various sampling schemes, the Good Lattice Points (GLP) and Symmetric Latin hypercube (SLH) are the most efficient methods and have the highest uniformity scores compared with the Halton and Sobol' quasi-random sampling methods. Note that none of the existing sequential sampling strategies can explicitly consider and maintain projection (e.g., Latin hypercube) properties, and this was our motivation to develop Progressive Latin Hypercube Sampling (PLHS), as introduced in the next section.

2.3 Progressive Latin Hypercube Sampling

2.3.1 Definition

In a *p*-dimensional space, let $\mathbf{S}_k(n_k, p)$, where $k = 1, 2, \cdots, T$, be a series of samples, each with a size of n_k, and let $\mathbf{L}^t$ be a new sample set formed by the union of these (sub-) samples such that $\mathbf{L}^t = \bigcup_{k=1}^{t} \mathbf{S}_k(n_k, p)$, where $t = 1, 2, \cdots, T$. We call sample $\mathbf{L}^t$ to be *progressive Latin hypercube* if and only if for any *t*, $\mathbf{L}^t$ is a Latin hypercube. In the case that $t = T$, we denote the new sample by $\mathbf{PLHS}(n, p, T)$ where n is the summation of the slice sizes, $n = \sum_{k=1}^{T} n_k$. As such, $\mathbf{PLHS}(n, p, T)$ is a Latin hypercube sample consisting of T sub-samples (also called *slices* hereafter) that their progressive union ($\mathbf{L}^t$ from $k = 1$ to $k = T$) holds Latin hypercube properties.

2.3.2 Mathematical formulation

Given that PLHS is an extension of LHS, in the following, we first reformulate LHS. Unlike the classic approach that considers LHS as a combinatorial problem, we formulate a real-valued problem. Let $\mathbf{S}(n, p)$ be a sample matrix, which consists of $n \times p$ elements (variables) $x_{i,j} \in [0,1]$ where $i = 1, \ldots, n$ and $j = 1, \ldots, p$. Also, consider the factor space $[0,1]^p$ divided into n disjoint intervals (strata or bins) $[0 - 1/n)$, $[1/n - 2/n)$, $\cdots$, $[(n-1)/n - 1]$ indexed by q ($q = 1, \ldots, n$) along each axis/dimension. We define a new set of auxiliary binary variables, $y_{q,j}$, such that:

$$y_{q,j} = \begin{cases} 1 & \text{if there exist any i for which } x_{i,j} \text{lies in the interval } q \\ 0 & \text{Otherwise.} \end{cases} \quad (1)$$

Then $\mathbf{S}(n, p)$ is said to be Latin hypercube when the following condition is satisfied:

$$\frac{\sum_{j=1}^{p} \sum_{q=1}^{n} y_{q,j}}{n.p} = 1 \quad (2)$$

The left-hand side of **Eq. (2)** is essentially a function of the sample matrix, $F\big(\boldsymbol{S}(n, p)\big)$, that varies between 1 (when the sample is Latin hypercube) and $1/n$ (when all sample points are in a single interval at every dimension).

To extend the formulation above to PLHS, we define the set of auxiliary binary variables as $y_{q,j}^t$, corresponding to sample matrix $\mathbf{L}^t = \bigcup_{k=1}^{t} \mathbf{S}_k(n_k, p)$, where $t = 1, 2, \cdots, T$ represents the

slice number. Note that the number of sample points in $\mathbf{L}^t$ is $n_t = \sum_{k=1}^{t} n_k$, and the factor space $[0,1]^p$ is divided into n_t disjoint intervals [0 - 1/n_t), [1/ n_t - 2/ n_t), · · ·, [(n_t –1)/ n_t - 1] indexed by q (q = 1, …, n_t) along each axis/dimension. Then the following equation is the necessary and sufficient condition for a sample to be said progressive Latin hypercube:

$$\sum_{t=1}^{T} \frac{\sum_{j=1}^{p} \sum_{q=1}^{n_t} y_{q,j}^{t}}{n_t . p} = T \tag{3}$$

The left-hand side of **Eq. (3)** is a summation of $F\left(\mathbf{L}^t(n_t, p)\right)$ from **Eq. (2)** for t = 1, . . ., T, $\sum_{t=1}^{T} F\left(\mathbf{L}^t(n_t, p)\right)$.

Generation of PLHS via the mathematical formulation derived is an optimization problem, as follows:

$$Maximize \sum_{t=1}^{T} \frac{\sum_{j=1}^{p} \sum_{q=1}^{n_t} y_{q,j}^{t}}{n_t . p} \tag{4}$$

with $x_{i,j}$ as decision variables and $y_{q,j}^{t}$ as auxiliary variables.

2.3.3 Practical implementation

The real-valued optimization formulation of **Eq. (4)** can be solved via various optimization solvers to generate a progressive Latin hypercube sample. As an alternative to directly solving this optimization problem, however, we also introduce two heuristic algorithms that efficiently construct PLHS for practical applications. The first algorithm, referred to as "doubling procedure" in this study, is an iterative approach which constructs a PLHS such that, at each step, the size of the sample is doubled. As such, the doubling procedure has limited flexibility in sample size. To gain flexibility in sample size, we develop the second algorithm for generating the PLHS, using an optimization approach. Any of these algorithms can be used in conjunction with other desired criteria such as such as maximum space filling and/or minimum correlation in addition to PLHS properties to improve the overall quality of sampling. **Section 2.3.3.2** explains a range of criteria that can be used during sampling or to test the quality of an already generated sample.

2.3.3.1 An iterative approach: doubling procedure (perfect-PLHS)

Perhaps, the only possible way to *iteratively* generate PLHS – adding new slices to an already generated LHS – is through the following algorithm. In this algorithm, the user chooses the size of the first slice $\mathbf{S}_1$ (i.e., n_1), and then the size of the second slice $\mathbf{S}_2$ will be $n_2 = n_1$ resulting in a sample $\mathbf{L}_2$ of size $2n_1$. Subsequently, the size of the third slice $\mathbf{S}_3$ will be $n_3 = 2n_1$, resulting in a sample $\mathbf{L}_3$ of size $4n_1$ and so on. This means the size of sample ($\mathbf{L}_j$) in this algorithm grows geometrically, as $n_1 \times 2^{(j-1)}$.

Here we use an example shown in **Fig. 2-4** to clarify this and explain how the algorithm works. With no loss of generality and for the sake of simplicity, this example is designed in a 2-dimensional space. **Fig. 2-4a** shows the first slice with $n_1 = 3$, which is Latin hypercube. In **Fig. 2-4b**, each of the three intervals for each dimension is divided into two equal intervals, and in **Fig. 2-4c**, three new sample points are added to create a Latin hypercube sample of size 6. Again, in **Fig. 2-4d**, each of the six intervals at each dimension is divided into two equal intervals resulting in 12 disjoint intervals for each dimension, and in **Fig. 2-4e**, 6 new sample points are added such that the resulting 12-point sample is Latin hypercube. This doubling procedure of sample size at each slice can be continued until obtaining the required sample size (3, 6, 12, 24, and so on). A similar approach to the above algorithm was proposed by Sallaberry et al. (2008), called the "two-multiple" algorithm, and further developed by Williamson (2015) for uncertainty quantification.

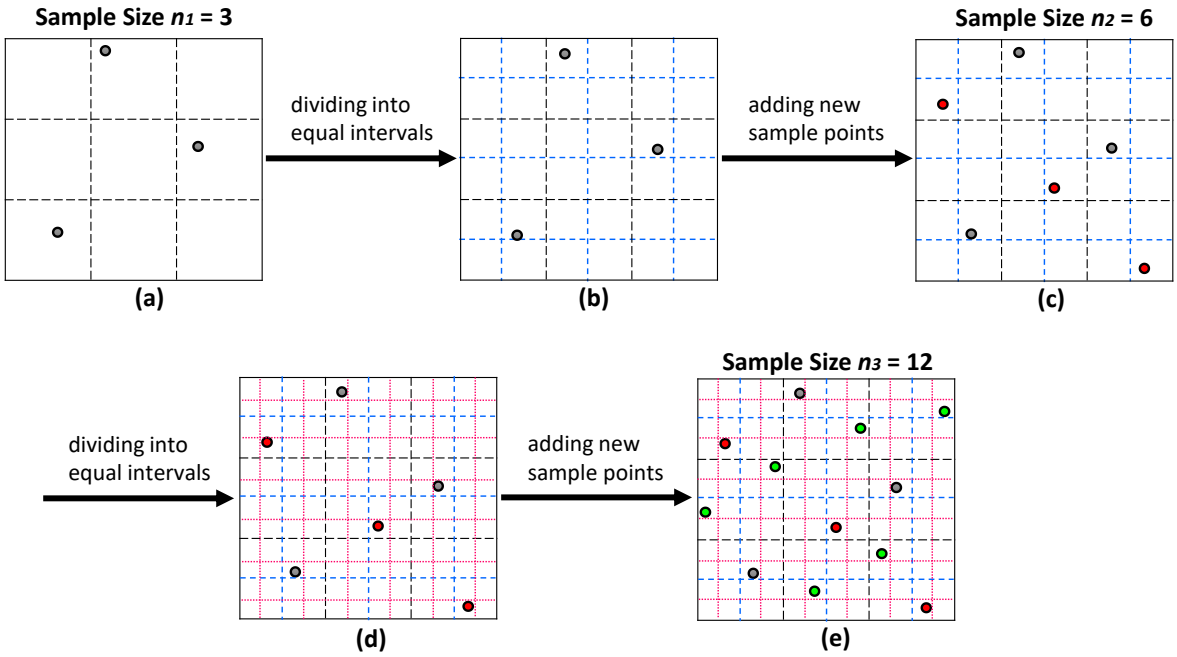


Figure 2-4 A doubling procedure of sample size for generating perfect-PLHS with n = 12 and p = 2: (a) An initial LHS with 3 sample points, (b) Dividing an initial sample domain into 6 intervals with equal marginal probability (c) The second slice with 6 sample points, (d) Dividing the second slice into 12 intervals with equal marginal probability, and (e) The third slice with 12 sample points.

2.3.3.2 An optimization approach (quasi-PLHS)

This approach utilizes an existing sampling strategy as a building block, called Sliced Latin hypercube Sampling (SLHS), originally developed by Qian (2012) and further enhanced by Ba et al. (2015) and Chen et al. (2016). The SLHS is a special type of LHS that generates a Latin hypercube sample (say with size n) formed by a collection of smaller, equally sized Latin hypercube samples (say with size m=n/T where T is the number of sub-samples). Mathematically stated, let $\mathbf{LHS}_k(m,p)$ for k = 1, 2,$\cdots$, T be a set of sample matrices and $\mathbf{S}^t$ be a new sample set formed by the aggregation of these sub-samples such that $\mathbf{S}^t = \cup_{k=1}^{t} \mathbf{LHS}_k(m,p)$, where 1< $t \leq$ T. A sample matrix is $\mathbf{SLHS}(n,p,T)$ if and only if $\mathbf{S}^T$, the union of all sub-samples, is Latin hypercube (i.e., $\mathbf{LHS}(n,p)$ where n = m×T). As such, although the entire sample is Latin

hypercube, the progressive addition of the sub-samples may *not* be Latin hypercube. This is the fundamental difference between SLHS and our proposed PLHS.

Utilizing SLHS, we developed an algorithm to approximately generate (quasi-) PLHS efficiently. The algorithm has two main steps. In the first step, SLHS is used to generate a set of T slices. In the second step, the order (arrangement) of theses slices is permutated to maximize one-dimensional projection properties when the slices are progressively combined. In other words, in this step, we search for an optimal permutation (ordering) of the slices to maximize objective function defined by **Eq. (4)**. The two steps can be repeated (with different initial random seed for SLHS) until a desirable sample is found.

Mathematically stated, we define a permutation π as a bijective function from $\{1, 2, \cdots, T\}$ to $\{1, 2, \cdots, T\}$. It is convenient to think of permutation π as sequence $\{\pi(1), \pi(2), \cdots, \pi(T)\}$. For example, if $\pi = \{2, 1, 3\}$ then $\pi(1) = 2$, $\pi(2) = 1$, and $\pi(3) = 3$; hence, $\pi(k)$ is the kth element of this sequence. Let $\mathbf{\Pi}$ be a set of random permutations of the integers $\{1, 2, \cdots, T\}$. Then, the problem is finding an optimal permutation of slices, $\pi^* \in \mathbf{\Pi}$, to maximize $F\left(\bigcup_{k=1}^{t} \mathbf{LHS}_{\pi^*(k)}(m, p)\right)$, for $t = 1, \ldots, T$ (Eq. (4)).

This optimization formulation is an ordering problem of Latin hypercube slices to maximize PLHS properties. Here, we employ a greedy search algorithm explained in the following to find a near optimal solution to this optimization problem. Note that in this algorithm, the initial slice $\pi^*(1)$ is chosen randomly. **Table 2-2** shows a straightforward heuristic **Algorithm 1** which is based on the so-called nearest neighbor greedy heuristic. Using **Algorithm 1**, we construct a permutation of slices $\pi^* = \{\pi^*(1), \pi^*(2), \cdots, \pi^*(T)\}$ with the initial slice $\pi^*(1)$ chosen randomly. In general, the next $\pi^*(k)$ is selected such that it maximizes F at each stage. The proposed procedure can be further improved if we repeat the algorithm by running it for different initial SLHS trials and then choosing the best PLHS among them. For each experiment in this study, we ran the algorithm 100 times with different random seeds and reported the best sample found based on the objective function. Note that in general, the choice of m and T is rather arbitrary. However, to generate a sample with size $T \times m$, the algorithm is more computationally

efficient for smaller numbers of slices, T. For example, producing T = 10 slices with m = 1,000 samples in each slice is much more efficient than producing T = 1,000 slices with m = 10.

Table 2-2 The pseudo-code of the greedy search for generating quasi-PLHS

Algorithm 1

- Select sample size n, number of dimensions/factors p, and total number of slices T.
- Generate a set of T slices using SLHS algorithm, $\mathbf{SLHS}(n,p,T) = \cup_{k=1}^{T} \mathbf{LHS}_k(m,p)$
- Randomly arrange slices in a sequence, $\{\pi(1), \pi(2), \cdots, \pi(T)\}$.
- Select an integer randomly, r, between 1 and T, and set k = 1, $\pi^*(1) = r$, and $\mathbf{S} = \mathbf{LHS}_{\pi^*(1)}(m,p)$.
- Mark the rth slice as visited and the other T–1 slices as unvisited.

while k is smaller than T
  Set $F_{\text{best}} = 0$ and $k = k + 1$.
  For j = all unvisited slices
    Construct a new set as $\mathbf{S}^{\pi(j)} = \mathbf{S} \cup \mathbf{LHS}_{\pi(j)}(m,p)$.
    Evaluate the objective function value using Eq. (4), $F(\mathbf{S}^{\pi(j)})$.
    If $F(\mathbf{S}^{\pi(j)}) > F_{\text{best}}$
     $F_{\text{best}} = F(\mathbf{S}^{\pi(j)})$
      $\pi^*(k) = \pi(j)$
    end if
  end for
  Mark the $\pi^*(k)$-th slice as visited and the other (T–k) slices as unvisited.
  $\mathbf{S} = \mathbf{S} \cup \mathbf{LHS}_{\pi^*(k)}(m,p)$.
end while

- Set $\mathbf{PLHS}(n,p,T) = \cup_{k=1}^{T} \mathbf{LHS}_{\pi^*(k)}(m,p)$

2.3.3.3 Other criteria

Like many other sampling strategies, a range of secondary criteria is available to be used in conjunction with the primary criteria such as the PLHS requirements (i.e. progressively attaining the projection properties). Any of these criteria can be optimized during sampling or used to evaluate an already generated sample. Below, we outline three possible criteria that work for maximum space-filling, lower discrepancy, and minimum pairwise correlation.

The maximin distance criterion is a measure of space-filling, which aims to maximize the minimum distance among every pair of points in a sample, *dist*:

$$Maximize: dist = \min_{\substack{i,j=1,\ldots,n \\ i \neq j}} \left\{ d\left(\mathbf{X}^i, \mathbf{X}^j\right) \right\} \quad (5)$$

where $d(.,.)$ is a distance measure (here the Euclidean measure) and n is the sample size.

The maximin criterion is mainly intended to improve space-filling properties to ensure that generated sample points are evenly spread across the entire factor space – i.e., sample points are located almost equally far apart. This optimality criterion has reportedly worked well to ensure adequate space-filling (Santner et al., 2003).

The so-called *L2*-star discrepancy is a metric to evaluate the uniformity and discrepancy of sample points in an input space. Given a set of n points $\mathbf{S} = \{\boldsymbol{X}_1, \cdots, \boldsymbol{X}_n\}^T$ with $\boldsymbol{X}_i \in \mathbb{R}^p$ in the p-dimensional unit cube $C^p = [0, 1)^p$, this metric can be approximately calculated by (Warnock, 1972):

$$D^{L2}(\mathbf{S}) = \left(\frac{1}{3}\right)^p + \frac{1}{n^2}\sum_{k=1}^{n}\sum_{i=1}^{n}\prod_{j=1}^{p}\left[1 - \max\left(x_{k,j}, x_{i,j}\right)\right] - \frac{2^{1-p}}{n}\sum_{i=1}^{n}\prod_{j=1}^{p}\left(1 - x_{i,j}^2\right) \quad (6)$$

where $x_{i,j}$ is the jth coordinate of the ith point ($\boldsymbol{X}_i$) in the sample set $\mathbf{S}$. This metric varies between 0 and 1, and lower discrepancy values indicate better spread of points in the input space. In addition to the *L2*-star discrepancy, other types of discrepancy metrics are available in the literature such as centered discrepancy, symmetric discrepancy, and wrap-around discrepancy (Hickernell, 1998; Gong et al., 2016). Here, we used the *L2*-star discrepancy (Eq. (6)), which is a popular metric representing the overall uniformity in the high-dimensional space (Niederreiter, 1992).

Finally, the maximum pairwise correlation is a standard measure of linear dependence between two variables. This measure is based on the Pearson product-moment correlation coefficient. For any two columns ($i \neq j$) of a sample matrix, it can be calculated by:

$$\rho_{ij} = \frac{\sum_{k=1}^{n}\left(x_{k,i} - \bar{x}_{:,i}\right)\left(x_{k,j} - \bar{x}_{:,j}\right)}{\sqrt{\sum_{k=1}^{n}\left(x_{k,i} - \bar{x}_{:,i}\right)^2 \sum_{k=1}^{n}\left(x_{k,j} - \bar{x}_{:,j}\right)^2}} \quad (7)$$

where $\bar{x}_{:,i} = \sum_{k=1}^{n} x_{k,i}/n$ and $\bar{x}_{:,j} = \sum_{k=1}^{n} x_{k,j}/n$.

The maximum absolute value of ρ_{ij} for all pair-wise combinations of factors is commonly used as a measure of sample quality and denoted by ρ_{max} (Cioppa and Lucas, 2007). Therefore, a sample with smaller ρ_{max} is deemed to be of higher quality.

In our experiments in this study, we utilized the maximin distance criterion directly during sampling. We also used the *L2*-star discrepancy and maximum pairwise correlation criteria to independently evaluate the generated samples via different strategies. In PLHS, the maximin distance criterion for every slice as well as for the entire sample were calculated and aggregated via a weighting approach.

2.4 Computational Experiments

We used two test problems and designed four experiments to evaluate the performance of PLHS against other sampling methods in the input space in terms of space-filling, correlation, and projective properties (first experiment), and in the output space in terms of the statistical measures and sensitivity metrics (last three experiments). The two test problems and the experimental setup are described as follows.

2.4.1 Test problems

2.4.1.1 A 2-D analytical test function

We compared the performance of the different sampling algorithms on a two-dimensional toy function, *Y*, defined by

$$Y(x_1, x_2) = 2(x_1)^2 + 3(x_2)^2 + x_1.x_2 \quad (8)$$

where x_1 and x_2 are random variables uniformly distributed in interval [–1, 1]. This function, adapted from the analysis of Razavi and Gupta (2015), is a quadratic function with an interaction term (the third term).

2.4.1.2 A real-world problem

We employed the HYMOD conceptual hydrologic model (**Fig. 2-5)** to assess PLHS in real-world problems. HYMOD has five parameters that need to be specified/calibrated by the user (**Table 2-3**). Details of HYMOD can be found in Boyle (2000) and Wagener et al. (2001). The HYMOD model used in this study was adopted from Vrugt et al. (2003), developed for the Leaf River watershed located north of Collins, Mississippi, USA. We used the Nash-Sutcliffe metric on streamflows (NS) as well as on the logarithm of streamflows (NS-log) as the model outputs.

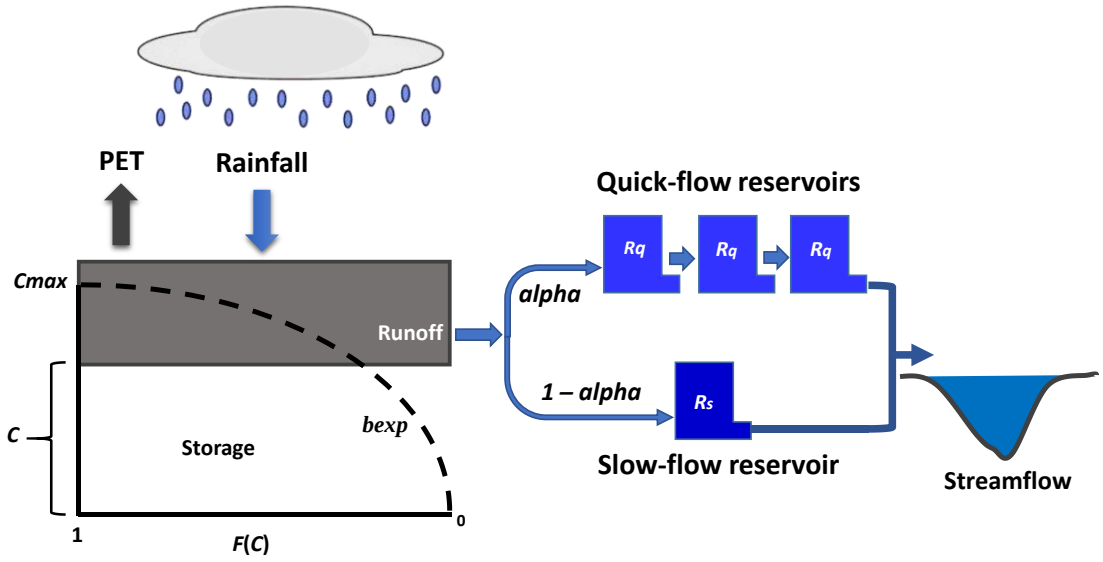


Figure 2-5 The structure of the HYMOD rainfall-runoff model that consists of a soil moisture module (parameters: b_{exp} and C_{max}) and a routing module (parameters: *alpha*, R_s, and R_q).

Table 2-3 Description of the HYMOD parameters

Parameter	Range	Unit	Description
C_{max}	1.00 –500.00	[mm]	Maximum storage capacity
b_{exp}	0.10 –2.00	[–]	Degree of spatial variability of the soil moisture capacity
alpha	0.10 –0.99	[–]	Factor distributing the flow between two series of reservoirs
R_q	0.10 –0.99	[day]	Residence time of the quick release reservoirs
R_s	0.00 –0.10	[day]	Residence time of the slow release reservoir

We ran the model with 500,000 randomly generated parameter sets (uniformly distributed in the parameter ranges of **Table 2-3**) by original LHS to generate the "true" cumulative

distribution functions (CDFs) of the model responses. **Fig. 2-6** shows the CDFs for NS and NS-log metrics.

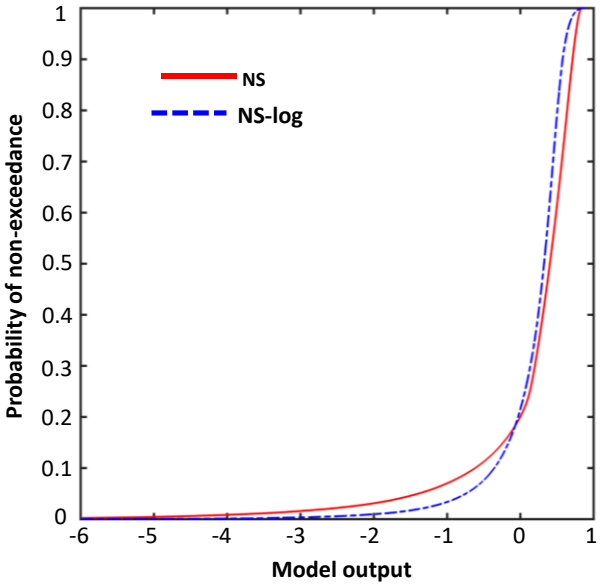


Figure 2-6 The true cumulative distribution functions (CDFs) of the HYMOD model outputs: Nash-Sutcliffe metrics on streamflows (NS) and on the logarithm of streamflows (NS-log)

2.4.2 Design of experiments

2.4.2.1 Experiment I

We designed the first experiment to evaluate PLHS in terms of achieving the maximum stratification when projected onto univariate margins using the proposed objective function (**Eq. (4)**) in 2-, 5-, and 100-dimensional input spaces. Moreover, in this experiment the quality of different sampling strategies were compared using the low-discrepancy (**Eq. (6)**) and maximum pairwise correlation (**Eq. (7)**) as the performance measures.

2.4.2.2 Experiment II

One of the frequent uses of sampling is in Monte Carlo simulation. This experiment investigated the effectiveness of PLHS in Monte-Carlo-based estimation of the first- and second-order moments (mean and variance) of variable Z as a function of two uniformly distributed variables, as defined by **Eq. (8)**. Here, the performance of sampling methods was evaluated based on the

deviation (error) of the estimates of the mean, *E(Z)*, and variance, *Var(Z)*, from their true values, which can be analytically calculated as *E(Z)* = 1.6667 and *Var(Z)* = 1.2667.

2.4.2.3 Experiment III

In this experiment we assessed how PLHS works to improve the convergence and robustness of global sensitivity analysis (GSA). We used a recently developed GSA framework, known as Variogram Analysis of Response Surfaces (VARS), proposed by Razavi and Gupta (2016*a*). VARS is a general framework that utilizes directional variogram and covariogram functions to characterize the full spectrum of sensitivity-related information, thereby providing a comprehensive set of global sensitivity metrics with minimal computational cost. VARS generates a new set of sensitivity metrics called IVARS (Integrated Variogram Across a Range of Scales) that summarize the variance of change (rate of variability) in model response at a range of perturbation scales in the parameter space. VARS also generates the Sobol' (variance-based) total-order effect and the Morris (derivative-based) elementary effects. Here, we utilized the STAR-VARS implementation of VARS developed in Razavi and Gupta (2016*b*). It has been shown that STAR-VARS is highly efficient and statistically robust, providing stable results within 1-2 orders of magnitude smaller numbers of sampled points (model runs), compared with the original Sobol' and Morris approaches (Razavi and Gupta (2016 *a,b*)).

STAR-VARS utilizes a star-based sampling, which consists of two elements: (1) Latin hypercube sampling to identify star centers, and (2) a structured-sampling approach to identify star points. In this experiment, we replaced the first element by PLHS (and the benchmark sampling algorithms) to generate star centers. The sample size reported in this experiment refers to the number of star centers taken by the first element above. The total number of function evaluations (the total sample including sample points from the second element) were 2,300 and 950 for the HYMOD and 2-dimensional case studies, respectively. We set Δh (VARS resolution parameter) to 0.1 (as recommended in Razavi and Gupta (2016b)). We compared the performance of VARS enabled with the different sampling algorithms in generating $IVARS_{10}$ (integrated variogram in range 0 to 10% of the parameter range), $IVARS_{50}$ (integrated variogram in range 0 to 50% of the parameter range), and VARS-TO (VARS-derived Total-Order effect).

These global sensitivity metrics are calculated simultaneously in a VARS run. The global sensitivity of NS to the five HYMOD model parameters and two-dimensional test function were assessed. The "true" values of these global sensitivity metrics for this case study were adopted from Razavi and Gupta (2015; 2016b).

2.4.2.4 Experiment IV

This experiment was designed to evaluate PLHS in an uncertainty analysis context, where of interest is to understand how uncertainty in model inputs propagates to the model output via Monte-Carlo simulations. Here, we tested how PLHS works to approximate the distribution of the output of the HYMOD model. This experiment has fundamental elements in common with GLUE-type analyses and was also intended to reflect how PLHS could improve the efficiency and robustness of GLUE. The goal here was to assess how *accurate* and *robust* PLHS could approximate the true distribution of the model output as the sample size grows. The accuracy was assessed by comparing the approximate CDF with the true CDF using two similarity metrics that measure the difference between the distributions of random variables.

The first metric is the Kolmogorov-Smirnov (K-S) distance measure (Kolmogorov, 1933). As a measure of similarity, the K-S calculates the maximum distance between the two CDFs as follows

$$K-S=\max_{y}\left|CDF_{true}(y)-CDF_{apprx}(y)\right| \tag{9}$$

where y is the model output (NS or NS-log), and CDF_{true} and CDF_{apprx} are the true and approximated distributions of the model output.

The second metric is called energy distance which characterizes the equality of two distributions (Székely and Rizzo, 2005). Suppose that X and Y are independent sets of real-valued random variables with cumulative distribution functions CDF_{true} and CDF_{apprx}, respectively; the energy distance is the squared root of

$$D^2\left(CDF_{true}\,,CDF_{apprx}\right)=2E\|X-Y\|-E\left\|X-\acute{X}\right\|-E\left\|Y-\acute{Y}\right\| \tag{10}$$

where ||.|| denotes the Euclidean norm, and $\acute{X}$ denotes an independent and identically distributed (iid) copy of X; and similarly, Y and $\acute{Y}$ are iid. Mathematically, it has been proven that the Energy distance is twice the Cramér's distance (Székely and Rizzo, 2013), i.e.

$$D^2\left(CDF_{true}\,, CDF_{apprx}\right) = 2\int_{-\infty}^{\infty}\left(CDF_{true}(y) - CDF_{apprx}(y)\right)^2 dx \qquad (11)$$

In this experiment, we used the normalized version of energy distance, D^2_N, which can be estimated by dividing D^2 by an estimate of $E\|X - Y\|$. For more details on energy distance, see Székely and Rizzo (2013, 2005) and Aslan and Zech (2005).

The K-S and D^2_N values vary between zero and one, with zero indicating the two distributions are identical. In this experiment, we compared different sampling strategies based on how rapidly their values of K-S and D^2_N converge to zero as the sample size grows.

2.4.3 Setup for sampling strategies

The two developed PLHS algorithms, using the doubling procedure (called perfect-PLHS) and optimization approach (called quasi-PLHS), were tested in the four different experiments explained above. For comparison purposes, the same experiments were also carried out with random sampling (RAND), original LHS, Sobol' sequence, and SLHS. **Table 2-4** summarizes the experimental setup for each case study.

Table 2-4 Setup summary of computational experiments

Case study	Sample size (n)	Number of slices (T)	Slice size (m)	Dimension (p)
Experiment I	1,000	100	10	2-D
	1,000	10	100	5-D
	1,000	10	100	100-D
Experiment II	1,000	100	10	2-D
Experiment III	50	10	5	2-D
	50	10	5	5-D
Experiment IV	500	10	50	5-D

For RAND and LHS techniques, to construct samples sequentially, we simply generated sample points at each new slice and added them to previous ones, regardless of the properties of

previously generated slices. It should be noted that the samples generated for Experiment I were also used for numerical simulations in Experiment II. For Sobol' sequence (enabled with leaping and scrambling), RAND, and original LHS techniques, we used the built-in functions of MATLAB to generate the samples and for the SLHS technique, we employed the method introduced by Ba et al. (2015). Moreover, for each replicate of LHS, SLHS, and PLHS algorithms, we ran 100 trials with different random seeds and reported the best sample according to the maximin criterion. Also, to ensure a fair comparison, we accounted for sampling variability due to randomness in the comparisons by carrying out 100 replicates of each experiment. This allowed us to see a range of possible performances for each algorithm and to assess their robustness against their random components.

2.5 Results and Discussion

2.5.1 Performance evaluation in the input space

Fig. 2-7 compares the average objective functions (**Eq. (4)**), F, as the sample size grows for different sampling methods obtained from the 100 replicates (to normalize this value, it is divided by the total number of slices *T*). The results are based on 2, 5, and 100-dimensional input spaces. The value of *F* associated with perfect-PLHS remains at one for all the slice numbers (not shown), and that of quasi-PLHS (using **Algorithm 1**) is higher than those produced by RAND, LHS, Sobol' and SLHS. This indicates that PLHS better preserves the projective properties and achieves the best stratification. Importantly, the *F* values of RAND and LHS tend to degrade by increasing the sample size, suggesting they might not be appropriate for sequential sampling. As shown in **Fig. 2-7**, in lower dimension the Sobol' sequence is better than RAND and LHS; however, by increasing the dimensionality of the space to 100, it performs almost like the RAND technique in terms projection properties.

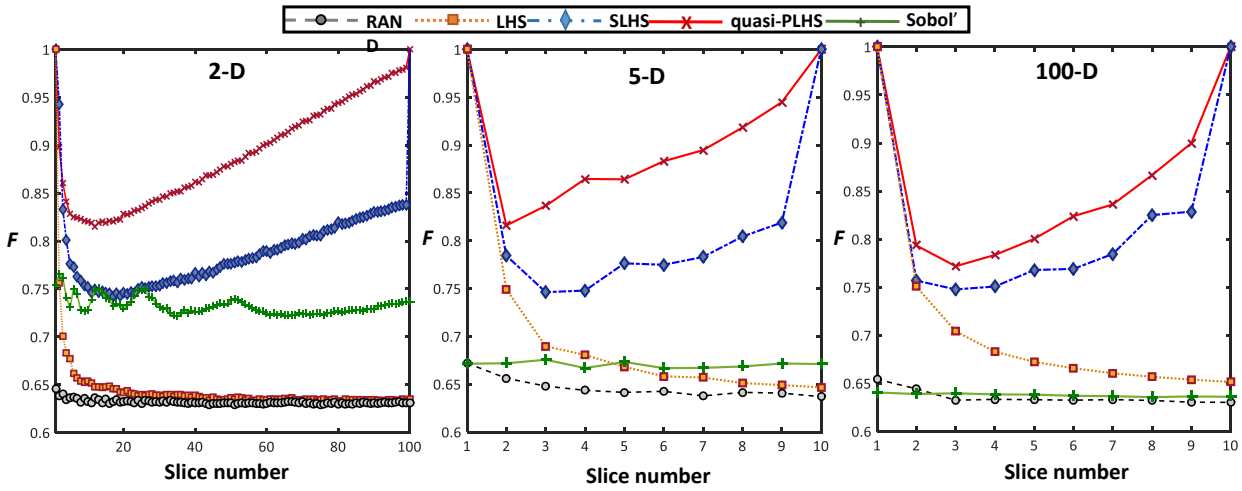


Figure 2-7 Comparison of different sampling algorithms in preserving one-dimensional projection properties as the sample size grows (the average of objective function defined by Eq. (4) over 100 replicates) – the objective function value of one indicates perfect performance – perfect-PLHS not shown as it remains on one for all slice numbers.

Fig. 2-8 compares the discrepancy metric D^{L2} of different sampling strategies defined by **Eq. (7)** for the 100-dimensional input space. Here, the perfect-PLHS was constructed by doubling the initial sample size of 100, resulting in 100, 200, 400, and 800 sample points, equivalent to slice numbers 1, 2, 4, and 8 of the other sampling strategies. As can be seen, the Sobol' sequence has the lowest D^{L2} discrepancy values, which is comparable with the perfect-PLHS. The best performance of Sobol' is because this method is inherently constrained by a low-discrepancy criterion. Then, the quasi-PLHS is the superior algorithm, as it consistently has the lower discrepancy values for any sample size. SLHS comes fourth, followed by LHS and RAND.

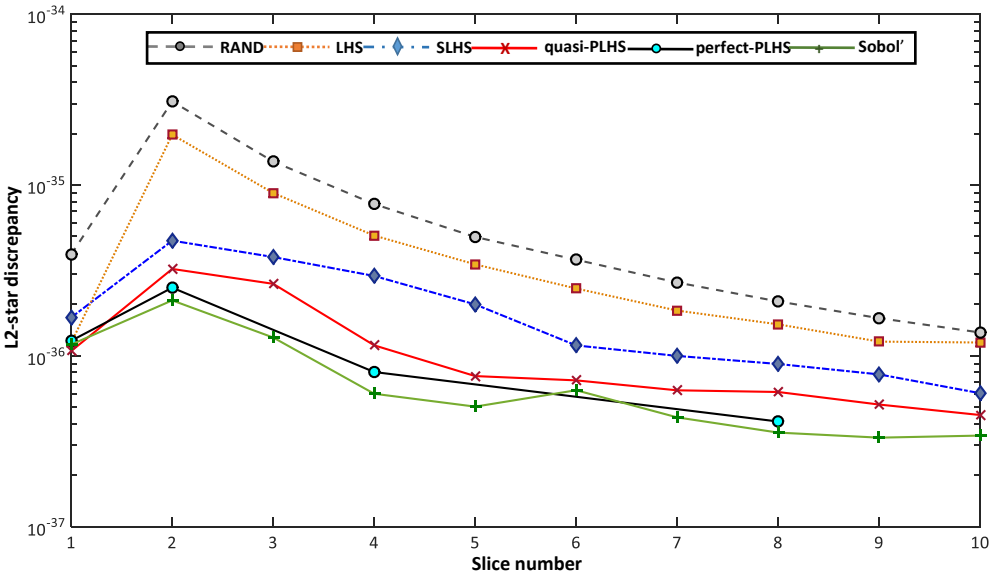


Figure 2-8 Comparison of different sampling strategies in uniformly spreading sample points based on the discrepancy metric (Eq. (7)) – The results are average of 100 replicates for the 100-dimenasional case – a lower discrepancy metric indicates a better dispersion of sample points.

The maximum pairwise correlations (ρ_{max}) for different sampling strategies in the 100-dimenasional input space are illustrated in **Fig. 2-9**. Like **Fig. 2-8**, the perfect-PLHS was constructed with total sample size of 800 (equivalent to slice numbers 1, 2, 4, and 8). Overall, at each slice number, the performance of all the algorithms are close, with the perfect- and quasi-PLHS strategies having the minimum (best) ρ_{max} of all. The results of RAND, LHS, and SLHS are comparable in terms of ρ_{max} in most cases, except for the last slice, where SLHS demonstrates considerable improvement. Furthermore, the Sobol' method has the higher ρ_{max} values compared to the SLHS and LHS.

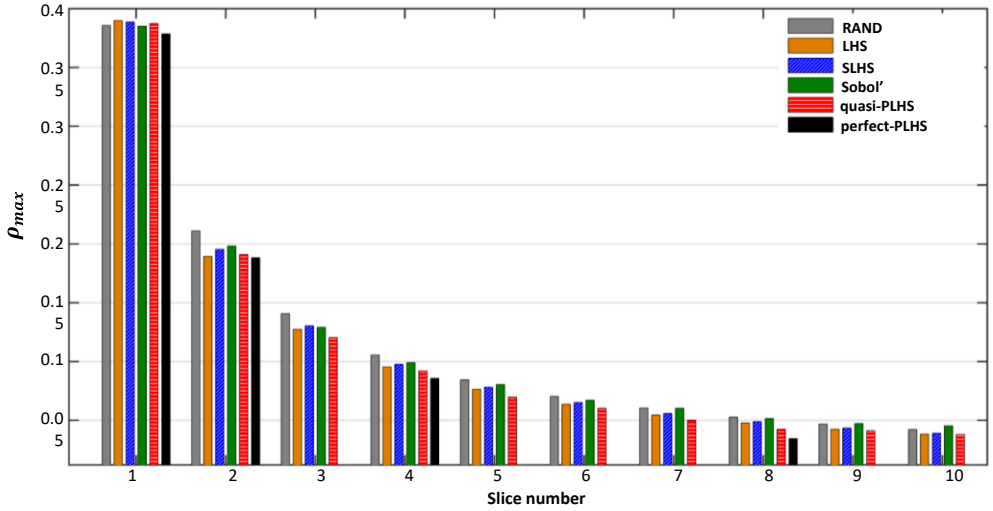


Figure 2-9 Comparison of different sampling strategies in terms of maximum pairwise correlation between the factors (Eq. (8)) – The results are average of 100 replicates for the 100-dimenasional input space.

2.5.2 Estimating the mean and variance of the 2-D problem

Fig. 2-10 shows the performance of the different sampling algorithms, as their sample size grows, in estimating the mean and variance of the 2-dimensional test problem described by **Eq. (8)**. The average error of these estimates was computed as the proportion of the "true" mean (1.6667) and variance (1.2667). According to **Fig. 2-10a** and b, the Sobol', perfect- and quasi-PLHS methods outperformed the other algorithms, as they resulted in the lowest average error (over 100 replicates) in estimating both of the mean and variance. To further scrutinize this observation, **Fig. 2-10c** and d show the standard deviation of the estimates of the mean and variance over the 100 replicates. Both perfect- and quasi-PLHS resulted in the least standard deviation among all the algorithms for any given number of function evaluations; however, the Sobol' method has the higher values of standard deviations. SLHS performs better than LHS and RAND in these analyses. Moreover, the empirical cumulative distribution functions (CDFs) of the estimates are compared in **Fig. 2-10c** and d for the 80th slice, i.e., after 800 function evaluations. These results indicate the superiority of the proposed variants of PLHS compared with the alternatives (1) in approximating statistics of a model response surface with fewer function evaluations (model runs), and (2) in terms of robustness against sampling variability and

randomness, as with perfect- and quasi-PLHS, the estimates of the mean and variance across the 100 replicates with different random seeds were significantly closer to each other.

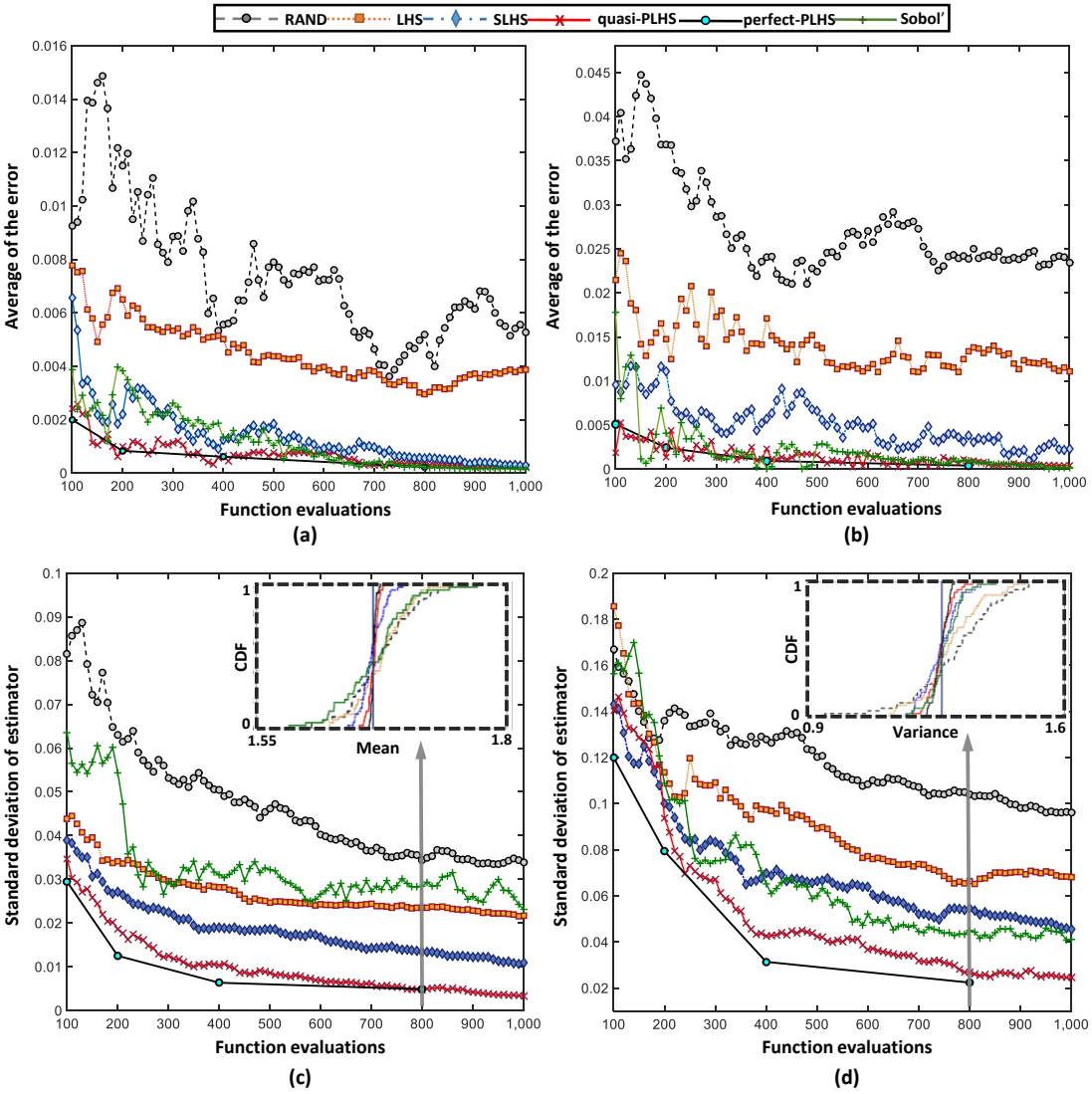


Figure 2-10 Case study 2: Comparison of different sampling strategies in estimating the mean (a & c) and variance (b & d) of the 2-dimensional problem. In (a) and (b), the deviations (errors) from the true mean and true variance were averaged over 100 replicates. In (c) and (d), the standard deviation of the estimated mean and variance (regardless of the true values) over the 100 replicates were calculated.

2.5.3 Comparison in a sensitivity analysis context

Fig. 2-11 & **12** demonstrate the performance of different sampling algorithms in global sensitivity analysis of the test problems. The 5th and 95th percentiles (90% interval) of the 100 replicates of this experiment with different random seeds were investigated to gain a view of the range of possible performances of different algorithms. This also helped to assess the robustness of the sensitivity analysis to sampling variability. The first test problem has only two input variables (x_1 and x_2) where parameter x_2 is more sensitive than x_1 according to all used sensitivity metrics. The second test problem has 5 parameters, and their ranks according to $IVARS_{50}$ and VARS-TO are as follows R_q, C_{max}, *alpha*, b_{exp}, and R_s. Results of $IVARS_{10}$ and VARS-TO for test problem 1 and $IVARS_{50}$ and VARS-TO for test problem 2 were (arbitrarily) chosen to be shown.

As can be seen, for the first test problem perfect-PLHS, quasi-PLHS, and Sobol' techniques outperformed the other sampling algorithms, as their 90% intervals are consistently narrower for any sample size for the 2-D case study. However, the performance of Sobol' method decreased for the sensitivity analysis of the 5-D HYMOD model, particularly in lower number of function evaluations. As expected, perfect-PLHS worked slightly better than quasi-PLHS, at the trade-off of not providing flexibility in sample size. The performance of SLHS was significantly better than LHS and RAND in most cases, and RAND comes the last one. These results indicate that enabling VARS with PLHS improves the convergence and robustness of sensitivity analysis, specifically in more complex problems.

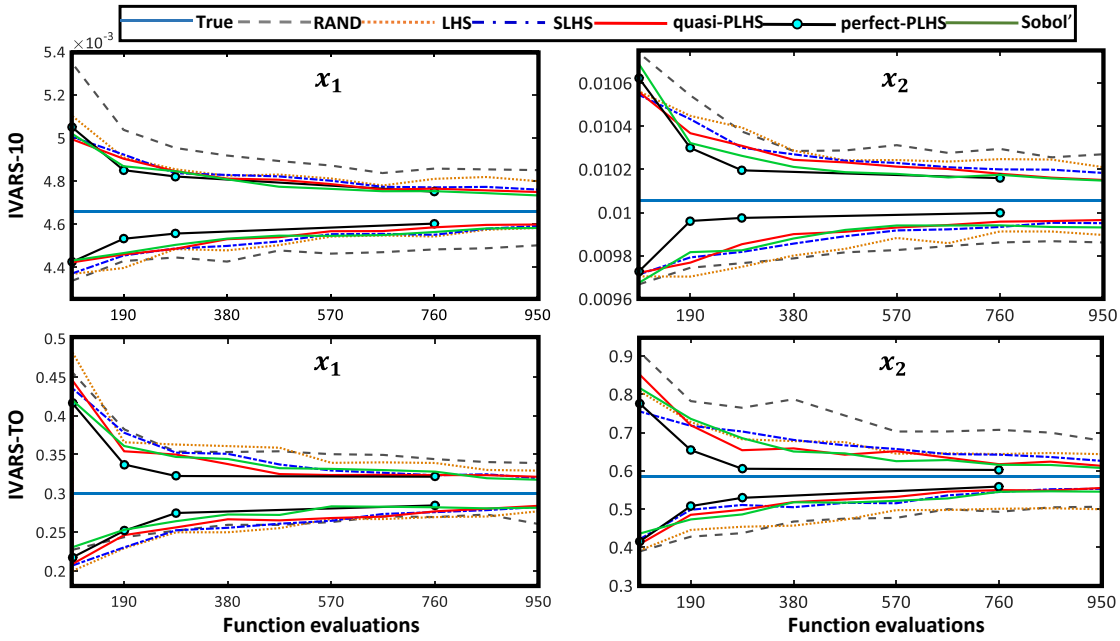


Figure 2-11 Comparison of different sampling strategies in global sensitivity analysis of 2-D problem using the VARS method. The 5th and 95th percentiles of the 100 replicates are shown along with true values. Here, the $IVARS_{10}$ and VARS-TO (Total-Order effect) metrics were illustrated.

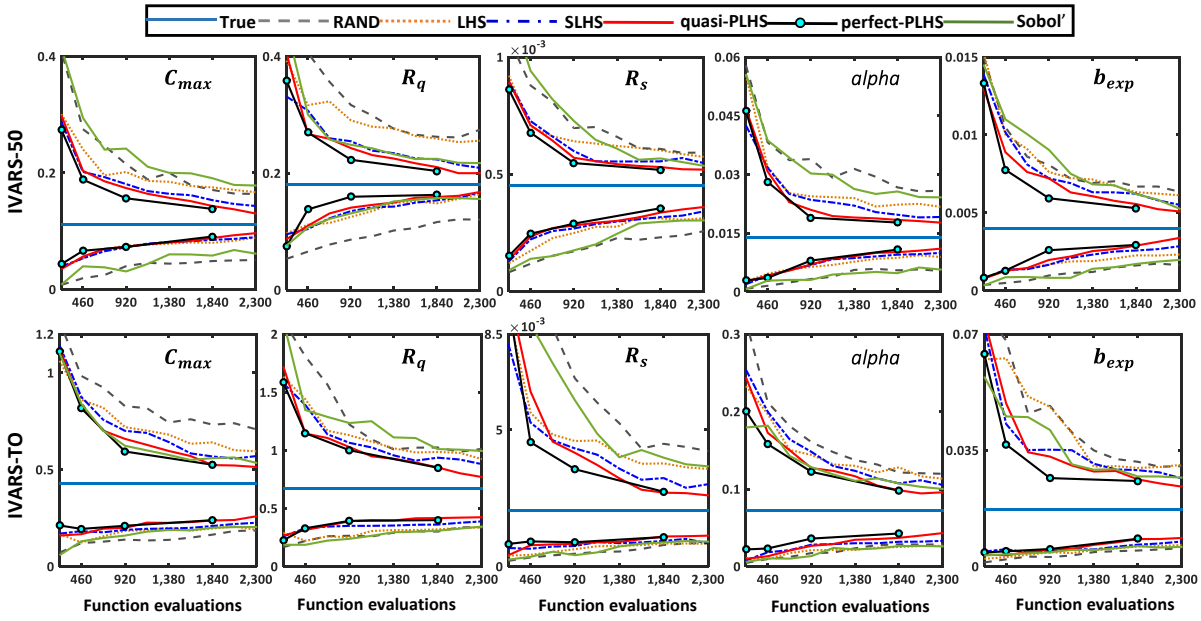
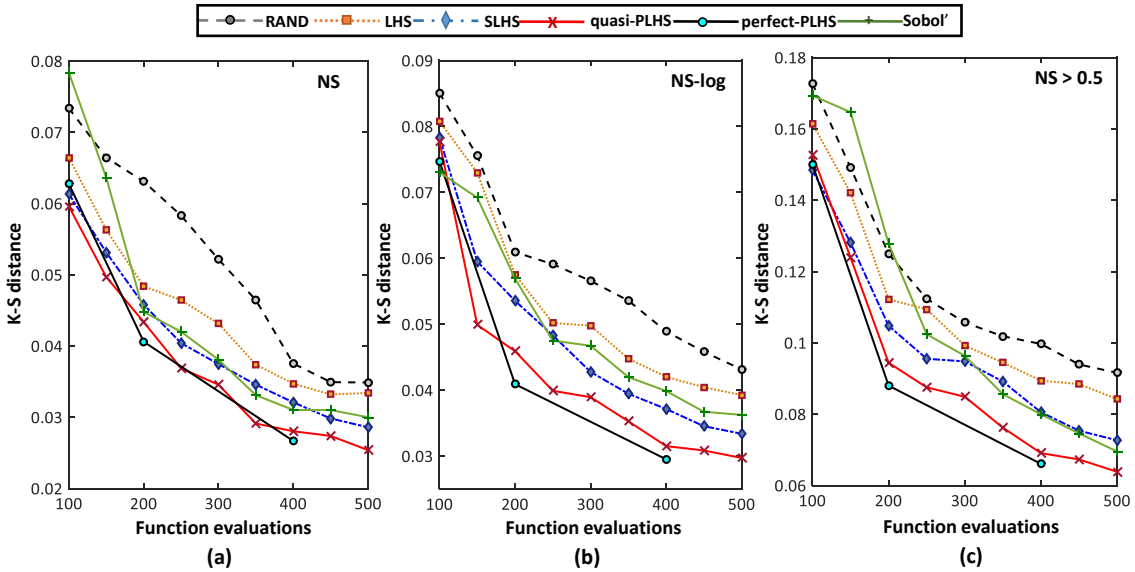


Figure 2-12 Comparison of different sampling strategies in global sensitivity analysis of NS criterion to HYMOD model parameters. The top panel is for the $IVARS_{50}$ metric and the bottom panel is for VARS-TO (Total-Order effect) metric. The 5th and 95th percentiles of the 100 replicates are shown along with true values as the sample size grows.

2.5.4 Comparison in an uncertainty analysis context

Fig. 2-13 shows the performance of the different sampling algorithms in approximating the CDFs of the model outputs in a Monte-Carlo simulation setting. Also, to assess the efficiency of different sampling algorithms in GLUE-type analyses (Beven and Binley, 1992), we set a behavioral/non-behavioral threshold of NS = 0.50 (**Fig. 2-13c**). As can be seen, both variants of PLHS resulted in minimum average K-S and D^2_N similarity measures almost everywhere, followed by Sobol', SLHS, LHS, and RAND. This superiority demonstrates that PLHS can adequately explore the model response surface for any sample size and can characterize the CDF of the model response more efficiently, within less numbers of model runs.



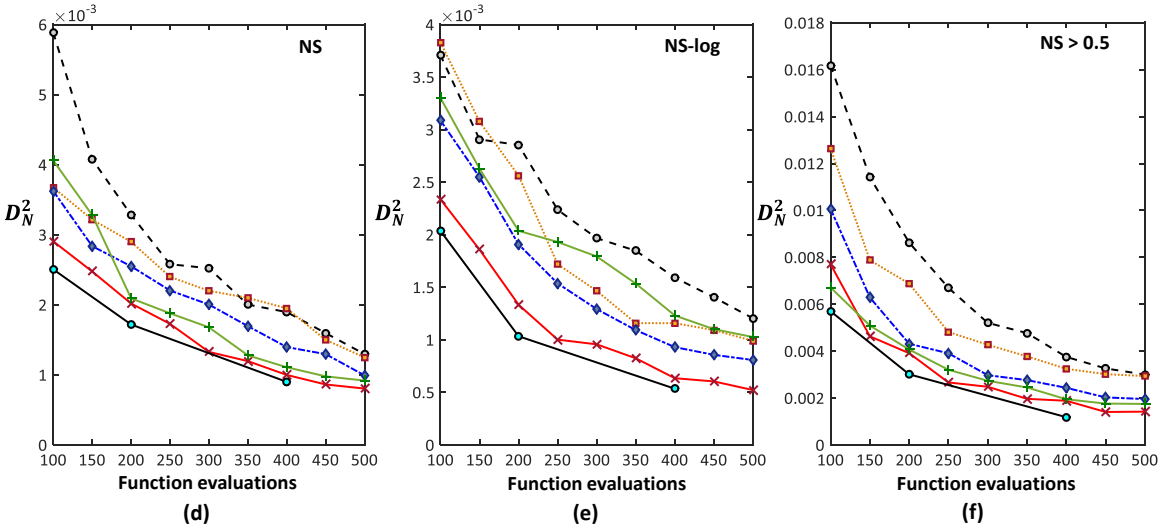


Figure 2-13 Comparison of different sampling strategies in approximating the CDFs of the HYMOD model using K-S distance (top panel) and energy distance (bottom panel) metrics. Subplots (a) and (b) show the results for NS and NS-log, respectively. Subplot (c) shows the results for NS when of interest is to approximate the CDF of the model outputs with NS > 0.5 (only model outputs for behavioral parameter sets) to assess the sampling performance in a GLUE-type analysis – in each plot, the values were averaged over 100 replicates.

This feature of PLHS is helpful when performing density-based (moment-free) uncertainty and sensitivity analysis, such as in regional sensitivity analysis approach of Hornberger and Spear (1981). The density-based techniques aim to characterize uncertainty and sensitivity in terms of the entire distribution of the model output (density functions). Typically, these methods measure the difference between the conditional and unconditional density functions using a distance-based metrics such as Minkowski class of distance (Chun et al., 2000) or δ-density (Borgonovo, 2007) or entropy-based metrics such as Shannon entropy (Krykacz-Hausmann, 2001) or Kullback-Leibler entropy (Liu et al., 2006). Regardless of the chosen metric, quantifying density-based statistics with desirable accuracy and robustness requires many function evaluations, which may come with a high computational burden (Castaings et al., 2012). The proposed PLHS can be an alternative sampling approach in performing density-based techniques by reducing the computational cost and improving the robustness and accuracy of estimations.

2.6 Conclusions

Modern environmental models are typically characterized by complex response surfaces, large parameter/problem spaces, and high computational demands. These attributes impede effective implementation of various sampling-based analyses which require running such computationally intensive models many times to adequately explore and characterize the model response surface across the parameter/problem space. In this context, the proper choice of sample size to maximize the amount of information extracted from the model and the proper distribution of the sample points in the input space are very important.

To address these issues, in this chapter we introduced a novel strategy, called PLHS (Progressive Latin hypercube sampling), for sequentially sampling the input space while progressively maintaining the Latin hypercube properties. The proposed PLHS is composed of a series of smaller slices generated in a way that the union of these slices from the beginning to the current stage optimally preserves the desired distributional properties and at the same time achieves maximum space-filling. Motivations behind developing PLHS include:

- PLHS rectifies a disadvantage of the original LHS that new sample points cannot be added sequentially to the sample set. The lack of this capability limited the utility of LHS in any sampling-based analyses where termination criteria are often checked incrementally.
- PLHS is superior to traditional sequential sampling strategies such as Halton, Hammersley, and Sobol' sequences in most cases, as unlike those, it preserves projection properties along with other desired sample properties, particularly in high-dimensional problems.

We tested the performance of PLHS against benchmark sampling strategies by numerical experiments across four case studies using two test problems. These experiments were designed to evaluate PLHS in the input space in terms of space-filling, correlation, and projection properties and in the output space in terms of statistical measures and sensitivity metrics. The numerical experiments indicated that PLHS can minimize the computational burden of sampling-

based analyses of computationally expensive models by conducting only model runs that are necessary to achieve the results of desired quality.

Author contributions

RS developed the method, wrote the computer codes, and performed all the numerical experiments. RS and SR contributed to the interpretation of the results, structuring and formulation of the paper. RS wrote the paper with contributions from SR. All co-authors contributed to editing of the paper.

Chapter 3

Characterizing the Role of Internal Parameters on the Functioning of River Ice Model Using Global Sensitivity Analysis

This chapter is a mirror of the following published article with minor changes to increase its consistency with the body of the dissertation. Changes were only made to avoid repeating the contents that have been presented more appropriately in other parts. References are unified at the end of the dissertation.

Sheikholeslami, R., Yassin, F., Lindenschmidt, K.E. and Razavi, S., 2017. Improved understanding of river ice processes using global sensitivity analysis approaches. *Journal of Hydrologic Engineering*, 22(11), p.04017048. https://doi.org/10.1061/(ASCE)HE.1943-5584.0001574

Synopsis

The high impact of river ice phenomenon on the hydrology of cold regions has led to the extensive use of numerical models in simulating and predicting river ice processes. Consequently, there is a need to utilize efficient and robust sensitivity analysis methods to characterize the role of different parameters on the functioning of these models. To gain greater insight into how the internal parameters affect a river ice model's behavior, this chapter presents a comparative performance investigation of the two global SA methods: (1) the recently proposed Variogram Analysis of Response Surfaces (VARS), and (2) the widely-used Regional Sensitivity Analysis (RSA). The methods are benchmarked on a one-dimensional hydrodynamic river ice model of the Lower Dauphin River, Manitoba, Canada. Furthermore, using a bootstrapping strategy, a procedure is developed to estimate confidence intervals on the resulting sensitivity indices and evaluate reliability of the inferred parameter rankings.

3.1 Introduction

For a majority of cold regions, river ice has a significant impact on hydrologic processes and is a potential cause of extreme floods and low winter flows (river ice formation can abstract water making this stored water unavailable for flow until the next spring). Dynamics of the formation, transport, and decay of the river ice –as complex physical phenomena– are generally governed by the interactions between river geometry, flow conditions, and mechanical and thermal energy transfer between ice, water, and atmosphere (Beltaos and Prowse, 2009; Ohara et al., 2014). River ice, in turn, may significantly affect the morphological and hydraulic properties of rivers. For example, water velocity distribution, channel conveyance capacity and sediment transport can be changed by ice cover development because the ice adds an extra boundary layer on top of the river and increases the flow resistance (Ettema and Daly, 2004; Aghaji Zare et al., 2015). The accuracy of real-time streamflow data can also be reduced during ice-affected periods due to the blocking of the channel by ice and increased flow resistance (Holtschlag and Grewal, 1998). River ice also modifies chemical processes (e.g., oxygen exchange), stream and wetland ecology, and various in-stream, deltaic, and riparian habitats (Prowse, 2001; Hicks et al., 2006; Lindenschmidt and Sereda, 2014).

Furthermore, river ice can degrade the performance of engineering structures, by damaging hydraulic structures (e.g., dams and bridges), hampering water supply or intakes, and impeding river navigation. Ice-induced flooding can be more severe than open water flooding, thereby threatening human safety and quality of life (Hicks, 2009, Xiong and Xu, 2009; Zufelt and Walton, 2012). Therefore, the provision of technically feasible options for river ice management (monitoring, control, and mitigation) plays a crucial role in flood control and risk assessment and is vital for water resources planning and management in cold regions.

A successful river ice management plan may heavily utilize computer-aided analysis (i.e., mathematical models) to simulate and predict river ice processes based on the available amount of field data. Since the 1990s, research into river ice modelling has steadily advanced through the introduction of several numerical models, such as RICE (Lal and Shen, 1991), RIVJAM (Beltaos, 1993), RICEN (Shen et al., 1995), ICEPRO, ICESIM (Carson et al., 2001), DynaRICE (Shen et al., 2001), CRISSP1D (Chen et al., 2006), RIVICE (EC, 2013), and YRIDM (Fu et al.,

2014). These models can be divided into static and dynamic models. Dynamic models, which consider dynamic ice conditions, can be further divided into one-dimensional (e.g., YRIDM) and two-dimensional models (e.g., DynaRICE). Generally, river ice models are non-linear and characterized by several parameters, which may have wide variation domains and be difficult to measure, leading to high uncertainty in their values.

Despite significant research efforts focused on understanding and modelling river ice processes, there is still a need for an improved characterization of the role and impact of different parameters on the functioning of the system. Sensitivity analysis (SA) can be helpful in this respect by providing diagnostic insights into the river ice models and identifying the key factors (i.e., boundary conditions, forcings, and parameters) controlling ice dynamics and model performance. Hence, in modelling practice, it is advisable to employ SA, adjusted to the specific needs of river ice modelling.

Various methods are available to carry out SA, for example derivative-based methods, one-factor-at-a-time methods, regression-based methods, regional sensitivity analysis, and variance-based methods (see, e.g., Hall et al., 2009; Saltelli et al., 2010 for general discussion). These techniques can be broadly classified into local and global SA methods. Local methods measure how model outputs respond to local variations of the uncertain input factors around a single point in the factor space. The limitations of local SA methods have been shown in previous studies, particularly, when applied to hydraulic and hydrologic models (Tang et al., 2007; Saltelli and Annoni, 2010). By contrast, global methods focus on the influence of an uncertain input factor across the entire factor space. Apart from the many successful applications of existing global SA (GSA) methods, two major challenges persist with GSA (Razavi and Gupta, 2015), namely: (1) an ambiguous definition of global sensitivity, and (2) high-computational demand. The former indicates that different GSA methods are based on different philosophies; hence, different methods lead to different and sometimes conflicting assessments of sensitivity. The latter is due to current-generation GSA methods requiring simulation models (e.g., river ice models) to be run many times to achieve reliable and robust sensitivity results. These challenges and outlook of SA have been comprehensively discussed in Gupta and Razavi (2017).

To overcome these challenges, Razavi and Gupta (2016a) recently proposed a novel GSA framework, called Variogram Analysis of Response Surfaces (VARS). VARS provides a comprehensive spectrum of information about the underlying sensitivities of a model to its factors. Unlike existing global methods, VARS characterizes important sensitivity-related properties of the model outputs, including local sensitivities and their global distribution, the global distribution of model responses, and the spatial structure of the model response surface. VARS develops and utilizes directional variograms and covariograms that contain useful information about the variability of the model response surfaces in a factor space across a full range of parameter "perturbation scales". The concept of perturbation scale is included in the VARS framework, which can overcome the perturbation scale issue of traditional GSA methods (from the small-scale features such as roughness and noise to large-scale features such as multimodality). Its effectiveness, efficiency, and robustness has been tested in dealing with several mathematical benchmark problems and two real-world hydrological models (Razavi and Gupta, 2016a,b).

With the use of GSA, this chapter aims to evaluate the importance and identifiability of different parameters of river ice modelling. Here, our numerical experiments are performed through a one-dimensional hydrodynamic river ice model, RIVICE, which was originally developed by Environment Canada (EC, 2013). The model has been successfully applied to a number of studies in Canada, such as simulating the behavior of ice jams along the Red River in Winnipeg (Lindenschmidt et al., 2011), modelling the ice cover formation during winter freeze-up in Lake St. Martin and the Dauphin River (Lindenschmidt et al., 2012), and evaluating the impact of macrophyte growth on the probability of overbank flooding during winter in the Upper Qu'Appelle River (Lindenschmidt and Sereda, 2014).

In this chapter, the VARS and the widely used regional sensitivity analysis (RSA) method of Hornberger and Spear (1981) are utilized and compared using the RIVICE model. In addition, a bootstrapping strategy is developed to assess the reliability of the results obtained from both methods. The model was used to simulate river ice processes in the Lower Dauphin River, Manitoba, Canada. To understand the geomorphological controls on the formation of ice covers along the Dauphin River, Lindenschmidt and Chun (2013) previously applied the traditional

RSA method and compared the parameter sensitivities of two modelling exercises of the upper and lower stretches of the river. However, in this chapter, we used a newly developed VARS framework as a new GSA approach, with the aim of providing a comparative performance investigation of the two methods. Moreover, as discussed in the Methodology section, we developed an algorithm to estimate confidence intervals on the resulting sensitivity metrics and evaluate reliability of the inferred parameter rankings. Therefore, specific contributions of this study are: (1) improving our understanding of the importance and identifiability of different parameters involved in river ice modelling; (2) investigating performance of the newly developed VARS for GSA of a river ice model; and (3) comparing the effectiveness and reliability of the VARS with the well-known RSA method.

3.2 RIVICE Model Overview

The key river ice processes simulated in RIVICE that are relevant to this study are shown in **Fig. 3-1** and briefly described below.

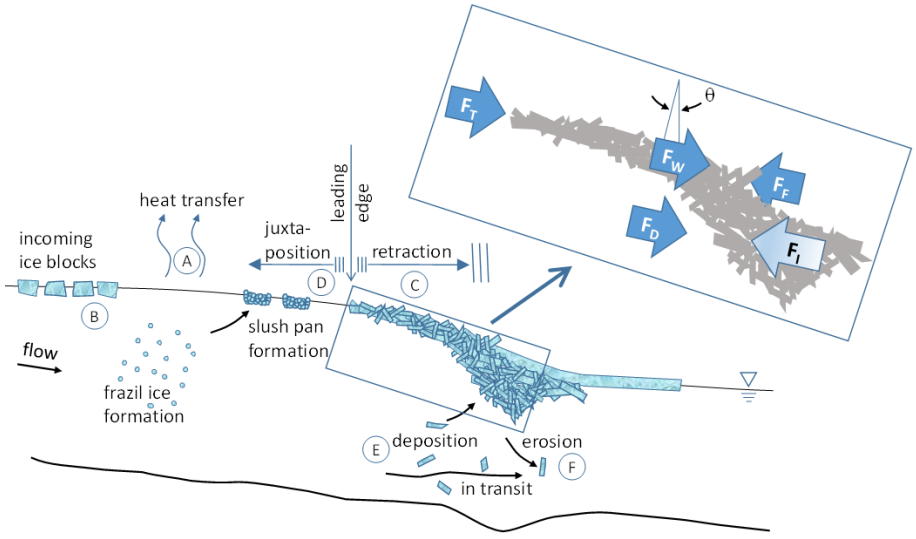


Figure 3-1 River ice processes simulated in RIVICE.

There are two sources of ice for the establishment of an ice cover and/or ice jam. The first source is frazil ice ('A' in **Fig. 3-1**) that is generated in the river when the overlying air temperature is freezing, inducing a transfer of heat from the river water to the atmosphere, and

the river water temperature drops to a fraction below 0 deg. C (supercooling). The frazil crystals conglomerate into flocs and further into slush pans that float to the top and flow along the water surface to the leading edge of the downstream ice cover. The second source is the volume of inflowing ice per time step ('B' in **Fig. 3-1**), representing ice blocks broken apart from upstream ice sheets or border ice, or additional slush pans from frazil ice generated upstream of the model control volume. This ice floats along the water surface at the mean flow velocity of the river until it reaches the downstream ice cover's leading edge.

Once the ice reaches the leading edge, two processes are at hand for the progression of the ice cover:

(1) The first process is the retraction of the ice cover ('C' in **Fig. 3-1**) in the downstream direction through shoving of the ice, which thickens the already existing ice cover further downstream (telescoping). Shoving occurs when the summation of external forces on the cover – thrust of the flowing water against the leading edge FT, the weight of the ice cover in the sloping direction FW and the drag force on the ice cover's underside by the flowing water FD – exceed the ice cover's internal resistance FI plus the frictional force of the ice cover along the river banks FF. Shedding of longitudinal forces in the ice cover laterally to the river banks constitutes the frictional force and the thickening of the ice cover, which distributes longitudinal forces along the thickness of the ice. Shoving continues until FI > FT + FW + FD – FF. External forces due to the cohesion of the ice cover to the river banks were not incorporated into the Dauphin River model due to the high river discharge at freeze-up that led to a rapid progression of the ice front and formation of the ice cover.

(2) The second process is the progression of the ice cover upstream through juxtapositioning of the ice cover ('D' in **Fig. 3-1**) when the internal resistance within the cover FI plus the frictional force FF remain larger than the summation of the external forces, FT, FW and FD, and the ice blocks and/or slush pans accumulate at the leading edge, stacking up against each other to extend the ice cover upstream. As more and more ice accumulates, external forcing anywhere along the juxtapositioned ice cover may be large enough for collapsing and shoving of the ice cover to occur.

Ice under the cover may be eroded and transported downstream as ice in-transit. Should the mean flow velocity drop to below a velocity threshold value $v_{deposit}$, the ice will deposit on the ice cover underside ('E' in **Fig. 3-1**). If the mean flow velocities underneath the ice cover increase and exceed a threshold value v_{erode} the ice will erode from the underside ('F' in **Fig. 3-1**).

Roughness of the river bed and the undersurface of the ice cover are important parameters controlling the hydraulics of the flow and ice regimes. Bed roughness is a constant value represented by Manning's coefficient, while ice cover roughness is a function of ice cover thickness. Important boundary conditions are the upstream discharge of the water entering the modelled stretch of the river and the downstream water level elevation where the water exits the stretch.

3.3 Methodology

This section briefly describes the utilized GSA methods as well as the bootstrapping procedure used to assess the level of confidence in the GSA results.

3.3.1 Regional sensitivity analysis (RSA)

Regional sensitivity analysis (RSA), also called generalized SA and Monte Carlo filtering, is a widely-used SA method proposed by Hornberger and Spear (1981). In RSA the basic idea is to generate distributions for each parameter, extracted from a portioning of the Monte Carlo simulations into "behavioral" and "non-behavioral" classes, and comparison of the similarities between their distributions. To do this, first, *n* input parameters are randomly sampled from a feasible range of the parameter space. Then the model is evaluated using these random parameters. In the second step, the parameters are divided into behavioral and non-behavioral sets according to the associated model output, whether it is above or below a pre-defined threshold value based on a goodness of fit measure. Finally, using the empirical cumulative distribution functions (CDFs) of the model outputs, the behavioral parameters are compared to the non-behavioral parameters for detection of the significant differences between both groups. If the CDFs of an input parameter in the two sets (i.e., behavioral and non-behavioral) are

dissimilar, then the parameter is deemed influential. On the other hand, strong similarity between the CDFs reveals an insensitive parameter.

The RSA method has several advantages such as being conceptually simple, ease of use, and being model-independent. However, a major disadvantage is that results of RSA are highly dependent on the choice of threshold value (filtering criterion) to separate the model outputs into behavioral and non-behavioral sets, which is usually subjective. In other words, different filtering criteria may result in different sensitivities and parameter ranks (Fraga et al., 2016; Song et al., 2015; Yang, 2011).

In RSA, the classical Kolmogorov-Smirnov (K-S) measure (Kolmogorov, 1933) is usually employed to determine if the behavioral and non-behavioral parameter sets come from the same probability distribution. The K-S measure calculates the maximum distance between the two CDFs as follows:

$$\text{K–S}(i) = \max_{x_i} |CDF_b(x_i | y \in Y_b) - CDF_{nb}(x_i | y \in Y_{nb})| \quad (1)$$

where y is the model output, CDF_b and CDF_{nb} are the behavioral and non-behavioral CDFs of an input parameter x_i, and Y_b and Y_{nb} are the subsets of behavioral/non-behavioral outputs, respectively. Hence, a larger K–S(i) value corresponds to a higher sensitivity of the parameter x_i.

3.3.2 Variogram analysis of response surfaces (VARS)

Consider a response surface of a numerical model as $y(\boldsymbol{X}) = \boldsymbol{f}(x_1, x_2, \ldots, x_d)$, where $\boldsymbol{X} = \{x_1, x_2, \ldots, x_d\}$ is a vector representing parameters in d-dimensional space. Let $\boldsymbol{X}^i$ and $\boldsymbol{X}^j$ be the locations of two distinct points in the parameter space separated by a distance $\boldsymbol{h}$. The variogram can be defined as the expected difference between any pair of variables $y(\boldsymbol{X}^i)$ and $y(\boldsymbol{X}^j)$. Assuming a constant mean, the variogram can be formulated as (Cressie, 1993):

$$\gamma(\boldsymbol{h}) = \frac{1}{2} E\left[\left(y(\boldsymbol{X} + \boldsymbol{h}) - y(\boldsymbol{X})\right)^2\right] \quad (2)$$

Using **Eq. (2)**, an empirical variogram can be estimated for response surface from square increments computed from a sample set of values $y(\boldsymbol{X}^i)$ taken at several locations $\boldsymbol{X}^i$. That is

$$\gamma(\boldsymbol{h}) = \frac{1}{|N(\boldsymbol{h})|}\sum_{(i,j)\in N(\boldsymbol{h})}\left(y(\boldsymbol{X}^i) - y(\boldsymbol{X}^j)\right)^2 \quad (3)$$

where $N(\boldsymbol{h})$ denotes the set of all pairs (i , j) such that $\left\|\boldsymbol{X}^i - \boldsymbol{X}^j\right\| = \boldsymbol{h}$, and the number of pairs in $N(\boldsymbol{h})$ is $|N(\boldsymbol{h})|$.

For any given set of ordered h_k (for $k = 1, 2, \ldots, d$), the associated variogram estimates (**Eq. (3)**) are denoted by $\gamma_k = \gamma(h_k)$, and the empirical variogram for a d-dimensional response surface can be obtained by calculating the set of all (h_k, γ_k). The VARS framework utilizes information about the regional variability of the response surfaces in a parameter space obtained from $\gamma(\boldsymbol{h})$. This variability determines the spatial structure of the model response surfaces as well as the sensitivity of the parameter concerned.

Generally, the larger the variability γ_k the more heterogeneous is the response surface along the kth direction/parameter, at the perturbation scale represented by h_k. Accordingly, in the VARS framework, the γ_k values for any given h_k can be considered as a comprehensive illustration of sensitivity information, through linking variogram analysis to both direction and perturbation scale concepts. In fact, the variograms are defined over the full range of scales; therefore, the rate of variability at a particular perturbation scale can represent the perturbation scale-dependent sensitivity of the response surfaces. In other words, the γ_k at very small values of h_k shows the average small-scale sensitivity of the model to kth parameter, whereas at large values of h_k, it indicates the average larger-scale sensitivity. The effect of parameter perturbation scale on sensitivity analysis of environmental models has been discussed by Haghnegahdar and Razavi (2017). They showed that perturbation scale has a significant impact on the results of GSA. Hence, using the strategies that consider this scale-dependency (e.g., VARS) may be advisable.

Based on the above discussion, the VARS framework provides a set of sensitivity metrics by integrating the directional variograms within a perturbation scale range of interest. Considering a perturbation scale, ranged from zero to H_k, for the kth input parameter, the ‘‘*Integrated Variogram*’’ can by defined as:

$$\Gamma(H_k) = \int_0^{H_k}\gamma(h_k)dh_k \quad (4)$$

In VARS, the integrated variogram across a range of perturbation scales, $\Gamma(H_k)$, is called IVARS. Therefore, the IVARS-based sensitivity indices can be employed to rank the different parameters in terms of their influence on model outputs. In this study, $IVARS_{10}$, $IVARS_{30}$, and $IVARS_{50}$ are used, which means that **Eq. (4)** is computed for H_k = 10%, 30%, and 50% of the parameter range, respectively.

3.3.3 Bootstrapping strategy for reliability assessment

Sampling from the parameter space is a building block of implementing GSA algorithms such as RSA and VARS. However, sampling-based techniques which explore the space of possible model inputs often require a large sample size to obtain reliable and robust results. The bootstrap technique, introduced by Efron (1979, 1982), can be used as an efficient way of estimating the confidence level of statistical parameters. Bootstrapping is based on the idea of generating *p* samples by re-sampling with replacement from the initial sample set (*p* is a large number, e.g., 100 or 1,000). Based on these bootstrap re-samplings, *p* estimations of the statistical parameters (e.g., sensitivity metrics) can be calculated. In the re-sampling process, the new samples are extracted from the original sample set, and thus additional model runs are not necessary, which makes the method computationally efficient. Finally, the distribution of the parameters can be derived through these p estimated values.

It should be noted that the representativeness of the distributions resulting from bootstrapping depends on the quality of the original sample set. In other words, the resulting distributions are only approximations of the true distributions. The VARS framework is originally enabled by bootstrapping (Razavi and Gupta, 2016b). In this study, based on the above-mentioned bootstrapping strategy, an algorithm is developed to provide confidence level estimates (95% intervals) for the RSA sensitivity metrics and to evaluate the reliability of inferred sensitivity results (i.e., parameter sensitivity ranking). A brief description of the steps in the utilized algorithm for bootstrapping is given below:

Step 1. Initialization.

- Generate a sample with size of *n* and set the number of bootstrap re-samplings as *p*.

Step 2. Bootstrapping.

- Draw *p* samples of size *n* (the bootstrap samples) with replacement from initial sample set.
- For each bootstrap sample, calculate the sensitivity metrics $\hat{\theta}_i$ for $i = 1, 2, \ldots, p$, using the related SA method.
- Estimate the confidence intervals on the results from the empirical distribution of $\hat{\theta}$.

Step 3. Reliability assessment.

- Compute the fraction of times among all *p* bootstrap attempts that the sensitivity ranks of the parameters are equal to the original sensitivity ranks obtained by the initial sample set.

3.4 Study Site

The Dauphin River serves as the terminus of the Lake Winnipegosis/Lake Manitoba/Lake St. Martin/Dauphin Lake catchment area, draining into Lake Winnipeg (see **Fig. 3-2**). Extensive flooding along the Assiniboine River in 2011 forced a diversion of the floodwaters from the Assiniboine River into Lake Manitoba, via the Portage Diversion, to reduce flood risk in Winnipeg. The diversion was as much as half of the Assiniboine River's discharge at the flood peak. This caused extensive flooding along the lake shores, particularly the southern basin of Lake Manitoba, whose surface area almost doubled at its peak floodwater elevation. Wind-driven seiches exacerbated the shoreline flooding, causing extensive damage to property and infrastructure. Many First Nations communities along Lake St. Martin had to be evacuated.

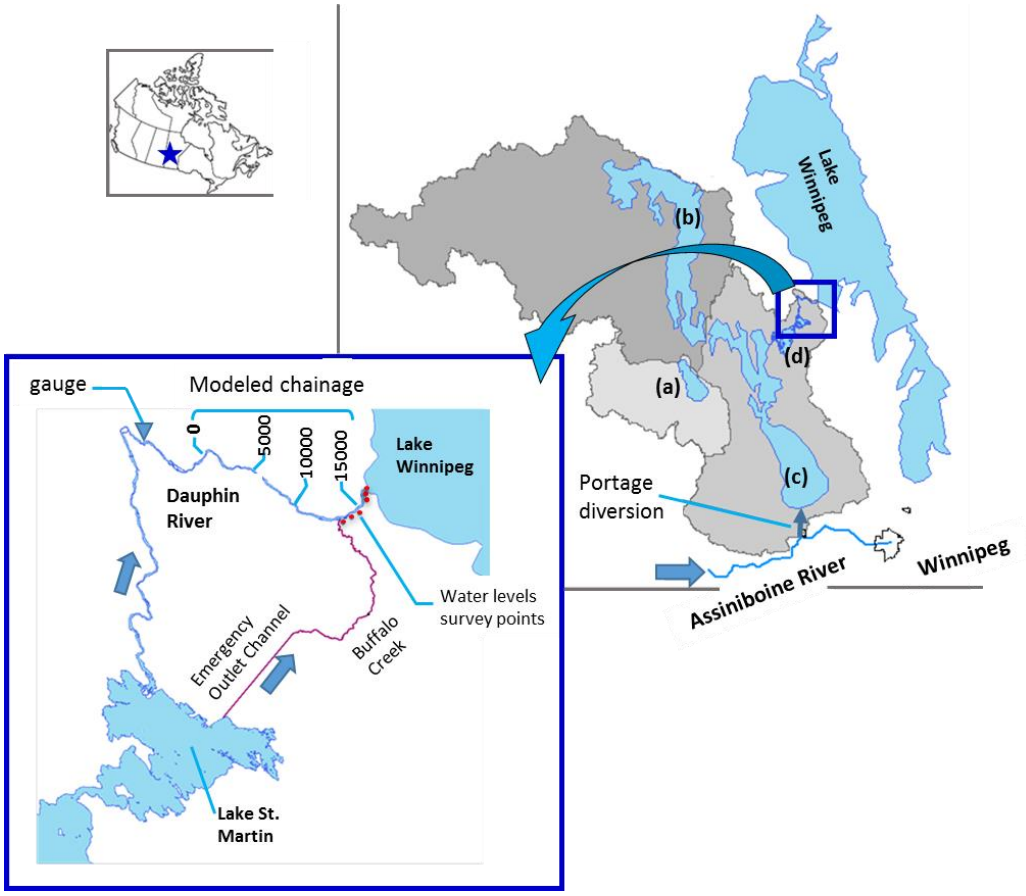


Figure 3-2 Catchment area of the Dauphin River consisting of the subbasins of the lakes: (a) Dauphin Lake; (b) Lake Winnipegosis; (c) Lake Manitoba; (d) Lake St. Martin; inset shows the Emergency Outlet Channel diverting water from Lake St. Martin into Buffalo Creek and onward into Lake Winnipeg [*adapted from Cold Regions Science and Technology, Vol. 82, Karl-Erich Lindenschmidt, Maurice Sydor, Richard W. Carson, "Modelling Ice Cover Formation of a Lake–River System with Exceptionally High Flows (Lake St. Martin and Dauphin River, Manitoba)," 36–48, 2012, with permission from Elsevier*].

The Dauphin River is the "bottleneck" for drainage from its upstream catchment area. Flows at freeze-up would have been double those ever recorded during the previous 60 years of gauge recordings. Due to the steep bed of the river's lower reach (= 0.017 m/m) and the high flows, extensive frazil ice generation and freeze-up jamming was anticipated, which would have exacerbated flooding in the communities along the Dauphin River and Lake St. Martin. Hence, an emergency outlet channel was constructed to divert approximately half of the Dauphin

River's flow from Lake St. Martin to Buffalo Creek, exiting near the Dauphin River's outlet at Lake Winnipeg.

Because of the long extent of the Dauphin River (~53 km), a longitudinal change of variables can be approximated in a 1-D simulation of RIVICE. The intensive river ice modelling was carried out to predict the degree of backwater flooding along the river and Lake St. Martin. Maximum backwater levels were predicted to serve as a benchmark for the construction of new dikes and raising of existing dikes prior to the freeze-up season (see Lindenschmidt et al., 2012; Lindenschmidt and Chun, 2013 for more details).

3.5 Computational Experiments

The RIVICE parameters and their allowable ranges that were used for the sensitivity analysis are shown in **Table 3-1**.

Table 3-1 RIVICE parameters considered for the RSA and VARS sensitivity analysis, and their ranges of variation

Parameter	Description	Lower bound [a]	Upper bound [b]	Unit
Ice cover properties				
PC	Porosity of ice cover	0.4	0.9	–
FT	Thickness at the ice cover front	0.16	0.26	*m*
Slush ice properties				
PS	Porosity of the slush	0.3	0.7	–
ST	Thickness of the slush pans	0.1	0.5	*m*
Strength properties				
K1TAN	Lateral: longitudinal forces (ratio)	0.10	0.22	–
K2	Longitudinal:vertical forces (ratio)	7	10	–
Hydraulic roughness				
n_{ice}	Ice roughness	0.07	0.13	$s/m^{1/3}$
n_{bed}	River bed roughness	0.025	0.035	$s/m^{1/3}$
Boundary conditions				
Q	Upstream discharge	240	300	m^3/s
W	Downstream water level	214	219	*masl*

[a,b] The values are adopted from the RIVICE manual (EC, 2013) and Lindenschmidt and Chun (2013)

To assess the performance of RIVICE under each parameter set, we used the root-mean-squared-errors (RMSE) metric between the simulated and observed values of the water levels:

$$F_{obj} = RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_s - y_o)^2} \quad (5)$$

where y_s and y_o are the simulated and the observed water level values, respectively, and N is the number of observation points for the Lower Dauphin River. The maximum water level elevations recorded at the gauges located along the river (as shown in **Fig. 3-2**) was used as the observation points for the Lower Dauphin River model.

Moreover, in this study progressive Latin hypercube sampling (PLHS), introduced in **Chapter 2**, was used instead of ordinary Latin hypercube sampling (LHS) to generate sample points for function evaluation. Sheikholeslami and Razavi (2017) showed that PLHS has some advantages over traditional LHS in terms of different criteria such as space-filling and one-dimensional projection properties. For RSA using PLHS algorithm, 1,000 parameter sets were randomly selected from a uniform distribution within the parameter ranges listed in **Table 3-1**. After calculating the F_{obj} values for each parameter set, the best 10% of the parameter sets in terms of F_{obj} (the simulations most consistent with the observations) were deemed behavioural; the other 90% was deemed non-behavioural.

For VARS, we applied the STAR-VARS implementation of VARS developed by Razavi and Gupta (2016*b*). STAR-VARS uses a specific sampling procedure to generate sample points, called star-based sampling. First, using PLHS technique, star centers are generated randomly, and then a structured-sampling approach is used to sample star points from the parameter space. Here, the number of star centers was set to 10, which results in a total of 910 function evaluations of RIVICE for VARS (the total sample size includes all sample points and star centers). Also, the Δh (VARS resolution parameter) was set to 0.1 (as recommended by Razavi and Gupta (2016b)).

Using a computer with an Intel Core i7 CPU, 3.6-GHz processor, and 16.0 GB RAM each run of RIVICE took approximately 90 minutes. To allow for a more comprehensive analysis of the

performance and reliability of the RSA and VARS methods in our numerical experiments, 95% confidence intervals of the sensitivity metrics were determined using p = 1,000 bootstrap replicates.

3.6 Results and Discussion

3.6.1 Sensitivity analysis and parameter rankings

Fig. 3-3 presents GSA results of the RSA method, by empirical cumulative distribution functions (CDFs) of behavioural versus CDFs of all parameter values. A larger distance between a behavioral (blue) CDF and the corresponding non-behavioral CDF (grey) indicates a higher sensitivity of the associated parameter. When parameters are insensitive (see, e.g., the n_{ice} or n_{bed} results given in **Fig. 3-3**), the behavioral and non-behavioral curves plot over each other in a linear trend, illustrating uniformly distributed F_{obj} values. As can be seen in **Fig. 3-3**, Q and PC are the most sensitive parameters. In addition, **Fig. 3-3** provides useful information for identifying sub-ranges of the model parameters that have no influence on the output above/below the threshold, which can clarify the identifiability of different parameters. These are the sub-ranges where the CDFs are either zero or one, for example $FT > 0.25$.

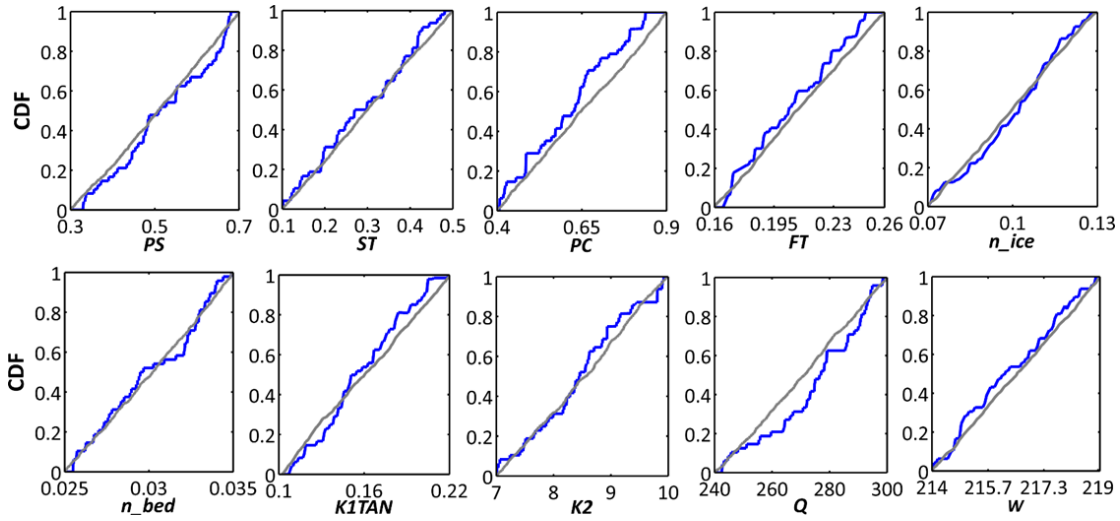


Figure 3-3 Evaluating the *RIVICE* parameters sensitivities using RSA; each subplot belongs to one model parameter, with its feasible range on the horizontal axis and the distribution of the *RIVICE* responses (cumulative RMSE distribution) on the vertical axis.

The VARS-based sensitivity indices are demonstrated in **Fig. 3-4,** where plots (**a**) and (**b**) show the directional variograms and integrated variograms, respectively, for the RIVICE parameters. Plots (**c**) and (**d**) display zoom-in images of plots (**a**) and (**b**). The values of the directional variograms (**Fig. 3-4c**) for less sensitive parameters (i.e., *ST*, n_{ice}, *PS*, and n_{bed}) are similar for the perturbation scale $h < 0.1$, but above this value the variogram of *ST* becomes significantly larger. As clearly shown in **Fig. 3-4a** and b, *PC*, *FT*, and *Q* are the most sensitive parameters. The IVARS values (**Fig. 3-4b**) for *FT* remain less than those for *Q* over the perturbation scale from $h = 0$ to about $h = 0.3$, indicating less small-scale sensitivity, while *FT* becomes more sensitive than *Q* for $h > 0.3$. These results confirm that, unlike the most traditional GSA methods, VARS method can characterize perturbation scale-dependency in sensitivity analysis (different sensitivity rankings at different perturbation scales) of the model response surfaces. In addition, by looking at the respective variograms, useful insights into the spatial structure of the model response surface can be gained. As demonstrated in **Fig. 3-4c**, the response surface exhibits strong non-monotonic and nonlinear behaviour for some parameters, such as *W* and n_{ice}.

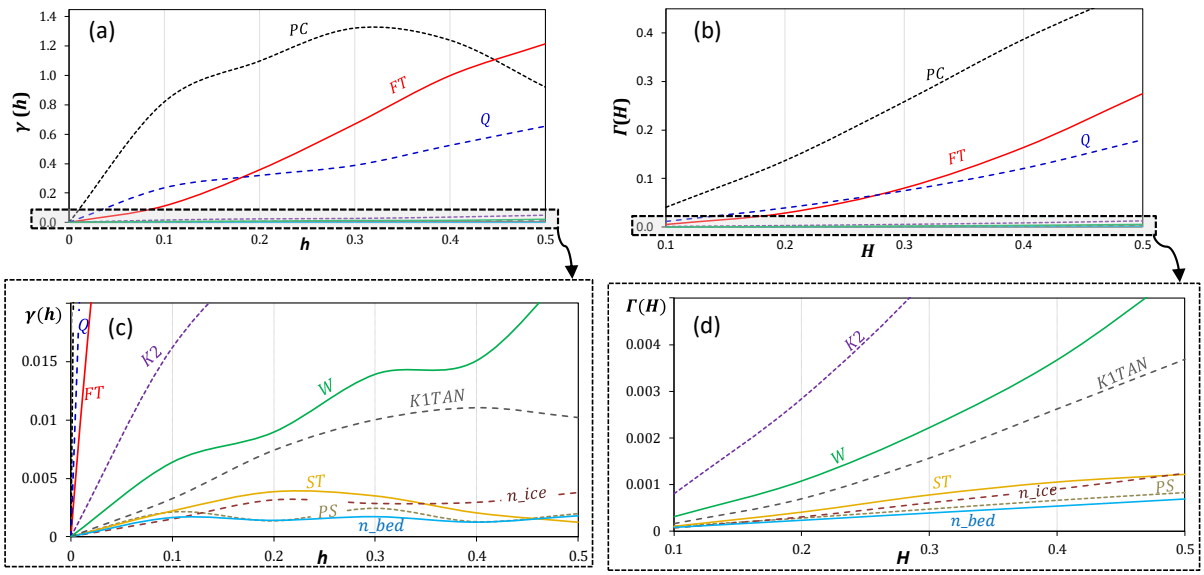


Figure 3-4 Evaluating *RIVICE* parameter sensitivities using VARS: (a) directional variograms; (b) integrated variograms (IVARS); the bottom plots (c and d) show a zoom-in of the top plots for very small values on the vertical axis (note that for variograms to remain meaningful, the distance between any two points within a given parameter range cannot exceed half of its range, i.e., Hk ≤ 50%).

Fig. 3-5 shows the sensitivity indices and associated parameter rankings using VARS (subplots (**a**) to (**c**)) and RSA (subplot (**d**)), where a ranking of 1 represents the least sensitive and a ranking of 10 represents the most sensitive parameter. As shown in **Fig. 3-5**, parameter sensitivity rankings of different SA metrics vary from each other. $IVARS_{50}$ is the most comprehensive metric generated by VARS to estimate the global sensitivity of the RIVICE response surface to each parameter. On the contrary, $IVARS_{10}$ and $IVARS_{30}$ provide smaller perturbation scale assessments of parameter sensitivity. It is interesting to note that the sensitivity of the n_{ice} parameter has a higher dependency on the perturbation scale as its ranking changes from 1 considering the $IVARS_{10}$ to 3 considering $IVARS_{30}$, while $IVARS_{50}$ ranks the same parameter in 4th place.

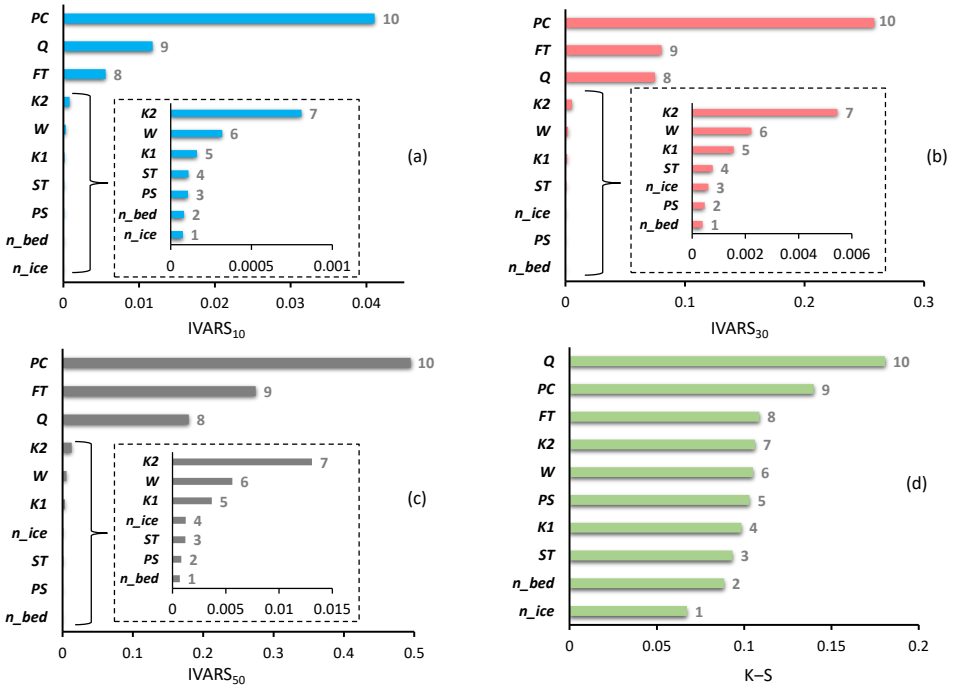


Figure 3-5 Sensitivity indices for the *RIVICE* parameters using the RSA and VARS methods: (a) IVARS10; (b) IVARS30; (c) IVARS50; (d) K-S indices for RMSE measure; the numbers on the bars represent the parameter ranking obtained based on the sensitivity indices.

It is apparent that the more sensitive parameters identified by these two methods are consistent, *i.e.* parameters *Q*, *PC*, and *FT*. Overall, based on these results (**Fig. 3-5**), the

parameters can be categorized into three groups according to their sensitivities: (1) the ice cover characteristics (*PC* and *FT*) and upstream discharge (*Q*) as the most important parameters in the Lower Dauphin River, which is a straight and steep section; (2) the hydraulic roughness parameters (n_{ice} and n_{bed}) and slush ice properties (*PS* and *ST*), with lower impacts on the RIVICE model; and (3) the ice strength parameters (*K2* and *K1*) and downstream water level (*W*) as the medium sensitive parameters. However, the VARS method leads to more distinctive sensitivity indices, as compared to RSA. As can be seen in **Fig. 3-5**, $IVARS_{50}$ and K-S measures provide very conflicting assessments regarding *PS* and n_{ice} sensitivities. The VARS approach identifies *PS* as a low-sensitivity parameter, whereas the RSA determines it to be one of the medium-sensitivity parameters. Also, n_{ice} has the 4th rank based on the $IVARS_{50}$, while it takes first place in the sensitivity ranking (most insensitive parameter) based on the RSA method. This example illustrates how inconsistent sensitivity assessment can arise through use of these philosophically different approaches.

The obtained sensitivity results have the potential for facilitating a deeper understanding of the physical processes affecting rive ice phenomenon. For example, **Fig. 3-5** revealed that *PC* is one of the highly sensitive parameters, which is reasonable because the steeped channel of the Lower Dauphin River can have more compact ice shoving than mildly steep channels. Additionally, considering the RIVICE boundary condition parameters, *W* has a relatively low impact on water level predictions compared to *Q*, due to the large downstream cross section of the Lower Dauphin River at the inlet to Lake Winnipeg (see **Fig. 3-2**).

It is worth mentioning that the sensitivity results can also provide guidance on the design of the flood mitigation scheme along the Lower Dauphin River. For example, having the discharge *Q* as a very sensitive parameter to the ice cover formation and backwater staging justifies the diversion of water from the Dauphin River via the Emergency Outlet Channel (see **Fig. 3-2**). Thus, due to the high impact of the ice cover characteristics and upstream discharge on water level predictions, it is necessary to monitor them at the Lower Dauphin River using effective tools such as observations by personnel or remote sensing (for a general review of different methods for monitoring real-time ice conditions the reader is referred to Vuyovich et al. (2009)). Furthermore, the sensitivity of *K1 (K1TAN)*, the parameter associated with the shedding of the

longitudinal compressive forces within the ice cover laterally to the banks, suggests that armouring of the dike sides facing toward the river could increase the friction between the ice cover and dikes to provide extra internal resistance to ice cover shoving. This would reduce the compaction and ice thickening of the ice cover, leading to an overall reduction in the backwater staging.

3.6.2 Assessment of confidence in GSA results by bootstrapping

Fig. 3-6 depicts the degree of uncertainty in SA results, which was estimated via bootstrapping. To assess the uncertainty in estimated sensitivity indices, the width of the 95% confidence intervals (CI) of the sensitivity metrics distributions obtained by bootstrapping were compared. Considering the width of the CIs across all the model parameters, it can be concluded that the CIs of the VARS products are significantly narrower than the RSA. A wider CI shows higher uncertainty in associated sensitivity assessment. Another feature that can be observed is the lower uncertainties of the results generated by $IVARS_{50}$, which is a representation of the comprehensive assessment of sensitivities.

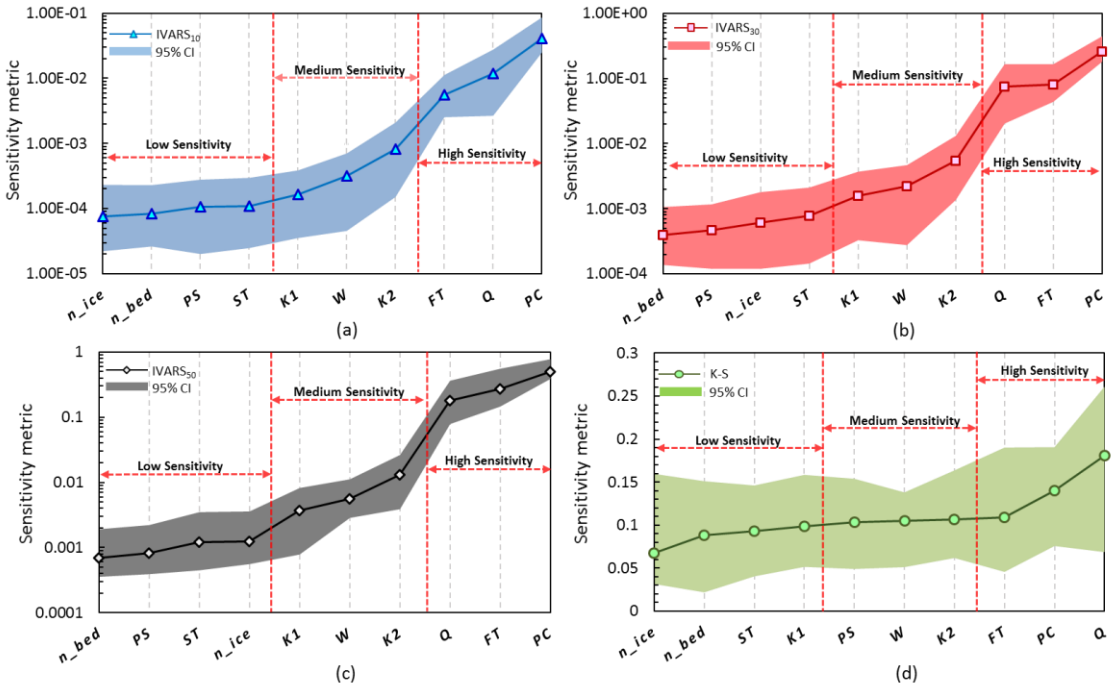


Figure 3-6 Ninety-five-percent confidence intervals (CIs) estimated based on the bootstrapping; subplots: (a) IVARS10; (b) IVARS30; (c) IVARS50 show the VARS-based metrics; subplot (d) shows the 95% CIs of K-S measure for RSA.

A further assessment of the reliability that can be associated with parameter sensitivity rankings is given in **Fig. 3-7**. The success rates were calculated based on the percentage of times among 1,000 bootstrap re-samplings that the sensitivity ranks of the parameters were equal to the original rank obtained by the initial sample. The results in **Fig. 3-7** reveal that $IVARS_{10}$, $IVARS_{30}$, and $IVARS_{50}$ provide higher success rates than those of RSA for all parameters, thereby confirming the reliability of the VARS method in parameter sensitivity rankings.

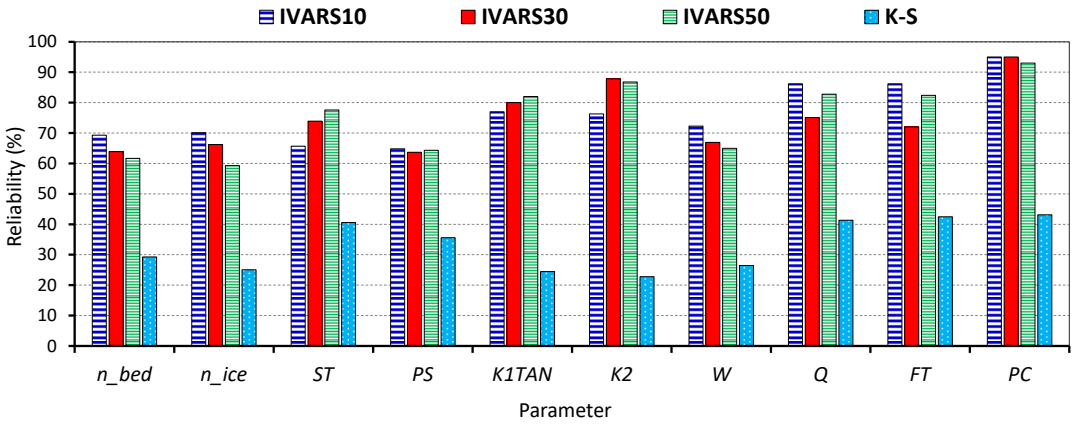


Figure 3-7 Reliability assessment of VARS and RSA for *RIVICE* model parameter ranking based on bootstrapping.

3.7 Conclusions

The significant impacts and complexity of river ice processes have led to the extensive use of numerical models within the water resources research community in cold regions. Parameters of these models can lead to large variation in model outputs. Hence, it is crucial to identify the key parameters that affect river ice model performance. To address this issue, we conducted a comprehensive evaluation of the effectiveness and reliability of two global sensitivity analysis methods, namely Regional Sensitivity Analysis (RSA) and Variogram Analysis of Response

Surfaces (VARS) using the RIVICE model as a test problem. The Lower Dauphin River at Manitoba, Canada, which has a very steep and straight reach, was selected as a case study.

The results show that the water level simulated by the RIVICE model in the Lower Dauphin River is more sensitive to ice cover characteristics, due to the greater compaction and thickening of the ice cover along this steep gradient. In addition, RIVICE simulations are very sensitive to changes in upstream discharge. It may be because the steeper slopes will form denser, more compact frazil ice covers, which are more sensitive to inflowing discharge than those in milder sloping sections. Furthermore, the results revealed that hydraulic roughness parameters and slush ice properties are medium- and low-sensitivity parameters, respectively.

To explore the strengths and limitations of both methods, sensitivity rankings obtained by VARS were compared to those obtained by RSA, with significant differences found regarding porosity of the slush and ice roughness parameter sensitivities. Moreover, the reliability of VARS and RSA methods, as well as the uncertainty in SA results, were assessed using the bootstrapping procedure. The bootstrapping results confirmed that the sensitivity rankings were estimated more reliably using the newly developed VARS framework than the traditional RSA. Consequently, the VARS-based rankings are less ambiguous, as compared with the RSA method.

The results of this study can be used by modelers for a variety of purposes, such as uncertainty quantification and model calibration. By way of example, in RIVICE model calibration, the insensitive parameters can be fixed at constant values to improve model parameter identifiability. Additionally, the insight gained through this study can increase the transparency of the RIVICE model and provide a deeper understanding of how the model simulates river ice processes, which is an aid to robust decision-making.

Author contributions

RS developed the methodology together with SR and KEL. RS wrote the computer codes, designed the experiments, and performed them all. FS was responsible for the verification of the

RSA results. RS prepared the manuscript and other co-authors contributed to editing of the paper at all stages.

Chapter 4

An Automated Factor Grouping Strategy for Robustness and Convergence Assessment of the Global Sensitivity Analysis Algorithms

This chapter is a mirror of the following published article with minor changes to increase its consistency with the body of the dissertation. Changes were only made to avoid repeating the contents that have been presented more appropriately in other parts. References are unified at the end of the dissertation.

Sheikholeslami, R., Razavi, S., Gupta, H.V., Becker, W., and Haghnegahdar, A. 2019. Global sensitivity analysis for high-dimensional problems: how to objectively group factors and measure robustness and convergence while reducing computational cost. *Environmental Modelling and Software*, 111, 282–299. https://doi.org/10.1016/j.envsoft.2018.09.002

Synopsis

Dynamical earth and environmental systems models are typically computationally intensive and highly parameterized with many uncertain parameters. Together, these characteristics severely limit the applicability of Global Sensitivity Analysis (GSA) to high-dimensional models because very large numbers of model runs are typically required to achieve convergence and provide a robust assessment. Paradoxically, only 30 percent of GSA applications in the environmental modelling literature have investigated models with more than 20 parameters, suggesting that GSA is under-utilized on problems for which it should prove most useful. In this chapter, we develop a novel grouping strategy, based on bootstrap-based clustering, that enables efficient application of GSA to high-dimensional models. We also provide a new measure of robustness that assesses GSA stability and convergence. For two models, having 50 and 111 parameters, we show that grouping-enabled GSA provides results that are highly robust to sampling variability, while converging with a much smaller number of model runs.

4.1 Introduction

4.1.1 Motivation

Computational models are widely used to understand and simulate the complex physical behaviors of Complex Environmental Systems Models (CESMs) (Benett et al., 2013; Provenzale 2014; Bathiany et al., 2016). By enabling prediction and scenario analysis regarding the quality and quantity of Earth's future resources (Poff et al., 2015; Guo et al., 2016; Maier et al., 2016), such models have become essential to management and decision making under uncertainty and non-stationary conditions (Castelletti and Soncini-Sessa, 2007). However, the drive to incorporate our ever-growing understanding of underlying system processes and their feedback mechanisms leads to progressively more complex and computationally intensive model formulations. With growth in complexity, and presumably fidelity, it is now not uncommon for CESMs models to contain hundreds, and even thousands, of parameters and factors whose values are uncertain and need to be characterized.

Global sensitivity analysis (GSA) has proven to be an effective means for characterizing the impact and significance of uncertainty in various model components (e.g., parameters, initial conditions, boundary conditions, etc.) on model behavior and predictions (Razavi and Gupta, 2015). GSA techniques are now widely used for a variety of purposes, including uncertainty apportionment (e.g., Marino et al., 2008; Borgonovo et al., 2012), parameter screening (e.g., Trocine and Malone, 2000; Touzani and Busby, 2014), and diagnostic testing (e.g., Saltelli et al., 2006; Gupta et al., 2008; Haghnegahdar et al., 2017a). However, two major interrelated challenges limit their application to advanced CESMs, namely (1) the curse of dimensionality and (2) computational expense. The former refers to the fact that, as the number of uncertain parameters/factors increases, the volume of the problem space increases so rapidly that any attempt to investigate and characterize it in a stable, robust, and statistically sound manner requires an exponentially-increasing sample size. The latter refers to the typically computationally intensive nature of CESMs models, leading to long run times that, together with the former, can make any meaningful analysis of such models computationally prohibitive.

These two challenges are the primary reason why GSA applications reported in the literature are often limited to relatively simple models having smaller numbers of uncertain parameters. However, this is *paradoxical* to the underlying goals of GSA to facilitate the development, understanding, and use of more realistic CESMs models. In this regard, it is interesting to note that a major goal of GSA, commonly stated in the literature, is to help ***reduce*** the dimensionality of a problem by identifying non-influential parameters. However, as the dimensionality of the problem space grows beyond a few tens of parameters, most GSA methods quickly become handicapped by their need for impractically large sample sizes (due to the excessively large computational times involved) and are therefore unable to provide robust assessments of sensitivity that have an adequate level of confidence (Razavi and Gupta, 2016b).

A further class of sensitivity analysis approaches, known as *metamodel-enabled* GSA, can substantially reduce the number of model runs needed to estimate sensitivity indices, but is mostly restricted to low or moderate-dimensionality problems. In fact, as discussed in Razavi et al. (2012a) about 85% of metamodel applications have been on problems having less than 20 factors. In higher dimensions, the performance of metamodel-enabled GSA algorithms can substantially deteriorate due to over-fitting (Becker, 2015). As another potential remedy, analysts sometimes use local sensitivity analysis (LSA) methods that require lower computational demand. However, it is well-known that LSAs provide inadequate assessments that can often be misleading (Saltelli and Annoni, 2010).

In this chapter, we present an approach that addresses the problem of robust sensitivity assessment for high-dimensional problems. There are two main aspects to this approach. The first aspect is based on the "*sparsity of factors*" principle (otherwise known as the Pareto Principle), which states that a small subset of factors is often responsible for most of the system output uncertainty (Box and Meyer, 1986). To exploit this principle, we need a way to identify which of the individual factors have similar properties in terms of their influence on model output variations. In doing so we also recognize that, when the number of factors is very large, the user is typically not interested in an ***exact*** ranking of factor importance; for example, with 100 factors, it may not matter whether a given factor has a sensitivity ranking of 49 or 50. Instead, it may be more profitable to use the available computational budget to reliably

categorize factors into a small number of distinct groups; for example, these could be labelled as "*strongly influential*", "*influential*", "*moderately influential*", "*weakly influential*", and "*non-influential*".

The second aspect deals with an essential (but often neglected) element of GSA, which is that of characterizing and improving the "robustness" of the GSA results. This is extremely important, given that GSA is a sampling-based technique and as such is prone to statistical uncertainty due to sampling variability; i.e., the results will be sensitive to randomness in the selection of the sample. Hence, robustness can be defined as the stability of the GSA results (i.e., the degree of insensitivity to sampling variation). In other words, lower variability of the results obtained over multiple trials of the algorithm (performed with different, identically distributed, sample sets) indicates more robustness (see, e.g., Montgomery (2008)).

Razavi and Gupta (2016b) developed one of the first techniques to assess the robustness of GSA factor rankings (incorrectly termed "reliability" measure in their paper) based on the use of bootstrapping. Here, we extend that approach and integrate it with optimal factor grouping (i.e., the first aspect mentioned above) to provide a GSA solution for high-dimensional problems that are often intractable with traditional approaches.

4.1.2 Objectives and Scope

The primary goal of this chapter is to introduce an automated "*factor grouping*" strategy that is based on the integration of clustering with bootstrapping. The method is designed to group input factors into a certain number of groups based on information gained during GSA where the resulting groups can be of any size. Importantly, if the number of groups is not pre-specified, the algorithm efficiently determines an *optimal* number of groups.

Further, to evaluate performance of the grouping-based GSA, we develop a measure of robustness that provides a way to monitor convergence of the GSA results. The overall procedure can be used in conjunction with any GSA technique. Here, we demonstrate the utility of this approach on two high-dimensional problems, by coupling it with the VARS methodology.

The rest of this chapter is structured as follows. In **Section 4.2**, we discuss the curse-of-dimensionality challenge associated with GSA of high-dimensional problems, review existing strategies for factor grouping, and discuss their limitations. In **Section 4.3**, we introduce the new grouping technique and explain the details of its implementation. The two case studies are developed in **Section 4.4** and the numerical results and analyses are presented in **Section 4.5**. Finally, conclusions and future recommendations appear in **Section 4.6**.

4.2 Review of the Literature

4.2.1 Difficulties associated with GSA of high-dimensional problems

In the vast majority of GSA studies reported in the literature, the dimensionality of the problem space has been quite low. **Fig. 4-1** shows a cumulative distribution function (CDF) of the factor space dimensionality for *Environmental Modelling* studies during the period 2007-2017 that reported applications of GSA. In ~70% of the cases, the number of factors was less than or equal to 20, while ~85% had less than 40 and almost all (97%) had less than 90 factors. Only very few studies used models with more than 100 factors; Radomyski et al., (2016) used 156, Tang et al., (2007) used 403, and Herman et al., (2013) used 1092. So, although one of the main goals of GSA is to assist in reducing the dimensionality of a problem by screening out non-influential parameters, the evidence suggests that it is seldom applied to the high-dimensional problems in which it is most needed.

The reason for this can (in part) be attributed to the curse of dimensionality which, coupled with the high computational costs typically associated with running complex CESMs, makes application of GSA to high-dimensional problems very expensive. The curse of dimensionality manifests through the fact that, to achieve stable and robust results, current GSA techniques require large numbers of sample points (i.e., model runs) to be drawn in some representative manner from the factor space. While this issue can be partially addressed using optimized sampling algorithms such as Progressive Latin Hypercube Sampling (PLHS; Sheikholeslami and

Razavi, 2017) and Good Lattice Points (GLP; Gong et al., 2016), the computational cost remains high (Song et al., 2017).

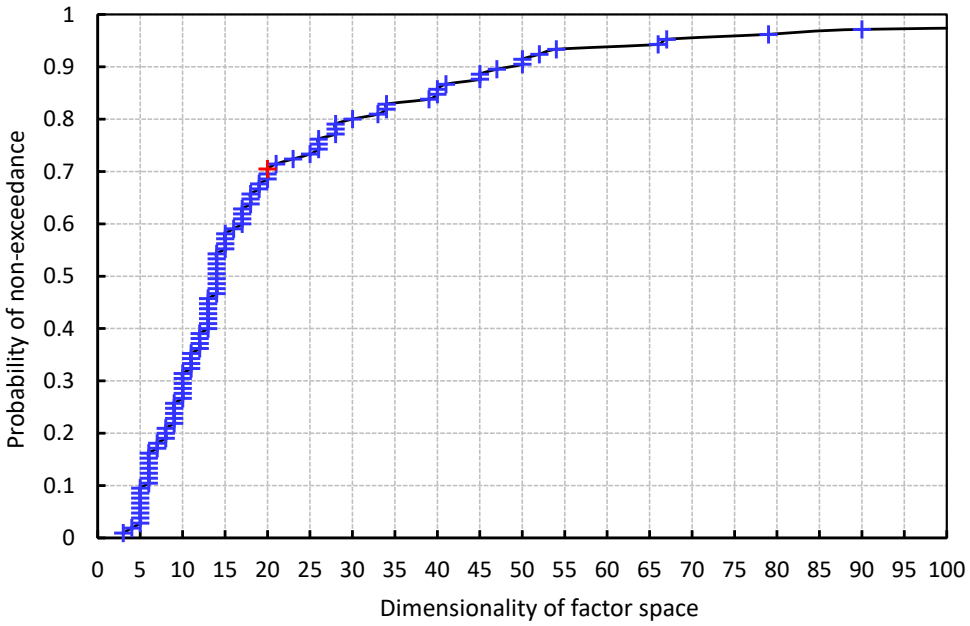


Figure 4-1 Cumulative distribution function for total number of input factors in GSA of environmental models. To make this plot we updated the data provided by Vanrolleghem et al. (2015) and Song et al. (2013). The Thomson Reuters Web of Science was used (August 2017) based on the search terms ‘‘Global Sensitivity Analysis’’ + “Environmental Model” + “Hydrology”

To further synthesize the status quo, **Table 4-1** provides an overview of recent papers that have used GSA to study “higher-dimensional” models; here we have arbitrarily selected only studies where the number of factors was greater than or equal to 40. It is important to note that the sample size in these studies was typically chosen based on available computational budget, without considerations of GSA stability and convergence. Consequently, the results are likely to have been highly sensitive to sampling variability (i.e., may not have been robust). Studies of the convergence behavior of various GSA techniques generally show that factor ***ranking*** (order of relative sensitivity) typically converges more quickly than factor ***sensitivity*** indices computed by the GSA (perhaps as expected), and that factor ranking is more robust to sampling variability (Benedetti et al., 2011; Yang, 2011; Cosenza et al., 2013; Wang et al., 2013; Vanrolleghem et al.,

2015; Razavi and Gupta, 2016b; Sarrazin et al., 2016). Overall, this fact indicates that if a modeler is interested in factor *screening or prioritization*, rather than in generating accurate estimates for the sensitivity indices, the number of required model simulations (and hence computational cost) can be reduced. Interestingly, Vanrolleghem et al. (2015) and Wang et al. (2013) found that, for the GSA methods they used, sensitivity index convergence rates were typically slowest for the factors having the highest importance while, conversely, Sarrazin et al. (2016), Nossent et al. (2011), and Yang (2011) reported that the sensitivity index convergence rates were slowest for factors having lowest importance. This lack of agreement in their results suggests that convergence rates may be case-specific and depend on other aspects of the problem at hand including, but not limited to, sampling variability and the choice of GSA algorithm.

From this review of the literature (**Table 4-1**), we make three observations:

- Regardless of GSA method used, it is typical for only a small sub-group of factors (on average ~20%) to exert significant control over variations in the model outputs (a manifestation of the "sparsity of factors" principle).
- Historically, the grouping of factors into "*strongly influential*" versus "*non-influential*" has typically been done in a subjective and case-specific manner, the most common approach being to specify a threshold sensitivity value and to group factors based on sensitivity relative to the threshold value.
- When different GSA methods were applied to the same problem, while actual rankings varied, the factor rankings were typically such that, the relative positions of factors in the higher, middle or lower parts of the rankings were often quite similar (e.g., Cosenza et al., 2013; Vanuytrecht et al., 2014; Sheikholeslami et al., 2017). In other words, different methods generally tended to identify the same "*groups*" of factors in terms of the strength of their influence on model response. From a practical point of view, this suggests that one might focus on whether a factor belongs to a high, medium, or low-influence group rather than on its exact ranking, particularly given the demonstrated effects of sampling variability on estimated rankings.

Table 4-1 Recent studies that applied GSA to environmental models with 40 or more uncertain parameters

Model	*Description*	*Dimensionality of factors space*	*# of highly influential factors* [a]	*#of model runs*	*Stability and convergence considerations*	*GSA method*	*Reference*
ASM2d-SMP-P	Wastewater treatment systems	79	10 16	800 39500	No	Morris EFAST	Cosenza et al. (2013)
CLM3.5	Land surface model	66	10	1000	No	Eigen decomposition	Göhler et al. (2013)
CoupModel v5	Peatland carbon dioxide model	54	10	3200	No	RSA	Metzger et al. (2016)
EFDC	Water quality model	54	9	3300[b]	Yes	Morris	Yi et al. (2016)
MESH	Hydrologic model	50	6 or 8[c]	22550	No	VARS	Yassin et al. (2017)
SWAT	Hydrologic model	50	3	102000 30000 520000[b]	Yes	Morris RSA Sobol	Sarrazin et al. (2016)
WOFOST	Crop growth model	47	6	16385[b]	Yes	EFAST	Wang et al. (2013)
MESH	Hydrologic model	45	NA		Yes	VARS	Razavi and Gupta (2016b)
AquaCrop	Crop simulation model	43 32	33 19	24500	Yes	Morris EFAST	Vanuytrecht et al. (2014)
SiB3	Land surface model	42	27-31	45,000	No	Sobol	Rosolem et al., (2012)
LU4-R-N	Water quality model	41	4	100000	No	RSA	Medici et al. (2012)
WOFOST & Noah LSM	Crop growth and soil moisture model	40	4 or 7[d]	N/A	No	Morris	Eweys et al. (2017)
CoLM	Land surface model	40	2-8	410 2,000	Yes No	Morris MARS+ Sobol	Li et al. (2013)

[a] Numbers in this column is based on the reported results in the cited papers.
[b] Maximum number of function evaluations
[c] Depending on the objective function
[d] Depending on the case study

In high-dimensional problems, grouping input factors of similar sensitivity can help reduce the effective dimensionality of the factor space. This is useful because when the problem has more than ~20 factors, it can be difficult to analyze the GSA results for extraction of relevant information. It seems beneficial, therefore, to categorize factors into subsets to facilitate interpretation. Together, these issues highlight the need for an effective, automated, and non-subjective strategy for grouping model factors based on information developed during the execution of a GSA. In the next subsection, we review the few strategies for grouping that have been proposed in the literature, discuss their characteristics, and identify shortcomings.

4.2.2 Review of existing grouping strategies for GSA

Despite significant research dedicated to the advancement of GSA, there is a surprising paucity of literature on the development of efficient grouping strategies. Relevant studies have focused mainly on enabling GSA methods to compute an overall sensitivity measure for a pre-identified group of factors. For example, in the variance-based Sobol' method (Sobol', 1993), the set of uncertain factors $\boldsymbol{X}$ can be partitioned into pre-specified groups $\boldsymbol{X}_u$ and $\boldsymbol{X}_w$, where $\boldsymbol{X}_u \cup \boldsymbol{X}_w = \boldsymbol{X}$ and $\boldsymbol{X}_u \cap \boldsymbol{X}_w = \emptyset$, so that the variance of the response $Var(Y)$ can be decomposed as $V_{\boldsymbol{X}_u} + V_{\boldsymbol{X}_w} + V_{\boldsymbol{X}_u,\boldsymbol{X}_w} = 1$, where $V_{\boldsymbol{X}_w} = Var_{\boldsymbol{X}_w}\left(E_{\boldsymbol{X}_u}(Y|\boldsymbol{X}_w)\right)/Var(Y)$ is the normalized variance of the conditional expectation that measures the first order effect of $\boldsymbol{X}_w$ on the model output. This decomposition can be performed for any number of factors (Saltelli et al., 2006). Extending this, Saltelli et al. (2008) provided a multi-stage factor grouping strategy to perform variance-based sensitivity analysis based on the concept presented above. In their approach, the number of groups is specified a priori and then the members of each group are populated by testing a variety of combinations.

Similarly, Campolongo et al. (2007) modified the Morris' Elementary Effects method (Morris, 1991) to work with pre-specified groups of factors for which total sensitivity indices can be estimated (Ciuffo and Azevedo, 2014; Patelli et al., 2010). In all the grouping techniques reviewed here, the factor grouping scheme is determined before application of GSA, based on the nature of the problem, physical interpretation, previous experience, intuition, randomized grouping, etc.

Another commonly-used approach is to group factors based on arbitrary, pre-specified sensitivity index thresholds (Eweys et al., 2017; Hou et al., 2015; Göhler et al. 2013; Tang et al., 2007). For example, in variance-based GSA, a group of factors can be deemed "*strongly influential*" if each member individually contributes at least 10% of the overall model output variance, or "*weakly/non-influential*" if each member individually contributes less than 1% of the model variance (note that these thresholds are selected rather arbitrarily). However, in the Morris' method of Elementary Effects, the definition of thresholds is necessarily case-specific because the estimated sensitivity indices can have different ranges of variation, depending on the model output and particularities of the case study.

For screening purposes (identifying non-influential factors), Khorashadi Zadeh et al. (2017) proposed the interesting approach of adding a "dummy" variable (i.e., a variable known a priori to be fully non-influential) to the set of factors under investigation. Factor screening has also been accommodated by applying a statistical testing approach originally introduced by Andres (1997) and further extended by Sarrazin et al. (2016), Tang et al. (2007), and Nossent et al. (2011). As a more objective way to achieve factor grouping, Klepper (1997) used a clustering method to categorize model factors based on the sensitivity analysis results. To our knowledge, this method is the only method to date that is based on the analysis on a clustering concept. It does not, however, provide an "optimal" grouping and is prone to uncertainty associated with sampling variability.

A major disadvantage with a priori specification of the grouping is that the results can depend strongly on the user's selection of groups, a task that can become complicated (if not impractical) as the number of factors grows. Overall, our review leads us to conclude that an effective grouping strategy must address the following questions as part of the GSA methodology itself:

1) What is an optimal grouping of model factors into a given number of groups?

2) What is the optimal number of groups?

3) To what extent are the grouping results robust?

Regarding the first question, the members of a factor group should be as similar (in some sense) as possible while being as distinct as possible from the members of the other groups. To

achieve this, a metric must be defined that quantifies similarity between input factors according to the way they influence the model outputs. Regarding the second question, removal of subjectivity about the number of groups should be somehow based on the goal of obtaining a maximum level of homogeneity within groups and distinction across groups. Finally, the third question addresses the need for some degree of confidence in the grouping results, meaning that the results are not overly sensitive to sampling variability. None of the aforementioned grouping strategies addresses these challenges explicitly and in a systematic manner. This gap motivated our development of the automated factor grouping method introduced in the next section.

4.3 The Proposed Factor Grouping Strategy

Our proposed method for factor grouping (**Fig. 4-2**) combines agglomerative hierarchical clustering with bootstrapping and introduces a new robustness measure that enables an objective assessment of GSA convergence. The algorithm consists of five steps; details of each are provided in the following subsections.

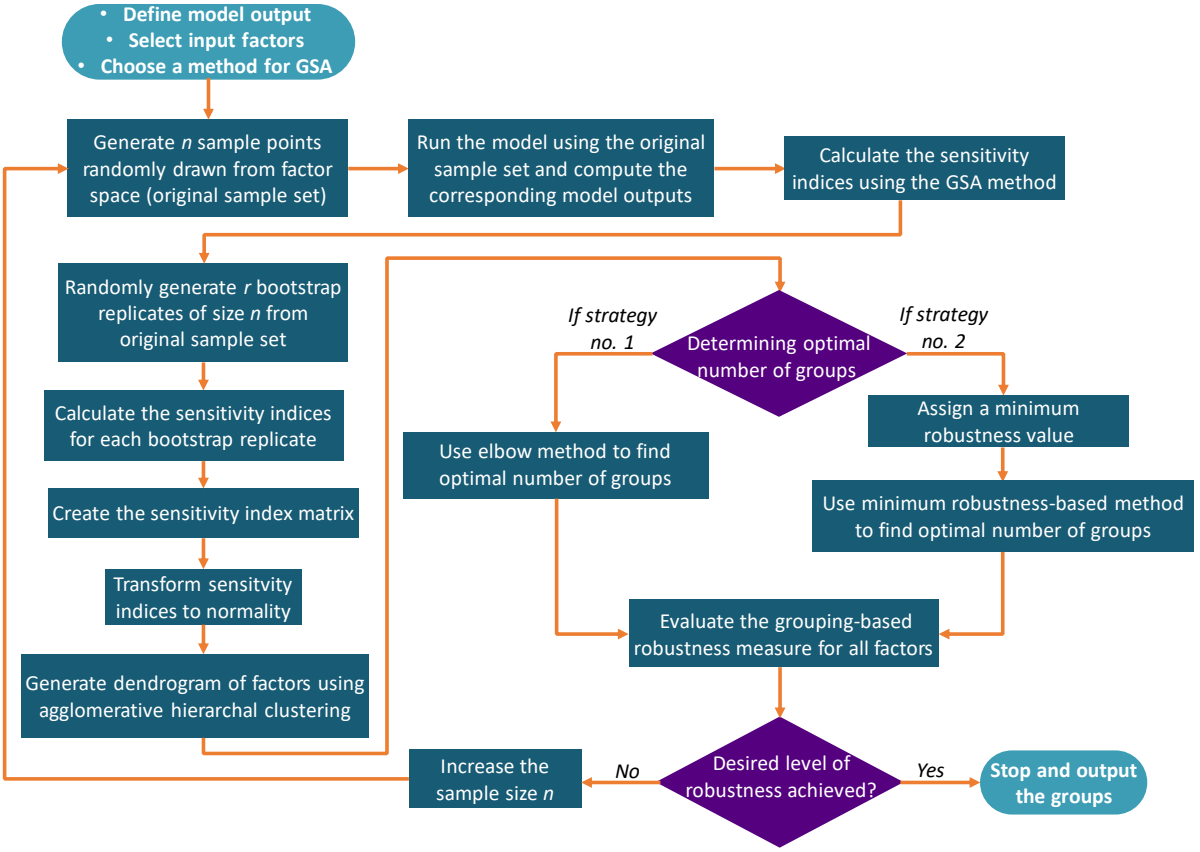


Figure 4-2 Flowchart of the factor grouping algorithm developed in this study

4.3.1 Generating the sensitivity matrix by bootstrapping

In the traditional approach to GSA, the model is run n times to generate the (original) sample of n points drawn from factor space, and then a vector of sensitivity indices for the d factors is computed using the information provided by these points:

$$\boldsymbol{S}_{1,d} = \{s_1, \ldots, s_d\}$$

In general, the size n of the randomly drawn sample used to estimate $\boldsymbol{S}_{1,d}$ will be limited and so, due to sampling variability, we can expect a degree (often considerable) of statistical uncertainty to be associated with these estimates. The distribution-free (and easy to implement) bootstrap method (Efron, 1979) can be used to estimate the magnitude of that uncertainty (Davison et al., 2003); example applications in the GSA context can be found in Razavi and Gupta (2016b) and Archer et al. (1997). Importantly, in bootstrapping, the set of bootstrap

samples is extracted from the original sample set, and thus additional model runs are not necessary. We apply the following two steps to implement the bootstrap technique:

1. Select the number of bootstrap re-samplings to be r and randomly draw (with replacement from original sample set) r bootstrapped samples, each having the same size n as the original sample. The result is the so-called set of r bootstrap samples. Note that r is selected to be some arbitrarily large number (e.g., 1,000) as has been commonly used in the literature).
2. For each bootstrap sample $i(i = 1, \dots, r)$ in the set, re-compute the vector of sensitivity indices $\boldsymbol{S}_{i,d}^{B} = \{s_{i,1}^{B}, \quad \dots, \quad s_{i,d}^{B}\}$ using the GSA method.

Following the above procedure, a two-dimensional $r \times d$ bootstrapped sensitivity matrix $\boldsymbol{M}$ containing r bootstrap replicates for the sensitivity indices is formed:

$$\boldsymbol{M} = \begin{pmatrix} s_{1,1}^{B} & \cdots & s_{1,d}^{B} \\ \vdots & \ddots & \vdots \\ s_{r,1}^{B} & \cdots & s_{1,d}^{B} \end{pmatrix} \tag{1}$$

Because the matrix $\boldsymbol{M}$ contains information regarding estimation uncertainty, it provides valuable information that enables a robust evaluation of how each factor impacts the model outputs. Note that the values that make up the matrix $\boldsymbol{M}$ should be "normalized" (transformed) as discussed in the next section, before proceeding with the analysis.

4.3.2 Transforming the distribution of sensitivity indices to be un-skewed

As shown in **Fig. 4-3(a)**, when analyzing many factors, the distribution of factor sensitivity indices obtained during GSA will usually be heavily skewed to the right. This is because typically only a small subset of the parameters of a high-dimensional model exerts a strong influence on the model response. To reduce bias, and to improve discrimination of factor importance across the full range of sensitivities, it is helpful to transform the sensitivity indices to be more "normalized" as shown in **Fig. 4-3(b)**. This transformation makes it possible to better distinguish factors on the left end of the sensitivity axis so that, for example, groups that are "*moderately influential*", "*weakly influential*", and "*non-influential*" can be easily identified as distinct categories. Without such transformation, if the level of skewness is high, there will be a

tendency for the factor grouping to become highly biased, with a large group at the left end (containing non-influential to moderately influential factors) and smaller groups towards the right (containing strongly influential factors), leading to conclusions that may not be sufficiently granular to be informative.

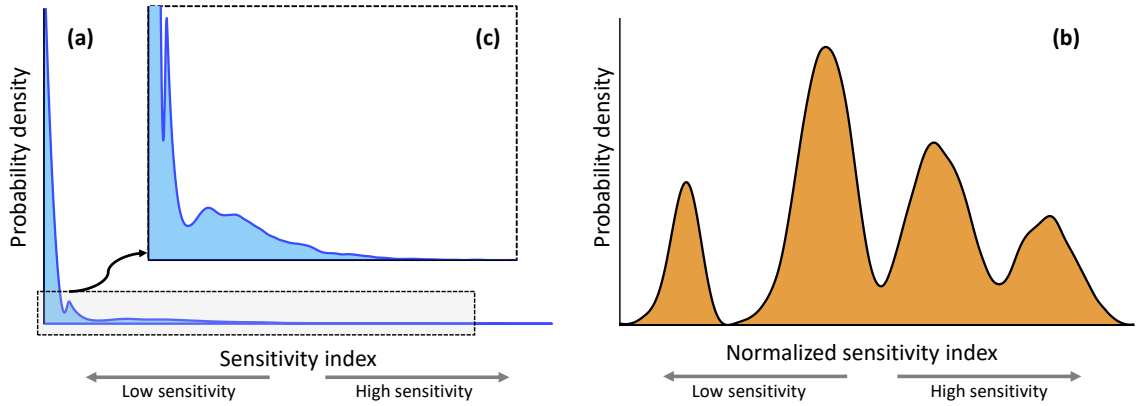


Figure 4-3 Conceptual distributions of (a) original sensitivity indices without normalization and (b) normalized sensitivity indices. Subplot (c) shows a zoom-in of the subplot (a) for small values on the vertical axis. When transforming the data in subplot (a), the differences between smaller sensitivity indices (moderately influential factors) should be expanded, whereas the differences between larger sensitivity indices (strongly influential factors) should be reduced.

A further reason for normalization is that our proposed grouping scheme divides factors into subsets using a similarity metric based on the Euclidean distance. Consequently, if the distribution is highly skewed, the Euclidean distance (and resulting grouping) will be strongly affected by large magnitudes (e.g., outliers) associated with a very small number of sensitivity indices. Normalization helps to remove any such bias, by ensuring that the distance metric assigns appropriate importance to each variable.

Logarithmic, square root, and arcsine transformations are among the more frequently-used methods to normalize data. Here we used the well-known Box-Cox transformation (Box and Cox, 1964) that provides a particularly flexible approach encompassing many of the aforementioned transformations (logarithm, square root, reciprocal square root, and many others). Given that the sensitivity indices have a skewed distribution, a Box-Cox transformation is applied, parametrized by a non-negative value λ:

$$ns_{i,j}^{B}(\lambda) = \begin{cases} \dfrac{\left(s_{i,j}^{B}\right)^{\lambda} - 1}{\lambda} & (\lambda \neq 0) \\ \log\left(s_{i,j}^{B}\right) & (\lambda = 0) \end{cases} \quad (2)$$

By selection of an appropriate value for λ, matrix $\boldsymbol{M}$ is transformed into matrix $\boldsymbol{nM}$ consisting of normalized sensitivity indices $ns_{i,j}^{B}$, where $i = 1, \ldots, r$ and $j = 1, \ldots, d$. The distribution of the elements of $\boldsymbol{nM}$ is approximately symmetrical (i.e., having skewness close to zero). We used Nelder-Mead simplex direct search optimization (Lagarias et al., 1998) to find an optimal value for λ that maximizes the following log-likelihood function (Sakia, 1992; Box and Cox, 1964):

$$LLF(\lambda) = -\frac{n}{2}\log\left(\hat{\sigma}_{\lambda}{}^{2}\right) + (\lambda - 1)\sum_{i=1}^{r}\sum_{j=1}^{d}\log\left(ns_{i,j}^{B}\right) \quad (3)$$

where $\hat{\sigma}_{\lambda}$ is the standard deviation of the normalized sensitivity indices for a given λ. Note that the matrix $\boldsymbol{nM}$ of normalized sensitivity indices will be used in the proposed grouping algorithm.

4.3.3 Factor grouping using agglomerative hierarchal clustering

Having constructed the matrix $\boldsymbol{nM}$ of normalized sensitivity indices, factor grouping can proceed with the aid of cluster analysis to identify groups of sensitivity indices whose values are as similar as possible, while being as distinct as possible from other groups. Many clustering algorithms are available, including hierarchical clustering, *k*-means, and density-based clustering (see Tan et al. (2006) and Hair et al. (2006) for more details).

Here we use the popular agglomerative hierarchical clustering method (AHC; Johnson, 1967), a bottom-up approach that clusters data based on iterative merging of the two closest groups. In our implementation, AHC starts with *d* groups by assigning each column of $\boldsymbol{nM}$ (each factor) to a group having a single-element (called a leaf). At each successive step, the two groups that are deemed most similar are joined (or agglomerated) into a new, larger group (called a node). Iterative application of this procedure is continued until all the columns of $\boldsymbol{nM}$ (all factors) are contained in one single large group (called the root). The result is a clustering tree, commonly referred to as a "dendrogram". Once the dendrogram has been constructed, the user can examine the resulting cluster hierarchy, and cut the dendrogram at any level (cutoff threshold) to

determine the groups. A major advantage of AHC over other clustering techniques, such as the *k*-means algorithm, is that it does not require a pre-specification of the final (i.e., desired) number of groups.

A critical step in implementation of AHC is selection of a metric to quantify the pairwise similarities between factors. Most available methods are based on Euclidean distance, such as single linkage (Sneath, 1957), complete linkage (Sørensen, 1948), weighted average linkage (Sokal and Michener, 1958; Lance and William, 1967), centroid (Lance and William, 1967), Ward's method (Ward, 1963), etc. Here we used Ward's method, because it attempts to both maximize between-cluster distances and minimize within-cluster distances using a single objective function called "merging cost". The literature suggests that Ward's method typically outperforms other distance metrics and is among the most commonly-used techniques (see e.g., Hands and Everitt, 1987; Milligan and Cooper, 1988; Ferreira and Hitchcock, 2009; Terada, 2013).

4.3.4 A measure of robustness and convergence of GSA

Given an ability to monitor convergence rates of a GSA experiment, the user can improve efficiency by avoiding unnecessary model runs. Monitoring can be performed through subjective visual inspection of the results (e.g., Vanrolleghem et al. (2015)) or by objective quantitative criteria (e.g., Sarrazin et al., 2016; Yang, 2011, Razavi and Gupta, 2016b). Bear in mind that convergence rates can differ from one GSA algorithm and experiment to another.

Razavi and Gupta (2016b) developed the first robustness measure for factor ranking (incorrectly termed a "reliability" measure therein) based on a bootstrap method, by following these steps:

1. Based on the vector of sensitivity indices $\boldsymbol{S}_{1,d} = \{s_1, \dots, s_d\}$ obtained by application of GSA to the original sample set, compute the vector of the original factor rankings $\boldsymbol{FR}_{1,d} = \{fr_1, \dots, fr_d\}$.
2. Using the set of bootstrap-based vectors of sensitivity indices $\boldsymbol{S}_{i,d}^{B} = \left\{s_{i,1}^{B}, \quad \dots, \quad s_{i,d}^{B}\right\}$ $(i = 1, \dots, r)$, construct the matrix of factor rankings:

$$\boldsymbol{FR}^B = \begin{pmatrix} fr_{1,1}^B & \dots & fr_{1,d}^B \\ \vdots & \ddots & \vdots \\ fr_{r,1}^B & \dots & fr_{r,d}^B \end{pmatrix}$$

3. For each column j of $\boldsymbol{FR}^B$ $(j = 1, \dots, d)$, count the number of times n_j that $fr_{i,j}^B = fr_j$ (for $i = 1, \dots, r$).
4. The robustness measure for the j-th factor is estimated by computing n_j/r. This measure can be any positive number smaller than or equal to 1, where a value equal to 1 indicates that the obtained factor ranking is fully (i.e., 100%) robust to sampling variability.

Here, we extend the above robustness measure to accommodate factor grouping. Our proposed measure quantifies the robustness of the factor rankings based on its membership within a factor group. After performing a grouping operation on matrix $\boldsymbol{nM}$, model factors can be organized into k groups $g_1, \dots, g_k$. Then, we implement the following procedure to calculate the factor grouping-based robustness measure for the j-th factor:

1. Identify the group g_m that contains the j-th factor.
2. For all factors in g_m, based on the original sample set, compute the vector of the original factor rankings $\boldsymbol{FR}_{1,c_m}^{g_m} = \{fr_1^{g_m}, \dots, fr_{c_m}^{g_m}\}$, where c_m is the number of factors in g_m.
3. For the j-th column in $\boldsymbol{FR}^{\boldsymbol{B}}$, count the number of times q_j that $fr_{i,j}^B$ (for $i = 1,2, \dots r$) is equal to either $fr_1^{g_m}, fr_2^{g_m}, \dots,$ or $fr_{c_m}^{g_m}$.
4. Compute the grouping-based robustness measure for the j-th factor by q_j/r.

For example, assume that four factors of a d-dimensional model are clustered into the second group as e.g., $g_2 = \{x_3, x_1, x_8, x_5\}$ and that the corresponding ranks for these factors obtained from the original sample set are $\boldsymbol{FR}_{1,4}^{g_2} = \{3, 4, 5, 6\}$. To evaluate the robustness of factor ranking for each factor in g_2, say x_1, we count the number of times, q_1, out of r bootstrap replicates that the ranking of x_1 is either 3, 4, 5, or 6. Thus, the probability that the rank of x_1 belongs to $\boldsymbol{FR}_{1,4}^{g_2}$ is q_1/r which provides an estimate of the grouped factor ranking robustness. By monitoring the convergence behavior of this measure, we can terminate the algorithm at any desired level of robustness for that factor. Given that in high-dimensional models, it is the general position of the

factors in the higher, middle or lower parts of the ranking that is of actual interest, as opposed to the precise ranking, the approach developed here is of more practical relevance.

4.3.5 Determining an optimal number of groups

In this study, we used two efficient strategies to determine the optimal number of groups. The first, known as the elbow method, finds the number of groups by analysis of the cluster hierarchy of the dendrogram, while the second chooses the number of groups by assessing the robustness measure

4.3.5.1 An elbow method for finding optimal number of groups

To determine the optimal number of groups, one can simply examine changes in the merging cost of combining groups across all successive merging steps and select the point at which the merging cost approaches a plateau and thereafter decreases gradually. This approach, known as the elbow method (Kodinariya and Makwana, 2013), is because the clustering procedure typically reaches a point (elbow point) after which it is no longer worth further grouping the factors. In other words, the merging cost corresponding to this point is "good enough", and the performance improvement achieved by clustering levels off as the number of groups grows further.

As shown in **Fig. 4-4**(**a**), the elbow method can be objectively implemented by plotting the merging cost versus the number of groups and finding the elbow point of this curve, which is mathematically the point of maximum curvature. We apply a widely-used heuristic for identifying this point based on the concept of Menger curvature (Satopää et al., 2011), which defines the curvature of a triple of points as the curvature of the circle circumscribed about those points. Thus, the elbow point can be simply found by drawing a line from start to end of the curve, and then calculating the perpendicular distance from each point to the curve. The point that is farthest away from that line (maximum perpendicular distance) is the elbow point corresponding to the optimal cluster number. In the dendrogram, this point is analogous to the cutoff threshold (as shown in **Fig. 4-4**(**b**)).

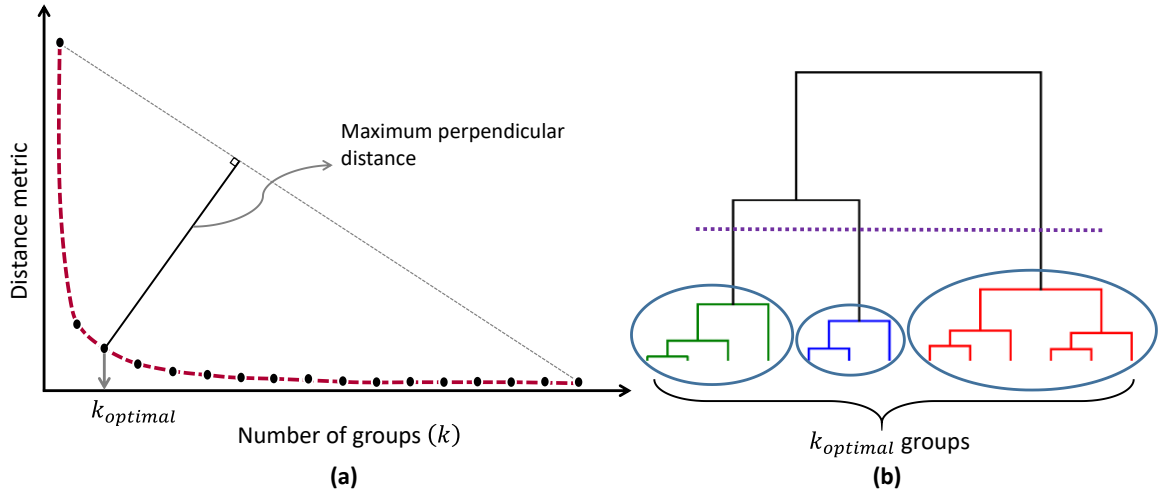


Figure 4-4 Subplot (a) shows a typical plot of a distance metric (*y*-axis) for a cluster analysis versus the number of groups (*x*-axis). As can be seen, the distance metric decreases monotonically by increasing the number of clusters $\boldsymbol{k}$, but from some $\boldsymbol{k_{optimal}}$ onwards it flattens significantly. Subplot (b) shows the corresponding clustering tree or dendrogram constructed by AHC. The height of each node in (b) represents the distance value between the right and left sub-branch clusters. The dashed line in (b) is the cutoff threshold for cutting the dendrogram into $\boldsymbol{k_{optimal}}$ groups.

4.3.5.2 Identifying optimal number of groups based on robustness assessment

After generating a dendrogram, one can cut it at different cutoff thresholds and evaluate the respective factor grouping-based robustness values. Given a user-chosen minimum acceptable robustness value, the optimal number of groups can be selected such that the estimates of robustness values are equal to or greater than the minimum value. In other words, at each iteration, once the dendrogram has been created, we cut the dendrogram at different levels to determine the maximum number of groups that guarantees the estimated grouping-based robustness of all factors to be higher than the pre-specified minimum value. An important attribute of this method (hereafter called "minimum robustness method") is that it provides the user with flexibility in selecting the number of groups based on the obtained robustness values. However, unlike the elbow method, the subjectivity involved in finding the optimal number of groups is not completely removed, because this method requires the user to specify a minimum acceptable robustness value. In general, choosing a higher value for this minimum acceptable

robustness value will result in a smaller number of groups, so the user might find it useful to try several values (e.g., in range 0.50 to 0.95) and pick the one that best suits the objectives of the problem at hand.

4.3.6 The GSA method

The grouping strategy developed here is "GSA method-free" and can be implemented in the context of any GSA algorithm, including the variance-based (e.g., FAST (Cukier et al., 1973) and eFAST (Saltelli et al., 1999)) or density-based (e.g., δ-density (Borgonovo, 2007) and PAWN (Pianosi and Wagener 2015)) methods. In the next section, we illustrate its use with the VARS approach introduced by Razavi and Gupta (2016a), which has been applied to several real-world problems of varying dimensionality and complexity (Sheikholeslami et al., 2017; Yassin et al., 2017; Haghnegahdar and Razavi, 2017; Razavi and Gupta 2016b). Note that VARS generates a set of sensitivity indices called IVARS (Integrated Variogram Across a Range of Scales), that evaluate the rates of variability in model outputs at a range of different perturbation scales. The precise implementation of VARS used here is the STAR-VARS method developed by Razavi and Gupta (2016b). Here, we use progressive Latin hypercube sampling (PLHS; Sheikholeslami and Razavi, 2017) to sequentially locate star centers and STAR sampling to sample star points in the parameter space. It has been shown that PLHS outperforms other traditional sampling strategies in terms of a variety of evaluation criteria (e.g., space-filling and one-dimensional projection properties).

4.4 Numerical Experiments

In the first case study, we demonstrate application of the grouping strategy to a theoretical benchmark function. In the second case study, we apply the method to a real-world modelling problem having more than 100 uncertain parameters.

4.4.1 Illustration using the Sobol g-function

A commonly used benchmark problem for GSA known as the Sobol g-function (Saltelli and Sobol, 1995) has the following mathematical form:

$$Y(x_1, x_2, \ldots, x_d) = \prod_{j=1}^{d} \frac{|4x_j - 2| + a_j}{1 + a_j} \quad (3)$$

where factors x_j $(j = 1,2, \ldots, d)$ are uniformly distributed over a $[0,1]^d$ hypercube, and the a_j are non-negative constants.

The non-linearity and non-monotonicity of this function along with the availability of analytical sensitivity indices make it a suitable problem for the study of GSA techniques. Moreover, since the Sobol g-function is the product of contributions from each input factor, the function is non-additive and features interactions of all orders (Archer et al., 1997). The strength of contribution of each factor x_j to the variability of the response Y can be controlled by changing the values of the a_j terms. When a_j is smaller, the factor x_j becomes more influential, and thus factors can be classified in terms of their importance by assigning appropriate values to these coefficients. This makes it particularly useful for evaluating the performance of any factor grouping scheme. For illustrative purposes, the number of factors d was set to 50, and the coefficients were chosen (as listed in **Table 4-2**) to form four groups of factors with differing relative sensitivities.

To implement STAR-VARS, the number of star centers was arbitrarily set to 200 and the resolution was set to 0.05, resulting in a total of 190,200 evaluations of the Sobol g-function. The total number of bootstrap replicates was set to r = 1,000. For the sensitivity index, we used IVARS-50, also referred to as "total-variogram effect", as it encompasses sensitivity analysis information across a comprehensive range of perturbation scales.

Table 4-2 The coefficients used in this study for 50-dimensional Sobol g-function

Group	***Coefficient***
Strongly influential	$a_1 = a_2 = 0$ $\{a_3, a_4 \dots, a_9\} = \{0.005, 0.020, 0.040, 0.060, 0.08, 0.090, 1\}$
Moderately influential	$a_{10} = a_{11} = \cdots = a_{16} = 2;$ $\{a_{17}, a_{18} \dots, a_{24}\} = \{2.10, 2.25, 2.75, 3, 3.10, 3.15, 3.25, 3.50\}$
Weakly influential	$a_{25} = a_{26} = \cdots = a_{30} = 9;$ $\{a_{31}, a_{32} \dots, a_{44}\}$ $= \{8, 8.5, 9, 10, 10.5, 11, 12, 12.5, 13, 13.5, 14, 14.5, 15, 16\}$
Non-influential	$\{a_{45}, a_{46} \dots, a_{50}\} = \{70, 75, 80, 85, 90, 99\}$

4.4.2 Modelling case study

4.4.2.1 Model description

This case study was adopted from Haghnegahdar et al. (2017b) where the highly-parameterized MESH (Modélisation Environmentale–Surface et Hydrologie; Pietroniro et al., 2007) model was calibrated to the Nottawasaga river basin in Southern Ontario, Canada (**Fig. 4-5**). MESH is a semi-distributed coupled land surface-hydrology modelling system developed by Environment and Climate Change Canada (ECCC) for large-scale watershed modelling with consideration of lateral and cold region processes in Canada.

Figure 4-5 Location of the Nottawasaga river basin in Canada

MESH treats a watershed as being discretized into grid cells and accounts for within pixel heterogeneity using the concept of Grouped Response Units (GRUs, Kouwen et al., 1993). The dominant land cover in the area is cropland followed by deciduous forest and grassland. The dominant soil type in the area is sand followed by silt and clay loam. MESH version 1.3.006 was implemented in this study. More details are provided in **Appendix**.

4.4.2.2 Experimental setup

In the sensitivity analysis study reported here, a total of 111 model parameters were considered. Most of these parameters are related to land cover and soil classes tied to the GRU types. Out of the existing 16 GRU types, only parameters corresponding to the top five GRU types covering areas greater than 5% were included. The rest of parameters are associated with the interflow process, initial conditions, ponding, and channel routing. Parameter ranges were specified using a combination of expert knowledge, previous studies (Dornes et al., 2008), and recommendations in the manual (Verseghy, 2012).

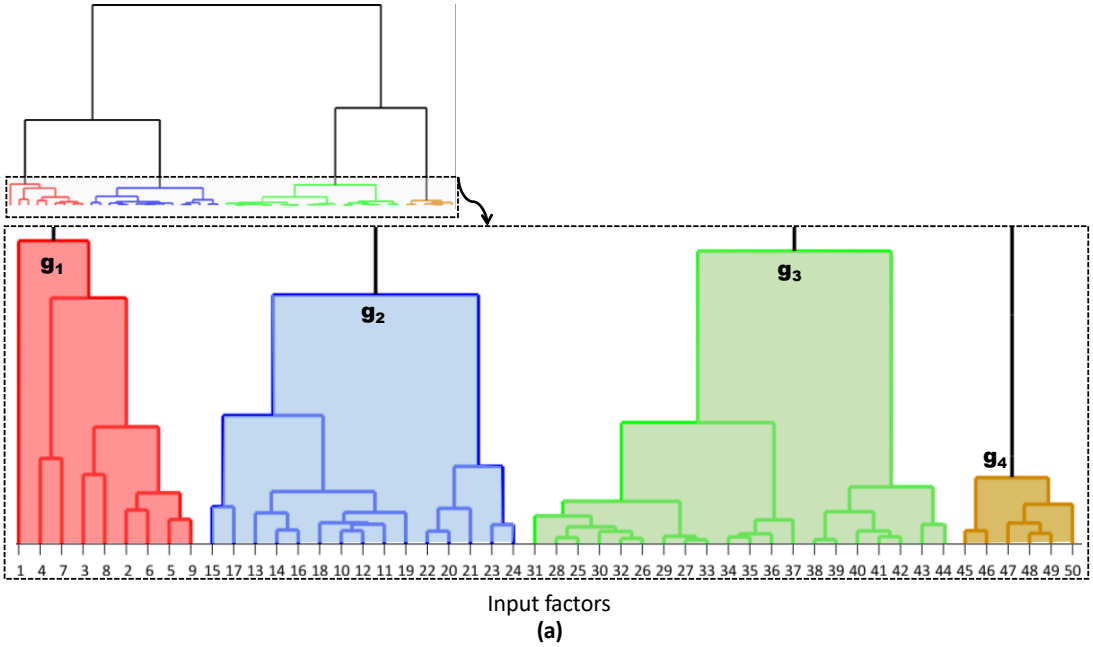
Taking the available computational budget into consideration, 100 star centers were generated with a resolution of 0.1 (chosen arbitrarily) for implementation of STAR-VARS, requiring a total of 100,000 model runs. This large number of MESH runs was performed using the University of Saskatchewan's Linux-based high-performance computing cluster called Plato. Approximately 3 minutes were required to complete a single evaluation of the model on one core of Plato. If a single CPU core is used, the entire set of 100,000 samples will take approximately 6.85 months of computational time to generate (we used 160 CPU cores). The Nash-Sutcliffe coefficient of efficiency (NS) was used to measure daily model streamflow performance, calculated for a period of three years (October 2003 to September 2007) following a "one year" model warmup period that was excluded in the NS calculations. Sensitivity was assessed using the IVARS-50 index, called "total-variogram effect", of the VARS methodology. The total number of bootstrap replicates was set to r = 1,000.

4.5 Results and Discussion

4.5.1 Factor grouping results

4.5.1.1 Results for the Sobol g-function

Fig. 4-6(**a**) shows the factor grouping dendrogram for the first case study (Sobol g-function), obtained by the normalized IVARS-50 sensitivity indices. Applied to **Fig. 4-6**(**a**), the elbow method automatically identified the four groups of factors labeled as $\{g_1, g_2, g_3, g_4\}$; this result is consistent with the groups defined in **Table 4-2** (for clarity we use lower case g to represent normalized grouping). To demonstrate the value of applying a normalizing transformation prior to factor grouping, **Fig. 4-6**(**b**) presents the dendrogram obtained using the untransformed IVARS-50 sensitivity index (upper case G is used to represent un-normalized grouping). Without normalization, the 9 strongly influential factors of **Table 4-2** are clustered into 3 smaller groups as $G_1 = \{x_1\}, G_2 = \{x_4, x_7\}$, and $G_3 = \{x_3, x_8, x_2, x_6, x_5, x_9\}$, while the remaining 41 factors are gathered into one big group (G_4).



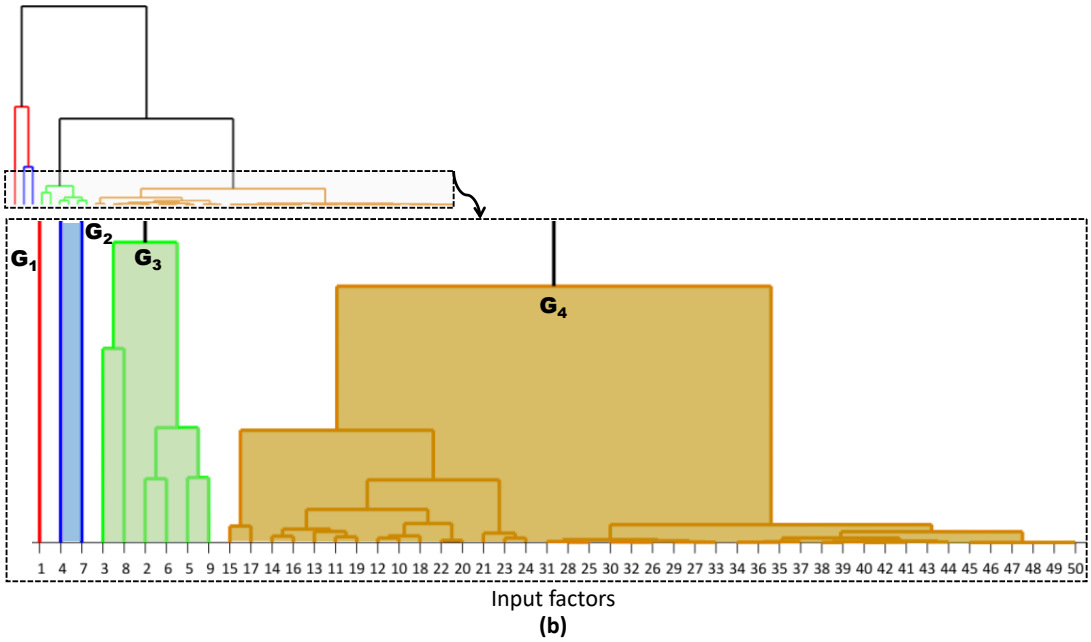


Figure 4-6 Factor grouping results for Sobol g-function based on (a) normalized sensitivity indices, and (b) sensitivity indices without normalization. The dashed lines show a zoom-in of the dendrogram for small values. Note that groups are labeled in order of importance (g_1 is the most important one).

Overall, the proposed factor grouping strategy makes the problem more tractable and results in a significantly higher convergence rate. To elaborate, consider factors x_1 and $x_2$that are (by

design) equally the most influential factors ($\alpha_1 = \alpha_2 = 0$, see **Table 4-2**). While we would expect x_1 and x_2 to converge to the same sensitivity index value, **Fig. 4-7** shows that convergence has not been achieved after performing 190,200 function evaluations. In contrast, application of the grouping algorithm **(Fig. 4-6(a))**, quickly clustered x_1 and x_2 into the same group g_1 (the group of most influential factors), indicating a high convergence rate. Comparison of **Fig. 4-6(a)** and **Fig. 4-6(b)** additionally highlights the importance of normalization because without normalization x_1 and x_2 are clustered into two different groups $G_1$and G_2 respectively (see **Fig. 4-6(b)**).

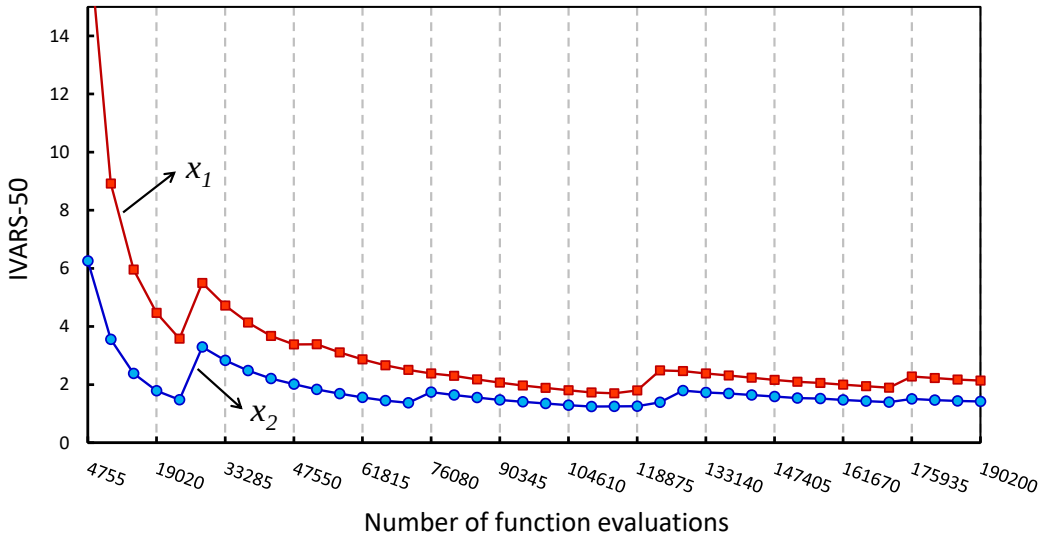


Figure 4-7 Convergence plot for the sensitivity indices associated with $\boldsymbol{x_1}$ and $\boldsymbol{x_2}$ for the Sobol g-function estimated using an increasing number of model evaluations.

4.5.1.2 Effect of sampling variability on factor grouping results

Robustness of the factor grouping algorithm depends directly on how robust our estimate of the sensitivity matrix is. If the GSA algorithm provides sufficiently robust estimates of the sensitivity indices, the grouping results can be expected to be robust as well. The AHC algorithm that has been used in our proposed grouping strategy is a deterministic algorithm without any random components. As a result, it provides a unique dendrogram (i.e., clustering) for a given set of sensitivity indices until these indices change. Therefore, one can evaluate the impact of

sampling variability on factor grouping results by performing multiple trials of sensitivity analysis with different sample sets and comparing the resultant dendrograms. However, accounting for sampling variability using multiple trials of GSA may be an impractical approach for high-dimensional and computationally intensive models such as MESH. On the other hand, to ensure that our proposed grouping technique operates as expected, various testing methods should be implemented during the numerical experiments. Therefore, in this section, we investigate the effects of sampling variability on grouping-enabled VARS algorithm by carrying out several trials of sensitivity analysis initiated using different random seeds. We do this only for the Sobol g-function which is computationally cheaper than MESH, by running 40 independent trials of STAR-VARS.

Comparing multiple trials of sensitivity analysis through factor grouping requires a method for comparing dendrograms. Different measures for assessing the degree of similarity (association) between various classifications (i.e., dendrograms) exist, including the cophenetic correlation coefficient (Sokal and Rohlf, 1962) and Baker's index (Baker, 1974). Here, we used Baker's index to investigate how the results of the factor grouping algorithm vary between several replicates of the STAR-VARS. Given a pair of dendrograms, Baker's index can be calculated by taking two factors (model parameters) and finding the highest possible level of k (number of groups created when cutting the dendrogram) for which these two factors belongs to the same tree. This procedure is repeated for the same two factors in the other dendrogram. Overall, in a d-dimensional model, there are $d(d-1)/2$ possible combinations of such pairs of factors, and accordingly all these numbers can be computed for each of the two dendrograms. Finally, these two sets of numbers are paired with respect to the pairs of factors compared, and a rank correlation (e.g., Spearman's correlation coefficient) between them is calculated (Galili, 2015). Accordingly, unlike other measures, the Baker's index is only sensitive to the relative position of branches in the dendrograms and is insensitive to the heights of the branches. This important feature makes it suitable for comparing factor grouping results.

Calculation of the Baker's indices associated with pairwise comparison of the 40 dendrograms obtained from the GSA of Sobol g-function resulted in a 40-by-40 symmetric correlation matrix containing the Baker's indices between each dendrogram and the others as shown in **Fig. 4-8**.

Inspection of these results reveals that almost all runs yielded very similar factor groupings, i.e., the minimum value of Baker's index we obtained is 0.991 and the median is 0.999 in **Fig. 4-8**. This is not surprising, since it has been shown that STAR-VARS is a robust GSA technique that provides stable results (Razavi and Gupta, 2016b), and consequently the comparison of the grouping results confirms the robustness of STAR-VARS to sampling variability.

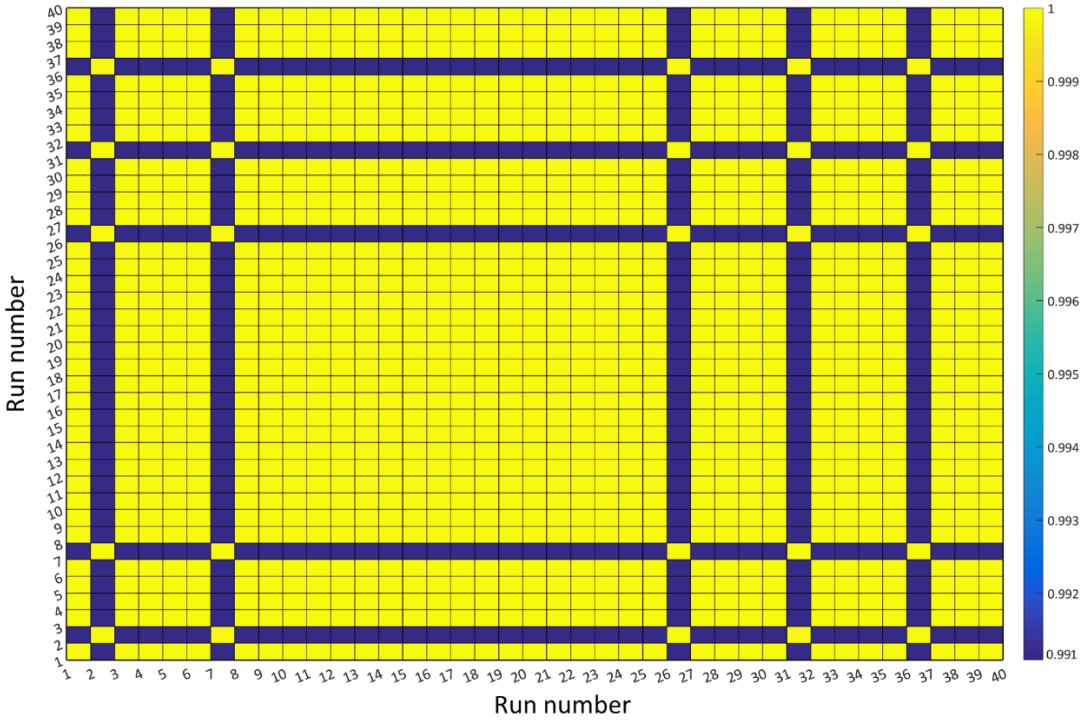


Figure 4-8 Baker's index for comparing factor grouping results obtained from 40 runs of Sobol g-function. This measure varies between -1 to 1, with values close to 0 indicates that the two dendrograms are not statistically similar. The diagonal elements in this matrix is equal to 1 as they indicate the similarity of a dendrogram with itself.

4.5.1.3 Results for the MESH model

The model parameter grouping obtained for the second case study (MESH model) is shown in **Fig. 4-9**. Using the elbow method, the 111 MESH model parameters were automatically classified into 7 groups. These groups are ordered based on their importance, i.e., g_1 contains the most strongly-influential parameters, while parameters in g_7 are minimally-influential. The first group (g_1) consists of 4 parameters controlling water storage and movement in the soil (SDEP,

DRN), river channel routing (WFR22), and snow cover fraction (ZSNL). Groups 2 through 6 contain parameters related to soil and vegetation properties, control of overland and interflow generation, and initial conditions. Note that the parameters specifying initial conditions, including soil initial moisture content (THLQ1,2,3), temperature (TBAR1,2,3) and initial surface ponding (TPOND) are all located in g_4, except for initial canopy temperature (TCANO), which is in g_7(least influential). This placement of TCANO may be due to the fact that model simulations start in October, which is not part of the canopy growing season. Other parameters in g_7 are ponding parameters corresponding to GRU number 7 that has small area fractions compared to that of other GRU types (ZPLG7 and ZPLS7), and the Manning's coefficient (MANNG) used in calculation of overland flow (see **Appendix** for parameter description).

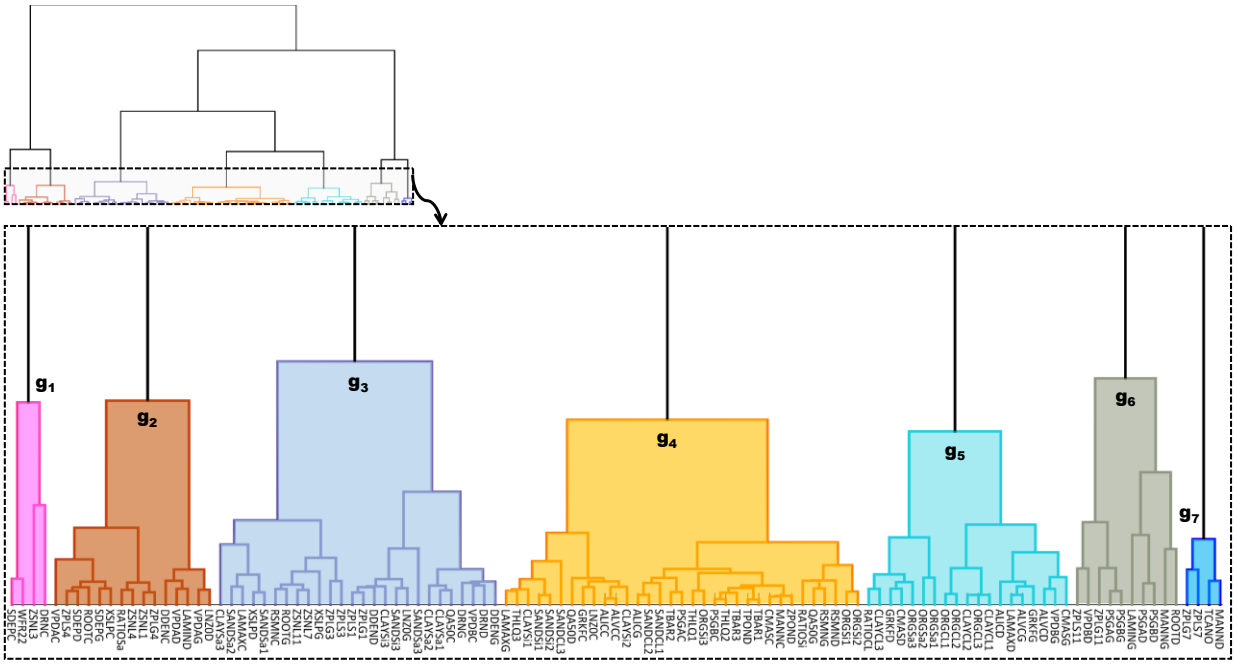


Figure 4-9 Factor grouping results for MESH model. The dashed line shows a zoom-in of the dendrogram for small values. The *x*-axis labels correspond to model parameters. The groups are labeled in order of importance.

4.5.2 Assessment of the convergence and robustness of factor ranking results

4.5.2.1 Results for the Sobol g-function

For the first case study, **Fig. 4-10**(**a**) illustrates the convergence behavior of the proposed robustness measure, while **Fig. 4-10**(**b**) shows the robustness values based on individual factor ranking computed as described in Razavi and Gupta (2016b). Whereas the robustness values estimated based on individual factors deteriorates for the first 28,350 evaluations (clearly indicating insufficient sample size), the factor grouping converges quickly towards probability 1 (i.e., 100%) after about only half that number (~15,000), and from 28,350 function evaluations onwards the grouping is stable (**Fig. 4-10**(**a**)). Note also that the robustness measure based on individual factor rankings continue to vary between 0.18 and 1 (median = 0.55) even for well over 150,000 evaluations.

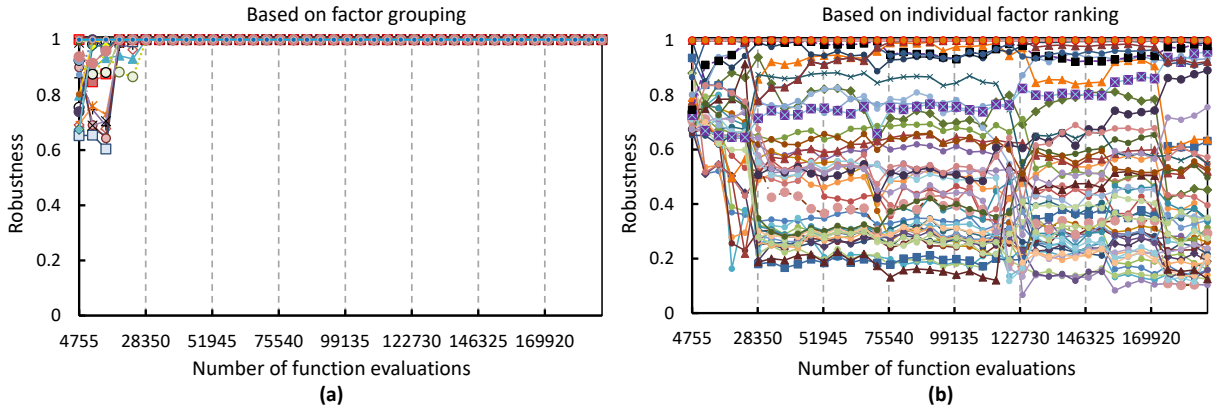


Figure 4-10 Comparison of the assessment of robustness based on (a) factor grouping and (b) individual factor ranking. In subplots (a) and (b) each line represents the evolution of robustness values associated with each factor of the Sobol g-function.

4.5.2.2 Results for the MESH model

The robustness assessment results for the second case study are shown in **Fig. 4-11**(**a**) and (**b**). After 90,000 function evaluations, the estimated factor grouping-based robustness values are all higher than 0.50 (**Fig. 4-011**(**a**)), whereas for individual factor ranking-based values they continue to vary between 0.04 and 1.0 (**Fig. 4-11**(**b**)). For clarity, **Fig. 4-11(c)** shows the trajectories of the medians of both the individual (red dashed line) and group-based (blue solid line) factor ranking results. Notice that the group-based median rank is already quite high (0.80)

after 20,000 function evaluations and increases rapidly thereafter to be above 0.90 from 50,000 evaluations on, finally reaching 0.99 at 100,000 model runs. Given there are 111 parameters, this means the grouping-based approach has achieved greater than 90% robustness in identifying the 55+ most sensitive parameters of this model with only 40,000-50,000 sample points. In contrast, the median rank for the individual factor method remains low and is only 0.17 at the termination of the experiment.

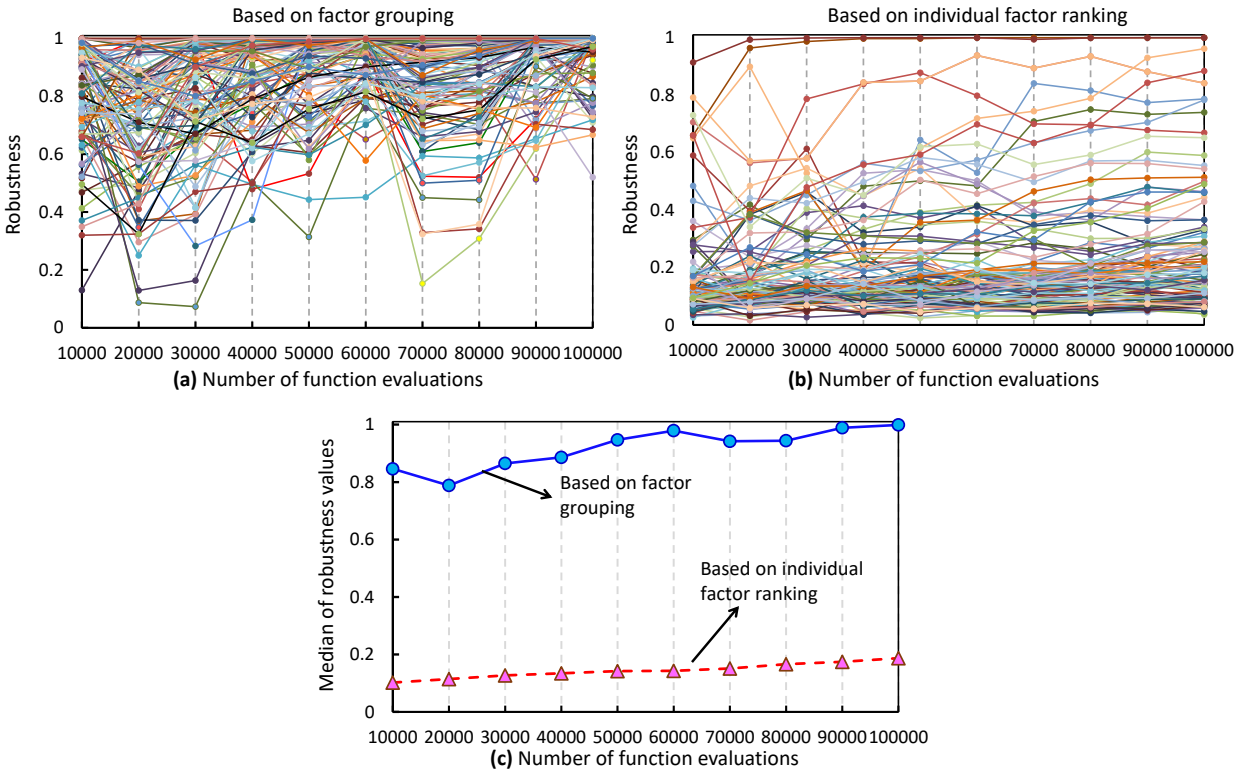


Figure 4-11 Comparison of the assessment of robustness based on (a) factor grouping and (b) individual factor ranking. Subplot (c) shows the median of the individual (red dashed line) and group-based (blue solid line) factor ranking results. In subplots (a) and (b) each line represents the evolution of robustness values associated with each parameter of the MESH model.

In support of this, **Fig. 4-12(a-g)** shows the trajectories of robustness values for each member of each of the 7 identified groups (from most strongly influential to non-influential). We see clearly that groups 1 (**Fig. 4-12(a)**) and 2 (**Fig. 4-12(b)**) (the two most strongly-influential) and group 7 (**Fig. 4-12(g)**) (the least influential) have been well established (with high robustness) after about 50,000 function evaluations. In contrast, the robustness values for members of the

intermediate groups (3-6) (**Fig. 4-12(c-f)**) tend to be more variable, with some members having high robustness and others having relatively lower robustness. For the latter, this indicates that their group membership is less certain and that they may continue to shift between groups as the sampling proceeds. Nonetheless, the most and least influential parameter groups are relatively well established quite early in the GSA procedure. Finally, **Fig 4-12(h)** shows, together, the trajectory of median robustness values for each of the seven groups. This plot can be useful for oversight monitoring of the convergence of the factor grouping procedure.

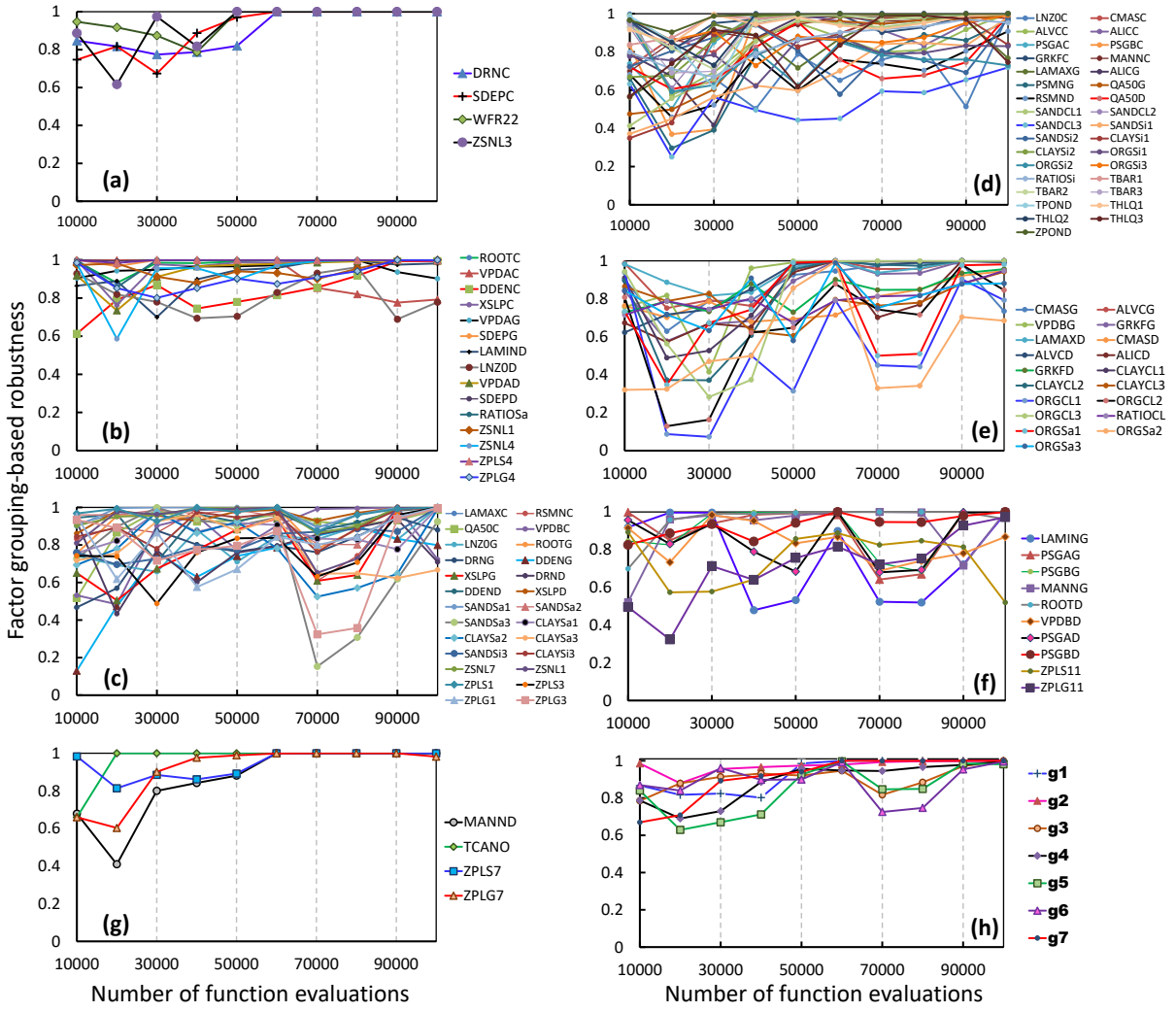


Figure 4-12 Trajectories of robustness values for each parameter of the MESH model in groups: (a) g_1, (b) g_2, (c) g_3, (d) g_4, (e) g_5, (f) g_6, and (g) g_7. Subplot (h) shows the evolution of median of the robustness values for each of the seven groups for the increasing number of function evaluations.

Overall, these results clearly illustrate the value of using factor grouping-based robustness estimates for monitoring convergence of GSA applied to high-dimensional models. As indicated in **Fig. 4-2**, when convergence has not yet been achieved, the user can iterate by increasing the original sample size *n* using a sequential sampling scheme such as PLHS.

4.5.3 Comparison of the proposed strategies for finding optimal number of groups

Fig. 4-13 (top panel) illustrates the evolution of the optimal number of groups obtained by the elbow method. As the STAR-VARS algorithm progresses (**Fig. 4-13**(**a**) and (**b**)), the grouping algorithm evolves and discovers groups of parameters that share specific properties in terms of their sensitivity. To further investigate the performance of the elbow method, the merging costs (distance metric) versus the number of groups are plotted for the Sobol g-function (**Fig. 4-13**(**c**)) and MESH model (**Fig. 4-13**(**d**)) when the number of function evaluations is 190,200 and 100,000 respectively. **Fig. 4-13** (bottom panel) confirms the ability of the elbow method to successfully determine the optimal number of groups by finding the elbow point of the curve (point of maximum curvature). For the Sobol g-function, the elbow method converges rather quickly to an optimal number of groups; however, for the MESH model, the optimal number continues to vary over the range 7-13 for the number of function evaluation examined.

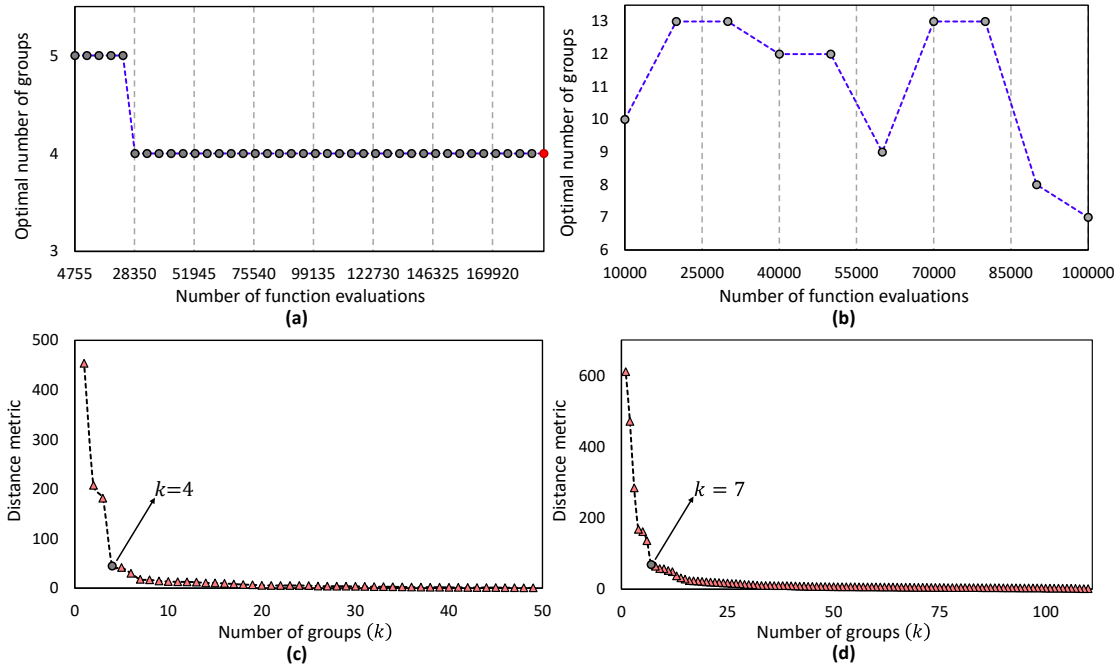


Figure 4-13 Top panel shows the evolution of the optimal number of groups with computational budget for (a) Sobol g-function and (b) MESH model. Bottom panel shows plots of the distance metric (*y*-axis) versus number of groups (*x*-axis) for (c) Sobol g-function and (d) MESH model when number of function evaluations is maximum. From $\boldsymbol{k = 4}$ in subplot (c) and $\boldsymbol{k = 7}$ in subplot (d) onwards the curves flatten notably.

To illustrate the performance of the minimum robustness method in determining optimal number of groups, we calculated the maximum number of groups that is required to have minimum robustness values of 0.90 for the Sobol g-function (**Fig. 4-14**(**a**)) and 0.45 for the MESH model (**Fig. 4-14**(**b**)), as the number of function evaluations grows. The top panel of **Fig. 4-14** indicates that the number of groups tends to be small for low numbers of function evaluations and tends to increase as the number of function evaluations increases.

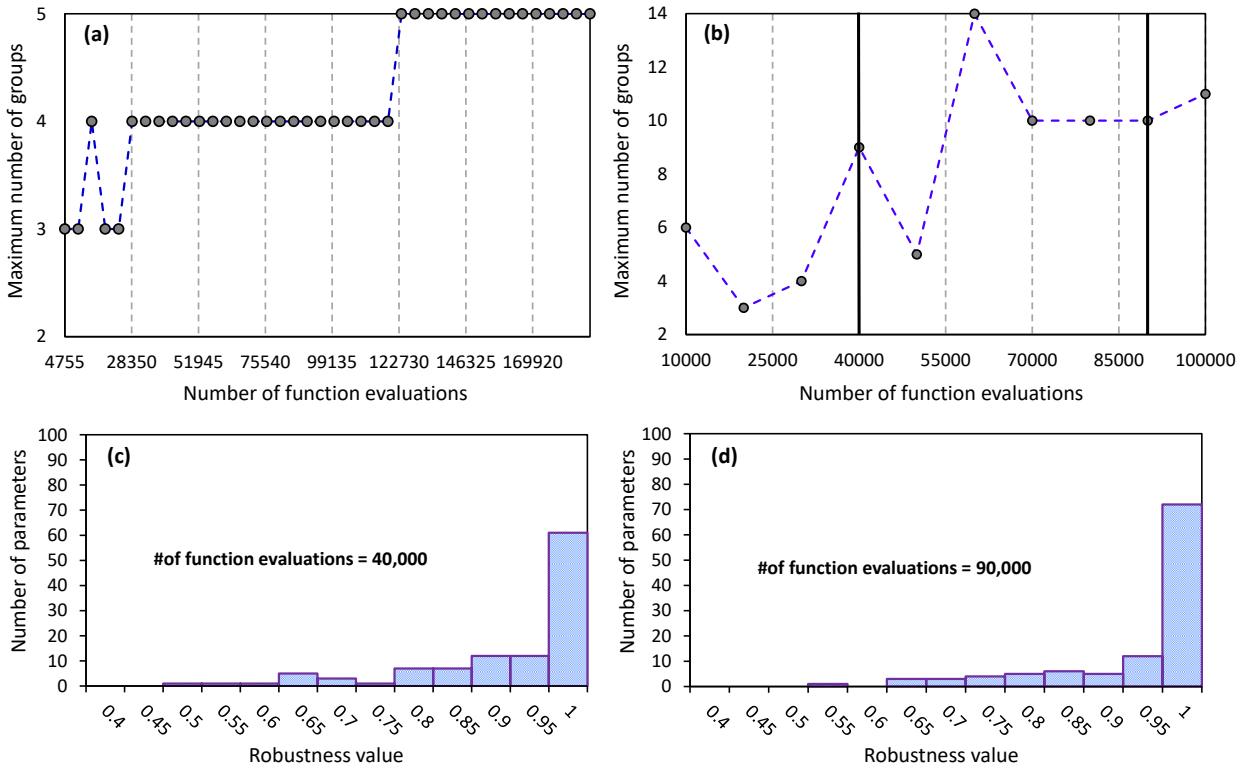


Figure 4-14 Maximum number of groups that is required to achieve minimum robustness values of (a) 0.90 for Sobol g-function and (b) 0.45 for MESH model parameters as the number of function evaluations grows. Bottom panel shows the histograms of the estimated robustness values for MESH model after (c) 40,000 and (d) 90,000 function evaluations.

Finally, histograms of the estimated robustness values are shown in **Fig. 4-14 (c)** and **(d)** for the MESH model for 40,000 and 90,000 function evaluations, when the number of groups is equal to 9 and 10, respectively. Although the minimum desired robustness level was set to 0.45, the estimated robustness value for most of the parameters is greater than 0.60. In fact, when the number of function evaluations is 40,000, only two parameters, out of 111 parameters, have robustness values less than 0.60, while after 90,000 function evaluations, except one parameter, all the parameters have the robustness values greater than 0.60.

4.6 Conclusions

Global sensitivity analysis (GSA) is a powerful tool for deepening our understanding of Complex Environmental Systems Models (CESMs), providing information helpful for model-development, parameterization, calibration, and data-acquisition. Advanced CESMs are commonly characterized by large parameter/problem spaces and high computational overheads, which impede the effective implementation of the modern GSA techniques because an extensive sensitivity analysis often requires a computationally infeasible number of model runs. To break down this barrier, we have introduced an automated "*factor grouping*" strategy that can be used with any GSA algorithm to reliably cluster input factors into groups of different sizes using information gained during the GSA. Our proposed grouping approach is based in the use of an efficient strategy (elbow method and/or minimum robustness method) to determine the optimal number of groups. While the elbow method removes the subjectivity involved in selecting the number of groups, the minimum robustness method is more flexible but requires specifying a minimum robustness value. Additionally, we developed and tested a new measure of robustness based on factor grouping to monitor and evaluate convergence of the GSA algorithm.

To illustrate the approach, we implemented the factor grouping algorithm the VARS variogram-based GSA technique and demonstrated its utility for parameter sensitivity analysis of two high-dimensional case studies, the 50 parameter Sobol g-function and the 111 parameter MESH large-scale land surface-hydrology model. For the Sobol g-function, we assessed the effect of sampling variability on the grouping-enabled VARS algorithm by running multiple replicates of the algorithm with different original sample sets. The results of our experiment illustrate the robustness of the grouping strategy combined with VARS method for GSA. The results confirm that grouping-enabled GSA approach successfully recognizes the dominant groups of factors that contribute significantly to the variability of the model outputs, while requiring only a limited number of function evaluations to converge. Of course, to better understand how robustness of the chosen GSA method can affect the robustness of factor grouping results, several GSA methods should be tested in conjunction with the proposed grouping strategy.

Author contributions

RS developed the factor grouping scheme and wrote the codes in MATLAB. RS, SR, and HVG contributed to the method conceptualization. RS and SR designed the experiments. All the numerical experiments were carried out by RS. AH performed the MESH model runs. RS analyzed the results and primarily wrote the manuscript. All co-authors critically reviewed the manuscript.

Chapter 5

Efficient Strategies for Handling Simulation Model Crashes in Global Sensitivity Analysis

This chapter is a mirror of the following manuscript with minor changes to increase its consistency with the body of the dissertation. Changes were only made to avoid repeating the contents that have been presented more appropriately in other parts. References are unified at the end of the dissertation.

Sheikholeslami, R., Razavi, S., and Haghnegahdar, A. 2019. What do we do with model simulation crashes? Recommendations for global sensitivity analysis of earth systems models. *Geoscientific model Development*, Discussion, https://doi.org/10.5194/gmd-2019-17

Synopsis

Complex, software-intensive, technically advanced, and computationally demanding models, presumably with ever-growing realism and fidelity, have been widely used to simulate and predict the dynamics of the Earth and Environmental Systems. The parameter-induced simulation crash (failure) problem is typical across most of these models, despite considerable efforts that modelers have directed at model development and implementation over the last few decades. A simulation failure mainly occurs due to the violation of the numerical stability conditions, non-robust numerical implementations, or mistakes made in the course of programming. However, the existing sampling-based analysis techniques such as global sensitivity analysis (GSA) methods, which require running these models under many configurations of parameter values, are ill-equipped to effectively deal with model failures. To tackle this problem, we propose a novel approach that allows users to cope with failed designs (samples) during the GSA, without knowing where they will take place and without re-running the entire experiment. This approach deems model crashes as missing data and uses strategies such as median substitution, single nearest neighbor, or response surface modelling to fill in for

model crashes. We test the proposed approach on a 10-parameter conceptual rainfall-runoff model and a 111-parameter land surface-hydrology model. Our results show that response surface modelling is a superior strategy, out of the data filling strategies tested, and can comply with certain requirements concerning the dimensionality of the model, sample size, and the ratio of number of failures to the sample size. Further, we conduct a "failure analysis" and discuss some possible causes of the MESH model failure.

5.1 Introduction

5.1.1 Background and motivation

Since the start of the digital revolution and subsequent increase in computers' processing power, the advancement of information technology has led to significant development of the modern software programs for Complex Environmental Systems Models (CESMs). The current-generation CESMs typically span upwards of several thousand lines of code and require huge amounts of data and computer memory. The flip side of the growing complexity of CESMs is that running these models will pose many types of software development and implementation issues such as simulation crashes/failures. The simulation crash problem happens mainly due to violation of the numerical stability conditions needed in CESMs. Certain combinations of model parameter values, improper integration time step, inconsistent grid resolution, or lack of iterative convergence as well as model thresholds and sharp discontinuities in model response surfaces, all associated with imperfect parameterizations, can cause numerical artefacts and stop CESMs from properly functioning.

When model crashes occur, the accomplishment of automated sampling-based model analyses such as sensitivity analysis, uncertainty analysis, and optimization (e.g., Raj et al., 2018; Williamson et al., 2017; Metzger et al., 2016; Safa et al., 2015) becomes challenging. These analyses are often carried out by running CESMs for a large number of parameter configurations randomly sampled from a domain (parameter space). In such situations, for example, the model's solver may break down because of the implausible combinations of parameters ("unlucky

parameter set" as termed by Kavetski et al., (2006)), failing to complete the simulation. It is also possible that a model may be stable against perturbation of one parameter, while it may crash when several parameters are perturbed simultaneously. "Failure analysis" is a process that is performed to determine the cause(s) that have led to such crashes while running CESMs. Before achieving a conclusion on the most important causes of crashes, it is necessary to check the software code used in the CESMs and make sure if it is error-free; for example, a proper numerical scheme was adopted and correctly coded in the software. This often requires investigating both the software documentation and a series of nested modules. However, the existence of numerous nested programming modules in a typical CESMs can make the identification and removal of all software defects so tedious. In addition, as argued by Clark and Kavetski (2010), the numerical solution schemes implemented in CESMs are sometimes not presented in detail. This is one important reason why detecting the causes of simulation crashes in DESMs is usually troublesome. For example, Singh and Frevert (2002) and Burnash (1995) described the governing equations of their models without explaining the numerical solvers that were implemented in their codes.

Importantly, the impact of simulation crashes on the validity of global sensitivity analysis (GSA) results has often been overlooked in the literature, where simulation crashes are commonly classified as ignorable (see section 1.2). As such, a surprisingly limited number of studies have reported simulation crashes (examples related to uncertainty analysis include Annan et al., 2005; Edwards and Marsh, 2005; Lucas et al., 2013). This is despite the fact that these crashes can be very computationally costly for GSA algorithms because they can waste the rest of the model runs, prevent completion of GSA, or inevitably introduce ambiguity into the inferences drawn from GSA. For example, Kavetski and Clark (2010) demonstrated how numerical artefacts can contaminate the assessment of parameter sensitivities in six hydrological models. Therefore, it is important to devise solutions that minimize the effect of crashes on GSA results. In the next subsection, we critically review the very few strategies for handling simulation crashes that have been proposed in the literature and identify their shortcomings.

5.1.2 Existing Approaches to handling simulation crashes

We have identified four types of approaches in the modelling community to handling simulation crashes, outlined below. The first two are perhaps the most common approaches (based on our personal communications with different modelers), however, we could not identify any publication that formally reports their application.

1. After the occurrence of a crash, modelers commonly adopt a *conservative* strategy to address this problem by altering/reducing the feasible ranges of parameters and re-starting the experiment in a hope to prevent a recurrence of the crash for new analyses.
2. Instead of GSA that runs many configurations of parameter values, analysts may choose to employ local methods such as local sensitivity analysis (LSA) through running the model in the vicinity of the known plausible parameter configurations.
3. Some modelers may adopt an *ignorance-based* approach by using only a set of ‘‘good’’ (or behavioral) outcomes/responses in sampling-based analyses and ignoring unreasonable (or non-behavioral) outcomes such as simulation crashes. This can be done via defining a performance metric to determine which simulations should be excluded from the analysis (see, e.g., Pappenberger et al., 2008; Kelleher et al., 2013).
4. The most rigorous approach seems to be a *non-substitution* approach that tries to predict whether a set of parameter values will lead to a simulation crash. Webster et al. (2004), Edwards et al. (2011), Lucas et al. (2013), Paja et al. (2016), and Treglown (2018) are among few studies that mainly aimed at developing statistical methods to predict whether a given combination of parameters is likely to result in a simulation failure. For example, Lucas et al. (2013) adopted a machine learning method to estimate the probability of crash occurrence as a function of model parameters. They further applied this approach to investigate the impact of various model parameters on simulation failures.

The above approaches, however, have major shortcomings and limitations in handling computer crashes in the GSA context, because:

1. Locating regions of parameter space responsible for crashes (i.e., “implausible regions”) is difficult and requires analyzing the behavior of CESMs throughout the often high-dimensional parameter space. Implausible regions usually have irregular, discontinuous, and complex shapes, and thus are too effortful to identify. Also, changing/reducing the parameter space changes the original problem at hand.
2. It is well-known that local methods (e.g., LSA) can provide inadequate assessments that can often be misleading (see e.g., Saltelli and Annoni, 2010, Razavi and Gupta, 2015).
3. When applying a sampling-based technique that uses an ad-hoc sampling strategy with particular spatial structure (e.g., the variance-based GSA proposed by Saltelli et al. (2010) or STAR-VARS of Razavi and Gupta (2016b)), ignorance-based procedures become impractical. In this case, excluding sample points associated with simulation crashes will distort the structure of the sample set, causing the failure of the entire GSA experiment. As a result, a new sample set (or a succession of sample sets) must be generated to resume the experiment, leading to a waste of previous model runs.
4. The implementation of the non-substitution procedures necessitates significant prior efforts to identify many model crashes based on which a statistical model can be built to predict and avoid simulation failures in the subsequent use of the model. Such procedures can easily become infeasible in high-dimensional models, as then they would require an extremely large sample size to ensure an adequate coverage of the parameter space for characterizing implausible regions and building a reliable statistical model. These strategies can be more challenging when a model is computationally intensive. For example, to determine which parameters or combinations of parameters in a 16-dimensional climate model were predictors of failure, Edwards et al. (2011) used 1,000 evaluations (training samples) for constructing a statistical model to identify parameter configurations with high probability of failure in the next 1,087 evaluations (2,087 model runs in total). As pointed out by Edwards et al. (2011), although 2,087 evaluations might impose high computational burdens, a much larger sample size spreading out over the parameter space is required to guarantee reasonable exploration of the 16-dimensional space.

These shortcomings and gaps motivated our investigation to develop effective and efficient crash handling strategies suitable for GSA of CESMs introduced in **section 5.2**.

5.1.3 Objective and scopes

The primary goal of this chapter is to identify practical "substitution" strategies to handle parameter-induced crash problem in GSA of high-dimensional CESMs. Here, we treat computer crashes as missing data and investigate the effectiveness of three efficient strategies to replace them using available information rather than directly discarding them. Our approach allows the user to cope with failed designs in GSA without knowing where they will take place and without re-running the entire experiment. The overall procedure can be used in conjunction with any GSA technique. In this chapter, we assess the performance of this approach on two hydrological models, by coupling it with the variogram-based GSA technique (VARS; Razavi and Gupta (2016a,b)).

The rest of this chapter is structured as follows. We begin in the next section (**5.2**) by introducing our proposed solution methodology for dealing with computer crashes. In **section 5.3**, two real-world hydrological modelling case studies are presented. Next, in **section 5.4**, we evaluate and discuss the performance of the proposed methods across these real-world problems, before drawing conclusions and summarizing major findings in **section 5.6**.

5.2 Methodology

5.2.1 Problem statement

We denote the output of each model run (realization) $y(\boldsymbol{X})$, which corresponds to a d-dimensional input vector $\boldsymbol{X} = \{x_1, x_2, \dots, x_d\}$, where $x_i\ (i = 1,2, \dots, d)$ is a factor that may be perturbed for the purpose of GSA (e.g., model parameters, initial conditions, or boundary conditions). Running a GSA algorithm usually requires generating n realizations of a computer code using an experimental design $\boldsymbol{X}_s = \{\boldsymbol{X}_1, \boldsymbol{X}_2, \dots, \boldsymbol{X}_n\}^{\mathrm{T}}$. Then, the model responses will form an output space as $\boldsymbol{Y} = \{y(\boldsymbol{X}_1), y(\boldsymbol{X}_2), \dots, y(\boldsymbol{X}_n)\}^{\mathrm{T}}$. Here we deem simulation crashes as missing data and consider the model mapping of $\boldsymbol{X}_s \rightarrow \boldsymbol{Y}$ as an incomplete data matrix. For a given $\boldsymbol{Y} \in$

$\Re^{1\times n}$ with missing values, let the vector $\boldsymbol{Y}_a$ consist of the n_a locations in the input space for which, in the given $\boldsymbol{Y}$, the model responses are available, and let the vector $\boldsymbol{Y}_m$ consist of the remaining n_m locations ($n_m = n - n_a$) for which, in the given $\boldsymbol{Y}$, the model responses are missing due to simulation crashes. For convenience of expression and computation, we use the "NAN_j" symbol to represent the jth missing value in vector $\boldsymbol{Y}$. The main goal now here is to develop and test data recovery methods that can be used to substitute model crashes $\boldsymbol{Y}_m$ using available information (i.e., $\boldsymbol{Y}_a$ and $\boldsymbol{X}_s$).

5.2.2 Proposed strategies for crash handling in GSA

We propose and test three techniques adopted from the "incomplete data analysis" for missing data replacement; the process known as imputation (Little and Rubin, 1987). We use imputation techniques to fill in missing values that ignore the mechanisms leading to the missingness. Therefore, only the non-missing responses and the associated sample points are included in our analysis to infill model crashes during GSA, as described in the next sub-sections.

5.2.2.1 Median substitution

Perhaps replacing each simulation crash with some "central" value is the easiest and computationally simple method for imputation. Depending on the distribution of the model response variables $\boldsymbol{Y}$, the central value can be mean or median. For example, if the distribution of model responses is not highly skewed, the crashes may be imputed with the mean of the non-missing values. Otherwise, if the distribution exhibits skewness, then the median may be better replacement. This strategy, known as statistical imputation, treats each model response as a realization of a random function, while ignoring the covariance structure of model responses, and thus considers the mean/median as a reasonable estimate for missing data. Although mean substitution preserves the mean of $\boldsymbol{Y}$, a major shortcoming of this technique is that, depending on the number of crashes, it can distort other statistical characteristics of $\boldsymbol{Y}$ through reducing its variance. In this chapter, the median substitution technique has been utilized.

5.2.2.2 Nearest neighbor substitution

The Nearest Neighbor (NN) technique (also known as hot deck imputation) uses observations in the neighborhood to fill in missing data. Let $\boldsymbol{X}_j \in \boldsymbol{X}_s$ be an input vector for which a simulation model fails to return an outcome, i.e., $y(\boldsymbol{X}_j) = NAN$. Basically, in the NN-based techniques, $y(\boldsymbol{X}_j)$ is replaced by either a response value corresponding to a single nearest neighbor (single NN) or a weighted average of the response variables corresponding to k nearest neighbors (k-NN) where $k > 1$. In the k-NN techniques weights are typically assigned based on the degree of similarity between $\boldsymbol{X}_j$ and kth nearest neighbor $\boldsymbol{X}_k$ where $y(\boldsymbol{X}_k) \in \boldsymbol{Y}_a$ (Tutz and Ramazan, 2015).

The underlying rationale behind NN procedure is that the sample points closer to $\boldsymbol{X}_j$ may provide better information for imputing $y(\boldsymbol{X}_j)$. An important feature of the NN technique is that the variance of the $\boldsymbol{Y}$ variables tend to be preserved for $k = 1$ but not for $k > 1$ (Moeur and Stage, 1995; McRoberts, 2009). Another advantage of the single NN over the k-NN techniques is that it does not require a pre-specification of the number of neighbors. Furthermore, single NN substitution does not extrapolate outside the range of the sampled output space and, instead, fill-in values are determined from the pool of non-missing values. Here, we used the single NN technique with Euclidean distance measure.

5.2.2.3 Model emulation-based substitution

Model emulation is a strategy that develops statistical, cheap-to-run surrogates of response surfaces of complex, often computationally intensive models (Razavi et al., 2012a; Castelletti et al., 2012a,b). Here we develop an emulator $\hat{y}(.)$which is a statistical approximation of the simulation model based on response surface modelling concept. This strategy consists in finding an approximate/surrogate model with low computational cost that fits the non-missing response values $\boldsymbol{Y}_a$ to predict the fill-in values for the missing responses $\boldsymbol{Y}_m$. In the literature various types of response surface surrogates exist and are extensively discussed (see e.g. Razavi et al., 2012a). Examples are polynomial regression, radial basis functions (RBF), neural networks, kriging, support vector machines, and regression splines. In this chapter, we employed RBF

approximation as a well-established surrogate model. It has been shown that RBF can provide an accurate model for high-dimensional problems (Jin et al., 2001; Herrera et al., 2011), particularly when the computational budget is limited (Razavi et al., 2012b). The predictive response $\hat{y}(\boldsymbol{X})$ at design point $\boldsymbol{X}$ can be approximated by an RBF model as a weighted summation of n_a basis functions (and a polynomial or constant value) as follows:

$$\hat{y}(\boldsymbol{X}) = \sum_{i=1}^{n_a} \omega_i f(\|\boldsymbol{X} - \boldsymbol{X}_i\|) = \boldsymbol{f}(\boldsymbol{X})\boldsymbol{\omega} \tag{1}$$

where $\boldsymbol{f} = \{f_1, f_2, \dots, f_{n_a}\}$ is the vector of the basis functions, ω_i is the ith component of the radial basis coefficient vector $\boldsymbol{\omega} = \{\omega_1, \omega_2, \dots, \omega_{n_a}\}^{\mathrm{T}}$, and $\|\boldsymbol{X} - \boldsymbol{X}_i\|$ is the Euclidian distance between two sample points.

There are various choices for the basis function, such as Gaussian, thin-plate spline, multi-quadric, and inverse multi-quadric (Jones, 2001). In the present study, we choose the widely-used Gaussian kernel function for RBF as

$$f(\|\boldsymbol{X} - \boldsymbol{X}_i\|) = exp\left(\frac{\|\boldsymbol{X} - \boldsymbol{X}_i\|^2}{c_i^2}\right) \tag{2}$$

where c_i is the shape parameter which determines the spread of the ith kernel function f_i.

After choosing the form of the basis function, the coefficient vector $\boldsymbol{\omega}$ can be obtained by enforcing the accurate interpolation condition, i.e.,

$$\begin{bmatrix} y(\boldsymbol{X}_1) \\ y(\boldsymbol{X}_1) \\ \vdots \\ y(\boldsymbol{X}_{n_a}) \end{bmatrix} = \begin{bmatrix} f_{11} & f_{12} & \dots & f_{1n_a} \\ f_{21} & f_{22} & \dots & f_{2n_a} \\ \vdots & \vdots & \ddots & \vdots \\ f_{n_a1} & f_{n_a2} & \dots & f_{n_an_a} \end{bmatrix} \begin{bmatrix} \omega_1 \\ \omega_2 \\ \vdots \\ \omega_{n_a} \end{bmatrix} \tag{3}$$

where $f_{uv} = f(\|\boldsymbol{X}_u - \boldsymbol{X}_v\|)$. In a matrix form, Eq. (3) can be simply rewritten as $\boldsymbol{Y}_a = \boldsymbol{F}\boldsymbol{\omega}$. This equation has a unique solution $\boldsymbol{\omega} = \boldsymbol{F}^{-1}\boldsymbol{Y}_a$ if and only if all the sample points are different

from each other. Therefore, the fill-in values for remaining n_m locations, for which the model responses are missing due to simulation crashes, can be approximated by

$$\hat{y}(\boldsymbol{X}_j) = \boldsymbol{f}(\boldsymbol{X}_j)\boldsymbol{F}^{-1}\boldsymbol{Y}_a \qquad (j = 1,2,\dots,n_m) \tag{4}$$

To reduce the computational cost and avoid overfitting when building RBF, for each failed design at $\boldsymbol{X}_j$ we only chose k non-missing nearest neighbors of that missing value (here we arbitrarily set k to 100). Then a function approximation was built using these 100 sample points to fill in that missing value, i.e., in Eq. (3), n_a was set to 100. Moreover, the shape parameter c in the Gaussian kernel function, which is an important factor in the accuracy of the RBF, was determined using an optimization approach. We used the Nelder-Mead simplex direct search optimization algorithm (Lagarias et al., 1998) to find an optimal value for c by minimizing the RBF fitting error (for more details see Forrester and Keane (2009) and Kitayama and Yamazaki (2011)).

Note that in general depending on the complexity and dimensionality of a model response surface, other types of emulations can be incorporated into our proposed framework. However, for the crash handling problem, it is beneficial to utilize the function approximation techniques that exactly fit to the all sample points (i.e., the response surface surrogates categorized as "Exact Emulators" in Razavi et al. (2012a)) such as kriging and RBF. This is mainly because CESMs are deterministic, and therefore generate identical outputs/responses given the same set of input factors. In other words, an exact emulator at any successful design point $\boldsymbol{X}_k$ (not crashed) reflects our knowledge about the true value of the model's output at that point, i.e., it returns $\hat{y}(\boldsymbol{X}_k)$ without uncertainty. Thus, exact emulators can be appropriate surrogates to adequately characterize the shape of the response surfaces in deterministic CESMs for handling simulation crashes.

5.2.3 The utilized GSA framework

We illustrate the incorporation of the proposed crash handling methodology into a variogram-based GSA approach called Variogram Analysis of Response Surfaces (VARS; Razavi and Gupta (2016a,b)). The VARS framework has successfully been applied to several real-world

problems of varying dimensionality and complexity (see e.g., Sheikholeslami et al., 2017; Yassin et al., 2017; Krogh et al., 2017; Leroux and Pomeroy, 2019). VARS is a general GSA framework that utilizes directional variograms and covariograms to quantify the full spectrum of sensitivity-related information, thereby providing a comprehensive set of the sensitivity measures called IVARS (Integrated Variogram Across a Range of Scales) at a range of different "perturbation scales" (Haghnegahdar and Razavi, 2017). Here, we used IVARS-50, referred to as "total-variogram effect", as a comprehensive sensitivity measure since it contains sensitivity analysis information across a full range of perturbation scales.

Here, the STAR-VARS implementation of the VARS framework has been used. STAR-VARS is highly efficient and statistically robust algorithm that provides stable results with minimum number of model runs compared with other GSA techniques, and thus is suitable for high-dimensional problems (Razavi and Gupta, 2016b). This algorithm employs a star-based sampling scheme, which consists of two steps: (1) randomly selecting star centres in the parameter space, and (2) using a structured sampling technique to identify sample points revolved around the star centres. Due to the structured nature of the generated samples in STAR-VARS, ignorance-based procedures (see section 1.2) cannot be useful in dealing with simulation crashes because deleting sample points associated with crashed simulations will demolish the structure of the entire sample set. In this study, to achieve a well-designed computer experiment, we used PLHS algorithm in the first step of the STAR-VARS to sequentially locate samples in the parameter space. It has been shown that PLHS can grasp maximum amount of information from output space with minimum sample size, while outperforming traditional sampling algorithms (for more details see Sheikholeslami and Razavi, (2017)).

5.3 Case Studies

5.3.1 A conceptual rainfall-runoff model

As an illustrative example we employ the HBV-SASK conceptual hydrologic model to assess the performance of the proposed crash handling strategies in a real-world problem. HBV-SASK is

based on Hydrologiska Byråns Vattenbalansavdelning model (Lindström et al., 1997) and developed for educational purposes. We applied HBV-SASK to simulate daily steamflows in the Oldman river basin in Western Canada (**Fig. 5-1**) with watershed area of 1434.73 km^2. Historical data are available for periods 1979-2008, from which we estimate average annual precipitation to be 611 mm, and average annual streamflow to be 11.7 m^3/s with a runoff ratio of approximately 0.42. HBV-SASK has 10 parameters that need to be specified/calibrated by the user (**Table 5-1**).

Table 5-1 HBV-SASK model parameters and their feasible ranges.

Parameter	***Range***	***Description***
TT	[-4,4]	Air temperature threshold in °C for melting/freezing and separating rain and snow
C0	[0,10]	Base melt factor, in mm/°C per day
ETF	[0,1]	Temperature anomaly correction in 1/°C of potential evapotranspiration
LP	[0,1]	Limit for PET as a multiplier to FC, i.e., soil moisture below which evaporation becomes supply limited
FC	[50,500]	Field capacity of soil, in mm. The maximum amount of water that the soil can retain
beta	[1,3]	Shape parameter (exponent) for soil release equation (unitless)
FRAC	[0.1,0.9]	Fraction of soil release entering fast reservoir (unitless)
K1	[0.05,1]	Fast reservoir coefficient, which determines what proportion of the storage is released per day (unitless)
alpha	[1,3]	Shape parameter (exponent) for fast reservoir equation (unitless)
K2	[0,0.05]	Slow reservoir coefficient which determines what proportion of the storage is released per day (unitless)

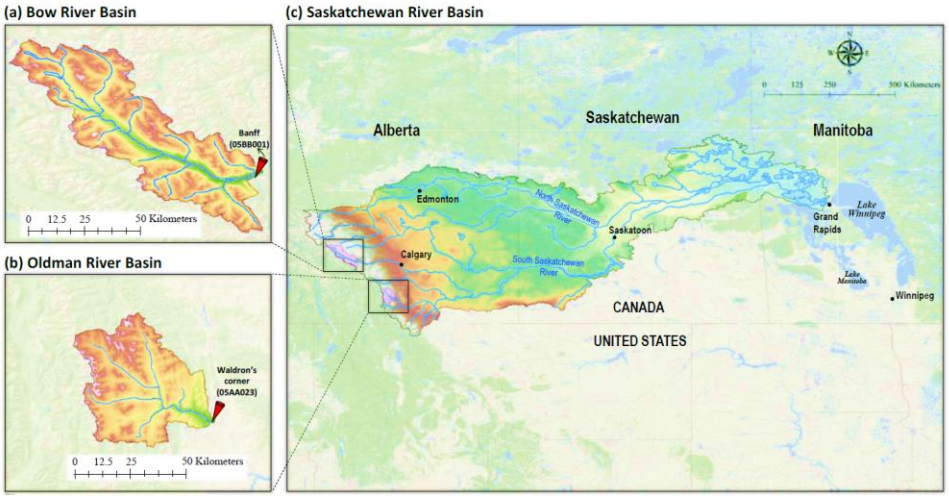


Figure 5-1 Oldman river basin located in the Rocky Mountains in Alberta, Canada, flows into the Saskatchewan River Basin (adapted from Razavi et al., 2019).

5.3.2 A land surface-hydrology model

In the second case study, we demonstrate the utility of the imputation-based methods in crash handling via their application to the GSA of a high-dimensional problem. The model used is Modélisation Environmentale– Surface et Hydrologie (MESH; Pietroniro et al., (2007)), which is a semi-distributed, highly-parameterized land surface-hydrology modelling framework developed by Environment and Climate Change Canada (ECCC) mainly for large-scale watershed modelling with consideration of cold region processes in Canada. MESH combines the vertical energy and water balance of the Canadian Land Surface Scheme (CLASS, Verseghy, 1991; Verseghy et al., 1993) with the horizontal routing scheme of the WATFLOOD (Kouwen et al., 1993). We encountered a series of simulation failures while assessing the impact of uncertainties in 111 model parameters (see **Appendix**) on simulated daily streamflows in Nottawasaga river basin in Ontario, Canada (Fig. 3).

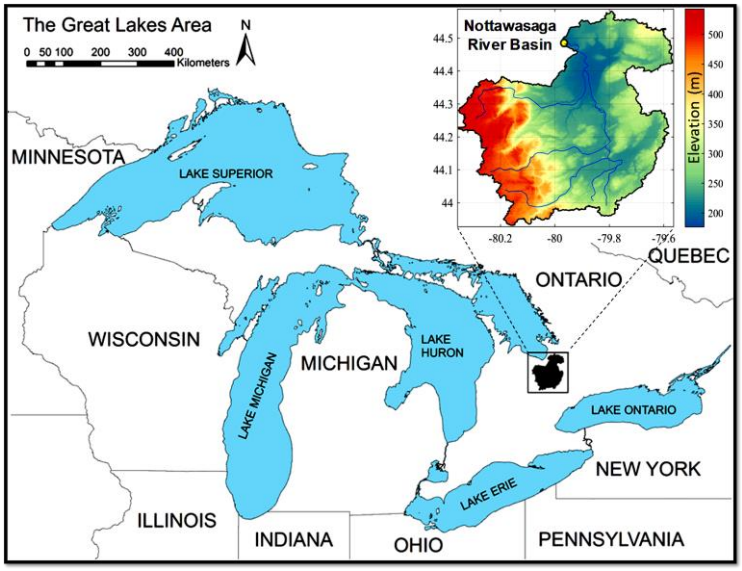


Figure 5-2 Nottawasaga river basin in in Southern Ontario, Canada (adapted from Sheikholeslami et al., 2018).

5.3.3 Experimental setup

For the first case study, we ran the HBV-SASK model with 9,100 randomly selected parameter sets from the feasible ranges of **Table 1** generated by the STAR-VARS (100 star centers with a resolution of 0.1). The Nash-Sutcliffe metric on streamflows (NS) has been used as the model output for sensitivity analysis. After calculating the NS values, we ran a series of experiments each with a different assumed "ratio of failure" (from 1% to 20%), defined as the percentage of failed parameter sets to the total number of parameter sets. In each experiment, we randomly chose a number of sampled points based the associated ratio of failure and assume them as simulation failures (hypothetical failures). Then, we evaluated the performance of the crash handling strategies to replace simulation failures during GSA of the HBV-SASK model and compared the results with the case when there are no failures. Also, we accounted for the randomness in the comparisons by carrying out 50 replicates of each experiment with different random seeds. This allowed us to see a range of possible performances for each strategy and to assess their robustness when crashes occurred at different locations in the parameter space.

In the second case study with 111 parameters, 100 star centers were randomly generated using STAR-VARS algorithm with a resolution of 0.1, resulting in a total of 100,000 MESH runs. The

NS performance metric was used to measure daily model streamflow performance, calculated for a period of three years (October 2003-September 2007) following a one-year model warmup period. Due to various physical and/or numerical constraints inside MESH (or more precisely in CLASS), some combinations of the 111 parameters caused model crashes. Here, approximately 3% of our simulations failed (3,084 out of 100,000 runs). We applied the proposed crash handling strategies to infill the missing model outcomes in GSA of the MESH model.

5.4 Numerical Results

5.4.1 Results for the HBV-SASK model

According to the IVARS-50 sensitivity index, the VARS algorithm ranks (after 9,100 function evaluations) the parameters of the HBV-SASK in order of importance as follows *FRAC*, *FC*, *C0*, *TT*, *alpha*, *K1*, *LP*, *ETF*, *beta*, and *K2*, when there are no crashes (we consider the corresponding assessments to be "true"). Based on the dendrogram (**Fig. 5-3**) generated by the factor grouping algorithm introduced by Sheikholeslami et al., (2019), we categorized these parameters into three groups with respect to their importance, i.e., {*FRAC*, *FC*, *C0*} are the strongly influential parameters, {*TT*, *alpha*, *K1*} are moderately influential parameters, and {*LP*, *ETF*, *beta*, *K2*} are weakly influential parameters.

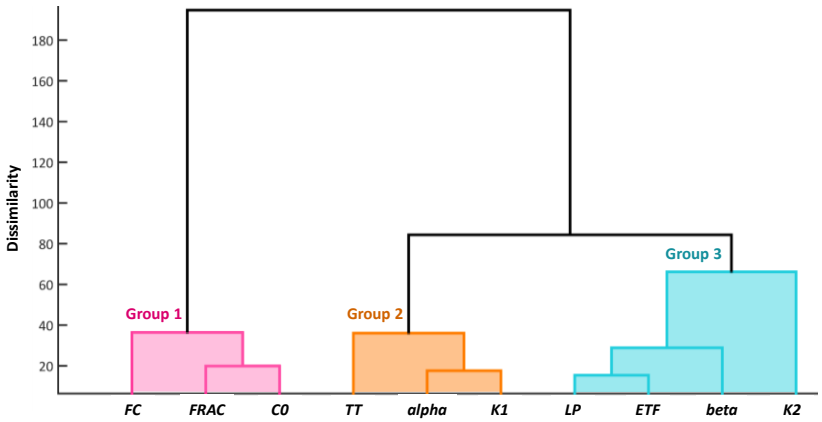
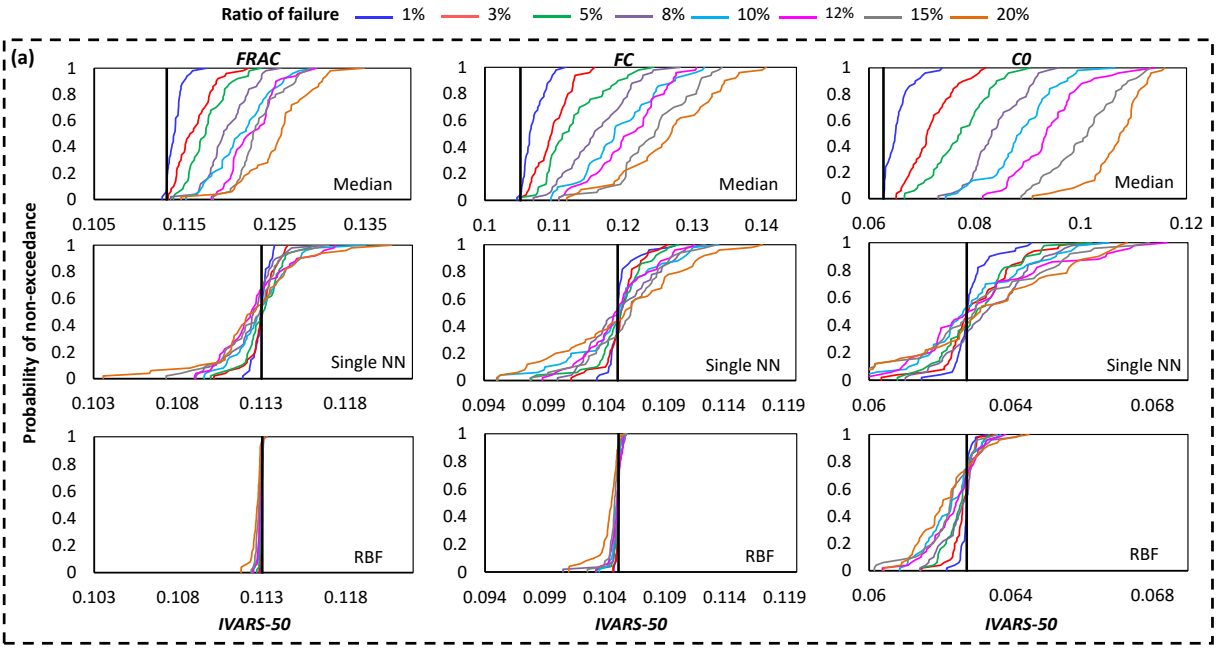


Figure 5-3 Grouping of the 10 parameters of the HBV-SASK model when applied on the Oldman River Basin. The parameters are sorted from the most influential (to the left) to the least influential (to the right).

Fig. 5-4 and **5-5** show cumulative distribution functions (CDFs) for the 50 independent estimates of IVARS-50, obtained when 1%, 3%, 5%, 8%, 10%, 12%, 15%, and 20% of model runs were deemed to be simulation failures. Overall, the RBF and single NN techniques outperformed the median substitution in terms of closeness to the true GSA results and robustness when crashes happened at different locations of parameter space. As can be seen, by increasing the rations of failure, the performance of the crash handling strategies, particularly the median substitution became progressively worse. Note that the median substitution technique resulted in a significant bias manifested through over-estimation of the sensitivity indices for all the parameters. From the results, we see that using the RBF technique the sensitivity indices of the most important parameters {*FRAC*, *FC*} (**Fig. 5-4(a)**) and less important parameters {*LP*, *ETF*, *beta*, *K2*} (**Fig. 5-5**) were estimated with a high degree of accuracy and robustness. However, for moderately influential parameters (**Fig. 5-4(b)**) its performance was reduced (i.e., the CDFs are wider in **Fig. 5-4(b)**).



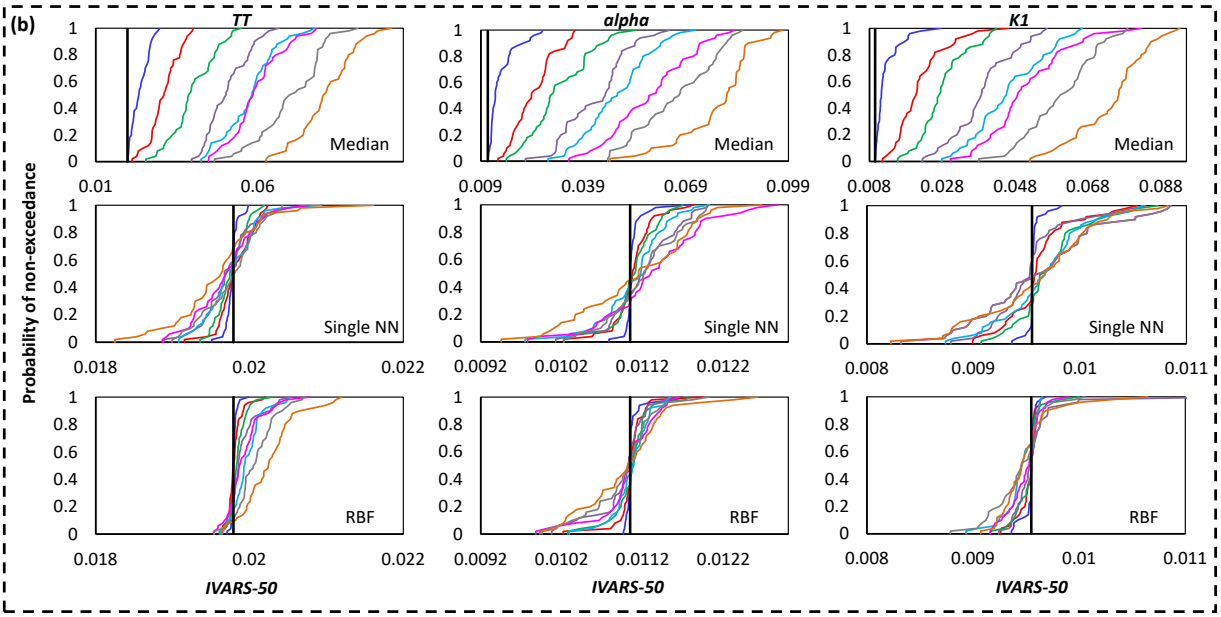


Figure 5-4 Comparison of the proposed crash handling strategies in sensitivity analysis of the HBV-SASK model using the STAR-VARS algorithm for different ratios of failures. The CDFs of the sensitivity indices for **(a)** strongly influential parameters {*FRAC*, *FC*, *C0*} (upper panel) and **(b)** moderately influential parameters {*C0*, *TT*, *alpha*, *K1*} (lower panel) are compared in this plot. The vertical line (solid black) on each subplot represents the corresponding ‘‘true’’ sensitivity index obtained when there were no failures.

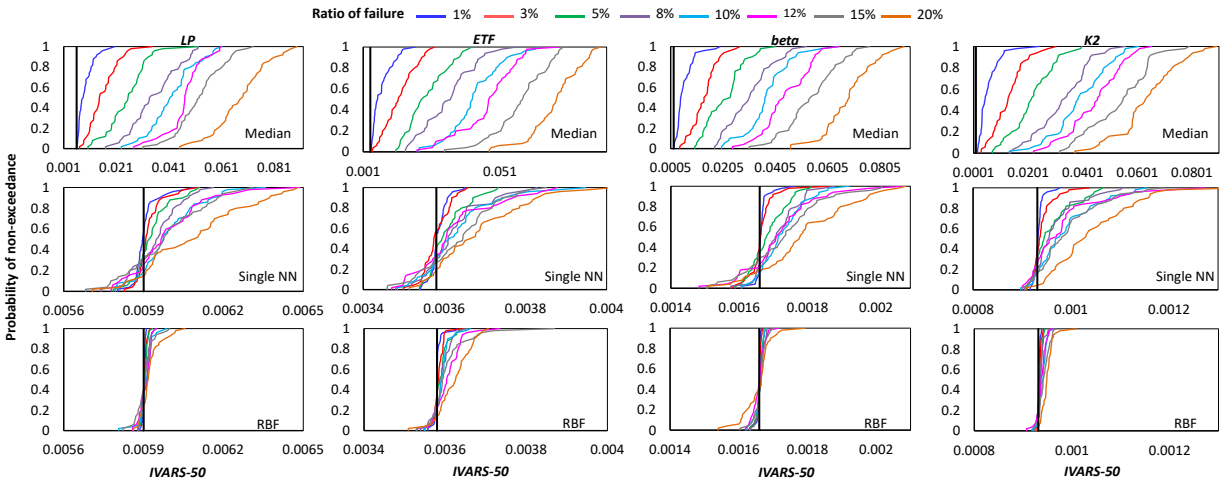


Figure 5-5 Comparison of the proposed crash handling strategies in sensitivity analysis of the HBV-SASK model using the STAR-VARS algorithm for different ratios of failures. The CDFs of the sensitivity indices for weakly influential parameters {*LP*, *ETF*, *beta*, *K2*} are shown in this plot. The vertical line (solid black) on each subplot represents the corresponding ‘‘true’’ sensitivity index obtained when there were no failures.

More importantly, as the number of crashes increases, ranking of the parameters in terms of their importance may change. For example, **Fig. 5-6** shows the number of times out of 50 independent runs that the rankings of the parameters were equal to the "true" ranking. In all 50 runs, regardless of the number of model crashes, the rankings obtained by VARS algorithm using the RBF technique were the same as the "true" ranking which is an indication of high degree of robustness in terms of parameter ranking. The performance of the single NN slightly reduced when the crash percentage were more than 15%, while the VARS algorithm wrongly determined the rankings in more than 50% percent of the replicates using median substitution technique (see **Fig. 5-6c** and **d**). This highlights that the rankings can be estimated much more accurately than the sensitivity indices in the presence of simulation crashes. Also, it can be seen that while the RBF-based strategy performed perfectly in this example, the performance of the single NN technique was comparably well.

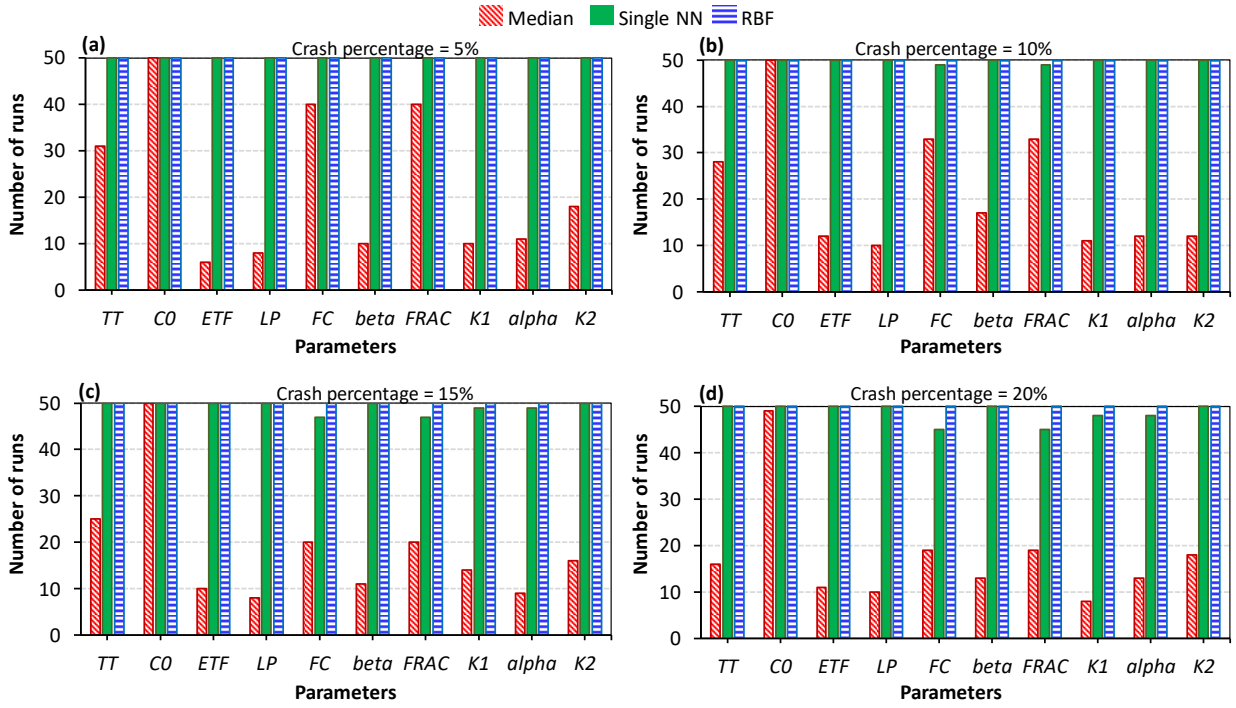


Figure 5-6 Comparison of the crash handling strategies in estimating the parameter rankings for HBV-SASK model when **(a)**5%, **(b)**10%, **(c)**15%, and **(d)**20% of model runs were simulation crashes. The *y*-axis in each subplot shows the number of times out of 50 replicates that the rankings of the parameters are equal to the true ranking.

Finally, **Fig. 5-7** presents the performance of the single NN (**Fig. 5-7a**) and RBF (**Fig. 5-7b**) strategies in approximating the fill-in values for the missing responses when 20% of HBV-SASK simulations were deemed to be failures. As shown, the RBF outperformed single NN technique in terms of closeness to the true NS values. The linear regression has an R^2 value of 0.834 when single NN was used, while the RBF strategy achieved a linear regression with an R^2 value of 0.996. Also, the result of the RBF strategy is almost unbiased as the linear regression plotted on **Fig. 5-7b** is very close to the ideal (1:1) line.

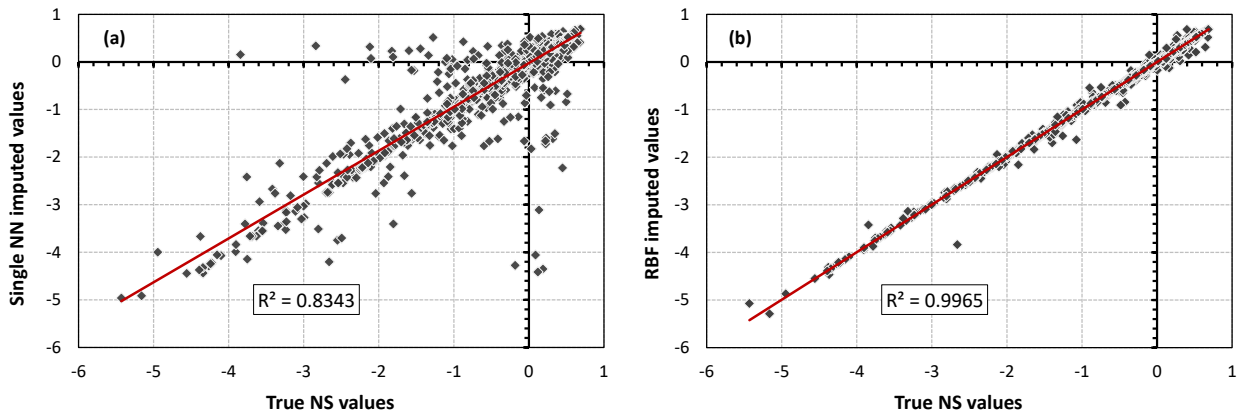


Figure 5-7 Scatter plots of the true NS values versus the imputed NS values when 20% of the HBV-SASK model simulations were deemed as model crashes. The accuracy of crash handling techniques is demonstrated in subplot **(a)** for the single NN method and in subplot **(b)** for the RBF method. These results belong to one replicate (arbitrarily chosen) out of 50 independent runs.

5.4.2 Results for the MESH model

Here we demonstrate the GSA results by categorizing the 111 MESH model parameters into three groups as shown in **Fig. 5-8** (for more details on grouping see Sheikholeslami et al. (2019)). **Fig. 5-9** to **5-11** present the sensitivity analysis results obtained by the VARS algorithm for the MESH model, when different crash handling techniques were applied. These groups were labeled according to their importance, i.e., **Group 1** (**Fig. 5-9**) contains the strongly influential parameters, while parameters in **Group 2** (**Fig. 5-10**) are moderately influential, and **Group 3** (**Fig. 5-11**) is the group of weakly influential parameters.

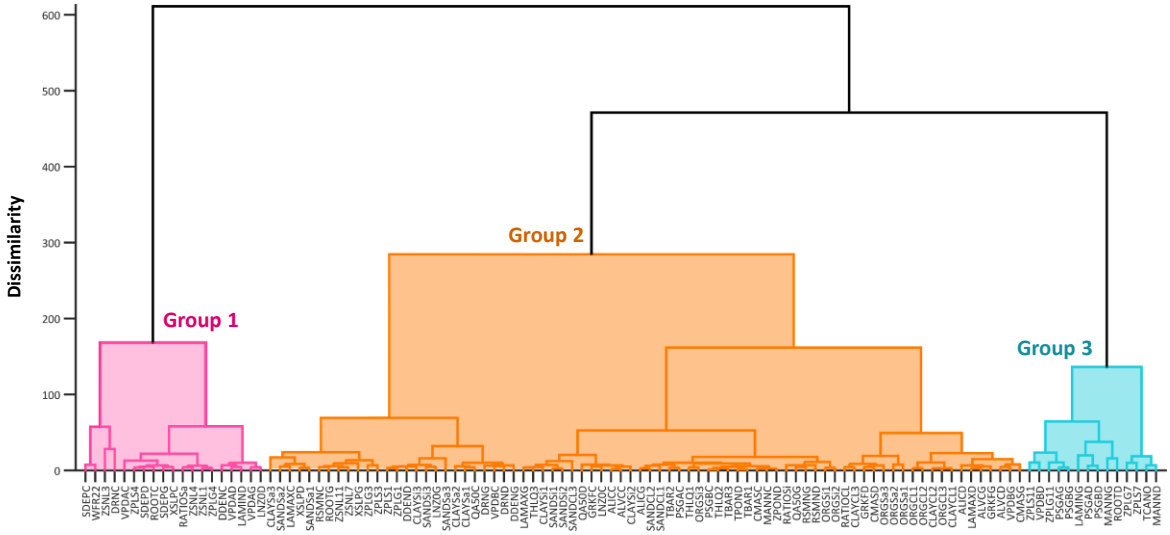


Figure 5-8 Grouping of the 111 parameters of the MESH model. The parameters are sorted from the most influential (to the left) to the least influential (to the right). This grouping is based on the results of the RBF method.

Four most strongly influential parameters in **Group 1** are *SDEPC* and *DRNC* ("C" stands for crops in this case study) controlling water storage and movement in the soil, *WFR22* (river channel routing), and *ZSNL* (snow cover fraction). As shown in **Fig. 5-9** (upper panel), the sensitivity indices associated with these parameters are similar regardless of the employed crash handling technique. It is worth mentioning that, as discussed in our failure analysis (see **Section 5.5**), we also identified three of these parameters (i.e., *SDEPC*, *DRNC*, and *ZSNL*) responsible for at least some of the model crashes. In other words, the parameters which strongly contribute to the variability of the MESH model output can also be convicted of model crashes. To enhance the future development and application of the MESH model, it is necessary that more efforts should be done to better understand the functioning of these parameters and their effects acting individually or in combination with other parameters over their entire range of variations.

For the other 15 influential parameters in **Group 1** (**Fig. 5-9**, bottom panel), there is general agreement with three crash handling techniques about the sensitivity indices calculated by VARS except for the parameter *ROOTC* which defines the annual maximum rooting depth of vegetation category. The RBF and median substitution methods give more importance to *ROOTC* compared

to the single NN technique. It is noteworthy that the oversaturation of soil layer, which can cause many model runs to fail, is subject to the interaction between *ROOTC* and *SDEPC*.

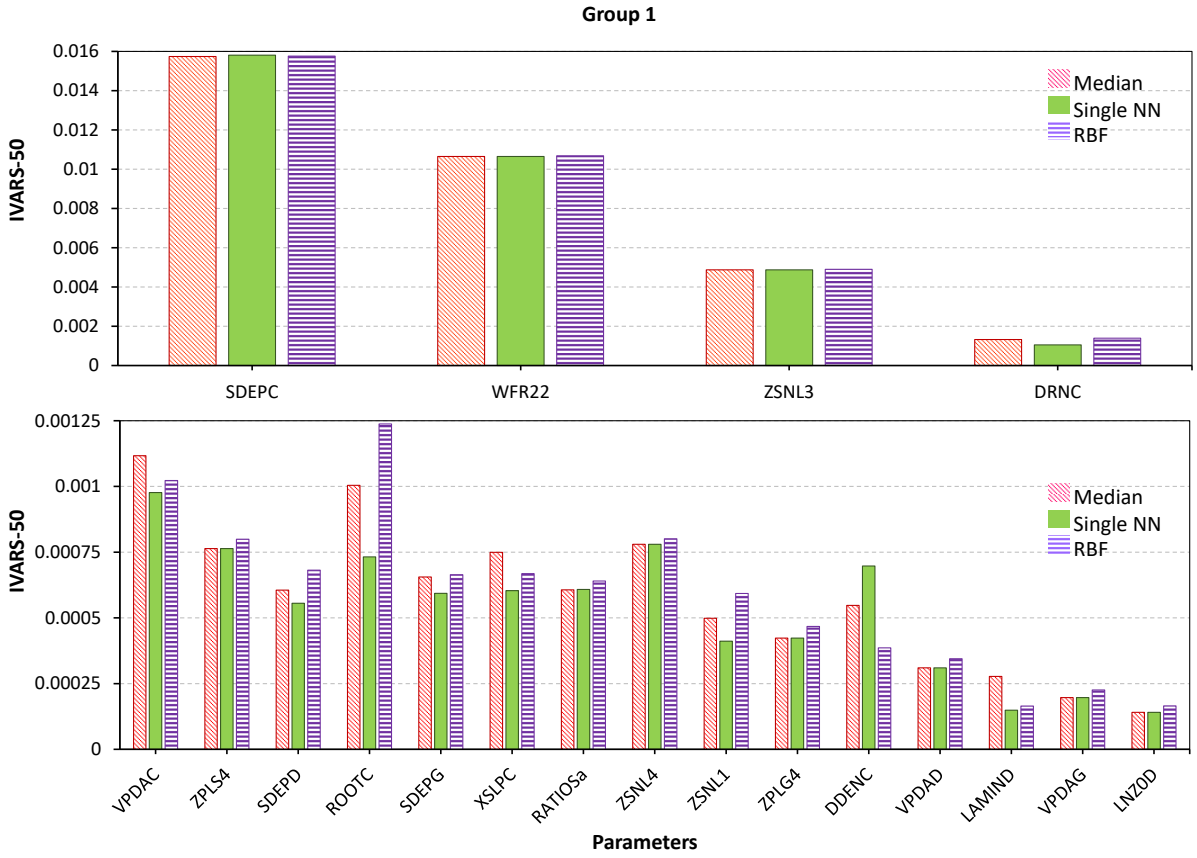


Figure 5-9 Sensitivity analysis results of the MESH model using different crash handling strategies for the most influential parameters. To better illustrate the results, the highly influential parameters in **Group 1** are separately shown in two subplots.

Fig. 5-10 illustrates the sensitivity indices for the moderately influential parameters (i.e., **Group 2**). Note that for all these 78 parameters the sensitivity analysis results were highly dependent on the chosen crash handling strategy. As can be seen, the sensitivity indices associated with the median substitution and RBF techniques are higher than those obtained by the single NN technique (this difference is considerable for the parameters in upper and lower subplots than those in middle subplot).

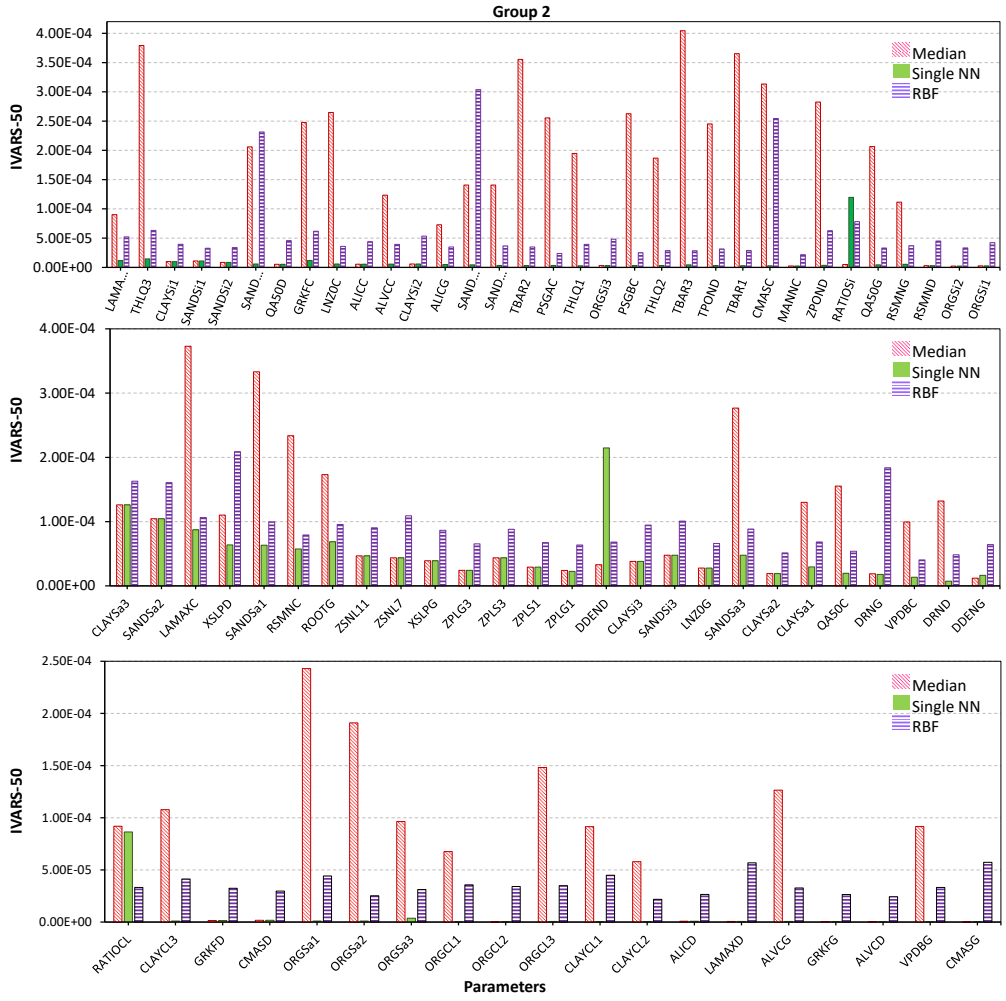


Figure 5-10 Sensitivity analysis results of the MESH model using different crash handling strategies. To better illustrate the results, the moderately influential parameters in **Group 2** are separately shown in three subplots.

Finally, the results of the sensitivity analysis for the weakly/non-influential (**Group 3**) parameters of the MESH model are plotted in **Fig. 5-11**. As shown, although the VARS algorithm identified these parameters as minimally-influential (very low IVARS-50 values) using the proposed crash handling techniques, the associated sensitivity indices obtained by the RBF imputation method are about two order of magnitude larger for the parameters in the left panel (**Fig.11** (**a**, **c**)) and about four order of magnitude larger for the parameters in the left panel (**Fig. 11** (**b**, **d**)) than compared to those obtained by the single NN and median substitution methods.

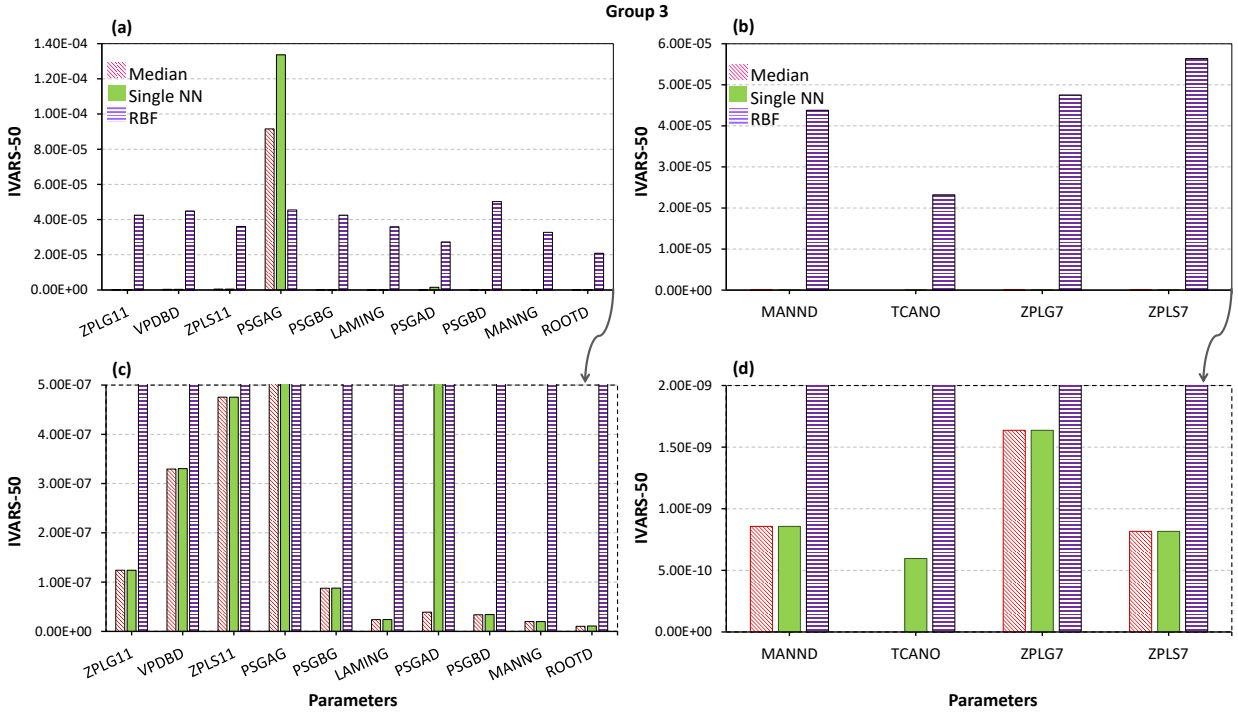


Figure 5-11 Sensitivity analysis results of the MESH model using different crash handling strategies for weakly/non- influential parameters in **Group 3**. The bottom panel (**c** and **d**) shows a zoom-in of the top subplots for very small values on the vertical axis.

However, it is important to note that in high-dimensional CESMs, when the number of parameters is very large, the estimation of sensitivity indices is likely to not be robust to sampling variability. On the other hand, parameter ranking (order of relative sensitivity) is often more robust to sampling variability and converges more quickly than factor sensitivity indices (see e.g., Vanrolleghem et al., 7 2015; Razavi and Gupta, 2016b; Sheikholeslami et al., 2018). To investigate how different crash handling strategies can affect the ranking of the model parameters in terms of their importance, **Fig. 5-12** compares the rankings obtained by the RBF, single NN, and median substitution techniques.

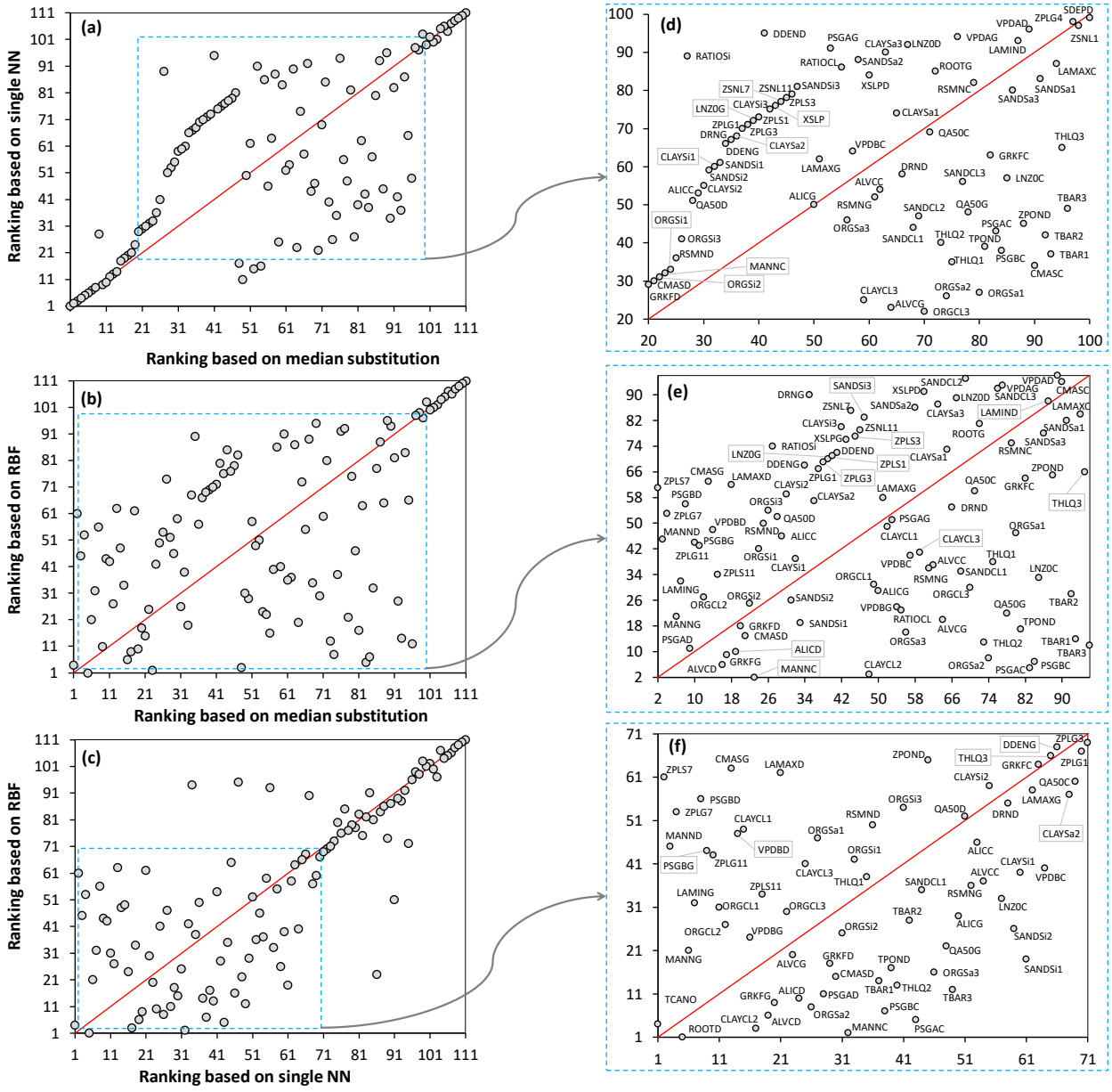


Figure 5-12 Plots comparing rankings of the MESH model parameters obtained by different crash handling strategies. Subplots (**d**), (**e**), and (**f**) (right column) show a zoom-in of the subplots (**a**), (**b**), and (**c**) (left column), respectively. The red line is the ideal (1:1) line. Note that a ranking of 1 represents the least sensitive and a ranking of 111 represents the most sensitive parameter.

As shown in **Fig. 5-12a**, the single NN and median substitution techniques resulted in almost similar parameter rankings for the (strongly) influential (**Group 1**) and minimally-influential (**Group 3**) parameters, while for moderately influential parameters (**Group 2**) the rankings are

significantly different. Meanwhile, the RBF and median substitution techniques yielded very distinctive rankings except for the (strongly) influential parameters (**Fig. 5-12b**). Furthermore, **Fig. 5-12c** indicates that the single NN and RBF method give similar rankings for most sensitive parameters.

A closer examination, however, reveals that rankings can be very contradictory for some of the parameters, when using different crash handling strategies (see **Fig. 5-12(d-f)**). For example, consider the soil moisture suction coefficient for crops (*PSGAC*) which is used in calculation of the stomatal resistance in the evapotranspiration process of the MESH (for more details see Fisher et al., 1981; Choudhury and Idso 1985; Verseghy, 2012). As can be seen, according to the RBF method, *PSGAC* is one of the low-sensitivity parameters (ranked 5th), while using the single NN it is determined to be one of the medium-sensitivity parameters (ranked 43rd). In contrast, it is one of the high-sensitivity parameters based on the median substitution (ranked 83rd). However, in a comprehensive study of the MESH model using various model configurations and different hydroclimatic regions in Eastern and Western Canada, Haghnegahdar et al. (2017) found that *PSGAC* is one of the least sensitive parameters considering three model performance criteria with respect to high flows, low flows, and total flow volume of the daily hydrograph. As another example, consider *ZPLS7* (maximum water ponding depth for snow-covered areas) and *ZPLG7* (maximum water ponding depth for snow-free areas) which are used in surface runoff algorithm of the MESH (i.e., PDMROF). The single NN and median substitution methods both ranked *ZPLS7* as 2nd and *ZPLG7* as 3rd least sensitive parameters, whereas the RBF ranked them as 61 and 45 (i.e., medium-sensitivity) which is in accordance with results reported by Haghnegahdar et al. (2017).

5.5 Discussion

5.5.1 Potential causes of failure in MESH

Considering the existing difficulties in failure analysis, however, our further investigations of the MESH model revealed at least two possible causes responsible for many of the simulation

failures. First, we observed that the threshold behavior of a parameter called *ZSNL*, which represents the snow depth threshold below which snow coverage is considered less than 100%, can cause many model crashes. When *ZSNL* was relatively large, it resulted in calculation of overly thick snow columns inside the model violating the snow energy balance constraints there and triggering a simulation abort. This situation became more severe when the calculated snow depth was invariantly larger than the maximum vegetation height(s) depending on their assigned values via parameter perturbations. **Fig. 5-13** (left column) shows the scatterplots of *ZSNL* values sampled from the feasible ranges for all model simulations used for GSA of MESH, with failed designs marked by red dots. To alleviate this issue, a strategy may be to reduce the upper bound of the *ZSNL* parameter to lower values as used by Haghnegahdar et al. (2017) or to fix *ZSNL* at a lower value of, for example, 0.1 m as suggested by CLASS manual (Verseghy, 2012).

We also found that the second reason responsible for the MESH failure was oversaturation of the soil layers. Our investigations revealed that this oversaturation can happen especially at lower values of the soil permeable depth (*SDEP*) and when it becomes less than the maximum vegetation rooting depth (*ROOT*). The situation is more severe when the soil drainage index (*DRN*) is also reduced (all these three parameters are part of the 111 perturbed parameters in here). These interactions can collectively cause a thinner soil column for water storage and movement that now has a lower chance for transpiration and drainage. This will result in over accumulation of the water beyond the physical limits set for the soil in the model, thus leading to simulation failures (this is evidently a numerical scheme problem associated with crossing boundary values). **Fig. 5-13** (right column) displays the scatterplots of these three parameters for the crop vegetation type. To avoid model crashes, it is necessary to ensure that *SDEP* and *ROOT* values are not unrealistically low and that their values and/or their ranges are assigned as accurately as possible using available data as discussed in Haghnegahdar et al. (2017). Also, fixing *DRN* to 1, may allow for the maximum physically-meaningful drainage from the soil column and reduces the risk of oversaturation.

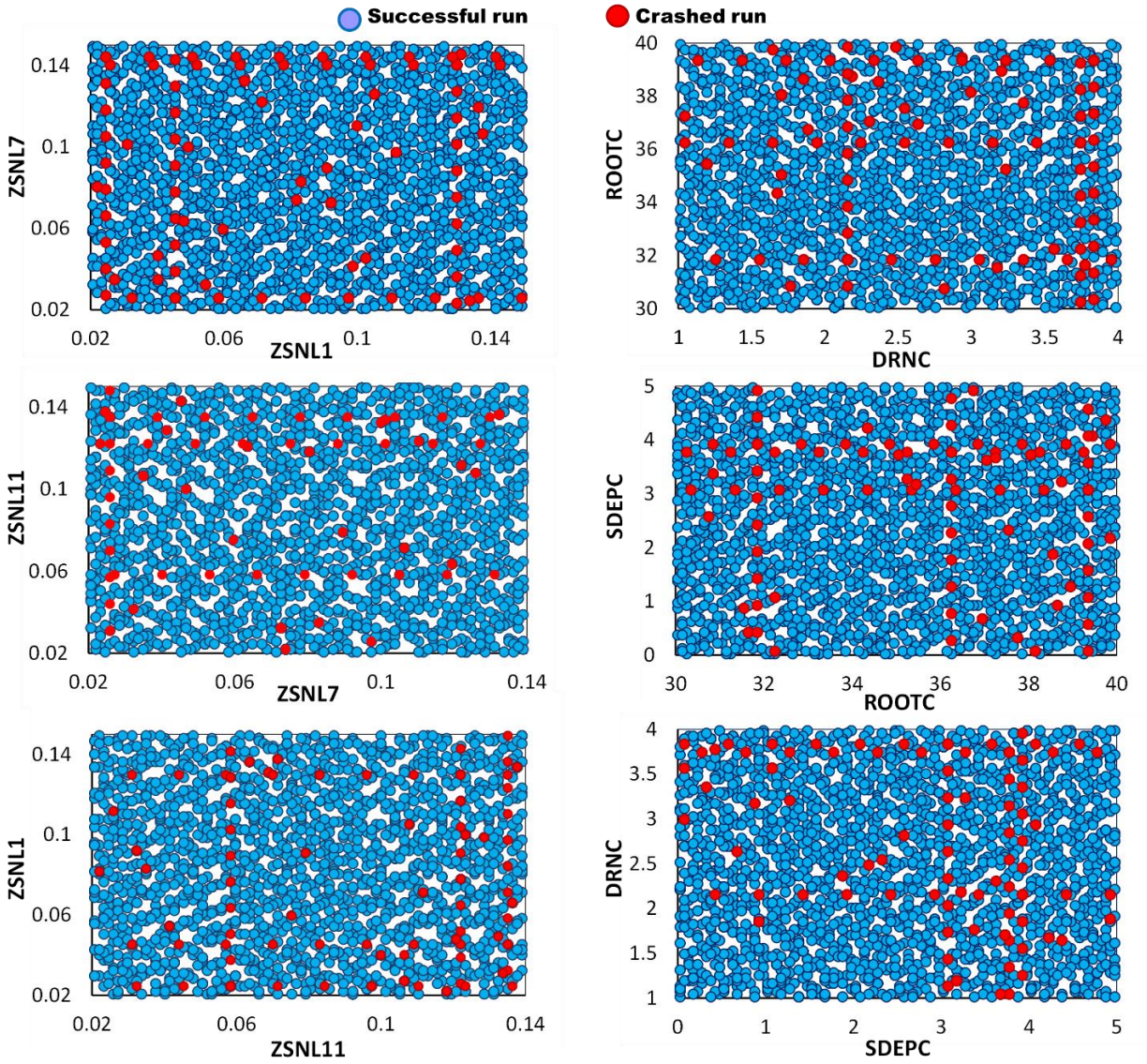


Figure 5-13 A 2-D projections of the MESH parameters for successful (blue dots) and crashed (red dots) simulations. Left column shows the threshold snow depth parameters *ZSNL* and right columns shows soil permeable depth (*SDEP*), maximum rooting depth (*ROOT*), and drainage index (*DRN*) for crop vegetation type.

As can be seen from **Fig. 5-13**, very high values of parameters DRNC and SDEPC can also cause simulation crashes, while these crashes were happened at lower values of ZSNL7. Note that from these 2-dimensional projections of the 111-dimensional parameter space of the MESH no general conclusions can be drawn. This even becomes more complicated when

noticing some isolated crashes in regions where most of the simulations were successful. Furthermore, as shown in **Fig. 5-13**, there are considerable overlaps between successful simulations and crashed ones in the feasible ranges of parameters. For example, there are many crashed simulations when DRNC was sampled from [3.5-4], at the same time a high density of successful simulations can also be observed in the same range. This indicates that locating regions of parameter space responsible for crashes is difficult, if not impossible, and necessitates analyzing the MESH's response surface throughout a high-dimensional parameter space.

5.5.2 The role of sampling strategies in handling model crashes

Due to the extremely large parameter space ($\boldsymbol{X}$) of high-dimensional CESMs, it may require many properly distributed sample points ($\boldsymbol{X}_s$) to generate/explore a full spectrum of model behaviors such as simulation crashes, discontinuities, stable regions, optima, etc. Together with the computationally intensive nature of CESMs, this issue can make both non-substitution procedures and imputation-based methods (those proposed in the present study) very costly in dealing with crashes, if not impractical.

Because the non-substitution procedures rely on constructing a statistical model based on observed crashes to predict and avoid them in the follow-up experiments, they need a good coverage of the domain to attain a reliable statistical model. This issue also challenges the use of imputation-based methods. For example, in the NN techniques one major concern is that the sparseness of sample points may affect the quality of the results. In regions of the parameter space where sample points are sparsely distributed, distances to nearest neighbors can be relatively large, leading to choosing physically incompatible neighbors. Also, in response surface modelling-based techniques, building an accurate and robust function approximation is directly depends on the utilized sampling strategy and how dense mappings between parameter and output spaces are (see, e.g., Jin et al., 2001; Mullur and Messac, 2006; Zhaou and Xue, 2010).

A crucial consideration in the use of any sampling strategy is the ***exploration*** ability of that strategy (i.e., space-fillingness), which significantly influences the effectiveness of the utilized

crash handling approach. When having this feature enabled, the non-substitution procedures can reliably identify implausible regions in the entire parameter space, meaning that the sample set is not confined to only a limited number of regions. Furthermore, it can notably improve the predictive accuracy of the response surface modelling-based methods (Crombecq et al., 2011). Exploration requires sample points to be evenly spread across the entire parameter space to ensure that all regions of the domain are equally explored, and thus sample points should be located almost equally apart. This feature rectifies the problem relating to the distances between sample points when using NN techniques since in space-filling designs these distances are as even as possible.

Given this, regardless of the chosen method for solving simulation crash problem in GSA, it is advisable to spend some time up front to find an *optimal* sample set before submitting it for evaluation to the computationally expensive CESMs. It is, therefore, necessary to prudently use improved sampling algorithms such as Progressive Latin Hypercube Sampling (PLHS; Sheikholeslami and Razavi (2017)), Sequential Exploratory Experimental Design (SEED; Li (2004)), or Symmetric Latin Hypercube Design (SLHD; Ye et al. (2000)). Generally, these sampling techniques optimize some characteristics of the sample points such as sample size, space-fillingness, projective properties, and so on.

5.6 Conclusions

Understanding complex physical processes in Earth and environmental systems, prediction and scenario analysis regarding the Earth's future resources rely routinely on high-dimensional, computationally expensive models, typically comprising model calibration, and/or uncertainty and sensitivity analysis. If a simulation failure/crash occurs at any of these stages, these models will stop functioning, and thus need user intervention. Generally, there are many reasons for failure of a simulation in models, including those that come from an inconsistent integration time step or grid resolution, lack of convergence, and existing of model thresholds. Determining whether these "defects" exist in the utilized numerical schemes or they are programming bugs

can only be done through analysing a high-dimensional parameter space and characterizing implausible regions responsible for crashes. This imposes a heavier computational burden on analysts. More importantly, every "crashed" simulation can be very demanding in terms of computational cost for global sensitivity analysis (GSA) algorithms because they can prevent completion of the analysis and introduce ambiguity into the GSA results.

These challenges motivated us to implement three missing data imputation-based strategies for handling simulation crashes, which involves substituting plausible values for failed simulations without a priori knowledge regarding the nature of the failures. Here, our focus was to find simple yet computationally frugal techniques to palliate the effect of model crashes on the GSA of dynamical Earth systems models (DESMs). Thus, we utilized three techniques, including median substitution, single nearest neighbor, and emulation-based substitution (here we used radial basis functions as a surrogate model) to fill in a value for a failed simulation using available information and other non-missing model responses. Compared to other crash handling strategies (ignorance-based and non-substitution procedures), the efficiency of our proposed substitution-based strategy was shown to be remarkable, particularly when dealing with GSA of the computationally expensive models since this strategy does not need repeating the entire experiment again. We compared the performance of the proposed strategy in GSA of the two modelling case studies in Canada, including a 10-parameter HBV-SASK conceptual hydrologic model and a 111-parameter MESH land surface-hydrology model. Our analyses revealed that:

- Overall, the emulation-based substitution can effectively handle the simulation crashes and produce promising sensitivity analysis results compared to the single nearest neighbor and median substitution techniques.
- As expected, the performance of the proposed methods degrades as the ratio of failures increases. The rate of degradation is dependent on the number of model parameters (dimensionality of parameter space).
- We observed in our experiments that the rankings of strongly and weakly influential parameters identified by the utilized GSA algorithm (i.e., VARS) are not affected by the

chosen crash handling technique, whereas for the moderately influential parameters, different techniques yielded different rankings.

Furthermore, we conducted a failure analysis of the MESH model and identified the parameters that seem to be frequently casing model failures. Such analyses are helpful and much needed to improve the fidelity and numerical stability of DESMs and may constitute a promising avenue of research. In doing so, applying other advanced methods (see e.g., Lucas et al. (2013)) can be beneficial to diagnose existing defects of the complex models.

Future work should include extending the proposed crash handling strategy to time-varying sensitivity analysis of the DESMs because a comprehensive GSA requires a full consideration of the dynamical nature of the DESMs. Our proposed approach for handling simulation crashes can be integrated with any time-varying sensitivity analysis algorithm, for example, with the recently developed Generalized Global Sensitivity Matrix (GGSM) method (Gupta and Razavi, 2018). This further helps understand the temporal variation of the parameter importance and model behavior.

Author contributions

RS and SR designed the method and experiments. RS developed the MATLAB codes for the proposed crash handling strategy, conducted all the experiments, and analyzed the results. The simulations for the first case study were carried out by RS. AH performed the MESH simulations. RS wrote the manuscript with contributions from SR and AH. All authors contributed to structuring and editing of the paper.

Chapter 6
Conclusions and Future Directions

6.1 Summary of Dissertation Outcomes

This dissertation proposed three advanced efficiency-increasing strategies for Global Sensitivity Analysis (GSA) of Complex Environmental Systems Models (CESMs), including (1) a new sampling algorithm called PLHS; (2) an automated factor grouping method; and (3) efficient techniques for handling simulation failures occurred during GSA. These novel strategies were assessed using both analytical test functions and real-world case studies, and the results revealed that they facilitate sensitivity analysis of high-dimensional and computationally intensive CESMs by alleviating the computational burden associated with GSA and monitoring the performance of the GSA in terms of robustness and convergence. Although their usefulness has been demonstrated in the context of watershed modelling, these strategies can be further used to resolve a range of decision making-related problems such as (1) characterizing the main causes of risk; (2) exploring the CESMs' sensitivity to a wide range of plausible future changes; and (3) assessing the sensitivity of the ranking of alternatives obtained from multi-criteria decision analysis.

The main contributions of this dissertation were explicitly stated in **section 1.3**. In the following, the achievements and outcomes of the research in this dissertation are summarized:

Chapter 2 reviewed the existing sampling strategies and their shortcomings when applied to sampling-based analysis of CESMs. A new sampling strategy called Progressive Latin Hypercube Sampling (PLHS) was proposed to overcome these challenges. The numerical results showed that PLHS enables improved characterization of the model input spaces and output spaces (i.e., response surfaces) with less computational budget (smaller sample size), compared to other sampling strategies. Moreover, PLHS provided improved convergence rate and increased robustness to sampling variability and randomness when used in uncertainty and

sensitivity analysis context, compared to other sampling strategies. PLHS can help avoid over- or under-sampling by enabling users to monitor the performance of the associated sampling-based analysis as the sample size grows.

Chapter 3 presented a comparative performance investigation of the two GSA methods: (1) the recently proposed Variogram Analysis of Response Surfaces (VARS) and (2) the widely-used Regional Sensitivity Analysis (RSA) using the PLHS strategy when applied to a hydrodynamic river ice model. Results revealed that (1) the water levels simulated by the river ice model are most sensitive to the ice cover characteristics (i.e., porosity and thickness at the ice cover front) and upstream discharge; (2) the hydraulic roughness parameters and slush ice properties (i.e., porosity and thickness of the slush pans) are medium- and low-sensitivity parameters, respectively; (3) the VARS and RSA methods provide contradictory assessments regarding the sensitivity of the model output to variations in the slush ice porosity and ice roughness parameters; and (4) the VARS method appears to be superior to RSA in terms of generating robust estimates of the parameter sensitivity rankings.

Chapter 4 discussed the curse-of-dimensionality challenge associated with GSA of high-dimensional problems, reviewed existing strategies for factor grouping, and discussed their limitations. A new factor grouping algorithm was developed and the details of its implementation was explained. In addition, to evaluate performance of the grouping-based GSA, a new measure of robustness was introduced, providing a new way to monitor convergence of the GSA results. Overall, the results confirmed that the strategy of grouping factors can significantly reduce the computational effort required to perform GSA on high-dimensional models (in the case of the MESH model by as much as ~50%). This is mainly because factor groupings typically converge faster than factor sensitivity indices. A further potential benefit is that the proposed algorithm can provide information useful for reducing the complexity of the problem in follow-up experiments (e.g., model calibration or model order reduction) by identifying dominant/influential groups of factors that significantly contribute to the variability in the model outputs.

Chapter 5 identified four main approaches to handle simulation crashes in sampling-based analysis of CESMs and discussed their drawbacks. To tackle these problems, three practical remedies (i.e., median substitution, single nearest neighbor substitution, and response surface modelling) were applied to circumvent parameter-induced crash problem in GSA of high-dimensional CESMs. Overall, the results confirmed that the response surface modelling strategy can effectively handle the simulation crashes and produce promising results compared to other techniques. However, its performance may degrade as the ratio of failure becomes larger. Moreover, the results revealed that a high percentage of failed simulations can lead to wrong inferences drawn from the GSA results. By increasing the dimensionality of the model this issue becomes more critical where the GSA results can be completely misleading even with a small number of model crashes. Furthermore, a failure analysis of the MESH model was conducted, and accordingly the parameters that seem to be frequently causing model failures were identified. The use of proposed techniques in handling simulation failures when performing GSA can provide significant time savings over other methods such as non-substitution procedures.

6.2 Scope for Future Work

6.2.1 Thoughts and reflections on GSA of CESMs

In this section, I provide some thoughts, reflections, and guidance on a number of critical issues that have to be addressed by researchers and practitioners to conduct a comprehensive sensitivity analysis.

1) Specification of an objective function for sensitivity analysis

The existing misuse of GSA methods in hydrologic modelling community, and the subsequent misunderstanding and misinterpretation of the sensitivity analysis results are mainly due to the fact that typically some error metrics (e.g., MSE, NSE, etc.), which aggregate model performance by measuring discrepancy between model outputs and observations, have been often used in GSA studies as objective functions. Recently, Gupta and Razavi (2018) analytically showed that when using such objective functions, importance of each sensitivity coefficient term

can be significantly deteriorated by a residual term, leading to a serious bias in the sensitivity analysis results. Of course, the use of an error metric for sensitivity analysis provides useful information for model calibration but this is a form of identifiability analysis rather than a sensitivity analysis. As argued by Gupta and Razavi (2018), model sensitivity analysis is different from model identifiability analysis in a sense that sensitivity analysis is basically a "forward" problem, whereas identifiability analysis belongs to the class of "inverse" problems. In other words, model factor identification depends on the sensitivity of model outputs to perturbations of the factors, but the opposite case is not true. In the forward sense, the goal is to understand how model outputs are sensitive to the perturbations in the factors.

Therefore, much more attention should be directed to the selection of target model response(s) (model outputs) or objective functions for GSA in future studies. Considering the modeler's needs, the objective functions should sufficiently reflect the intended physical characteristics of the CESMs such as overall water balance (e.g., runoff ratio), behavior of long-term baseflow (e.g., total volume of low-flow), discharge seasonality (e.g., timing of snowmelt-induced spring runoff), etc. In this manner, GSA will further help model diagnostic analyses (Gupta et al., 2008; Yilmaz et al., 2008). Since the strategies developed in this research are independent of the objective function, model, or the GSA method used, they can be effectively integrated in sensitivity analysis for diagnostic testing of CESMs.

2) Time-varying parameter sensitivity analysis

CESMs typically generate time-dependent model simulations. For such dynamical model outputs, it may not be informative to conduct sensitivity analysis only on a scalar objective function, for example, by aggregating into some statistical measures. Temporal sensitivity analysis can identify time periods in which a certain parameter (or a set of parameters) has the highest influence on the model outputs, thereby providing greater insights into the modeled physical processes (Herman et al., 2013; Pfannerstill et al., 2015; Razavi and Gupta, 2019). As such, by performing sensitivity analysis on the model output at each time step valuable information on time-dependent nature of the sensitivities can be gained.

In practice, however, dynamic CESMs usually comprise high-dimensional, multivariate time series which will result in a time-consuming GSA. Therefore, when performing time-varying GSA, it is profitable to use the efficiency-increasing strategies that have been proposed throughout this dissertation.

3) Accounting for multi-output nature of models in sensitivity analysis

A major drawback of previous sensitivity analysis studies in the context of environmental modelling and particularly in hydrological modelling, regardless of the utilized GSA method, was their failure to properly consider the multi-output nature of the models (Klepper, 1997; Gupta et al., 1998; Rosolem et al., 2012; Haghnegahdar et al., 2017). Unlike the calibration procedure where modelers typically tend to use multiple objective functions, the sensitivity analysis has often been carried out separately for each objective function. However, to conduct a comprehensive GSA, it is necessary to fully consider the multi-output nature of CESMs. Our proposed methods can, in principle, be used in conjunction with multi-criteria sensitivity analysis to study the properties of parameters while recognizing multivariate aspects of the model behavior, thereby better supporting model development and understanding.

For example, one possible option for implementing the proposed factor grouping is to use multi-criteria GSA techniques which measure the global contribution of parameters to multivariate outputs through estimating generalized sensitivity indices. Having generated the sensitivity matrix consisting of these generalized indices, the grouping algorithm can use cluster analysis to identify distinct groups of parameters. GSA methods such as MOGSA (multi-objective generalized sensitivity analysis) method (Bastidas et al., 1999; Liu et al., 2004), which is an extension of regional sensitivity analysis, and the generalization of the variance-based Sobol GSA for multivariate outputs (Lamboni et al., 2011; Gamboa et al., 2014) may be useful in this regard. The aforementioned methods apply bootstrapping to ensure statistical robustness of the generalized sensitivity indices, and accordingly can yield robust groups of parameters.

6.2.2 Further research

There are some other issues that should be considered but were left out of the present work. Future research informed by this dissertation may include:

- The efficiency-increasing strategies developed in this research have been tested on case studies in the context of watershed modelling. Future work may include application and testing of these strategies on other CESMs developed for modelling water quality, landscape evolution, ecology, and erosion and sediment transport.
- The efficiency-increasing strategies proposed in this dissertation have been demonstrated to be effective for GSA of CESMs by coupling it with the VARS methodology. It would be interesting to test their effectiveness by integrating them to other advanced GSA frameworks.
- The sampling strategy developed in this dissertation (PLHS) has great potential to be used in solving problems in engineering design optimization and constructing surrogate models. In this regard, PLHS can facilitates the fast and effective analysis of high-dimensional models and can help achieve satisfying design solutions with very small numbers of function evaluations of the computationally intensive models.
- There are still many aspects of the proposed strategies that can be improved or modified. By way of example, work is needed on (1) applying other types of metamodels (e.g., kriging, support vector machine, ANN, etc.) to handle simulation failures during GSA; (2) incorporating other clustering algorithms into the proposed factor grouping strategy; and (3) employing other efficient optimization techniques in the process of building PLHS.

6.3 Software Availability

To promote best practices in sensitivity analysis, the MATLAB codes for the proposed strategies is included in the VARS-TOOL software package, which is a set of programs for next generation sensitivity and uncertainty analysis. VARS-TOOL is a multi-approach toolbox designed for comprehensive sensitivity analysis of high-dimensional problems (see Razavi et al. (2019) for

more details). This software is freely available for noncommercial use upon request from the author and can be downloaded from http://vars-tool.com/.

Appendix

In this **Appendix**, the parameters of the MESH model are described in detail. This case study was adopted from Haghnegahdar et al. (2015) where the MESH model was calibrated to the Nottawasaga river basin in Southern Ontario, Canada (see **Fig. 4-5**). MESH treats a watershed as being discretized into grid cells and accounts for within pixel heterogeneity using the concept of Grouped Response Units (GRUs, Kouwen et al., 1993). Here, the drainage basin of nearly 2700 km^2 is discretized into 20 grid cells with a spatial resolution of 0.1667 degrees (~15 *km*). The dominant land cover in the area is cropland followed by deciduous forest and grassland. The dominant soil type in the area is sand followed by silt and clay loam. Sixteen GRU types are formed by combining land cover and soil types in the region.

In this case study, many parameters are tied to a GRU type, and the same set of parameters is assigned for distinct GRU types delineated in each basin. Moreover, all parameters associated with the GRU type that represents the dominant land cover and soil type in the river basin are included in GSA. These GRUs are formed by adding the soil spatial data to the land cover classes consisting of clay loam, clay, gravelly, impermeable, organic, rock, sandy and silty. Then, for both land cover and soil type classes, the ones with less than 10% of area coverage were classified as the dominant class.

Finally, the remaining land cover and soil classes were merged to construct 16 new GRU types shown in **Table I**. As a result, there are a total of 111 parameters in this case study. **Table II** lists the MESH parameters and their feasible ranges. The first 11 parameters are GRU-dependent. The last parameter (channel roughness) is linked to river class types. Note that Nottawasaga River basin contains only one river class type, and only a single channel roughness is used. Furthermore, the 7 groups of parameters for the MESH model (see **Fig. 4-9**) are listed in **Table III.** The description of parameters and their feasible ranges can be found in Haghnegahdar et al.

(2015, 2017). MESH version 1.3.006 was implemented in this study. This case study (data, model setup, etc.) was included in the VARS-TOOL software package (Razavi et al., 2019) and can be downloaded from http://vars-tool.com/.

Table A-1 16 GRU types ranked by coverage area

GRU no.	***GRU type***	***Area covered (%)***
1	Cropland, sandy	36.3
2	Cropland, silty	18.3
3	Deciduous forest, sandy	8.7
4	Grassland, sandy	7.4
5	Cropland, clay loam	7.3
6	Mixed Forest, sandy	4.2
7	Cropland, organic	4.1
8	Deciduous forest, silty	3.9
9	Grassland, silty	3.7
10	Deciduous forest, organic	2.0
11	Mixed forest, organic	1.3
12	Mixed forest, silty	1.2
13	Grassland, clay loam	0.7
14	Deciduous forest, clay loam	0.5
15	Grassland, organic	0.3
16	Mixed forest, clay loam	0.1

Table 0-2 MESH model parameters and their feasible ranges

Parameter	***Description***	***(Lower bound, upper bound)***
ROOT	Annual maximum rooting depth of vegetation category (m)	(0.2, 1.0), (1, 3.5) for deciduous forest
RSMN	Minimum stomatal resistance of vegetation category (s m^{-1})	(60, 110) crop, (75, 125) grass and (100, 150) deciduous forest
VPDA	Vapour pressure deficit coefficient (used in stomatal resistance formula)	(0.5, 1)
SDEP	Soil permeable (Bedrock) depth (m)	(0.35, 4.10)
DDEN	Drainage density (km/km^2)	(2, 100)
SAND	Percent sand of all soil layers (%)	(0, 100), In 16-GRU scheme: (30, 65) for clay loam, (85, 100) for sandy soil and (0, 20) for silty soil
CLAY	Percent clay of all soil layers (%)	(0, 100), In 16-GRU scheme: (30, 40) for clay loam, (0, 10) for sandy soil and (0, 15) for silty soil
RATIO	The ratio of horizontal to vertical saturated hydraulic conductivity	(2, 100)

ZSNL	Limiting snow depth below which coverage is <100% (m)	(0.05, 1)
ZPLS	Maximum water ponding depth for snow-covered areas (m)	(0.02, 0.15)
ZPLG	Maximum water ponding depth for snow-free areas (m)	(0.02, 0.15)
WFR2	Channel roughness factor	(0.02, 2)

Table 0-3 Grouping of 111 parameters of the MESH

Group number	Parameters
1	SDEPC, WFR22, ZSNL3, DRNC
2	VPDAC, ZPLS4, SDEPD, ROOTC, SDEPG, XSLPC, RATIOs, ZSNL4, ZSNL1, ZPLG4, DDENC, VPDAD, LAMIND, VPDAG, LNZ0D
3	CLAYSa3, SANDSa2, LAMAXC, XSLPD, SANDSa1, RSMNC, ROOTG, ZSNL11, ZSNL7, XSLPG, ZPLG3, ZPLS3, ZPLS1, ZPLG1, DDEND, CLAYSi3, SANDSi3, LNZ0G, SANDSa3, CLAYSa2, CLAYSa1, QA50C, DRNG, VPDBC, DRND, DDENG
4	LAMAXG, THLQ3, CLAYSi1, SANDSi2, SANDCL3, QA50D, GRKFC, LNZ0C, ALICC, ALVCC, CLAYSi2, ALICG, SANDCL2, SANDCL1, TBAR2, PSGAC, THLQ1, ORGSi3, PSGBC, THLQ2, TBAR3, TPOND, TBAR1, CMASC, MANNC, ZPOND, RATIOSi, QA50G, RSMNG, RSMND, ORGSi2, ORGSi1
5	RATIOCL, CLAYCL3, GRKFD, CMASD, ORGSa3, ORGSa2, ORGSa1, ORGCL1, ORGCL2, CLAYCL2, ORGCL3, CLAYCL1, ALICD, LAMAXD, ALVCG, GRKFG, ALVCD, VPDBG, CMASG
6	ZPLS11, VPDBD, ZPLG11, PSGAG, PSGBG, LAMING, PSGAD, PSGBD, MANNG, ROOTD
7	ZPLG7, ZPLS7, TCANO, MANND

List of Publications

The following papers were published during candidature:

- **Journal Papers:**

Sheikholeslami, R., Razavi, S., 2017. Progressive Latin hypercube sampling: an efficient approach for robust sampling-based analysis of environmental models. Environmental Modelling & Software, 93, 109–126. https://doi.org/10.1016/j.envsoft.2017.03.010

Sheikholeslami, R., Yassin, F., Lindenschmidt, K.E., Razavi, S., 2017. Improved understanding of river ice processes using global sensitivity analysis approaches. Journal of Hydrologic Engineering, 22(11), p.04017048. https://doi.org/10.1061/(ASCE)HE.1943-5584.0001574

Sheikholeslami, R., Razavi, S., Gupta, H.V., Becker, W., Haghnegahdar, A., 2019. Global sensitivity analysis for high-dimensional problems: how to objectively group factors and measure robustness and convergence while reducing computational cost. *Environmental Modelling and Software*, 111, 282–299. https://doi.org/10.1016/j.envsoft.2018.09.002

Sheikholeslami, R., Razavi, S., Haghnegahdar, A., 2019. What do we do with model simulation crashes? Recommendations for global sensitivity analysis of earth systems models. *Geoscientific model Development*, Discussion, https://doi.org/10.5194/gmd-2019-17

Razavi, S., **Sheikholeslami, R.**, Gupta, H., Haghnegahdar, A., 2019. VARS-TOOL: a toolbox for comprehensive, efficient, and robust sensitivity and uncertainty analysis. *Environmental Modelling & Software*, 112, 95–107 https://doi.org/10.1016/j.envsoft.2018.10.005

Sheikholeslami, R., Razavi. S., 2018. Avoiding the Guise of An Anonymous Review, Eos, 99, https://doi.org/10.1029/2018EO098217. Published on 09 May 2018.

- **Conference Publications/Presentations:**

Sheikholeslami, **R.**, Haghnegahdar, A., Razavi, S. Strategies for Handling Simulation Model Crashes in Global Sensitivity Analysis. *AGU Fall Meeting*, Dec 2018.

Sheikholeslami, **R.**, Razavi, S., Gupta, H.V., Becker, W., Haghnegahdar, A. Addressing

Curse of Dimensionality in Global Sensitivity Analysis of Large Environmental Models: An Automated Grouping Strategy. *9th International Congress on Environmental Modelling and Software, iEMSs 2018*, Fort Collins, USA, June 24-28, 2018.

Sheikholeslami, **R**., Razavi, S. Dynamics of Water-related Poverty Trap: Stochastic Modelling of the Interplay between Economic Growth and Water Security, *Resilience 2017–Research Frontiers for Global Sustainability*, Stockholm, Sweden, Aug 20-23, 2017.

Sheikholeslami, **R**., Razavi, S. Finding Positive Feedback Loops in Environmental Models: A Mathematical Investigation, *AGU Fall Meeting*, Dec 2016.

Sheikholeslami, **R**., Razavi, S. A Novel Sampling Approach for Efficient and Robust Uncertainty and Sensitivity Analysis of Environmental Models, *8th International Congress on Environmental Modelling and Software*, *iEMSs 2016*, Toulouse, France, Jul 10-14, 2016.

Sheikholeslami, **R**., Razavi, S. On the Impact of Uncertainty in Initial Conditions of Hydrologic Models on Prediction, *AGU Fall Meeting*, Dec 2015.

References

Aghaji Zare, S.G., Moore, S.A., Rennie, C.D., Seidou, O., Ahmari, H., Malenchak, J., 2015. Boundary shear stress in an ice-covered river during breakup. *Journal of Hydraulic Engineering*, 142(4), 04015065 https://doi.org/10.1061/(ASCE)HY.1943-7900.0001081

Andres, T.H., 1997. Sampling methods and sensitivity analysis for large parameter sets. *Journal of Statistical Computation and Simulation*, 57(1-4), 77–110.

Annan, J.D., Hargreaves, J.C., Edwards, N.R., Marsh, R., 2005. Parameter estimation in an intermediate complexity earth system model using an ensemble Kalman filter. *Ocean Modelling*, 8, 135–154. https://doi.org/10.1016/j.ocemod.2003.12.004

Archer, G.E.B., Saltelli, A., Sobol', I.M., 1997. Sensitivity measures, ANOVA-like techniques and the use of bootstrap. *Journal of Statistical Computation and Simulation*, 58(2), 99–120. http://dx.doi.org/10.1080/00949659708811825

Asadzadeh, M., Razavi, S., Tolson, B.A. and Fay, D., 2014. Pre-emption strategies for efficient multi-objective optimization: Application to the development of Lake Superior regulation plan. *Environmental Modelling & Software*, 54, 128–141. https://doi.org/10.1016/j.envsoft.2014.01.005

Asher, M.J., Croke, B.F.W., Jakeman, A.J., Peeters, L.J.M., 2015. A review of surrogate models and their application to groundwater modelling. *Water Resources Research*, 51(8), 5957–5973. https://doi.org/10.1002/2015WR016967

Aslan, B., Zech, G., 2005. Statistical energy as a tool for binning-free, multivariate goodness-of-fit tests, two-sample comparison and unfolding. *Nucl. Instrum. Meth. A*. 537, 626–636. https://doi.org/10.1016/j.nima.2004.08.071

Ba, S., Myers, W.R., Brenneman, W. A., 2015. Optimal sliced Latin hypercube designs. *Technometrics* 57(4), 479–487. https://doi.org/10.1080/00401706.2014.957867

Baker, F.B., 1974. Stability of two hierarchical grouping techniques Case I: Sensitivity to data errors. *Journal of the American Statistical Association*, 69(346), 440–445.

Bastidas, L.A., Gupta, H.V., Sorooshian, S., Shuttleworth, W.J., Yang, Z.L., 1999. Sensitivity analysis of a land surface scheme using multicriteria methods. *Journal of Geophysical Research: Atmospheres*, 104(D16), 19481–19490. http://dx.doi.org/10.1029/1999JD900155

Bates, S.J., Sienz, J., Toropov, V.V., 2004. Formulation of the optimal Latin hypercube design of experiments using a permutation genetic algorithm. In: *Proceedings of the 5th ASMO-UK/ISSMO Conference on Engineering Design Optimization*, California, US, p.p. 19–22.

Bathiany, S., Dijkstra, H., Crucifix, M., Dakos, V., Brovkin, V., Williamson, M.S., Lenton, T.M., Scheffer, M., 2016. Beyond bifurcation: using complex models to understand and predict abrupt climate change. *Dynamics and Statistics of the Climate System*, 1(1), 1–31. https://doi.org/10.1093/climsys/dzw004

Beck, M.B., 1987. Water quality modelling: a review of the analysis of uncertainty. *Water Resources Research*, 23(8), 1393–1442. https://doi.org/10.1029/WR023i008p01393

Becker, W., 2015. Applications of dynamic trees to sensitivity analysis, In: *Proceedings of the 12th International Conference on Applications of Statistics and Probability in Civil Engineering (ICASP12)*, Haukaas, T. (Ed.), Vancouver, Canada, July 12-15. https://doi.org/10.14288/1.0076191

Beltaos, S., 1993. Transport and mixing processes. Environmental aspects of river ice, T. D. Prowse and N. C. Gridley, (Eds.), Envir. Canada, National Hydrology Research Institute, Saskatoon, Canada, 31–42.

Beltaos, S., Prowse, T., 2009. River-ice hydrology in a shrinking cryosphere. *Hydrological Processes*, 23(1), 122–144. https://doi.org/10.1002/hyp.7165

Benedetti, L., Claeys, F., Nopens, I., Vanrolleghem, P.A., 2011. Assessing the convergence of LHS Monte Carlo simulations of wastewater treatment models. *Water Science and Technology*, 63(10), 2219–2224. https://doi.org/10.2166/wst.2011.453

Bennett, N.D., Croke, B.F., Guariso, G., Guillaume, J.H., Hamilton, S.H., Jakeman, A.J., Marsili-Libelli, S., Newham, L.T., Norton, J.P., Perrin, C., Pierce, S.A., 2013. Characterising performance of environmental models. *Environmental Modelling & Software*, 40, 1–20. https://doi.org/10.1016/j.envsoft.2012.09.011

Beven, K., 1993. Prophecy, reality and uncertainty in distributed hydrological modelling. *Advances in Water Resources*, 16(1), 41–51. https://doi.org/10.1016/0309-1708(93)90028-E

Beven, K., Binley, A., 1992. The future of distributed models: model calibration and uncertainty predication. *Hydrological Process*. 6 (3), 279–298. https://doi.org/10.1002/hyp.3360060305

Borgonovo, E., 2007. A new uncertainty importance measure. *Reliability Engineering & System Safety*, 92(6), 771–784. https://doi.org/10.1016/j.ress.2006.04.015

Borgonovo, E., Castaings, W., Tarantola, S., 2012. Model emulation and moment independent sensitivity analysis: an application to environmental modelling. *Environmental Modelling & Software* 34, 105–115. https://doi.org/10.1016/j.envsoft.2011.06.006

Box, G.E., Cox, D.R., 1964. An analysis of transformations. *Journal of the Royal Statistical Society. Series B (Methodological)*, 26(2), 211–252.

Box, G.E., Meyer, R.D., 1986. An analysis for unreplicated fractional factorials. *Technometrics*, 28(1), 11–18.

Box, G.E.P., Hunter, J.S., 1961. The 2k-p fractional factorial designs: part I. *Technometrics*, 3(3), 311–351. https://doi.org/10.2307/1266725

Boyle, D., 2001. *Multicriteria Calibration of Hydrological Models* (Ph.D. thesis). Dep. of Hydrol. and Water Resour., Univ. of Ariz., Tucson.

Broad, D.R., Dandy, G.C., Maier, H.R., 2015. A systematic approach to determining metamodel scope for risk-based optimization and its application to water distribution system design. *Environmental Modelling & Software* 69, 382–395. https://doi.org/10.1016/j.envsoft.2014.11.015

Burnash, R. J. C., 1995. The NWS River Forecast System—Catchment modelling, in Computer Models of Watershed Hydrology, V. P. Singh, (Ed.), pp. 311–366, Water Resour. Publ., Highlands Ranch, Colo.

Campolongo, F., Cariboni, J. and Saltelli, A., 2007. An effective screening design for sensitivity analysis of large models. *Environmental Modelling & Software*, 22(10), 1509–1518. https://doi.org/10.1016/j.envsoft.2006.10.004

Carson, R.W., Andres, D., Beltaos, S., Groeneyeld, J., Healy, D., Hicks, F., Liu, L.W., Shen, H.T., 2001. Tests of river ice jam models. In: *Proceedings of the 11th Workshop on River Ice. River Ice Processes Within a changing environment*. Canadian Committee on River Ice Processes and the Environment, CGU-HS: Ottawa, ON; pp. 39–55.

Castaings, W., Borgonovo, E., Morris, M., Tarantola, S., 2012. Sampling strategies in density-based sensitivity analysis. *Environmental Modelling & Software* 38, 13–26. https://doi.org/10.1016/j.envsoft.2012.04.017

Castelletti, A. and Soncini-Sessa, R., 2007. Bayesian Networks and participatory modelling in water resource management. *Environmental Modelling & Software*, 22(8), 1075–1088. https://doi.org/10.1016/j.envsoft.2006.06.003

Castelletti, A., Galelli, S., Ratto, M., Soncini-Sessa, R. and Young, P.C., 2012a. A general framework for dynamic emulation modelling in environmental problems. *Environmental Modelling & Software*, 34, 5–18. https://doi.org/10.1016/j.envsoft.2012.01.002

Castelletti, A., Galelli, S., Restelli, M. and Soncini-Sessa, R., 2012b. Data-driven dynamic emulation modelling for the optimal management of environmental systems. *Environmental Modelling & Software*, 34, 30–43. https://doi.org/10.1016/j.envsoft.2011.09.003

Castelletti, A., Lotov, A.V. and Soncini-Sessa, R., 2010. Visualization-based multi-objective improvement of environmental decision-making using linearization of response surfaces. *Environmental Modelling & Software*, 25(12), 1552–1564. https://doi.org/10.1016/j.envsoft.2010.05.011

Chen, F., Shen, H., Jayasundra, N., 2006. A one-dimensional comprehensive river ice model. In: *Proceedings of the 18th IAHR International Symposium on ICE*, pp. 61-68. IAHR, Vancouver, Canada.

Chen, H., Huang, H., Lin, D.K.J., Liu, M.Q., 2016 Uniform sliced Latin hypercube designs. *Applied Stochastic Models in Business and Industry*, 32, 574–584. https://doi.org/10.1002/asmb.2192

Choudhury, B.J., Idso, S.B., 1985. An empirical model for stomatal resistance of field-grown wheat. *Agricultural and Forest Meteorology*, 36(1), 65–82. https://doi.org/10.1016/0168-1923(85)90066-8

Chun, M. H., Han, S. J., Tak, N.I.L., 2000. An uncertainty importance measure using a distance metric for the change in a cumulative distribution function. *Reliability Engineering & System Safety*, 70(3), 313–321. https://doi.org/10.1016/S0951-8320(00)00068-5

Cioppa, T.M., Lucas, T.W., 2007. Efficient nearly orthogonal and space-filling Latin hypercubes. *Technometrics* 49 (1), 45–55. https://doi.org/10.1198/004017006000000453

Ciuffo, B., Azevedo, C.L., 2014. A sensitivity-analysis-based approach for the calibration of traffic simulation models. *IEEE Transactions on Intelligent Transportation Systems*, 15(3), 1298–1309. https://doi.org/10.1109/TITS.2014.2302674

Clancy, D., Tanner, J.E., McWilliam, S., Spencer, M., 2010. Quantifying parameter uncertainty in a coral reef model using Metropolis-Coupled Markov Chain Monte Carlo. *Ecological Modelling*, 221(10), 1337–1347. https://doi.org/10.1016/j.ecolmodel.2010.02.001

Clark, M.P. and Kavetski, D., 2010. Ancient numerical daemons of conceptual hydrological modelling: 1. Fidelity and efficiency of time stepping schemes. *Water Resources Research*, 46(10). https://doi.org/10.1029/2009WR008894

Cosenza, A., Mannina, G., Vanrolleghem, P.A., Neumann, M.B., 2013. Global sensitivity analysis in wastewater applications: A comprehensive comparison of different methods. *Environmental modelling & software*, 49, 40–52. https://doi.org/10.1016/j.envsoft.2013.07.009

Crane, H., Martin, R., 2018. Is statistics meeting the needs of science? *PsyArXiv*, https://doi.org/10.31234/osf.io/q2s5m

Cressie, N.A.C., 1993. *Statistics for Spatial Data*, Revised Edition, John Wiley & Sons, NJ, USA. https://doi.org/10.1002/9781119115151.ch1

Crombecq, K., Laermans, E., Dhaene, T., 2011. Efficient space-filling and non-collapsing sequential design strategies for simulation-based modelling. *European Journal of Operational Research*, 214(3), 683–696. https://doi.org/10.1016/j.ejor.2011.05.032

Cukier, R.I., Fortuin, C.M., Shuler, K.E., Petschek, A.G., Schaibly, J.H., 1973. Study of the sensitivity of coupled reaction systems to uncertainties in rate coefficients. I Theory. *Journal of Chemical Physics*, 59(8), 3873–3878. https://doi.org/10.1063/1.1680571

Davison, A.C., Hinkley, D.V., Young, G.A., 2003. Recent developments in bootstrap methodology. *Statistical Science*, 18(2), 141–157.

Doherty, J., Simmons, C.T., 2013. Groundwater modelling in decision support: reflections on a unified conceptual framework. *Hydrogeology Journal*, 21(7), 1531–1537. https://doi.org/10.1007/s10040-013-1027-7

Dornes, P.F., Tolson, B.A., Davison, B., Pietroniro, A., Pomeroy, J.W., Marsh, P., 2008. Regionalisation of land surface hydrological model parameters in subarctic and arctic environments. *Physics and Chemistry of the Earth, Parts A/B/C*, 33(17), 1081–1089. https://doi.org/10.1016/j.pce.2008.07.007

EC, January 2013. RIVICE Model – User's Manual. Environment Canada Steering Committee. http://giws.usask.ca/rivice/Manual/RIVICE_Manual_2013-01-11.pdf (Nov. 20, 2016).

Edwards, N.R. and Marsh, R., 2005. Uncertainties due to transport-parameter sensitivity in an efficient 3-D ocean-climate model. *Climate Dynamics*, 24(4), 415–433. https://doi.org/10.1007/s00382-004-0508-8

Efron, B., 1979. Bootstrap methods: another look at the jackknife. *Annals of Statistics*, 7(1), 1–26. https://doi.org/10.1214/aos/1176344552

Efron, B., 1982. *The Jackknife, the Bootstrap and Other Resampling Plans*, SIAM, Philadelphia.

Ettema, R., and Daly, S.F., 2004. Sediment transport under ice. U.S. Army Engineer Research and Development Center, Hanover, New Hampshire, ERDC/CRREL Technical Report TR-04-20.

Eweys, O.A., Elwan, A.A., Borham, T.I., 2017. Integrating WOFOST and Noah LSM for modelling maize production and soil moisture with sensitivity analysis, in the east of The Netherlands. *Field Crops Research*, 210, 147–161. https://doi.org/10.1016/j.fcr.2017.06.004

Ferreira, L., Hitchcock, D.B., 2009. A comparison of hierarchical methods for clustering functional data. *Communications in Statistics-Simulation and Computation*, 38(9), 1925–1949. http://dx.doi.org/10.1080/03610910903168603

Fisher, B.E.A., Ireland, M.P., Boyland, D.T., Critten, S.P., 2002. Why use one model? An approach for encompassing model uncertainty and improving best practice. *Environmental Modeling and Assessment*, 7, 291–299. https://doi.org/10.1023/A:102092131

Florian, A., 1992. An efficient sampling scheme: updated Latin hypercube sampling. *Probabilistic Engineering Mechanics*, 7(2), 123–130. https://doi.org/10.1016/0266-8920(92)90015-A

Fraga, I., Cea, L., Puertas, J., Suárez, J., Jiménez, V., Jácome, A., 2016. Global Sensitivity and GLUE-based uncertainty analysis of a 2D-1D dual urban drainage model. *Journal of Hydrologic Engineering*, 21(5), p.04016004. https://doi.org/10.1061/(ASCE)HE.1943-5584.0001335

Fu, C., Popescu, I., Wang, C., Mynett, A. E., Zhang, F., 2014. "Challenges in modelling river flow and ice regime on the Ningxia–Inner Mongolia reach of the Yellow River, China. *Hydrology and Earth System Sciences*, 18, 1225–1237, https://doi.org/10.5194/hess-18-1225-2014

Galili, T., 2015. dendextend: an R package for visualizing, adjusting and comparing trees of hierarchical clustering. *Bioinformatics*, 31(22), 3718–3720. https://doi.org/10.1093/bioinformatics/btv428

Gan, Y., Duan, Q., Gong, W., Tong, C., Sun, Y., Chu, W., Ye, A., Miao, C., Di, Z., 2014. A comprehensive evaluation of various sensitivity analysis methods: A case study with a hydrological model. *Environmental Modelling & Software*, 51, 269-285. https://doi.org/10.1016/j.envsoft.2013.09.031

Ganji, A., Maier, H.R., Dandy, G.C., 2016. A modified Sobol′ sensitivity analysis method for decision-making in environmental problems. *Environmental Modelling & Software*, 75, 15–27. https://doi.org/10.1016/j.envsoft.2015.10.001

Göhler, M., Mai, J., Cuntz, M., 2013. Use of eigendecomposition in a parameter sensitivity analysis of the Community Land Model, *Journal of Geophysical Research: Biogeosciences*, 118, 904–921, https://doi.org/10.1002/jgrg.20072

Gong, W., Duan, Q.A., Li, J.D., Wang, C., Di, Z.H., Ye, A.Z., Miao, C.Y., Dai, Y.J., 2016. An intercomparison of sampling methods for uncertainty quantification of environmental dynamic models, *Journal of Environmental Informatics*, 28 (1), 11–24, https://doi.org/10.3808/jei.201500310

Guo, D., Westra, S., Maier, H.R., 2016. An inverse approach to perturb historical rainfall data for scenario-neutral climate impact studies. *Journal of Hydrology*, 556, 877–890. https://doi.org/10.1016/j.jhydrol.2016.03.025

Gupta, H. V., Razavi, S., 2018. Revisiting the basis of sensitivity analysis for dynamical earth system models. *Water Resources Research*, 54(11), 8692–8717. https://doi.org/10.1029/2018WR022668

Gupta, H., Razavi, S., 2017. Challenges and Future Outlook of Sensitivity Analysis. In G. Petropoulos and P. Srivastava (Eds.), *Sensitivity Analysis in Earth Observation Modelling*, Elsevier, 397–415, https://doi.org/10.1016/B978- 0-12-803011-0.00020-3

Gupta, H.V., Wagener, T., Liu, Y., 2008. Reconciling theory with observations: elements of a diagnostic approach to model evaluation. *Hydrological Processes*, 22(18), 3802–3813. https://doi.org/10.1002/hyp.6989

Gupta, H.V., Sorooshian, S., Yapo, P.O., 1998. Toward improved calibration of hydrologic models: Multiple and noncommensurable measures of information. *Water Resources Research*, 34(4), 751–763. https://doi.org/10.1029/97WR03495

Haghnegahdar, A., Razavi, S., 2017. Insights into sensitivity analysis of earth and environmental systems models: On the impact of parameter perturbation scale. *Environmental Modelling & Software*, 95, 115–131. https://doi.org/10.1016/j.envsoft.2017.03.031

Haghnegahdar, A., Razavi, S., Yassin, F., Wheater, H., 2017. Multicriteria sensitivity analysis as a diagnostic tool for understanding model behaviour and characterizing model uncertainty. *Hydrological Processes*, 31(25), 4462–4476., https://doi.org/10.1002/hyp.11358

Haghnegahdar, A., Tolson, B.A., Craig, J.R., Paya, K.T., 2015. Assessing the performance of a semi-distributed hydrological model under various watershed discretization schemes. *Hydrological Processes*, 29(18), 4018–4031. https://doi.org/10.1002/hyp.10550

Hair, J. F., Tatham, R. L., Anderson, R. E., Black, W., 2006. *Multivariate data analysis*. Upper Saddle River, NJ, Pearson Prentice Hall.

Hall, J., Boyce, S., Wang, Y., Dawson, R., Tarantola, S., Saltelli, A., 2009. Sensitivity analysis for hydraulic models. *Journal of Hydraulic Engineering*, https://doi.org/10.1061/(ASCE)HY.1943-7900.0000098

Halton, J.H., 1960. On the efficiency of certain quasi-random sequences of points in evaluating multi-dimensional integrals. *Numerische Mathematik*, 2(1), 84–90. https://doi.org/10.1007/BF01386213

Hammersley, J.M., 1960. Monte Carlo methods for solving multivariable problems. *Annals of the New York Academy of Sciences*, 86 (3), 844–874. https://doi.org/10.1111/j.1749-6632.1960.tb42846.x

Hands, S., Everitt, B., 1987. A Monte Carlo study of the recovery of cluster structure in binary data by hierarchical clustering techniques. *Multivariate Behavioral Research*, 22(2), 235–243. http://dx.doi.org/10.1207/s15327906mbr2202_6

Hanna, S.R., 1993. Uncertainties in air quality model predictions. *Boundary-Layer Meteorology*, 62(1–4), 3–20. https://doi.org/10.1007/BF00705545

Helton, J.C., Davis, F.J., 2003. Latin hypercube sampling and the propagation of uncertainty in analyses of complex systems. *Reliability Engineering & System Safety*, 81(1), 23–69. https://doi.org/10.1016/S0951-8320(03)00058-9

Helton, J.C., Johnson, J.D., Sallaberry, C.J., Storlie, C.B., 2006. Survey of sampling-based methods for uncertainty and sensitivity analysis. *Reliability Engineering & System Safety*, 91(10–11), 1175–1209. https://doi.org/10.1016/j.ress.2005.11.017

Herman, J.D., Reed, P.M., Wagener, T., 2013. Time-varying sensitivity analysis clarifies the effects of watershed model formulation on model behavior. *Water Resources Research*, 49(3), 1400–1414. https://doi.org/10.1002/wrcr.20124

Herman, J.D., Kollat, J.B., Reed, P.M., Wagener, T., 2013. Technical note: method of Morris effectively reduces the computational demands of global sensitivity analysis for distributed watershed models. *Hydrology and Earth System Sciences*, 17(7), 2893–2903. https://doi.org/10.5194/hess-17-2893-2013.

Hernandez A.S., 2008. *Breaking Barriers to Design Dimensions in Nearly Orthogonal Latin Hypercubes* (Ph.D. thesis), Naval Postgraduate School, Monterey, California.

Herrera, L.J., Pomares, H., Rojas, I., Guillén, A., Rubio, G., Urquiza, J., 2011. Global and local modelling in RBF networks. *Neurocomputing*, 74(16), 2594–2602. https://doi.org/10.1016/j.neucom.2011.03.027

Hickernell, F.J., 1998. A generalized discrepancy and quadrature error bound. *Mathematics of Computation of the American Mathematical Society*, 67(221), 299–322. https://doi.org/10.1090/S0025-5718-98-00894-1

Hicks, F., 2009. An overview of river ice problems: CRIPE07 guest editorial. *Cold Regions Science and Technology*, 55(2), 175–185. https://doi.org/10.1016/j.coldregions.2008.09.006

Hicks, F., Andrishak, R., She, Y., 2006. Modelling thermal and dynamic river ice processes. M. Davies, J.E. Zufelt (Eds.), In *Proceedings of the 13th International Conference on Cold Regions Engineering*, ASCE, Orono, Maine, USA (2006), pp.1-11.

Higdon, D., Gattiker, J., Lawrence, E., Jackson, C., Tobis, M., Pratola, M., Habib, S., Heitmann, K., Price, S., 2013. Computer model calibration using the ensemble Kalman filter. *Technometrics* 55(4), 488–500. https://doi.org/10.1080/00401706.2013.842936

Holtschlag, D., Grewal, M., 1998. Estimating ice-affected streamflow by extended Kalman filtering. *Journal of Hydrologic Engineering*, 3(3), 174–181. https://doi.org/10.1061/(ASCE)1084-0699(1998)3:3(174) , 174–181.

Homma, T., Saltelli, A., 1996. Importance measures in global sensitivity analysis of nonlinear models. *Reliability Engineering & System Safety*, 52(1), 1–17. https://doi.org/10.1016/0951-8320(96)00002-6

Hornberger, G., Spear, R., 1981. Approach to the preliminary analysis of environmental systems. *Journal of Environmental Management*, 12, 7–18.

Hou, T., Zhu, Y., Lü, H., Sudicky, E., Yu, Z., Ouyang, F., 2015. Parameter sensitivity analysis and optimization of Noah land surface model with field measurements from Huaihe River Basin, China. *Stochastic environmental research and risk assessment*, 29(5), 1383–1401. https://doi.org/10.1007/s00477-015-1033-5

Husslage, B.G., Rennen, G., van Dam, E.R., den Hertog, D., 2011. Space-filling Latin hypercube designs for computer experiments. *Optimization and Engineering*, 12(4), 611–630. https://doi.org/10.1007/s11081-010-9129-8

Iman, R.L., Conover, W.J., 1980. Small sample sensitivity analysis techniques for computer models, with an application to risk assessment. *Communications in Statistics - Theory and Methods*, 9(17), 1749–1874. https://doi.org/10.1080/03610928008827996

Iooss, B., Boussouf, L., Feuillard, V., Marrel, A., 2010. Numerical studies of the metamodel fitting and validation processes. *International Journal of Advances in Systems and Measurements*, 3, 11–21.

Islam, T., Pruyt, E., 2016. Scenario generation using adaptive sampling- the case of resource scarcity. *Environmental Modelling & Software*, 79, 285–299. https://doi.org/10.1016/j.envsoft.2015.09.014

Jin, R., Chen, W. and Simpson, T.W., 2001. Comparative studies of metamodelling techniques under multiple modelling criteria. *Structural and multidisciplinary optimization*, 23(1), 1–13. https://doi.org/10.1007/s00158-001-0160-4

Jin, R., Chen, W., Sudjianto, A., 2005. An efficient algorithm for constructing optimal design of computer experiments. *Journal of Statistical Planning and Inference*, 134(1), 268–287. https://doi.org/10.1016/j.jspi.2004.02.014

Johnson, S.C., 1967. Hierarchical clustering schemes. *Psychometrika*, 32(3), 241–254.

Jourdan, A., Franco, J., 2010. Optimal Latin hypercube designs for the Kullback-Leibler criterion. *AStA Advances in Statistical Analysis*, 94 (4), 341–351. https://doi.org/10.1007/s10182-010-0145-y

Kalagnanam J.R., Diwekar U.M., 1997. An efficient sampling technique for off-line quality control. *Technometrics* 39 (3), 308–319.

Kavetski, D. and Clark, M.P., 2010. Ancient numerical daemons of conceptual hydrological modelling: 2. Impact of time stepping schemes on model analysis and prediction. *Water Resources Research*, 46(10). https://doi.org/10.1029/2009WR008896

Kavetski, D., Kuczera, G., Franks, S.W., 2006. Bayesian analysis of input uncertainty in hydrological modelling: 1. Theory. *Water Resources Research*, 42(3). https://doi.org/10.1029/2005WR004368

Kelleher, C., Wagener, T., McGlynn, B., Ward, A.S., Gooseff, M.N., Payn, R.A., 2013. Identifiability of transient storage model parameters along a mountain stream. *Water Resources Research*, 49(9), 5290–5306. https://doi.org/10.1002/wrcr.20413

Klepper, O., 1997. Multivariate aspects of model uncertainty analysis: tools for sensitivity analysis and calibration. *Ecological Modelling*, 101(1), 1–13. https://doi.org/10.1016/S0304-3800(96)01922-9

Kocis, L., Whiten, W.J., 1997. Computational investigations of low-discrepancy sequences. *ACM Transactions on Mathematical Software*, 23(2), 266–294. https://doi.org/.1145/264029.264064

Kodinariya, T.M., Makwana, P.R., 2013. Review on determining number of cluster in k-means clustering. *International Journal of Advance Research in Computer Science and Management Studies*, 1(6), 90–95.

Kolmogorov, A., 1933. "Sulla determinazione empirica di una legge di distribuzione [On the empirical determination of a distribution law]." *Giorn. Ist. Ital. Attuar.*, 4, 83–91.

Kouwen, N., Soulis, E.D., Pietroniro, A., Donald, J., Harrington, R.A., 1993. Grouped response units for distributed hydrologic modelling. *Journal of Water Resources Planning and Management*, 119(3), 289–305. https://doi.org/10.1061/(ASCE)0733-9496(1993)119:3(289)

Krykacz-Hausmann, B., 2001. Epistemic sensitivity analysis based on the concept of entropy. In: *Proceedings of SAMO2001*, CIEMAT, Madrid, pp. 31–35.

Kuczera, G., Mroczkowski, M., 1998. Assessment of hydrologic parameter uncertainty and the worth of multiresponse data. *Water Resources Research*, 34(6), 1481–1489. https://doi.org/10.1029/98WR00496

Kuczera, G., Parent, E., 1998. Monte Carlo assessment of parameter uncertainty in conceptual catchment models: the Metropolis algorithm. J. *Hydrol*. 211 (1), 69–85. https://doi.org/10.1016/S0022-1694(98)00198-X

Lagarias, J.C., Reeds, J.A., Wright, M.H., Wright, P.E., 1998. Convergence properties of the Nelder-Mead simplex method in low dimensions. *SIAM Journal on optimization*, 9(1), 112–147. https://doi.org/10.1137/S1052623496303470

Lal, A., Shen, H., 1991. Mathematical model for river ice processes. *Journal of Hydraulic Engineering*, 117(7) 851–867. https://doi.org/10.1061/(ASCE)0733-9429(1991)117:7(851)

Lamboni, M., Monod, H., Makowski, D., 2011. Multivariate sensitivity analysis to measure global contribution of input factors in dynamic models. *Reliability Engineering & System Safety*, 96(4), 450–459. https://doi.org/10.1016/j.ress.2010.12.002

Li, J., Duan, Q.Y., Gong, W., Ye, A., Dai, Y., Miao, C., Di, Z., Tong, C., Sun, Y., 2013. Assessing parameter importance of the Common Land Model based on qualitative and quantitative sensitivity analysis. *Hydrology and Earth System Sciences*, 17(8), 3279-3293. https://doi.org/10.5194/hess-17-3279-2013

Lin, Y., 2004. *An efficient robust concept exploration method and sequential ex- ploratory experimental design* (Ph.D. thesis). Georgia Institute of Technology.

Lindenschmidt, K.E., Chun, K.P., 2013. Evaluating the impact of fluvial geomorphology on river ice cover formation based on a global sensitivity analysis of a river ice model. *Canadian Journal of Civil Engineering*, 40(7), 623–632. https://doi.org/10.1139/cjce-2012-0274

Lindenschmidt, K.E., Das, A., Rokaya, P., Chun, K., Chu, T., 2015. Ice jam flood hazard assessment and mapping of the Peace River at the Town of Peace River. Proc., *18th Workshop on the Hydraulics of Ice Covered Rivers*, Quebec.

Lindenschmidt, K.E., Sereda, J., 2014. The impact of macrophytes on winter flows along the Upper Qu'Appelle River. *Canadian Water Resources Journal*, 39(3), 342–355. https://doi.org/10.1080/07011784.2014.942165

Lindenschmidt, K.E., Sydor, M., Carson, R., 2012. Modelling ice cover formation of a lake-river system with exceptionally high flows (Lake St. Martin and Dauphin River, Manitoba). *Cold Regions Science and Technology*, 82, 36–48. https://doi.org/10.1016/j.coldregions.2012.05.006

Lindenschmidt, K.E., Sydor, M., Carson, R., Harrison, R., 2011. Ice jam modelling of the Red River in Winnipeg. In *Proceedings of the 16th CRIPE Workshop on the Hydraulics of Ice Covered Rivers*, CGU HS Committee on River Ice Processes and the Environment, pp.274–290.

Lindström, G., Johansson, B., Persson, M., Gardelin, M., Bergström, S., 1997. Development and test of the distributed HBV-96 hydrological model, *Journal of hydrology*, 201(1–4), 272–288.

Linkov, I., Ramadan, A.B., 2004. *Comparative Risk Assessment and Environmental Decision Making*. Springer, Netherlands. https://doi.org/10.1007/1-4020- 2243-3

Little, R. J. A., Rubin, D.B., 1987. *Statistical Analysis with Missing Data*. New York: John Wiley & Sons.

Liu, H., Chen, W., Sudjianto, A., 2006. Relative entropy based method for probabilistic sensitivity analysis in engineering design. *Journal of Mechanical Design*, 128 (2), 326–336. https://doi.org/10.1115/1.2159025

Liu, Y., Gupta, H.V., 2007. Uncertainty in hydrologic modelling: toward an integrated data assimilation framework. *Water Resources Research*, 43(7). W07401, https://doi.org/10.1029/2006WR005756

Liu, Y., Gupta, H.V., Sorooshian, S., Bastidas, L.A., Shuttleworth, W.J., 2004. Exploring parameter sensitivities of the land surface using a locally coupled land-atmosphere model. *Journal of Geophysical Research: Atmospheres*, 109, D21101. https://doi.org/10.1029/2004JD004730

Loyola, D., Pedergnana, M., García, S.G., 2016. Smart sampling and incremental function learning for very large high dimensional data. *Neural Networks* 78, 75–87. https://doi.org/10.1016/j.neunet.2015.09.001

Lucas, D.D., Klein, R., Tannahill, J., Ivanova, D., Brandon, S., Domyancic, D., Zhang, Y., 2013. Failure analysis of parameter-induced simulation crashes in climate models. *Geoscientific Model Development*, 6(4), 1157–1171. https://doi.org/10.5194/gmd-6-1157-2013

Maier, H.R., Guillaume, J.H., van Delden, H., Riddell, G.A., Haasnoot, M., Kwakkel, J.H., 2016. An uncertain future, deep uncertainty, scenarios, robustness and adaptation: How do they fit together?. *Environmental Modelling & Software*, 81, 154–164. https://doi.org/10.1016/j.envsoft.2016.03.014

Maier, H.R., Kapelan, Z., Kasprzyk, J., Kollat, J., Matott, L.S., Cunha, M.C., Dandy, G.C., Gibbs, M.S., Keedwell, E., Marchi, A., Ostfeld, A., 2014. Evolutionary algorithms and other metaheuristics in water resources: Current status, research challenges and future directions. *Environmental Modelling & Software*, 62, 271–299. https://doi.org/10.1016/j.envsoft.2014.09.013

Marino, S., Hogue, I.B., Ray, C.J., Kirschner, D.E., 2008. A methodology for performing global uncertainty and sensitivity analysis in systems biology. *Journal of Theoretical Biology*, 254(1), 178–196. https://doi.org/10.1016/j.jtbi.2008.04.011

Matott, L.S., Rabideau, A.J., 2008. Calibration of complex subsurface reaction models using a surrogate-model approach. *Advances in Water Resources*, 31(12), 1697–1707. https://doi.org/10.1016/j.advwatres.2008.08.006

McKay, M.D., Conover, W.J., 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 (2), 239–245. https://doi.org/10.2307/1268522

McMillan, H., Krueger, T., Freer, J., 2012. Benchmarking observational uncertainties for hydrology: rainfall, river discharge and water quality. *Hydrological Processes*, 26(26), 4078–4111. https://doi.org/10.1002/hyp.9384

McRoberts, R.E., 2009. Diagnostic tools for nearest neighbors techniques when used with satellite imagery. *Remote Sensing of Environment*, 113(3), 489–499. https://doi.org/10.1016/j.rse.2008.06.015

Medici, C., Wade, A.J., Francés, F., 2012. Does increased hydrochemical model complexity decrease robustness? *Journal of hydrology*, 440, 1–13. https://doi.org/10.1016/j.jhydrol.2012.02.047

Metzger, C., Nilsson, M.B., Peichl, M., Jansson, P.E., 2016. Parameter interactions and sensitivity analysis for modelling carbon heat and water fluxes in a natural peatland, using CoupModel v5. *Geoscientific Model Development*, 9(12), 4313–4338. https://doi.org/10.5194/gmd-9-4313-2016

Milligan, G.W., Cooper, M.C., 1988. A study of standardization of variables in cluster analysis. *Journal of classification*, 5(2), 181–204. https://doi.org/10.1007/BF01897163

Moeur, M., Stage, A.R., 1995. Most similar neighbor: an improved sampling inference procedure for natural resource planning. *Forest science*, 41(2), 337–359. https://doi.org/10.1093/forestscience/41.2.337

Montgomery, D. C., 2008. *Design and Analysis of Experiments*. John Wiley & Sons, Inc., Hoboken, N. J.

Morris, M.D., 1991. Factorial sampling plans for preliminary computational experiments. *Technometrics*, 33(2), 161–174. https://doi.org/10.2307/1269043

Morris, M.D., Mitchell, T.J., 1995. Exploratory designs for computational experiments. *Journal of Statistical Planning and Inference*, 43(3), 381–402. https://doi.org/10.1016/0378-3758(94)00035-T

Mugunthan, P., Shoemaker, C.A., 2006. Assessing the impacts of parameter uncertainty for computationally expensive groundwater models. *Water Resources Research*, 42, 1–15. https://doi.org/10.1029/2005WR004640

Mullur, A.A., Messac, A., 2006. Metamodelling using extended radial basis functions: a comparative approach. *Engineering with Computers*, 21(3), 203–217. https://doi.org/10.1007/s00366-005-0005-7

Niederreiter, H., 1992. Random number generation and quasi-Monte Carlo methods. In: *Proceedings of CBMS-NSF Regional Conference Series in Applied Mathematics*, SIAM, Philadelphia.

Nossent, J., Elsen, P., Bauwens, W., 2011. Sobol' sensitivity analysis of a complex environmental model. *Environmental Modelling & Software*, 26(12), 1515–1525. https://doi.org/10.1016/j.envsoft.2011.08.010

Ohara, N., Jang, S., Kure, S., Richard Chen, Z.Q., Kavvas, M.L., 2014. Modelling of interannual snow and ice storage in high-altitude regions by dynamic equilibrium concept. *Journal of Hydrologic engineering*, 19(12), https://doi.org/10.1061/(ASCE)HE.1943-5584.0000988

Ong, M.S., Kuang, Y.C., Ooi, M.P-L., 2012. Statistical measures of two dimensional point set uniformity. *Computational Statistics and Data Analysis*56(6), 2159–2181. https://doi.org/10.1016/j.csda.2011.12.005

Oreskes, N., 2003. The Role of quantitative models in science. Chapter 2. In: C.D. Canham, J.J. Cole, and W.K. Lauenroth, (Eds.). *Models in ecosystem science*. Princeton, NJ: Princeton Univ Press, 13–31.

Owen, A.B., 1992. Orthogonal arrays for computer experiments, integration and visualization. *Statistica Sinica*, 2(2), 439–452.

Owen, A.B., 1994. Controlling correlations in Latin hypercube samples. *Journal of the American Statistical Association*, 89(428), 1517–1522. https://doi.org/10.2307/2291014

Paja, M., Wrzesien, M., Niemiec, R., Rudnicki, W.R., 2016. Application of all-relevant feature selection for the failure analysis of parameter-induced simulation crashes in climate models. *Geoscientific Model Development*, 9(3), 1065–1072. https://doi.org/10.5194/gmd-9-1065-2016

Pappenberger, F., Beven, K.J., 2004. Functional classification and evaluation of hydrographs based on multicomponent mapping (Mx). *International Journal of River Basin Management*, 2(2), 89–100. https://doi.org/10.1080/15715124.2004.9635224

Pappenberger, F., Beven, K.J., Ratto, M., Matgen, P., 2008. Multi-method global sensitivity analysis of flood inundation models. *Advances in Water Resources*, 31(1), 1–14. https://doi.org/10.1016/j.advwatres.2007.04.009

Patelli, E., Pradlwarter, H.J., Schuëller, G.I., 2010. Global sensitivity of structural variability by random sampling. *Computer Physics Communications*, 181(12), 2072–2081. https://doi.org/10.1016/j.cpc.2010.08.007

Perz, S.G., Muñoz-Carpena, R., Kiker, G., Holt, R.D., 2013. Evaluating ecological resilience with global sensitivity and uncertainty analysis. *Ecological Modelling*, 263, 174–186. https://doi.org/10.1016/j.ecolmodel.2013.04.024

Pfannerstill, M., Guse, B., Reusser, D., Fohrer, N., 2015. Process verification of a hydrological model using a temporal parameter sensitivity analysis. *Hydrology and Earth System Sciences*, 19(10), 4365–4376. https://doi.org/10.5194/hess-19-4365-2015

Pianosi, F., Wagener, T., 2015. A simple and efficient method for global sensitivity analysis based on cumulative distribution functions. *Environmental Modelling & Software*, 67, 1–11. http://dx.doi.org/10.1016/j.envsoft.2015.01.004

Pietroniro, A., Fortin, V., Kouwen, N., Neal, C., Turcotte, R., Davison, B., Verseghy, D., Soulis, E.D., Caldwell, R., Evora, N., Pellerin, P., 2007. Development of the MESH modelling system for hydrological ensemble

forecasting of the Laurentian Great Lakes at the regional scale. *Hydrology and Earth System Sciences* 11(4): 1279–1294. https://doi.org/10.5194/hess-11-1279-2007

Poff, N.L., Brown, C.M., Grantham, T.E., Matthews, J.H., Palmer, M.A., Spence, C.M., Wilby, R.L., Haasnoot, M., Mendoza, G.F., Dominique, K.C., Baeza, A., 2015. Sustainable water management under future uncertainty with eco-engineering decision scaling. *Nature Climate Change*, 6(1), 25–34. https://doi.org/10.1038/nclimate2765

Posselt, D.J., Fryxell, B., Molod, A., Williams, B., 2016. Quantitative sensitivity analysis of physical parameterizations for cases of deep convection in the NASA GEOS-5. *Journal of Climate*, 29(2), 455–479. https://doi.org/10.1175/JCLI-D-15-0250.1

Pronzato, L., Müller, W.G., 2012. Design of computer experiments: space filling and beyond. *Statistics and Computing*, 22(3), 681–701. https://doi.org/10.1007/s11222-011-9242-3

Provenzale, A. Climate models. 2014. *Rendiconti Lincei-Scienze Fisiche E Naturali*, 25(1), 49–58. https://doi.org/10.1007/s12210-013-0268-7

Prowse, T.D., 2001. River-ice ecology. II: biological aspects. *Journal of Cold Regions Engineering*, 15(1), 17–33. https://doi.org/10.1061/(ASCE)0887-381X(2001)15:1(17)

Qian, P.Z.G., 2012. Sliced Latin hypercube designs. *Journal of the American Statistical Association*, 107(497), 393–399. https://doi.org/10.1080/01621459.2011.644132

Radomyski, A., Giubilato, E., Ciffroy, P., Critto, A., Brochot, C., Marcomini, A., 2016. Modelling ecological and human exposure to POPs in Venice lagoon–Part II: Quantitative uncertainty and sensitivity analysis in coupled exposure models. *Science of The Total Environment*, 569–570, 1635–1649. https://doi.org/10.1016/j.scitotenv.2016.07.057

Rainville, F.-M.D., Gagné, C., Teytaud, O., Laurendeau, D., 2012. Evolutionary optimization of low-discrepancy sequences. A*CM Transactions on Modeling and Computer Simulation*, 22, 1–25. https://doi.org/10.1145/2133390.2133393

Rajabi M.M., Ataie-Ashtiani B., Janssen H., 2015. Efficiency enhancement of optimized Latin hypercube sampling strategies: application to Monte Carlo uncertainty analysis and meta-modelling. *Advances in Water Resources*, 76, 127–39. https://doi.org/10.1016/j.advwatres.2014.12.008

Razavi, S., Sheikholeslami, R., Gupta, H., Haghnegahdar, A., 2019. VARS-TOOL: a toolbox for comprehensive, efficient, and robust sensitivity and uncertainty analysis. *Environmental Modelling & Software*, 112, 95–107 https://doi.org/10.1016/j.envsoft.2018.10.005

Razavi, S. and Gupta, H.V., 2019. A multi-method generalized global sensitivity matrix approach to accounting for the dynamical nature of earth and environmental systems models. *Environmental Modelling & Software*, 114, 1–11. https://doi.org/10.1016/j.envsoft.2018.12.002

Razavi, S., 2017. Feature: When Uncertainty Matters, American Geophysical Union-Hydrology Section, December 2017, Newsletter, Invited Commentary (https://hydrology.agu.org/wp-content/uploads/sites/19/2017/11/HS-December-2017-Newsletter_final.pdf)

Razavi, S., Gupta, H.V., 2015. What do we mean by sensitivity analysis? The need for comprehensive characterization of "global" sensitivity in Earth and Environmental systems models. *Water Resources Research*, 51(5), 3070-3092. https://doi./org/10.1002/2014WR016527

Razavi, S., Gupta, H.V., 2016a. A new framework for comprehensive, robust, and efficient global sensitivity analysis: 1. Theory. *Water Resources Research*, 52, 423–439. https://doi.org/10.1002/2015WR017558

Razavi, S., Gupta, H.V., 2016b. A new framework for comprehensive, robust, and efficient global sensitivity analysis: 2. Application. *Water Resources Research*, 52, 440–455. https://doi.org/10.1002/2015WR017559

Razavi, S., Tolson, B.A., Burn, D.H., 2012a. Review of surrogate modelling in water resources. *Water Resources Research*, 48(7), W07401. https://doi.org/10.1029/2011WR011527.

Razavi, S., Tolson, B.A., Burn, D.H., 2012b. Numerical assessment of metamodelling strategies in computationally intensive optimization. *Environmental Modelling & Software*, 34, 67–86. https://doi.org/10.1016/j.envsoft.2011.09.010.

Razavi, S., Tolson, B.A., Matott, L.S., Thomson, N.R., MacLean, A. and Seglenieks, F.R., 2010. Reducing the computational cost of automatic calibration through model preemption. *Water Resources Research*, 46(11). https://doi.org/10.1029/2009WR008957

Regis, R.G., Shoemaker, C.A., 2007. Parallel radial basis function methods for the global optimization of expensive functions. *European Journal of Operational Research*, 182, 514–535. https://doi.org/10.1016/j.ejor.2006.08.040

Rezaie K, Amalnik M.S., Gereie A, Ostadi B, Shakhseniaee M., 2007. Using extended Monte Carlo simulation method for the improvement of risk management: consideration of relationships between uncertainties. *Applied Mathematical Computation*, 190(2), 1492–1501. https://doi.org/10.1016/j.amc.2007.02.038

Rosolem, R., Gupta, H.V., Shuttleworth, W.J., Zeng, X., Gonçalves, L.G.G., 2012. A fully multiple-criteria implementation of the Sobol′ method for parameter sensitivity analysis. *Journal of Geophysical Research*, 117(D7). https://doi.org/10.1029/2011JD016355

Sakia, R.M., 1992. The Box-Cox transformation technique: a review. *The Statistician*, 41(2), 169–178. https://doi.org/10.2307/2348250

Sallaberry, C.J., Helton, J.C., Hora, S.C., 2008. Extension of Latin hypercube samples with correlated variables. *Reliability Engineering and System Safety*, 93, 1047–1059. https://doi.org/10.1016/j.ress.2007.04.005

Saltelli, A., Annoni, P., 2010. How to avoid a perfunctory sensitivity analysis. *Environmental Modelling & Software*, 25(12), 1508-1517. https://doi.org/10.1016/j.envsoft.2010.04.012

Saltelli, A., Annoni, P., Azzini, I., Campolongo, F., Ratto, M., Tarantola, S., 2010. Variance based sensitivity analysis of model output. Design and estimator for the total sensitivity index. *Computer Physics Communications*, 181(2), 259–270. https://doi.org/10.1016/j.cpc.2009.09.018

Saltelli, A., Ratto, M., Andres, T., Campolongo, F., Cariboni, J., Gatelli, D., Saisana, M., Tarantola, S., 2008. *Global Sensitivity Analysis: The Primer*. John Wiley & Sons.

Saltelli, A., Ratto, M., Tarantola, S., Campolongo, F., 2006. Sensitivity analysis practices: strategies for model-based inference. *Reliability Engineering & System Safety*, 91(10), 1109–1125. https://doi.org/10.1016/j.ress.2005.11.014

Saltelli, A., Sobol, I.M., 1995. About the use of rank transformation in sensitivity analysis of model output. *Reliability Engineering & System Safety*, 50(3), 225–239. https://doi.org/10.1016/0951-8320(95)00099-2

Saltelli, A., Tarantola, S., Chan, K.P.S., 1999. A quantitative model-independent method for global sensitivity analysis of model output. *Technometrics* 41(1), 39–56. https://doi.org/10.2307/1270993

Santner, T.J., Williams, B.J., Notz, W.I., 2003. *The Design and Analysis of Computer Experiments*. Springer Series in Statistics. Springer. New York.

Sarrazin, F., Pianosi, F., Wagener, T., 2016. Global sensitivity analysis of environmental models: convergence and validation. *Environmental Modelling & Software*, 79, 135–152. https://doi.org/10.1016/j.envsoft.2016.02.005

Satopää, V., Albrecht, J., Irwin, D., Raghavan, B., 2011. Finding a ‘‘Kneedle” in a haystack: detecting knee points in system behavior. In: *31st International Conference on Distributed Computing Systems Workshops*, 166–117, Minneapolis.

Schretter, C., Kobbelt, L., Dehaye, P.O., (2012). Golden ratio sequences for low-discrepancy sampling. *Journal of Graphics Tools*, 16(2), 95–104. https://doi.org/10.1080/2165347X.2012.679555

Sheikholeslami, R., Razavi, S., 2017. Progressive Latin hypercube sampling: an efficient approach for robust sampling-based analysis of environmental models. *Environmental Modelling & Software*, 93, 109–126. https://doi.org/10.1016/j.envsoft.2017.03.010

Sheikholeslami, R., Yassin, F., Lindenschmidt, K.E., Razavi, S., 2017. Improved understanding of river ice processes using global sensitivity analysis approaches. *Journal of Hydrologic Engineering*, 22(11), p.04017048. https://doi.org/10.1061/(ASCE)HE.1943-5584.0001574

Sheikholeslami, R., Razavi, S., Gupta, H.V., Becker, W., Haghnegahdar, A., 2019. Global sensitivity analysis for high-dimensional problems: how to objectively group factors and measure robustness and convergence while reducing computational cost. Environmental Modelling and Software, 111, 282–299. https://doi.org/10.1016/j.envsoft.2018.09.002

Sheikholeslami, R., Razavi, S., Haghnegahdar, A., 2019. What do we do with model simulation crashes? Recommendations for global sensitivity analysis of earth systems models. *Geoscientific Model Development*, Discussion, https://doi.org/10.5194/gmd-2019-17

Shen H.T., Liu L., Chen Y.C., 2001. River ice dynamics and ice jam modelling. In: Dempsey J.P., Shen H.H. (Eds.) *IUTAM Symposium on Scaling Laws in Ice Mechanics and Ice Dynamics*. Solid Mechanics and Its Applications, vol 94. Springer, Dordrecht. pp. 349–362

Shen, H., Wang, D., Lal, A., 1995. Numerical simulation of river ice processes. *Journal of Cold Regions Engineering*, 9(3), 107–118. https://doi.org/10.1061/(ASCE)0887-381X(1995)9:3(107)

Simpson, T.W., Lin, D.K.J., Chen, W., 2001. Sampling strategies for computer experiments design and analysis. *International Journal of Reliability and Applications*, 2(3), 209–240.

Singh, V. P., Frevert, D.K., 2002a. *Mathematical Models of Small Watershed Hydrology and Applications*, 950 pp., Water Resour. Publ., Highlands Ranch, Colo.

Singh, V. P., Frevert, D.K., 2002b. *Mathematical Models of Large Watershed Hydrology*, 891 pp., Water Resour. Publ., Highlands Ranch, Colo.

Sneath, P.H., 1957. The application of computers to taxonomy. *Microbiology*, 17(1), 201–226.

Snowling, S.D., Kramer, J.R., 2001. Evaluating modelling uncertainty for model selection. *Ecological Modelling*, 138(1-3), 17–30. https://doi.org/10.1016/S0304-3800(00)00390-2

Sobol', I., 1967. On the distribution of points in a cube and the approximate evaluation of integrals. *USSR Computational Mathematics and Mathematical Physics*, 7(4), 86–112. https://doi.org/10.1016/0041-5553(67)90144-9

Sobol', I.M., 1993. Sensitivity estimates for nonlinear mathematical models. *Mathematical Modelling and Computational Experiments*, 1(4), 407–414.

Sobol', M.I., Kucherenko, S., 2009. Derivative based global sensitivity measures and their link with global sensitivity indices. *Mathematics and Computers in Simulation*, 79(10), 3009–3017. https://doi.org/10.1016/j.matcom.2009.01.023

Sokal, R.R., 1958. A statistical method for evaluating systematic relationship. *University of Kansas science bulletin*, 28, 1409–1438.

Sokal, R.R., Rohlf, F.J., 1962. The comparison of dendrograms by objective methods. *Taxon*, 11(2), 33–40.

Song, X., Bryan, B.A., Gao, L., Zhao, G., Dong, M., 2016. Sensitivity in ecological modelling: from local to regional scales. In: Petropoulos, G. and Srivastava, P.K., editors. *Sensitivity Analysis in Earth Observation Modelling*. Elsevier.

Song, X., Zhang, J., Zhan, C., Xuan, Y., Ye, M., Xu, C., 2015. Global sensitivity analysis in hydrological modelling: Review of concepts, methods, theoretical framework, and applications. *Journal of hydrology*, 523, 739–757. https://doi.org/10.1016/j.jhydrol.2015.02.013

Sørensen, T., 1948. A method of establishing groups of equal amplitude in plant sociology based on similarity of species and its application to analyses of the vegetation on Danish commons. *Biol. Skr.*, 5, 1–34.

Sorooshian, S., Gupta, V.K., 1985. The analysis of structural identifiability: Theory and application to conceptual rainfall-runoff models. *Water Resources Research*, 21(4), 487–495. https://doi.org/10.1029/WR021i004p00487

Székely, G.J., Rizzo, M.L., 2005. A new test for multivariate normality. *Journal of Multivariate Analysis*, 93(1), 58–80. https://doi.org/10.1016/j.jmva.2003.12.002

Székely, G.J., Rizzo, M.L., 2013. Energy statistics: a class of statistics based on distances. *Journal of Statistical Planning and Inference*, 143(8), 1249–1272. https://doi.org/10.1016/j.jspi.2013.03.018

Tan, P-N, Steinbach, M., and Kumar, V., 2006. *Introduction to Data Mining.* Boston, MA, Pearson Addison Wesley.

Tang, B., 1993. Orthogonal array-based Latin hypercubes. *Journal of the American Statistical Association*, 88(424), 1392–1397. https://doi.org/10.2307/2291282

Tang, Y., Reed, P., van Werkhoven, K., Wagener, T., 2007. Advancing the identification and evaluation of distributed rainfall–runoff models using global sensitivity analysis. *Water Resources Research*, 43, W06415. https://doi.org/10.1029/2006WR005813

Tang, Y., Reed, P., Wagener, T. and Van Werkhoven, K. 2007. Comparing sensitivity analysis methods to advance lumped watershed model identification and evaluation. *Hydrology and Earth System Sciences*, 11, 793–817. https://doi.org/10.5194/hess-11-793-2007

Tarantola, S., Gatelli, D., Mara, T.A., 2006. Random balance designs for the estimation of first order global sensitivity indices. *Reliability Engineering & System Safety*, 91(6), 717–727. https://doi.org/10.1016/j.ress.2005.06.003

Terada, Y., 2013. Clustering for high-dimension, low-sample size data using distance vectors. *arXiv preprint arXiv*:1312.3386.

Tutz, G., Ramzan, S., 2015. Improved methods for the imputation of missing data by nearest neighbor methods. *Computational Statistics & Data Analysis*, 90, 84–99. https://doi.org/10.1016/j.csda.2015.04.009

van Dam, E.R., Husslage, B., den Hertog, D., Melissen, H., 2007. Maximin Latin hypercube designs in two dimensions. *Operations Research*, 55(1), 158–169. https://doi.org/doi:10.1287/opre.1060.0317

Vanrolleghem, P.A., Mannina, G., Cosenza, A. and Neumann, M.B., 2015. Global sensitivity analysis for urban water quality modelling: Terminology, convergence and comparison of different methods. *Journal of Hydrology*, 522, 339–352. https://doi.org/10.1016/j.jhydrol.2014.12.056

Vanuytrecht, E., Raes, D., Willems, P., 2014. Global sensitivity analysis of yield output from the water productivity model. *Environmental Modelling & Software*, 51, 323–332. https://doi.org/10.1016/j.envsoft.2013.10.017

Verseghy, D. L. 1991. CLASS—A Canadian land surface scheme for GCMs, I. Soil model. *International Journal of Climatology*, 11(2), 111–133, https://doi.org/10.1002/joc.3370110202

Verseghy, D. L., McFarlane, N.A., Lazare, M. 1993. CLASS— A Canadian land surface scheme for GCMs, II. Vegetation model and coupled runs. *International Journal of Climatology*, 13(4), 347–370, https://doi.org/10.1002/joc.3370130402

Verseghy, D., 2012. CLASS – the Canadian Land Surface Scheme (Version 3.6), Technical Documentation, Tech. Rep., Science and Technology Branch, Environment and Climate Change Canada, Toronto, 179 pp.

Viana, F.A., 2013. Things you wanted to know about the Latin hypercube design and were afraid to ask. In: *Proceedings of the 10th World Congress on Structural and Multidisciplinary Optimization*, Orlando, Florida.

Viana, F.A.C., Venter, G., Balabanov, V., 2010. An algorithm for fast generation of optimal Latin hypercube designs. *International Journal for Numerical Methods*, 82(2), 135–156. https://doi.org/10.1002/nme.2750

Von Bertalanffy, L., 1950a. The theory of open systems in physics and biology. *Science*, 111(2872), 23–29.

Von Bertalanffy, L., 1950b. An outline of general system theory. *British Journal for the Philosophy of science*.

Vořechovský, M., 2009. Hierarchical subset Latin hypercube sampling for correlated random vectors. In: *Proceedings of the First International Conference on Soft Computing Technology in Civil, Structural and Environmental Engineering*, Madeira, Portugal.

Vrugt, J.A., Gupta, H.V., Bouten, E., Sorooshian, S., 2003. A shuffled complex evolution Metropolis algorithm for optimization and uncertainty assessment of hydrologic model parameters. *Water Resources Research*, 39(8). https://doi.org/10.1029/2002WR001642

Vrugt, J.A., Nualláin, B.Ó., Robinson, B.A., Bouten, W., Dekker, S.C., Sloot, P.M., 2006. Application of parallel computing to stochastic parameter estimation in environmental models. *Computers & Geosciences*, 32(8), 1139–1155. https://doi.org/10.1016/j.cageo.2005.10.015

Vuyovich, C., Daly, S., Gagnon, J., Weyrick, P., Zaitsoff, M., 2009. Monitoring river ice conditions using web-based cameras. *Journal of Cold Regions Engineering*, 23(1). https://doi.org/10.1061/(ASCE)0887-381X(2009)23:1(1)

Wagener, T., Boyle, D.P., Lees, M.J., Wheater, H.S., Gupta, H.V., Sorooshian, S., 2001. A framework for development and application of hydrological models. *Hydrology and Earth System Sciences*, 5, 13–26. https://doi.org/10.5194/hess-5-13-2001

Wallis, W.D., George, J., 2011. *Introduction to Combinatorics*. CRC Press. London.

Wang, J., Li, X., Lu, L, Fang, F., 2013. Parameter sensitivity analysis of crop growth models based on the extended Fourier Amplitude Sensitivity Test method. *Environmental Modelling & Software*, 48, 171–182. https://doi.org/10.1016/j.envsoft.2013.06.007

Ward, J. H., Jr., 1963. Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association*, 236–244.

Warnock, T.T., 1972. Computational investigations of low-discrepancy pointsets, in: Zaremba, S.K. (Ed.), Applications of Number Theory to Numerical Analysis. Academic Press, New York, pp. 319–343.

Webster, M., Scott, J., Sokolov, A., Stone, P., 2004. Estimating probability distributions from complex models with bifurcations: The case of ocean circulation collapse, *Journal of Environmental Systems*, 31, 1–21, https://doi.org/10.2190/A518-W844-4193-4202

Williams, W.T., Lance, G.N., 1967. A general theory of classificatory sorting strategies I Hierarchical systems. *Computer Journal*, 9, 378–380.

Williamson, D., 2015. Exploratory ensemble designs for environmental models using k-extended Latin Hypercubes. *Environmetrics* 26, 268–283. https://doi.org/10.1002/env.2335

Xiong, F., Xiong, Y., Chen, W., Yang, S., 2009. Optimizing Latin hypercube design for sequential sampling of computer experiments. *Engineering Optimization*, 41 (8), 793–810. https://doi.org/10.1080/03052150902852999

Xiong, F., Xu, G. 2009. Numerical investigation of river ice-bridge pier interaction. *Proc., Structures 2009: Don't Mess with Structural Engineers Congress*, ASCE, https://doi.org/10.1061/41031(341)6

Yang, J., 2011. Convergence and uncertainty analyses in Monte-Carlo based sensitivity analysis. *Environmental Modelling & Softwar*e, 26(4), 444–457. https://doi.org/10.1016/j.envsoft.2010.10.007

Yassin, F., Razavi, S., Wheater, H., Sapriza-Azuri, G., Davison, B., Pietroniro, A., 2017. Enhanced identification of a hydrologic model using streamflow and satellite water storage data: a multi-criteria sensitivity analysis and optimization approach. *Hydrological Processes*, 31, 3320–3333. https://doi.org/10.1002/hyp.11267

Ye, K.Q., 1998. Orthogonal column Latin hypercubes and their application in computer experiments. *Journal of the American Statistical Association*, 93(444), 1430-1439. https://doi.org/10.2307/2670057

Ye, K.Q., Li, W., Sudjianto, A., 2000. Algorithmic construction of optimal symmetric Latin hypercube designs. *Journal of Statistical Planning and Inference*, 90(1), 145–159. https://doi.org/10.1016/S0378-3758(00)00105-1

Yi, X., Zou, R., Guo, H., 2016. Global sensitivity analysis of a three-dimensional nutrients-algae dynamic model for a large shallow lake. *Ecological Modelling*, 327, 74–84. https://doi.org/10.1016/j.ecolmodel.2016.01.005

Yilmaz, K.K., Gupta, H.V., Wagener, T., 2008. A process-based diagnostic approach to model evaluation: Application to the NWS distributed hydrologic model. *Water Resources Research*, 44(9). https://doi.org/10.1029/2007WR006716

Zadeh, F.K., Nossent, J., Sarrazin, F., Pianosi, F., van Griensven, A., Wagener, T. and Bauwens, W., 2017. Comparison of variance-based and moment-independent global sensitivity analysis approaches by application to the SWAT model. *Environmental Modelling & Software*, 91, 210–222. https://doi.org/10.1016/j.envsoft.2017.02.001

Zhan, Y., Zhang, M., 2013. Application of a combined sensitivity analysis approach on a pesticide environmental risk indicator. *Environmental Modelling & Software*, 49, 129–140. https://doi.org/10.1016/j.envsoft.2013.08.005

Zhang, J.L., Li, Y.P., Huang, G.H., Wang, C.X., Cheng, G.H., 2016. Evaluation of uncertainties in input data and parameters of a hydrological model using a Bayesian framework: a case study of a snowmelt–precipitation-driven watershed. *Journal of Hydrometeorology*, 17(8), 2333–2350. https://doi.org/10.1175/JHM-D-15-0236.1

Zhao, D., Xue, D., 2010. A comparative study of metamodelling methods considering sample quality merits. *Structural and Multidisciplinary Optimization*, 42(6), 923–938. https://doi.org/10.1007/s00158-010-0529-3

Zufelt, J., Walton, R., 2012. A river ice management plan for the Gyeong-In Ara waterway. *Proc., Cold Regions Engineering 2012: Sustainable Infrastructure Development in a Changing Cold Environment*, ASCE, https://doi.org/10.1061/9780784412473.026