MANY-OBJECTIVE GROUNDWATER MONITORING NETWORK
DESIGN USING BIAS-AWARE ENSEMBLE KALMAN
FILTERING, EVOLUTIONARY OPTIMIZATION, AND VISUAL ANALYTICS

A Dissertation in
Civil Engineering
by
Joshua Brian Kollat

© 2010 Joshua Brian Kollat

Submitted in Partial Fulfillment
of the Requirements
for the Degree of

Doctor of Philosophy

May 2010
The dissertation of Joshua Brian Kollat was reviewed and approved* by the following:

Patrick M. Reed  
Associate Professor of Civil Engineering  
Dissertation Advisor  
Chair of Committee

Christopher J. Duffy  
Professor of Civil Engineering  
High Performance Computing Minor Representative

Thorsten Wagener  
Associate Professor of Civil Engineering

Timothy W. Simpson  
Professor of Mechanical and Industrial Engineering

Peggy A. Johnson  
Department Head and Professor of Civil Engineering

*Signatures are on file in the Graduate School.
Abstract

This dissertation contributes the ASSIST (Adaptive Strategies for Sampling in Space and Time) framework for improving long-term groundwater monitoring (LTGM) decisions across space and time while accounting for the influences of systematic model errors (or predictive bias). The new framework combines Monte Carlo based contaminant flow-and-transport modeling, bias-aware ensemble Kalman filtering (EnKF), many-objective evolutionary optimization, and visual analytics-based decision support. The ASSIST framework allows decision makers to forecast the value of investments in new observations for many objectives simultaneously. Information tradeoffs are evaluated using an EnKF to forecast plume transport in space and time in the presence of uncertain and biased model predictions that are conditioned on uncertain measurement data. The goal of the ASSIST framework is to provide decision makers with a fuller understanding of the information tradeoffs they must confront when performing long-term groundwater monitoring network design.

Each chapter of this dissertation focuses on and addresses a specific challenge to LTGM network design. The scaling challenges of LTGM design are first explored in order to provide a basis for advancing the size and scope of LTGM design problems that can be effectively solved using multi-objective evolutionary algorithms (MOEAs). In addition, complex decision variable interdependencies that exist in large LTGM design problems cause traditional MOEAs to fail as problem sizes increase (defined in terms of increasing numbers of decisions and objectives). To address this, a new more robust MOEA termed the Epsilon-Dominance Hierarchical Bayesian Optimization Algorithm (ε-hBOA) was developed to learn and exploit the complex interdependencies that exist for large LTGM design problems. Building on the scalable many-objective optimization capabilities of ε-hBOA, the
ASSIST framework contributes visual analytical tools, capable of providing decision makers with an improved understanding of the complex spatial and temporal tradeoffs that often exist between their LTGM design objectives. Finally, a bias-aware EnKF framework was developed that dramatically enhances the accuracy of groundwater flow-and-transport forecasts in the presence of systematic modeling errors (or biases), while making computational innovations that again expand the size and scope of LTGM problems that can be addressed.

This dissertation demonstrates that the forecasting, search, and visualization components of the ASSIST framework combine to represent a significant advance for LTGM network design that has a strong potential to innovate our future characterization, prediction, and management of groundwater systems.
# TABLE OF CONTENTS

List of Figures ........................................... ix
List of Tables ............................................. xvi
List of Symbols ............................................. xvii
Acknowledgments ........................................... xxv

## Chapter 1
### Introduction ........................................... 1
1.1 Observation Network Design .......................... 1
1.2 Long-Term Groundwater Monitoring Design ............. 3
1.3 Prior Work ............................................ 3
1.4 Scope and Overview .................................... 8
  1.4.1 Chapter 2 ........................................... 9
  1.4.2 Chapter 3 ........................................... 9
  1.4.3 Chapter 4 .......................................... 10
  1.4.4 Chapter 5 .......................................... 10
  1.4.5 Chapter 6 .......................................... 11
  1.4.6 Chapter 7 .......................................... 12
  1.4.7 Chapter 8 .......................................... 12

## Chapter 2
### Background ............................................ 13
2.1 Many-objective Optimization .......................... 13
  2.1.1 Multiobjective Evolutionary Algorithms .......... 16
Chapter 5
A Framework for Visually Interactive Decision-Making and Design Using Evolutionary Multiobjective Optimization (VIDEO) 87
5.1 Abstract 87
5.2 Introduction 88
5.3 Methodology 90
5.3.1 A Priori and A Posteriori Approaches 90
5.3.2 VIDEO Framework Components 91
5.3.2.1 Overview 91
5.3.2.2 Generating Multiobjective Solution Sets 93
5.3.2.3 Interactive Visualization 93
5.3.3 Test Case Description 97
5.3.3.1 Design Objectives 97
5.3.3.2 Spatial Interpolation 98
5.3.3.3 Generating the Pareto-set 98
5.4 VIDEO Case Study 99
5.4.1 Exploring and Understanding Objective Tradeoffs 99
5.4.2 Exploring and Understanding Designs 101
5.4.3 Negotiated Design Selection 105
5.5 Discussion 110
5.6 Conclusions 112

Chapter 6
Addressing Model Bias and Uncertainty in Three Dimensional Groundwater Transport Forecasts for a Physical Aquifer Experiment 113
6.1 Abstract 113
6.2 Introduction 114
6.3 Physical Aquifer Experiment 115
Chapter 7

Many-Objective Groundwater Monitoring Network Design Using Bias-Aware Ensemble Kalman Filtering, Evolutionary Optimization, and Visual Analytics

7.1 Abstract ................................. 124
7.2 Introduction ............................... 125
7.3 Prior Work ............................... 126
7.4 Methodology .............................. 127
  7.4.1 Physical Aquifer Experiment ............ 127
  7.4.2 Numerical Flow and Transport Modeling 128
  7.4.3 Bias-Aware Ensemble Kalman Filtering .... 130
  7.4.4 Optimization Algorithm .................. 130
  7.4.5 Design Objectives ..................... 131
7.5 Computational Experiment ...................... 137
  7.5.1 Flow and Transport Using ParFlow and SLIM 137
  7.5.2 Parameterization of EnKF ................ 139
  7.5.3 Parameterization of Many-Objective Search 140
7.6 Results and Discussion ......................... 141
  7.6.1 The Forecasted Observation Design Tradeoffs 141
  7.6.2 Exploring Lower Dimensional Problems .... 143
  7.6.3 Exploration to Inform Decision Making .... 147
  7.6.4 The Costs of Compromise ............... 149
7.7 Conclusions ............................... 153

Chapter 8

Conclusions, Contributions, and Future Work

8.1 Conclusions ............................... 155
8.2 Contributions .............................. 157
8.3 Future Work ............................... 158

Bibliography ................................. 163
## LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Two objective example illustrating dominance, nondominance, and Pareto-optimality.</td>
<td>15</td>
</tr>
<tr>
<td>2.2</td>
<td>Schematic diagram of the $\varepsilon$-NSGAII illustrated using the notation of Deb et al. [101]. This figure shows the “connected runs” and dynamic population sizing concepts of the $\varepsilon$-NSGAII. In this figure, $p$ represents population size and $e$ represents the $\varepsilon$-nondominated archive size.</td>
<td>19</td>
</tr>
<tr>
<td>2.3</td>
<td>Illustration of the $\varepsilon$-dominance concept.</td>
<td>20</td>
</tr>
<tr>
<td>3.1</td>
<td>Cross-sectional slice of the PCE contamination plume representing the LTGM test case. There are 29 well locations available for sampling with one to three sampling locations available along each well’s vertical axes. Wells eliminated to create each of the 18 through 25 well test cases are shown in the figure.</td>
<td>41</td>
</tr>
<tr>
<td>3.2</td>
<td>Example calculation of the $\varepsilon$-performance metric.</td>
<td>43</td>
</tr>
<tr>
<td>3.3</td>
<td>Number of Pareto-optimal solutions versus problem size for each of the 18 through 25 well test cases.</td>
<td>44</td>
</tr>
<tr>
<td>3.4</td>
<td>Four objective Pareto fronts for the 18 and 25 well test cases. Plots A and C show the actual Pareto surface and plots B and D show the planar projections of the Cost, Error, and Uncert objectives onto the planes formed by the X, Y, and Z coordinate axes. Mass error is represented by the color of the markers.</td>
<td>45</td>
</tr>
</tbody>
</table>
3.5 Computational scaling results for the $\varepsilon$-NSGAII applied to the 18 through 25 well LTGM test cases. Also shown are the results attained using the $\varepsilon$-dominance concept to approximate the Pareto set. The $\varepsilon$-NSGAII’s computational demands increase approximately quadratically $[O(l^2)]$ where $l$ is problem size] when solving successively larger LTGM problems. However, using $\varepsilon$-dominance to constrain the Pareto-set size to approximately 500 solutions results in approximately linear scaling $[O(l)]$ of computational demands with increasing problem size. ........................................... 47

3.6 Using $\varepsilon$-dominance to control Pareto set size of the 25 well test case. Starting at the full Pareto-optimal set shown in plot A, successively relaxing the precision requirements of the problem through the manipulation of the $\varepsilon$-dominance parameters for each objective (shown in square brackets for plots B through D), reduces the set size while maintaining geometric similarity to the full Pareto set. .............. 49

3.7 Percent computational savings (circle markers) and percent reduction in variability (star markers) resulting from the use of $\varepsilon$-dominance to control Pareto set size. .......................... 50

3.8 Dynamic algorithm performance for the 18 through 25 well test cases. These plots show the mean $\varepsilon$-performance of the 50 random seed trial runs for each test case plotted versus function evaluations. Plot A shows the dynamic performance of the $\varepsilon$-NSGAII when searching for the full Pareto-optimal set and plot B shows the dynamic performance of the algorithm when $\varepsilon$-dominance is used to approximate the set. .......................... 52

4.1 Potential conditional dependencies of sampling wells for a contamination plume. Figure A demonstrates the case when all wells are sampled independently and Figure B illustrates the case where conditional probabilities may exist in sampling various wells. ............ 63

4.2 Schematic diagram of the $\varepsilon$-hBOA illustrated using the notation of Deb et al. [101]. This figure shows the connected runs and dynamic population sizing concepts of the $\varepsilon$-NSGAII combined with Bayesian network model building and simulation (both features of the hBOA). In the figure, $p$ represents population size and $e$ represents $\varepsilon$-nondominance archive size. .......................... 65
4.3 Plots A and B show the 25 well Pareto-set and 58 point Pareto-approximation respectively. Plot C shows an enlarged view of a densely populated area of the 58 point Pareto-approximation. Cost, Error, and Uncert are plotted on the X, Y, and Z-axes, and Mass is plotted using color.

4.4 Histogram displaying the relative contributions of each algorithm configuration to the 58 point test case reference set. Light gray bars indicate total contribution across all random seed trials, medium gray bars, the average per seed contribution, and the black bars, unique contribution across all seeds.

4.5 Epsilon-performance success rates plotted versus function evaluations (NFE) at specified thresholds for both the 25 well and 58 point test cases.

4.6 Convergence and ε-performance dynamics for the ε-NSGAII, ε-hBOA-Base, and ε-hBOA-Static configurations. Plots show each of the 50 random seed trials for each configuration plotted as metric value versus function evaluations (NFE).

4.7 Pareto approximation set generated by combining the best trial runs of each the ε-NSGAII, ε-hBOA-Base, and ε-hBOA-Static configurations. Cost, Error, and Uncert are plotted on the X, Y, and Z-axes while the Mass objective is portrayed by the orientation of the solution cone (up = high Mass, down = low Mass). Red solutions were contribution by the ε-NSGAII, green solutions by the ε-hBOA-Base, and blue solutions by the ε-hBOA-Static configuration.

5.1 Snapshot of VIDEO decision making software. The interface is divided into an objective space window, a design space window, and a set of tools associated with each of these windows.

5.2 Examples of how a four-objective Pareto-optimal solution set can be viewed within the VIDEO framework. Panel A shows the full four objective Pareto set with Cost, Error, and Uncert, plotted on X, Y, and Z, and Mass plotted as color. Panel B reverses the Uncert and Mass objectives such that Mass is plotted on Z and Uncert plotted using color. Panel C shows the two objective subset Cost-Error and panels D and E show this subset in the context of the full four objective set. Panel F shows how a third objective (Cost) can be plotted using color to reveal additional decision maker insight.
5.3 Example of solution comparison and decision space probing functionality within the VIDEO framework. Panels A and F show alternative views of the locations of the two selected solutions in the objective space. Panels B and G show the associated plume estimates at \( Z = 85 \) m. Panels C and H show an iso-surface plotted at concentration = 0.70. Panels D and I display the estimation error at \( Z = 85 \) m. Panels E and J show the spatial extent of estimation uncertainty at \( Z = 85 \) m.

5.4 Negotiated design selection within the VIDEO framework using two-objective subsets of the larger four-objective problem. Panel A shows the tradeoff \( \text{Cost} - \text{Error} \) marked in red and panel B, the tradeoff \( \text{Cost} - \text{Uncert} \) marked in green. Panel C shows the other two objective subsets \( \text{Cost} - \text{Mass}, \text{Error} - \text{Mass}, \) and \( \text{Uncert} - \text{Mass} \) marked in blue, purple, and orange respectively. Panel D shows the marked two objective subsets in the context of the full four objective space. Panel E shows the selection of three solutions of interest to be explored more fully in the negotiated design selection section.

5.5 VIDEO comparison of solutions 1 through 3, which were selected for further consideration. Columns 1 through 4 of the figure represent the concentration estimates at \( Z = 85 \) m, a concentration iso-surface at 0.70, and Kriging error and uncertainty at \( Z = 85 \) m respectively. The first row of the figure represents the maps associated with the highest cost solution and the remaining rows, the maps associated with solutions 1 through 3 respectively.

6.1 Diagram of the UVM physical tank experiment’s dimensions, media layering and average calibrated K values, sensor and well locations, and constant head reservoirs.
6.2 Panel A shows the tracer breakthrough curves associated with the sampling ports in layer 4 plotted as normalized concentration, \( c/c_0 \), versus time for the model, observed, and filtered concentration time-series. Panel B shows the BT curves associated with the tracer source location (port B4) for the various assimilation cases (2 hours, 6 hours, 12 hours, and 24 hours) where the gray shading represents one standard deviation above and below the ensemble mean. Panel C provides normalized tracer concentration maps (two top views of the layer 4 and layer 3 sampling ports, a side view along the center of the tank, and a 3 dimensional view) provided by the model at \( t = 13 \) days. In addition, the coloring of the sampling ports indicates observed \( c/c_0 \) concentrations at this time step. Panel C shows the numerically modeled mean tracer concentration at \( t = 13 \) days. Panel D shows the mean filtered maps for assimilation every 2 hours at \( t = 13 \) days (formatted similarly to Panel C for comparison).

7.1 Top view of UVM physical aquifer tank with breakthrough curves of normalized tracer concentration versus time plotted at the approximate geometric location of each monitoring well. The dashed blue curves show the high resolution tracer concentration observations while the solid green curves show the simulated tracer concentration using ParFlow and SLIM-FAST for monitoring ports located in media layer 4. Note that the entire 19 year simulation period is shown.

7.2 The resulting forecasted six-objective Pareto-approximate set generated by the \( \varepsilon \)-hBOA. \( Cost, DF, \) and \( K \) objectives are plotted on the X, Y, and Z axes respectively. The \( Flux \) objective is shown by the color of the cones, \( Mass \) by the orientation of the cones, and \( Centroid \) by the size of the cones as is indicated in the legend.

7.3 Projections of the full six objective solution set onto a sampling of two objective subproblems. Tradeoffs between the two objective subproblems are highlighted with a solid red line. Two interesting breakpoint solutions are annotated using stars. The locations of low \( Mass \) and \( Centroid \) error are highlighted in each subproblem using a light blue circle/ellipse. An interesting \( Cost \) threshold is annotated using a dashed magenta line.
7.4 Identification of five interesting solutions from the full six-objective Pareto-approximate forecast set. This figure represents the possible exploratory analysis that a decision maker might undergo to identify interesting compromise solutions that capture multiple system properties simultaneously.

7.5 Sampling plans associated with solutions 2 and 5 identified in Section 7.6.3. Breakthrough curves are shown for the prior management period \( k - 1 \) years 5 through 6 and the forecast period \( k \) years 6 through 7. Breakthrough curves are shown similarly to Figure 7.1, except now the assimilation and forecast curve of the EnKF is shown using a dash-dot red curve. The decision of whether or not to sample at each quarter is indicated by either a filled or open circle respectively. Note that the time scale shown in this figure represents years 5 through 7 of the simulation.

7.6 Snapshots of tracer plume at year 7 (the end of the current management period). Panel A shows the plume snapshot obtained by assimilating all available samples in the prior and current management period (a maximum information baseline). Panel B shows the plume obtained by the biased flow and transport model. Panel C shows the plume obtained by assimilating available samples from the prior management period into the EnKF and then forecasting only for the current management period. Panel D shows the plume obtained by assimilating that data that would be available from the sampling plan associated with solution 5. Panels E through F show the relative error between the plumes shown on panels B through D and the maximum data plume shown in panel A where red indicates high error and grey, low error.

8.1 This figures demonstrates how sampling rules might be developed from the hierarchical Bayesian network models that are generated by the \( \varepsilon \)-hBOA algorithm during optimization of LGTM design problems. Panel A shows high probability sampling rules for the 25 well LTGM test case explored in Chapters 3 through 5. Panel B shows how these rules might be applied to inform both optimization and decision making. For example, if well 18 is not chosen for sampling, then well 20 should be chosen.
8.2 Example of how the information available in the Kalman gain matrix might be used to inform decision making for the UVM tank experiment (from Chapters 6 and 7. Shaded regions in the plots show the impacts that individual sampling decisions have on the performance of the EnKF. For example, in panel A, sampling from well B4 impacts the performance of the EnKF in the immediate vicinity of this well, but also affects its performance at the opposite end of the tank near the leading edge of the tracer plume. . . . . 161
LIST OF TABLES

1.1 Historical summary of relevant LTGM literature to date. The classification of each study’s design approach, methodology, and objective formulation is included. ............................................. 7

3.1 Enumeration data for the 18 through 25 well test cases. Data shown includes the number of possible solutions, the number of Pareto-optimal solutions, and the percentage of the search space which was infeasible. ............................................................ 46

3.2 Epsilon settings used to approximate the Pareto-optimal sets of the 20 through 25 well test cases. The original $\varepsilon$ settings used for the 18 and 19 well test cases are shown as well. .......................... 48

4.1 NFE = 25,000 and end-of-run mean and (standard deviation) metric results for the 25 well test case. Metric values in bold text indicate statistical superiority at a 95-percent confidence level. . . . 75

4.2 NFE = 250,000 and end-of-run mean and (standard deviation) metric results for the 28 point test case. Metric values in bold text indicate statistical superiority at a 95-percent confidence level. . . . 76

5.1 Summary of objective values expressed as a percentage of the maximum for each of the three solutions chosen as negotiated designs . 108

7.1 Summary of media properties used to simulate permeability field realizations. Hydraulic conductivity values (as well as their corresponding standard deviations) are expressed in units of cm-hr$^{-1}$. . . 138

7.2 Five interesting compromise solutions that might be selected by a decision maker during exploration of the six objective Pareto-approximate forecast set. ......................................................... 149
List of Symbols

Acronyms

ASSIST  Adaptive Strategies for Sampling in Space and Time

DM  Decision Maker

EnKF  Ensemble Kalman Filter

hBOA  Hierarchical Bayesian Optimization Algorithm

$\varepsilon$-hBOA  Epsilon-dominance Hierarchical Bayesian Optimization Algorithm

KF  Kalman Filter

LTGM  Long-Term Groundwater Monitoring

MOEA  Multiobjective Evolutionary Algorithm

$\varepsilon$MOEA  Epsilon-dominance Multiobjective Evolutionary Algorithm

NSGAII  Nondominated Sorted Genetic Algorithm II
ε-NSGAII  Epsilon-nondominated Sorted Genetic Algorithm II

OF    Objective Function

QK    Quantile Kriging

SPEA2  Strength Pareto Evolutionary Algorithm 2

VIDEO  Visually Interactive Decision Making and Design using Evolutionary Optimization

**Notation Index**

1\(N\)  EnKF - Matrix containing 1\(N\) as its elements (\(N \times N\))

\(a\)  EnKF - Denotes data assimilation step

\(A\)  EnKF - Ensemble of model \(\psi\) and bias \(\eta\) states (\(2n \times N\))

\(\bar{A}\)  EnKF - Ensemble mean of \(A\) (\(2n \times N\))

\(A'\)  EnKF - Perturbations on \(A\) (\(2n \times N\))

\(b\)  MOEA - Building block size

\(c\)  OF - Contaminant concentration [mass][volume\(^{-1}\)]

\(C\)  OF - Cost vector for each sample (\(1 \times M\))
$d$ EnKF, QK - Data measurement

$d^*$ QK - Quantile transformed data measurement

$D$ EnKF - Ensemble of perturbed samples ($m \times N$)

$\text{DI}$ OF - Detection indicator vector ($1 \times M$)

$e$ MOEA - Epsilon-nondominated archive size

$f$ EnKF - As subscript denotes EnKF forecast step

$f_{\text{Centroid}}$ OF - Plume centroid error minimization [length]

$f_{\text{Cost}}$ OF - Sampling cost minimization

$f_{\text{Error}}$ OF - QK estimation error minimization ($[\text{mass}][\text{volume}^{-1}])^2$

$f_{\text{DF}}$ OF - Detection failure minimization [percentage]

$f_{\text{Flux}}$ OF - Normalized flux detection maximization [length][time$^{-1}$]

$f_{K}$ OF - EnKF information maximization

$f_{\text{Mass}}$ OF - Plume mass error minimization [percentage]

$f_{\text{Uncert}}$ OF - QK estimation uncertainty minimization [mass][volume$^{-1}$]

$F$ OF - Number of design objectives
F  OF - Design objective vector

g  EnKF, QK - Grid location index

G  QK - Number of QK estimation points

H  EnKF - Mapping of samples to model states \((m \times 2n)\)

i  EnKF, QK, OF - Index

I  EnKF - Identity matrix

j  EnKF - time step

k  EnKF - Management period index

K  OF - Hydraulic conductivity \([\text{length}][\text{time}^{-1}]\)

K  EnKF - Kalman Gain matrix \((2n \times m)\)

l  MOEA - Length of LTGM decision string

m  EnKF - Number of locations used in a sampling plan

M  EnKF, QK, OF, MOEA - Total number of sampling locations

M  EnKF - Non-linear model operator

n  EnKF - Number of model states
\( N \) EnKF - Number of ensemble members

\( o \) MOEA - Number of nondominated solution fronts

\( O \) MOEA - Computational complexity “Big-O” notation

\( p \) MOEA - Population size

\( P \) EnKF - Covariance of \( \mathbf{A} \) \((2n \times 2n)\)

\( q \) OF - Darcy flux

\( Q \) EnKF - Spatially correlated noise matrix \((n \times N)\)

\( QK^* \) QK - Estimate at a location

\( r \) MOEA - Number of runs

\( R \) EnKF - Covariance of \( \Upsilon \) \((m \times m)\)

\( \mathbb{R} \) Real number space

\( s \) EnKF, QK, OF - Sampling location index

\( t \) EnKF, OF - Sample time within management period

\( T \) EnKF, OF - Total sampling times in management period

\( u \) EnKF, QK, OF - A sampling location
$U$ QK, OF - The number of unestimated points resulting from QK

$V$ OF - Volume [length$^3$]

$W$ EnKF - Sequential Gaussian simulated bias states ($n \times N$)

$x$ OF - Binary yes/no decision for a sample

$x$ OF - Sampling decision vector ($1 \times MT$)

$X$ Euclidean X direction/distance

$Y$ Euclidean Y direction/distance

$Z$ Euclidean Z direction/distance

$\alpha_c$ MOEA - Probability of crossover

$\alpha_m$ MOEA - Probability of mutation

$\gamma^*$ QK - Quantile semivariogram representation of spatial correlation

$\delta$ EnKF - Scaling factor for ensemble noise matrix $Q$

$\Delta$ MOEA - Termination criteria

$\varepsilon$ MOEA - Epsilon precision of design objective

$\sigma$ EnKF, QK, OF - Standard deviation
\( \tau_c \) MOEA - Crossover distribution index

\( \tau_m \) MOEA - Mutation distribution index

\( \phi \) OF - Media porosity

\( \omega \) OF - Sampling decision

\( \mu_0 \) OF - Zeroth plume moment [mass]

\( \mu'_0 \) OF - Reference plume zeroth moment [mass]

\( \mu_1 \) OF - First plume moment [coordinate location]

\( \mu'_1 \) OF - Reference plume first moment [coordinate location]

\( \eta \) EnKF - Vector of bias states \((1 \times n)\)

\( \psi \) EnKF - Vector of model states \((1 \times n)\)

\( \Psi \) QK - Vector of true system states \((1 \times n)\)

\( \lambda \) QK - Kriging weights that minimize error variance

\( \Lambda \) EnKF - Time correlation factor for model bias

\( \Omega \) OF - The entire sampling decision space

\( \nu \) EnKF - Errors on data measurements \(d\)
EnKF - Perturbations on sample matrix $D (m \times N)$

$\| \cdot \|_1$ OF - One norm calculation
Acknowledgments

This work represents several years of effort and was only possible through the support of numerous faculty, co-workers, friends, and family. First and foremost, I would like to gratefully thank my advisor, Dr. Patrick Reed, Associate Professor of Civil Engineering at Penn State University. Dr. Reed not only provided academic support throughout my tenure as a graduate student, but also acted as one of my most trusted mentors. I would also like to thank the members of my dissertation committee, Chris Duffy, Thorsten Wagener, and Timothy Simpson for their time spent providing valuable feedback throughout this entire process. It has truly been a pleasure working with all of you.

There are various other individuals who I would like to personally thank for their support and contributions throughout my work. Ray Masters, retired professor and visualization expert, for joyfully passing along much of his visualization programming expertise. Art and Mary Miller of AECOM Water for providing me the opportunity to move beyond degree-seeking and into consulting. Matthew Ferringer of The Aerospace Corporation for his continued collaboration and support and whose innovations are moving much of the work presented in this dissertation into real-world decision-making and design. The U.S. EPA for supporting the last three years of my research through a Science to Achieve Results (STAR) Graduate Research Fellowship. Martin Pelikan, the original developer of the Hierarchical Bayesian Optimization Algorithm, for providing the source code required to make this study possible. Reed Maxwell, Associate Professor at the Colorado School of Mines, for providing source code and support for Parflow and SLIM. And finally, Donna Rizzo, Associate Professor at the University of Vermont who provided access to the UVM scaled aquifer tank experimental data.

Lastly, I would like to thank my friends and colleagues, Joe Kasprzyk, Ruchit Shah, and Yong Tang, for providing both stimulating discussion and moral support throughout my graduate work. Also to my friends, Rob and Rachel Johnson, Ian
and Kathleen Reese, Denis and Sandra Gerstorf, the Roeber family, Bryon May, Sue and John Hiester, Peg and Don Hambrick, Ron Quinn, and Sally Lenker. I am also deeply thankful for the love and support I have always received from my parents and family. I certainly would not be where I am today without all of you. Last but certainly not least, I would like to thank my wife, Sarah, for her unwavering love and support throughout this long endeavor. Without you, none of this would have been possible. Every day with you is a blessing from God, through Whom all things are possible.

Disclaimer: The author of this work was funded in part by a Science to Achieve Results (STAR) Graduate Research Fellowship (agreement No. FP-916820) awarded by the U.S. Environmental Protection Agency (EPA). This dissertation has not been formally reviewed by the EPA. In addition, the views expressed in this document are solely those of the author and the EPA does not endorse any products or commercial services mentioned in this dissertation.
Dedication

To my loving wife Sarah and our wonderful children
Elvis, Katarina, and Elisabete

Manai mīļajai sievai Sārai un mūsu brīnišķīgajiem bērniem
Elvim, Katarinai, un Elizabetei
Evaluating and addressing human impacts on environmental systems represents one of the most challenging and significant scientific problems of the present time. Characterizing human impacts to hydrologic systems is of particular importance since growing pressures on our fresh water resources pose a significant threat to their long-term sustainability. The decisions we make in the present and near future to characterize and mitigate these impacts will have far reaching economic and ecological consequences. Increased scientific interest in the long-term quality and sustainability of numerous environmental resources has instigated a shift towards “big science” initiatives and significant federal investment in improving our ability to characterize impacted environmental systems [1–8]. These initiatives and investments will lead to innovations in observation systems that will enable and enhance the collection and transmission of increasingly more complex environmental data in near real-time [9]. There is broad interest in optimally equipping environmental systems with observation networks (i.e., networks of sensors) in the hopes that they will provide improved insights into impacted systems’ states and responses.

### 1.1 Observation Network Design

Historically speaking, streamflow and precipitation network design are two of the oldest domains of hydrologic data collection [10]. However, recent observatory
design initiatives have considered a variety of hydrologic variables such as evapotranspiration rates, soil moisture, groundwater levels, surface water and groundwater quality, and subsurface contamination [8, 9, 11]. Huge investments of time and money as well as the goal of characterizing and improving our understanding of complex environmental systems make understanding the long-term costs of managing and improving observation networks a problem of critical importance.

Formal observation (or monitoring) network design has a long history dating back to the pioneering work of Drozdov and Sepelevskij [12] who developed a formalized framework for evaluating the spatial coverage of meteorological gauge networks. Early work in observation network design is strongly dominated by the meteorological community (for a review see Arnold and Dey [13]) beginning in the 1950’s and continuing to the present using the concept of observation system simulation events (OSSEs) in which the value of observables are forecasted in terms of their ability to enhance system predictions. The meteorological OSSE formalizes the use of predictive models and statistical data assimilation to discover the need for and benefits of new observation technologies. The classical OSSE combines predictive simulation and Bayesian data assimilation to forecast the value of observations. This literature poses an important exemplar for the scientific importance of developing feedbacks between observation and simulation systems, each of which benefit from the rigorous evaluation of predictive skill using forecasts of actual system events.

Observation network design is governed by the needs of diverse groups of stakeholders with potentially conflicting goals and priorities [14, 15]. This motivates significant investment of both time and money, with the ultimate goal of producing high quality environmental data while balancing the efficiency and effectiveness of data collection efforts [16]. Constant inflow of new observations, as well as innovations in modeling strategies and technology over time requires adaptive management techniques and design strategies that ensure optimal data quality over long time scales [11, 15]. Moss [15] noted that the “ideal” hydrologic network represents a balance between understanding the nature of the physical process of interest and understanding the decisions that this data will impact while accounting for uncertainty in both.
1.2 Long-Term Groundwater Monitoring Design

Groundwater resources play a vital role in the support and sustainability of human and ecological services. In the United States, the growing concerns related to the sustainability of our national groundwater resources are strongly highlighted in a recent Advisory Committee on Water Information (ACWI) proposal [17] for the establishment of a national groundwater monitoring network similar to that available for streamflow. The ACWI proposal presents a vital and severe long-term monitoring design challenge. The expense of groundwater flow-and-transport observations combined with the locality of aquifers’ governing processes motivated the ACWI Subcommittee on Ground Water to propose that the national groundwater monitoring network be developed as an emergent “network-of-networks” bridging local, state, and regional efforts to form a national scale observation system. Although groundwater plays a significant role in droughts, climate change, ecological services and the risks posed by contaminants [18–23], our groundwater monitoring design approaches remain largely *ad hoc* and limited in their usage of formal network design frameworks. Evaluating and managing the long-term value of groundwater observations from an emergent network-of-networks raises a fundamental question: How should we evaluate the value of new or existing observables while balancing a broad suite of monitoring objectives that vary with scale, focus, and stakeholder? There is a fundamental need to advance our ability to successfully forecast the value of observables and this dissertation makes significant contributions toward that goal.

1.3 Prior Work

Observation network design has been a significant focus of the water resources research literature as evidenced by the inaugural volume of the journal *Water Resources Research* (see Fiering [24]). However, hydrological observation network design frameworks have lagged behind the formality and innovations provided by the meteorological OSSEs. Langbein’s [16] summary of one of the most comprehensive efforts in the water resources literature to focus on the science of observation network design provides cogent criticisms and challenges to the state of hydro-
logical science at present. There are few examples of OSSE type hydrological experiments where forecasts of system dynamics are used to inform subsequent laboratory or field-based experimental design. Instead, the dominant approach is ad hoc observation and post-event analysis. Moss [25] highlights that our ability to understand the space and time tradeoffs implicit to hydrological observation network design requires the consideration of a third fundamental dimension for the problem - model errors. Systematic errors in our models of hydrological systems provide a barrier to our ability to utilize OSSE frameworks to advance our observation networks. Exacerbating this barrier, Lettenmaier [14] highlights that as a problem class, observation network design poses a curse of dimensionality where there are large numbers of objectives and uses for data as well as exponentially scaled growth rates for the range of alternative space and time decisions that can be considered.

Early network design literature predominately focused on assessing streamflow and precipitation networks (e.g., see the reviews in [14–16, 26]). The growth of the environmental movement and the promulgation of the U.S. Clean Water Act motivated an increased focus on characterizing and managing contaminated groundwater resources in the 1980s. Historically, LTGM network design problems have fallen into two basic categories: (i) hydrogeological approaches and (ii) statistical approaches, which were formally identified and characterized by Loaiciga et al. [27]. Hydrogeological approaches use an expert’s knowledge of the underlying hydrogeology of a site to design a monitoring network capable of achieving goals set forth by scientists, regulators, and/or various other stakeholders [27]. These goals could include characterizing groundwater quality variations, detection of contamination, regulatory compliance regarding contaminant remediation, or research [27].

Statistical approaches differ from hydrogeological approaches by realizing the uncertainty associated with our knowledge of the underlying hydrogeology at a site, hence treating system properties as random or spatially correlated variables. The often heterogenous nature of subsurface properties and our inability to fully characterize these properties motivates accounting for this uncertainty, especially when actual field data regarding these properties may be sparse or non-existent at a site. In support of this realization, geostatistical frameworks became a dominant
technical focus within the LGTM design literature [28–32]. These studies largely focused on the ability of Kriging frameworks to provide spatial measures of the value of new groundwater contaminant observations in reducing the interpolation scheme’s estimation variance.

Loaiciga et al. [27] further divided statistical approaches into simulation based approaches, variance-based approaches, and risk (or probability) based approaches. Simulation based approaches account for uncertainty by treating hydrogeologic properties as realizations of random fields, which may or may not be spatially correlated. Modeling is conducted based on these realizations to generate a distribution of potential outcomes, in an attempt to account for the uncertainty associated with our knowledge of the subsurface properties. Variance based approaches encompass a variety of methodologies ranging from reducing uncertainty through the systematic addition of sampling points, to optimization of a sampling network based on a set of design objectives. Risk-based approaches attempt to incorporate the importance of the magnitude of a variable into the network design (e.g., detecting whether or not a contaminant exists at a dangerous level at various locations throughout the sampling domain).

LTGM design has and continues to predominantly focus on the goals of characterizing new sites that have limited data, augmenting of existing networks based on evaluations of perceived observation gaps, and the reduction of redundant monitoring data. In new site design problems, a monitoring network is designed from little to no existing network data (e.g., contamination is present but little else is known). The augmentation problem consists of adding sampling points to an existing network with the goal of better characterizing the data of interest. Redundancy analysis attempts to reduce costs by eliminating sampling points which contribute little to the overall understanding of a monitored system.

The LTGM literature has demonstrated a range of methodological approaches that include expert-based hydrogeologic judgments [33, 34], deterministic physics-based simulation [35–41], geostatistical interpolation approaches (see the review by Loaiciga et al. [27]), Monte Carlo physics-based simulation [42–51], and statistical filtering/data assimilation [41, 52–56]. In many studies, an attempt is made to draw on the strengths of multiple approaches. For instance, the work of Hudak and Loaiciga [34], combined both qualitative (hydrogeological) and quantitative
(statistical) design concepts to detect contamination in multilayered strata. In the work by Meyer and Brill [43], Monte Carlo simulation was combined with risk-based approaches to design an optimal network for maximizing plume detection. Storck et al. [47], combined simulation with risk-based analysis (plume detection) to generate an ensemble of landfill failure scenarios. Additionally, they extended their network optimization framework to include multiple design objectives which included maximizing detection probability, minimizing the extent of contamination and minimizing the cost of sampling. More recently, there are an emerging number of LTGM applications that use machine learning techniques for augmentation and redundancy analysis (e.g., support vector machines [57, 58] and relevance vector machines [59]).

Both quantitative and qualitative objectives have been used in combination with the above methodologies to judge the value of LTGM observables. Beyond monitoring costs, the LTGM objectives that have been employed in prior studies can be broadly grouped into four categories: (1) parameter identification [35, 36, 39, 51, 60], (2) contaminant detection [42, 43, 47, 49, 50, 56], (3) estimation of contaminant plume moments in space and/or time [40, 43, 44, 47, 49, 61–64], and (4) minimizing the variance of groundwater flow or transport predictions [41, 54, 56–59, 65–68]. Although the prior literature has predominately focused on spatial network decisions, the use of Kalman filtering has yielded notable exceptions where spatial and temporal monitoring decisions are considered simultaneously [41, 52, 54–56, 69]. These Kalman filtering applications represent important advances in LTGM that more directly address the statistical nonstationarity of groundwater heads and contaminant concentrations that strongly limit the value of geostatistical frameworks in dynamical systems. However, traditional Kalman filter based LTGM approaches are still limited in their representation of the uncertainties and biases associated with groundwater observations and models.

Table 1.1 contains an exhaustive summary of most relevant LTGM literature published to date. Each study is characterized in terms of the design approach utilized (i.e., new site characterization, augmentation of an existing network, or redundancy reduction). The methodology utilized by each study is also categorized into hydrogeological approaches, deterministic flow and transport approaches, geostatistical techniques, Monte Carlo sampling approaches, machine learning,
<table>
<thead>
<tr>
<th>Study</th>
<th>Design</th>
<th>Methodology</th>
<th>Objectives</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Char</td>
<td>Aug</td>
<td>Red</td>
</tr>
<tr>
<td>Hughes and Lettenmeier (1981)</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Olea (1984)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bogardi et al (1985)</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rouhani (1985)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Massman and Freeze (1987a, b)</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Knopman and Voss (1988a, b)</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Christakos and Olea (1989)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rouhani and Hall (1988)</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Meyer and Brill (1988)</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Loaiciga (1989)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Knopman and Voss (1989)</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Andricovice (1990)</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cleveland and Yeh (1990)</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Knopman and Voss (1991)</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Yangxiao et al (1991)</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cleveland and Yeh (1991)</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Wilson et al (1992)</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>McKinney and Loucks (1992)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hudak and Loaiciga (1992)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hudak and Loaiciga (1993)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Christakos and Killam (1993)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Meyer et al (1994)</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>James and Gorelick (1994)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Wagner (1995)</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Gieniewski et al. (1995)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Storck et al. (1997)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mahar and Datta (1997)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rizzo et al. (2000)</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reed et al. (2000)</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Montas et al. (2000)</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Asa et al. (2004)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nunes et al. (2004a,b)</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reed and Minsker (2004)</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Nunes et al. (2005)</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Herrera and Pinder (2005)</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Wu et al. (2005)</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Zhang and Pinder (2005)</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Hudak (2006)</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Yeh et al. (2006)</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dhar and Datta (2007)</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Kim and Lee (2007)</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sun and Yeh (2007)</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ammar et al. (2008)</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Chalalaava and Datta (2008)</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Kokou and Pinder (2009)</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Kollat (2010) (This Dissertation)</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
</tbody>
</table>

**Key:**
- Geo = Hydrogeologic Approach
- Char = Characterization
- Aug = Augmentation
- Red = Redundancy
- Sim = Deterministic flow/transport
- Geostat = Geostatistics/Kriging/spatial functions
- MC = Monte Carlo/probabilistic
- ML = Machine Learning
- Filter = Bayesian filters (KF and EnKF; other close variants)
- Param = Parameter identification
- Det = Detection of contamination
- Mom = Characterize plume moments
- Var = Reduction of variance
- S = spatial network objectives
- T = temporal network objectives

**Table 1.1.** Historical summary of relevant LTGM literature to date. The classification of each study’s design approach, methodology, and objective formulation is included.
and/or Bayesian statistical filtering. Additionally, design objectives are classified as parameter identification, contaminant detection, characterization of contaminant plume spatial moments, and variance minimization objectives and are also classified as focusing on spatial and/or temporal system components. The work presented in this dissertation represents a significant contribution toward effectively combining a wide variety of existing approaches under one comprehensive and flexible framework (i.e., the design approaches, methodologies, and objectives of Table 1.1 can be considered separately or simultaneously).

1.4 Scope and Overview

While this dissertation focuses specifically on advancing long-term groundwater monitoring (LTGM) network design, the work makes significant contributions toward observation network design in general. It has recently been noted that advancement of real-time monitoring and prediction of complex environmental systems will require that observation networks consider real-time data, improve characterization of subsurface flow and transport processes, and better link experimentation with theoretical prediction [2, 8, 11, 70]. However, balancing the economic costs of real-time observation networks with scientific advancement is currently one of the most challenging design aspects, and of foremost importance. This dissertation advances the tractability of real-time LTGM network design by effectively combining and demonstrating a suite of state-of-the-art computational tools on a real-time scaled physical aquifer transport experiment. This work presents an improved space-time LTGM design methodology that allows decision makers to (i) integrate a wide variety of model and site data into network monitoring decisions, (ii) understand optimal design tradeoffs for their sites, and (iii) reduce network costs while effectively monitoring states of interest. Specifically, this dissertation demonstrates how bias-aware Ensemble Kalman Filtering (EnKF) [71], many objective (i.e., greater than three objectives) search using hierarchical Bayesian optimization [64], and interactive high dimensional visual analytics [72] can be combined to facilitate discovery and negotiation in the LTGM design process. These tools combined greatly improve the computational tractability of LTGM network design while also advancing decision makers’ understanding of the complex relationships
that exist between their models, observations, and design objectives.

1.4.1 Chapter 2

Chapter 2 provides a conceptual background of the computational tools utilized throughout this dissertation. LTGM design is often characterized by numerous and conflicting design objectives. Multiobjective evolutionary algorithms (MOEAs) are introduced as heuristic search tools capable of searching the large combinatorial decision spaces associated with LTGM network design problems. Specifically, the Epsilon-Dominance Non-Dominated Sorted Genetic Algorithm II ($\varepsilon$-NSGAII) is introduced as a baseline MOEA for optimizing LTGM network design problems. However, the scope of spatiotemporal LTGM network design problems considered in this work extend beyond what has traditionally been explored, motivating the development of a new, more robust MOEA based on the Hierarchical Bayesian Optimization (hBOA) algorithm. Further background includes a discussion and basic formulation of quantile Kriging and ensemble Kalman filtering, both used extensively in this work. Finally, interactive visualization techniques as a means of exploring the high-dimensional spaces associated with many-objective LTGM design problems are discussed.

1.4.2 Chapter 3

Chapter 3 contributes a detailed assessment of how increasing LTGM network problem sizes (measured in terms of the number of decision variables being considered) impacts the computational complexity of using existing (MOEAs) to optimize LTGM network design applications. The $\varepsilon$-NSGAII, which has been shown to be an effective, efficient, and reliable MOEA, was chosen for the computational scaling study. Four design objectives were chosen for the analysis, all of which were minimized: (i) sampling cost, (ii) contaminant concentration estimation error, (iii) local uncertainty, and (iv) contaminant mass estimation error. The true Pareto-optimal solution set was generated for 18 through 25 well LTGM test cases in order to provide rigorous algorithm performance assessment for problems of increasing size. Results indicate that the $\varepsilon$-NSGAII exhibits quadratic computational scaling with increasing LTGM problem size. However, if the user is willing to accept an ap-
proximation to the Pareto-optimal solution set, \( \varepsilon \)-dominance can be used to reduce the computational scaling of MOEAs to be approximately linear with increasing problem size. This chapter provides a basis for advancing the size and scope of water resources problems that can be effectively solved using current MOEAs. The work presented in Chapter 3 has been published in the journal *Advances in Water Resources* [73].

### 1.4.3 Chapter 4

Chapter 4 focuses on the development of a next generation MOEA that can learn and exploit complex interdependencies and/or correlations between decision variables in LTGM design applications to provide more robust performance for large problems (defined in terms of both the number of objectives and decision variables). The proposed MOEA is termed the Epsilon-Dominance Hierarchical Bayesian Optimization Algorithm (\( \varepsilon \)-hBOA), which is representative of a new class of probabilistic model building evolutionary algorithms. The \( \varepsilon \)-hBOA was tested relative to a top performing traditional MOEA, the Epsilon-Dominance Nondominated Sorted Genetic Algorithm II (\( \varepsilon \)-NSGAII) for solving a four-objective LTM design problem. A comprehensive performance assessment of the \( \varepsilon \)-NSGAII and various configurations of the \( \varepsilon \)-hBOA were performed for both a 25 well LTM design test case (representing a relatively small problem with over 33 million possible designs), and a 58 point LTM design test case (with over \( 2.88 \times 10^{17} \) possible designs). Results indicate that the model building capability of the \( \varepsilon \)-hBOA greatly enhances its performance relative to the \( \varepsilon \)-NSGAII, especially for large monitoring design problems. This work also indicates that decision variable interdependencies appear to have a significant impact on the overall mathematical difficulty of the monitoring network design problem. The work presented in Chapter 4 has been published in the journal *Advances in Water Resources* [64].

### 1.4.4 Chapter 5

Chapter 5 presents a new framework for interactive high dimensional visual analytics termed Visually Interactive Decision-making and Design using Evolutionary Multiobjective Optimization (VIDEO). The VIDEO framework enables decision
makers to visually navigate large many-objective solution sets, ultimately allowing them to identify one or more designs that meet multiple conflicting design criteria. Specifically, the interactive visualization framework provides a set of innovative exploration tools for many-objective Pareto-optimal solution sets (i.e., solution sets resulting from three or more design objectives). The framework is demonstrated on a LTGM design application in which users can explore and visualize tradeoffs for up to four design objectives simultaneously. Interactive functionality within the framework allows the decision maker to select solutions within the design objective space and visualize the corresponding LTGM network’s performance in the design decision space. VIDEO provides the user with a more holistic view of the information provided by many-objective LTGM design, ultimately allowing them to make a more informed decision. In addition, the ease with which the VIDEO framework allows users to navigate and compare solutions and design tradeoffs leads to a time efficient analysis, even among thousands of potential solutions. The work presented in Chapter 5 has been published in the journal *Environmental Modelling and Software* [72].

### 1.4.5 Chapter 6

Chapter 6 contributes a combination of laboratory-based aquifer tracer experimentation and bias-aware Ensemble Kalman Filtering (EnKF) to demonstrate that systematic modeling errors (or bias) in source loading dynamics and the spatial distribution of hydraulic conductivity pose severe challenges for groundwater transport forecasting under uncertainty. The impacts of model bias are evaluated using an ammonium chloride tracer experiment conducted in a three dimensional laboratory tank aquifer with 105 near real-time sampling locations. This chapter contributes a bias-aware EnKF framework that (i) dramatically enhances the accuracy of concentration breakthrough forecasts in the presence of systematic, spatio-temporally correlated modeling errors, (ii) clarifies in space and time where transport gradients are maximally impacted by model bias, and (iii) expands the size and scope of flow-and-transport problems that can be considered in the future. The work presented in Chapter 6 has been published in the journal *Geophysical Research Letters* [71].
1.4.6 Chapter 7

Chapter 7 contributes the ASSIST (Adaptive Strategies for Sampling in Space and Time) framework for improving long-term groundwater monitoring decisions across space and time while accounting for the influences of systematic model errors (or predictive bias). The ASSIST framework combines contaminant flow-and-transport modeling, bias-aware ensemble Kalman filtering (EnKF) and many-objective evolutionary optimization. The EnKF allows decision makers to enhance plume transport predictions in space and time in the presence of uncertain and biased model predictions by conditioning them on uncertain measurement data. This chapter demonstrates the ASSIST framework using a laboratory based physical aquifer tracer experiment. In this initial demonstration, the position and frequency of tracer sampling was optimized to: (1) minimize monitoring cost, (2) maximize information provided to the EnKF, (3) minimize failure to detect the tracer, (4) maximize the detection of tracer flux, (5) minimize error in quantifying tracer mass, and (6) minimize error in quantifying the centroid of the tracer plume. In combination, the forecasting, search, and visualization components of the ASSIST framework represent an advance in observation networks design that has a strong potential to innovate our characterization, prediction, and management of groundwater systems. The work presented in Chapter 7 has been submitted for review to the journal *Water Resources Research* [74].

1.4.7 Chapter 8

Chapter 8 provides a summary of the major contributions of this dissertation as well as avenues of future research that have the potential to further advance LTGM network design.
CHAPTER 2

Background

This chapter provides a conceptual background for the various computational tools and concepts utilized throughout this dissertation. Long-term groundwater monitoring (LTGM) network design problems are often characterized by numerous and conflicting design objectives. Multiobjective evolutionary algorithms (MOEAs) are introduced in this chapter as heuristic search tools capable of searching the large combinatorial decision spaces associated with LTGM network design problems. Specifically, the Epsilon-Dominance Non-Dominated Sorted Genetic Algorithm II (ε-NSGAII) is introduced as a baseline MOEA for optimizing LTGM network design problems. However, the scope of spatiotemporal LTGM network design problem considered in this work extends beyond what has traditionally been explored, motivating the development of a new, more robust MOEA based on the Hierarchical Bayesian Optimization (hBOA) algorithm. Further background includes background discussion and basic formulations for quantile Kriging (used in Chapters 3, 4, and 5) and ensemble Kalman filtering (used in Chapters 6 and 7). Finally, visualization techniques for exploring the high-dimensional spaces associated with many-objective LTGM design problems are discussed as well.

2.1 Many-objective Optimization

Recent environmental systems research has emphasized the need for accounting for multiple design objectives simultaneously [75–78]. While multiobjective prob-
lem formulations have traditionally focused on two objectives, Fleming et al. [79] recently emphasized the growing body of engineering applications using “many-objective” formulations which might contain anywhere from four to twenty or more objectives. LTGM design problems are inherently suited to many-objective formulations because of the frequent need to satisfy the demands of diverse stakeholders [14]. In the context of LTGM design, regulatory bodies and the general public would be primarily interested in minimizing the risk associated with an environmental contaminant, while cost is likely an important factor to the industrial body responsible for monitoring the contaminant. As another example, researchers would most likely prefer to focus on characterizing the contaminant and physical processes that govern its passage through the system. Design objectives often conflict with one another, motivating the need to quantify and understand design tradeoffs. As noted by Reed and Minsker [63], the consideration of a suite of three of more objectives (i.e., high order Pareto optimization) for environmental systems research can aid in generating a suite of design alternatives that satisfy the negotiated needs of diverse groups of stakeholders at a reasonable cost to society. The work by Knopman and Voss [80] effectively illustrated this diversity of stakeholder preferences in monitoring network design by simultaneously meeting research (transport model identification), regulatory (transport model effectiveness), and cost minimization objectives using USGS tracer test experiment data.

Terminology associated with multi- or many-objective optimization is now discussed to aid the reader in understanding this important concept throughout the remainder of this dissertation. An optimal many-objective solution set is defined as the set of solutions which are better than all other solutions in terms of at least one design objective and is termed the Pareto-optimal set [81]. The Pareto-optimal solution set can be visualized by plotting the objective values of each solution, resulting in an $F - 1$ dimensional surface with $F$ being the number of design objectives. For example, in Figure 2.1, a hypothetical solution set is plotted for two design objectives, $Cost$ and $Error$, each of which are to be minimized. The enlarged solutions bordering the set represent the Pareto-optimal or nondominated set because each of these solutions are said to dominate all of the smaller solutions in the plot. Solution 1 labeled in the figure dominates the region enclosed within the dashed lines because any solutions found in this region represent poorer per-
performance than solution 1 in terms of the design objectives. If a box were drawn for each of the Pareto-optimal solutions, the region of the space covered by these boxes is said to be dominated.

In the presentation of the Pareto front, it is initially assumed that every solution is equally optimal prior to a decision maker’s selection of a preferred solution(s). In Figure 2.1, in order to decrease Error, the Cost of collecting data from the system has to increase. This represents a tradeoff relationship between the two design objectives where performance in one objective cannot be improved without degrading performance in another objective. This example represents a simple case where two design objectives result in a relatively small set of nondominated solutions which can easily be compared to one another. In addition, a decision maker could theoretically select a solution from this set based on some preference information which has not been modeled in the formulation of the problem. However, adding

Figure 2.1. Two objective example illustrating dominance, nondominance, and Pareto-optimality.
additional objectives (as in many-objective formulations) can have a profound impact on the size and complexity of the Pareto-optimal solution set. For example, adding an uncertainty objective to the hypothetical problem posed in Figure 2.1 has the potential to increase the Pareto-optimal set size to over 500 solutions and adding a fourth objective, contaminant mass estimation error increases the size of the Pareto-optimal set to over 2500 solutions. This explosive growth in solution set size as a result of adding design objectives motivates the need for improved tools that allow the decision maker to efficiently explore the Pareto set, ultimately enabling an improved understanding of their problem [82].

### 2.1.1 Multiobjective Evolutionary Algorithms

Innovations in multiobjective evolutionary algorithms (MOEAs) have served to catalyze the development of a broad suite of multiobjective design and decision support methodologies within the environmental and water resources literature [46, 63, 83–91]. MOEAs can potentially evolve entire tradeoffs surfaces in a single algorithm run for large, complex problem spaces [92–94]. In addition, MOEAs have the potential to effectively solve problems which are non-differentiable, discontinuous, or are otherwise characterized by unfavorable search space characteristics that can greatly limit the applicability of other multiobjective optimization methods [95]. Evolutionary algorithms in general represent a population based search heuristic that evolves solutions towards Pareto-optimal designs through a process analogous to Darwinian natural selection, whereby highly fit “parent” solutions are selected and combined with one another to produce improved “child” solutions [92]. The primary evolutionary operators of selection, crossover, and mutation play an integral role in the optimization process. Selection helps to ensure that favorable characteristics of a solution are passed from generation to generation, crossover “blends” favorable solutions in an attempt to improve their fitness, and mutation represents the addition of randomness into the population of solutions in an attempt to improve the diversity of the search [92]. Recent innovations in MOEAs have resulted in a rapidly growing field with a wide variety of applications (as reviewed by [78, 96–98]).

Schaffer [99] developed one of the first MOEAs in the mid 1980’s called the vec-
tor evaluated genetic algorithm (VEGA). After the development of VEGA, various MOEAs were developed over the next several years which emphasized simplicity in their implementations [95]. The Nondominated Sorted Genetic Algorithm (NSGA) [100] was one such algorithm that utilized a new approach of sorting its population of solutions according to nondomination ranking, but suffered from inefficiencies [95]. In a subsequent wave of development during the previous decade, an emphasis was placed on algorithm efficiency, and it was here that an improved version of the NSGA, the NSGAIi [101], emerged as an efficient benchmark MOEA [95]. One of the major improvements implemented within the NSGAIi (and most other second generation MOEAs) was the addition of elitism [102] which helps to ensure that the best solutions survive from generation to generation. In addition, the NSGAIi improved the diversity (or spread) of its search by implementing a crowding distance operator which provides preference to solutions found in portions of the search space with a low solution density [101]. The performances of the NSGA and NSGAIi have been demonstrated on various test functions developed within the operations research and computer science literature [100, 101, 103–105] and both have been shown to be quite competitive with various other second generation MOEAs such as the Strength Pareto Evolutionary Algorithm (SPEA) [102] and the SPEA2 [106] as well as the Epsilon-Dominance Multiobjective Evolutionary Algorithm (εMOEA) [105].

2.1.2 Epsilon-Nondominated Sorted Genetic Algorithm II

The original NSGAIi was shown to be quite effective in long-term groundwater monitoring (LTGM) network design by Reed et al. [63, 87, 107] making it a desirable choice for multiobjective LTGM network design in general. The NSGAIi’s initial success at solving LTGM design problems motivated further research into improving the performance and usability of the algorithm, ultimately resulting in the development of the Epsilon-dominance Nondominated Sorted Genetic Algorithm-II (ε-NSGAIi) [108–110]. In LTGM design, an MOEA evaluates potential designs in terms of a vector of design objectives. The concept of Pareto-dominance is used to assign fitness values to the designs based on their performance in terms of each design objective.
The $\varepsilon$-NSGAII evaluates potential LTGM designs as shown in Figure 2.2. Initially, a population of $p$ random designs is generated and non-domination and crowding distance \[101\] are used to assign fitness values to each design based on their performance across a suite of design objectives. SBX crossover and polynomial mutation are then performed to generate $p$ child solutions created from the selected group of highly fit parents. The algorithm then combines $p$ child solutions and $p$ parent solutions into a temporary pool of $2p$ candidate solutions. Non-domination sorting is then used to rank each of the $2p$ designs based on the number of solutions that dominate them in all objectives. In addition, crowding distance is calculated based on the average Euclidean distance between a design and the remaining designs that have been assigned the same rank. At this point, a new population of $p$ solutions is filled by gathering the highest ranked solutions. When solutions of equal rank exceed the size of the new population, two-step crowded binary tournament selection is performed to fill the available population slots while giving preference to solutions with higher crowding distance values (see “Front $i$” in Figure 2.2). Designs with higher crowding distance values add diversity to the population of designs, which helps to ensure that the $\varepsilon$-NSGAII will find solutions along the full extent of the Pareto surface. The $p$ children that have been selected now become the parents of a subsequent generation from which the process is repeated. In addition these $p$ children are eligible for entry into an off-line archive that stores the best solutions found throughout the run. To achieve entry into the archive, child solutions must be $\varepsilon$-nondominated with respect to the currently stored solutions of the archive.

**Epsilon-Dominance.** The concept of $\varepsilon$-dominance \[105, 111\] allows the user to specify the precision with which they want to quantify each design objective in a many-objective problem. Figure 2.3 illustrates the concept of $\varepsilon$-dominance. First, a user specified $\varepsilon$ grid is applied to the search space of the problem based on the decision makers precision goals for each design objective. Large $\varepsilon$ values result in a coarse grid (and ultimately fewer solutions) while small $\varepsilon$ values produce a fine grid. Grid blocks containing multiple solutions prefer the solution closest to the lower left-hand corner of the block (assuming minimization of all objectives). In the second step, nondomination sorting based on the grid blocks is performed. For example, the solution in the box labeled “1” dominates the shaded region of grid
blocks above and to the right. As a result, the solution above it is dominated in terms of the required precision, and hence eliminated. A similar procedure can be followed for solutions “2” and “3”, resulting in a thinning of the solutions (step 3) and promoting a more even search of the objective space. It is important to note that the addition of \( \varepsilon \)-dominance archiving does not add additional parameters to the algorithm. Rather, it allows the user to define the precision requirements that make sense for their particular application. In addition, \( \varepsilon \)-dominance provides the following key benefits: (i) it enhances the diversity of search by ensuring an even spread of nondominated solutions, (ii) the size of the Pareto optimal set can be bounded to a reasonable precision for large applications, (iii) it provides a means of controlling the computational demands of an application [73], and (iv)
the population size of the algorithm is informed by the \( \varepsilon \)-nondominated set size which has been shown to be an important factor in effectively evolving the Pareto-optimal set [87, 112]. The interested reader can refer to prior work by Laumanns et al. [111] and Deb et al. [105] for a more detailed description of \( \varepsilon \)-dominance.

**Dynamic Population Sizing.** The \( \varepsilon \)-NSGAII uses a series of “connected runs” where small populations are initially exploited to pre-condition search and are then automatically adapted commensurate with problem difficulty/Pareto-optimal set size [see Figure (2.2)]. Throughout the optimization process, the population size is automatically adapted [113] based on the number of \( \varepsilon \)-nondominated solutions that the algorithm has found. Epsilon-nondominated solutions found after each generation are stored in an archive, a common method of guaranteeing elitism [95]. The archived solutions are subsequently used to direct the search using 25-percent injection. This means that after each run when the population is reinitialized, 25-percent of the new population will be composed of the \( \varepsilon \)-nondominated archive solutions and the other 75-percent will be generated randomly. This assists the search in two ways: (i) by directing the search using previously evolved solutions and (ii) by adding new, random solutions to facilitate exploration. In addition, because \( \varepsilon \)-dominance is used to control the precision of the solution set, this injection scheme bounds the population size based on the number of solutions that exist according to the user specified \( \varepsilon \) resolution. Theoretically, this approach allows the MOEA’s population size to increase or decrease, and in the limit when the \( \varepsilon \)-nondominated archive size stabilizes, the \( \varepsilon \)-NSGAII’s “connected runs” are equivalent to a diversity-based EA search enhancement recommended by Goldberg [114] termed “time continuation”. The search can be terminated across all
runs (i.e., across all populations used) if the number and quality of solutions has not increased above Δ-percent across two successive runs. However, various other termination criteria may be more applicable depending on the application, such as specifying a maximum number of function evaluations or a target performance criteria.

The primary goal in the development of the ε-NSGAII was to provide a highly reliable and efficient MOEA that minimizes the need for traditional EA parametrization and allows the user to focus on problem specific search quality goals. Computational savings can be viewed in three contexts: (i) the use of minimal population sizes, (ii) the elimination of trial-and-error application runs to determine search parameters, and (iii) the elimination of random seed analysis. Although the adaptation of population size will differ depending on the random seed chosen, exploiting small populations to precondition search will on average greatly reduce computation times. Moreover, this approach minimizes unnecessary runtime by terminating search based on user defined precision goals.

The performance of the ε-NSGAII has been demonstrated [108] on a suite of two-objective test functions [103, 115] popular within computer science and optimization literature and has exhibited superior performance relative to the NSGAII [101] and the εMOEA [105]. More recently, the ε-NSGAII was demonstrated on a real-world LTGM network design application [109] and was shown to be competitive to superior to several other benchmark second generation MOEAs including the NSGAIi, the εMOEA , and the SPEA2 [106]. Two parallel implementations of the ε-NSGAII have also been developed and shown to perform and scale well on several test functions and two real-world water resources applications [91]. Reduced parametrization requirements and intuitive objective precision specification make the algorithm a desirable choice for real-world water resources applications such as LTGM network design.

### 2.1.3 Understanding LTGM Design

In LTGM design problems, decision variables representing sampling points are generally specified using a binary notation where “1” would indicate that a sample is taken and “0” indicates no sampling is performed at a location in space and/or
time. Using this binary notation, the decision space (or total number of possible designs) of the network design problem grows according to $2^{MT}$, where $M$ is the number of possible sampling locations, and $T$ is the number of sampling periods. For example, designing a LTGM network with 25 available sample locations for a single point in time results in a network design problem containing $2^{(25)(1)}$ (or over 33-million) possible designs. Adding just one additional sampling time (i.e., $T = 2$) increases the search space to over $1.12 \times 10^{15}$ possible designs. The ε-NSGAII algorithm was shown to quantify 80-percent of the Pareto-optimal set for a 25 well LTGM design test case sampling for one time period at a 98-percent reduction in computational cost over full enumeration (i.e., 600,000 evaluations as opposed to 33-million evaluations) [73]. However, for large LTGM network design problems (e.g., many sampling points in space and time), improvements will be required to adequately solve these problems.

Chapter 3 of this dissertation motivates the need for improved computational scaling where it is shown that the ε-NSGAII requires approximately quadratic computational scaling with increasing LTGM design problem size. Although Chapter 3 shows that the computational scaling of the algorithm can be improved through careful specification of the ε-dominance archive precision (i.e., accepting a coarser approximation of the Pareto-optimal set), algorithms that show potential for solving difficult problems while maintaining sub-quadratic computational scaling should be explored in order to optimize large LTGM design problems to a higher degree of accuracy. For network design applications where there are very likely strong correlations between decision variables, a search algorithm that can “learn” these correlations has the potential to greatly improve the computational scaling characteristics of these design problems. This motivates the exploration and refinement of MOEAs that are capable of learning the complex correlations or linkages between LTGM decision variables such as the Hierarchical Bayesian Optimization Algorithm (hBOA) [116] described in the next section.

2.1.4 Hierarchical Bayesian Optimization Algorithm

The Hierarchical Bayesian Optimization Algorithm (hBOA) [116] attempts to overcome the difficulties posed by large sets of correlated (or linked) decision variables
by replacing the traditional MOEA operators of crossover and mutation, with Bayesian network models of how decision variables impact solution quality. The hBOA algorithm “learns” the structure of the problem through construction of a Bayesian network based on the most promising solutions found (parents). New solutions (children) are then simulated based on the constructed Bayesian network to produce improvements in the solution population. In constructing the Bayesian network, each decision variable of the problem is represented as a node, and edges are used to establish relationships between the nodes. Edges represent the conditional probabilities that exist between each of the decision variables. For example, if two variables (nodes) are completely independent of one another, no edge would exist connecting the two. Learning the structure of the problem (i.e., the conditional probabilities) can be difficult and is accomplished using a greedy search algorithm [117] that performs various elementary network operations such as adding, removing, or reversing the direction of the edges connecting various nodes [116]. A performance metric is then used to evaluate how each of these elementary operations improves the network. If no improvements are made, the construction of the network is assumed adequate and the search is terminated.

LTGM design problems have the potential for conditional dependencies to exist between sampling decisions due to the underlying physics of flow-and-transport, and the influences of domain geometry and performance criteria. While the dependency structure of network design problems has the potential to be quite complex, the identification of at least some of the dependencies that exist may greatly aid in improving the search efficiency of MOEAs. The performance of the hBOA has been demonstrated on both test functions [116, 118–120] and various real-world applications such as the notably difficult Ising spin-glass systems [121, 122]. This dissertation (see Chapter 4), develops a multiobjective extension of hBOA that builds on the strengths of the $\varepsilon$-NSGAII framework. This served to expand the size, complexity, and scope of network design problems that could be considered.

Since the $\varepsilon$-NSGAII’s performance has been validated extensively on both test functions and real-world applications, the incorporation of the hBOA’s Bayesian network model building within the $\varepsilon$-NSGAII resulted in an extremely effective MOEA that utilizes the strengths of both algorithms. The original hBOA algorithm incorporates model building and simulation within the traditional framework
of a single objective evolutionary algorithm. Chapter 4 explores incorporating the Bayesian network model building and simulation components of the hBOA within the multiobjective framework of the \( \varepsilon \)-NSGAII. This new algorithm is termed the Epsilon-Dominance Hierarchical Bayesian Optimization Algorithm (\( \varepsilon \)-hBOA) and it includes all the model building strengths of the hBOA along with the benefits of the \( \varepsilon \)-NSGAII's \( \varepsilon \)-dominance archiving and dynamic population sizing within a multiobjective framework.

2.2 Interpolation and Filtering

Data assimilation methods such as optimal interpolation and statistical filtering aid in maximizing data quality by reducing data uncertainty and merging information from multiple sources and scales. Since environmental data is often collected at a sparse resolution relative to the scale at which variables of interest vary, data assimilation techniques can aid in providing a more complete and accurate picture of the data [123]. More specifically, spatial interpolation methods attempt to reconstruct a more complete image of the data based on scattered measurements at a single point in time. Examples of interpolation methods include inverse distance weighting [124] and various least-squares linear regression techniques (commonly referred to as Kriging) [125]. In contrast, statistical filtering attempts to assimilate data measurements into model forecasts while accounting for both uncertainty in the model and the measurements [123]. In other words, actual field measurements are used to update the model, often in real-time. Common filtering techniques include the various Kalman Filtering methods that originated from the work of R.E. Kalman in the 1960’s [126]. In this work, Quantile Kriging is utilized in Chapters 3 and 4 as an effective spatial interpolator based on the work of Reed et al. [127], and Ensemble Kalman Filtering [128] is used in Chapters 6 and 7 to assimilate uncertain data measurements with biased model predictions in both space and time. Both methods are now summarized in more detail.
2.2.1 Quantile Kriging

Quantile Kriging (QK) is used in this work to provide spatial interpolations of hydrologic data provided by LTGM networks. The selection of QK is based on the recommendations of Reed et al. [127] in which they found that QK showed the least bias with respect to variability of data and preferential sampling, and was most robust in representing data when compared to five other interpolation methods. Kriging provides a minimum error variance estimate value at an unsampled location provided the data at the sampled locations [129]. QK extends Ordinary Kriging (OK) by transforming the data values to quantile space according to their rank using Equation 2.1

\[ d^*(u_s) = \frac{\text{rank of } d(u_s)}{M + 1} \]  

where \( d^*(u_s) \) is the quantile transformed sample value at well \( u \) and sampling location \( s \), \( d(u_s) \) is the original data value at well \( u \) and location \( s \), and \( M \) is the total number of sampling points. The quantile values represent the empirical cumulative distribution function (CDF) of the data values, which always results in normalized data. Samples are Kriged in quantile space and then transformed back to concentration space using a transform table generated from the empirical CDF [130, 131]. The OK estimator is then a linear combination of the \( d^*(u_s) \) quantile data at the \( M \) sampled locations as shown in Equation 2.2.

\[ QK^*(u_g) = \sum_{s=1}^{M} \lambda(u_s)d^*(u_s) \text{ where } \sum_{s=1}^{M} \lambda(u_s) = 1 \]  

In Equation 2.2, the \( QK^*(u_g) \) represents the minimum error variance quantile estimate at location \( u \) within the Kriging interpolation grid \( g \), and the \( \lambda(u_s) \) represents the Kriging weights that assure that the estimate is unbiased. Structural analysis of the underlying data is typically performed in order to generate a semivariogram describing the spatial correlation of the data. Modeling the semivariogram allows Equation 2.2 to be rewritten as:
\[
\gamma^*(d^*(u_{s_1}) - d^*) = \sum_{s_2=1}^{M} \lambda(u_{s_1}) \gamma^*(d^*(u_{s_1}) - d^*(u_{s_2}))
\]

for \(s_1 = 1, \ldots, M\) and \(\sum_{s=1}^{M} \lambda(u_s) = 1\) \hspace{1cm} (2.3)

where \(\gamma^*\) is the semivariogram representation of the spatial correlations between quantile transformed data values. This results in a set of \(M\) equations that can be used to solve for the Kriging weights \(\lambda(u_s)\). OK assumes stationarity of the mean. However, for environmental data such as contaminant concentrations, the local mean typically varies greatly over the domain of interest [125]. OK accounts for this fluctuation in the local mean by using moving local search neighborhoods to estimate the expected value at each unsampled location in the interpolation grid, \(u_g\) [125, 129]. This capability also greatly improves the computational speed of OK by limiting the number of equations needed to solve for the Kriging weights in Equation 2.3 to the number of data values which will locally influence the estimation location. In Chapters 3, 4, and 5, a C translation of KT3D, a 3-dimensional Kriging library written in Fortran as part of the GSLIB software package [129] was used to perform spatial interpolation based on available sample data.

### 2.2.2 Ensemble Kalman Filtering

Ensemble Kalman Filtering (EnKF) [128] is utilized in Chapters 6 and 7 to enhance plume transport predictions in space and time in the presence of uncertain and biased model predictions by conditioning them on uncertain measurement data. Kalman Filtering (KF) encompasses a general class of time-controlled state estimators that account for both process noise and measurement uncertainty (which are assumed independent of one another [25]) simultaneously [132]. Prior knowledge of the process and measurement devices allows prediction of this uncertainty, ultimately producing estimates with a statistically minimized error [133]. The KF provides three key advantages in network design as highlighted by [14]: (i) the performance of the network can be evaluated \textit{a priori}, (ii) the KF can handle high dimensional systems, and (iii) not all variables which are modelled need to be sampled.
The general KF proceeds iteratively in two basic steps: a forecast (or prediction), and assimilation of measurements [132]. The forecast step projects the state and error covariance of the system ahead in time while the measurement update step corrects the estimate and its associated error covariance by assimilating uncertain measurement data. The basic KF is now derived using the notation of Evensen [128]. Given a non-linear dynamical model, the forecasted state of a system is given by Equation 2.4

\[ \psi_{j+1} = \mathcal{M}(\psi_j) \]  

(2.4)

where \( \psi \) is the system state, \( j \) the current time step, and \( \mathcal{M} \), a non-linear model operator. Likewise, the error covariance is given by Equation 2.5

\[ P_{j+1} = \mathcal{M}P_j\mathcal{M}^T + Q_j \]  

(2.5)

where \( P \) is the error covariance at time \( j \), and \( Q \) is the covariance matrix representing the model errors. In general, a model state forecast is denoted as \( \psi^f \) and a model state update (i.e., assimilation of measurement data) is denoted as \( \psi^a \). The measurements \( d \) of the true system states \( \Psi \) at time step \( j \) are assumed to have error \( \nu \) as shown in Equation 2.6

\[ d_j = H\Psi_j + \nu_j \]  

(2.6)

where \( H \) is a measurement operator that relates the data measurements \( d \) at time step \( j \) to the system state space. Whenever data is available, a model state update is performed using what is commonly referred to as the analysis equation (Equation 2.7).

\[ \psi^a_j = \psi^f_j + P^f_jH^T(HP^f_jH^T + R_j)^{-1}(d_j - H\psi^f_j) = \psi^f_j + K_j(d_j - H\psi^f_j) \]  

(2.7)

The Kalman Gain notation is often used in the analysis update equation where the Kalman Gain, \( K \), is equivalent to Equation 2.8.

\[ K_j = P^f_jH^T(HP^f_jH^T + R_j)^{-1} \]  

(2.8)
In both Equations 2.7 and 2.8, \( R \) represents the covariance of the measurement errors \( \nu \) at time step \( j \). The Kalman Gain matrix can be thought of as a blending factor which minimizes the *a posteriori* error covariance [132]. Following the analysis update, the analysis covariance can be obtained using Equation 2.9

\[
P^a_j = \hat{P}^f_j - \hat{P}^f_jH^T(HP^f_jH^T + \hat{R})^{-1}HP^f_j = \hat{P}^f_j - K_j\hat{H}\hat{P}^f_j
\]  

(2.9)

Equations 2.4 through 2.9 represent the basic KF formulation. The EnKF extends the KF using a Markov Chain Monte Carlo solution technique that models the state field’s mean and covariance by conditioning Monte Carlo simulation based forecasts on noisy observation data [128]. In order to extend the KF representation to an ensemble formulation, we define an ensemble of model states

\[
A = (\psi_1, \psi_2, \ldots, \psi_N) \in \mathbb{R}^{n \times N}
\]  

(2.10)

which is of dimension \( n \times N \) where \( n \) is the number of model states and \( N \), the number of ensemble members. The ensemble mean is stored in the columns of \( \overline{A} \) where \( \mathbf{1}_N \) is an \( N \times N \) matrix with elements \( 1/N \).

\[
\overline{A} = A\mathbf{1}_N
\]  

(2.11)

The perturbations of the model states can then be found by subtracting the ensemble mean from \( A \) as shown in Equation 2.12.

\[
A' = A - \overline{A}
\]  

(2.12)

The ensemble covariance of model states is now defined as

\[
P = \frac{A'(A')^T}{N-1}
\]  

(2.13)

In addition, the measurements, \( d \) can be perturbed with an ensemble of errors \( v \) where the ensemble of perturbations is defined by Equation 2.14

\[
\Upsilon = (v_1, v_2, \ldots, v_N) \in \mathbb{R}^{m \times N}
\]  

(2.14)
and the perturbed measurements can be stored in the matrix $D$ shown in Equation 2.15.

$$D = (d_1, d_2, \ldots, d_N) \in \mathbb{R}^{m \times N} \quad (2.15)$$

$D$ and $\Upsilon$ are now $m \times N$ matrices containing the ensemble of perturbed measurements and the perturbations on those measurements respectively. Likewise, the measurement error covariance matrix, $R$ can now be defined as:

$$R_e = \frac{\Upsilon \Upsilon^T}{N - 1} \quad (2.16)$$

The analyzed ensemble is now given by Equation 2.17

$$A^a_j = A^f_j + P^f_j H^T (H P^f_j H^T + R_j)^{-1} (D_j - H A^f_j) = A^f_j + K_j (D_j - H A^f_j) \quad (2.17)$$

and the analyzed ensemble covariance by Equation 2.18.

$$P^a_j = P^f_j - P^f_j H^T (H P^f_j H^T + R_j)^{-1} H P^f_j = P^f_j - K_j H P^f_j \quad (2.18)$$

EnKF provides a variety of key benefits: (i) it can be easily implemented for high dimensional systems [128], (ii) it provides mean and variance estimates for nearly any distribution [128], (iii) it is effective in modeling highly non-linear systems [133], (iv) it eliminates the need to compute derivative based linearizations of non-linear systems [133], and (v) it captures the anisotropic properties of the modeled system [134]. One of the first applications of the KF to hydrologic network design (specifically, water quality networks) was by Moore in the early 1970’s [135]. Since then, KF has been used extensively throughout the hydrologic sciences (for excellent reviews of the work done throughout the 1990’s, see Drécourt [136] and Eigbe et al [137]).

### 2.3 Visualization to Inform Decision Making

Classical multiobjective decision making methodologies [75, 138] have generally focused on transformation methods that commensurate multiple objectives into new formulations that are more easily solved using single objective optimization algo-
MOEAs represent a departure from traditional multi-criteria operations research where it is no longer necessary to transform problem structures into single objective formulations. Instead, new decision support tools that facilitate visualization and exploration of high-dimensional data can allow the decision maker to focus on the structure and content of Pareto surfaces. More interestingly, providing powerful visualization and interaction capabilities can even allow the decision maker to “steer” computationally intensive applications based on their preference information in a pseudo real-time fashion [139–145].

The work in this dissertation contributes a visualization framework tailored specifically to many-objective LTGM applications. Ultimately the new framework seeks to demonstrate that effective visualization can facilitate discovery and negotiation in the design and decision-making process for high-order multiobjective solution sets [63]. The use of the terms discovery and negotiation is motivated by the potential for high-order multiobjective solution sets to generate alternatives that (i) capture a broad suite of system behaviors relevant to diverse stakeholder preferences and (ii) are capable of providing insight into both modeled and unmodeled objectives [82]. This ultimately aids decision makers in discovering system dependencies and/or tradeoffs and allows them to exploit this information in the negotiated selection of a design solution.

Traditionally, decision making methodologies have been divided into a priori, interactive, and a posteriori approaches [78, 145]. A priori methods seek to model decision maker preference before searching for designs/decisions [145]. However, a priori approaches have been criticized because (i) they do not condition the decision maker’s preference on potential alternatives, (ii) they suffer from decision maker contradictions, and (iii) utility representations of preference are non-unique for groups [78]. Interactive approaches represents the “progressive articulation of preferences” [78] by the decision maker during the search for designs. These types of approaches can be used to “steer” or guide applications as new preference information becomes available during the search. Interactive approaches suffer from a variety of criticisms, the most notable of which is that decision maker input can often times be so contradictory that any information provided to the search is often equivalently achieved using simple trial and error analysis [78]. A posteriori decision tools allow the decision maker to account for preferences after Pareto op-
timal alternatives have already been identified. In this approach, all solutions are initially assumed to have equal preference during the search process, with the ultimate goal of generating as many solutions as possible. Decision maker preferences are then expressed in the exploration of the generated tradeoffs and selection of solutions. A posteriori decision tools have been criticized in the operations research literature due to the mathematical complexity of finding tradeoff solutions and the contention that large solution sets tend to overwhelm and confuse the decision maker while providing limited insights into their design preferences [77, 78, 146].

MOEAs have significantly enhanced our ability to search for and quantify large multiobjective solution sets in the environmental area [89, 91, 109]. In addition, multiobjective engineering and environmental applications represent complex systems that can be meaningfully visualized in space and/or time. To address the criticisms of interactive and a posteriori decision making approaches, the implication that large solution sets overwhelm decision makers assumes that decision maker expertise combined with powerful visualization tools has limited value. In addition, the provision of meaningful visualization tools which provide the decision maker with new problem insight and understanding, ultimately aids in reducing or eliminating contradictions. Exploring tradeoffs and visualizing system performance can be useful in terms of both the objective space (which represents the design objectives of the problem) and the decision space (which represents the actual physical design). There are an increasing number of studies demonstrating that visualization combined with optimization can promote design innovations and provide decision makers with an improved understanding of their system [63, 79, 142, 143, 147–149]. In the context of this prior work, Chapter 5 contributes the Visually Interactive Decision-making and Design using Evolutionary Multiobjective Optimization (VIDEO) framework, a highly interactive environment for exploring tradeoffs within many-objective design problems.
3.1 Abstract

This chapter contributes a detailed assessment of how increasing problem sizes (measured in terms of the number of decision variables being considered) impacts the computational complexity of using multiple objective evolutionary algorithms (MOEAs) to solve long-term groundwater monitoring (LTGM) applications. The Epsilon-Dominance Non-Dominated Sorted Genetic Algorithm II (\(\varepsilon\)-NSGAII), which has been shown to be an efficient and reliable MOEA, was chosen for the computational scaling analysis. Four design objectives were chosen: (i) sampling cost, (ii) contaminant concentration estimation error, (iii) local uncertainty, and (iv) contaminant mass estimation error. The true Pareto-optimal solution set was generated for 18 through 25 well LTGM test cases in order to provide for rigorous algorithm performance assessment for problems of increasing size. Results indicate that the \(\varepsilon\)-NSGAII exhibits quadratic computational scaling with increasing LTGM problem size. However, if the user is willing to accept an approximation to the Pareto-optimal solution set, \(\varepsilon\)-dominance can be used to reduce the computational scaling of MOEAs to be linear with increasing problem sizes. This chapter provides a basis for advancing the size and scope of water resources problems that
can be effectively solved using MOEAs. The work presented in this chapter has been published in the journal *Advances in Water Resources* [73].

### 3.2 Introduction

This chapter contributes a detailed assessment of how increasing problem sizes impacts the computational complexity of using multiple objective evolutionary algorithms (MOEAs) in long-term groundwater monitoring (LTGM) applications. Problem size in this context is measured in terms of the number of design decision variables being considered. Computational complexity (or scaling) can be defined as a measure of how problem sizes impact the growth rate of the average number of design evaluations required by an MOEA to approximate a solution to an application. Building on some recent comparative analyses of MOEA effectiveness [90, 108, 109], this chapter characterizes the computational complexity of the Epsilon-Dominance Non-Dominated Sorted Genetic Algorithm II (ε-NSGAII) [108, 109]. This algorithm has been shown to be more efficient and reliable relative to other state-of-the-art MOEAs [108, 109] in the LTGM application area. The computational scaling analysis is based on a suite of LTGM test cases formulated to test a range of problem sizes. Formally, LTGM design can be defined as the assessment of groundwater quality over long time-scales to provide “sufficient and appropriate information” to assess and possibly modify mitigation or contaminant control measures to ensure that they are adequately protective of human and ecological health [7].

In general, groundwater monitoring design has been shown to be a challenging optimization problem with multiple conflicting objectives and very large discrete decision spaces [7, 38, 39, 45–47, 63, 87, 150, 151]. Lettenmaier [14] referred to the scaling challenges posed by the LTGM network design problem as being a “curse of dimensionality”. Knopman and Voss [80, 150] recognized that the groundwater quality network design problem has many mathematical similarities to the classical combinatorial knapsack problem (i.e., discrete decision spaces that grow exponentially with increasing problem size). Reed and Minsker [63] used the LTGM problem to illustrate that MOEAs are capable of solving a new problem class [79, 152] that they termed high-order Pareto optimization (i.e., problems with three or more
design objectives). MOEAs’ can solve highly nonlinear, discrete, and non-convex problems without differentiation [92–94] and their population-based search enables them to evolve entire tradeoff (or Pareto) surfaces within a single optimization run for problems with huge decision spaces.

There has been a modern confluence of systems analysis research towards approaches that emphasize multiple objectives (see reviews [75–78]). This trend is clearly evident in the water resources literature over the past decade [46, 63, 83–88, 90]. Recent MOEA applications demonstrate that a growing body of researchers in both the water resources and broader systems analysis communities are seeking to use MOEAs in high-order Pareto optimization [63, 78, 79, 96, 105, 152, 153]. Moreover, many recent multiobjective optimization applications within the water resources literature are seeking to solve applications with large numbers of continuous, integer, and binary decisions [154–158]. For example, recent MOEA applications in hydrologic model calibration [159], non-point source pollution management [154], groundwater management [156], and distribution systems [155] consider complex integer, continuous, or mixed decisions. It should be noted that a key strength of MOEAs is their ability to rapidly approximate the true Pareto surface even if it is not exactly quantified, which can be sufficient given computational constraints.

The LTGM problem provides an excellent means of assessing MOEA computational scaling due to the problem’s large array of potential design objectives and the discrete 0/1 decision variable formulation. The 0/1 decision variable formulation represents yes/no decisions on whether to sample from predetermined monitoring well locations. The 0/1 programming formulation allows for enumerative analyses of modestly sized LTGM applications. Enumerations were developed for various problem sizes with an upper bound representative of what could be practically enumerated given current computational constraints. In the broader context of multiple objective water resources applications, the 0/1 decision variable formulation provides a lower bound estimate of the computational complexities for using MOEAs in water resources applications with more complex decisions (e.g., mixed integer formulations). The purpose of this chapter is to provide guidance on the current computational complexity of MOEAs to clarify future research paths that will allow them to solve larger water resources applications efficiently and reliably.
The MOEA computational scaling analysis presented in this chapter proceeds as follows. Section 3.3 presents the LTGM test cases and design objectives. The $\varepsilon$-NSGAII is then introduced in Section 3.4. The methodology used to develop the various sized LTGM test cases used to demonstrate the computational scaling characteristics of the $\varepsilon$-NSGAII is presented in Section 3.5. Section 3.6 provides a detailed description of the computational experiment as well as the parametrization of the $\varepsilon$-NSGAII. Section 3.7 provides detailed illustrations of the scaling characteristics of the LTGM problem and the ability of the $\varepsilon$-NSGAII to approximate the Pareto-optimal solution set. The $\varepsilon$-dominance concept is also demonstrated in this section as a means of approximating the Pareto-optimal solution set, ultimately reducing computational requirements. Section 3.8 provides a discussion regarding the algorithm’s computational scaling and the implications of this work for future water resources research. Conclusions are presented in Section 3.9.

### 3.3 Methodology

#### 3.3.1 Test Case Development

The LTGM test case is based on a 50-million node flow and transport simulation originally developed by Maxwell et al. [160]. This test case represents the migration of a hypothetical perchloroethylene (PCE) plume originating from an underground storage tank. The hydrogeology of the site has been extensively characterized and is based on a highly heterogeneous alluvial aquifer located at the Lawrence Livermore National Laboratory in Livermore, California [161–164]. Concentration data are provided at 58 hypothetical sampling locations in a 29 well monitoring network for a snapshot in time 8 years following the initial release of contaminant. Each well has one to three predetermined sampling locations available along its vertical axis and the sampling domain extends 650 m in the $X$ direction, 168 m in the $Y$ direction, and 38.4 m in the $Z$ direction with a minimum horizontal spacing of 10 m between wells. Additional details on this test case can be found in Reed et al. [127].
3.3.2 Objective Formulation

Four design objectives were chosen, each of which were minimized: (i) sampling cost, (ii) relative error of local contaminant concentration estimates, (iii) local contaminant concentration estimation uncertainty, and (iv) contaminant mass estimation error. Objectives (ii)-(iv) were obtained using the Quantile Kriging (QK) method described in Chapter 2, Section 2.2.1. Equation 3.1 represents the general objective formulation where $F(x_\omega)$ is a vector valued performance function in which the four objectives: cost ($f_{\text{Cost}}$), concentration estimation error ($f_{\text{Error}}$), local uncertainty ($f_{\text{Uncert}}$), and mass estimation error ($f_{\text{Mass}}$) are minimized.

Minimize:

$$F(x_\omega) = \left( f_{\text{Cost}}(x_\omega), f_{\text{Error}}(x_\omega), f_{\text{Uncert}}(x_\omega), f_{\text{Mass}}(x_\omega) \right), \forall \omega \in \Omega \quad (3.1)$$

Subject to $U(x_\omega) = 0 \quad (3.2)$

The objectives are all a function of the vector $x_\omega$ representing the $\omega$th sampling plan in the decision space $\Omega$. Each sampling decision, $s$, of a sampling plan, $\omega$, is determined from Equation 3.3 resulting in a string of binary digits indicating whether or not a well is sampled. Equation 3.2 subjects $F(x_\omega)$ to the constraint that no points in the interpolation domain remain unestimated. QK (described in Chapter 2, Section 2.2.1) was used to interpolate contamination estimates at unsampled locations throughout the plume. Well sampling schemes that contain too few wells or wells that are poorly distributed in space may not have a sufficient number of data points in the Kriging neighborhoods to perform interpolation and hence result in a number of unestimated points, $U(x_\omega)$, in the interpolation domain (violating the constraint described by Equation 3.2).

$$x_{\omega,s} = \begin{cases} 
1, & \text{if the } s\text{th well is sampled} \\
0, & \text{otherwise} 
\end{cases}, \forall \omega, s \quad (3.3)$$

Cost. The sampling cost objective quantifies the monitoring cost of a particular sampling scheme using Equation 3.4. The coefficient, $C$, defines the cost to sample a location $u_s$ (normalized to one). Additionally, if a well is sampled, it is assumed that all locations along its vertical axis are sampled resulting in a cost coefficient
ranging from 1 to 3. The cost objective is ultimately quantified by summing the cost coefficients of each of the $M$ wells sampled in a particular scheme.

$$f_{\text{Cost}}(x_\omega) = \sum_{s=1}^{M} C(u_s)x_{\omega,s} \quad (3.4)$$

**Error.** The relative error of local contaminant concentration estimates objective measures how the Kriged estimate of the plume using the $\omega$th sampling plan differs from that obtained by sampling from all well locations. Equation 3.5 quantifies the concentration error objective by summing the squared differences between the concentration estimate at a grid location $u_g$ using all wells, $c_{\text{alt}}(u_g)$, and the concentration estimate at the same grid location using the $\omega$th sampling plan, $c_{\omega}(u_g)$.

$$f_{\text{Error}}(x_\omega) = \sum_{g=1}^{G} \left( c_{\text{alt}}(u_g) - c_{\omega}(u_g) \right)^2 \quad (3.5)$$

**Uncert.** Local contaminant concentration estimation uncertainty is quantified by summing the estimation standard deviations obtained from Kriging at each grid location $u_g$ using Equation 3.6. The standard error weight coefficient, in this case $2\sqrt{3}$ based on the standard deviation of a uniform distribution, can be used to assign importance to uncertainty estimates at different locations in the interpolation domain. The coefficient was assumed constant across the interpolation domain.

$$f_{\text{Uncert}}(x_\omega) = \sum_{g=1}^{G} 2\sqrt{3}[\sigma(u_g)] \quad (3.6)$$

**Mass.** The contaminant mass estimation error objective quantifies the relative error between the total mass of dissolved contaminant (the zeroth spatial moment, $\mu_0$), estimated using all well locations ($\mu_{0,\text{alt}}$) and the contaminant mass estimated from the $\omega$th sampling plan ($\mu_{0,\omega}$). Equation 3.7 expresses the relative mass estimation error in terms of a percentage.

$$f_{\text{Mass}}(x_\omega) = \left| \frac{\mu_{0,\text{alt}} - \mu_{0,\omega}}{\mu_{0,\text{alt}}} \right| \cdot 100\% \quad (3.7)$$
If a well sampling scheme results in unestimated points in the interpolation domain (violating the constraint described by Equation 3.2), the objectives are penalized to ensure that infeasible sampling schemes are eliminated from consideration. Equation 3.8 is applied to each objective function if a feasibility violation occurs, resulting in solutions with lower fitness (i.e., higher objective values in a minimization problem).

\[
F_{\text{penalty}}(x_\omega) = \begin{cases} 
  f_{\text{penalty, Cost}} = f_{\text{Cost}} + f_{\text{Cost}}^\text{max} \\
  f_{\text{penalty, Error}} = f_{\text{Error}} + G + U(x_\omega) + f_{\text{Cost}}^\text{max} \\
  f_{\text{penalty, Uncert}} = f_{\text{Uncert}} + G + U(x_\omega) + f_{\text{Cost}}^\text{max} \\
  f_{\text{penalty, Mass}} = f_{\text{Mass}} + G + U(x_\omega) + f_{\text{Cost}}^\text{max}
\end{cases}
\]

Equation 3.8 penalizes the objective functions based on the maximum cost of a sampling scheme, \( f_{\text{Cost}}^\text{max} \) (which is dependent on the test case size), the total number of estimation points in the grid, \( G \) (in this case 1666 - chosen based on computational feasibility), and the total number of unestimated points, \( U(x_\omega) \), in the infeasible sampling plan. For example, if a particular sampling plan for the 25 well test case resulted in 10 unestimated points in the interpolation grid, the fitness penalty added to the design’s cost objective would be 47, and the fitness penalty added to the values for the Error, Uncert, and Mass objectives would be 1723. Since the maximum cost of the system was known based on the test case data, Equation 3.8 is defined so that all infeasible solutions will have costs that exceed the maximum feasible cost (i.e., for this test case, 47). The exact ranges of the other objectives were not known \textit{a priori}, so 1723 is a conservative penalty for the Error, Uncert, and Mass objectives that ensures that when penalized, their fitness values will exceed their maximum feasible values. Penalizing solutions rather than eliminating them ensures that sampling schemes which are almost feasible are given the opportunity to be further evolved by the MOEA into feasible designs (for more details on this problem formulation, see Reed and Minsker [63]).

### 3.3.3 Spatial Interpolation

Spatial interpolation of the contamination plume was conducted using Quantile Kriging (QK) based on the recommendations of Reed et al. [127]. Reed et al.
found that QK showed the least bias with respect to variability of PCE concentrations and preferential sampling, and was most robust in representing the plume when compared to five other interpolation methods. For a full description of QK, please refer to Chapter 2, Section 2.2.1.

Before the Kriging is performed, a variogram analysis is conducted to reveal the spatial correlation structure of the data, and a model is chosen which will best represent this structure when computing estimates [125]. For a base, 29 well test case, a geostatistical analysis revealed a spherical variogram structure with 0.005 nugget and 18 m range. The interpolation grid was defined by 34 blocks in the $X$, 7 blocks in the $Y$, and 7 blocks in the $Z$ directions, resulting in 1666 regularly spaced estimation points. QK assumes a locally stationary concentration mean within local estimation neighborhoods at each grid location. The search neighborhood size was based on an ellipsoid structure with axes lengths equal to half of each the $X$, $Y$, and $Z$ extents of the study region. The search neighborhood was divided into octants to help ensure that the closest data points were well distributed about the estimation point (for more details see [129]), and a maximum of one data point from each octant was used in the estimation, ensuring that clustered data points did not bias interpolation estimates. Smaller test cases which were generated to demonstrate the computational scaling characteristics of the LTGM problem utilized the same geostatistical parametrization of the 29 well test case because of its higher information content. Reed et al. [127] provides a more detailed description of QK for interested readers.

### 3.4 Optimization Algorithm

The $\varepsilon$-NSGAII was chosen based on its superior performance relative to the original Non-Dominated Sorted Genetic Algorithm II (NSGAII) and the Epsilon-Dominance Multi-Objective Evolutionary Algorithm ($\varepsilon$MOEA), and its competitive to superior performance relative to the Strength Pareto Evolutionary Algorithm 2 (SPEA2) on the LTGM problem formulated in Section 3.3 [109]. All of these MOEAs use real parameter simulated binary crossover (SBX) [165], polynomial mutation [96], and elitism [96]. Since all of the algorithms use the same primary search operators, it is expected that their performance will scale simi-
larly to the $\varepsilon$-NSGAII on this application. For a more detailed description of the $\varepsilon$-NSGAII, please refer to the background material provided in Chapter 2, Section 2.1.2.

### 3.5 Test Case Descriptions

For the computational scaling analysis, test cases of varying size were derived from the full 29 well test case (described in Section 3.3.1) to demonstrate the effects of increasing problem sizes (defined as the number of sampling wells) on the computational demands of using the $\varepsilon$-NSGAII to approximate the four-objective Pareto surface. To accomplish this, the least important wells were eliminated from the full 29 well test case based on previous results attained by Reed and Minsker [63] in order to develop a set of smaller test cases which could be enumerated within a reasonable time frame. The relative importance of wells was defined in this case by the well sampling frequency distribution associated with Reed and Minsker’s best approximation to the true Pareto-front obtained using the original NSGAII. Based on preliminary enumeration analyses, it was then determined that a 25 well test case represented the upper bound problem size which could be enumerated in a reasonable time frame (approximately six days of continuous computing on a Pentium IV 3.0 GHz processor). The true four-objective Pareto-optimal solution set was then generated for the 25 well test case by evaluating all $2^{25}$ (over 33.5 million) possible well sampling schemes in terms of the four design objectives defined previously in Section 3.3.2. For the enumeration, infeasible solutions (violating the constraint described by Equation 3.2) and inferior solutions (i.e., solutions dominated in terms of at least one design objective) were eliminated from consideration, resulting in the true Pareto-optimal solution set. Smaller test cases were then generated by eliminating individual wells from the 25 well test case based on the well sampling frequency distribution obtained from the 25 well enumeration. Test cases ranging from 18 to 25 wells were then developed based on this methodology and each test case was subsequently enumerated to obtain the true four-objective Pareto-optimal solution set for each test case. Knowing the Pareto-optimal set for each test case allows for rigorous assessment of the computational scaling of the $\varepsilon$-NSGAII algorithm. A cross-sectional slice of the simulated PCE contamination...
Figure 3.1. Cross-sectional slice of the PCE contamination plume representing the LTGM test case. There are 29 well locations available for sampling with one to three sampling locations available along each well’s vertical axes. Wells eliminated to create each of the 18 through 25 well test cases are shown in the figure.

plume is shown in Figure 3.1 along with the 29 well sampling locations. The table associated with the figure indicates which wells were eliminated in order to create each of the 18 through 25 well test cases explored.

3.6 Computational Experiment

The computational scaling characteristics of the $\varepsilon$-NSGAII is tested using the true Pareto-optimal solution sets for each of the 18 through 25 well test cases. Following enumeration, the $\varepsilon$-NSGAII was used to approximate these tradeoffs and its evolutionary operators were parameterized as follows: probability of crossover, $\alpha_c = 1.0$, probability of mutation, $\alpha_m = 1/p$ where $p$ is the population size, crossover distribution index, $\tau_c = 15$, and the mutation distribution index, $\tau_m = 20$. The $\varepsilon$-NSGAII’s adaptive population sizing was initialized using 10 individuals. Epsilon resolution settings for the four design objectives: $\varepsilon_{\text{Cost}}$, $\varepsilon_{\text{Error}}$, $\varepsilon_{\text{Uncert}}$, and $\varepsilon_{\text{Mass}}$ were set to 1.0, 0.0001, 0.0001, and 0.000001 respectively. These values represent the precision with which each objective is quantified and were chosen in this case to represent a high precision Pareto-optimal set. Since MOEA search is initialized with randomly generated populations and since evolutionary operators
are probabilistic, the process can yield high variability in search efficiency and reliability. It is standard practice to overcome this variability by running EMO algorithms for a distribution of “seeds” for the random number generator which is used to initialize and guide their probabilistic search. For this work, scaling analysis across the eight different LTGM test cases was characterized using 50 random seed trial runs for each problem size (i.e., a total of $8 \times 50 = 300$ trial runs).

Since MOEAs are stochastic search methods, the approach to the true Pareto-front of difficult problems usually occurs asymptotically in terms of the number of design evaluations needed to perfectly capture the optimal set of solutions. To ensure computational tractability, 80 percent of the true Pareto-optimal set was approximated for each of the problem sizes. In order to accurately quantify the percentage of Pareto-optimal solutions found, the $\varepsilon$-performance metric [108] was used to determine the percentage of algorithm solutions found within a user specified $\varepsilon$ distance of the reference (i.e., Pareto-optimal) set, which was generated by enumerating all possible well sampling schemes. The first step in the calculation of the metric is to apply the $\varepsilon$-dominance concept to the reference set according to user specified precision values (see Figure 3.2). The proportion of solutions found within $\varepsilon$ hypercubes of the $\varepsilon$-dominated reference set is then measured by matching solutions from the algorithm set to the reference set. Reference set solutions with a matching algorithm solution receive an indicator score while those with no matching solution receive no score (see Figure 2.3 for an example calculation of the metric). Reference solutions with multiple matching approximation solutions use the solution which is closest in terms of Euclidean distance, allowing the additional solutions to be matched with other reference solutions which may have overlapping $\varepsilon$ hypercubes. The values of this metric range from zero to one, where a metric value of one indicates 100 percent convergence to within $\varepsilon$ of the reference set.

In this chapter, the $\varepsilon$-performance metric was used to rigorously evaluate the algorithm’s progress towards the true Pareto-front by greatly restricting the $\varepsilon$-tolerance around the reference set solutions. Epsilon-tolerance values used by the $\varepsilon$-performance metric for each of the four design objectives ($\varepsilon_{\text{Cost}}, \varepsilon_{\text{Error}}, \varepsilon_{\text{Uncert}},$ and $\varepsilon_{\text{Mass}}$) were set similarly to the $\varepsilon$-dominance settings (1.0, 0.0001, 0.0001, and 0.000001 respectively) as these values were chosen for a high level of objective precision. This means that if a solution was not found within these tolerance limits
for each reference set solution, then it was not considered an optimal solution to the problem and was not included in the calculation of the metric. A target $\varepsilon$-performance level of 80 percent convergence to within the specified $\varepsilon$ tolerance of the reference set (i.e., 80 percent of the reference set has been quantified to a very close precision) was then used as the sole basis for the runtime of the algorithm.

Although the LTGM problem described is inherently suited to a binary 0/1 representation within the MOEA framework (i.e., individuals are represented as strings of ones and zeros to indicate whether or not a well is sampled), a real-coded representation was used. This choice was based on previous study results which indicated superior performance of the algorithm when using a real-coded representation for this particular problem (the improved performance was attributed to the utilization of real-coded evolutionary operators). Therefore, the binary representation of the well sampling schemes was converted to a real-coded representation using variables ranging from 0.0 to 1.0. If the algorithm generated a variable less than 0.5, it was changed to 0.0 and variables greater than or equal to 0.5 were changed to 1.0.
Figure 3.3. Number of Pareto-optimal solutions versus problem size for each of the 18 through 25 well test cases.

3.7 Results

Enumeration of the 18 through 25 well test cases for the four design objectives revealed a linear relationship ($R^2 = 0.98$) between problem size and the number of Pareto-optimal solutions ranging from 525 solutions for the 18 well test case, to 2439 solutions for the 25 well test case. Figure 3.3 shows the number of Pareto-optimal solutions plotted versus problem size (in terms of the number of sampling wells). Annotations shown in the figure provide the exact number of Pareto-optimal solutions for each test case.

Plots of the four-objective Pareto fronts for both the 18 and the 25 well enumerated test cases are presented in Figure 3.4. The Cost, Error, and Uncert objectives are plotted on the $X$, $Y$, and $Z$ coordinate axes and the Mass error objective is represented by the color of the marker. Figures 3.4A and 3.4C show the actual Pareto-optimal surfaces (in this case a volume of points because there are four design objectives) for the 18 and 25 well test cases respectively. Figures 3.4B and 3.4D show planar projections of the Pareto-optimal solution sets onto the planes.
Figure 3.4. Four objective Pareto fronts for the 18 and 25 well test cases. Plots A and C show the actual Pareto surface and plots B and D show the planar projections of the Cost, Error, and Uncert objectives onto the planes formed by the X, Y, and Z coordinate axes. Mass error is represented by the color of the markers.

formed by the X, Y, and Z coordinate axes of the 18 and 25 well test cases respectively. Viewing the results using the projections shown in Figures 3.4B and 3.4D provides a better representation of the tradeoffs between the design objectives. The interested reader is invited to explore the electronic version of this article which provides a full color illustration of Figure 3.4. The enumeration of the 18 through 25 well test cases reveals a great deal regarding the scaling properties of the LTGM problem. The main observation to be drawn from Figure 3.4 is the strong geometric similarity between all of the Pareto-optimal solution sets (Figure 3.4 illustrates this for both the 18 and 25 well test cases). This indicates that for each of the problem sizes explored, problems with very similar structures are being solved.

Table 3.1 provides an overview of the total number of solutions in each of the problems’ decision spaces, the total number of Pareto-optimal solutions, and the
Table 3.1. Enumeration data for the 18 through 25 well test cases. Data shown includes the number of possible solutions, the number of Pareto-optimal solutions, and the percentage of the search space which was infeasible.

<table>
<thead>
<tr>
<th>Wells</th>
<th>Total Possible Solutions</th>
<th>Pareto-optimal Solutions</th>
<th>% Infeasible</th>
</tr>
</thead>
<tbody>
<tr>
<td>18</td>
<td>262,144</td>
<td>525</td>
<td>65.1</td>
</tr>
<tr>
<td>19</td>
<td>524,288</td>
<td>512</td>
<td>64.0</td>
</tr>
<tr>
<td>20</td>
<td>1,048,576</td>
<td>790</td>
<td>55.8</td>
</tr>
<tr>
<td>21</td>
<td>2,097,152</td>
<td>1193</td>
<td>54.1</td>
</tr>
<tr>
<td>22</td>
<td>4,194,304</td>
<td>1544</td>
<td>51.4</td>
</tr>
<tr>
<td>23</td>
<td>8,388,608</td>
<td>1846</td>
<td>48.2</td>
</tr>
<tr>
<td>24</td>
<td>16,777,216</td>
<td>2005</td>
<td>47.8</td>
</tr>
<tr>
<td>25</td>
<td>33,554,432</td>
<td>2439</td>
<td>45.6</td>
</tr>
</tbody>
</table>

percentage of their decision spaces that were infeasible. The percentage of search space which is infeasible according to the objective formulation presented in Section 3.3.2 decreases (from 65.1 to 45.6 percent) as the number of wells is increased. For the 18 well test case, there are a total of 525 Pareto-optimal designs out of the 262,144 potential sampling schemes and for the 25 well test case, there are a total of 2439 Pareto-optimal designs out of the 33,554,432 potential sampling schemes.

Figure 3.5 shows the average number of function evaluations required by the \( \varepsilon \)-NSGAII to attain an \( \varepsilon \)-performance measure of 80 percent versus problem size. The upper curve (composed of circular markers) indicates a quadratic growth rate between the \( \varepsilon \)-NSGAII’s computational requirements and the LTGM problem size. The points on the curve represent an average of the total function evaluations required for 50 random seed trials and the error bars represent the 90th percentile range of random seed performance. In this plot, we can see that the computational cost of solving the LTGM problem grows quadratically from approximately 34,000 function evaluations for the 18 well test case to over 590,000 function evaluations for the 25 well test case (on average). In addition, the variability in the computational demands posed by achieving the target performance level increases substantially with increasing problem size. The increasing variability in computational demands with larger problem sizes shows that random seed effects will have a more severe impact on the reliability of MOEAs with increasing problem sizes. For instance, the range of performance of the 18 well case is approximately...
Computational scaling results for the \( \varepsilon \)-NSGAII applied to the 18 through 25 well LTGM test cases. Also shown are the results attained using the \( \varepsilon \)-dominance concept to approximate the Pareto set. The \( \varepsilon \)-NSGAII’s computational demands increase approximately quadratically \([O(l^2)]\) where \( l \) is problem size\] when solving successively larger LTGM problems. However, using \( \varepsilon \)-dominance to constrain the Pareto-set size to approximately 500 solutions results in approximately linear scaling \([O(l)]\) of computational demands with increasing problem size.

16,900 function evaluations while the range of performance of the 25 well case is approximately 292,000 function evaluations. Both ranges represent approximately 50 percent of the mean, but for the 25 well test case, this level of reliability in terms of random seed performance has huge implications with regard to the computational requirements of the LTGM problem (especially when these runs required approximately four hours per random seed running on a Dell Pentium IV 3.0 GHz processor running Microsoft® Windows XP).

Epsilon-dominance archiving can be used to limit the quadratic growth of the LTGM problem if users are willing to accept an increasingly courser approximation to the Pareto front with increasing problem size. Since both the 18 and 19 well test cases contained approximately 500 Pareto-optimal solutions, \( \varepsilon \)-dominance settings were manipulated for the 20 through 25 well test cases to result in Pareto-optimal
Table 3.2. Epsilon settings used to approximate the Pareto-optimal sets of the 20 through 25 well test cases. The original $\varepsilon$ settings used for the 18 and 19 well test cases are shown as well.

<table>
<thead>
<tr>
<th>Wells</th>
<th>$\varepsilon_{\text{Cost}}$</th>
<th>$\varepsilon_{\text{Error}}$</th>
<th>$\varepsilon_{\text{Uncert.}}$</th>
<th>$\varepsilon_{\text{Mass}}$</th>
<th>Sols.</th>
</tr>
</thead>
<tbody>
<tr>
<td>18</td>
<td>1.0</td>
<td>0.0001</td>
<td>0.0001</td>
<td>0.000001</td>
<td>516</td>
</tr>
<tr>
<td>19</td>
<td>1.0</td>
<td>0.0001</td>
<td>0.0001</td>
<td>0.000001</td>
<td>498</td>
</tr>
<tr>
<td>20</td>
<td>1.0</td>
<td>0.4</td>
<td>0.7</td>
<td>0.1</td>
<td>503</td>
</tr>
<tr>
<td>21</td>
<td>1.0</td>
<td>0.9</td>
<td>1.4</td>
<td>0.5</td>
<td>503</td>
</tr>
<tr>
<td>22</td>
<td>1.0</td>
<td>1.0</td>
<td>1.5</td>
<td>0.5</td>
<td>499</td>
</tr>
<tr>
<td>23</td>
<td>1.0</td>
<td>1.1</td>
<td>1.7</td>
<td>0.8</td>
<td>505</td>
</tr>
<tr>
<td>24</td>
<td>1.0</td>
<td>1.0</td>
<td>1.5</td>
<td>0.8</td>
<td>500</td>
</tr>
<tr>
<td>25</td>
<td>1.0</td>
<td>1.1</td>
<td>1.7</td>
<td>0.8</td>
<td>499</td>
</tr>
</tbody>
</table>

sets containing approximately 500 solutions each. This results in a reduction of the Pareto-optimal set to a size similar to the 18 and 19 well test cases while at the same time, incorporating additional sampling wells. Epsilon-dominance settings used to scale the 20 through 25 well test cases are shown in table 3.2. Readers should note that this analysis clearly shows that the computational demands of MOEAs are closely linked to the size of the Pareto-optimal set being approximated. Since the total number of evaluations used by an MOEA is a function of its population size and run duration, these results also enforce the importance of the adaptive population sizing used by the $\varepsilon$-NSGAII.

Results achieved by the $\varepsilon$-NSGAII through the $\varepsilon$-dominance approximation of the 20 through 25 well test cases are shown in Figure 3.5 (star markers). This figure demonstrates that the $\varepsilon$-NSGAII’s $\varepsilon$-dominance archiving provides a mechanism for approximating the Pareto-optimal set, limiting its size, and potentially attaining a linear scaling of the computational cost with respect to problem size. Similarly to the quadratic scaling results, the linearly scaled approximation results are shown with the 90th percentile range of random seed performance indicated by error bars.

Figure 3.6 illustrates the application of $\varepsilon$-dominance to the full Pareto-optimal set of the 25 well test case to yield successive reductions in the set size. Figure 3.6A shows the planar projections of the full Pareto-optimal set for the 25 well test case shown in Figure 3.4D. The Cost, Error, and Uncert objectives are represented by X, Y, and Z coordinates and the Mass objective is represented by the color of the markers. Proceeding from Figure 3.6A to Figure 3.6B, applying $\varepsilon$ values of 1.0, 0.2,
Figure 3.6. Using $\varepsilon$-dominance to control Pareto set size of the 25 well test case. Starting at the full Pareto-optimal set shown in plot A, successively relaxing the precision requirements of the problem through the manipulation of the $\varepsilon$-dominance parameters for each objective (shown in square brackets for plots B through D), reduces the set size while maintaining geometric similarity to the full Pareto set.

0.2, and 0.01 to the Cost, Error, Uncert, and Mass objectives respectively reduces the Pareto-optimal set (through approximation) to 1794 solutions. Proceeding to Figures 3.6C and 3.6D, applying even more relaxed $\varepsilon$ settings to the objectives reduces the set to 499 solutions (as was used for the scaling analysis). Figure 3.6 clearly shows that although the Pareto-optimal set size is reduced, the geometric properties of the set (including the extents of the objectives) are well preserved. This also highlights that using $\varepsilon$-dominance archiving to approximate the Pareto-optimal solution set will still yield an excellent representation of the design tradeoff geometries to support decision making.

Figure 3.7 illustrates the computational savings achieved by using $\varepsilon$-dominance to control the size of the Pareto optimal sets. Computational savings are expressed in terms of a percentage reduction in the average function evaluations required by
Figure 3.7. Percent computational savings (circle markers) and percent reduction in variability (star markers) resulting from the use of $\varepsilon$-dominance to control Pareto set size.

the $\varepsilon$-NSGAII (circle markers) and percentage reduction in random trial variability (star markers). Random trial variability was measured using the 90th percentile interquantile range of random trial performance. In other words, the best 5 percent and worst 5 percent of the random trials were eliminated from consideration when computing the range (to eliminate outliers). For the 25 well test case, the average computational cost decreases from over 593,000 function evaluations to generate the full Pareto-optimal set, to approximately 140,000 function evaluations required to generate a 500 solution approximation to the full set. This represents a 76 percent decrease in the computational requirements of the 25 well test case when an approximation is accepted. Both Figures 3.5 and 3.7 show that the reliability of the algorithm is greatly improved for increasing problem sizes (error bars in Figure 3.5 and star markers in Figure 3.7) when $\varepsilon$-dominance is used for approximation. For example, the random seed trial variability (90th percentile) for the 25 well test case is reduced from over 191,000 function evaluations when generating the full Pareto-optimal set, to approximately 54,000 function evaluations when generating an approximation containing 500 solutions. This represents a 72 percent reduction (or improvement) in the range of random seed performance. In general, Figure 3.7 shows that $\varepsilon$-dominance approximation can provide increasing computational savings and improved reliability for increasing problem sizes.
Figure 3.8 presents the $\varepsilon$-NSGAII’s runtime results for the $\varepsilon$-performance metric versus total design evaluations for each of the 18 through 25 well test cases. The dynamic results shown in the figure represent the mean performance across the 50 random seed trials used to solve each test case. The test cases associated with each performance line are annotated above the plot. Figure 3.8A shows the scaling dynamics which the $\varepsilon$-NSGAII achieves when seeking the full Pareto-optimal set for each test case. This plot clearly shows the influence of problem size on search efficiency as the dynamics of each test case clearly differ from one another. Smaller test cases (namely the 18 through 21 well cases) approach the target level of performance ($\varepsilon$-performance = 80 percent) very quickly. At approximately the 21 well test case and beyond, the algorithm’s progress toward the target level of performance becomes increasingly more computationally demanding and reflects the quadratic growth in computational cost associated with increasing problem size as observed previously in Figure 3.5. Figure 3.8B presents the runtime dynamics of the $\varepsilon$-NSGAII when $\varepsilon$-dominance archiving is used to approximate the Pareto-optimal solution set. This plot presents the dynamics of the LTGM test cases which were scaled according to the $\varepsilon$ settings presented in Table 3.2. In this figure, search progress towards the target $\varepsilon$-performance level of 80 percent is dramatically faster than when seeking the full Pareto-optimal set. In fact, if the dynamics achieved when approximating the Pareto set for all test cases in Figure 3.8B were projected onto the dynamics achieved when searching for the full Pareto set shown in Figure 3.8A, they would easily fit within the dynamics of the 18 through 22 well test cases.

3.8 Discussion

This chapter demonstrated the effects of increasing LTGM problem size on the computational requirements of the $\varepsilon$-NSGAII. The decision space of the LTGM problem increases according to $2^l$ where $l$ represents the number of binary decisions (in this case, a yes/no decision of whether or not to sample from a predetermined well location). However, enumeration of the 18 through 25 well test cases revealed a linear relationship between problem size and the number of Pareto-optimal solutions. Test cases below 18 wells are easily tractable given current computational ability but enumeration of the 25 well test cases required 6 days of continuous com-
Figure 3.8. Dynamic algorithm performance for the 18 through 25 well test cases. These plots show the mean $\varepsilon$-performance of the 50 random seed trial runs for each test case plotted versus function evaluations. Plot A shows the dynamic performance of the $\varepsilon$-NSGAII when searching for the full Pareto-optimal set and plot B shows the dynamic performance of the algorithm when $\varepsilon$-dominance is used to approximate the set.

puting on a Pentium IV, 3 GHz processor. Each time a well is added, the computational requirements of complete enumeration are doubled, ultimately motivating the need for efficient algorithms capable of generating the Pareto-optimal solution set (or an approximation thereof) without evaluating every possible solution. This chapter showed that the $\varepsilon$-NSGAII was capable of generating a close approximation (to within 80 percent in terms of the $\varepsilon$-performance metric) of the Pareto-optimal solution set for the 25 well test case in approximately 4.3 hours (on average) as opposed to the 6-days required for enumeration. This represents a 97 percent reduction in computational requirements. Computational requirements were further reduced (76 percent) by accepting a 499 solution approximation to the true 2439 solution Pareto-optimal front through the utilization of $\varepsilon$-dominance. The use of $\varepsilon$-dominance revealed that through approximation, the computational complexity of using MOEAs to solve LTGM problems could be reduced from quadratic to approximately linear scaling within the range of test cases examined.

This has significant implications for the future of LTGM design. For example, the largest test case examined consisted of 25 decision variables occurring at a
single point in time. If however, the site required quarterly sampling, the addition
of a temporal component quickly increases the decision space to 100 variables or
$2^{100}$ potential sampling schemes. As it currently requires approximately 500,000
function evaluations to solve the 25 well test case, the introduction of quarterly
sampling represents a 16 fold increase in the number of function evaluations (i.e.,
8-million) because of the $\varepsilon$-NSGAII’s quadratic scaling. In addition, if a space-
time evaluation scheme is used, the computational requirements of each function
evaluation will be greatly increased from that required by QK as was used in this
chapter.

The LTGM test cases analyzed represent a lower bound computational com-
plexity for MOEAs applied in water resources applications. More complex water
resources applications with tens or hundreds of integer or continuous decision vari-
ables may have a more severe growth rate for their Pareto-optimal set sizes and
computational demands. Epsilon-dominance was explored as a method of control-
ling the computational scaling of the LTGM problem as the number of monitoring
wells was increased. This method provides a means of approximating the Pareto-
optimal set based on user defined precision goals and a willingness to accept the
approximation. However, as problem size increases, the severity of the approxi-
mation increases as well. Although optimization algorithms which are capable of
attaining the true Pareto-front are the ideal goal, this chapter demonstrates that in
reality, water resources scientists and engineers will have to accept approximations
to their applications’ Pareto-sets and advance the field by investigating innovative
new algorithms that scale subquadratically.

3.9 Conclusions

This chapter contributed a detailed assessment of how increasing LTGM problem
sizes (measured in terms of the number of decision variables being considered)
impacted the computational complexity of using the $\varepsilon$-NSGAII to solve an LTGM
application. LTGM test cases composed of 18 to 25 sampling wells were first
enumerated using four design objectives. The $\varepsilon$-NSGAII was then used to approx-
imate the Pareto-optimal solution set of each test case to an $\varepsilon$-performance level of
80 percent. Results indicated linear scaling of Pareto set size versus problem size.
The $\varepsilon$-NSGAII required quadratic scaling but the incorporation of $\varepsilon$-dominance to approximate the Pareto-optimal set resulted in approximately linear computational scaling. Although MOEAs are capable of solving challenging water resources applications, the consideration of larger problems will require users to accept approximations to their Pareto optimal sets as well as research that will develop improved algorithms that are capable of scaling subquadratically.
CHAPTER 4

The Epsilon-Dominance Hierarchical Bayesian Optimization Algorithm for Large Multiobjective Monitoring Design Applications

4.1 Abstract

This chapter focuses on the development of a next generation Multi-Objective Evolutionary Algorithm (MOEA) that can learn and exploit complex interdependencies and/or correlations between decision variables in monitoring design applications to provide more robust performance for large problems (defined in terms of both the number of objectives and decisions). The proposed MOEA is termed the Epsilon-Dominance Hierarchical Bayesian Optimization Algorithm (\(\varepsilon\)-hBOA), which is representative of a new class of probabilistic model building evolutionary algorithms. The \(\varepsilon\)-hBOA has been tested relative to a top performing traditional MOEA, the Epsilon-Dominance Nondominated Sorted Genetic Algorithm II (\(\varepsilon\)-NSGAII) for solving a four-objective LTGM design problem. A comprehensive performance assessment of the \(\varepsilon\)-NSGAII and various configurations of the \(\varepsilon\)-hBOA have been performed for both a 25 well LTGM design test case (representing a relatively small problem with over 33-million possible designs), and a 58 point LTGM design test case (with over \(2.88 \times 10^{17}\) possible designs). The results from this comparison indicate that the model building capability of the \(\varepsilon\)-hBOA greatly enhances its performance relative to the \(\varepsilon\)-NSGAII, especially for
large monitoring design problems. This work also indicates that decision variable interdependencies appear to have a significant impact on the overall mathematical difficulty of the monitoring network design problem. The work presented in this chapter has been published in the journal *Advances in Water Resources* [64].

### 4.2 Introduction

This chapter contributes a new Multi-Objective Evolutionary Algorithm (MOEA) termed the Epsilon-Dominance Hierarchical Bayesian Optimization Algorithm ($\varepsilon$-hBOA), which has been developed to solve large, long-term groundwater monitoring (LTGM) design problems (defined in terms of both the number of design objectives and decision variables). LTGM networks use spatially distributed wells to characterize groundwater contamination over long time scales (ranging from years to decades) to ensure that the contamination does not pose an unacceptable risk to humans and the environment [7]. Management strategies for the health risks associated with contaminated groundwater must also consider the long-term economic costs associated with site monitoring. Federal expenditures on LTGM for the years 2000 through 2010 are estimated at 5-billion US dollars [7], motivating the need for cost effective LTGM design strategies that are protective of human and ecological systems. However, designing LTGM networks for contaminated groundwater is a challenging problem that has long been recognized to suffer from a “curse of dimensionality” [14]. This is largely due to their discrete decision spaces that grow exponentially as the number of contaminant measurements, their locations, and sampling rates are considered.

For example, the decision space (or total number of possible designs) of a LTGM network design problem grows according to the equation $2^{MT}$ where $M$ represents the number of possible sampling locations, and $T$ is the number of sampling periods. The 2 in this equation assumes a binary decision where there are two options, sample (1) or do not sample (0). If there are 25 potential sampling locations ($M = 25$) to characterize a contaminant at a snapshot in time ($T = 1$), this particular network design problem contains $2^{25}$ (or over 33-million) possible designs. Adding just one additional sampling period results in a significant increase in the size of the decision space (over $1.12 \times 10^{15}$ possible designs for $M = 25$ and
The exponential scaling of the LTGM problem as well as its discrete decisions have motivated several researchers to utilize evolutionary algorithms (EAs) in its solution since the early 1990s [39, 40, 63, 109, 110, 166, 167]. EAs evolve populations of designs toward optimality using processes that are analogous to selection, mating, and mutation. While previous LTGM design applications using EAs have generally been deemed successful in their attempts to generate approximately optimal solutions, they have typically either been applied to problems of limited size and complexity or have been limited in their exploration of designs. In general, the LTGM design problem is inherently suited to formulations that take into account multiple design objectives simultaneously (e.g., cost, uncertainty, reliability, etc. [14]). For the past 20 years, a variety of MOEAs have been developed and shown to be capable of optimizing highly nonlinear, discrete, and non-convex objective space landscapes without differentiation [92–94]. In addition, MOEAs’ population-based search enables them to evolve approximations for entire tradeoff (or Pareto [81]) surfaces within a single optimization run.

Building on prior multiobjective LTGM design applications [7, 31, 38–40, 45–47, 63, 65, 66, 80, 87, 150, 151], recent studies [63, 72] have shown that MOEAs can be combined with visualization tools to solve LTGM problems with three or more objectives (termed high-order Pareto optimization problems). Multiobjective formulations add to the complexity of LTGM design problems by increasing the number of solutions that must be evolved. It has also been shown that the required population size for an MOEA to maintain a diverse representation of the LTGM design tradeoffs is directly related to the complexity of the objective space, and that adding objectives generally increases this complexity [112, 168]. Moreover, functional interdependencies between decision variables also increases the difficulty and computational search requirements for optimization problems [116]. In short, this work posits that a new class of combinatorial search algorithm is necessary to resolve the LTGM problem’s potential for large numbers of design objectives and sampling decisions that can have complex spatial and/or temporal interdependencies depending on the contaminant plume.

The work presented in Chapter 3 showed that as the number of LTGM sampling decisions increases linearly, the computational complexity of using the
Epsilon-Dominance Nondominated Sorted Genetic Algorithm II (\(\varepsilon\)-NSGAII) grows quadratically. Computational scaling defines how problem size (i.e., number of wells, contaminants sampled, sampling times) increases the average number of design evaluations required by an MOEA to evolve a high quality solution set for a given LTGM design problem. The \(\varepsilon\)-NSGAII has been shown previously to be highly robust at solving LTGM design problems relative to other state-of-the-art MOEAs \[109\]. Kollat and Reed \[109\] concluded that traditional MOEAs such as the \(\varepsilon\)-NSGAII are limited to increasingly lower quality approximations for large LTGM problems given their large solution set sizes and the limits posed by their quadratic computational complexities. This chapter builds on the work presented in Chapter 3 \[73\] by introducing a new MOEA termed the Epsilon Dominance Hierarchical Bayesian Optimization Algorithm (\(\varepsilon\)-hBOA), which uses Bayesian networks (a form of probabilistic model building) to optimize large LTGM design problems. This work seeks to determine if the incorporation of probabilistic modeling building improves the ability of an MOEA to solve LTGM design applications, and if so, what is the best implementation of model building to maximize search success?

The remainder of this chapter proceeds as follows. Section 4.3 provides details of the four-objective LTGM test case problem formulation. Section 4.4 discusses the \(\varepsilon\)-NSGAII and the \(\varepsilon\)-hBOA, as well as the various configurations of the \(\varepsilon\)-hBOA tested in this chapter. Section 4.5 describes the computational experiment used to demonstrate the performance of the new \(\varepsilon\)-hBOA algorithm. Sections 4.6 and 4.7 present and discuss the results. Section 4.8 concludes with key findings and recommendations for future work.

4.3 Monitoring Case Studies

4.3.1 Test Case Descriptions

The LTGM network design test case used in this work is identical to that described in Chapter 3, Section 3.3.1. For this test case, there are 29 sampling wells located throughout the contaminant plume, each with no more than three sampling ports available along its vertical axis. This results in a total of 58 sampling locations.
where PCE concentration data is available. Additional details on this test case can be found in Maxwell et al. [169].

In order to test how LTGM problem size impacts the performance of a traditional MOEA versus that of the new \( \varepsilon \)-hBOA algorithm, two LTGM network design test cases were developed - one small and one large. The small test case was developed by eliminating four wells from the 29 well test case for a total of 25 sampling wells. Each of the 25 well locations was then treated as a decision. This means that if a particular well is chosen for sampling and it has multiple sampling ports available, all sampling ports are utilized. The size of this test case was chosen in part because it can be enumerated (solved to completion) by evaluating all \( 2^{25} \) (or over 33.5-million) possible designs given current computational constraints. Knowing the true solution to the LTGM test case ultimately provides the most rigorous means of assessing MOEA performance. In order to test MOEA performance for a large network design problem, the second larger test case treats each of the 58 sampling ports available at the 29 well locations as decision variables. This increases the size of the decision space to \( 2^{58} \) (or over \( 2.88 \times 10^{17} \)) possible designs. Given that an LTGM problem of this size cannot be enumerated, a “best known” reference set was developed by combining all of the solutions found by all algorithm runs. Hereafter, the small test case will be referred to as the 25 well test case, and the large test case will be referred to as the 58 point test case.

### 4.3.2 Design Objectives

Four design objectives were chosen, each of which were minimized. The design objectives included: (1) the cost of sampling the contaminant plume, (2) the error relative to using all available data points to create spatial plume maps, (3) the spatial uncertainty associated with characterizing the plume, and (4) the error associated with estimating the mass of contaminant in the plume. Hereafter, each of the four design objectives are referred to as Cost, Error, Uncert, and Mass respectively. The design objectives and penalization scheme are defined identically as presented in Chapter 3, Section 3.3.2 and will not be repeated here. The only minor differences are that the Cost objective can now range from 0 to 58 for the larger 58 point test case and the corresponding maximum penalty, \( f_{\text{Cost}}^{\text{max}} \) utilized in
Equation 3.8 is 58 as well. In addition, PCE concentration estimates were obtained at unsampled locations throughout the sampling domain using QK similarly to Chapter 3, Section 3.3.3.

### 4.4 Multiobjective Evolutionary Optimization

The goal of multiobjective optimization is to identify the Pareto-optimal tradeoffs between an application’s objectives. These tradeoffs are composed of the set of solutions that are better than all other solutions in at least one objective and are termed non-dominated or Pareto-optimal solutions [81]. MOEAs’ population-based search enables them to evolve entire tradeoff (or Pareto [81]) surfaces within a single optimization run for problems with huge decision spaces. For additional resources on multiobjective optimization, the reader is encouraged to refer to the books by Deb [96] and Coello Coello [78] for comprehensive introductions to these topics.

#### 4.4.1 Epsilon-Nondominated Sorted Genetic Algorithm-II (\(\varepsilon\)-NSGAII)

The \(\varepsilon\)-NSGAII has been shown to perform very well relative to other state-of-the-art MOEAs at solving LTGM network design problems [63, 73, 109]. In addition, the \(\varepsilon\)-NSGAII’s performance has been validated extensively on a variety of test functions and applications [91, 108, 170, 171]. The \(\varepsilon\)-NSGAII is based on the original NSGAII [101], which uses nondomination sorting and crowding distance to maintain solution diversity, simulated binary crossover (SBX) [165], polynomial mutation[96], and elitism [96]. The \(\varepsilon\)-NSGAII expands on the original NSGAII through the inclusion of dynamic population sizing [113] and \(\varepsilon\)-nondominance archiving [105, 111]. Please refer to Chapter 2, Section 2.1.2 for a detailed description of the \(\varepsilon\)-NSGAII.
4.4.2 Epsilon-Dominance Hierarchical Bayesian Optimization Algorithm (\(\varepsilon\)-hBOA).

For the LTGM design problem, decisions potentially have complex interdependencies or correlations as a result of the underlying physics of flow-and-transport, the influence of domain geometry, and the underlying properties of the performance criteria considered. Recent literature in the field of evolutionary computation [114, 172] has shown that identifying these complex interdependencies and preserving them while generating new solutions is vital for EAs solving difficult engineering and science problems. The traditional MOEA operators of crossover and mutation assume that all of a problem’s decisions are statistically independent, which may limit their performance for some challenging applications such as LTGM design. A new class of EAs termed Probabilistic Model Building Genetic Algorithms (PMBGAs) “learn” the linkages between decision variables by building probabilistic models that express the interdependencies between decision variables to better preserve these links throughout the evolutionary process [172]. PMBGAs differ from traditional genetic algorithms by replacing the crossover and mutation operators with a probabilistic model [116]. Each generation, promising solutions are selected from the population and a probability distribution is estimated based on the conditional relationships of decision variables. New child solutions are then generated by sampling the estimated probability distribution. PMBGAs can be useful on any class of problem where decision variables are correlated (or linked) with one another. For example, in LTGM design, the decision to sample from a certain location in space and time affects other sampling decisions because of the spatio-temporal structure of a contaminant plume. Prior PMBGA literature has shown that model building algorithms are generally less effective than traditional EAs for easy problems that are linearly separable (i.e., independent decisions) and increasingly superior as problem difficulty increases (i.e., increasing interdependencies between decisions) [172].

The Hierarchical Bayesian Optimization Algorithm (hBOA) developed by Pelikan [116] attempts to overcome the difficulties posed by large sets of inter-related decision variables by building Bayesian network models of the decision space. Bayesian networks [173] use directed acyclic graphs (defined as graphs with edges...
directed between vertices where there are no cycles which center on only one vertex \([174]\)), to model sets of conditional probabilities between variables. The network contains a structure showing which variables are independent and a set of conditional probabilities for each variable \([173]\). Each decision variable of the problem is represented as a node in the Bayesian network. Edges, which represent the conditional probabilities between each of the decision variables, are used to establish relationships between the nodes. For example, if two decisions (nodes) are completely independent of one another, no edges connect these nodes.

Figure 4.1 presents a hypothetical illustration of the potential for conditional dependencies that might exist between monitoring wells for a contamination plume in a typical LTGM application. Figure 4.1A illustrates the case where the probability of sampling from each of the five potential monitoring points are completely independent of one another (i.e., there are no edges connecting the decisions). This represents the assumption made by the \(\varepsilon\)-NSGAII. Figure 4.1B illustrates the case when sampling certain wells may be conditionally dependent on whether or not a sample has been taken at another well. For example, the decision to sample from well 3 may depend on whether or not contamination exists at wells 1 or 2. While the dependency structure of network design problems has the potential to be quite complex, the identification of at least some of these dependencies may greatly aid in improving the search efficiency of MOEAs.

The Bayesian network model is built iteratively using a greedy search algorithm \([117]\) that performs elementary network operations such as edge additions, removals, and reversals, intended to maximally improve the quality of the model. Changes in model quality can be measured using a variety of information theory based metrics \([116]\), some of which include the Bayesian Dirichlet metric (BD) \([117]\), the K2 Metric \([175]\), or the Bayes Information Criterion (BIC) metric \([176]\). These metrics enforce Occam’s Razor by penalizing complex models that do not significantly improve their predictive skill. Regardless of the chosen metric, network operations that result in the greatest metric score increase are the basis for building the Bayesian network model \([116]\). Once elementary network operations no longer exhibit significant improvements in the model, the construction of the network is assumed adequate and the Bayesian network model building is terminated. In each generation, the Bayesian network models are built from the binary
Figure 4.1. Potential conditional dependencies of sampling wells for a contamination plume. Figure A demonstrates the case when all wells are sampled independently and Figure B illustrates the case where conditional probabilities may exist in sampling various wells.

strings that compose the current population. Models are rewarded when they find binary variable combinations within the population that are associated with highly ranked solutions.

In complex problems such as LTGM, there may be interactions within highly correlated groups of decisions as well as across those groups (e.g., well clusters in the source area or sampling points along a plume’s boundaries). The hBOA was selected because of its ability to solve hierarchically difficult problems. Mathematically, hierarchy is defined by the ability to break down a system into subsystems, each of which in turn represents a hierarchy themselves, until some decomposable bottom level is reached [177]. For example in the context of the LTGM design test case used, combinations of wells are important for defining major plume zones such as the source area or leading edge. Within each area, each well can define concentrations at up to three vertical locations, and it would seem reasonable that a bottom level of problem decomposability would correspond to individual sampling point decisions. The goal in developing a hierarchical solver such as the hBOA is to extend its capabilities to problem structures that exhibit strong multivariate links between single decisions as well as clusters of decisions. The hBOA
learns proper hierarchical problem decomposition using a technique referred to as *chunking* within the Bayesian network model building process. Chunking is used within the model building process to allow groups of decision variables which are related at lower hierarchical levels to be clustered, and subsequently used to model higher order interactions [116]. Decision graphs are used to store the conditional probabilities of each decision variable. The decision graphs contain various nodes, all of which are permitted to have multiple parents (except for the root node).

The performance of the hBOA has been demonstrated on both test functions [116, 118–120] and various real-world applications such as the notably difficult Ising spin-glass systems [121, 122]. However, there are some known limitations to the hBOA. The Bayesian network model building is challenging and adds complexity to the algorithm. In addition, population size while seen as important within conventional EAs [178], is very important to the hBOA since the quality of its probabilistic models is directly correlated with the size of the population sampled.

The hBOA algorithm is fundamentally different from a traditional MOEA like the $\varepsilon$-NSGAII because it eliminates the crossover and mutation operators, and replaces these with Bayesian network model building. Also, the original hBOA algorithm was developed to handle only a single design objective. Since the $\varepsilon$-NSGAII had been shown to be highly effective at solving LTGM design problems in the past [63, 73, 91, 109], it was decided that the new $\varepsilon$-hBOA should represent a combination of the strengths of both the $\varepsilon$-NSGAII and the original hBOA. To accomplish this, the Bayesian network model building of hBOA was used to replace the SBX crossover and polynomial mutation operators of the $\varepsilon$-NSGAII. This results in a multiobjective algorithm with dynamic population sizing and $\varepsilon$-nondominance archiving options. The new $\varepsilon$-hBOA algorithm proceeds as shown in Figure 4.2. Initially, a population of $p$ random designs is generated and the concept of Pareto-dominance is used to assign fitness values to each design based on its performance in terms of each design objective. A Bayesian network is then iteratively generated based on this *parent* population of designs using the BIC metric as a termination criteria. *Child* solutions are then generated by sampling the resulting Bayesian network’s modeled joint probability distribution for sampling designs. The algorithm then proceeds similarly to the $\varepsilon$-NSGAII where Pareto ranking and crowded binary tournament selection are used to fill a new population.
of \( p \) superior designs. The \( p \) children of the new population are eligible for inclusion in the \( \varepsilon\)-nondominated archive and become the parents of a subsequent generation from which the process is repeated until some termination criteria is met. Dynamic population sizing can also be utilized by the new \( \varepsilon\)-hBOA, resulting in a series of “connected runs” each with a unique population size based on search progress (\( \varepsilon\)-nondominated archive size).
4.5 Computational Experiment

The $\varepsilon$-NSGAII and the $\varepsilon$-hBOA algorithms were tested on both the 25 well and 58 point LTGM network design test cases described in Section 4.3.1 to determine the effects of Bayesian network model building on algorithm performance. Several configurations of the new $\varepsilon$-hBOA algorithm were tested to determine the important factors affecting algorithm performance.

4.5.1 Reference Set Generation

The true solution to the 25 well test case was fully enumerated, ultimately providing a rigorous evaluation framework with which to assess algorithm performance. The Pareto-optimal solution set for the 25 well test case is composed of 2,472 solutions and is shown in Figure 4.3A with the Cost, Error, and Uncert objectives plotted on the X, Y, and Z-axes respectively. Since this is a four objective problem, the color of the solution is used to represent the Mass objective where red solutions indicate high mass error and blue solutions indicate low mass error. Objectives values for this Pareto-set range from 8 to 47 in the Cost objective, 0 to 43.7 in the Error objective, 1396 to 1672 in the Uncert objective, and -7 to 1.72 in the Mass objective (which is now scaled logarithmically).

The 58 point test case is significantly more difficult and cannot be enumerated since the decision space is composed of over $2.9 \times 10^{17}$ possible designs. However, a best known solution was generated by combining all solutions found by all trials of all algorithm configurations explored. The union of all solution sets from all algorithm configurations was then reduced to a Pareto-approximation of the true Pareto set. The Pareto-approximation for the 58 point test case contains 22,333 solutions and is shown in Figure 4.3B, where again, the Mass objective is represented by the color of the solutions. Objectives for this Pareto-approximation range from 5 to 58 in the Cost objective, 0 to 48.8 in the Error objective, 1376 to 1702 in the Uncert objective, and -7 to 2.31 in the Mass objective. The most notable feature of the 58 point case is that it generally results in increased ranges of objective values as a result of being able to sample from individual locations along each well. Figure 4.3C shows a closer view of the central region of the Pareto-approximation that highlights the density of solutions. Here it can be seen that at
Figure 4.3. Plots A and B show the 25 well Pareto-set and 58 point Pareto-approximation respectively. Plot C shows an enlarged view of a densely populated area of the 58 point Pareto-approximation. Cost, Error, and Uncert are plotted on the X, Y, and Z-axes, and Mass is plotted using color.

each level of Cost, a three objective tradeoff surface exists between Error, Uncert, and Mass. Figure 4.3C shows that the reference set’s solution density is greater for sampling schemes that yield larger numbers of combinations (e.g., selecting 57 of 58 total locations yields far fewer combinations than does selecting 30 locations as expected).
4.5.2 Performance Assessment of MOEAs

When evaluating the performance of MOEAs, metrics that measure the effectiveness, efficiency, and reliability of the algorithm are important. Effectiveness refers to the ultimate performance of the algorithm (i.e., how well did it optimize the problem), and is evaluated by measuring end-of-run statistics for how well reference sets were captured. Measuring algorithm efficiency is important for the computational costs and overall search dynamics of the algorithm. Was it slow to get started initially? Did it reach its final solution quickly? Algorithm efficiency can be assessed using runtime performance metrics which record the progress of the algorithm throughout its entire run. Reliability provides a measure of the reproducibility of the results of the algorithm for different random seeds. Since MOEAs require random initial populations, the choice of this initial population may or may not influence the effectiveness and efficiency of the algorithm. Ideally, it is desired that the initial population have no bearing on algorithm performance. The reliability of the algorithm is assessed using random seed analysis where multiple random initializations (trials) are run.

Runtime convergence [179] is used in this chapter to measure the average Euclidean distance between an approximation set (i.e., the set of solutions found by the algorithm [180]) and a reference set (i.e., the true Pareto set or best known solution). Small values of convergence are preferred indicating a small average distance to the reference set. The runtime \( \varepsilon \)-performance metric, recently proposed by Kollat and Reed [108] measures the proportion of \( \varepsilon \)-nondominated solutions which have been found within an acceptable \( \varepsilon \)-error for the reference set. This metric uses the concept of \( \varepsilon \)-nondominance where the desired precision for each objective is specified, and the approximation and reference sets are then sorted based on this precision. In this way, the “strictness” of the \( \varepsilon \)-performance metric can be strengthened or relaxed based on the precision requirements of the user. The \( \varepsilon \)-performance metric ranges from zero to one where a metric value of one indicates 100-percent convergence based on the specified \( \varepsilon \)-nondominance precision of the reference set. The unary \( \varepsilon \)-indicator metric [180] quantifies the smallest distance that an approximation set must be translated in order to completely dominate a reference set. Small values of this metric are desirable as this indicates a closer approximation to the reference set. For additional details on the unary \( \varepsilon \)-indicator
metric, see Zitzler and Thiele [181] and Zitzler et al. [180]. The hypervolume metric [182] quantifies the volume of the approximation set with respect to some reference or nadir point. When a reference set is available, this metric can be calculated with respect to the reference set. In other words, when minimizing objectives, the hypervolume can be calculated as the difference in volume between an approximation set and the reference set. When calculating the hypervolume as a difference, small values are optimal indicating small difference with respect to the reference set. The hypervolume metric is an excellent measure of solution set diversity or spread across the full range of tradeoffs. The statistical metrics-based evaluation framework provides a direct measure of performance differences for the tested MOEAs with respect to their ability to converge while maintaining a diverse representation of tradeoffs [180, 183].

4.5.3 Algorithm Configurations

Various algorithm configurations were tested on both the 25 well and 58 point LTGM test cases. Preliminary analysis of a variety of configurations permitted the author to focus on six major configurations which embody the major findings of this work. These six configurations are now described in detail.

ε-NSGAII. This configuration refers to the original ε-NSGAII algorithm, and was chosen as a current performance benchmark. The ε-NSGAII was parameterized to use an initial population size of 12 individuals and an ε-nondominated archive injection rate of 25-percent. This means that following each connected run, a new population is generated based on the archived solutions (at a rate of 25-percent), and 75-percent random solutions. This injection rate was previously found to produce the most efficient algorithm performance for the ε-NSGAII relative to other injection rates [108]. Additional relevant parameters include the probability of SBX crossover = 1.0 [165], the probability of polynomial mutation = 0.02 [96], the distribution index for SBX crossover = 15, and the distribution index for polynomial mutation = 20, all of which are based on prior literature recommendations.

ε-hBOA-Base. This notation refers to a base version of the ε-hBOA algorithm as described in Section 4.4.2 and Figure 4.2. The ε-hBOA-Base implementation re-
places the $\varepsilon$-NSGAII’s traditional crossover and mutation operators with Bayesian network model building and simulation of a new population based on this model. This configuration then uses binary crowded tournament selection available within the $\varepsilon$-NSGAII and originally utilized by the NSGAII [101] to determine whether newly simulated population members should replace their parents. In addition, $\varepsilon$-NSGAII’s dynamic population sizing techniques are utilized within the $\varepsilon$-hBOA with an $\varepsilon$-nondominated archive injection rate of 25-percent. Pelikan [116] showed previously that the population size required for optimal model building within hBOA is on the order of $O(2^{b1.05})$ where $b$ is the building block order and $l$ is the number of binary decision variables. Assuming a lower bound building block complexity of $b = 4$ decisions (i.e., no more than four sampling locations are jointly important), this would require a population of approximately 1000 individuals. This lower bound population size was chosen as the initial population size for the $\varepsilon$-hBOA-Base configuration. However, as stated, this reflects a lower bound complexity and the $\varepsilon$-nondominated archive injection will ensure that the population size increases as search progresses.

$\varepsilon$-hBOA-Archive. This notation refers to a version of the $\varepsilon$-hBOA which utilizes the current $\varepsilon$-nondominated archive in the Bayesian network model building. In other words, following each generation, the current best found solutions that are stored in the $\varepsilon$-nondominated archive are combined with the currently evolving population to build the Bayesian network model. This configuration was tested to confirm if the inclusion of additional (high quality) information in the model building would improve algorithm performance.

$\varepsilon$-hBOA-RTR. The original hBOA algorithm utilized restricted tournament recombination (RTR) as recommended by Pelikan [116] to determine which children should replace parent members of the population. Based on this recommendation, a configuration of the $\varepsilon$-hBOA algorithm which utilized RTR instead of binary crowded tournament recombination was tested.

$\varepsilon$-hBOA-Static. According to Pelikan’s [116] theoretical population sizing requirements for the hBOA, the required population can become quite large, especially for problems with many decision variables. To assess the importance of population size, a static population size variant of the $\varepsilon$-hBOA algorithm was tested that eliminated the dynamic population sizing available within the $\varepsilon$-NSGAII. Re-
quiring the specification of a static population size adds an additional parameter to the algorithm that is difficult to estimate \textit{a priori}. For the smaller 25 well test case, the static population size was chosen based on $2l$ (where $l$ is the number of decisions) generations of evolution [184, 185] (i.e., 50 for the 25 well case) and a maximum runtime of 200,000 evaluations for a total static population of 4,000 individuals. For the larger 58 point test case, the population size was based on the average archive size attained for individual runs of the $\varepsilon$-hBOA-Base. Several prior EA studies [87, 112, 186] have shown that MOEA population size is a direct function of the Pareto set size. The population size is very important to the hBOA since the quality of its probabilistic models is directly correlated with the size of the population sampled.

\textbf{$\varepsilon$-hBOA-Hybrid.} Since one of $\varepsilon$-NSGAII’s main strengths is utilizing small populations initially to pre-condition search and since the hBOA requires large populations to optimally construct the Bayesian network, a hybrid version of the two techniques is tested. The $\varepsilon$-hBOA-Hybrid configuration utilizes the $\varepsilon$-NSGAII to pre-condition search for 10-percent of the total run duration and then switches to a static population size using the $\varepsilon$-hBOA. The static population size utilized by $\varepsilon$-hBOA is strictly based on what the $\varepsilon$-NSGAII found during the initial portion of the run at a 25-percent injection rate. If the $\varepsilon$-NSGAII found 1000 $\varepsilon$-nondominated archive solutions during the first 10-percent of the run, the static population size used by the $\varepsilon$-hBOA would be 4000 individuals, 1000 of which are from the archive and the remaining 3000 of which are initially generated at random. Once the $\varepsilon$-hBOA-Hybrid version switches to the $\varepsilon$-hBOA, the population size remains static.

\textbf{Other Algorithm Parameters.} A number of algorithm parameters were specified identically across all configurations. The number of generations of evolution per connected run was specified as $2l$ [184, 185] where $l$ is the number of decisions. This means that for the 25 well test case, each run contained 50 generations and for the 58 point case, each run contained 116 generations. The maximum runtime was chosen based on computational feasibility and was expressed as a total number of function evaluations. For the 25 well test case, the maximum number of evaluations was set at 200,000 and for the 58 point test case, the maximum number of evaluations was set at 2-million. The $\varepsilon$-dominance precision settings for each of
the four design objectives were chosen to result in very high precision Pareto-sets for each test case and were specified as $\varepsilon = (1.0, 0.01, 0.01, 0.01)$ for each the Cost, Error, Uncert, and Mass objectives respectively. Finally, 50 random seed trials were conducted for each algorithm configuration to provide a means of assessing reliability. Overall, 900 algorithm runs were conducted, 300 for the 25 well test case, 300 to generate the 58 point Pareto-approximation, and another 300 to obtain metric results for the 58 point test case relative to the generated reference set. This represents well over 1-year of continuous computing on a serial machine. However, individual runs were distributed across the LION-XO high performance computing cluster available at The Pennsylvania State University in order to be completed within a reasonable time frame of two weeks.

4.6 Results

4.6.1 Effectiveness of Search

Figure 4.4 displays a histogram showing the relative effectiveness of each algorithm configuration at identifying reference set solutions for the 58 point test case. Light grey bars signify the total percentage of the reference set’s solutions identified by a particular configuration across all random seed trials. The medium grey bars signify the average per seed effectiveness for each configuration and the error bars indicate one standard deviation. The black bars indicate the total unique contribution of each configuration, defined as the number of solutions that a particular configuration contributed across all random seed trials that no other configuration found. From the histogram, it is clear that the $\varepsilon$-hBOA-RTR configuration contributed the least to the reference set in all categories. The $\varepsilon$-hBOA-Base, $\varepsilon$-hBOA-Archive, $\varepsilon$-hBOA-Static, and $\varepsilon$-hBOA-Hybrid configurations contributed the largest percentages overall (between 80 and 85-percent), while the average per seed contribution of the $\varepsilon$-hBOA-Static configuration was the highest at 41.7-percent, more than twice that of all other configurations except the $\varepsilon$-hBOA-Archive. In addition, the $\varepsilon$-hBOA-Static configuration provided the highest contribution of unique solutions at 4.1-percent, more than twice that of any other configuration except the $\varepsilon$-hBOA-Hybrid. These results indicate that on a per seed basis, the
Figure 4.4. Histogram displaying the relative contributions of each algorithm configuration to the 58 point test case reference set. Light gray bars indicate total contribution across all random seed trials, medium gray bars, the average per seed contribution, and the black bars, unique contribution across all seeds.

\( \varepsilon \)-hBOA-Static configuration is capturing the highest percentage of the 58 point test case’s reference set.

Table 4.1 displays metric values for each algorithm configuration for the smaller 25 well test case. Metrics were evaluated when the number of function evaluations (NFE) reached 25,000 and again at the end of the run (NFE = 200,000). The reader is reminded that minimal values of convergence, \( \varepsilon \)-indicator, and hypervolume are preferred while \( \varepsilon \)-performance expresses the proportion of the reference set found, meaning that 1.0 is optimal for this metric. Hypervolume metric results are not available at NFE = 25,000 because the non-dominated sets required to calculate the metric were not output throughout the run, only at the end of the run. This table shows the average metric values for all 50 random seed trials as well as the standard deviations in parentheses. In addition, a Kruskal-Wallis [187] non-parametric statistical test was used to test the statistical significance of performance differences at the 95-percent confidence level for the metric distri-
butions attained for each algorithm configuration. The results of the statistical tests were used to rank the relative performances of each algorithm configuration and the top performing algorithm configurations for each metric are highlighted in bold in Table 4.1. In Table 4.1, the ε-hBOA-Static configuration is the worst performer early in the run in terms of all metrics, but subsequently becomes the best performer by the end of the run. In terms of early run results, it appears that the ε-hBOA-Archive and ε-hBOA-Base configurations are the top performers. The ε-hBOA-Base configuration maintains good overall performance throughout the run, and generally ends the run in second place. It seems that the ε-hBOA-Archive and ε-hBOA-Hybrid configurations are comparable to one another by the end of the run and the ε-hBOA-RTR configuration is the worst performing configuration overall. It is interesting to note that the ε-NSGAII performs well in terms of both the ε-indicator and hypervolume metrics. The performance of the ε-NSGAII in terms of the ε-indicator and hypervolume metrics is likely due to early rapid approximation of the 25 well test case reference set as was shown by Kollat and Reed [73].

Table 4.2 displays early run (NFE = 250,000) and end-of-run (NFE = 2,000,000) metric results for the 58 point test case. Statistically significant rankings are displayed similarly to Table 4.1. Early in the run, the ε-hBOA-Base configuration is the top performer while the ε-hBOA-Archive configuration trails in second place. The computational scaling limitations highlighted by Kollat and Reed [73] have caused the ε-NSGAII to be the worst performer overall. At the end of the run, the ε-hBOA-Static configuration is again the top performer, similarly to the 25 well test case. Although the ε-hBOA-Static configuration has the drawback of requiring a careful specification of the population size and exhibits an early run lag in terms of finding reference set solutions, it ultimately produces the best performance over all other configurations. Again, only top ranked configurations with statistically significant performance differences are highlighted in bold in Table 4.2. The reader should also note that there are many cases for the ε-indicator and hypervolume metrics where multiple configurations share statistically similar performance. To further elucidate performance differences, the next section provides a detailed analysis of search performance dynamics.
Table 4.1. NFE = 25,000 and end-of-run mean and (standard deviation) metric results for the 25 well test case. Metric values in bold text indicate statistical superiority at a 95-percent confidence level.

<table>
<thead>
<tr>
<th>Method</th>
<th>Conv. $\times 10^3$</th>
<th>$\varepsilon$-Perf.</th>
<th>$\varepsilon$-Ind.</th>
<th>Hyper. $\times 10^{-5}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>NFE = 25,000</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\varepsilon$-NSGAII</td>
<td>14.210 (2.876)</td>
<td>0.127 (0.025)</td>
<td><strong>3.965 (0.313)</strong></td>
<td>NA</td>
</tr>
<tr>
<td>$\varepsilon$-hBOA-Base</td>
<td><strong>2.751 (0.507)</strong></td>
<td>0.420 (0.015)</td>
<td>4.297 (1.155)</td>
<td>NA</td>
</tr>
<tr>
<td>$\varepsilon$-hBOA-Archive</td>
<td>2.911 (0.665)</td>
<td><strong>0.427 (0.018)</strong></td>
<td>4.005 (0.857)</td>
<td>NA</td>
</tr>
<tr>
<td>$\varepsilon$-hBOA-RTR</td>
<td>8.880 (1.807)</td>
<td>0.205 (0.023)</td>
<td>4.518 (0.975)</td>
<td>NA</td>
</tr>
<tr>
<td>$\varepsilon$-hBOA-Static</td>
<td>36.910 (1.923)</td>
<td>0.008 (0.002)</td>
<td>6.668 (0.468)</td>
<td>NA</td>
</tr>
<tr>
<td>$\varepsilon$-hBOA-Hybrid</td>
<td>15.610 (2.940)</td>
<td>0.106 (0.022)</td>
<td>4.003 (0.344)</td>
<td>NA</td>
</tr>
<tr>
<td>NFE = 200,000</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\varepsilon$-NSGAII</td>
<td>1.560 (0.270)</td>
<td>0.688 (0.015)</td>
<td>3.494 (0.308)</td>
<td>5.085 (0.619)</td>
</tr>
<tr>
<td>$\varepsilon$-hBOA-Base</td>
<td>0.866 (0.147)</td>
<td>0.698 (0.015)</td>
<td>3.460 (0.571)</td>
<td>5.381 (0.444)</td>
</tr>
<tr>
<td>$\varepsilon$-hBOA-Archive</td>
<td>0.994 (0.274)</td>
<td>0.683 (0.020)</td>
<td>3.539 (0.657)</td>
<td>5.658 (0.352)</td>
</tr>
<tr>
<td>$\varepsilon$-hBOA-RTR</td>
<td>1.740 (0.304)</td>
<td>0.557 (0.021)</td>
<td>3.478 (0.256)</td>
<td>5.709 (0.392)</td>
</tr>
<tr>
<td>$\varepsilon$-hBOA-Static</td>
<td><strong>0.787 (0.154)</strong></td>
<td><strong>0.752 (0.011)</strong></td>
<td><strong>3.133 (0.317)</strong></td>
<td>5.176 (0.617)</td>
</tr>
<tr>
<td>$\varepsilon$-hBOA-Hybrid</td>
<td>1.140 (0.253)</td>
<td>0.703 (0.018)</td>
<td>3.465 (0.174)</td>
<td>5.511 (0.337)</td>
</tr>
</tbody>
</table>
Table 4.2. NFE = 250,000 and end-of-run mean and (standard deviation) metric results for the 28 point test case. Metric values in bold text indicate statistical superiority at a 95-percent confidence level.

<table>
<thead>
<tr>
<th>NFE=250,000</th>
<th>Conv.×10^3</th>
<th>ε-Perf.</th>
<th>ε-Ind.</th>
<th>Hyper.×10^{-6}</th>
</tr>
</thead>
<tbody>
<tr>
<td>ε-NSGAII</td>
<td>16.028 (1.403)</td>
<td>0.005 (0.002)</td>
<td>3.796 (0.240)</td>
<td>NA</td>
</tr>
<tr>
<td>ε-hBOA-Base</td>
<td><strong>7.574 (1.393)</strong></td>
<td><strong>0.066 (0.014)</strong></td>
<td>3.713 (0.372)</td>
<td>NA</td>
</tr>
<tr>
<td>ε-hBOA-Archive</td>
<td>7.707 (1.185)</td>
<td>0.061 (0.015)</td>
<td><strong>3.666 (0.305)</strong></td>
<td>NA</td>
</tr>
<tr>
<td>ε-hBOA-RTR</td>
<td>12.959 (1.703)</td>
<td>0.018 (0.006)</td>
<td>3.783 (0.202)</td>
<td>NA</td>
</tr>
<tr>
<td>ε-hBOA-Static</td>
<td>11.945 (0.778)</td>
<td>0.005 (0.001)</td>
<td>3.908 (0.258)</td>
<td>NA</td>
</tr>
<tr>
<td>ε-hBOA-Hybrid</td>
<td>14.952 (1.532)</td>
<td>0.007 (0.004)</td>
<td>3.787 (0.272)</td>
<td>NA</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>NFE=2,000,000</th>
<th>Conv.×10^4</th>
<th>ε-Perf.</th>
<th>ε-Ind.</th>
<th>Hyper.×10^{-6}</th>
</tr>
</thead>
<tbody>
<tr>
<td>ε-NSGAII</td>
<td>8.029 (1.001)</td>
<td>0.086 (0.014)</td>
<td>2.903 (0.170)</td>
<td>2.014 (0.296)</td>
</tr>
<tr>
<td>ε-hBOA-Base</td>
<td>4.897 (1.075)</td>
<td>0.243 (0.040)</td>
<td>3.252 (0.415)</td>
<td>2.096 (0.323)</td>
</tr>
<tr>
<td>ε-hBOA-Archive</td>
<td>5.688 (1.015)</td>
<td>0.208 (0.045)</td>
<td>3.136 (0.288)</td>
<td>2.139 (0.265)</td>
</tr>
<tr>
<td>ε-hBOA-RTR</td>
<td>10.413 (1.964)</td>
<td>0.074 (0.021)</td>
<td>3.365 (0.222)</td>
<td>2.492 (0.326)</td>
</tr>
<tr>
<td>ε-hBOA-Static</td>
<td><strong>3.187 (0.388)</strong></td>
<td><strong>0.414 (0.023)</strong></td>
<td>2.927 (0.223)</td>
<td><strong>1.830 (0.388)</strong></td>
</tr>
<tr>
<td>ε-hBOA-Hybrid</td>
<td>6.375 (1.478)</td>
<td>0.195 (0.049)</td>
<td>2.908 (0.192)</td>
<td>1.992 (0.324)</td>
</tr>
</tbody>
</table>
4.6.2 Search Efficiency and Reliability

Figure 4.5 plots $\varepsilon$-performance success rates for each algorithm configuration for the 25 well and 58 point test cases. The $\varepsilon$-performance success rate is defined as the percentage of random seed trials that exceed some threshold of metric performance for a given NFE to provide insight into algorithm dynamics. Cases with nearly vertical success rate cumulative distribution curves represent highly reliable performance where all of a configuration’s random seed trials exceeded the threshold of performance at approximately the same NFE. Likewise, shallow slopes for success rate curves indicate low search reliability. For the 25 well test case, $\varepsilon$-performance success rates are plotted at thresholds of 0.25, 0.5, and 0.7 (corresponding with quantification of 25, 50, and 70-percent of the reference set) in Figures 4.5A through 4.5C respectively. Figure 4.5A indicates that all configurations are fairly reliable at quantifying 25-percent of the 25 well test case’s reference set. However, it appears that the $\varepsilon$-hBOA-Base and $\varepsilon$-hBOA-Archive configurations exceed this threshold fastest, while the $\varepsilon$-NSGAII and $\varepsilon$-hBOA-Static configuration are the slowest at meeting this goal. A shift in relative configuration success rates can be noted in Figure 4.5B where the $\varepsilon$-hBOA-Static configuration moves from last place to third place, and $\varepsilon$-hBOA-Hybrid moves from 5th place to 4th place. It should be noted that the $\varepsilon$-NSGAII and $\varepsilon$-hBOA-Hybrid versions are equivalent in Figure 4.5A since $\varepsilon$-NSGAII is performing the search early in the $\varepsilon$-hBOA-Hybrid configuration’s run. In Figure 4.5C, a pronounced shift in success rate is exhibited where the $\varepsilon$-hBOA-Static configuration moves to first place and all other configurations fail to find the reference set to an accuracy of 70-percent in 200,000 evaluations.

For the 58 point test case, the success rate plots shown in Figures 4.5D through 4.5F indicate that the $\varepsilon$-hBOA-Static configuration is again the fastest and most reliable for all $\varepsilon$-performance thresholds of 0.1, 0.2, and 0.3. In addition, as the success rate threshold becomes more rigorous, other configurations fail to quantify the reference set at increased accuracy. For example, the $\varepsilon$-NSGAII and $\varepsilon$-hBOA-RTR configurations completely fail to find 20-percent of the reference set, while all but the $\varepsilon$-hBOA-Static configurations fail to find anything above 30-percent of the reference set in 2-million design evaluations.
**Figure 4.5.** Epsilon-performance success rates plotted versus function evaluations (NFE) at specified thresholds for both the 25 well and 58 point test cases.

### 4.6.3 Balancing Efficiency and Reliability

Figure 4.6 shows the dynamic performances of three configurations: $\varepsilon$-NSGAII, $\varepsilon$-hBOA-Base, and $\varepsilon$-hBOA-Static. The choice of these configurations was based in part on their overall performance and because they represent three fundamentally different algorithms. Figure 4.6 displays runtime dynamics for the convergence and $\varepsilon$-performance metrics on both the 25 well and 58 point test cases. Dynamic performance is captured by plotting the metric attained by each seed as a function of the number of designs evaluated for all 50 random seed trials of the three algorithms.

For the 25 well test case (Figures 4.6A and 4.6B), it is clear that the final effectiveness of all configurations is somewhat comparable, which is expected given that the problem’s size is modest. Kollat and Reed [109] have already demonstrated that the $\varepsilon$-NSGAII can successfully solve this test case. This is an interesting lower bound problem complexity where the $\varepsilon$-NSGAII has the maximal chance of
exceeding the $\varepsilon$-hBOA configurations’ performances. The $\varepsilon$-NSGAII’s search proceeds very quickly initially (i.e., NFE = 10,000), but is rapidly overtaken by the $\varepsilon$-hBOA-Base configuration. The $\varepsilon$-hBOA-Base exhibits the fastest initial performance of the three algorithms compared. The $\varepsilon$-hBOA-Static configuration is slow to start, but ultimately achieves the best performance of the three configurations after an initial lag period for evolutionary search.

For the 58 point test case (Figures 4.6C and 4.6D), the difference in final performance of each configuration is highly pronounced. The $\varepsilon$-NSGAII does not scale well for the 58 point test case. While the $\varepsilon$-hBOA-Base configuration provides a very rapid initial solution approximation, the $\varepsilon$-hBOA-Static configuration quickly overtakes the performance of the $\varepsilon$-hBOA-Base configuration. It should be noted that the $\varepsilon$-hBOA-Base and the $\varepsilon$-hBOA-Static configurations represent a tradeoff between efficiency and the ultimate search effectiveness. The $\varepsilon$-hBOA-Base has the strong advantage of eliminating the need to specify any search parameters while the $\varepsilon$-hBOA-Static provides highly reliable search. In general, for users with severe computational constraints, the $\varepsilon$-hBOA-Base would be the superior choice for attaining rapid approximations while requiring minimal specification of search parameters. If users can use high performance computing or other means for reducing their computational constraints, use of the $\varepsilon$-hBOA-Static configuration would be beneficial. As was done in this work, the two configurations could be used in tandem by using the $\varepsilon$-hBOA-Base’s archive size from a preliminary run to determine an effective population size for the $\varepsilon$-hBOA-Static configuration.

4.6.4 Understanding Convergence Versus Diversity

In order to further assess performance differences between the $\varepsilon$-NSGAII, $\varepsilon$-hBOA-Base, and $\varepsilon$-hBOA-Static configurations, the best random seed trial from each configuration (based on $\varepsilon$-performance end-of-run metric values) was used to generate a Pareto approximation for the 58 point test case containing 15,355 solutions. In total, 8-percent of this set was contributed by all three configurations, and 32.4-percent of the set was contributed by both configurations of the $\varepsilon$-hBOA. However, the $\varepsilon$-hBOA-Static configuration contributed 37.4-percent uniquely to the set (meaning that no other configuration found these particu-
Figure 4.6. Convergence and $\varepsilon$-performance dynamics for the $\varepsilon$-NSGAII, $\varepsilon$-hBOA-Base, and $\varepsilon$-hBOA-Static configurations. Plots show each of the 50 random seed trials for each configuration plotted as metric value versus function evaluations (NFE).

lar solutions). This represents nearly four times the unique contribution of the $\varepsilon$-hBOA-Base configuration (10.5-percent), and eight times the unique contribution of the $\varepsilon$-NSGAII (4.2-percent). In all, the $\varepsilon$-hBOA-Static configuration would have found 82-percent of this set.

Figure 4.7 shows the approximate Pareto set generated from the best trials of the $\varepsilon$-NSGAII, $\varepsilon$-hBOA-Base, and $\varepsilon$-hBOA-Static configurations for the 58 point test case. Cost, Error, and Uncert are plotted on the X, Y, and Z-axes. The plot shown in Figure 4.7 differs from the plotting technique of Figure 4.3 in that the Mass objective is portrayed by the orientation of the solution cones (where cones pointing up mean high Mass error and cones pointing down mean low Mass error).
This is because color is reserved in this plot to reflect which algorithm generated the solution (blue represents $\varepsilon$-hBOA-Static, green represents the $\varepsilon$-hBOA-Base, and red represents the $\varepsilon$-NSGAII). Subplots B, C, and D in Figure 4.7 show detailed views of the regions highlighted in subplot A. These detailed views show the predominant contribution of the $\varepsilon$-hBOA-Static configuration indicated by the blue points. In general, structural differences between the $\varepsilon$-hBOA-Base, and $\varepsilon$-hBOA-Static solution contributions seem to be limited to the shear quantity of solutions discovered. In other words, the $\varepsilon$-hBOA-Static configuration finds many more solutions, ultimately filling out portions of the objective space more densely. An interesting feature of these detailed views is that they show how the $\varepsilon$-NSGAII contributes uniquely to the set. Figure 4.7B shows that the $\varepsilon$-NSGAII finds several solutions at the farthest extent of the set corresponding to very high Cost where most of the potential monitoring points are sampled. In addition, there are regions of unique contribution of the $\varepsilon$-NSGAII visible in Figures 4.7C and 4.7D. The unique contributions of the $\varepsilon$-NSGAII are all characterized by their sparsity. This indicates that the $\varepsilon$-NSGAII is better at finding solutions in regions of the set with low solution density. This explains the highly ranked performance of the $\varepsilon$-NSGAII in terms of the $\varepsilon$-indicator and hypervolume metrics in Tables 4.1 and 4.2, because these metrics are sensitive to the geometric distribution of solutions across the objective space.

4.7 Discussion

Results indicate that the $\varepsilon$-hBOA is more effective than the $\varepsilon$-NSGAII at optimizing both the 25 well and 58 point LTGM test cases. Performance differences between the $\varepsilon$-NSGAII and $\varepsilon$-hBOA are notably more pronounced on the larger 58 point LTGM test case. Relative performance differences between the various configurations of the $\varepsilon$-hBOA are informative with regard to the factors that affect the quality of Bayesian network models. The relatively poor performance of the $\varepsilon$-hBOA-RTR configuration can be attributed to a lack of selection pressure, ultimately slowing the progress of the algorithm. Restricted tournament recombination randomly selects solutions from the parent and child populations and places them in competition within binary tournaments to determine which solution pro-
Figure 4.7. Pareto approximation set generated by combining the best trial runs of each the $\varepsilon$-NSGAII, $\varepsilon$-hBOA-Base, and $\varepsilon$-hBOA-Static configurations. Cost, Error, and Uncert are plotted on the $X$, $Y$, and $Z$-axes while the Mass objective is portrayed by the orientation of the solution cone (up = high Mass, down = low Mass). Red solutions were contribution by the $\varepsilon$-NSGAII, green solutions by the $\varepsilon$-hBOA-Base, and blue solutions by the $\varepsilon$-hBOA-Static configuration.

ceeds to the next generation. Alternatively, crowded binary tournament selection utilized by the $\varepsilon$-NSGAI first ranks both parents and children according to their relative position with respect to non-domination. The subsequent population is then filled by allowing the highest ranked solutions to proceed to the next generation. This recombination method places more emphasis on highly fit solutions and hence increases the selection pressure of the algorithm.

The changes in the relative performances of each configuration early versus late in the runs (refer to Tables 4.1 and 4.2 as well as Figures 4.5 and 4.6) can be explained in terms of population size effects. The $\varepsilon$-hBOA-Base configuration
uses an arbitrarily small initial population size, which allows it to rapidly approximate the Pareto set. However, the initial small population provides a reduced statistical sample for optimal model building [116] and as a result, the long-term performance of this configuration suffers. Alternatively, the $\varepsilon$-hBOA-Static configuration uses a static population size that is initially very large, which initially slows the algorithm’s evolutionary exploration. In the long term however, the $\varepsilon$-hBOA-Static configuration yields more accurate models for decision interdependencies and enhanced search.

Performance differences between the $\varepsilon$-NSGAII and the $\varepsilon$-hBOA configurations can largely be attributed to how solution density impacts $\varepsilon$-NSGAII’s SBX crossover operator versus $\varepsilon$-hBOA’s Bayesian network model building. Figure 4.7 showed that the $\varepsilon$-NSGAII is better at finding solutions in sparsely populated regions of the objective space. The SBX crossover operator utilized by the $\varepsilon$-NSGAII is a parent-centric mating approach whereby child solutions have a higher probability of being generated close to their parents [188]. When the $\varepsilon$-NSGAII finds a solution in these sparsely populated regions, the SBX crossover operator emphasizes a relatively local search (i.e., small changes in the decisions), ultimately allowing it to search these regions of the space more thoroughly. However, Figure 4.7 also showed that the $\varepsilon$-hBOA was more effective at generating solutions in densely populated regions. There are clearly more ways to choose 30 of the 58 sampling points than 56 of the 58 points. These combinatorial differences affect solution density throughout the search space. This ultimately diminishes the ability of the $\varepsilon$-hBOA to model sparse areas of the search space. Since it is more difficult for the $\varepsilon$-hBOA to learn the structure of these sparse regions, the net result is that it is less likely that Pareto-optimal solutions will be simulated in these regions.

It was shown that the $\varepsilon$-hBOA-Base and $\varepsilon$-hBOA-Static versions were the top-performing search schemes. In choosing between these two configurations, the computational resources and expertise of the user must be considered. Overall, the $\varepsilon$-hBOA-Static configuration had the best long-term performance, but required careful specification of the population size and significant run-time to achieve high quality model building. In contrast, the $\varepsilon$-hBOA-Base configuration is superior for rapid solution set approximations and it eliminates the need to specify the popula-
tionsize parameter. If appropriate computational resources are available, these two configurations could be used in tandem. If no prior knowledge is available regarding the Pareto set size (which is more typical), the $\varepsilon$-hBOA-Base configuration could be run to determine the optimal population size for the $\varepsilon$-hBOA-Static configuration. Once an approximation of the Pareto optimal solution set size is determined, the population size of the $\varepsilon$-hBOA-Static configuration can be specified based on a one-to-one ratio relative to the Pareto set size. In addition, since the reliability of the $\varepsilon$-hBOA-Static configuration is high, random seed trial analysis can be reduced or eliminated, dramatically reducing the computational burden of using the $\varepsilon$-hBOA-Static configuration.

The success of the $\varepsilon$-hBOA relative to the $\varepsilon$-NSGAII implies that there are strongly inter-related decisions within the LTGM design problem and that the $\varepsilon$-hBOA can exploit these dependencies through Bayesian network model building. The physical contaminant plume structure implicit to the LTGM design problem likely does present some degree of hierarchical difficulty, also made apparent by the success of the $\varepsilon$-hBOA relative to the $\varepsilon$-NSGAII. The 25 well and 58 point test cases provide a means for testing if hierarchical problem difficulty is impacting the performance of the search algorithms. The $\varepsilon$-NSGAII has been well documented [72] in solving the smaller 25 well test case efficiently and reliably. If the sampling decisions were statistically independent as assumed by the $\varepsilon$-NSGAII’s mating and mutation operators, then the $\varepsilon$-NSGAII should have been able to attain superior performance for the smaller test case. Figure 4.6 demonstrates that this is not the case and that even for the small 25 well test case, the $\varepsilon$-hBOA demonstrated superior performance. Moreover, the large test case success rate plots and runtime dynamics for the $\varepsilon$-NSGAII indicate that the algorithm is not reliable and that the best case performance for the algorithm poorly represents its expected performance.

The region where the $\varepsilon$-NSGAII has the most distinct performance advantage is very near the solution where all points are sampled. This region is actually the least challenging to predict by the decision maker and likely of limited interest. The compromise regions near the center of the reference set shown in Figure 4.7 were best quantified by the $\varepsilon$-hBOA and are generally of most interest for exploring the impacts of cost-savings on the remaining objectives. The lower bound cost portion
of the tradeoffs (i.e., regions of high errors in Figure 4.7) is also an interesting region in terms of search difficulty. This portion of the tradeoff surface is at the feasibility boundary for the constrained formulation of this problem. Somewhat surprisingly, this region’s large objective ranges and sparseness do not impair the \( \varepsilon \)-hBOA as illustrated in Figure 4.7. The algorithm is able to find a superior representation of this portion of the objective space relative to the \( \varepsilon \)-NSGAII. This result implies that the overall success or failure of the \( \varepsilon \)-hBOA is more heavily influenced by statistical sampling than by the problem’s constraint. The feasibility boundary still has a large number of potential solution combinations that can be sampled and modeled (i.e., designs consisting of five sampling points drawn from the full set of 58 potential locations).

### 4.8 Conclusions

This chapter contributes a new MOEA termed the Epsilon-dominance Hierarchical Bayesian Optimization Algorithm (\( \varepsilon \)-hBOA), for solving large, multiobjective groundwater monitoring design problems. The new \( \varepsilon \)-hBOA was tested on a 25 well LTGM design test case (a relatively small test case with around 33-million possible designs) and a larger 58 point LTGM design test case containing over \( 2.88 \times 10^{17} \) possible designs. Both test cases were optimized for four design objectives - Cost, Error, Uncert, and Mass. The Pareto-optimal set for the 25 well test case contained 2,472 designs and the Pareto-approximation for the 58 point test case contained 22,333 designs. The \( \varepsilon \)-NSGAII, a top performing MOEA was tested relative to five configurations of the \( \varepsilon \)-hBOA. The \( \varepsilon \)-hBOA-Base configuration utilized dynamic population sizing based on a 25-percent \( \varepsilon \)-nondominance archive injection rate similar to the \( \varepsilon \)-NSGAII. The \( \varepsilon \)-hBOA-Archive configuration included the current \( \varepsilon \)-nondominance archive in the model building following each generation. The \( \varepsilon \)-hBOA-RTR configuration utilized restricted tournament recombination rather than the crowded binary tournament selection utilized by the \( \varepsilon \)-NSGAII. The \( \varepsilon \)-hBOA-Static configuration replaced dynamic population sizing with a static population size that was specified \textit{a priori}. The \( \varepsilon \)-hBOA-Hybrid configuration utilized the low cost search and dynamic population sizing of \( \varepsilon \)-NSGAII initially, and subsequently switched to a static population size based on this initial search. A
comprehensive, metrics based analysis was performed to determine each the effectiveness, efficiency, and reliability of each $\varepsilon$-hBOA configuration relative to the $\varepsilon$-NSGAII.

Results demonstrate the scaling limitations of the $\varepsilon$-NSGAII relative to the $\varepsilon$-hBOA. The $\varepsilon$-hBOA uses Bayesian network model building to learn the complex interdependencies which exist within the LTGM design problem. In addition, because of the $\varepsilon$-hBOA's success relative to the $\varepsilon$-NSGAII, the LTGM problem is believed to exhibit some degree of hierarchical difficulty due to the physical structure of the contaminant plume. It was shown that the $\varepsilon$-hBOA-Base and $\varepsilon$-hBOA-Static versions were the top-performing configurations of the $\varepsilon$-hBOA. Overall, the $\varepsilon$-hBOA-Static configuration performed best, but required careful specification of the population size, and significant run-time to achieve high quality solutions. In contrast, the $\varepsilon$-hBOA-Base configuration was superior for rapid approximations and it eliminated the need to specify an optimal population size a priori since it automatically adapts its size based on search progress. If computational resources are limited, the $\varepsilon$-hBOA-Base configuration would be an appropriate choice to obtain rapid approximation to the Pareto set. However, the $\varepsilon$-hBOA-Base configuration could also be used to determine the optimal population size needed by the $\varepsilon$-hBOA-Static configuration and a subsequent run can be conducted using the $\varepsilon$-hBOA-Static configuration to achieve the highest quality solution possible. Since the reliability of the $\varepsilon$-hBOA-Static configuration is high, random seed trial analysis becomes less important, ultimately reducing the computational burden of this approach.

While this chapter demonstrated that the $\varepsilon$-hBOA is an effective improvement over current MOEAs at solving LTGM design applications, the monitoring design test cases examined represent a lower bound problem complexity because they represent only a snapshot in time. The size and complexity of the search space rapidly increases when spatiotemporal monitoring decisions are considered. In addition, monitoring complexities (e.g., the existence of multiple contaminants) also adds to the difficulty of the monitoring design problem. Work is currently being done to develop a highly reliable, parallel extension of the $\varepsilon$-hBOA capable of running on massively parallel computing resources to further address the computational scaling issues associated with groundwater monitoring design.
5.1 Abstract

This chapter presents a framework for Visually Interactive Decision-making and Design using Evolutionary Multi-objective Optimization (VIDEO). The VIDEO framework allows users to visually navigate large multi-objective solution sets while aiding decision makers in identifying one or more optimal designs. Specifically, the interactive visualization framework is intended to provide an innovative exploration tool for high-order Pareto-optimal solution sets (i.e., solution sets for three or more objectives). The framework is demonstrated for a long-term groundwater monitoring (LTGM) application in which users can explore and visualize tradeoffs for up to four design objectives, simultaneously. Interactive functionality within the framework allows the user to select solutions within the objective space and visualize the corresponding monitoring plan’s performance in the design space. This functionality provides the user with a holistic picture of the information provided by a particular solution, ultimately allowing them to make a more informed decision. In addition, the ease with which the framework allows users to navigate and compare solutions as well as design tradeoffs leads to a time efficient analysis, even when there are thousands of potential solutions. The work presented in this
chapter has been published in the journal *Environmental Modelling and Software* [72].

### 5.2 Introduction

This chapter presents a framework for Visually Interactive Decision-making and Design using Evolutionary Multiobjective Optimization (VIDEO). The VIDEO framework allows users to visually navigate large multiobjective solution sets while identifying one or more optimal designs. Development of the VIDEO framework was motivated by the increasing emphasis of environmental systems research on multi-objective methods [75–78]. Recent innovations in multi-objective evolutionary algorithms (MOEAs) have served to catalyze the development of a broad suite of multiobjective design and decision support methodologies within the environmental and water resources literature [46, 63, 83–91]. MOEA’s population-based search allows users to find entire tradeoff (or Pareto) surfaces using a single algorithm run for large, complex problem spaces [92–94].

In general, optimality for multiple objectives is defined by the set of solutions that are globally (or locally) better than all other solutions in at least one objective and are termed Pareto-optimal (or non-dominated) solutions [81]. High-order Pareto optimization problems as defined by Reed and Minsker [63] have three or more objectives. They have also been termed “many-objective” problems within the operations research (OR) literature [79]. There is a growing body of literature exploring the challenges and benefits associated with solving high-order Pareto optimization problems [63, 79, 89, 152, 153, 171]. Please note that in this chapter, the general term “Pareto optimal” is used instead of “nondominated” when describing approximate multiobjective solution sets because the term is generally appropriate for the best known solution set for an application.

Although our ability to solve high-order Pareto optimization problems using techniques such as MOEAs reflects more recent computational advances, the systems analysis literature from the past 30 years has recognized that design problems often have several conflicting objectives, especially in the environmental arena [77]. As highlighted by [77] (p. 17-18), modern systems research has its origins dating back to Wiener’s 1961 text Cybernetics [189] which successfully forecasted a myr-
iad of technological advances that have shaped human-computer interaction as well as design/decision making processes. Classical multiobjective decision making methodologies [75, 138] have generally focused on transformation methods that commensurate multiple objectives into new formulations that are amendable to solution using single objective optimization algorithms. MOEAs represent a break from traditional multi-criteria operations research where it is no longer necessary to transform or commensurate problem structures into single objective formulations and instead new decision support tools can focus on the structure and content of Pareto surfaces.

This chapter seeks to: (i) affirm the need and value of combining interactive visualization with high-order Pareto optimization for improved a posteriori decision making [see Section 5.3.1] and (ii) provide a specific demonstration of the VIDEO framework within an illustrative long-term groundwater monitoring (LTGM) design application [see Section 5.3.2]. This work builds on Reed and Minsker [63] by contributing a visualization framework tailored specifically to many-objective LTGM applications. LTGM design has long been recognized to have “many objectives” [15] and is an excellent example application for showing that visualization combined with high-order multiobjective solution sets can facilitate discovery and negotiation in the design and decision-making process [63]. Use of the terms discovery and negotiation is motivated by the potential of high-order multiobjective solution sets to generate alternatives that capture a broad suite of system behaviors relevant to both modeled and unmodeled objectives [82], helping decision makers to discover system dependencies and/or tradeoffs and exploit this information in the negotiated selection of a solution.

In the remainder of this chapter, Section 5.3 provides a more detailed discussion of a posteriori decision making, the VIDEO framework’s components, and the LTGM application being used to demonstrate the framework. Section 5.4 provides illustrative results for the LTGM case study demonstrating how the VIDEO framework facilitates exploration of tradeoffs and negotiated focus on specific solutions. Sections 5.5 and 5.6 discuss the implications of using interactive visualization to facilitate a posteriori decision making.
5.3 Methodology

5.3.1 A Priori and A Posteriori Approaches

A priori methods [78] seek to model decision maker preference before searching for designs/decisions. A classic example of an a priori method is the “normative” decision making methodology developed by Keeney and Raifa [76, 190] termed multi-attribute utility analysis (MAUA). MAUA requires extensive surveys of decision maker (DM) preferences prior to searching for potential system designs. In MAUA, DM surveys must be designed carefully to assure methodological assumptions are satisfied [i.e., preferential and utility independence (for more details see [76, 138]). MAUA is an example of a transformative method where the original decision objectives are analytically represented within a utility function that can be optimized using traditional single objective algorithms. In MAUA, all search and decision making is then in reference to utility. A priori methods such as MAUA have been criticized because they do not condition DM preference on potential alternatives, they suffer from DM contradictions (or intransitivity), and utility representations of preference are non-unique for groups (i.e., Arrow’s Paradox [191]).

In contrast to a priori methods, the VIDEO framework represents an a posteriori decision tool where DM preferences for alternatives are expressed after non-dominated or Pareto optimal alternatives have been identified. In this process, all solutions are initially assumed to have equal preference during the search process. DM preferences are then expressed in the exploration tradeoffs and selection of design solutions. A posteriori decision tools have been criticized in the OR literature due to (i) the mathematical complexity of finding tradeoff solutions and (ii) the contention that large solution sets tend to overwhelm/confuse DMs while providing limited insights into their design preferences [77, 78, 146].

In reference to the criticisms of a posteriori decision making methods, MOEAs have significantly enhanced our ability to search for and quantify large multi-objective solution sets in the environmental area [89, 91, 109]. Moreover, the second criticism’s implication that large solution sets overwhelm DMs assumes that system expertise and visualizations have limited value. Multi-objective engineering and environmental applications represent complex systems that can be meaningfully visualized in space and/or time. Exploring tradeoffs and visualizing system per-
formance can be useful in both the objective space and design/system space. As an example, a pollution remediation system can have a suite of objectives (minimize cost, maximize reliability, maximize resiliency, minimize redundancy, etc.) each of which shape the potential real-world system’s spatiotemporal performance. There is an increasing number of studies demonstrating that visualization combined with optimization can promote design innovations and provide DMs with an improved understanding of system behaviors [63, 79, 142, 143, 147–149]. In the context of this prior work, the VIDEO framework contributes a highly interactive environment for exploring LTGM design tradeoffs for up to four objectives and detailed analysis of their consequences in the resultant design space.

5.3.2 VIDEO Framework Components

5.3.2.1 Overview

The VIDEO framework has been developed using the Python programming language. Python is a dynamic, object-oriented scripting language which integrates easily with other languages, and is virtually platform independent [192]. Additionally, Python has a large number of graphical user interface (GUI) frameworks, including Tkinter, PyQt, and WxPython. The VIDEO framework has been developed using Tkinter (Python’s standard GUI package) which is built on top of Tcl/Tk and is portable across Windows, Unix, and Mac platforms [193]. Aside from Python’s ease-of-use and GUI development capabilities, it allows easy integration with other, lower level (i.e., faster) languages such as C and C++. The VIDEO framework takes advantage of this capability by utilizing the Visualization ToolKit (VTK) [194] which is written in C++ for the visualization components of the framework, and a spatial interpolator (KT3D) [129] written in C. The Visualization ToolKit is an open-source software system developed for three dimensional visualization and image processing developed in C++ and wrapped with various other common programming languages (namely Java, Python, and Tcl/Tk). One of the main strengths of VTK is its fast visualization rendering and excellent interaction capabilities. KT3D has been utilized in the initial version of the VIDEO framework to aid in visualizing designs that require spatial interpolation using Kriging. KT3D has been coded in C and is called by Python to perform the re-
Figure 5.1. Snapshot of VIDEO decision making software. The interface is divided into an objective space window, a design space window, and a set of tools associated with each of these windows.

required spatial interpolation on-the-fly during interaction events such as when a user is selecting and comparing solutions. When Pareto-optimal solution sets are large with potentially thousands of solutions (as is the case with the LTGM design problem explored in this chapter), it is more efficient for gridded spatial datasets to be generated on-demand.

The VIDEO decision making software is divided into two primary components, an objective space window and a decision space window shown in Figure 5.1. The objective space window contains the visualization data associated with the design objectives of the problem, as well as tools that allow the user to easily manipulate how objective tradeoffs can be visualized. For example, the user can quickly change which objectives are displayed, how they are displayed (i.e., plotting axes and color representations) as well as change the display precision of the solutions,
and the plotting limits. The decision space component of the software displays the design space associated with the problem and is therefore more problem specific. The decision space component is linked with the objective space component, allowing the user to select specific solutions from within the objective space, and see what these solutions actually represent in terms of a real-world design. Tools are provided within the decision space component to allow the user to explore the implications associated with the tradeoffs between their design objectives. These components and tools are discussed in more detail in Section 5.3.2.3.

5.3.2.2 Generating Multiobjective Solution Sets

The VIDEO framework is an *a posteriori* decision support tool that has been designed with the intent of promoting exploration of multi-objective solution sets generated by MOEAs. However, any solution set ranging from low quality initial approximations generated using Monte Carlo analysis to true Pareto-optimal solutions sets generated through enumeration of every potential solution can be explored within the software. In this chapter, a 25-well LTGM application is used as the test case to demonstrate the VIDEO framework [73, 109]. The authors have successfully obtained high quality solution sets for the 25-well LTGM application using the Epsilon-Non-dominated Sorted Genetic Algorithm 2 (ε-NSGAII) [73, 91], which has been shown to perform as well or better than other top-performing MOEAs [109, 171]. These previous studies have motivated our interest in advancing *a posteriori* decision making using visualization in combination with large Pareto-optimal sets.

5.3.2.3 Interactive Visualization

*Objective Space Visualization.* The objective space component of the VIDEO framework currently allows the user to visualize up to four design objectives simultaneously. Up to three design objectives can be plotted on the spatial coordinate axes (X, Y, and Z), and the fourth objective can be plotted using color. Each potential solution is represented as a spherical glyph positioned at the appropriate coordinates and is colored based on the fourth objective. Future versions of the software may allow for visualizing additional objectives through the manipulation
of glyph size, shape, or orientation [148] (note the work presented in Chapters 4 and 7). Located immediately above the objective space window are a set of radio buttons that allow the user to change which of the objectives are plotted on each of the coordinate axes and which objective is represented using glyph color. These radio buttons are arranged in a grid fashion with each row of the grid representing the $X$, $Y$, $Z$, and color axis respectively and each column of the grid representing design objectives one through four. An additional “off” column is provided to allow the user to turn one or more plotting axes off. This GUI design allows the user to quickly and efficiently explore the wide range of plotting possibilities within the objective space, ultimately enhancing their understanding of the problem. For example, by specifying only two plotting axes and switching the others off, two objective subsets of the larger four objective problem can be plotted to quickly identify tradeoffs between objectives. Alternatively, glyph color can be utilized as a third objective while two other objectives are plotted on the spatial coordinates, possibly providing enhanced insight into the many relationships which may exist between the design objectives. Alternative plotting techniques are demonstrated in more detail for the four objective LTGM test case in Section 5.4.1.

**Objective Space Navigation.** The VIDEO framework provides the user with two means of focusing on areas of interest within the objective space. The first is a set of text boxes associated with each of the plotting axes ($X$, $Y$, $Z$, and color) which allow the user to numerically specify the plotting limits of each objective. This is useful if the user has problem specific knowledge that allows them to specify thresholds of interest for each of the design objectives. In addition, an interactive tool has been included that allows the user to dynamically resize a plotting box within the actual objective space window in order to focus on sub-regions of interest quickly and efficiently. This interactive tool can be instantiated by pressing “i” while the objective space rendering window is active, or by selecting “Interactive Limits Adjustment” under the “Objective Space Tools” menu.

**Objective Space Thinning.** Previous studies have emphasized the need for understanding objective precision requirements in multiobjective optimization [73, 110, 111]. For example, should the concentration estimates of a system be quantified in parts per trillion (ppt), or is parts per million (ppm) a sufficient level of precision? Design objective precision specification has been added into
the VIDEO framework so that high resolution Pareto-optimal solution sets can be thinned based on the DM’s precision requirements. The VIDEO framework uses ε-dominance [111] to reduce the precision of the Pareto-optimal solution set. In this approach, values reflecting the required precision for each design objective are specified by the DM. Non-domination sorting is then performed based on the reduced precision rather than the full precision, ultimately resulting in a reduced precision set. For a detailed discussion and analysis of ε-dominance applied to a LTGM problem and its implications for reducing computational demand, please refer to a recent study by Kollat and Reed [73]. Within the VIDEO framework, ε-dominance values can be specified for each objective, and the solutions displayed in the objective space will be updated to reflect the DM’s required precision.

**Interactive Solution Selection.** To facilitate comparison of Pareto-optimal solutions, the user is provided with the ability to manually pick solutions in the objective space window at the click of a mouse button. When the mouse pointer is navigated over the objective space visualization window, a left mouse button click will select the glyph which is underneath the mouse pointer. Once the user has selected a solution, it is highlighted by a turquoise blue bounding box and several events can then occur. First and foremost, the design associated with the selected solution is displayed in the decision space window. This in and of itself is a very useful way to visually correlate how regions of the objective space map to various types of designs. In addition, the user also has the ability to extensively probe the data associated with the selected solution. Users can “mark” a selected solution, doubling its glyph size and permanently associating a bounding box with the solution. Marked solutions are always displayed in the visualization window until they are unmarked by the user. This allows the user to track various solutions of interest throughout the interactive decision making process. Solutions which have been marked can subsequently be exported to a file, which can then be read back into the software at a later time to facilitate further comparison.

**Decision Space Probing.** The decision space component of the VIDEO framework is highly problem specific. However, the demonstration version of the framework utilizes an LTGM design problem where the objectives are to effectively monitor a contamination plume using a predetermined set of well locations while minimizing system Cost, Error, and Uncert. Specific details of the LTGM test
case are described in Section 5.3.3. The VIDEO framework represents the LTGM decision space by plotting the available well locations throughout the sampling domain. Each well location is represented by a transparent cylinder, and within each cylinder, the actual sampling locations available along the well’s vertical axis are marked. When a user selects a solution in the objective space, the wells associated with the selected solution are highlighted in red. If the decision space probing functionality is activated, the selected solution will be Kriged on-the-fly to produce maps of concentration estimates, estimation error, and estimation uncertainty throughout the sampling domain.

For the LTGM test case, Quantile Kriging (QK) was used to obtain contaminant estimates throughout the sampling domain by using the data at the sampled locations. Estimation error was calculated by comparing the estimates obtained when all sampling locations are utilized, to the data obtained for a particular Pareto-optimal solution in which only a subset of the data points are utilized. In addition, since Kriging is a minimum error variance estimator, the local uncertainty associated with each estimate throughout the sampling domain is also available.

Following Kriging, the user is provided with the ability to interactively move a probing plane throughout the sampling domain in order to explore the implications of the design. When using a plane probe, the user can choose an X-Y, X-Z, or Y-Z plane depending on the geometry of the decision space domain. This plane can then be moved throughout the domain either through the use of a slider bar (available in the decision space tool frame) or by manually moving the plane throughout the domain using the mouse. The plane will display the underlying data by using red to represent areas of highest concentration estimates, error, or uncertainty (depending on which is selected) and blue to represent areas of lowest concentration estimates, error, or uncertainty. Data associated with different objective space solutions can be directly compared using the plane probe by setting the plane position to some location in the design domain, and subsequently selecting and comparing various solutions in the objective space.

An additional probing option allows the user to plot and manipulate a colored iso-surface representing a constant data value throughout the domain. The iso-surface data value can be controlled using a slider bar, and the color of the iso-surface changes with changing data values based on the data range (i.e., red
represents high values and blue represents low values). While a probing plane provides a 2D slice of the decision space for all data values, an iso-surface provides a volumetric view of the decision space at a constant data value. One useful way in which to utilize the iso-surface functionality is to explore how the iso-surfaces change at a given data value for various objective space solutions.

The color mapping of both the plane probe and the iso-surface can be normalized to a minimum and maximum data value through the use of the “Clamp Lower” and “Clamp Upper” color limit check boxes. This ensures that the color scale used to represent the underlying data is normalized, allowing for efficient and accurate comparison between decisions. However, the solutions which should be used to clamp the color limits are highly problem specific, and often difficult to identify. To address this issue, the ability to manually clamp the color limits has been included within the VIDEO framework. Clamping the color limits ensures that when navigating the objective space, probed data in the decision space is always represented on the same color scale, making direct comparison between various solutions fast and accurate.

5.3.3 Test Case Description

The LTGM test case used to demonstrate the VIDEO framework represents a simulated perchloroethylene (PCE) contamination plume originating from an underground storage tank and has been described previously in Chapter 3, Section 3.3.1.

5.3.3.1 Design Objectives

Four design objectives were chosen for the LTGM test case, each of which was minimized. The design objectives included Cost, concentration estimation error (Error), concentration estimation uncertainty (Uncert), and mass estimation error (Mass). The Cost objective reflects the normalized cost of sampling the system. There are 25 well locations with one to three sampling locations at each well for a total of 47 available sampling locations. This implies that the Cost objective can range from zero (no locations sampled) to 47 (all locations sampled). The Error objective reflects the concentration estimation error between a Kriged map of the
plume utilizing all available sampling locations, and a Kriged map of the plume which utilizes a sub-set of the sampling locations. The Uncert objective reflects the uncertainty associated with the Kriged map of the contaminant plume by summing the estimation variances attained for each estimation location. The Mass objective reflects the error between the total mass of PCE estimated by Kriging the domain based on all available well sampling locations, and the estimated mass of PCE obtained by Kriging the domain based on a sub-set of well sampling locations. These objectives are defined identically to those used in Chapters 3 and 4.

5.3.3.2 Spatial Interpolation

PCE concentration estimates were obtained at unsampled locations throughout the sampling domain of the LTGM test case using Quantile Kriging (QK) described in Chapter 2, Section 2.2.1. A C translation of KT3D developed by the authors has been integrated into the VIDEO framework to produce on-the-fly spatial interpolations of the LTGM decision space.

5.3.3.3 Generating the Pareto-set

A very close approximation to the true Pareto-optimal solution set was generated for the LTGM test case using a Master-Slave parallelization of the $\varepsilon$-NSGAII [91]. A constraint was placed on the LTGM test case such that if a well sampling scheme contained too few sampling points to fully Krig the entire sampling domain (as defined by the Kriging parameters described in Section 5.3.3.2), the scheme was considered infeasible and was penalized by the algorithm similarly to that used in Chapter 3. The $\varepsilon$-NSGAII was run for 3.2 million function evaluations and a Pareto-optimal approximation set containing 2,570 solutions was generated. Previous studies and experience indicate that this approximation set very likely represents at least 90-percent of the true Pareto-optimal solution set [91]. The Pareto set generated by the $\varepsilon$-NSGAII contained solutions with Cost ranging from 7 to 47, Error ranging from 0 to 34.4, Uncert ranging from 1284 to 1564, and Mass ranging from -10.3 to 3.9 (negative values result because the mass error objective is scaled using a log transform). This Pareto-optimal solution set is provided for demonstration purposes within the initial version of the VIDEO framework and
is subsequently used to demonstrate the framework. The VIDEO framework has been designed to complement MOEAs or any other Pareto optimization algorithms.

5.4 VIDEO Case Study

5.4.1 Exploring and Understanding Objective Tradeoffs

Visualizing “many-objective” Pareto-optimal surfaces can be facilitated by representing three objectives using the traditional spatial coordinates, and additional objectives by changing representation characteristics. Representation of additional objectives can be accomplished in a number of ways including the use of color, shape, size, and orientation [148, 149]. For example, in the VIDEO framework, a fourth design objective is portrayed through the use of color, where blue indicates low objective values and red indicates high objective values. The VIDEO framework’s visualization of the four objective Pareto-optimal set is shown in Figure 5.2A, with the Cost, Error, and Uncert objectives plotted on the X, Y, and Z axes respectively, and the Mass objective plotted using color. However, as noted in Section 5.3.2.2, this can easily be changed through the manipulation of the plotting axes radio buttons. For example, starting with the configuration shown in Figure 5.2A, simply switching the Mass objective to the Z-axis, and the Uncert objective to the color axis [see Figure 5.2B] results in a configuration that provides new insights into the tradeoffs between the objectives because trends in color emphasize the Uncert objective.

It is interesting to note at this point that by solving for the high-order Pareto-optimal surface for four objectives, all of the sub-problems based on three objective, two objectives, or even a single objective are implicitly solved at the same time. Non-domination sorting can be performed on the high-order Pareto-surface in terms of fewer objective subsets. For example, the two-objective tradeoff between Cost and Error is shown in Figure 5.2C and is obtained by simply isolating solutions which dominate the full, high-order Pareto-set with respect to these two objectives, but are non-dominated with respect to one another. Figure 5.2D shows the Cost versus Error non-dominated subset highlighted with the remaining solutions composing the full four-objective Pareto set displayed using reduced marker
Figure 5.2. Examples of how a four-objective Pareto-optimal solution set can be viewed within the VIDEO framework. Panel A shows the full four objective Pareto set with Cost, Error, and Uncert, plotted on X, Y, and Z, and Mass plotted as color. Panel B reverses the Uncert and Mass objectives such that Mass is plotted on Z and Uncert plotted using color. Panel C shows the two objective subset Cost-Error and panels D and E show this subset in the context of the full four objective set. Panel F shows how a third objective (Cost) can be plotted using color to reveal additional decision maker insight.
sizes with transparency added. When manipulating the plotting axes within the VIDEO framework, the user can choose whether the whole high-order Pareto-set should be displayed [Figure 5.2D], or the non-dominated subset corresponding with the objectives selected for plotting [Figure 5.2C] by simply selecting a check box that tells the software to display the full set, with the current non-dominated set highlighted. It is also interesting to note that non-dominated subsets can be viewed in the context of the full four-objective space. For example, the Cost versus Error Pareto subset shown in Figures 5.2C and 5.2D can be viewed in the Cost versus Error versus Uncert versus Mass objective space [see Figure 5.2E]. This can aid in revealing tradeoff relationships between objective subsets, and the high-order Pareto-optimal space.

Other interesting ways of viewing the Pareto-optimal surface are illustrated in the following examples. The three-objective Pareto front associated with Cost, Error, and Uncert can be viewed by simply turning the Mass objective off. If viewing the three-dimensional surface associated with this three-objective Pareto-front is difficult, two of the objective can be plotted on the X and Y axis, and the third objective plotted using color. Any combination of objectives can be explored in this manner, and color can always be used to represent any objective. Alternatively, if Error and Uncert are plotted on the X and Y axes respectively and Cost is plotted using color [see Figure 5.2F], the relative distributions of solutions at each cost level can be viewed very easily.

5.4.2 Exploring and Understanding Designs

Before exploring the LTGM decision space, it is prudent to identify likely solutions with which the color limits should be clamped so that the color scale used to probe the decision space is normalized (see “Decision Space Probing” in Section 5.3.2.3). For the LTGM test case, it is anticipated that the highest Cost solution (i.e., the solution which samples from all available locations) will generally result in the lowest Error, Uncert, and Mass objective values. Thus, the lower color limit was clamped to the Cost = 47 solution. Likewise, the upper color limit was clamped to a Cost = 7 solution, which is anticipated to represent the highest Error, Uncert, and Mass objective values.
After normalizing the color limits, the best place to begin to understand the LTGM test case is to examine the highest cost solution because this solution will theoretically provide the most accurate picture of the contamination plume. The objective values associated with this solution are Cost = 47, Error = 0.0, Uncert = 1284, and Mass = 0.0. The highest cost solution is marked in Figure 5.3A (remember that the user can simply click on the solution within the objective space window to make a selection). The Kriged maps associated with the high cost solution are shown in Figures 5.3B through 5.3E. Figure 5.3B shows the quantile ranked estimates of the contamination plume using the plane probe at $Z = 85$ m. In this figure, the contaminant source at the southern boundary of the domain is clearly evident. At this elevation, there is also a region of high concentration toward the center of the sampling domain and a region of high concentration at the northwest boundary of the sampling domain. It is useful to note at this point that the plane probe can be moved throughout the sampling domain. Its current location ($Z = 85$ m) was chosen for illustrative purposes because it provides a good picture of the contaminant source, the general shape of the plume, and several other areas of high concentration that may be of interest to a decision maker. Figure 5.3C displays an iso-surface plotted at quantile concentration estimates of 0.70. Again, this data value was chosen somewhat arbitrarily, but to a large extent because it represents a relatively high quantile estimate (or spatial volumes containing high PCE concentrations). The estimation error is shown in Figure 5.3D at the same elevation used in 5.3B. The estimation error is quantified as the difference between the estimates obtained by sampling from all locations, and the estimate obtained using a subset of sampling locations. Since the highest cost solution (i.e., all locations are sampled) has been selected in this case, the error is zero throughout the domain. The estimation uncertainty is shown in Figure 5.3E at $Z = 85$ m. In this figure, the uncertainty is very low near the contamination source at the southern boundary of the domain (because of the preferential sampling that occurs in this region). In addition, the uncertainty is low around each sampling location because Kriging is an unbiased estimator (meaning that the estimates are true to the data values at the sampled locations). The area of highest uncertainty occurs at the north-east boundary of the domain where there are no well sampling locations available.
Figure 5.3. Example of solution comparison and decision space probing functionality within the VIDEO framework. Panels A and F show alternative views of the locations of the two selected solutions in the objective space. Panels B and G show the associated plume estimates at $Z = 85$ m. Panels C and H show an iso-surface plotted at concentration = 0.70. Panels D and I display the estimation error at $Z = 85$ m. Panels E and J show the spatial extent of estimation uncertainty at $Z = 85$ m.

In order to demonstrate the VIDEO framework’s ability to effectively combine visualization and interaction, a compromise solution is selected from the remaining 2569 Pareto-optimal designs. As shown in Figure 5.2B, plotting $\text{Uncert}$ as color and $\text{Mass}$ on the Z axis emphasizes $\text{Uncert}$ trends that may not otherwise be apparent. This plotting configuration was utilized in Figure 5.3E to select a compromise solution for comparison to the highest cost solution selected previously. The interactive plotting limits window was adjusted to display solutions between $\text{Cost} = 16$ to 32, $\text{Error} = 3.3$ to 17.7, and $\text{Mass} = -3.5$ to 3.9. In addition, the precision adjustment feature was used to “thin” the objective space, making it easier to locate a compromise solution. The precision settings for each objective were $\text{Cost} = 2.0$, $\text{Error} = 2.0$, $\text{Uncert} = 5.0$, and $\text{Mass} = 2.0$. These settings were selected to thin the space based on the nominal ranges of each design objective.
After selecting and comparing several solutions from the Pareto set subject to the above constraints, the solution shown in Figure 5.3F was selected for comparison because of its desirable characteristic of providing a fairly accurate picture of the plume but at a lower cost. The selected solution represents the objective values: 

\[
\begin{align*}
\text{Cost} &= 23, \\
\text{Error} &= 8.79, \\
\text{Uncert} &= 1354, \\
\text{Mass} &= 2.78.
\end{align*}
\]

Compared to the high cost solution examined previously, this represents a 60-percent decrease in Cost, but only a 25-percent increase in Error and Uncert and a 20-percent increase in Mass. The concentration estimates, estimation error, and associated uncertainty are shown in Figures 5.3G through 5.3J (similarly to Figures 5.3B through 5.3D). Figure 5.3G shows that this particular solution seems to capture the source of the contamination relatively well. However, the solution tends to overestimate the contaminant concentration at the northwest boundary of the plume and underestimate the concentration at the center-west boundary. Figure 5.3H provides insight into the volumetric data associated with this solution. In comparing Figure 5.3C with 5.3H, the iso-surfaces plotted at quantile concentration estimates of 0.7 differ widely between the two solutions. However, both solutions reveal the high contaminant concentration that occurs near the center of the domain, mostly because the well at this location is sampled in the second solution and the adjacent wells are not. The estimation error presented in Figure 5.3I further reveals this overestimation at the northwest boundary. In addition, high error in the southwest corner can be observed in Figure 5.3I. Returning to Figure 5.3G reveals that this solution does not capture what appears to be an area of high concentration just west of the source. The uncertainty map shown in Figure 5.3J shows a band-like region of very high uncertainty occurring at the northeast corner of the domain.

The VIDEO framework allows for interactive probing of the entire volume of the decision space. In the above example, relocation of the probing planes may reveal characteristics of the comparison solution (shown in Figure 5.3F) that make it more (or less) desirable as a possible compromise solution. Generally, the objective of selecting and comparing solutions would be to find a lower cost solution that accurately represents important aspects of the contamination plume while remaining within acceptable error and uncertainty bounds throughout the sampling domain. The VIDEO framework provides a solution marking feature that allows the user to mark (and unmark) solutions as they are navigating the
objective space. Solutions that have been marked are doubled in size and are bounded with a user specified color. In addition, solutions that have been marked are locked for visualization, meaning that they will never be turned off (unless the user “unmarks” them) regardless of whether or not they are within the plotting limits, or within the precision specification, or dominated by a subsequent non-domination subset sort. This allows the user to efficiently search the space in a variety of ways, all the while tracking solutions of interest. A solution export feature then allows the user to export the marked solutions to a file, which can subsequently be read back into the software to further refine the selections. This process of selection and refinement is referred to as negotiated design selection and is illustrated in Section 5.4.3.

5.4.3 Negotiated Design Selection

The process of selecting solutions of interest and subsequent refinement based on decision maker preferences is referred to as negotiated design selection. In this section, this process will be illustrated using one approach, although many others are possible because the VIDEO framework provides a multitude of visualization tools that can provide many means of selecting interesting solutions. We begin by identifying two objective subsets of the larger four objective LTGM test case. The two objective tradeoff representing Cost versus Error (Cost-Error) is shown in Figure 5.4A. The solutions in this figure have been marked in red using the “Mark Non-Dominated Subset” option available under the “Objective Space Tools” menu. The color used to mark the solutions can be specified under the “Mark Solution” button available in the objective tools. Keeping the Cost-Error tradeoff solutions marked from the previous set, the two objective Cost-Uncert tradeoff is shown in Figure 5.4B as solutions marked in green. This process can be repeated for all possible two objective tradeoff subsets: Cost-Mass, Error-Mass, and Uncert-Mass. Figure 5.4C shows all of the two objective tradeoffs marked within the Cost-Error space as shown in Figure 5.4A where the Cost-Error tradeoff is shown in red, the Cost-Uncert tradeoff in green, the Cost-Mass tradeoff in blue, the Error-Mass in purple, and the Uncert-Mass in orange. Looking closely at Figures 5.4A through 5.4C reveals that some solutions that are highlighted occur on more than one two
objective tradeoff. The VIDEO framework accounts for this by placing multiple marking boxes around solutions that occur on multiple non-dominated subsets. This allows the user to quickly identify intersections that exist between multiple tradeoffs.

At this point, all two objective LTGM tradeoffs have been identified and marked within the full four objective test case. Manipulating the plotting axes within the framework allows the decision maker to see what these tradeoffs look like in the context of the full four objective space. Figure 5.4D shows the full four objective Pareto set plotted similarly to Figure 5.2A with Cost, Error, and Uncert plotted on the spatial axes and Mass represented by the color of the solutions (where blue represents low Mass and red is high Mass). In Figure 5.4D, only the marked solutions are highlighted and the remaining solutions are shown in the background for locational perspective. One of the most interesting features identified from this figure is that the solutions within the Cost>Error tradeoff (red) and the Cost>Uncert tradeoff (green) generally have very high Mass error (red) solutions. Another very interesting feature is that the solutions between these tradeoffs appear as a geometric compromise region. In addition, based on the color of the actual solutions in the compromise region, it appears that they generally exhibit a lower Mass error (more blue and green solutions) than the Cost>Error and Cost>Uncert tradeoffs. These compromise solutions shown in closer detail and marked in Figure 5.4E are now the focus of further investigation.

Figure 5.4E focuses the objective space window on three solutions of particular interest that have been identified as compromise solutions between the Cost>Error and Cost>Uncert tradeoffs. These solutions are also marked in Figure 5.4D to provide a reference of where they are located in the full four objective space. First focusing on the dark blue solution (labeled 1) that represents the lowest Mass objective value in Figure 5.4E, this solution intersects the Cost>Mass, the Error>Mass, and the Uncert>Mass tradeoffs. Recall that the VIDEO framework provides a means of identifying this by retaining multiple bounding boxes of various colors indicating that the solution is intersected by multiple tradeoffs. The objective values expressed as a percentage of the maximum objective value of this solution are shown in column one of Table 5.1. Solution 1 shown in Figure 5.5A reduces Cost by 30-percent while only increasing Error by 20-percent and Uncert by 14-
Figure 5.4. Negotiated design selection within the VIDEO framework using two-objective subsets of the larger four-objective problem. Panel A shows the tradeoff Cost-Error marked in red and panel B, the tradeoff Cost-Uncert marked in green. Panel C shows the other two objective subsets Cost-Mass, Error-Mass, and Uncert-Mass marked in blue, purple, and orange respectively. Panel D shows the marked two objective subsets in the context of the full four objective space. Panel E shows the selection of three solutions of interest to be explored more fully in the negotiated design selection section.
Table 5.1. Summary of objective values expressed as a percentage of the maximum for each of the three solutions chosen as negotiated designs

<table>
<thead>
<tr>
<th>Three Solutions of Interest</th>
<th>1st</th>
<th>2nd</th>
<th>3rd</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cost as Percentage of Maximum</td>
<td>70%</td>
<td>65%</td>
<td>63%</td>
</tr>
<tr>
<td>Error as Percentage of Maximum</td>
<td>20%</td>
<td>17%</td>
<td>24%</td>
</tr>
<tr>
<td>Uncert as Percentage of Maximum</td>
<td>14%</td>
<td>13%</td>
<td>10%</td>
</tr>
<tr>
<td>Mass as Percentage of Maximum</td>
<td>0%</td>
<td>24%</td>
<td>13%</td>
</tr>
</tbody>
</table>

percent, while the increase in Mass error is actually very close to 0-percent. The relative reduction of each of the objectives’ values for the second and third solutions highlighted in Figure 5.4E are also shown in Table 5.1. In the table, each of the solutions 1 through 3 are associated with a correspondingly lower Cost. However, the solution that samples from 33 points in the domain actually has a lower Error value than the solution that samples from 35 points. In addition, the Cost = 33 solution is also better than the Cost = 35 solution in terms of the Uncert objective. Even more interesting is that the third solution that samples from only 32 points yields even further improvements in the Uncert objective. In terms of the Mass objective, the third solution presents a compromise between the Mass error of the first and second solutions.

Now that the relative magnitudes of the objective values for the three solutions have been compared, their corresponding maps of concentration estimates, estimation error, and estimation uncertainty can be probed and compared using VIDEO’s design space features. Figure 5.5 presents these maps for each of the three solutions compared (rows 1 through 3 of Figure 5.5) as well as the maps produced when all available well locations are sampled (designated by the row labeled “Full Cost”). Similarly to Figure 5.3, each plane probe is positioned at elevation $Z = 85$ m, and the iso-surface is plotted at a quantile concentration value of 0.7. In comparing the contaminant estimate planes of the three solutions, solution 2 appears to produce a map that most closely represents the information provided by the full test case. However, shifting to an iso-surface view (column 2 of Figure 5.5) reveals that the volumetric extent of high concentrations as predicted by solution 2 is underestimated. Comparing the estimation error maps in column 3 of Figure 5.5 reveals that solution 1 exhibits the lowest error at $Z = 85$ m. However, moving to the uncertainty map associated with solution 1 reveals a region of very
Figure 5.5. VIDEO comparison of solutions 1 through 3, which were selected for further consideration. Columns 1 through 4 of the figure represent the concentration estimates at $Z = 85$ m, a concentration iso-surface at 0.70, and Kriging error and uncertainty at $Z = 85$ m respectively. The first row of the figure represents the maps associated with the highest cost solution and the remaining rows, the maps associated with solutions 1 through 3 respectively.
high uncertainty present at the northern portion of the domain because the well at this boundary is not sampled in this solution. This likely indicates a preference for solution 2 since quantifying the extents of the contamination is important. Comparing the uncertainty maps of solutions 2 and 3 reveals a preference toward solution 2 since the uncertainty generally decreases throughout many portions of the domain. Based on the above analysis, either solutions 1 or 2 could be suitable as a negotiated design. Interestingly, solution 2 may have been difficult to locate using traditional approaches because it is not a solution that is intersected by any of the two objective tradeoff subsets. However, the VIDEO framework provided an efficient means of locating this solution that would certainly be of interest to a decision maker.

5.5 Discussion

Tools such as MOEAs are allowing engineers and scientists to solve “many-objective” problems [79] through the generation of high dimensional Pareto optimal solution sets. Relative to traditional multicriterion decision methods, MOEAs inherently shift methodological focus towards the challenges posed by providing DMs with the ability to explore and understand large tradeoff solutions sets. Development of the VIDEO framework is based on the hypothesis that design expertise and interactive visualization of both objective tradeoffs and their design space consequences can maximize the value and validity of using MOEAs as a post-eri-ori decision tools. Although exact quantitative metrics can be used to assess the quality of MOEA search (e.g., see [91, 109, 180]), it is much more subjective to judge the value of high order Pareto optimal sets in the design and decision making process.

However, the results in Figures 5.2-5.5 demonstrate how the VIDEO framework can allow environmental engineers and scientists to better understand LTGM tradeoffs while seeking a negotiated compromise solution. Perhaps the most interesting result in the prior section is the demonstration that the two objective Cost-Uncert and Cost-Error tradeoffs bound a compromise region in the four objective space (see Figure 5.4D). The value of the compromise region within the full four objective space is particularly interesting given that to date, a vast majority
of environmental applications using MOEAs have focused on two objective formulations due to their simple interpretations as well as their reduced computational constraints. As demonstrated in Figure 5.4D, the two objective nondominated sets by themselves do not capture the full range of design alternatives available. This chapter demonstrates that as we overcome the computational constraints posed by representing our problems with many-objective formulations, we should also advance our ability to move beyond traditional two objective cost-benefit analyses.

High-order Pareto optimization supports emerging decision and design philosophies that seek to allow experts and DMs to “shop” \([148, 195]\)) through sets of alternative solutions that will promote design innovations and provide DMs with an improved understanding of system behaviors. Moreover, our ability to mathematically abstract design into a suite of functions or norms that represent our design or decision support objectives is limited. It is inevitable that some aspects of performance or design will be unmodeled but remain important in the decision process \([82]\). The VIDEO framework synergistically supports DMs in better understanding modeled and unmodeled objectives by allowing DMs to interactively explore high dimensional objective spaces with large solution sets to understand their design tradeoffs while also providing spatial analysis of the design consequences of the LTGM tradeoffs.

More generally, the VIDEO framework demonstrates the need and value for environmental and water resources professionals to consider high-order Pareto optimization as a new problem class. MOEAs when combined with advanced visualization tools can serve to elucidate complex and potentially unknown dependencies between our objectives (both modeled and unmodeled) for environmental systems. Moreover, the environmental area is clearly being shaped by the emergence of spatiotemporal simulation and statistical tools that are closely coupled with geospatial information systems and geodatabases to support integrated assessment and management of impacted systems. The emergence of these tools further supports the value and need for a posteriori decision tools that couple advanced visualizations and MOEAs (for an excellent example see \([89]\)).
5.6 Conclusions

This chapter presents a framework for Visually Interactive Decision-making and Design using Evolutionary Multiobjective Optimization (VIDEO). The VIDEO framework was developed to affirm the need and value of combining interactive visualization with high-order Pareto optimization for improved \emph{a posteriori} decision making as demonstrated in this chapter using a long-term groundwater monitoring design application. The VIDEO framework allows users to visually navigate large multiobjective solution sets while aiding decision makers in identifying one or more optimal designs. Specifically, the interactive visualization framework is intended to provide an innovative exploration tool for high-order Pareto-optimal solution sets (i.e., solution sets for three or more objectives).

The framework is demonstrated on a LTGM application in which users can explore and visualize tradeoffs for up to four design objectives, simultaneously. Interactive functionality within the framework allows the user to select solutions within the objective space and visualize the corresponding monitoring plan’s performance in the design space. If a spatial estimation algorithm is used in the design objective formulation (e.g., in this chapter, QK was used to provide interpolated maps of contaminant concentration estimates, and their associated error and uncertainty), the estimation can be performed on-the-fly when a solution is selected and the corresponding maps can then be displayed in the decision space. Although QK is used to demonstrate the framework, it should be noted that it is adaptable to any spatiotemporal evaluation of LTGM designs (e.g., interpolators, smoothers, or filters). This functionality provides the user with a holistic picture of the information provided by a particular solution, ultimately allowing them to make a more informed decision. In addition, the ease with which the framework allows users to navigate and compare solutions as well as design tradeoffs leads to a time efficient analysis, even when there are thousands of potential solutions.
6.1 Abstract

This chapter contributes a combination of laboratory-based aquifer tracer experimentation and bias-aware Ensemble Kalman Filtering (EnKF) to demonstrate that systematic modeling errors (or bias) in source loading dynamics and the spatial distribution of hydraulic conductivity pose severe challenges for groundwater transport forecasting under uncertainty. The impacts of model bias were evaluated using an ammonium chloride tracer experiment conducted in a three dimensional laboratory tank aquifer with 105 near real-time sampling locations. The bias-aware EnKF framework (i) dramatically enhances the accuracy of concentration breakthrough forecasts in the presence of systematic, spatio-temporally correlated modeling errors, (ii) clarifies in space and time where transport gradients are maximally impacted by model bias, and (iii) expands the size and scope of flow-and-transport problems that can be considered in the future. The work presented in this chapter has been published in the journal *Geophysical Research Letters* [71].
6.2 Introduction

Eigbe et al. [137] provide an excellent review of groundwater applications of the linear Kalman Filter (KF) as well as the extended KF for nonlinear systems. The review highlights that very few studies exist for three-dimensional groundwater flow-and-transport applications due to these problems’ high-dimensional, nonlinear state spaces (i.e., heads and concentrations) and their consequent computational barriers. KF encompasses a general class of time-controlled state estimators that account for both process noise, and measurement noise simultaneously [196]. The KF proceeds iteratively in two steps: (i) a time update (or forecast) that projects the state and error covariance of the system ahead in time, and (ii) a measurement update that corrects the estimate and its associated error covariance by assimilating noisy measurement data. The use of extended Kalman filters for nonlinear groundwater flow-and-transport applications has been limited by a curse of dimensionality largely associated with the groundwater head and contaminant covariance matrices as well as the computational burden and nonlinearity posed by the transport problem.

These challenges have presented themselves in several scientific domains and motivated Evensen [128] to formulate the EnKF framework for nonlinear oceanographic applications. The EnKF is well suited to high dimensional state spaces where traditional Kalman filtering performs poorly in terms of its accuracy and computational complexity. This chapter contributes a combination of numerical and physical three-dimensional aquifer modeling to test a computationally efficient formulation for a bias-aware EnKF. The overall goal of this work is to minimize the computational demands associated with the bias-aware EnKF to provide high fidelity groundwater contaminant forecasts that account for measurement uncertainties and dynamic, spatiotemporally correlated model errors. This chapter provides the first demonstration of the effectiveness of bias-aware filtering for fully three-dimensional groundwater transport. In the broader geophysical research context, this work uses physical tank aquifer modeling to highlight that typical sources of model bias in subsurface chemical transport forecasts yield severe, spatiotemporally complex impacts on predictions. The experimental tank aquifer test case shows that unknown historical dynamics (e.g., concentration source terms,
recharge, or discharge) coupled with hydrogeologic uncertainties will severely limit our predictions of groundwater or chemical fluxes across material or process interfaces. This poses a significant challenge to future water cycle research and motivates the need for advances in groundwater transport forecasting (see [19, 23]).

6.3 Physical Aquifer Experiment

The experimental tank aquifer located at the University of Vermont (UVM) was constructed by filling a 254 cm wide $\times$ 356 cm long $\times$ 243 cm high container constructed using steel frame and panels with layers of carefully characterized porous media. Constant head boundaries are established using inlet and outlet reservoirs at each end of the tank to establish hydraulic gradients during experimentation. Five distinct layers of media as well as a fine sand lens were introduced during construction by packing media in 3 cm lifts from a catwalk in order to maximize its uniformity. Coarse sand was placed in the lowest layer (1) of the tank, followed by a layer (2) of fine silt, and then three layers (3 through 5) of medium sand. The fine sand lens was placed in the center of the tank within the middle of the three layers (4) of medium sand. Media layering, the location of the fine sand lens, as well as the experimentally determined hydraulic conductivity of each media layer are shown in Figure 6.1.

A sampling network of 21 wells constructed from 5.08 cm diameter polyvinyl chloride (PVC) pipe were placed throughout the tank as shown in Figure 6.1 prior to media placement. Each well has a 10 cm screen at four locations along its vertical axis, one in each media layer except the silt layer. In addition, each sampling well was equipped with a Time Domain Reflectometer (TDR) for measuring contaminant concentration, a thermocouple for measuring temperature, and a pressure transducer in each of the five media layers, providing a total of 105 locations from which to sample a variety of parameters. The relative locations of the various sensors are shown in Figure 6.1.

An ammonium chloride tracer was injected through sampling port B4 (see Figure 6.1) at an average concentration of 1 g-L$^{-1}$ and a rate of 1.5 L-hr$^{-1}$ over a 15 day period. Concentration data were collected at the 63 sampling ports in layers 3 through 5 over a 19 day period at 17.5 minute intervals.
Figure 6.1. Diagram of the UVM physical tank experiment’s dimensions, media layering and average calibrated K values, sensor and well locations, and constant head reservoirs.

The temporal variability and systematic increasing trend in the source term’s loading dynamics (see Figure 6.2A, port B4) provides a severe test of the bias aware EnKF formulation proposed in this study. The complex loading history for port B4 would not be known in typical groundwater transport applications and would be modeled (as we have done) with a simplified source loading history within the MT3DMS transport model. Additionally, despite significant experimental effort spent in characterizing the hydraulic conductivity fields in the tank using slug and pump tests, the fine sand lens in the center of the flow field served as a second major source of bias. The fine lens has substantial, fully three dimensional effects on the
spatiotemporal tracer breakthroughs that are also not captured in the MT3DMS simulation.

6.4 Bias-Aware Ensemble Kalman Filter

Using similar notation to Evensen [128] and extending upon the EnKF formulation presented in Chapter 2, Section 2.2.2, the bias-aware version of the EnKF uses \( N \) ensemble members to update \( n \) modeled states based on \( m \) observations for the \( j \)th time step. The ensemble of model \((\psi)\) and bias \((\eta)\) states are defined as:

\[
A = \begin{bmatrix} A_\psi \\ A_\eta \end{bmatrix} = \begin{bmatrix} (\psi_1, \eta_1) \\ (\psi_2, \eta_2) \\ \vdots \\ (\psi_N, \eta_N) \end{bmatrix} \in \mathbb{R}^{2n \times N}. \tag{6.1}
\]

The mean of \( A \) (denoted as \( \bar{A} \)) is obtained through multiplication with the matrix containing \( 1/N \) as its elements. The ensemble perturbations of \( A \) are found from \( A' = A - \bar{A} \), and the covariance of \( A \) is found from \( P = A'(A')^T/(N-1) \). During a forecast step, \((f)\) the bias portion of the state ensemble is updated using Equation 6.2

\[
A_{\eta,j}^f = \Lambda_j A_{\eta,j-1}^{a,f} + \delta_\eta Q_{\eta,j}, \tag{6.2}
\]

where \( \Lambda_j \) describes the time correlation of the model bias, \( Q_\eta \) is an ensemble of independent sources of zero-mean, spatially correlated noise on the bias states, and \( \delta_\eta \) is a scaling factor on this noise. The prior bias state \( A_{\eta,j-1} \) may either be based on a forecast \((f)\) or update \((a)\). Once the bias state is forecast, the ensemble of tracer concentration states is forecast using Equation 6.3

\[
A_{\psi,j}^f = A_{\psi,j-1}^{a,f} - A_{\eta,j}^f + \delta_\psi Q_{\psi,j}, \tag{6.3}
\]

where \( Q_\psi \) is an ensemble of independent sources of zero-mean, spatially correlated noise on the model states, and \( \delta_\psi \) is a scaling factor on this noise. Notice that Equation 6.3 fully incorporates the bias state forecast into the model state forecast (referred to as full feedback, [197]). When observations are available, the ensemble of bias states can be updated based on those observations. First, a subset of updated bias states is calculated as the difference between the observations and the
model state forecasts at the observation locations. However, since \(m\), the number of observations is typically much fewer than \(n\), the number of model states, a Gaussian model is fit to the covariance structure of the bias obtained at the \(m\) observation locations. A contribution of this work is to spatially condition the mean bias field and its covariance on observed modeling errors using Sequential Gaussian Simulation (SGS, [129]). The bias present at the observation locations is used to conditionally simulate an ensemble of bias states at all \(n\) model state locations. Equation 6.4 is used to update the ensemble of bias states, \(\mathbf{A}_\eta\),

\[
\mathbf{A}_{\eta,j}^a = \Lambda_j \mathbf{A}_{\eta,j-1}^f + (1 - \Lambda_j) \mathbf{W}_j, \tag{6.4}
\]

where \(\Lambda_j\) again describes the time correlation of the model bias, and \(\mathbf{W}\) is the simulated ensemble of bias states based on the newly available observations. Our Bayesian conditioning of the mean and covariance of the bias field substantially improves the spatial representation of systematic model errors and takes advantage of observations to permit more advanced representations of the spatial anisotropy of model error fields. It is interesting to note that Equation 6.4 provides a mechanism for analyzing three-dimensional time varying maps of how transport gradients impact prediction errors. These maps can provide a unique mechanism for evaluating changes or improvements in the forward model’s process formulations. Once the bias ensemble has been reinitialized, the model states are updated using the analysis approach of [128]. Under the ensemble formulation, random, zero-mean perturbations based on the assumed observation error are used to obtain an ensemble of observations \(\mathbf{D}\) with perturbations \(\mathbf{Y}\). The covariance of the perturbations is then described by \(\mathbf{R} = \mathbf{Y}\mathbf{Y}^T/(N-1)\). The Kalman Gain matrix, \(\mathbf{K}\), which can be thought of as a blending factor to minimize the \textit{a posteriori} error covariance of the filter [196], is then calculated using Equation 6.5

\[
\mathbf{K}_j = \mathbf{P}_j^f \mathbf{H}^T (\mathbf{H} \mathbf{P}_j^f \mathbf{H}^T + \mathbf{R}_j)^{-1}, \tag{6.5}
\]

where \(\mathbf{P}\) is the forecast covariance of the state ensemble, \(\mathbf{H}\) is a measurement operator that maps the \(m\) observation locations onto the \(n\) model states, and \(\mathbf{R}\) is the covariance of the observation noise ensemble. \(\mathbf{K}\) is a \(2n \times m\) matrix where the first \(n\) rows represent the model state (\(\psi\)) portion of \(\mathbf{A}\) and the last \(n\) rows, the
bias state ($\eta$) portion of $A$. Finally, the state ensemble is updated using Equation 6.6

$$A_{\psi,j}^a = A_{\psi,j}^f + K(D_j - HA_{\psi,j}^f),$$

where $A_{\psi,j}^f$ is the state forecast and $A_{\psi,j}^a$, the updated or analyzed state. The updated covariance of $A$ is then found from the forecast covariance using Equation 6.7

$$P_{\psi,j}^a = (I - K_jH)P_{\psi,j}^f.$$

### 6.5 Computational Experiment

The tracer experiment was modeled in MODFLOW [198] and MT3DMS [199] using a grid resolution of $X = 35$ cells ($\delta_x = 7.837$ cm), $Y = 50$ cells ($\delta_y = 7.874$ cm), and $Z = 33$ cells ($\delta_z$ ranging from 7.3 cm to 5.08 cm) for a total of 57,750 model cells. No flow boundary conditions were established at the bottom and sides of the tank and constant head boundary conditions of 203.2 cm at the inlet and 200.6 cm at the outlet reservoirs were established.

A separate, non-point source ammonium chloride tracer test (1 g-L$^{-1}$) was used to calibrate MODFLOW and MT3DMS for the tank. Tracer breakthrough curves for the sampling ports in all media layers were fit simultaneously using PEST 10.2 [200] by adjusting hydraulic conductivity ($K$), porosity, and dispersion to minimize the RMSE between observed and modeled transport. The calibrated $K$ values were used as conditioning points for simulating multiple realizations of the $K$ field using SGS. This provides a distribution of hydraulic conductivity fields, in turn providing an ensemble of transport model realizations to be used within the ensemble framework of the filter.

Since the media in the tank was layered during construction, the $K$-field of each media layer was conditionally simulated individually to eliminate SGS smoothing effects. The flow realizations were used in MT3DMS to model 100 realizations of the tracer transport over time steps of 8.1 minutes and two stress periods (3330 steps of tracer injection and 209 steps with no tracer) for a total simulation period of 20 days. Longitudinal dispersivities ranged from 6.44 cm in the course sand layer to 0.44 cm in the silt layer and the dispersivity ratios (relative to longitudinal) used in the transverse and vertical directions were 0.17. The diffusion coefficient was
specified as \(1 \times 10^{-4}\), and a Courant number of 0.9 was used to minimize numerical dispersion. An initial tracer concentration of zero was assumed throughout the domain.

An ensemble size of 100 realizations was used for the EnKF. To enhance the computational efficiency of the EnKF, a filtering subdomain was defined at a courser resolution of 588 cells bounding the simulated regions with non-zero tracer concentration within the overall numerical model grid containing 57,750 cells. The filtering subdomain was defined to have at least one grid cell between every cell containing observation points in all three dimensions to support spatial evaluation of the filter. The time correlation factors, \(\Lambda\) for the bias forecast and reinitialization were set to 99 percent. The spatially correlated noise fields, \(Q_\eta\) and \(Q_\psi\) were generated for the filtering subdomain using a Gaussian covariance model with sill = 1.0 and range = 5 cells. These fields were then scaled using noise scaling factors \((\delta_b\text{ and } \delta_m)\) of 15 percent. The noise applied to the observations was normally distributed random noise with mean 0 and standard deviation 0.05 to represent approximately 5 percent error. The filter was then run for the full 19 day experiment with varying assimilation intervals. Filter forecasts were conducted at intervals of 2 hours while observation assimilation was conducted at intervals of 2 hours, 6 hours, 12 hours, and 24 hours for four separate test cases. The 2 hr forecast, 2 hour assimilation is defined as the maximum data case, meaning that the filter was used to forecast the tracer, and then was immediately updated in the same time step with the observation data. For the 24 hour case, 12 forecasts occurred before the filter was updated using the observations at the 24th hour.

6.6 Results and Discussion

Figure 6.2 shows that the bias-aware filtering is able to effectively capture complex tracer loading behavior, reduce errors due to the spatial distribution of hydraulic conductivities (i.e., the fine sand lens), and resolve advective gradients not captured in the original MODFLOW/MT3DMS simulation. It should be noted that complex contaminant source histories are rarely known and represent a very large source of model bias when simplified source area dynamics are assumed.

Figure 6.2A shows the tracer breakthrough (BT) curves associated with the
Figure 6.2. Panel A shows the tracer breakthrough curves associated with the sampling ports in layer 4 plotted as normalized concentration, $c/c_0$, versus time for the model, observed, and filtered concentration time-series. Panel B shows the BT curves associated with the tracer source location (port B4) for the various assimilation cases (2 hours, 6 hours, 12 hours, and 24 hours) where the gray shading represents one standard deviation above and below the ensemble mean. Panel C provides normalized tracer concentration maps (two top views of the layer 4 and layer 3 sampling ports, a side view along the center of the tank, and a 3 dimensional view) provided by the model at $t = 13$ days. In addition, the coloring of the sampling ports indicates observed $c/c_0$ concentrations at this time step. Panel C shows the numerically modeled mean tracer concentration at $t = 13$ days. Panel D shows the mean filtered maps for assimilation every 2 hours at $t = 13$ days (formatted similarly to Panel C for comparison).
sampling ports in layer 4 plotted as normalized concentration versus time. The green line indicates the modeled BT curve, the blue dots represent observed concentration, and the ensemble means from the bias-aware filter assimilation cases are portrayed using a red color scale. Dark red lines represent the maximum data case (assimilation every 2 hours) and the light pink lines represents the minimum data case (assimilation every 24 hours). Without bias awareness, the model would require reinitialization after each assimilation step, resulting in costly Monte Carlo simulations of the groundwater flow and transport models.

Figure 6.2A demonstrates that by modeling the bias, the filter can effectively correct for large systematic model errors. If this were not done, the filter would quickly track back to the modeled BT curves whenever an observation was not available. Figure 6.2B shows the BT curve associated with the tracer source location for each of the assimilation rates (2 hours, 6 hours, 12 hours, and 24 hours). The gray shading in these plots represents one standard deviation above and below the ensemble mean. Figure 6.2B shows that the complexity of source loading history is a significant source of error in the model. In addition, the figure shows a clear increase in uncertainty as the time between data assimilation is increased from 2 hours to 24 hours. However, it is interesting to note that even for the minimal data case (assimilation every 24 hours), the observed tracer concentration is always captured in the range of the bias aware filter’s projected uncertainty ranges without the need to reinitialize and rerun the full Monte Carlo ensemble.

The mean tracer concentration maps of the numerical model are shown in Figure 6.2C for $t = 13$ days. Four different views of the tracer plume are shown in this figure: (i) an $X$-$Y$ slice through the layer 4 sampling ports, (ii) an $X$-$Y$ slice through the layer 3 sampling ports, (iii) a $Y$-$Z$ slice (side view) through the center of the tank, and (iv) a 3-dimensional view of the tracer plume. The coloring of the maps is based on the normalized tracer concentration ($c/c_o$) where red indicates maximum concentration and blue indicates concentration below the detection limit. The coloring of the sampling ports indicates observed normalized tracer concentrations at this time step. Figure 6.2C shows that the numerical model fails to capture the true behavior of the plume because the sampling ports in layer 4 indicate low tracer concentration at the leading edge of the plume and the sampling ports in layer 3 indicate low tracer concentration at the source region.
of the plume. The impact of the fine-sand lens in the center of the tank results in significant systematic errors in the spatial and temporal contaminant BT behavior for the tracer. However, Figure 6.2D provides the mean filtered maps of the maximum data case (forecasts following by assimilation every 2 hours) at $t = 13$ days. The maps in Figure 6.2D indicate that bias-aware assimilation of observations is far more effective for capturing the experimental observations for the system. Not only are the observation locations appropriately corrected by the filter, but the spatially adjacent locations are improved as well. The effect of the fine sand lens is shown in these maps as a build-up of tracer developing at the source edge of the lens, and subsequent movement down and through the lens (as would be expected when hydraulic conductivity decreases at a boundary).

### 6.7 Conclusions

The water cycle research community has highlighted that uncertainties in the subsurface [201] and frameworks for improving flux predictions are top research needs [23]. This work contributes a combination of laboratory tracer experiments and numerical modeling to clearly demonstrate that commonly occurring limitations in our knowledge of historical dynamics (source loading in this study) and subsurface flow properties yield severe sources of model bias, even for a well characterized sand-dominated laboratory system. Using a 19-day ammonium chloride tracer experiment, this chapter shows that filtering systematic model errors in groundwater transport predictions separately and allowing them to feed back to concentration predictions yields enhanced forecasting accuracy while eliminating the need to reinitialize and rerun forward model ensembles. Consequently, this work considerably expands the size and scope of groundwater flow-and-transport problems where the bias-aware EnKF can be applied. It would be expected that the need for and value of the bias-aware EnKF framework will only increase for more complex field environments. As our ability to predict multi-process flow-and-transport at watershed scales advances [202], the bias-aware EnKF framework presented in this chapter can transform observation networks and forecasting frameworks to better account for limits in our knowledge of the subsurface.
7.1 Abstract

This chapter contributes the ASSIST (Adaptive Strategies for Sampling in Space and Time) framework for improving long-term groundwater monitoring decisions across space and time while accounting for the influences of systematic model errors (or predictive bias). The new framework combines contaminant flow-and-transport modeling, bias-aware ensemble Kalman filtering (EnKF), many-objective evolutionary optimization, and visual analytics-based decision support. The ASSIST framework allows decision makers to forecast the value of investments in new observations for many-objectives simultaneously. Information tradeoffs are evaluated using an EnKF to forecast plume transport in space and time in the presence of uncertain and biased model predictions that are conditioned on uncertain measurement data. This chapter demonstrates the ASSIST framework using a laboratory-based physical aquifer tracer experiment. In this initial demonstration, the position and frequency of tracer sampling was optimized to: (i) minimize monitoring costs, (ii) maximize the information provided to the EnKF, (iii) minimize failures to de-
tect the tracer, (iv) maximize the detection of tracer fluxes, (v) minimize error in quantifying tracer mass, and (vi) minimize error in quantifying the centroid of the tracer plume. Results demonstrate that the forecasting, search, and visualization components of the ASSIST framework represent a significant advance for observation network design that has a strong potential to innovate our characterization, prediction, and management of groundwater systems. The work presented in this chapter has been submitted for review to the journal *Water Resources Research* [74].

### 7.2 Introduction

Using a groundwater application context, this chapter contributes the ASSIST (Adaptive Strategies for Sampling in Space and Time) framework to advance our ability to manage the technical barriers posed by observation network design as a general class of problems. As demonstrated in this chapter, the ASSIST framework is a highly adaptable methodology for improving long-term groundwater monitoring (LTGM) decisions across space and time while accounting for the influences of systematic model errors (or predictive bias). This chapter demonstrates how bias-aware Ensemble Kalman Filtering (EnKF) [71], many objective (i.e., greater than three objectives) search using hierarchical Bayesian optimization [64], and interactive high dimensional visual analytics [72] can be combined to facilitate discovery and negotiation in the LTGM design process. Use of the terms discovery and negotiation is motivated by the potential of many objective solution sets to identify alternatives that capture a broad suite of system behaviors relevant to both modeled and unmodeled objectives (see Brill et al. [203]). This ultimately enables decision makers to discover system dependencies and/or tradeoffs and exploit this information in the adaptive long-term management of observation systems.

The ASSIST framework is demonstrated using a laboratory based physical aquifer tracer experiment where the position and frequency of tracer samples are optimized to (i) minimize monitoring costs, (ii) maximize information provided by the EnKF, (iii) minimize failures to detect tracer, (iv) maximize the detection of tracer flux (i.e., strong increases in concentration), (v) minimize error in quantifying tracer mass, and (vi) minimize the error in quantifying the centroid of the
tracer plume. This analysis represents a synthesis of prior LTGM methodologies as was discussed in Chapter 1 and highlights the potentially negative consequences of traditional one and two objective analysis techniques. In combination, the forecasting, search, and visualization components of the ASSIST framework represent an advance relative to the traditional OSSE concepts from the meteorological literature, which has a strong potential to innovate our characterization, prediction, and management of groundwater systems.

### 7.3 Prior Work

Prior literature has predominately focused on spatial network decisions. The use of Kalman filtering has yielded notable exceptions where spatial and temporal monitoring decisions are considered simultaneously [41, 52, 54–56, 69]. These Kalman filtering applications represent important advances in LTGM that more directly address the statistical nonstationarity of groundwater heads and contaminant concentrations, which strongly limits the value of geostatistical frameworks in dynamical systems. However, the traditional Kalman filter based LTGM approaches are still limited in their representation of the uncertainties and biases associated with groundwater observations and models.

Eigbe et al. [137] provide an excellent review of groundwater applications of the linear Kalman filter as well as the extended Kalman filter for nonlinear systems. The review highlights that very few studies exist for three dimensional groundwater flow and transport applications due to these problems’ high dimensional, nonlinear state spaces (i.e., heads and concentrations) and their consequent computational barriers. Notable contributions for the application of extended Kalman filtering have focused on analytical [163, 164, 204] and numerical techniques [205, 206] for reducing the computational constraints posed by recursively updating the mean and covariance of contamination fields. Although these studies advanced the applicability of the extended Kalman filter for both parameter and multi-state estimation, the methodology is still limited in its ability to model highly nonlinear systems, especially in the presence of systematic modeling errors and/or observation errors. These challenges motivated the development and use of the bias-aware EnKF [71, 128, 197] in the ASSIST framework.
The EnKF’s basis in Monte Carlo simulation provides the framework with numerous benefits, the details of which are discussed in Chapter 2, Section 2.2.2. The combined use of many objective evolutionary optimization coupled with bias aware EnKF allows the ASSIST framework to synthesize prior methodologies by allowing users to specify objectives from any or all of the categories summarized in Table 1.1 of Chapter 1. Moreover, by quantifying and interactively visualizing the tradeoffs between these objectives, the ASSIST framework can dramatically advance our understanding of the value of new or existing observables across space and time given the uncertainties and systematic biases that commonly constrain our predictions in practice.

7.4 Methodology

For this demonstration, a tracer experiment conducted within a tank-based physical aquifer experiment at the University of Vermont (UVM), was modelled with systematic errors (or biases) in its initial conditions and hydraulic conductivity. Bias-aware ensemble Kalman filtering (EnKF) forecasts of the location and extent of the tracer plume are combined with many-objective hierarchical Bayesian search to determine the approximately optimal tradeoffs for six monitoring design objectives. Each of these methodological components are discussed in more detail in Sections 7.4.1 through 7.4.5.

7.4.1 Physical Aquifer Experiment

A unique contribution of this work is to exploit the University of Vermont (UVM) experimental aquifer utilized in Chapter 6 to perform advective-scaled observation system simulation experiments [13]. This work expands the OSSE concept to include the formal exploration of forecasted observations using many-objective network optimization. As an OSSE platform, the UVM tank aquifer was designed to approximate a temporal advection scaling where 1 day of tracer transport would be equivalent to 1 year at field scale. Likewise, 1 cm of longitudinal transport within the tank is approximately equivalent to 1 m at field scale. A 19 day ammonium chloride tracer experiment where a hydraulic gradient of 2.9 cm was established
over the length of the tank is used to demonstrate the ASSIST framework.

In the tracer experiment, ammonium chloride was injected through port B4 (see Figure 7.1) at an average concentration of 1 g-L$^{-1}$ and at a rate of 1.5 L-hr$^{-1}$ over a period of 15 days. Concentration data was then collected from the TDR probes located in layers 3 through 5 (63 total locations) over a period of 19 days at intervals of 17.5 minutes. Note that the high variability and complex loading dynamics of the tracer injection at B4 would not be available for typical groundwater transport modeling efforts and would be modeled with a simplified source term similar to the one we have used (see port B4 in Figure 7.1). As highlighted in Chapter 6 [71], the simplification of the initial condition creates a very severe bias in transport forecasts that are compounded by uncertainties in the hydraulic conductivities (despite a broad range of tracer, slug, and pump tests). The uncertain hydraulic conductivity of the fine sand lens in the center of the tank exacerbated the fully three-dimensional systematic bias effects that arise from our model’s simplification of the complex ammonium chloride loading dynamics. Overall the experimental conditions pose a severe test for the ASSIST framework’s ability to forecast approximately optimal tradeoffs for sampling decisions from the tank’s pre-existing ports in both space and time. Additional experimental details on the UVM tank are available in Chapter 6.

7.4.2 Numerical Flow and Transport Modeling

Subsurface flow was modeled using the parallel, three-dimensional variably saturated groundwater flow model ParFlow [22, 207–209]. ParFlow is under collaborative development by the Colorado School of Mines, the Center for Applied Scientific Computing at Lawrence Livermore National Labs, and the University of Bonn. The software has been built to effectively utilize parallel supercomputing environments, which enables it to produce highly detailed simulations that set it apart from other groundwater models in terms of accuracy and performance [202, 210–212]. ParFlow uses a Newton-Krylov method [213] to solve Richards’ equation [214] for variably saturated groundwater flow and has also been fully coupled to integrated surface and sub-surface flow. In addition, ParFlow has been developed to efficiently handle subsurface heterogeneity and uncertainty by providing paral-
Figure 7.1. Top view of UVM physical aquifer tank with breakthrough curves of normalized tracer concentration versus time plotted at the approximate geometric location of each monitoring well. The dashed blue curves show the high resolution tracer concentration observations while the solid green curves show the simulated tracer concentration using ParFlow and SLIM-FAST for monitoring ports located in media layer 4. Note that the entire 19 year simulation period is shown.

Contaminant transport was modeled using the numerical particle transport model SLIM-FAST [216–220]. SLIM-FAST uses the Lagrangian Random Walk Particle Method (RWPM) to transport particles that represent the concentration and location of contaminant mass throughout a saturated media based on the velocity field resulting from numerical flow forecasts attained from a flow model like ParFlow. Chemical diffusivity, reactivity and radioactive decay are also effectively simulated by SLIM-FAST using a particle-probability approach (e.g., see Maxwell et al. [218], supplemental material). Particle based transport models such as SLIM-FAST have been shown to be effective simulation tools as they tend to not suffer as much from concentration negativity, numerical dispersion, and mass balance inconsistencies [221, 222]. SLIM-FAST has been effectively demonstrated
in tandem with ParFlow in a variety of studies [202, 212, 218]. In this demonstration, SLIM-FAST was used to simulate realizations of contaminant plumes based on their corresponding flow field realizations generated using ParFlow.

### 7.4.3 Bias-Aware Ensemble Kalman Filtering

The EnKF [128, 223] was developed and has garnered growing popularity for its ability to provide statistical forecasts of nonlinear system states while accounting for simulation uncertainty and measurement uncertainty simultaneously within an ensemble Monte Carlo framework. A bias-aware version of the EnKF [71, 197] is used in this work to correct system states by accounting for model uncertainty and using a Bayesian feedback that accounts for systematic model bias to improve transport forecasts. Full details of the bias-aware EnKF formulation are presented in Chapter 6, Section 6.4.

### 7.4.4 Optimization Algorithm

Efficiently and effectively sampling the UVM tracer experiment represents an extremely large combinatorial decision space in which complex interdependencies exist between sampling decisions in both space and time due to the underlying physics of flow-and-transport. The $\varepsilon$-Dominance Hierarchical Bayesian Optimization Algorithm ($\varepsilon$-hBOA) [64], a variant of the original hBOA algorithm [116, 224] was used in the ASSIST framework to search for space-time well sampling schemes that satisfy multiple conflicting stakeholder objectives simultaneously. Traditional multi-objective evolutionary algorithms (MOEAs) mimic Darwinian evolution to evolve populations of solutions toward optimality using selection, crossover, and mutation operators. However, the $\varepsilon$-hBOA represents a new class of MOEA that replaces the traditional crossover and mutation operators with Bayesian network model building, allowing the algorithm to learn and exploit the hierarchical interdependencies that exist between decisions [64]. Additional details of the $\varepsilon$-hBOA can be found in Chapter 4.

As highlighted in Chapter 5 [72], many-objective optimization coupled with visualization provides valuable insights to monitoring decision makers. If conflicts (or tradeoffs) exist between the objectives, a set of Pareto optimal [81] solutions
will result where increases in performance in one of the objectives results in a
degradation in performance of one or more of the remaining objectives [96]. For
combinatorial problems such as LTGM design where decisions may have complex
interdependencies, the $\varepsilon$-hBOA has proven to be quite effective at evolving high
quality approximations to many-objective tradeoff surfaces and has outperformed
other state-of-the-art MOEAs that assume independent decisions, particularly for
increasing numbers of objectives or decision variables [64, 225].

7.4.5 Design Objectives

The LTGM design objectives have been formulated to exploit the data available
from the prior management period in EnKF forecasts of contaminant migration
for the subsequent management period for which new monitoring decisions are to
be made. In other words, for any given planning horizon, the design objectives
utilize the data made available in the prior planning period when making uncertain
forecasts of the impacts of potential sampling decisions in the future period.

For this demonstration, a one year management period is assumed where each
potential sampling location can be sampled up to four times throughout the year
(i.e., quarterly sampling). Note that in the remainder of this chapter, the field
scale time analogs of the ammonium chloride tracer experiment (1 day of experi-
mental tank transport is equivalent to 1 year at field scale) is used. This provides
the convenience of discussing the LTGM problem in more conventional time units.
For each 1 year management period, the optimization framework seeks to forecast
where and when samples should be taken to maximize knowledge of the plume’s
movement. While this demonstration uses annual management periods and quar-
terly sampling frequencies based on common practice, the approach is highly flex-
ible and a broad range of alternative implementations are possible depending on
the availability of data.

A total of six design objectives were formulated, four of which were minimized
and two of which were maximized, in order to capture a wide range of decision
maker preferences and prior literature recommendations. The design objective
vector $\mathbf{F}$ shown in Equation 7.1 is a function of the decision vector $\mathbf{x}$ which is
composed of binary 1’s and 0’s indicating whether an available sampling location
is monitored in space and time.

\[ F(x) = (f_{\text{Cost}}, f_K, f_{\text{DF}}, f_{\text{Flux}}, f_{\text{Mass}}, f_{\text{Centroid}}) \] (7.1)

The current management period is defined by \( k \), which represents the time period for which uncertain forecasts of expected plume movement must be made. Hence \( k - 1 \) refers to the prior management period for which a sampling decision has already been made and its consequent data are available. Within each management period \( k \), there are \( t = 1 \ldots T \) sampling periods, where for this demonstration, \( T = 4 \) since quarterly sampling is assumed. Consistent with the notation for the EnKF, \( M \) is defined as the total number of available sampling locations (63 for this demonstration) at a given time and \( m \) as the number of samples actually utilized based on a decision for a given time \( t \). Hence, \( m \) ranges from 0 \ldots M, or equivalently, no sampling to all available samples taken. Now define \( \Omega \) as the entire decision space for a management period (i.e., the \( 2^{MT} \) possible combinations of sampling decisions for the management period) and \( \omega \) as an instance of a sampling plan from this space. The coordinate of a sample at location index \( s \), at time index \( t \), is now defined as \( u_{s,t} \). The decision vector \( x_{\omega,k} \) represents a potential sampling plan for an entire management period and can be broken into sub-vectors for each corresponding sampling period \( 1 \ldots T \) within the larger management period as shown in Equation 7.2. Each \( x_t \) sub-vector is composed of binary decisions \( x \) within the management period as shown in Equation 7.3. The decision vector \( x_{\omega,k} \) is composed of \( M \cdot T \) binary decisions (1’s and 0’s) indicating searchable decisions on where and when samples can be taken within the \( k \)th management period.

\[ x_{\omega,k} = \begin{bmatrix} x_1, \ldots, x_t, \ldots, x_T \end{bmatrix} \] (7.2)

\[ x_t = \begin{bmatrix} x(u_{1,t}), \ldots, x(u_{s,t}), \ldots, x(u_{M,t}) \end{bmatrix} \]

where \( x(u_{s,t}) = \begin{cases} 1, & \text{if well } s \text{ is sampled at time } t \\ 0, & \text{otherwise} \end{cases} \) (7.3)

Cost. The Cost objective seeks to minimize the cost of a sampling plan \( x_{\omega,k} \). A
cost vector $C_t$ is now defined, which is a function of the cost of sampling from each location $s$ at time $t$. Since the planning horizon is one year, a normalized cost vector is used meaning that all locations in space and time are equally weighted and normalized to 1. Equation 7.4 describes the cost minimization objective where $C_t$ and $x_t$ are both vectors of size $1 \times M$.

$$\text{Minimize: } f_C(x_{\omega,k}) = \sum_{t=1}^{T} C_t \cdot x_t$$ (7.4)

The cost of a sampling plan for a management period is obtained by summing up the dot products between the cost vector $C_t$ and the decision vector $x_t$ for each $t$ in a management period $k$. The cost objective aims to minimize the cost in the present planning period.

$K$. The Kalman Gain matrix, $K$, represents a blending factor that weights measurements and model states based on their corresponding uncertainty and the sample innovation (the difference between concentration measurements and their mean EnKF forecasted values). Maximizing the Kalman Gain can be thought of as maximizing the “value of information” provided by the samples. During a forecast period, although no measurements are actually assimilated, it is still possible to calculate the filter covariance and the corresponding Kalman Gain that would be achieved if a measurement were in fact assimilated, ultimately providing a forecast of the potential corrective capability that would be provided by a sampling plan $x_{\omega,k}$. In the bias-aware EnKF, the Kalman Gain matrix $K$ is a $2n$ ($n$ model states + $n$ bias states) by $m$ (samples) matrix where elements close to zero represent little to no correction for either the concentration state prediction or the bias correction resulting from sampling a candidate measurement location. Elements in $K$ that deviate from zero either positively or negatively indicate larger corrections. Maximizing $K$ has two effects: (i) it minimizes uncertainty in forecasts (i.e., the forecasted covariance, see Equation 6.6 in Section 7.4.3) and (ii) it maximally corrects the model state forecasts and the systematic bias corrections. Equation 7.5 shows the mathematical formulation of the $K$ maximization objective.

$$\text{Maximize: } f_K(x_{\omega,k}) = \min_{1\leq t\leq T} \left\{ \frac{1}{m} \sum_{s=1}^{m} \left( \sum_{g=1}^{n} k^2_{g,s} \right)^{1/2} \right\}$$ (7.5)
A Euclidean norm is calculated along each column of \( K \), effectively providing the corrective distance for the entire model and bias state spaces that could be achieved by a sample \( s \) at some time \( t \). Since the maximum overall correction capability provided by a decision vector \( x_t \) is sought, this design objective is formulated to maximize the minimum average correction distance across each of the quarterly sampling periods as represented by each column of \( K \). In the notation, \( k_{g,s} \) is simply an individual element of \( K \) at row \( g \) and column \( s \). Using the max-min formulation in Equation 7.5 ensures maximization of the improvement of the worst performing forecast periods and consequently ensures that all other periods are as good or better. This max-min formulation has the effect of improving the robustness of the evaluations of the \( K \) objective [226].

**DF.** The ability of a sampling plan to detect the movement of contaminant is evaluated using the detection failure (DF) indicator objective. DF is defined here as the probability that a given sampling plan fails to sample at locations where contaminant is forecast to exist at some detectable level according to the EnKF. To calculate this objective, a detection indicator vector \( DI_i \) is developed for each ensemble member \( i \) of the EnKF. The detection indicator vector contains 1’s and 0’s indicating whether or not contaminant concentration \( c_i \) exists at a detectable level at ensemble member \( i \)’s sampling locations for the time step under consideration. In this demonstration, a detection indicator threshold of 0.075 g-L\(^{-1}\) is used. The DF objective is shown in Equation 7.6.

Minimize: 

\[
f_{DF}(x_{\omega,k}) = \max_{1 \leq t \leq T} \left\{ \frac{1}{N} \sum_{i=1}^{N} \left( 1 - \frac{x_t \cdot DI_i}{\|DI_i\|_1} \right) \right\},
\]

where \( DI_i(u_{s,t}) = \begin{cases} 
1, & \text{if } c_i(u_{s,t}) \geq c_{\text{threshold}} \\
0, & \text{if } c_i(u_{s,t}) < c_{\text{threshold}}
\end{cases} \quad (7.6)
\]

This objective seeks to minimize the maximum expected DF across each of the quarterly sampling periods. In other words, it is minimizing the worst performing sub-sampling period’s DF and consequently ensuring that all other periods are as good or better. The sum of the instances of contaminant detection are quantified by taking the dot product between the decision vector \( x_t \) and the detection indicator vector \( DI_i \) for each ensemble member (i.e., the number of sampled locations in the
sampling plan $x_t$ where contaminant is forecast by an ensemble member at a level $\geq 0.075$ g-L$^{-1}$). The total possible contaminant detection $\|D_i\|_1$ is obtained by taking the one-norm (element-wise summation) of the $D_i$ vector (i.e., the total number of available measurement locations $M$ where contaminant is forecast at a level $\geq 0.075$ g-L$^{-1}$).

**Flux.** The Flux objective seeks to maximize the detection of high mass flux gradients. Contaminant flux at a sampling location $s$ is quantified by multiplying the contaminant concentration $c_i(u_{s,t})$ for an ensemble member $i$ by the Darcy flux $q_i(u_{s,t})$ at that location and dividing by the media porosity $\phi(u_s)$. Equation 7.7 defines the Flux objective by incorporating the binary sampling decision $x(u_{s,t})$ to effectively detect mass fluxes in space and time.

Maximize: $f_{\Phi} (x_{\omega,k}) = \min_{1 \leq t \leq T} \left\{ \sum_{s=1}^{M} \left( \frac{1}{N} \sum_{i=1}^{N} x(u_{s,t})q_i(u_{s,t})c_i(u_{s,t})}{\phi(u_s)} \right) \right\}$  \hspace{1cm} (7.7)

The ensemble averaged concentration flux detected by a sampling plan is calculated within the parentheses of Equation 7.7. This objective seeks to maximize the minimum total contaminant flux detected by a sampling plan $x_t$, across each of the quarterly sampling periods for a management period. In other words, it is maximizing the worst performing sub-sampling period’s total Flux detection and consequently ensuring all other periods are as good or better than the worst.

**Mass and Centroid.** The zeroth and first spatial moments of the contamination plume represent the mass of contamination and the centroid of the plume, respectively. This formulation builds on Montas et al. [62] to define spatial moments based objectives to quantify error in tracking contamination plumes through space and time. Equations 7.8 and 7.9 describe the zeroth and first spatial moments of the contaminant plume. Note that the first moment in Equation 7.9 provides the X-direction only and has analogous centroid calculations in the Y- and Z-directions.

\[
\mu_{0,t} = \sum_{g=1}^{n} \phi(u_g)\bar{c}(u_{g,t})V(u_g) \hspace{1cm} (7.8)
\]

\[
\mu_{1X,t} = \frac{1}{\mu_{0,t}} \sum_{g=1}^{n} \phi(u_g)X(u_g)\bar{c}(u_{g,t})V(u_g) \hspace{1cm} (7.9)
\]
Each moment is calculated based on the ensemble mean concentration $\bar{c}(u_{g,t})$, at each filter grid location $g$ in time $t$. The summation is performed over all $n$ model states of the filter grid. Also, $\phi(u_g)$ represents the porosity of the media at each grid location and $V_g$, the volume of a grid cell.

The Mass and Centroid moment objectives formulated in Equations 7.10 and 7.11 are defined relative to the “best” available forecast of the plume (its moments being $\mu'_0,t$ and $\mu'_1,t$) which results from the full sampling decision implemented in the prior $k - 1$ management period. In other words, it is the best plume forecast that can be made because it assimilates all of the available data from the prior period into the EnKF forecasts. In Equations 7.10 and 7.11, the actual observations are exploited from period $k - 1$ to estimate how the proposed sampling plan $x_{\omega,k}$ in period $k$ may cause deviations from the best known reference plume’s moments. Only those points that are being proposed for sampling in period $k$ and that have actual observations in $k - 1$ are used to develop the alternative mass and centroid forecasts $\mu_0,t$ and $\mu_1,t$. The Mass objective shown in Equation 7.10 seeks to minimize the maximum relative error (across each of the quarterly sampling periods) between the mass moments of the reference forecast, $\mu'_0,t$, and the subset forecast, $\mu_0,t$. This objective is expressed as a percentage error between the forecasted mass of the plume using all available prior information and the mass of the plume forecasted using a subset of the prior information based on a proposed sampling plan $x_{\omega,k}$. In Equation 7.10, the objective seeks to minimize the maximum error in forecasting the Mass of the plume across each of the quarterly sampling periods $t$, for the $k$th management period. In other words, it is minimizing the worst performing sub-sampling period’s Mass error and consequently ensuring all other periods are as good or better.

Minimize: $f_0(x_{\omega,k}) = \max_{1 \leq t \leq T} \left( \frac{\mu'_0,t - \mu_0,t}{\mu'_0,t} \right) \times 100\% \quad (7.10)$

Equation 7.11 seeks to minimize the maximum distance error (across each of the quarterly sampling periods) between the first plume moments (in X, Y, and Z) of the reference forecast, $\mu'_1,t$, and the subset forecast, $\mu_1,t$. This objective is expressed as a distance (in cm) between the centroid of the plume forecast using all prior information in period $k - 1$ and the centroid of the plume forecast using
a subset of the prior information in period \( k \) based on a sampling plan \( x_{\omega,k} \). This objective seeks to minimize the maximum error in forecasting the Centroid of the plume across each of the quarterly sampling periods for a management period. In other words, it is minimizing the worst performing sub-sampling period’s Centroid error and consequently ensuring all other periods are as good or better.

Minimize: 
\[
    f_1(x_{\omega,k}) = \max_{1 \leq t \leq T} \left[ (\mu'_{1X,t} - \mu_{1X,t})^2 + (\mu'_{1Y,t} - \mu_{1Y,t})^2 + (\mu'_{1Z,t} - \mu_{1Z,t})^2 \right]^{1/2} \tag{7.11}
\]

The Mass and Centroid objectives can be thought of as redundancy based objectives where points are removed that have the least effect on the EnKF’s ability to forecast the Mass and Centroid of the plume. While this objective would drive the optimization toward sampling fewer points, the DF and Flux objectives drive the optimization toward sampling more locations, and thus counter-balance one another.

### 7.5 Computational Experiment

#### 7.5.1 Flow and Transport Using ParFlow and SLIM

The ammonium chloride tracer experiment was modeled using the parallel numerical groundwater flow model ParFlow and the numerical particle transport model SLIM-FAST. The modeling domain was established in ParFlow using grid cells of approximately 7 cm on each side. This resulted in a tank domain containing 39 cells in the X direction, 54 cells in the Y direction, and 29 cells in the Z direction for a total of 61,074 grid cells. No flow boundaries were set at the bottom, left, and right sides of the tank and Dirichlet constant head boundary conditions were set to 203.2 cm at the front of the tank and to 200.6 cm at the rear of the tank to simulate the constant head inlet and outlet reservoirs respectively. Since the tracer test flow was saturated, a no flow boundary was set at the top of the tank at elevation 200.7 cm to force the numerical flow domain to remain saturated. In addition, a vertical injection well was placed in the flow domain to correspond with the location of the screen located at well B4. Water was then injected through this
Table 7.1. Summary of media properties used to simulate permeability field realizations. Hydraulic conductivity values (as well as their corresponding standard deviations) are expressed in units of cm-hr$^{-1}$.

<table>
<thead>
<tr>
<th>Media</th>
<th>(K)</th>
<th>(\sigma_K)</th>
<th>(\phi)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Layer 1</td>
<td>Course Sand</td>
<td>558.33</td>
<td>89.31</td>
</tr>
<tr>
<td>Layer 2</td>
<td>Silt</td>
<td>0.0001</td>
<td>0.00</td>
</tr>
<tr>
<td>Layer 3</td>
<td>Med. Sand</td>
<td>72.31</td>
<td>6.93</td>
</tr>
<tr>
<td>Layer 4</td>
<td>Med. Sand</td>
<td>75.77</td>
<td>4.33</td>
</tr>
<tr>
<td>Layer 5</td>
<td>Med. Sand</td>
<td>58.93</td>
<td>5.88</td>
</tr>
<tr>
<td>Lens</td>
<td>Fine Sand</td>
<td>45.38</td>
<td>8.61</td>
</tr>
</tbody>
</table>

location at a rate of 1.5 L-hr$^{-1}$ for the first 15 days to simulate the injection of tracer solution at this location. No injection occurred at this location during days 15-19 of the simulation. Total simulation time was set to 19 days to correspond with the full duration of the tracer experiment. Hence, steady state flow conditions existed during days 0-15 of the simulation during the tracer injection stage, and a second set of steady flow conditions existed for days 15-19.

Uncertainty in characterizing subsurface permeability was accounted for by simulating an ensemble of 100 flow fields, each resulting from a single realization of the subsurface permeability field. Turning Bands (TB) simulation [215] was used to generate realizations of the permeability field. To accomplish this, separate sub-domains were established within the full model domain to represent each of the five media layers and the fine sand lens. Hydraulic conductivity data collected at each of the screened sampling ports was then used to develop mean hydraulic conductivity values (as well as their corresponding standard deviations) to use in conditionally simulating each media layer. A summary of the hydraulic conductivities and porosities used in each media layer is shown in Table 7.1. Additional parameters consistent across all media layers relevant to the TB simulation that were used based on literature recommendations given by [215] included X, Y, and Z correlation lengths of 30 cm, 100 simulation lines, a line process resolution of 5.0, a maximum normalized frequency of 100.0, and a normalized frequency increment of 0.2. All media were assumed to be isotropic. After attaining an ensemble of flow field realizations from ParFlow, SLIM-FAST was then used to simulate the movement of ammonium chloride tracer through the UVM tank. The computa-
tional grid utilized by SLIM-FAST was set up identically to that used by ParFlow. Two stress periods were then set up to correspond to the release of ammonium chloride tracer, one for days 0 through 15 during the injection of tracer and one for days 15 through 20 where no tracer was injected. Velocity fields obtained from ParFlow for each of these two stress periods and for each of the 100 realizations were then made available to SLIM-FAST for the purposes of particle transport. To simulate the injection of tracer during stress period 1, a total of 540 g of ammonium chloride was released from the two grid cells (270 g-cell$^{-1}$) that corresponded with the screened location at well B4 over the 15-day release period. This 540 g of ammonium chloride was represented by the release 150,000 particles from these two grid cells evenly over the 15 day period.

Since a finite number of particles are used to represent solute concentration, numerical effects can become an issue at low solute concentrations. SLIM-FAST allows a concentration threshold to be set below which particles are split in half to better quantify regions of low solute concentration. The concentration threshold was set to $10^{-6}$ g-m$^{-3}$, and the maximum number of particles that could be used (due to particle splitting) was set to 500,000. Since 150,000 particles were initially set to be injected during the 15 day stress period, each particle had a mass of 0.0036 g. However, since some particles may be split during the simulation, individual particle mass can change. No particles were injected during days 15 through 20 (stress period 2). Global time step increments were set to 1 hour. However, SLIM-FAST calculates locally optimized time steps within the global time step for each individual particle based on the velocity field, dispersion, diffusion, etc. Other relevant parameters include longitudinal and transverse dispersivities of 1.44 cm and 0.25 cm respectively and a molecular diffusivity of 0.0024 cm$^2$. Upon completion of the SLIM-FAST simulation runs, 100 realizations of the ammonium chloride tracer plume were available for subsequent use in the optimization framework (as was shown to be effective in Chapter 6 [71]).

7.5.2 Parameterization of EnKF

The EnKF was implemented within the ε-hBOA to provide the forecasted design objectives’ values for each sampling plan evaluated by the algorithm. An ensemble
size of 100 members was used corresponding to the 100 realizations of the flow and transport modeling generated prior to optimization. EnKF forecasts were provided across a filtering subgrid that matched the flow and transport model domain in extents, but used cells that were 18 cm on each side (as opposed to 7 cm cells used for the modeling). The subgrid was composed of 11 cells in the X direction, 19 cells in the Y direction, and 7 cells in the Z direction for a total of 1463 cells (or \( n = 1463 \) filtered model states). The filtering subgrid resolution was chosen to provide a balance between computational efficiency and forecast quality. Filter forecasts were provided at a monthly frequency while data (potential well sampling plans) was assimilated at a quarterly frequency.

Measurement uncertainty was modeled by adding uncorrelated Gaussian noise to measurements with zero mean and 5 percent standard deviation. The spatially correlated Gaussian noise fields \( Q \) applied to the model and bias states were generated using the TB simulation and were assumed to have a zero mean, a 7.5 percent standard deviation, and an isotropic correlation length of 60 cm. The time correlation of the model bias, \( \Lambda \), was set to 99 percent. The EnKF formulation and parametrization for the UVM test case is drawn from that used in Chapter 6.

### 7.5.3 Parameterization of Many-Objective Search

Optimization runs were conducted on the Texas Advanced Computing Center’s (TACC) Ranger Sun Constellation Linux Cluster (http://www.tacc.utexas.edu/). The TACC Ranger system is composed of 3936 nodes, where each node contains four AMD Opteron Quad-Core processors yielding a total of 62,976 processing cores. The \( \varepsilon \)-hBOA was parameterized to be run on 1024 processing cores simultaneously using a Master-Slave paradigm \([91, 170]\). Memory availability per processing core (2 GB of DDR2) was a primary driver in selection of the \( \varepsilon \)-hBOA population size of 4096 individuals (as was the selection of EnKF grid resolution). This population size also ensured that approximately four solution evaluations would be conducted on each processing core within one generation of evolution. The ASSIST framework has the potential for scaled use from small scale desktop computing to emerging Petascale computing systems.

Five random seed trials were conducted using one-million solution evaluations
per seed. The \( \varepsilon \)-dominance precision values of the algorithm were set to \( \text{Cost} = 1.0 \), \( K = 0.5 \), \( DF = 0.1 \), \( \text{Flux} = 0.5 \), \( \text{Mass} = 0.5 \), and \( \text{Centroid} = 0.1 \) based on our precision goals for each objective (see Chapter 3 for a detailed discussion of \( \varepsilon \)-dominance and its interplay with computational scaling). The resulting \( \varepsilon \)-nondominated solutions from all five random seed trials were then combined using offline \( \varepsilon \)-nondomination sorting to produce the Pareto approximate forecast solution set analyzed throughout the remainder of this chapter.

7.6 Results and Discussion

7.6.1 The Forecasted Observation Design Tradeoffs

The forecasted six-objective trade-offs for the next management period are shown in the Pareto-approximate set in Figure 7.2. This set contains 7,613 solutions and was created using \( \varepsilon \)-nondomination sorting [111] to combine the results of all five random seed trial runs. Recall that \( \text{Cost}, \text{DF}, \text{Mass}, \) and \( \text{Centroid} \) are minimization objectives and \( \text{Flux} \) and \( K \) are maximization objectives. The \( \text{Cost}, \text{DF}, \) and \( K \) objectives are plotted on the X, Y, and Z axes of Figure 7.2 respectively. The \( \text{Flux} \) objective is shown by the color of the cones where blue cones represent low \( \text{Flux} \) detection and red cones represent high \( \text{Flux} \) detection. The \( \text{Mass} \) objective is shown in Figure 7.2 using the orientation of the cones. Cones that point up represent high \( \text{Mass} \) error and cones that point down represent low \( \text{Mass} \) error. The \( \text{Centroid} \) objective is shown using the size of the cones where large cones represent high \( \text{Centroid} \) error and small cones represent low \( \text{Centroid} \) error. The green arrows on each axis of Figure 7.2 are meant to direct the viewer towards optimality in each objective. An ideal solution would thus be located toward the rear lower corner (low \( \text{Cost} \), low \( \text{DF} \), and high \( K \)) of the plot in Figure 7.2 and would be represented by a small (low \( \text{Centroid} \) error), red (high \( \text{Flux} \) detection) cone pointing down (low \( \text{Mass} \) error).

Interestingly, given the very limited number of function evaluations (6 million) relative to the total size of the decision space (\( 7.2 \times 10^{75} \) possible designs), the many-objective optimization identified non-dominated solutions throughout a wide range of \( \text{Cost} \) where the maximum value in the set still represents a 32 percent cost
Figure 7.2. The resulting forecasted six-objective Pareto-approximate set generated by the $\varepsilon$-hBOA. Cost, DF, and $K$ objectives are plotted on the X, Y, and Z axes respectively. The Flux objective is shown by the color of the cones, Mass by the orientation of the cones, and Centroid by the size of the cones as is indicated in the legend.

savings (i.e., 170 samples taken out of the 252 total possible over the four sampling periods in years 6 through 7). This also indicates that higher cost solutions ($Cost > 170$) are sub-optimal in terms of the six objective formulation used in this work. Forecasted performance in the $DF$ objective ranged from 8 percent to 95 percent. The Flux objective ranged from 0.0 to 7.6 m-day$^{-1}$ and the Mass error objective ranged from 0.05 percent to 39 percent. The Centroid error objective ranged from 0.20 cm to 14 cm. The $\varepsilon$-hBOA was allowed to fully explore the potential tradeoffs that exist between the objectives because it is difficult for a decision maker to know the marginal returns for increased sampling costs a priori.

Generally, Figure 7.2 demonstrates that with increasing Cost, DF decreases
and $\text{Flux}$ detection increases as one would expect. Trends in the $\text{Mass}$ and $\text{Centroid}$ error objectives are less apparent, indicating widely varying performance in these objectives for each level of sampling $\text{Cost}$. These two objectives are highly dependent on where samples are taken spatially in each sampling period and require a limited number of samples in a few key locations to very closely quantify the mass and centroid of the tracer plume.

More interesting however, is the trend visible in the $K$ objective. Figure 7.2 shows that as more money is spent on sampling, it becomes harder and harder to attain nondominated solutions that have high information content (i.e., high $K$). The high $\text{Cost}$ solutions tend to be more consistent in their information content as is evidenced by the decreasing variability in $K$ with increasing $\text{Cost}$. Lower $\text{Cost}$ solutions are associated with a wider range of information content. This issue highlights the benefits and the importance of the many-objective formulation. Prior Kalman filtering studies $[52, 54, 55, 69]$ have focused solely on minimizing the filter’s estimation variance (similar to the $K$ objective). These studies show that often very few samples are necessary to attain near optimal Kalman filter variances. Focusing solely on this statistical objective would have the impact of degrading physical objectives such as $DF$ and $\text{Flux}$. Moreover, the results in Figure 7.2 show that although it is possible to attain high information content solutions in low cost regions, the dearth of observations could yield highly negative consequences that would be unknown to a decision maker using formulations with fewer objectives.

### 7.6.2 Exploring Lower Dimensional Problems

To further illustrate objective interactions and the potential consequences of lower dimensional formulations, Figure 7.3 shows projections of the six objective solution set onto a sampling of the two objective subproblems where again, the arrows indicate which objectives are minimized or maximized. Solving the full six objective problem is analogous to having solved all 6 single objective problems, 15 two objective problems, 20 three objective problems, 15 four objective problems, and 6 five objective problems. In summary, solving the six objective problem is therefore equivalent to solving 62 smaller subproblems simultaneously. The decision maker can hence proceed through a process of discovery and negotiation by analyzing
Figure 7.3. Projections of the full six objective solution set onto a sampling of two objective subproblems. Tradeoffs between the two objective subproblems are highlighted with a solid red line. Two interesting breakpoint solutions are annotated using stars. The locations of low Mass and Centroid error are highlighted in each subproblem using a light blue circle/ellipse. An interesting Cost threshold is annotated using a dashed magenta line.

subproblems of reduced complexity, and then use this information to move into the more complex tradeoffs revealed by the full six objective problem.

In Figure 7.3, panel A shows the Cost versus DF sub-problem where the two-objective tradeoff that would have resulted if only Cost and DF were considered is highlighted by the red solid line in the plot. Note the rather clear break in slope of this tradeoff that occurs at Cost = 46 representing a 29.4 percent DF (marked by a star in panel A). Panel B shows the Cost versus Flux subset problem where the tradeoff between these two objectives is highlighted by the red solid line. Again, there is a very clear break in this tradeoff at Cost = 40 (marked by a star in panel B). However, this solution is associated with a DF of 37.5 percent which substantially exceeds that of the prior solution selected from the Cost versus DF tradeoff. These breakpoints in the two-objective tradeoffs would traditionally be considered areas of focus for decision makers as they represent points of diminishing return. Interestingly, if the K objective is also considered, the prior breakpoint solutions contribute minimally to the corrective capability of the EnKF (i.e., they are located at the lower bound of the K objective as shown by the stars in Panel E). The selection of either solution yields very different decisions,
the first emphasizing $DF$ while the other emphasizes $Flux$ detection, but neither emphasizes filter performance.

The $Cost$ versus $Mass$ and $Centroid$ subproblems are shown in panels C and D, respectively. The low $Cost$ regions of panels C and D reveal a large variation in the $Mass$ and $Centroid$ objective values. This further highlights the observation made in reference to Figure 7.2 with regard to low $Cost$ designs achieving a wide range of performance in the $Mass$ and $Centroid$ objectives. Recall that these objectives are calculated in the prior management period $(k-1)$ by applying a potential sampling design to the prior management period and examining how this design would degrade the mass and centroid moments attained relative to the actual design that was implemented. Many prior LTGM design studies have focused on one or more plume moment based design objectives [40, 62–64, 167]. This prior work has shown that fairly accurate characterization of plume moments can be achieved with relatively few monitoring wells. These results similarly show that the EnKF is capable of providing reasonable forecasts of the plume’s zeroth and first moments by sampling from key locations in space and time (i.e., there are many low $Cost$ solutions that achieve low $Mass$ and $Centroid$ error). However, many of these seemingly high performing solutions (highlighted within a light blue circle in panels C and D) would fail critically in other objectives such as contaminant detection (the light blue ellipse in panel A) and flux detection (the light blue ellipse in panel B), again underlining the importance of quantifying and exploring many objectives simultaneously.

Panels E through G of Figure 7.3 show three interesting subproblems that plot $Cost$, $DF$, and $Flux$ against the information objective $K$. In panel E, the tradeoff (red solid line) associated with $Cost$ and $K$ would clearly mislead the decision maker into choosing a low $Cost$, high $K$ solution that exhibits strong covariance minimizing capability as has been typically recommended in prior literature. However, seeing the full six-objective problem in Figure 7.2 reveals that this would be an inferior strategy. While low $Cost$ solutions exhibit a wide range of EnKF covariance minimizing capability, as $Cost$ increases the variability in the $K$ objective decreases substantially. This indicates that sampling from more locations has the potential to dramatically improve the stability of EnKF forecasts. In addition, visualizing the structure of this relationship reveals a dramatic improvement in
**Figure 7.4.** Identification of five interesting solutions from the full six-objective Pareto-approximate forecast set. This figure represents the possible exploratory analysis that a decision maker might undergo to identify interesting compromise solutions that capture multiple system properties simultaneously.

EnKF stability around \( \text{Cost} = 95 \) (shown as a magenta dashed line), indicating a physical space-time sampling threshold above which the performance of the EnKF benefits dramatically. Interestingly, this \( \text{Cost} = 95 \) threshold also corresponds relatively well with the improvement in EnKF stability achieved at the \( \text{DF} \) threshold of about 20 to 25 percent visible in panel F (the magenta dashed line) and at a \( \text{Flux} \) threshold of about 6 to 7 m·d\(^{-1}\) visible in panel G (the magenta dashed line). Additionally, the low \( \text{Cost} \), low \( \text{Mass} \) and \( \text{Centroid} \) error solutions marked in panels E through G (highlighted by the light blue ellipses), further emphasizes the failure of these objectives to reliably achieve good performance throughout the remaining objectives.

Finally, panel H of Figure 7.3 shows \( \text{Flux} \) versus \( \text{DF} \) and reveals that these two objectives are highly correlated with one another. If Cost was eliminated as an objective, panel H shows that the optimization problem would collapse to the single
optimal solution (circled in red) that corresponds to the maximum Cost. Below the maximum Cost, tradeoffs do exist for the corresponding Flux that can be attained at each level of DF. Panel H also highlights the very poor performance in DF and Flux of the seemingly good Cost versus Mass and Centroid solutions from panel D (highlighted in the circular light blue region). The Cost = 95 threshold is indicated in panel H as well, again indicating that solutions resulting in high EnKF stability perform well in the plume focused DF and Flux objectives. All of the observations noted throughout the discussions of the sub-problems shown in panels A through H of Figure 7.3 provide valuable insights regarding the structure and interactions of the six design objectives chosen and represent important knowledge discoveries for informed decision making.

7.6.3 Exploration to Inform Decision Making

Panels A through E of Figure 7.4 present the possible steps that a decision maker might take to identify interesting compromise solutions that perform well in multiple design objectives simultaneously. Panel A shows the Cost versus Mass tradeoff highlighted in blue that would have been found had only these two objectives been used in the formulation (similar to Reed et al. [40]). Solution 1 identified in panel A represents a solution occurring at the point of diminishing return.

Moving to panel B (where the Cost-Flux tradeoff is highlighted in green) demonstrates the inability of solution 1 to provide meaningful physical quantification of the tracer plume. The Cost-Mass tradeoff is shown as well in panel B for reference purposes (highlighted in blue). Solution 2 identified in panel B represents where the decision maker might now go to better capture the Flux of the system as this solution represents a point of diminishing return on the Cost-Flux tradeoff. The Cost-DF tradeoff highlighted in red is shown in panel C with the Cost-Flux and Cost-Mass tradeoffs also shown. Since the Cost-DF and Cost-Flux tradeoffs appear to be closely correlated, and hence the points of diminishing return on these tradeoffs are correlated, a decision maker could be inclined to select a higher Cost solution (solution 3) that improves the DF and Flux detection over that achieved by solution 2. The objective values of solutions 1, 2, and 3 are shown in Table 7.2. Comparison of the objective values associated with these three solutions reveals
Figure 7.5. Sampling plans associated with solutions 2 and 5 identified in Section 7.6.3. Breakthrough curves are shown for the prior management period \(k-1\) years 5 through 6 and the forecast period \(k\) years 6 through 7. Breakthrough curves are shown similarly to Figure 7.1, except now the assimilation and forecast curve of the EnKF is shown using a dash-dot red curve. The decision of whether or not to sample at each quarter is indicated by either a filled or open circle respectively. Note that the time scale shown in this figure represents years 5 through 7 of the simulation.

that increasing sampling cost (solutions 2 and 3) achieves improved performance in the \(DF\) and \(Flux\) objectives, but may degrade performance in the \(Mass\), \(Centroid\),
Table 7.2. Five interesting compromise solutions that might be selected by a decision maker during exploration of the six objective Pareto-approximate forecast set.

<table>
<thead>
<tr>
<th>Solution</th>
<th>Cost</th>
<th>DF</th>
<th>Flux</th>
<th>Mass</th>
<th>Centroid</th>
<th>$K$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>8</td>
<td>100.0</td>
<td>0.000</td>
<td>0.866</td>
<td>4.775</td>
<td>14.282</td>
</tr>
<tr>
<td>2</td>
<td>40</td>
<td>37.50</td>
<td>7.044</td>
<td>32.56</td>
<td>7.497</td>
<td>6.674</td>
</tr>
<tr>
<td>3</td>
<td>83</td>
<td>19.37</td>
<td>7.281</td>
<td>7.876</td>
<td>3.461</td>
<td>9.283</td>
</tr>
<tr>
<td>4</td>
<td>84</td>
<td>63.19</td>
<td>4.097</td>
<td>0.392</td>
<td>1.411</td>
<td>12.170</td>
</tr>
<tr>
<td>5</td>
<td>152</td>
<td>8.18</td>
<td>7.568</td>
<td>0.455</td>
<td>1.612</td>
<td>10.633</td>
</tr>
</tbody>
</table>

and $K$ objectives.

Panel D shows the projection of the six objective solution set onto the Cost-$K$ subproblem with all prior identified tradeoffs highlighted and the prior three solution selections marked. Here we see that the Cost-Mass tradeoff solutions lie entirely in regions of low Cost and low EnKF stability (i.e., a region where $K$ varies widely). In addition, the tradeoffs associated with the Cost-DF and Cost-Flux sub-problems border the lower bound of $K$ performance throughout the full range of Cost. Solution 4 has a very similar Cost to solution 3, but provides a dramatically higher impact on the information $K$ provides to the EnKF forecasts. The objective values associated with solution 4 are also shown in Table 7.2. While solution 4 does tend to greatly improve performance in the Mass and Centroid objectives, its performance in terms of DF and Flux fails critically. This failure is made obvious in panel E where the positioning and color of solution 4 reveals its modes of failure. Panels D and E demonstrate that a higher cost solution (solution 5) still provides significant cost savings while maintaining an overall balanced compromise in the other objectives (see Table 7.2). This high cost solution still represents a 40 percent cost savings relative to the Cost = 252 case where every available port is sampled quarterly throughout the management period.

7.6.4 The Costs of Compromise

Figure 7.5 shows the sampling plans associated with the low Cost (solution 2) and high Cost (solution 5) plans respectively. The breakthrough curves associated with the layer 4 sampling ports are shown in Figure 7.5 similarly to Figure 7.1 except that now, the EnKF assimilation and forecast curve is shown using a dash-dot
Figure 7.6. Snapshots of tracer plume at year 7 (the end of the current management period). Panel A shows the plume snapshot obtained by assimilating all available samples in the prior and current management period (a maximum information baseline). Panel B shows the plume obtained by the biased flow and transport model. Panel C shows the plume obtained by assimilating available samples from the prior management period into the EnKF and then forecasting only for the current management period. Panel D shows the plume obtained by assimilating that data that would be available from the sampling plan associated with solution 5. Panels E through F show the relative error between the plumes shown on panels B through D and the maximum data plume shown in panel A where red indicates high error and grey, low error.

The prior \( k - 1 \) management period associated with years 5 through 6 assimilated all available data at all sampling locations. For management period \( k \) representing years 6 through 7, the EnKF is used to provide forecasts of alternative sampling strategies’ objectives during this period (i.e., no observations are assimilated). Data assimilation creates corrections that move the EnKF forecast toward the observations taken in \( k - 1 \) as can be noted in the figure during the
prior management period.

Although the observation breakthrough curves are shown for both the prior and forecast management periods, they are strictly plotted for reference purposes during the forecast period and they were not used in the forecast period $k$ to evaluate predicted tradeoffs (as would be the case in a real world application). The quarterly sampling times of the forecast period are denoted by dashed lines in the breakthrough plots along with an open circle indicating that no sample is taken at that quarter, and a filled circle indicating the decision to sample at a given quarter. Recall that the tracer was released from port B4 for a period of 15 years so the $k−1$ and $k$ management periods shown reflect a snapshot during mid-release of the tracer. Most interesting in Figure 7.5 is the observation that solution 2 shown in panel A represents a minimally sampled plan that perfectly tracks high flux tracer locations in space and time. This result is striking given the enormous size of the decision space ($2^{252}$ possible sampling plans) relative to the search time provided (5 million evaluations).

One notable issue with the sampling plan provided by solution 2 is its failure to sample from location P4, especially late in the management period where the EnKF is predicting the potential for the leading edge of the plume to be in this area. Solution 2’s emphasis on Cost minimization represents a very low level of risk aversion with respect to the DF and Flux objectives. This failure to sample at the leading edge of the plume represents a major issue of concern with solution 2 as future management period decisions would only have the modeled tracer at this location to draw from (i.e., no observations would be available at this location in management period $k$). This result is a major innovation to monitoring practices where the space-time consequences of adding or removing observations are largely unknown. Focusing now on solution 5 shown in panel B reveals that a compromise to invest in a higher Cost solution yields significant improvements in the DF and Flux objectives. In solution 5, the later two quarters of the management period at location P4 are now sampled. This solution also tends to sample locations beyond the boundary of the tracer plume, which reduces the errors in forecasting the mass and centroid of the tracer plume as well.

Figure 7.6 shows snapshots of the tracer plume at year 7 resulting from (A) a hypothetical maximum information case where all 252 available samples are
assimilated into the ENKF, (B) using the strongly biased flow and transport model only, (C) using only the data from the prior management period \(k - 1\) to generate an EnKF forecast of the plume in the present period \(k\), and (D) assimilation of the sampling plan data provided by solution 5 into the EnKF in the present management period (i.e., the actual implementation of our decision). Panel A represents the optimal plume snapshot that can be obtained at the end of the current management period given the use of all available data at a maximum cost. Comparing panels A and B demonstrates the failure of the original biased model to capture key characteristics of the tracer plume. The biased simulation fails to predict the downward migration of tracer through the fine sand lens and consistently underestimates high tracer concentrations.

Panel C of Figure 7.6 shows the EnKF forecasted plume at year 7 obtained using only the available data from the prior \(k - 1\) management period (years 5 through 6). It is clear that even prior to the availability of data for the current management period, the EnKF is providing a more accurate estimate of tracer concentration and movement than that provided by the model. This demonstrates the significant benefits of bias modeling within the EnKF, allowing it to more accurately forecast tracer movement, even in the presence of systematic and severe model errors. The EnKF forecast error shown in panel F further supports this observation as it is much improved over the error resulting from using the model alone shown in panel B.

Panel D of Figure 7.6 shows the plume that would result from assimilating all of the data at the sampling ports chosen by solution 5 identified in Section 7.6.3 (see Table 7.2). This represents the result that would occur if the decision maker picked solution 5 from the forecasted tradeoffs in Figure 7.4 Panel E and implemented it. It is clear that the plume obtained using solution 5’s sampling strategy matches very closely with the maximally sampled plume shown in panel A, but at a Cost savings of about 40 percent. In addition, panel G shows that the error associated with implementing solution 5 is minimal.
7.7 Conclusions

This chapter introduced the highly adaptable ASSIST (Adaptive Strategies for Sampling in Space and Time) framework for improving LTGM design. The ASSIST framework combines Monte Carlo flow and transport simulation, bias-aware ensemble Kalman Filtering, many objective search using hierarchical Bayesian optimization, and decision support tools based on interactive high dimensional visual analytics. The new framework was demonstrated using a physical aquifer tracer experiment where the position and frequency of tracer sampling was optimized for six design objectives that combined a variety of methodologies historically utilized by the LTGM research community.

Throughout this chapter, the ASSIST framework was demonstrated as a highly adaptable methodology for improving LTGM decisions across space and time while accounting for the influences of systematic model errors (or predictive bias). In addition, this chapter illustrates the ASSIST framework’s ability to facilitate discovery and negotiation throughout the LTGM design process. Specifically, the many-objective solution set identified using the ASSIST framework elucidated relationships between the tracer plume’s fluxes, moments, and boundaries while advancing decision makers’ understanding of the potential consequences of their monitoring decisions. The many-objective observation system design approach provides a mechanism to discover system dependencies and tradeoffs that would have been missed using traditional one and two objective problem formulations.

The results provide an illustrative example of the process of discovery and negotiation that decision makers would be equipped to pursue using the tools available in ASSIST. Using this example, five interesting sampling plans were easily identified from a suite of over 7600 Pareto approximate design alternatives. From these five, one sampling plan was identified that achieved a cost savings of 40 percent over sampling from all available locations while also providing excellent performance in the remaining five design objectives. The information provided by this sampling plan and how it differed minimally from the information provided by sampling from all available locations was also shown. This high performing design would have been extremely difficult to identify using prior published LTGM design approaches. An important contribution of this work is the ASSIST framework’s
ability to overcome predictive bias to improve our understanding of the space-time benefits and impacts of our environmental monitoring strategies. This work is an explicit example of how many objective Pareto efficiency provides an integrated value of information measure that directly links observations and decisions, which has been a long recognized need and challenge in water resources research literature (see Moss [15]). Consequently, the ASSIST framework has a strong potential to innovate our characterization, prediction, and management of water resources systems.
CHAPTER 8

Conclusions, Contributions, and Future Work

8.1 Conclusions

Chapter 3 demonstrated the scaling challenges that many-objective LTGM design problems pose, and attempted to address these challenges by presenting a way to carefully control the precision at which design objectives are quantified during the multiobjective evolutionary algorithm (MOEA) search process. Although it was shown that a current state-of-the-art MOEA, the Epsilon-nondominated Sorted Genetic Algorithm II (\(\varepsilon\)-NSGAII) required quadratic scaling to adequately solve a set of LTGM design test cases, \(\varepsilon\)-dominance could be utilized to obtain a close approximation to the Pareto set at the same time achieving close to linear computational scaling. While this represented a significant advance in the size and scope of problems that could be solved using MOEAs, it was recognized that a more robust algorithm would be required to begin solving larger LTGM design problems that incorporate both space and time in their sampling decisions.

Further improvements to the solution of LTGM design problems were presented in Chapter 4 through the introduction of a new, more robust search algorithm, the Epsilon-dominance Hierarchical Bayesian Optimization algorithm (\(\varepsilon\)-hBOA), that was capable of learning the complex interdependencies that exist between LTGM design decisions. The benefits of Bayesian network model building became apparent when the \(\varepsilon\)-hBOA was able to quickly outperform the \(\varepsilon\)-NSGAII at solving a relatively small 25 well LTGM design problem. In addition, the performance of the
ε-hBOA far exceeded that of the ε-NSGAII at solving a much larger LTGM design problem in terms of the number of sampling decisions. This demonstrated very strong evidence that complex interdependencies do in fact exist between LTGM sampling decisions, and that the ε-hBOA was able to successfully exploit this information to more efficiently search for high performing designs. This study showed that a combination of ε-dominance and hierarchical Bayesian network model building produces a potent new class of MOEA that has the potential to solve much larger LTGM design problems than have traditionally been considered.

Advancing our ability to evolve high quality many-objective LTGM tradeoffs is not fully sufficient to advance our ability to design high quality LTGM networks. The decision maker must also be able to efficiently search the large solution sets that typically result from many-objective optimization of LTGM design. Chapter 5 presented the Visually Interactive Decision-making and Design using Evolutionary Multiobjective Optimization (VIDEO) framework that enabled improved decision maker understanding of the relationships between their design objectives, ultimately allowing them to efficiently choose one or more solutions of interest. This chapter showed that the combined and effective use of a variety of high dimensional visualization and interaction capabilities allowed the decision maker to ultimately choose a single high performing design from a suite of over 2400 design possibilities. The concepts of VIDEO were then utilized throughout the remaining chapters to inform decision making in the context of much larger, more complex LTGM design problems.

Bias-aware ensemble Kalman filtering (EnKF) was presented in Chapter 6 as a highly effective tool to improve our ability to combine highly biased model predictions with uncertain measurement data to produce high quality forecasts of contaminant location and extent. The bias-aware EnKF was demonstrated on a carefully controlled physical aquifer experimental tracer flow and transport test case with highly biased initial conditions. This work showed that the EnKF was able to provide significantly improved forecasts of the tracer plume’s location and extent over that of the model alone. Additionally, the bias modeling capability of the EnKF eliminated the need to reinitialize the flow-and-transport model simulations upon assimilation of data measurements, greatly reducing the computational requirements of utilizing the EnKF within an LTGM optimization framework.
Chapter 7 combined all of the tools presented in previous chapters to create the highly adaptable ASSIST (Adaptive Strategies for Sampling in Space and Time) framework for improving LTGM design. The ASSIST framework comprehensively combined Monte Carlo flow-and-transport simulation, bias-aware ensemble Kalman Filtering, many-objective search using hierarchical Bayesian optimization, and interactive high dimensional visual analytics. The new framework was demonstrated using a physical aquifer tracer experiment where the position and frequency of tracer sampling was optimized for six design objectives that combined a variety of methodologies historically utilized by the LTGM research community.

This chapter showed that the ASSIST framework successfully facilitated discovery and negotiation throughout the LTGM design process. Specifically, the many-objective solution set identified using the ASSIST framework provided a broad range of systems behaviors that aided the decision maker in more fully understanding both modeled and unmodeled objectives, ultimately allowing them to identify system dependencies and tradeoffs that would have been missed using traditional one and two objective problem formulations. Following the process of discovery and negotiation that a decision maker would be equipped to pursue using ASSIST, five interesting sampling plans were identified from a suite of over 7600 design alternatives. From these five, one sampling plan that achieved a cost savings of 40 percent over sampling from all available locations while also providing excellent performance in the remaining design objectives was selected. This high performing design would have been extremely difficult to identify using traditional LTGM design approaches.

8.2 Contributions

The work in this dissertation represents a series of contributions to the field of LTGM design that have the strong potential to aid in advancing our ability to characterize, predict, and manage groundwater systems. Chapter 3 contributed the first detailed assessment of how increasing LTGM problem size impacts the ability of traditional MOEAs to solve these types of combinatorial design problems, but also provided a means of addressing the severe growth in computational complexity through the use of $\varepsilon$-dominance. Chapter 4 contributed a new MOEA
that utilized both \( \varepsilon \)-dominance and hierarchical Bayesian network model building to learn the complex interdependencies that exist between LTGM design decisions. Chapter 5 then addressed the decision making aspect of many-objective LTGM design by contributing a new high-dimensional visual analytics framework for enabling improved decision maker understanding of LTGM design problems. Chapter 6 presented a bias-aware EnKF capable of integrating both highly biased flow-and-transport modeling simulations with uncertain measurement data to greatly improve our ability to forecast the location and extent of groundwater contamination in LTGM design problems.

The contributions from the studies presented in Chapters 3 through 6 were then combined to create the highly adaptable ASSIST framework presented in Chapter 7. A primary contribution of the ASSIST framework is its ability to overcome predictive bias to improve our understanding of the space-time benefits and impacts of our environmental monitoring strategies. This work is an explicit example of how many-objective optimization provides an integrated value of information measure that directly links observations and decisions, an issue of paramount importance as noted by Moss [15].

Many of the advances made in this dissertation are currently being used both within and beyond the water resources discipline. Kasprzyk et al. [227] recently demonstrated the benefits of many-objective analysis using the \( \varepsilon \)-NSGAII and many of the visual analytical tools used in this work for water portfolio planning. Beyond water resources, Shah and Reed [225] recently demonstrated the benefits of \( \varepsilon \)-dominance and Bayesian network model building for optimizing D-dimensional knapsack problems, a very difficult class of combinatorial problem that has real-world analogues across a variety of disciplines, including LTGM network design. Within the aerospace community, the \( \varepsilon \)-NSGAII is being effectively utilized to advance satellite constellation design [170] and airline scheduling.

### 8.3 Future Work

Exploring the information contained within the Bayesian network models generated by the \( \varepsilon \)-hBOA will be an area of focus in the near future. The models generated by the \( \varepsilon \)-hBOA have tremendous potential to inform decision making.
However, learning how to effectively distill the information contained in what are very often highly complex models containing hierarchical relationships will be an important part of this work. In addition, the complexity of the space-time LTGM optimization problem also adds to the difficulty of interpreting the information contained in these models.

Figure 8.1 shows a hierarchical Bayesian network model for the 25 well LTGM test case utilized in Chapters 3 through 5. This relatively small real-world design problem was chosen as a benchmark test case to demonstrate how the Bayesian network models resulting from the $\varepsilon$-hBOA optimization runs might be saved and used to inform decision making. For example, panel A in the figure shows conditional relationships (or rules) that exist between the 25 sampling locations that occur in the model with near 100-percent probability. For example, the rule labeled 1 in panel A means that if well 1 is sampled, well 11 should also be sampled. Likewise, the rule labeled 2 means that if well 14 is not sampled, then well 15 should be sampled.

These relationships are shown in panel B in the context of the actual positioning of the sampling locations. Rule 1 may be meaningful to a decision maker because it is saying that if the source well 1 is sampled, then well 11 should also be sampled. Well 11 is important in this case because it defines a plume boundary. Likewise, rule 2 says that if well 14 is not sampled, then well 15 should be sampled, likely in an attempt to define the plume boundary. Understanding the information contained in the models generated by the $\varepsilon$-hBOA and distilling “rules” from these models has the potential to be an invaluable resource to a decision maker.

The Kalman gain maximization objective presented in Chapter 7 has the potential to provide a great deal of information regarding the information feedback of individual sampling decisions to the variance minimizing capabilities of the EnKF. Figure 8.2 provides an example of how this information is used by the EnKF for the UVM tank experiment used in Chapters 6 and 7, and how the information could potentially be used to inform decision making.

Panel A in Figure 8.2 shows the relative corrective magnitude that sampling from well B4 in the fourth quarter of the current management period has on the entire sampling domain. While it is apparent that sampling from well B4 predominantly impacts the immediate vicinity of the well (the green colored cloud
Figure 8.1. This figure demonstrates how sampling rules might be developed from the hierarchical Bayesian network models that are generated by the \( \varepsilon \)-hBOA algorithm during optimization of LGTM design problems. Panel A shows high probability sampling rules for the 25 well LTGM test case explored in Chapters 3 through 5. Panel B shows how these rules might be applied to inform both optimization and decision making. For example, if well 18 is not chosen for sampling, then well 20 should be chosen.

surrounding B4), it would not be obvious that sampling from this location impacts EnKF performance in regions of the sampling domain near the leading edge of the tracer plume (see the green colored clouds at the opposite end of the tank from B4). Additionally, panel B shows similar impact characteristics when sampling from well I4. Interestingly however, panel C shows that sampling from well S4 at the leading edge of the tracer plume has a profound impact throughout that region of the tank, even extending its influence to regions close to well B4.

To the author’s knowledge, this is the first time that the information contained in the Kalman gain matrix has been visually distilled to provide a physically mean-
Figure 8.2. Example of how the information available in the Kalman gain matrix might be used to inform decision making for the UVM tank experiment (from Chapters 6 and 7. Shaded regions in the plots show the impacts that individual sampling decisions have on the performance of the EnKF. For example, in panel A, sampling from well B4 impacts the performance of the EnKF in the immediate vicinity of this well, but also effects its performance at the opposite end of the tank near the leading edge of the tracer plume.

A meaningful representation of the impact that individual sampling decisions have on the performance of the EnKF. This type of information also has the potential to aid in
furthering our understanding of the complex inter-relationships between sampling
decisions and will be an active area of future research as a means to further inform
space-time sampling decisions.


Vita
Joshua Brian Kollat

Education:
• 2003-2005. M.S. in Civil Engineering. The Pennsylvania State University, University Park, PA.
• 1998-2003. B.E. in Civil/Environmental Engineering. Youngstown State University, Youngstown, OH.

Experience:
• 2008-Present. Engineer II. AECOM Water. State College, PA.

Recent Publications: