Parameter and State Model Reduction for Bayesian Statistical Inverse Problems

by

Chad Eric Lieberman

B.S. Aerospace Engineering (2007)
B.S. Mathematics (2007)
Massachusetts Institute of Technology

Submitted to the School of Engineering
in partial fulfillment of the requirements for the degree of

Master of Science in Computation for Design and Optimization

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

February 2009

Author .....
School of Engineering
February 1, 2009

Certified by .....
Karen E. Willcox
Associate Professor of Aeronautics and Astronautics
Thesis Supervisor

Accepted by .....
Jaime Peraire
Professor of Aeronautics and Astronautics
Director, Computation for Design and Optimization

Parameter and State Model Reduction for Bayesian Statistical Inverse Problems

by

Chad Eric Lieberman

Submitted to the School of Engineering
on February 1, 2009, in partial fulfillment of the
requirements for the degree of
Master of Science in Computation for Design and Optimization

Abstract

Decisions based on single-point estimates of uncertain parameters neglect regions of significant probability. We consider a paradigm based on decision-making under uncertainty including three steps: identification of parametric probability by solution of the statistical inverse problem, propagation of that uncertainty through complex models, and solution of the resulting stochastic or robust mathematical programs.

In this thesis we consider the first of these steps, solution of the statistical inverse problem, for partial differential equations (PDEs) parameterized by field quantities. When these field variables and forward models are discretized, the resulting system is high-dimensional in both parameter and state space. The system is therefore expensive to solve. The statistical inverse problem is one of Bayesian inference. With assumption on prior belief about the form of the parameter and an assignment of normal error in sensor measurements, we derive the solution to the statistical inverse problem analytically, up to a constant of proportionality. The parametric probability density, or posterior, depends implicitly on the parameter through the forward model. In order to understand the distribution in parameter space, we must sample. Markov chain Monte Carlo (MCMC) sampling provides a method by which a random walk is constructed through parameter space. By following a few simple rules, the random walk converges to the posterior distribution and the resulting samples represent draws from that distribution. This set of samples from the posterior can be used to approximate its moments.

In the multi-query setting, it is computationally intractable to utilize the full-order forward model to perform the posterior evaluations required in the MCMC sampling process. Instead, we implement a novel reduced-order model which reduces in parameter and state. The reduced bases are generated by greedy sampling. We iteratively sample the field in parameter space which maximizes the error in full-order and current reduced-order model outputs. The parameter is added to its basis and then a high-fidelity forward model is solved for the state, which is then added to the state basis. The reduction in state accelerates posterior evaluation while the reduction in parameter allows the MCMC sampling to be conducted with a simpler, non-adaptive

Metropolis-Hastings algorithm. In contrast, the full-order parameter space is high-dimensional and requires more expensive adaptive methods. We demonstrate for the groundwater inverse problem in 1-D and 2-D that the reduced-order implementation produces accurate results with a factor of three speed up even for the model problems of dimension $N \approx 500$. Our complexity analysis demonstrates that the same approach applied to the large-scale models of interest (e.g. $N > 10^4$) results in a speed up of three orders of magnitude.

Thesis Supervisor: Karen E. Willcox
Title: Associate Professor of Aeronautics and Astronautics

Acknowledgements

My most sincere gratitude is due to my advisor, Professor Karen Willcox, for her steadfast support, guidance, and advice throughout my research career. I appreciate greatly her ability to point me in the right direction when I was stuck. Whenever it seemed I had tried everything and nothing would work, Karen always had another suggestion. It was in Karen's class that I first learned about computational methods and developed an appreciation for computer solutions to physical problems. I also owe her a debt of gratitude for opening my eyes to the world of academic research, which I had little appreciation for before a UROP that introduced me to model reduction.

My family has also lent tremendous support during my post-graduate work. I would like to thank my Mom for her words of encouragement especially during the times when research was slow and nothing would seem to work out. Thanks to my Dad for always believing in me and for many good sports-related conversations which served as a much-needed refuge from a sometimes intense stretch of work. I cannot thank enough Kyle and Kristina for their invitations to play tennis, board games, pool, and video games as well as the many times I have enjoyed a home-cooked meal away from home because of their hospitality. I cannot forget, of course, Mackenzie for greeting me always with a wagging tail when I returned home.

I owe the members of the Alpha Mu Chapter of Phi Kappa Sigma Fraternity for keeping me grounded and not letting me forget the most important parts of life. I would also like to thank the Alumni Corporation and the MIT Student Life Office for allowing me the opportunity to serve as Resident Advisor for the house. Even in just one semester I have already learned a lot about advising students both academically and socially.

I am also appreciative of the opportunity to work with several esteemed members of the computational science and engineering community including Professor Omar Ghattas and Professor Youssef Marzouk. I look forward to continued collaboration in the coming years.

Contents

List of Figures

List of Tables

Nomenclature

α	acceptance probability
$\bar{\mathbf{p}}$	parameter sample mean
$\bar{p}_i$	mean of the ith parameter
β	regularization parameter
$\boldsymbol{\Gamma}$	proposal covariance matrix
$\boldsymbol{\lambda}$	full-order adjoint variable
$\boldsymbol{\mu}$	reduced-order adjoint variable
δA	control volume boundary
η	transition probability
Γ	computational domain boundary
Γ_D	Dirichlet boundary
Γ_N	Neumann boundary
$\gamma_{\mathbf{p}}$	prior probability density
κ	transition kernel
$\mathbf{A}$	discrete forward operator
$\mathbf{A}_r$	reduced forward operator

$\mathbf{C}$ discrete observation operator

$\mathbf{f}$ state equation residual

$\mathbf{M}$ mass matrix

$\mathbf{P}$ reduced parameter basis

$\mathbf{p}$ parameter vector

$\mathbf{p}_r$ reduced-order parameter

$\mathbf{q}$ vector of log hydraulic conductivity

$\mathbf{S}$ prior covariance

$\mathbf{s}$ right-hand side

$\mathbf{u}$ vector of state variables

$\mathbf{u}_r$ reduced-order state

$\mathbf{V}$ reduced state basis

$\mathbf{y}$ output vector

$\mathbf{y}_d$ data

$\mathbf{y}_r$ reduced-order model outputs

$\mathcal{D}$ parameter domain

$\mathcal{J}$ objective function

$\mathcal{M}$ forward model map

$\mathcal{N}(\mathbf{z}, \mathbf{\Sigma})$ multivariate normal distribution with mean $\mathbf{z}$ and covariance $\mathbf{\Sigma}$

$\mathcal{T}$ finite element triangulation

$\mathcal{U}(\mathcal{D})$ uniform distribution over $\mathcal{D}$

$\mathcal{L}$ Lagrangian functional

∇ gradient operator

Ω computational domain

$\phi(\vec{x})$ piecewise linear nodal basis function

$\pi_{\mathbf{p}}$ posterior probability density

$\mathbb{P}$ space of piecewise linear polynomials

σ standard deviation

$\mathrm{var}(p_i)$ variance of the ith parameter

$\vec{n}$ outward-pointing unit normal

$\vec{x}$ coordinate vector

ξ candidate generating kernel

d domain dimension

dA differential area

dv differential arc length

$f(\vec{x})$ source field

$h(\vec{x})$ boundary flux

L likelihood function

N state degrees of freedom

n number of reduced states

N^D number of state Dirichlet conditions

N_o number of outputs

N_p parameter degrees of freedom

n_p number of reduced parameters

N_s number of samples

$p(\vec{x})$ parameter field

$q(\vec{x})$ log hydraulic conductivity field

$r(v)$ regularization functional

$u(\vec{x})$ pressure head field

$v(\vec{x})$ finite element test function

Chapter 1

Introduction

Inverse problems arise in many fields of science and engineering. Medical imaging [2, 53, 3, 26], geophysics [58, 18, 15, 45], heat transfer [59, 63, 72], quantum, wave, and electromagnetic scattering [27, 22, 21], and electrical impedance tomography [10, 47, 43] are a few examples. In each of these fields, we have models used to solve the *forward problem*. Given a description of the physics and the parametric inputs, we compute a set of outputs of interest. In many cases, the underlying model used to solve the forward problem represents a discretization of a set of partial differential equations (PDEs). Consequently, a forward model evaluation may be computationally costly. In contrast to the forward problem, the *inverse problem* represents one of inference on the parameters given observations or data. For example, in the case of heat transfer, one forward problem might consist of determining the temperature distribution across a component subject to a heat source given the spatially-varying thermal conductivity. The solution of the inverse problem, on the other hand, utilizes readings of temperature on the surface of the component to predict the thermal conductivity throughout.

There are several ways to formulate inverse problems. Two popular, but distinct, approaches are deterministic optimization and statistical inference. In the deterministic setting, the inverse problem is posed as a mathematical program. The objective function is usually constructed such that a normed difference between the data and model output is minimized. Regularization will be required in most cases to gener-

ate a unique solution. The solution is one point in the parametric input space. No estimate of uncertainty in that solution is generally available. The statistical inference problem is formulated by casting the parameter as a random variable, a process consistent with the Bayesian approach. Given normality assumptions on the output error and inclusion of prior knowledge of the parameter, a probability distribution over the parametric input space is obtained by leveraging Bayes's rule. In contrast to the deterministic solution, the solution of the statistical inference problem is one or more statistical estimators applied to an ensemble of parameters. Furthermore, the uncertainty in those estimates is quantified directly via the ensemble itself.

In this work, we study the solution of statistical inverse problems for models with many states and high-dimensional parametric input spaces but with relatively few outputs of interest. Markov-chain Monte Carlo (MCMC) sampling requires at least one forward model solution for each member of the ensemble. For the problems of interest, repeated forward model solution is a prohibitive endeavor. We restrict our attention to models which are linear in state and whose forward operator is linear in the parameter. For these models, we can bring to bear goal-oriented model reduction algorithms to make tractable an otherwise nearly impossible inference problem. Using a greedy approach, we solve a sequence of optimization problems to obtain the parameters at which the error between the reduced- and full-order models is maximal. The parameter and state bases are created directly from these samples. Projection of the state equations onto the state basis and of the forward operator onto the parameter basis produces a reduced-order model that is inexpensive to evaluate when compared to the full-order instantiation. This increase in forward solution efficiency makes tractable MCMC sampling to solve the statistical inverse problem.

In Section 1.1, we motivate the need for uncertainty quantification (UQ) in large-scale engineering inverse problems. Section 1.2 provides additional background on inverse problems, UQ, and model reduction. In Section 1.3, the thesis objectives are outlined. Finally, Section 1.4 provides a roadmap for the remainder of the work.

1.1 Motivation

In this section, we demonstrate the need for uncertainty quantification in the solution of science and engineering inverse problems. We consider the computational requirements of the solution of the statistical inverse problem thereby motivating the use of model reduction.

Uncertainty quantification is an integral component of engineering design. Whether the design be a bridge, aircraft, car, skyscraper, or oil-drilling plan, engineers must account for uncertainty in manufacturing, construction, and applied loads. In a jet engine, for example, there is manufacturing variability in the blades of a turbine blade row. The uncertainty in this manufactured part must be propagated through a model to estimate failure probabilities or to design maintenance plans. With a description of the uncertainty in part or all of the components of an assembly, an engineer can attempt to minimize the risk of failure or maximize the service duration. Stochastic or robust optimization might be utilized to design under this uncertainty. Using stochastic optimization, a company may derive a production plan based on maximizing an expected return subject to the uncertainty in supply costs. In robust optimization, a component may be produced to eliminate risk altogether.

Independent of field, scientists and engineers benefit from knowledge of the uncertainty in design parameters. Many engineering inverse problems are solved to infer parameters of a design component or an environmental factor. In the statistical inverse problem setting, solutions to these problems consist of parameter estimates as well as quantification of the uncertainty in those estimates. Engineers are then able to utilize the uncertainty in that component to properly design a complete model. Inverse problem solutions may also play a role in optimal control problems. In contaminant transport, the inverse problem is one of obtaining the initial distribution of the contaminant. A forward prediction and attempt to mitigate the effects by neutralizing reagent or flow control could be posed as a problem in optimal control. The resulting mathematical program is subject to the uncertainty in the inverse problem solution. The benefit of the statistical inverse formulation is that the quantification

of uncertainty necessarily accompanies the solution.

Markov chain Monte Carlo (MCMC) simulation has become a very popular method for sampling from complex probability distributions. In particular, it is utilized frequently in the solution of statistical inverse problems to generate samples from the posterior. The *posterior* distribution is that which characterizes the probability that a certain parameter, subject to prior knowledge about the parameter space, produces the given data when propagated through the forward model. In fact, the posterior is essentially the solution to the inverse problem; that is, if the parametric input space could reasonably be enumerated and the associated forward problems solved, then one could at least represent the posterior numerically.[1] In general, this will not be a tractable solution because the high-dimensional parametric input space could not be explored exhaustively irrespective of the complexity of a forward model solution. Instead, MCMC provides a framework through which the posterior can be sampled indirectly.

By the construction of an appropriate transition kernel, a Markov chain can be established. If the Markov chain has certain properties (e.g., irreducibility and ergodicity) and its invariant measure is that of the posterior, then samples of the chain will eventually be realizations of the posterior. With the assumption that measurement error is normally distributed, the forward model appears in the posterior. Therefore, the posterior depends on the parameter implicitly. There is no direct method for evaluating moments of the posterior: numerical quadrature is infeasible in high dimensions. Using MCMC, we produce an ensemble of parameters distributed according to the posterior. Statistical estimators of the resulting sample set approximate moments of the posterior including the mean and variance. As the number of samples increases, we expect the estimates derived from the sample set to converge to the true moments of the posterior.

Each sample in the MCMC simulation requires at least one solution of the forward model. If the forward model represents complex physics, the computational cost of

[1]Note that pictorial representations of the posterior are limited to low-dimensional parametric input spaces. While we may revert to such examples to gain insight, the problems of interest usually do not satisfy this criterion.

MCMC becomes prohibitive. When the forward model represents the discretization of a set of PDEs which are linear in state, we can reduce the complexity of a forward solve by finding a reduced-order model. A reduced-order model is a surrogate for the high-fidelity model which adequately captures the input-output maps of interest but is significantly cheaper to evaluate. If the map from parametric inputs to outputs inhabits a low-dimensional subspace of the state space, it stands to reason that a reduced-order model may capture the input-output relations of interest while reducing the computational complexity of a forward solve. With the reduced-order model, we can produce accurate forward solutions at low cost; therefore, we can utilize MCMC on some statistical inverse problems which would otherwise be intractable.

Next, we provide additional background on UQ, the solution of large-scale inverse problems, and model reduction. We provide several useful references for current research in these areas.

1.2 Background

We now provide the necessary background in uncertainty quantification (UQ), inverse problems, and model reduction. Epistemic and aleatoric uncertainty are defined. We focus on UQ related to design under uncertainty. Inverse problem theory is explored. The ill-posedness of inverse problems in the sense of Hadamard is stated and explained. We then focus on the large-scale inverse problem and highlight state-of-the-art methodologies in both the deterministic and statistical settings. Finally, we highlight important work in model reduction. The most widely used approaches are cited and the projection-based framework is outlined.

1.2.1 Uncertainty quantification

There are two types of uncertainty to be considered. Epistemic uncertainty is that which follows from a lack of knowledge; aleatoric uncertainty results directly from inherent variability in the system [61]. In the Bayesian framework, both types of

uncertainty can be cast as random variables.[2] Any parameter which is not certain is assumed to vary randomly. Information about its distribution is incorporated through the prior. The need for uncertainty quantification is motivated by the umbrella of design-under-uncertainty: model calibration, engineering system design, and policy-making are three examples.

Model calibration refers to the tuning of model parameters to fit experimental data often for, but not limited to, computational models. In [44], Kennedy and O'Hagan utilize a Bayesian approach to calibrate model parameters of a complex black-box computer code. The formulation is similar to that employed in the current work; however, the parameter space is low-dimensional and therefore numerical quadrature can be used to integrate moments of the posterior. This methodology is extended and implemented in [55]. Model validation via uncertainty propagation is undertaken in [16]. The authors propose the use of surrogates via polynomial regression and kriging in cooperation with data transformations to efficiently propogate uncertainty through a complex forward model. The work of McFarland et al. in [56] compares nonlinear regression and Bayesian techniques for model calibration.

In engineering system design, uncertainties must be propagated from model inputs to model outputs. Uncertainty analysis is critical in the design phase: component and assembly testing are very expensive aspects of development. If non-robust designs can be eliminated early in the process, the prototyping phase may be much cheaper. In [17] Chen et al. outline the importance of global sensitivity analysis in design under uncertainty. In particular, the authors utilize surrogate models including kriging, radial basis functions, and polynomial approximants for computationally expensive simulations. Ierapetritou et al. formulate a general process engineering problem with stochastic parametric inputs in [41]. Four different types of uncertainty are identified and a multi-step decomposition method is presented for solving the resulting stochastic optimization problem. A thorough review of optimal control under uncertainty appears in [67].

Policy-making can also benefit from uncertainty quantification. The allocation

[2]The same is not true of the frequentist viewpoint.

of resources to projects can be posed as an optimization problem with uncertain parameters. For example, one may have a probabilistic sense of a project's cost and value. A company or government organization might wish to maximize the ratio of value to cost under the uncertainties quantified in the projects. In this manner, different policies can be explored pertaining to different assumptions on the inputs – and the associated uncertainty would be propagated throughout the analysis. Fiscal policy problems also require design under uncertainty. In [49], Levin et al. consider monetary policy and its robustness to uncertainty in pre-defined forecast parameters.

1.2.2 Inverse problems

We now turn to the theory of inverse problems and study the current practice for large-scale problems.

Hadamard outlined the concept of a well-posed problem in [38]. He stated that mathematical models of physical problems must have three properties: (1) existence, (2) uniqueness, and (3) continuous dependence of the solution on the data. By this notion of well-posedness, an inverse problem is inherently ill-posed: the limited observation of outputs often necessarily implies nonuniqueness.[3] This fact is the entrypoint for an entire field of regularization theory.

The main idea of regularization is to modify the problem to favor solutions of a certain type. The exact implementation is usually problem dependent; however, many versions of regularization have been studied and implemented. Therefore, there is an extensive base of experience from which to draw. In the problems of interest to this work, the solution space is a field variable. Suitable regularization schemes penalize the continuous norm of some combination of the derivatives of that field. Tikhonov regularization corresponds to penalizing the norm of the zeroth derivative, for example. The present work penalizes a continuous norm of the first derivative of the parameter by choosing a Gaussian prior.[4]

As stated above, problems of interest are usually solved in one of two ways: either

[3] In addition, inverse problems may also suffer from nonlocality and noncausality.

[4] The relation between the prior and regularization schemes is explored further in Appendix A.

deterministically or stochastically. The deterministic inverse problem solution generally corresponds to the solution of an optimization problem over a high-dimensional design space. In contrast, the statistical inverse problem requires many forward solutions. The computational cost of the statistical approach depends on two parts: the cost of a forward solve and the number of realizations of the posterior required to obtain the desired convergence.

In the deterministic setting, the inverse problem is formulated as a mathematical program constrained by the forward model. In that case, the solution methodology coincides with that of a PDE-constrained optimization problem with high-dimensional design space. Biegler et al. present a reduced Hessian SQP algorithm in [8] and demonstrate its numerical performance in [9] on a problem with as many as 517 parameters. For problems with many variables, memory requirements can be constraining. In [51], Liu et al. propose a limited memory BFGS method with which they solve problems with as many as 10 000 variables. Recently, Haber demonstrated that one can take advantage of the special structure of a problem to speed convergence of quasi-Newton methods [37]. Solution of larger problems requires massively parallel implementation and supercomputing resources. Akcelik et al. perform an inversion over 17 million parameters in 29 minutes and also solve a problem with 139 billion spatiotemporal unknowns in under five hours with massively parallel algorithms [1].

Whereas our notion of a solution to the statistical inverse problem is characterized by moment estimators, one may prefer a maximum a posteriori (MAP) estimate. The MAP estimate is the point in parameter space which is most probable under the measure of the posterior. The formulation of the inverse problem still follows from Bayes's rule, but instead of estimating moments of the posterior, an optimization problem is solved to maximize it. The geostatistical approach [45] is a two-step process: structural analysis followed by model calibration. The goal is to ascertain the MAP estimate. When a gradient-based optimization method is used, the MAP estimate is subject to local maxima. Like the solution to the deterministic inverse problem, the solution omits quantification of uncertainty.

If uncertainty quantification is desired, the statistical inverse problem requires

many forward problem solutions. Those problems which arise from science and engineering may be solved with models representing a suitable discretization of a set of PDEs over a domain of interest. In some cases, a forward solution can be computationally expensive. Kaipio et al. recently proposed mesh-coarsening for solving the linear inverse problem [42]. In particular, they leverage the Bayesian formulation to quantify the statistics of the error associated with the coarser grid. The proposition is to utilize this knowledge to suitably reduce the dimensionality of the forward model, thereby achieving tractibility. Polynomial chaos expansions (PCE) have gained attention recently. In [54], Marzouk et al. propose the use of stochastic spectral methods to obtain a surrogate for the posterior which can be evaluated much more rapidly than the original (which necessarily incorporates the forward model). Najm highlights the use of PCE for UQ in [61].

As pointed out before, there are two primary sources of cost in the MCMC simulation. The first is the evaluation of the posterior. This challenge is addressed by surrogate methods including reduced-order models and PCE. The second cost is the number of realizations required to converge the estimator. Strategies to accelerate the convergence include adaptivity [66], delayed rejection [35], multiple-try methods [52], and Langevin proposals [6]. In delayed rejection methods, if a proposal is rejected, instead of remaining at the same point in parameter space, a new candidate is proposed. Reversibility of the Markov chain is preserved by adjusting the acceptance rate appropriately. Multiple-try methods draw several candidates at the outset of an iteration and select one to accept or reject by drawing randomly from the set of candidates according to a weighting function. In combination with local optimization, the authors of [52] demonstrate the speedup in convergence even for simple proposal distributions. Langevin proposals utilize gradient information to affect a hill-climbing shift on the proposal distribution. With appropriately selected step sizes, the Markov chain proceeds rapidly to the support of the posterior.

The current work uses a Metropolis-Hastings MCMC algorithm with delayed rejection and adaptive Metropolis [35] for full-order MCMC. Delayed rejection involves multiple proposals per iteration and improves mixing properties with local adaptiv-

ity. Adaptive Metropolis uses statistical estimators of the current chain to modify the proposal distribution to improve the acceptance rate.

1.2.3 Model reduction

Reduced-order models which accurately capture the important input-output relations of a large-scale high-fidelity model are required by the engineering community. Model reduction is applied primarily in two settings: real-time and multi-query. In both cases, the reduced-order model serves as a surrogate for the full-order model. In engineering applications, the full-order model arises from spatial and temporal discretization of a PDE. The resulting systems often consist of a large number of states, typically in the millions. For real-time applications, a reduced-order model is often required to obtain a forward solve in the given time limit. In the multi-query setting, e.g. in Bayesian inference, one wishes to repeatedly sample the high-dimensional parametric input space, thereby requiring many forward solves. While a full-order model could perform that task, the turn-around time is often far too long for it to be useful. A reduced-order model, however, could perform the same analysis within minutes.

Reduced-order models are generated by many methods: Krylov-subspace [28, 29, 33], balanced realization/truncation [60, 34, 50, 64, 69, 48], and proper orthogonal decomposition (POD) [23, 40, 68]. For the applications of interest to us, POD is the most widely used in large part due to its applicability to general parameterized systems. POD is a projection-based model reduction technique. For our problem, we seek two low-dimensional subspaces: one for the parameter and one for the state. A basis, often constructed from snapshots of the full-order variables, is then utilized to project the full-order equations to that subspace. In many applications, this reduced space can be very low-dimensional, capturing over 99% of the input/output relations with just tens of reduced state variables. The result is a reduced-order model whose computational complexity is dominated by factorization of a small dense matrix. In contrast, the full-order model requires solution of a linear system usually characterized by a large sparse matrix.

It is well known that the quality of the reduced-order model depends highly on the snapshots generated to construct the basis, yet there does not exist a generally accepted method for sampling to create those snapshots. One heuristic method used to produce the snapshots is the greedy sampling algorithm proposed in [71, 32] and utilized in [7, 11, 12]. This systematic approach attempts to generate the snapshots adaptively by iteratively sampling the point in parameter space at which the error between the full- and reduced-order models is maximal. Since no global optimization algorithms exist, however, it is only guaranteed to find local maxima. The greedy optimization problem for the application of steady porous media flow is derived in Section 3.3.

1.3 Thesis objectives

There are three primary objectives of this work:

- extend the greedy sampling procedure to high-dimensional parameter spaces (e.g. discretized field quantities);
- develop methodology for reduced-order modeling with order reduction in parameter and state spaces;
- solve a large-scale statistical inverse problem in reduced parameter space utilizing the reduced-order model for posterior evaluations.

We develop a novel approach to large-scale statistical inverse problem solutions by leveraging projection-based reduced-order modeling. The MCMC sampling process is conducted in the reduced parameter space where traditional Metropolis-Hastings algorithms are sufficient. Posterior evaluations are accelerated further by reduction in state. The parameter and state bases are simultaneously identified by an iterative greedy procedure. In problems of interest, the parameter is a field quantity whose discretization leads to a very high-dimensional input space. Our first objective seeks the extension of the greedy procedure to problems of this type. We address new challenges associated with the large-scale nonlinear programs which must be solved

at each iteration. Finally, we recast the statistical inverse problem in the reduced parameter space and compare the results with the full-order implementation.

In addition, the secondary objectives are:

- develop understanding for existing methods and challenges in high-dimensional Markov chain Monte Carlo sampling;

- identify potential extensions of current work including error estimation, parametric posterior representation, alternative model reduction methods, and adaptive sampling schemes.

In order to verify the reduced-order results, we must solve the large-scale statistical inverse problem with the high-fidelity model. Indeed, even for modestly-sized problems, this can be a very challenging task – one aspect of the motivation for the reduced-order method. An understanding of the existing adaptive MCMC sampling methods and remaining challenges is necessary to solve the full-order statistical inverse problem. With this further understanding, we propose several future research thrusts in adaptive sampling and the uncertainty quantification framework.

1.4 Roadmap

Chapter 2 presents the problem formulation and outlines the assumptions inherent in the remainder of the work. The application to the steady porous media flow equations is introduced and the governing equations derived. The Bayesian formulation of the statistical inverse problem is presented and then specified for the application in this work. We define the 1-D and 2-D groundwater inverse problems.

In Chapter 3 we develop the model reduction methodology. The concept of simultaneous reduction in parameter and state is introduced and justified for the application to the statistical inverse problem. The greedy sampling approach is explained in further detail. The optimality conditions for the sampling problem are derived for the porous media flow equations.

Chapter 4 focuses on Markov chain Monte Carlo sampling. The challenges of high-dimensional MCMC are highlighted through the exploration of traditional and adaptive methods. The delayed rejection adaptive Metropolis (DRAM) method is defined. We present two additional adaptive methodologies — Langevin sampling and componentwise adaptation — and comment on potential extensions based on techniques from large-scale optimization.

The results of the porous media flow inverse problem application are presented in Chapter 5. We solve the statistical inverse problem in 1-D and 2-D with the full- and reduced-order implementations. The results are compared and the computational cost evaluated. We provide a detailed complexity analysis to understand the extension to the typical application problem size.

We summarize the results in Chapter 6 and conclude the work with several suggestions for future research thrusts in the field.

Chapter 2

Inverse problem formulation and finite element discretization

In this chapter we describe the class of problems of interest through a discussion of properties of the discretized forward model. In Section 2.1, we outline the properties of the forward model corresponding to the steady groundwater flow equations. Once the forward model is established, the inverse problem is formulated in two ways. The deterministic formulation leads to a nonlinear program (NLP) over parameter space as we demonstrate in Section 2.2. Although we do not consider the solution to the deterministic inverse problem explicitly in this chapter, the formulation is noteworthy because of its close connection to the greedy sampling technique employed in the model reduction as described in Section 3.3. In Section 2.3, the statistical inverse problem formulation is derived through assumptions on the measurement error and prior knowledge about the parameter. Bayes's rule connects these assumptions to the posterior, the probability distribution considered the solution to the statistical inverse problem. While the problem definition up to this point is general, in Section 2.4 we introduce an application to groundwater flow. The governing equations are derived and discretized by Galerkin finite elements using a piecewise linear approximation. Boundary conditions are specified for model problems in one and two dimensions.

2.1 Forward model

We begin by defining the *forward model*.[1] For our purposes, the forward model is a map from parameter space to output space. Let $\mathbf{p} \in \mathbb{R}^{N_p}$ be a vector of parameters and the forward model $\mathcal{M} : \mathbb{R}^{N_p} \to \mathbb{R}^{N_o}$ compute outputs $\mathbf{y} \in \mathbb{R}^{N_o}$. The outputs are related to the parameters through a state $\mathbf{u} \in \mathbb{R}^{N}$. The forward problem consists of solving

$$\mathbf{f}(\mathbf{u}; \mathbf{p}) = \mathbf{0}, \tag{2.1}$$

$$\mathbf{y} = \mathbf{y}(\mathbf{u}), \tag{2.2}$$

the state and output equations, respectively, with $\mathbf{f} : \mathbb{R}^{N \times N_p} \to \mathbb{R}^{N}$. In the fully general form, $\mathbf{f}(\cdot)$ and $\mathbf{y}(\cdot)$ may be nonlinear functions. In the unsteady case, $\mathbf{f}(\cdot)$ and/or $\mathbf{y}(\cdot)$ may contain the state at a set of time steps for a finite time horizon. With some modification, the forthcoming methodology can be applied in the unsteady setting as well. We restrict our attention in this work to steady problems with the following characteristics:

(i) the state equation is linear in state,

(ii) the state equation is linear in parameter through the forward operator,

(iii) the output equation is linear in state,

(iv) and the number of parameters is much larger than the number of outputs.

Under these assumptions we write

$$\mathbf{A}(\mathbf{p})\mathbf{u} = \mathbf{s}, \tag{2.3}$$

$$\mathbf{y} = \mathbf{C}\mathbf{u} \tag{2.4}$$

where $\mathbf{A}(\mathbf{p}) \in \mathbb{R}^{N \times N}$ is the forward operator which depends linearly on $\mathbf{p}$, $\mathbf{s} \in \mathbb{R}^{N}$ is a source term which does not depend on state or parameter, and $\mathbf{C} \in \mathbb{R}^{N_o \times N}$

[1] During the course of the exposition, we may also refer to the *forward problem*, an evaluation of the outputs of the forward model given a specified input parameter.

is the observation operator. Although the equations are linear, the state depends nonlinearly on the parameter – a fact which contributes significantly to the challenges in solving the inverse problem. Note that (2.3)–(2.4) is a special case of (2.1)–(2.2). For the present work, the map of interest is that from parameter to output through state. While this relationship does depend on some hyperparameters (e.g. source and boundary conditions), these are assumed constant for the remainder of the work.[2] The applications of interest typically have $N, N_p > 10^6$ and $N_o < 10^2$, for example, when the finite element discretization is performed over a 3-D domain, the parameter is a field quantity defined over the mesh, and the outputs are state variables at nodes sparsely distributed throughout the domain.

We now turn to the inverse problem and outline the deterministic and statistical formulations. In contrast to the forward problem, the *inverse problem* uses data or observations to infer the underlying parameters. The deterministic inverse problem formulation attempts to minimize the misfit between data and model output subject to regularization. The statistical inverse problem formulation uses assumptions on the distribution of output error and *a priori* knowledge of the parameter distribution to determine the posterior, a probability distribution over the parameter space reflecting the incorporation of the data. In the following discussion, we utilize the general discrete equations (2.1)–(2.2) above.

2.2 Deterministic inverse problem

The deterministic inverse problem is formulated as a large-scale nonlinear program over the parameter space. Let $\mathcal{D}$ be the parametric space of interest. Additionally, let $r : \mathbb{R}^{N_p} \to \mathbb{R}$ be a regularization functional and $\beta \in \mathbb{R}_+$ be a regularization

[2]Throughout the text, we may relax the parametric specification for convenience. Unless otherwise specified, the reader may assume that $\mathbf{A} = \mathbf{A}(\mathbf{p})$, $\mathbf{u} = \mathbf{u}(\mathbf{p})$, and $\mathbf{y} = \mathbf{y}(\mathbf{u}(\mathbf{p}))$.

parameter. Then, $\mathbf{p}^*$ is the solution to the deterministic inverse problem if

$$\mathbf{p}^* = \arg\min_{\mathbf{p}\in\mathcal{D}} \mathcal{J} \;=\; \frac{1}{2}\|\mathbf{y}-\mathbf{y}_d\|^2 + \frac{1}{2}\beta r(\mathbf{p}), \tag{2.5}$$

$$\text{subject to} \quad \mathbf{f}(\mathbf{u};\mathbf{p}) \;=\; \mathbf{0}, \tag{2.6}$$

$$\mathbf{y} \;=\; \mathbf{y}(\mathbf{u}) \tag{2.7}$$

where $\mathbf{y}_d \in \mathbb{R}^{N_o}$ is the data and $\|\cdot\|$ is a suitably defined norm. The first term in the objective function $\mathcal{J}$ is a measure of the misfit between outputs corresponding to parameter $\mathbf{p}$ and the collected data $\mathbf{y}_d$. In the absence of measurement error, the first term vanishes for the parameter which produces the data. Without the regularization term, the inverse problem is ill-posed in the sense of Hadamard: There may be many parameters $\mathbf{p} \in \mathcal{D}^*$ where $\mathcal{D}^* = \{\mathbf{p} \mid \|\mathbf{y}(\mathbf{p}) - \mathbf{y}_d\| = 0\}$. When properly regularized, the inverse problem becomes well-posed provided that $r(\mathbf{p})$ is unique for all $\mathbf{p} \in \mathcal{D}^*$. In particular, the regularization functional provides a method by which solutions of a particular type may be penalized (or conversely, admitted). The regularization functional usually consists of a combination of $\mathbf{p}$ and its derivatives. Proper choice of regularization functional is both a science and an art. Indeed, regularization theory has become a field in and of itself. Although it is important to the solution of the deterministic inverse problem, we will not study regularization theory further here. A comprehensive background is provided in [25] for the interested reader.

Irrespective of the particular regularization scheme, the deterministic inverse problem is a PDE-constrained NLP. In general, we have no guarantees about the convexity of either the feasible set or the objective function. Coupled with the high-dimensionality of $\mathcal{D}$, these attributes yield a challenging optimization problem. In Chapter 3 we encounter a similar problem as a result of the formulation of the greedy sampling procedure for the model reduction. There, we apply a grid continuation technique in combination with a trust-region Newton method to solve the NLP. We delay detailed discussion until Section 3.3.

We note one disadvantage of the deterministic inverse problem formulation (2.5)-(2.7). The solution to the NLP is a single parameter $\mathbf{p}^* \in \mathcal{D}$. In particular, we have

no measure of the uncertainty in that solution. In contrast, uncertainty quantification follows naturally from the statistical formulation of the inverse problem.

2.3 Statistical inverse problem

In contrast to the single-point estimate of the deterministic formulation, the solution to the statistical inverse problem is a probability distribution over parameter space. Under reasonable assumptions, we obtain the posterior analytically. Unfortunately, the resulting expression depends implicitly on the parameter through the forward model. Therefore, it is difficult to glean any information of value from the analytical expression. In particular, we cannot obtain the moments of the posterior directly. Instead, we must approximate moments via Markov chain Monte Carlo (MCMC) sampling. We provide details on this matter in Chapter 4.

The statistical inverse problem formulation is an application of Bayesian inference. Let $\gamma_{\mathbf{p}}(\mathbf{p})$ be a probability distribution over parameter space $\mathcal{D}$ representing our *a priori* knowledge of the parameter. This distribution is appropriately termed the *prior*. Selection of the prior is closely related to the choice of regularization functional in the deterministic setting. In fact, precise links can be made when a maximum *a posteriori* (MAP) estimate is computed. When moments of the posterior are desired, the effect is not as clear. The prior is useful for incorporating problem-specific knowledge. In the absence of prior information, a uniform prior $\gamma_{\mathbf{p}}(\mathbf{p}) \sim \mathcal{U}(\mathcal{D})$ is often used. However, if any information is known *a priori*, it should be incorporated. The prior provides the tool to do so. In what follows, we select a Gaussian process prior,

$$\gamma_{\mathbf{p}}(\mathbf{p}) = \exp\left\{-\frac{1}{2}\beta\mathbf{p}^T\mathbf{S}^{-1}\mathbf{p}\right\} \tag{2.8}$$

where $\mathbf{S} \in \mathbb{R}^{N_p \times N_p}$ correlates points $\vec{x}_i$ and $\vec{x}_j$ in space such that $\mathbf{S} = \hat{\mathbf{S}} + c\mathbf{I}$ and

$$\hat{S}_{i,j} = a\exp\left\{\frac{-\|\vec{x}_i - \vec{x}_j\|_2^2}{2b^2}\right\}; \tag{2.9}$$

for scalar parameters a, b, and c. For some particular choices of prior (e.g., the

stiffness matrix), a direct link may be drawn between the prior of the statistical inverse problem and the regularization in the deterministic inverse problem. Please see Appendix A for further details.

The likelihood function establishes the relative probability of different measurement outcomes given a specified parameter. If we assume that the measurement is normally distributed with covariance $\sigma^2\mathbf{I}$, then we have the likelihood function

$$L(\mathbf{y}_d|\mathbf{p}) = \exp\left\{-\frac{1}{2\sigma^2}\|\mathbf{y}_d - \mathbf{y}(\mathbf{p})\|_2^2\right\}. \tag{2.10}$$

Given a parameter value $\mathbf{p}$, the likelihood function determines the relative probability of data $\mathbf{y}_d$. Here we assume $\mathbf{y}_d \sim \mathcal{N}(\mathbf{y}(\mathbf{p}), \sigma^2\mathbf{I})$; the data are normally distributed about $\mathbf{y}(\mathbf{p})$ with covariance $\sigma^2\mathbf{I}$. Note that the likelihood is not a probability distribution but instead is a function $L : \mathbb{R}^{N_p} \to \mathbb{R}$. Other choices of the likelihood are possible; however, we find this choice logical and convenient given the context of sensor measurement and our choice of prior.

The prior and the likelihood are connected to the posterior by Bayes's rule. The posterior is

$$\pi_{\mathbf{p}}(\mathbf{p}|\mathbf{y}_d) = \frac{L(\mathbf{y}_d|\mathbf{p})\gamma_{\mathbf{p}}(\mathbf{p})}{\int_{\mathcal{Y}_d} \pi_{\mathbf{y}_d}\, dY_d}. \tag{2.11}$$

The normalizing constant $c = \int_{\mathcal{Y}_d} \pi_{\mathbf{y}_d}\, dY_d$ is typically not known and may be as difficult to quantify as moments of the posterior itself. However, provided $c \neq 0$, the normalizing constant is of no particular interest: The relative probabilities of parameters is important. In particular, we write

$$\pi_{\mathbf{p}}(\mathbf{p}|\mathbf{y}_d) \propto L(\mathbf{y}_d|\mathbf{p})\gamma_{\mathbf{p}}(\mathbf{p}).$$

Under the assumptions of the prior knowledge of the parameter and a likelihood function, we obtain these relative probabilities in analytical form,

$$\pi_{\mathbf{p}}(\mathbf{p}|\mathbf{y}_d) \propto \exp\left\{-\frac{1}{2\sigma^2}\|\mathbf{y}_d - \mathbf{y}(\mathbf{p})\|_2^2 - \frac{1}{2}\beta\mathbf{p}^T\mathbf{S}^{-1}\mathbf{p}\right\}. \tag{2.12}$$

Technically speaking, this posterior distribution is the solution to the statistical inverse problem up to the normalizating constant. Unfortunately, useful aspects of the posterior are not evident from the analytics for two reasons: (1) the normalizing constant scales the distribution and can therefore have order of magnitude effects on the acquired probability for a single parameter and (2) the expression is implicit in the parameter through the forward model.

From the statistical formulation, we consider two courses by which to proceed. First, we define the maximum *a posteriori* (MAP) estimate,

$$\mathbf{p}^* = \arg\max_{\mathbf{p}\in\mathcal{D}} \pi_{\mathbf{p}}(\mathbf{p}|\mathbf{y}_d), \tag{2.13}$$

that parameter which maximizes the posterior. When the posterior has exponential form, the MAP estimate is obtained via large-scale optimization by maximizing $\log \pi_{\mathbf{p}}(\mathbf{p}|\mathbf{y}_d)$. Like the deterministic formulation, this method results in a single point estimate of the solution without natural quantification of the uncertainty and is subject to the same difficulties of all general NLPs.

On the other hand, we may inquire about the moments of the posterior. The first two moments are the mean and variance and are given componentwise by

$$\bar{p}_i = \int_{\mathcal{D}} p_i \pi_{\mathbf{p}}(\mathbf{p}|\mathbf{y}_d)\, d\mathcal{D}, \qquad \mathrm{var}(p_i) = \int_{\mathcal{D}} (p_i - \bar{p}_i)^2 \pi_{\mathbf{p}}(\mathbf{p}|\mathbf{y}_d)\, d\mathcal{D}, \qquad i = 1, 2, \ldots, N_p \tag{2.14}$$

respectively. Note that if the parameter space were low-dimensional, we could use numerical quadrature to approximate the normalizing constant, and in turn, the moments of (2.14). However, with $\mathcal{D} \subset \mathbb{R}^{N_p}$ and N_p large for problems of interest, numerical quadrature is intractible. Instead, we rely on discrete approximations to these integrals through an ensemble of parameters drawn from the posterior distribution.

Let $\mathcal{S} = \{\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_{N_s}\}$ be a set of N_s samples from the posterior distribution.

By the weak law of large numbers, for N_s large,

$$\bar{p_i} = \frac{1}{N_s}\sum_{j=1}^{N_s} p_i^j, \quad \text{var}(p_i) = \frac{1}{N_s - 1}\sum_{j=1}^{N_s}(p_i^j - \bar{p_i})^2, \quad i = 1, 2, \ldots, N_p$$

are accurate approximations of the mean and variance of the posterior, respectively. Here we use superscripts to denote the index of the sample in the ensemble and subscripts to denote the element of the parameter vector. In Chapter 4 we discuss how MCMC produces the set $\mathcal{S}$. We note here that the solution to the statistical inverse problem by MCMC leads to quantification of the mean and variance as well as estimates in the uncertainty in those solutions. It is this more complete probabilistic characterization of the parameter estimate which is consistent with the decision-making-under-uncertainty paradigm introduced in Section 1.1.

2.4 Steady flow through porous media

In this section, we introduce the application of our methodology to steady flow through porous media. We first develop the continuum equations via conservation of mass and Darcy's law. In Section 2.4.2 we develop the high-fidelity finite element model for solution of the steady groundwater flow equations in a domain with a suitable finite element mesh. In Section 2.4.3 and Section 2.4.4 the governing continuum and discretized equations are specified to 1-D and 2-D model problems which will be studied throughout the remainder of this work.

2.4.1 Derivation of the governing equations

Let $u(\vec{x})$ be the pressure head, $p(\vec{x})$ be the hydraulic conductivity, $\vec{q}(\vec{x})$ be the Darcy flux, and $f(\vec{x})$ be a source. Consider the control volume pictured in Figure 2-1. The control volume has differential area dA with boundary δA. By conservation of mass, the Darcy flux out of the control volume must balance the source inside. That is,

$$\oint_{\delta A} \vec{q} \cdot \vec{n}\, dv = \int\int_{dA} f\, dA \tag{2.15}$$

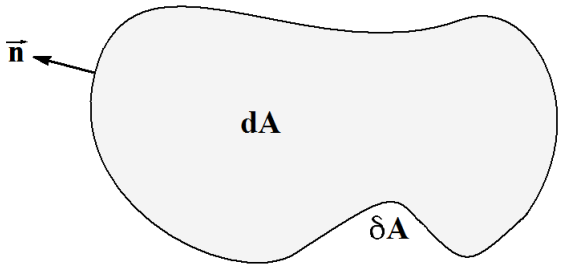


Figure 2-1: Control volume for derivation of the groundwater flow equations.

where $\vec{n}$ is the outward-pointing unit normal vector and dv is a differential arc along δA. Using Green's theorem, we convert the line integral over the boundary to an integral over the domain such that

$$\int\int_{dA} \nabla \cdot \vec{q}\, dA = \int\int_{dA} f\ dA \tag{2.16}$$

where we have introduced the divergence operator. For this equation to be satisfied for every differential element in domain Ω, we must have equality of the integrands

$$\nabla \cdot \vec{q} = f, \quad \vec{x} \in \Omega. \tag{2.17}$$

Darcy's law relates the pressure head and hydraulic conductivity to the Darcy flux by the relation

$$\vec{q} = -p\nabla u. \tag{2.18}$$

Substituting Darcy's law into (2.17), we arrive at the steady groundwater flow equation

$$-\nabla \cdot (p\nabla u) = f, \quad \vec{x} \in \Omega. \tag{2.19}$$

Note the connection to the heat equation. In particular, Darcy's law is analogous to Fourier's law. We make no assumption on the isotropy of the parameter p; it will be considered heterogeneous for the entirety of this exposition.

Let Γ_{D} and Γ_{N} divide the boundary of Ω into Dirichlet and Neumann boundaries respectively, i.e. $\Gamma_{\mathrm{D}} \cup \Gamma_{\mathrm{N}} = \Gamma$ and $\Gamma_{\mathrm{D}} \cap \Gamma_{\mathrm{N}} = \emptyset$. See Figure 2-2.

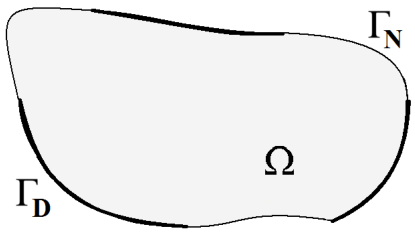


Figure 2-2: A general computational domain Ω with Neumann Γ_N and Dirichlet Γ_D boundary conditions.

The solution to (2.19) requires boundary conditions

$$p\nabla u \cdot \vec{n} = h \quad \text{on } \Gamma_N \qquad \text{and} \qquad u = g \quad \text{on } \Gamma_D. \tag{2.20}$$

On the Neumann boundary we specify the normal component of the Darcy flux. On the Dirichlet boundary, the pressure head is imposed.

2.4.2 Finite element model

The derivation of our high-fidelity finite element model begins with the variational formulation of the governing equations (2.19) along with boundary conditions (2.20). Then we introduce the discretization of Ω and set up the discrete linear system whose solution will be used to reconstruct the pressure head $u(\vec{x})$.

The variational formulation of the groundwater flow equation is derived by requiring (2.19) to be satisfied in the weak sense,

$$\int_\Omega v\left[-\nabla \cdot (p\nabla u) - f\right] \, d\Omega = 0 \tag{2.21}$$

for suitable test functions $v \in H_0^1(\Omega) \equiv \{v(\mathbf{x}) \mid \ v|_{\Gamma_D} = 0\}$. Then, (2.21) is integrated by parts to yield the variational form

$$\int_\Omega p\nabla u \cdot \nabla v \, d\Omega = \int_\Omega fv \, d\Omega + \int_{\Gamma_N} hv \, d\Gamma \qquad u \in X_D, \forall v \in H_0^1(\Omega), \tag{2.22}$$

where the prescribed flux h has been substituted and $X_D = \{u(\vec{x}) \in H^1(\Omega) \mid \ u|_{\Gamma_D} =$

$g\}$. Although we have introduced some regularity conditions through the restriction of the spaces in which the solution and test functions reside, the variational problem is still continuous. Next, we discretize by introducing a triangulation of the domain.

Let $\mathcal{T}$ be a finite element triangulation of Ω. We assume that $\mathcal{T}$ satisfies the regularity conditions of a finite element mesh and that the union of the closure of all simplices $T_i \in \mathcal{T}$ covers the closure of the domain, i.e. $\bar{\Omega} = \bigcup_{T_i \in \mathcal{T}} \bar{T}_i$. Let $X^{\mathrm{F}} = \{v \in H^1(\Omega) \mid v \in \mathbb{P}_1(T_i), \forall T_i \in \mathcal{T}\}$ be the finite element space.[3] Then consider the test space $X_0^{\mathrm{F}} = \{v \in X^{\mathrm{F}} \mid v|_{\Gamma_{\mathrm{D}}} = 0\}$ and trial space $X_D^{\mathrm{F}} = \{v \in X^{\mathrm{F}} \mid v|_{\Gamma_{\mathrm{D}}} = \tilde{g}\}$ where $\tilde{g}$ is the $\mathbb{P}_1$ approximation of g.

Given $u^F \in X_D^{\mathrm{F}}$, (2.22) becomes

$$\int_{\mathcal{T}} p^F \nabla u^F \cdot \nabla v^F \, d\Omega = \int_{\mathcal{T}} f v^F \, d\Omega + \int_{\Gamma_{\mathrm{N}}} h v^F \, d\Gamma, \qquad \forall v^F \in X_0^{\mathrm{F}}. \tag{2.23}$$

where $p^F \in X^{\mathrm{F}}$ is the $\mathbb{P}_1$ approximation of p. To continue the derivation of the discrete form, we introduce the piecewise linear nodal basis functions $\phi_i \in X^{\mathrm{F}}$ defined to be unity at node i and zero elsewhere. Let N be the number of nodes in $\Omega \backslash \Gamma_{\mathrm{D}}$ and N_{D} be the number of nodes lying on Γ_{D}. We make the following assumptions:

$$u^F = \sum_{j=1}^{N} \phi_j u_j + \sum_{j=N+1}^{N+N_{\mathrm{D}}} \phi_j g_j \qquad \text{and} \qquad p^F = \sum_{k=1}^{N_p} \phi_k p_k, \tag{2.24}$$

where g_j are the nodal values of the prescribed pressure head on Γ_{D}. Substituting (2.24) into (2.23), we find

$$\sum_{j=1}^{N} u_j \sum_{k=1}^{N_p} p_k \int_{\mathcal{T}} \phi_k \nabla \phi_j \cdot \nabla \phi_i \, d\Omega \;\; =$$
$$\int_{\mathcal{T}} f \phi_i \, d\Omega + \int_{\Gamma_{\mathrm{N}}} h \phi_i \, d\Gamma - \sum_{j=N+1}^{N+N_{\mathrm{D}}} g_j \sum_{k=1}^{N_p} p_k \int_{\mathcal{T}} \phi_k \nabla \phi_j \cdot \nabla \phi_i \, d\Omega, \quad \forall i = 1, 2, \ldots, N.$$

[3] Increased accuracy could be obtained by considering higher degree polynomial representations in Ω. Our concern in this work is not the accuracy of the high-fidelity model, but instead the accuracy of the reduced-order model to be developed in Chapter 3.

This formulation corresponds to the linear system

$$\mathbf{A}(\mathbf{p})\mathbf{u} = \mathbf{M}\mathbf{f} + \mathbf{H}\mathbf{h} - \mathbf{B}\mathbf{g}, \tag{2.25}$$

where $\mathbf{A}(\mathbf{p}) \in \mathbb{R}^{N\times N}$, $\mathbf{p} \in \mathbb{R}^{N_p}$, $\mathbf{u} \in \mathbb{R}^N$, $\mathbf{M} \in \mathbb{R}^{N\times N}$, $\mathbf{f} \in \mathbb{R}^N$, $\mathbf{H} \in \mathbb{R}^{N\times N}$, $\mathbf{B} \in \mathbb{R}^{N\times N_{\mathrm{D}}}$, and $\mathbf{g} \in \mathbb{R}^{N_{\mathrm{D}}}$ are defined to be

$$\mathbf{A}_{i,j} = \sum_{k=1}^{N_p} p_k \int_{\mathcal{T}} \phi_k \nabla\phi_j \cdot \nabla\phi_i \, d\Omega, \quad \mathbf{u} = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_N \end{bmatrix}, \quad \mathbf{M}_{i,j} = \int_{\mathcal{T}} \phi_j\phi_i \, d\Omega,$$

$$\mathbf{f} = \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_N \end{bmatrix}, \quad \mathbf{H}_{i,j} = \int_{\Gamma_{\mathrm{N}}} \phi_j\phi_i \, d\Gamma, \quad \mathbf{h} = \begin{bmatrix} h_1 \\ h_2 \\ \vdots \\ h_N \end{bmatrix},$$

$$\mathbf{B}_{i,j} = \sum_{k=1}^{N_p} p_k \int_{\mathcal{T}} \phi_k \nabla\phi_j \cdot \nabla\phi_i \, d\Omega, \quad \mathbf{g} = \begin{bmatrix} g_{N+1} \\ g_{N+2} \\ \vdots \\ g_{N+N_{\mathrm{D}}} \end{bmatrix}$$

where f_i is the nodal source value and h_i is the allocated flux to node i. Note that the right hand side term $-\mathbf{B}\mathbf{g}$ is a source arising from nonzero essential boundary conditions $g \neq 0$. Note that (2.25) is an example of (2.3) with $\mathbf{s} = \mathbf{M}\mathbf{f} + \mathbf{H}\mathbf{h} - \mathbf{B}\mathbf{g}$. We now briefly discuss properties of the state equation (2.25) and potential solution methods.

The forward operator $\mathbf{A}$ is symmetric in this Galerkin formulation. Its inverse is well-defined provided there is not a cluster of simplices in $\mathcal{T}$ over which $p = 0$. Note however that physical reality requires positive hydraulic conductivity $p > 0$, thereby implying positive definiteness of $\mathbf{A}$. If this positivity constraint is met, the state equation becomes a linear system defined by a symmetric positive-definite (SPD) matrix.

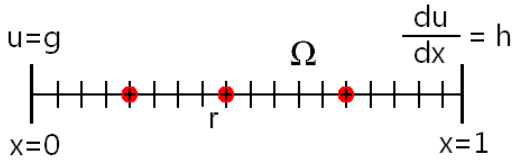


Figure 2-3: A sample uniform finite element mesh with mesh radius $h = 1/18$. Sensor nodes are indicated by red circles.

In preparation for our discussion of the model reduction methodology in Chapter 3, we now consider the division of the solution to this forward model into its homogeneous and particular solutions. Let $\mathbf{u}^0$ and $\mathbf{u}^P$ be the homogeneous and particular solutions respectively where $\mathbf{u}^0$ is the solution to (2.25) when $g = 0$ and $\mathbf{u}^P$ is the solution with nonzero essential boundary conditions but zero source $s = 0$ and zero flux $h = 0$. With these definitions, it is clear that $\mathbf{u} = \mathbf{u}^0 + \mathbf{u}^P$. Multiplying each side by $\mathbf{A}$ we have $\mathbf{Au} = \mathbf{Au}^0 + \mathbf{Au}^P = \mathbf{Ms} + \mathbf{Hh} - \mathbf{Bg}$. We show later that this division produces a reduced-order model whose error is invariant to scalings of p.

2.4.3 1-D model problem

In this section, we define the 1-D model problem used in Chapter 5 to demonstrate the efficacy of our reduced-order approach to the statistical inverse problem. Let the domain $\Omega = \{x \mid 0 < x < 1\}$ be the unit interval. The governing equations are

$$\begin{aligned} -\frac{d}{dx}\left(p\frac{du}{dx}\right) &= f \quad \text{in } \Omega, \\ \frac{du}{dx}(x{=}1) &= h, \\ u(x{=}0) &= g. \end{aligned}$$

Let $\mathcal{T}$ be a uniform mesh of triangles with mesh radius r such that there are a total of $N + N_{\mathrm{D}} = (1 + 1/r)^2$ nodes and $1/r$ elements in the mesh. See Figure 2-3. The outputs are given by the state at a subset of the nodes corresponding to sensors. For $i = 1, 2, \ldots, N_o$, we have $y_i(x) = u(x_i)$. The sensors are indicated in the figure by red circles. Typically we will equip approximately ten percent of the nodes with sensors.

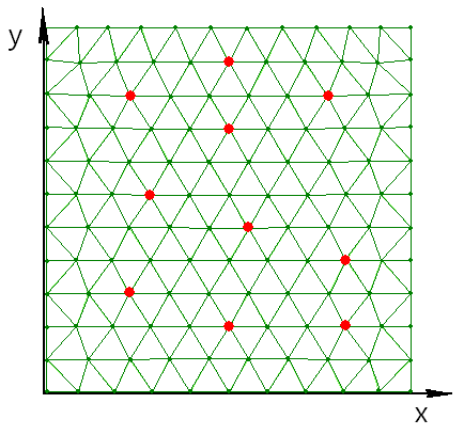


Figure 2-4: A sample finite element mesh with mesh radius $r = 0.1$ generated by `distmesh2d.m`, a finite element mesher developed by Professor Per-Olof Persson and Professor Gilbert Strang.

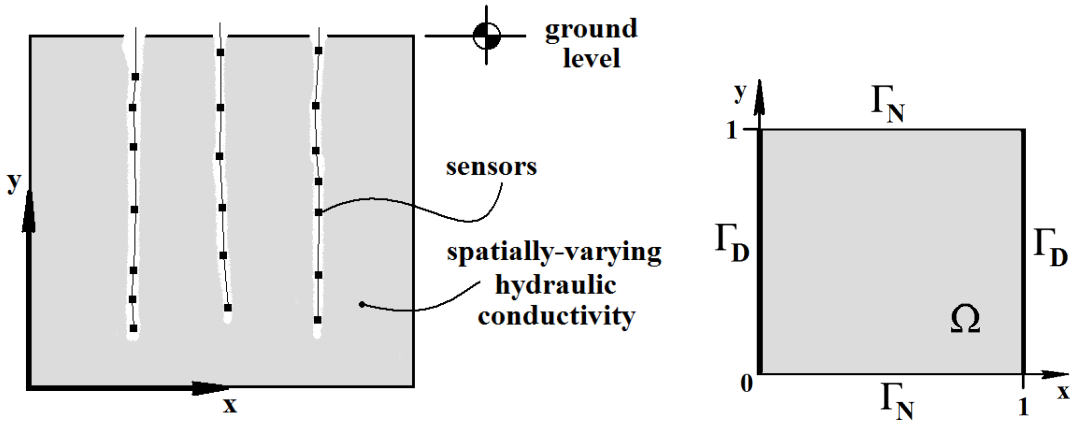
In the following, we take $g = 0$ and $h = 0$ to simplify some of the analysis; the results of Chapter 5 will still hold in the presence of inhomogeneities in the data, though the reduced-order model must be handled more delicately. The function f will be the sum of Gaussians though many other choices are also possible.

2.4.4 2-D model problem

We now describe in detail the 2-D model problem which is considered in this work. In particular, we specify the general continuum and finite element formulations above to a domain and appropriate boundary conditions. Let $\Omega = \{(x, y) \mid 0 \leq x, y \leq 1\}$ be the unit square, $\Gamma = \{(x, y) \mid x \in \{0, 1\} \cap y \in \{0, 1\}\}$ be the boundary, $\Gamma_{\mathrm{D}} = \{(x, y) \mid x \in \{0, 1\}\}$, and $\Gamma_{\mathrm{N}} = \Gamma \backslash \Gamma_{\mathrm{D}}$. An example of one of the meshes is pictured in Figure 2-4.

The governing equations are

$$\begin{aligned} -\nabla \cdot (p \nabla u) &= f \quad \text{in } \Omega, \\ p \nabla u \cdot \vec{n} &= h \quad \text{on } \Gamma_{\mathrm{N}}, \\ u &= g \quad \text{on } \Gamma_{\mathrm{D}} \end{aligned}$$



(a) (b)

Figure 2-5: The 2-D model problem: (a) pictorial representation of groundwater problem and (b) associated computational domain.

where, once again, we take $g = 0$ and $h = 0$. The source term f will be a sum of Gaussians to be specified in Chapter 5. The output equation is defined by a subset of the nodes such that $y_i = u(\vec{x}_i)$ for all $i = 1, 2, \ldots, N_o$. A picture of the problem setup and computational domain is provided in Figure 2-5.

Chapter 3

Model Reduction

In this chapter, we discuss a simultaneous parameter-state model reduction procedure. While the efficiency of a forward model solution depends mostly on the dimensionality of the state N, the cost of the construction or application of the forward operator scales with the number of parameters N_p in the groundwater flow equations. Furthermore, the dimensionality of the domain over which the statistical inverse problem is solved affects the rate of convergence as well. In [65] it is shown that the convergence of estimators scales with N_p^2 when a random-walk Metropolis algorithm is used and with $N_p^{1/3}$ when a Langevin proposal is chosen. Additionally, we expect that the admission of the important directions of the parameter space to the basis will relieve some of the anisotropy of the isoprobability contours of the posterior. It is demonstrated in [14] that exploration of the posterior via MCMC can be slowed significantly if the posterior has anisotropic isoprobability contours. For this reason, the authors suggest approximating the covariance of the posterior and modifying the proposal distribution accordingly. While this adaptation produces speedups in low-dimensional parameter spaces, a similar approach in high-dimensional domains can result in a large proposal generation cost. We address this issue by projecting into a low-dimensional parameter subspace where the anisotropic isoprobability contours in the full space become more regular. Therefore, performing MCMC sampling in the reduced space does not require expensive proposal modifications: we can utilize traditional Metropolis-Hastings samplers.

In the remainder of this chapter, we introduce the projection framework for model reduction, formulate a reduced-order model with reductions in both state and parameter, and discuss the greedy sampling method for obtaining the reduced parameter and state bases. In Section 3.1, the projection framework is discussed. The concept is applied to reductions in both state and parameter to formulate the reduced-order model in Section 3.2. In Section 3.3, the greedy sampling method is motivated and discussed. The application to the groundwater flow equations is delineated and optimality conditions are derived. We conclude with a note on implementational details.

3.1 Galerkin projection framework

In this section, we introduce the projection framework for model reduction. Let $\mathcal{M} : \mathcal{D} \to \mathcal{O}$ be a model which has input space $\mathcal{D} \subset \mathbb{R}^{N_p}$ and output space $\mathcal{O} \subset \mathbb{R}^{N_o}$ through a state space $\mathcal{S} \subset \mathbb{R}^N$. Suppose that the input-output maps of interest occupy a subset $\mathcal{S}_r$ of the state space $\mathcal{S}$. Then we can obtain a reduced-order model which is accurate for the input-output maps which travel through $\mathcal{S}_r$ by assuming that the state always resides in this subspace. Let $\mathbf{V} \in \mathbb{R}^{N \times n}$ be a basis for the state in the subspace $\mathcal{S}_r$ such that $\text{span}\{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n\} = \mathcal{S}_r$ and $n \ll N$. We obtain a reduced model by assuming $\mathbf{u} = \mathbf{V}\mathbf{u}_r$ and projecting the governing equations onto the reduced subspace. Consider the forward model (2.3)–(2.4). By substitution and projection, we obtain the reduced-order model

$$\mathbf{A}_r(\mathbf{p})\mathbf{u}_r = \mathbf{s}_r, \tag{3.1}$$

$$\mathbf{y}_r = \mathbf{C}_r\mathbf{u}_r \tag{3.2}$$

where $\mathbf{A}_r(\mathbf{p}) = \mathbf{V}^T\mathbf{A}(\mathbf{p})\mathbf{V}$, $\mathbf{s}_r = \mathbf{V}^T\mathbf{s}$, and $\mathbf{C}_r = \mathbf{C}\mathbf{V}$. When the governing equations arise from the discretization of a PDE, we transform a large sparse linear system into a small dense system. In a Galerkin manner, we require that the residual be orthogonal to the basis functions — our test and trial bases are the same. Other projections are possible, but we do not explore those here.

There are many interesting questions to be answered about the construction, usage, and accuracy of projection-based reduced-order models. How do we select the basis $\mathbf{V}$? What is the tradeoff between accuracy and computational efficiency for different numbers of basis vectors? Can we quantify the error in outputs between the full- and reduced-order models for a parameter $\mathbf{p}$ such that $\mathbf{u}(\mathbf{p}) \notin \mathcal{S}_r$? We address the challenge of selecting the reduced basis for large-scale models parameterized by field quantities. In this setting, we extend previous work to the new challenge of sampling a very high-dimensional — even infinite-dimensional — parametric input space; in addition, we pursue reduction in the parameter in addition to the traditional reduction of state.

The state basis is usually obtained by sampling the parametric input space, propagating through the forward model, and computing a basis which spans the space of the resulting states. Popular methods for sampling the inputs vary little in methodology but go by many different names. One very popular method is to select the inputs one expects for one's particular problem. Then, the resulting states are converted to an efficient basis by means of a singular value decomposition (SVD). This method goes by many titles including proper orthogonal decomposition (POD) [68], principal components analysis (PCA) [60], Karhunen-Loeve (K-L) expansion [46], and empirical orthogonal functions (EOF) [62] depending on the community. The SVD provides insight into redundant data and guides the user in deciding how many basis vectors to retain. In fact, the resulting basis can be shown to be optimal in the sense that the representation of the states in the basis produces minimal error in the 2-norm. Unfortunately, this error bound only holds for the data set computed, which is directly dependent on the samples chosen.

A projection-based reduced-order model is sensitive to the samples used to generate the subspace spanned by the basis. If the relevant inputs are not sampled, the reduced-order model will not be an accurate surrogate for the high-fidelity model. Therefore, reduced-order models constructed by sampling in an ad-hoc manner may not be sufficient. When a reduced-order model will be used in the context of inverse problems, the observability of the system to sampled inputs is especially important.

To this end, we utilize a systematic, yet heuristic, goal-oriented approach. In Section 3.3, we outline the greedy sampling method and discuss its implementation and application to the groundwater flow equations. First, however, we extend the reduction to the parameter space as well.

3.2 Parameter and state reduction

Forward models with high-dimensional parametric input spaces may also benefit from reduction in the parameter. Recall that the forward model (2.3)–(2.4) arises from discretization of a PDE where the parameter is a field quantity, and is therefore discretized over the mesh. Furthermore, recall we are interested in models whose forward operator is linear in the parameter. In this setting, we expect reduction in the parameter to be beneficial because the construction (or application) of the linear operator depends linearly on the number of parameters. If a direct method is used to solve the linear system (2.3), then the construction of the forward operator is accelerated by reducing the parameter. If instead an iterative method is utilized, the cost of the application of the forward operator can be reduced in the matrix-vector product.

There is an additional motivation for reduction in the parameter. In the statistical inverse problem, we must undertake a separate sampling procedure in the form of Markov chain Monte Carlo. Conducting MCMC in a high-dimensional parameter space can be a daunting task. In general, it may take many samples to explore the space completely. Furthermore, the support of the posterior may occupy a small region of an otherwise enormous space. If one does not begin the Markov chain near the body of the posterior, the chain may wander without reaching the support at all. Additionally, it is shown in [65] that the convergence of a random-walk Metropolis (RWM) algorithm scales with the square of the number of parameter dimensions. We propose to conduct MCMC in the reduced parameter space.

We pursue reduction in the parametric input space in an analogous manner to the reduction in state described in Section 3.1. Suppose the parameters of interest

reside in a subspace $\mathcal{D}_r$ of the parametric input space $\mathcal{D}$. Let $\mathbf{P} \in \mathbb{R}^{N_p \times n_p}$ be the reduced parameter basis with $n_p \ll N_p$ and $\text{span}\{\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_{n_p}\} = \mathcal{D}_r$. We obtain an accurate reduced-order model for parameters in the reduced subspace by requiring the parameter to reside in the subspace, i.e. $\mathbf{p} = \mathbf{P}\mathbf{p}_r$. The reduced-order model is then written

$$\mathbf{A}_r(\mathbf{p}_r)\mathbf{u}_r = \mathbf{s}_r, \tag{3.3}$$

$$\mathbf{y}_r = \mathbf{C}_r\mathbf{u}_r \tag{3.4}$$

where the dependence of $\mathbf{A}_r$ is now explicitly on the reduced parametric representation. For reduced parameters, the forward model outputs are computed directly using the reduced representation. Thus, the cost of a reduced-order forward evaluation is independent of the mesh dimension.

We build the reduced-order model by adaptively sampling in parameter space. At each iteration we sample the parameter space. Using that sample, we obtain the corresponding state by solving the high-fidelity forward model. The parameter and state samples are incorporated into the parameter and state bases respectively. See Algorithm 1.

Algorithm 1 Parameter and state model reduction.

Determine sample in parameter space
Solve the high-fidelity forward model for the state
Build basis $\mathbf{P}$ using the parameter sample and basis $\mathbf{V}$ using the state sample

3.3 Greedy sampling

We now discuss the selection of the reduced parameter basis and consequently, reduced state basis. The parametric input space is sampled by an iterative greedy approach first proposed in [71] and extended in [12]. Instead of sampling the parameter space at random, we attempt to sample in such a way that the reduced-order model is accurate over the map from input space to output space. In this way, our samples are

specified for the governing equations and the output equation, i.e. sensor locations.

3.3.1 Algorithm

The greedy sampling method is an iterative procedure. At each iteration, we attempt to sample the point in parameter space at which the error between full- and reduced-order model outputs is maximal [32, 70]. At each local optimum, an objective function evaluation determines the maximal error. The iteration continues until that error has decreased sufficiently.[1] See Algorithm 2. We note here two important points. Firstly, as advertised, the procedure is sub-optimal. We do not consider the effect of sampling a particular point in parameter space before doing so. The worst-case point is selected without forward looking. Secondly, the sampling procedure requires repeated solution of a large-scale NLP. The solution of such programs requires gradient-based algorithms, and therefore we are only guaranteed to find local optima.

Algorithm 2 Greedy sampling algorithm.

loop until convergence
Sample the parameter field which maximizes the error in full- and current reduced-order model outputs (subject to regularization)
Solve the high-fidelity forward model for the sampled parameter to find the corresponding state
Incorporate the parameter and state samples into their respective bases by Gram-Schmidt orthogonalization
Update the reduced-order model
end loop

Nevertheless, the goal-oriented approach accounts for the governing equations and model outputs and thereby establishes a reduced-order model which is tuned to be accurate over the range of inputs which have greater influence over the specified outputs. If we sample points in parameter space where the model mismatch is large, we eliminate the model reduction error at the sample. It stands to reason that if

[1] The stopping criterion may depend on the application, but typically we look for several orders of magnitude decrease from the first sample. In practice, one may set a stopping criterion on the objective function value (e.g. $\mathcal{J} < 10^{-4}$), but note that this provides no guarantee about the output error when the reduced-order model is utilized because of the limitation that gradient-based optimization can only find local optima.

we continue this process until the maximal error is small, the result is an accurate reduced-order model for the input-output map of interest. We now formulate the greedy sampling problem and derive the optimality conditions for full-order models of the form (2.3)–(2.4) and reduced-order models (3.3)–(3.4).

At each iteration, we solve an NLP over the full-dimensional parameter space. The objective function we wish to maximize is the sum of two terms: the first term accounts for the model mismatch and the second term is a regularization consistent with the prior we plan to employ in the MCMC simulation. We have

$$\mathbf{p}^* = \arg \max_{\mathbf{p}\in\mathcal{D}, \|\mathbf{p}\|_{\mathbf{X}}=1} \quad \mathcal{J} = \frac{1}{2}\|\mathbf{y}(\mathbf{p}) - \mathbf{y}_r(\mathbf{p})\|_2^2 - \frac{1}{2}\mathbf{p}^T\mathbf{S}^{-1}\mathbf{p}, \tag{3.5}$$

$$\text{subject to} \quad \mathbf{A}(\mathbf{p})\mathbf{u} = \mathbf{s}, \tag{3.6}$$

$$\mathbf{y} = \mathbf{C}\mathbf{u}, \tag{3.7}$$

$$\mathbf{A}_r(\mathbf{p})\mathbf{u}_r = \mathbf{s}_r, \tag{3.8}$$

$$\mathbf{y}_r = \mathbf{C}_r\mathbf{u}_r \tag{3.9}$$

where $\mathbf{S} \in \mathbb{R}^{N_p \times N_p}$ is the covariance of the chosen prior. Included in the formulation is a norm constraint on the parameter field. There are two reasons for including this constraint. Firstly, we are in search of a basis direction and are not concerned with the magnitude of the parameter we sample.[2] Secondly, the constraint limits the search space over which the optimization must be performed. Program (3.5)–(3.9) is a PDE-constrained NLP over high-dimensional parameter space. In this formulation, we assume the regularization parameter has been incorporated into the matrix $\mathbf{S}$. Note that the problem is constrained by the full-order model and the reduced-order model. We do not include bound constraints in this formulation though they could be incorporated if necessary.

A few remarks about NLP (3.5)–(3.9) are necessary here. The motivation for the

[2]Note that a representation of any scaled basis vector will be perfect; and furthermore, the resulting state will be represented perfectly as well. The norm $\|\cdot\|_{\mathbf{X}}$ should be chosen such that if $\mathbf{p}_1$ and $u(\mathbf{p}_1)$ are sampled where $\|\mathbf{p}_1\|_{\mathbf{X}} = 1$, then $u(\mathbf{p}_1) = cu(c\mathbf{p}_1)$ for any $c \in \mathbb{R}$. Therefore there is no value in sampling parameters in the direction of $\mathbf{p}_1$ with any other magnitude, thereby justifying the norm constraint.

inclusion of the model mismatch term is self-explanatory and consistent with previous work. Since our program attempts to find the high-fidelity parameter which maximizes the error between full- and current reduced-order model outputs, we must use the reduced-order model with the full-dimensional parameter as input. For example, we may compute the reduced parameter by requiring a low-fidelity reconstruction to be equivalent to the full-order parameter under a suitable inner product. In order to maintain efficiency, the basis vectors should be orthogonalized with respect to this inner product.

Whether one should or should not include the regularization term is not so clear. In the context of MCMC sampling, we believe such a penalty should be included for the following reason. If the MCMC were to be carried out in the full-order parameter space, then parameters for which $\mathbf{p}^T\mathbf{S}^{-1}\mathbf{p}$ is large are unlikely to be sampled when the prior has the form $\gamma_{\mathbf{p}}(\mathbf{p}) = \exp\{-\frac{1}{2}\mathbf{p}^T\mathbf{S}^{-1}\mathbf{p}\}$. Therefore, it is inefficient to sample unlikely parameters when deriving the reduced-order model, and thus, a penalty based on the prior is included in the objective function. Furthermore, (3.5)–(3.9) is nonconvex. In fact, as the greedy sampling procedure progresses, more and more points in parameter space have model mismatch near zero and the terrain becomes less and less friendly to an optimizer. The relative effect of this on the objective function will be determined by the relative weight assigned to the regularity and to the mismatch. If the mismatch dominates, then the space could become very unfriendly and special methods (e.g. grid continuation, see Appendix C) will likely be required. If the regularity dominates, the space will remain smoother, but then the reduced-order model is less sensitive to the model outputs.

3.3.2 Optimality conditions

We now derive the first- and second-order necessary conditions for an optimal solution $\mathbf{p}^*$ of the greedy sampling problem (3.5)–(3.9). It is important to note that differentiation and discretization do not always commute in this context. However, when the forward model represents the steady groundwater flow equations and the regularization term can be represented by an inner product over the domain, differentiation

and discretization do commute. We proceed with the discretized forms.

To derive the optimality conditions, we take a variational approach. First, we define the Lagrangian functional by dualizing the state equation constraints (and substituting the output equations directly into the objective function). Then, the first-order conditions require stationarity of the Lagrangian with respect to the parameter, state, and adjoint variables. Let

$$\mathcal{L} = \mathcal{J} + \boldsymbol{\lambda}^T(\mathbf{A}\mathbf{u} - \mathbf{s}) + \boldsymbol{\mu}^T(\mathbf{A}_r\mathbf{u}_r - \mathbf{s}_r) \tag{3.10}$$

be the Lagrangian functional with adjoint variables $\boldsymbol{\lambda} \in \mathbb{R}^N$ and $\boldsymbol{\mu} \in \mathbb{R}^n$.

We consider first a variation with respect to the adjoint variable $\boldsymbol{\lambda}$. For arbitrary variation $\hat{\boldsymbol{\lambda}} \in \mathbb{R}^N$, the optimality condition requires

$$\begin{aligned} 0 &= \frac{\partial}{\partial\epsilon}\left[\mathcal{L}(\boldsymbol{\lambda} + \epsilon\hat{\boldsymbol{\lambda}})\right]_{\epsilon=0}, \\ &= \frac{\partial}{\partial\epsilon}\left[\mathcal{J} + (\boldsymbol{\lambda} + \epsilon\hat{\boldsymbol{\lambda}})^T(\mathbf{A}\mathbf{u} - \mathbf{s}) + \boldsymbol{\mu}^T(\mathbf{A}_r\mathbf{u}_r - \mathbf{s}_r)\right]_{\epsilon=0}, \\ &= \hat{\boldsymbol{\lambda}}^T(\mathbf{A}\mathbf{u} - \mathbf{s}). \end{aligned}$$

Since this relation must hold for arbitrary variations, we recover the full-order state equation as expected, i.e. $\mathbf{A}\mathbf{u} = \mathbf{s}$.

Now consider an arbitrary variation $\hat{\boldsymbol{\mu}} \in \mathbb{R}^n$ to the reduced adjoint variable $\boldsymbol{\mu}$. First-order optimality requires

$$\begin{aligned} 0 &= \frac{\partial}{\partial\epsilon}\left[\mathcal{L}(\boldsymbol{\mu} + \epsilon\hat{\boldsymbol{\mu}})\right]_{\epsilon=0}, \\ &= \frac{\partial}{\partial\epsilon}\left[\mathcal{J} + \boldsymbol{\lambda}^T(\mathbf{A}\mathbf{u} - \mathbf{s}) + (\boldsymbol{\mu} + \epsilon\hat{\boldsymbol{\mu}})^T(\mathbf{A}_r\mathbf{u}_r - \mathbf{s}_r)\right]_{\epsilon=0}, \\ &= \hat{\boldsymbol{\mu}}^T(\mathbf{A}_r\mathbf{u}_r - \mathbf{s}_r). \end{aligned}$$

The equation must hold for arbitrary variations and we recover the reduced state equations $\mathbf{A}_r\mathbf{u}_r = \mathbf{s}_r$.

We now derive the adjoint equations by considering variations in the full-order

and reduced-order states. Let $\hat{\mathbf{u}} \in \mathbb{R}^N$ be arbitrary. Then optimality requires

$$\begin{aligned}
0 &= \frac{\partial}{\partial \epsilon}\left[\mathcal{L}(\mathbf{u}+\epsilon\hat{\mathbf{u}})\right]_{\epsilon=0}, \\
&= \frac{\partial}{\partial \epsilon}\left[\mathcal{J}(\mathbf{u}+\epsilon\hat{\mathbf{u}}) + \boldsymbol{\lambda}^T(\mathbf{A}(\mathbf{u}+\epsilon\hat{\mathbf{u}})-\mathbf{s}) + \boldsymbol{\mu}^T(\mathbf{A}_r\mathbf{u}_r-\mathbf{s}_r)\right]_{\epsilon=0}, \\
&= \frac{\partial}{\partial \epsilon}\left[\frac{1}{2}(\mathbf{C}(\mathbf{u}+\epsilon\hat{\mathbf{u}})-\mathbf{y}_r)^T(\mathbf{C}(\mathbf{u}+\epsilon\hat{\mathbf{u}})-\mathbf{y}_r) - \frac{1}{2}\mathbf{p}^T\mathbf{S}^{-1}\mathbf{p} + \boldsymbol{\lambda}^T(\mathbf{A}(\mathbf{u}+\epsilon\hat{\mathbf{u}})-\mathbf{s})\right]_{\epsilon=0}, \\
&= \hat{\mathbf{u}}^T\mathbf{C}^T(\mathbf{C}\mathbf{u}-\mathbf{y}_r) + \hat{\mathbf{u}}^T\mathbf{A}^T\boldsymbol{\lambda}
\end{aligned}$$

for all such variations. Thus we obtain the full-order adjoint equation $\mathbf{A}^T\boldsymbol{\lambda} = -\mathbf{C}^T(\mathbf{y}-\mathbf{y}_r)$. Note that the forward operator of the groundwater flow equations is self-adjoint, i.e. $\mathbf{A} = \mathbf{A}^T$. We can take advantage of this fact in the implementation by reusing information from the solution to the state equations. If an iterative method is used, we can recycle the Krylov vectors. If a direct method is chosen, the Cholesky factor can be stored and re-used.

There is an analogous variation for the reduced-order state. Let $\hat{\mathbf{u}}_r \in \mathbb{R}^n$ be an arbitrary variation. Then,

$$\begin{aligned}
0 &= \frac{\partial}{\partial \epsilon}\left[\mathcal{L}(\mathbf{u}_r+\epsilon\hat{\mathbf{u}}_r)\right]_{\epsilon=0}, \\
&= \frac{\partial}{\partial \epsilon}\left[\mathcal{J}(\mathbf{u}_r+\epsilon\hat{\mathbf{u}}_r) + \boldsymbol{\lambda}^T(\mathbf{A}\mathbf{u}-\mathbf{s}) + \boldsymbol{\mu}^T(\mathbf{A}_r(\mathbf{u}_r+\epsilon\hat{\mathbf{u}}_r)-\mathbf{s}_r)\right]_{\epsilon=0}, \\
&= \frac{\partial}{\partial \epsilon}\left[\frac{1}{2}(\mathbf{y}-\mathbf{C}_r(\mathbf{u}_r+\epsilon\hat{\mathbf{u}}_r))^T(\mathbf{y}-\mathbf{C}_r(\mathbf{u}_r+\epsilon\hat{\mathbf{u}}_r)) - \frac{1}{2}\mathbf{p}^T\mathbf{S}^{-1}\mathbf{p} + \boldsymbol{\mu}^T(\mathbf{A}_r(\mathbf{u}_r+\epsilon\hat{\mathbf{u}}_r)-\mathbf{s}_r)\right]_{\epsilon=0}, \\
&= -\hat{\mathbf{u}}_r^T\mathbf{C}_r^T(\mathbf{y}-\mathbf{C}_r\mathbf{u}_r) + \hat{\mathbf{u}}_r^T\mathbf{A}_r^T\boldsymbol{\mu}
\end{aligned}$$

for all reduced-order variations. This leads to the reduced-order adjoint equations as advertised. We have $\mathbf{A}_r^T\mathbf{u}_r = \mathbf{C}_r^T(\mathbf{y}-\mathbf{y}_r)$. Note the sign difference between the full- and reduced-order adjoint equations.

Finally, we consider variations of the parameter, thereby leading to the gradient of the Lagrangian. Let $\hat{\mathbf{p}} \in \mathbb{R}^{N_p}$ be an arbitrary variation in the parameter. Then,

the variation of the Lagrangian with respect to the parameter is given by

$$\begin{aligned}
\nabla_{\mathbf{p}}\mathcal{L} &= \frac{\partial}{\partial\epsilon}\left[\mathcal{L}(\mathbf{p}+\epsilon\hat{\mathbf{p}})\right]_{\epsilon=0}, \\
&= \frac{\partial}{\partial\epsilon}\left[\mathcal{J}(\mathbf{p}+\epsilon\hat{\mathbf{p}}) + \boldsymbol{\lambda}^T(\mathbf{A}\mathbf{u}-\mathbf{s}) + \boldsymbol{\mu}^T(\mathbf{A}_r\mathbf{u}_r-\mathbf{s}_r)\right]_{\epsilon=0}, \\
&= \frac{\partial}{\partial\epsilon}\left[-\frac{1}{2}(\mathbf{p}+\epsilon\hat{\mathbf{p}})^T\mathbf{S}^{-1}(\mathbf{p}+\epsilon\hat{\mathbf{p}}) + \boldsymbol{\lambda}^T(\mathbf{A}(\mathbf{p}+\epsilon\hat{\mathbf{p}})\mathbf{u}-\mathbf{s}) + \boldsymbol{\mu}^T(\mathbf{A}_r(\mathbf{p}+\epsilon\hat{\mathbf{p}})\mathbf{u}_r-\mathbf{s}_r)\right]_{\epsilon=0}, \\
&= -\hat{\mathbf{p}}^T\mathbf{S}^{-1}\mathbf{p} + \boldsymbol{\lambda}^T\mathbf{A}(\hat{\mathbf{p}})\mathbf{u} + \boldsymbol{\mu}^T\mathbf{A}_r(\hat{\mathbf{p}})\mathbf{u}_r.
\end{aligned}$$

Unaltered, this is the inner product of the gradient with $\hat{\mathbf{p}}$. For optimality, this gradient should converge to zero for all directions $\hat{\mathbf{p}} \in \mathbb{R}^{N_p}$.

The first-order optimality conditions are five coupled ordinary differential equations: state equations for the full and reduced systems, adjoint equations for the full and reduced systems, and the control (gradient) equation. At optimality, there will be a quintuple $(\mathbf{p}, \mathbf{u}, \mathbf{u}_r, \boldsymbol{\lambda}, \boldsymbol{\mu})$ at which all five variations vanish.

We utilize Newton's method in combination with a trust region globalization strategy to solve the NLP (3.5)–(3.9); therefore, it is necessary we provide the optimizer with a routine to compute the Hessian-vector product. We now derive the Hessian of the Lagrangian by using a total differential approach. We have from above

$$\begin{aligned}
\frac{\partial\mathcal{L}}{\partial\boldsymbol{\lambda}} &= \mathbf{A}\mathbf{u}-\mathbf{s} = 0, \\
\frac{\partial\mathcal{L}}{\partial\boldsymbol{\mu}} &= \mathbf{A}_r\mathbf{u}_r-\mathbf{s}_r = 0, \\
\frac{\partial\mathcal{L}}{\partial\mathbf{u}} &= \mathbf{A}^T\boldsymbol{\lambda} + \mathbf{C}^T(\mathbf{y}-\mathbf{y}_r) = 0, \\
\frac{\partial\mathcal{L}}{\partial\mathbf{u}_r} &= \mathbf{A}_r^T\boldsymbol{\mu} - \mathbf{C}_r^T(\mathbf{y}-\mathbf{y}_r) = 0, \\
\left(\frac{\partial\mathcal{L}}{\partial\mathbf{p}}\right)_i &= -\left(\mathbf{S}^{-1}\mathbf{p}\right)_i + \boldsymbol{\lambda}^T\mathbf{A}^i\mathbf{u} + \boldsymbol{\mu}^T\mathbf{A}_r^i\mathbf{u}_r, \quad i = 1, 2, \ldots, N_p
\end{aligned}$$

where $\mathbf{A}^i = \mathbf{A}(\mathbf{e}_i)$ and $\mathbf{e}_i$ is the unit vector in the ith coordinate direction. We now consider total differentials of these equations. In the following we use the shorthand $\mathcal{L}_{\mathbf{x}}$ for $\partial\mathcal{L}/\partial\mathbf{x}$.

The total differential for the gradient expression is

$$\mathbf{d}\mathcal{L}_{\mathbf{p}} = \frac{\partial}{\partial \mathbf{p}}\mathcal{L}_{\mathbf{p}}\mathbf{dp} + \frac{\partial}{\partial \mathbf{u}}\mathcal{L}_{\mathbf{p}}\mathbf{du} + \frac{\partial}{\partial \mathbf{u}_r}\mathcal{L}_{\mathbf{p}}\mathbf{du}_r + \frac{\partial}{\partial \boldsymbol{\lambda}}\mathcal{L}_{\mathbf{p}}\mathbf{d}\boldsymbol{\lambda} + \frac{\partial}{\partial \boldsymbol{\mu}}\mathcal{L}_{\mathbf{p}}\mathbf{d}\boldsymbol{\mu}, \quad (3.11)$$

$$(\mathbf{d}\mathcal{L}_{\mathbf{p}})_i = -\left(\mathbf{S^{-1}dp}\right)_i + \boldsymbol{\lambda}^T\mathbf{A}^i\mathbf{du} + \boldsymbol{\mu}^T\mathbf{A}_r^i\mathbf{du}_r + \mathbf{u}^T\mathbf{A}^i\mathbf{d}\boldsymbol{\lambda} + \mathbf{u}_r^T\mathbf{A}_r^i\mathbf{d}\boldsymbol{\mu}, \quad (3.12)$$

for $i = 1, 2, \ldots, N_p$. The incrementals $\mathbf{du}$, $\mathbf{du}_r$, $\mathbf{d}\boldsymbol{\lambda}$, and $\mathbf{d}\boldsymbol{\mu}$ must satisfy the total differential relations of the other four first-order optimality conditions,

$$\mathbf{d}\mathcal{L}_{\boldsymbol{\lambda}} = \mathbf{A}(\mathbf{dp})\mathbf{u} + \mathbf{A}\mathbf{du} = 0, \quad (3.13)$$

$$\mathbf{d}\mathcal{L}_{\boldsymbol{\mu}} = \mathbf{A}_r(\mathbf{dp})\mathbf{u}_r + \mathbf{A}_r\mathbf{du}_r = 0, \quad (3.14)$$

$$\mathbf{d}\mathcal{L}_{\mathbf{u}} = \mathbf{A}(\mathbf{dp})^T\boldsymbol{\lambda} + \mathbf{A}^T\mathbf{d}\boldsymbol{\lambda} + \mathbf{C}^T(\mathbf{Cdu} - \mathbf{C}_r\mathbf{du}_r) = 0, \quad (3.15)$$

$$\mathbf{d}\mathcal{L}_{\mathbf{u}_r} = \mathbf{A}_r(\mathbf{dp})^T\boldsymbol{\mu} + \mathbf{A}_r^T\mathbf{d}\boldsymbol{\mu} - \mathbf{C}_r^T(\mathbf{Cdu} - \mathbf{C}_r\mathbf{du}_r) = 0. \quad (3.16)$$

This set of incrementals and the total differential of the gradient equation define the Hessian-vector product. For vector $\mathbf{dp} \in \mathbb{R}^{N_p}$ we solve (3.13) for $\mathbf{du}$ and (3.14) for $\mathbf{du}_r$. The results are then substituted into (3.15) and (3.16) to compute $\mathbf{d}\boldsymbol{\lambda}$ and $\mathbf{d}\boldsymbol{\mu}$. Substitution into (3.12) yields the Hessian-vector product as required by an optimizer. We obtain the Hessian-vector product by one forward and one adjoint solve at full-order and one forward and one adjoint solve at reduced-order.

The reduced-order models utilized to generate the results in this work are obtained by solving the greedy sampling problem with MATLAB's `fminunc` function. The large-scale optimization routine is a subspace trust-region Newton method [20, 19] which we now outline. The optimization begins at an initial location and moves from iterate to iterate by solving approximately the trust-region subproblem. The trust-region subproblem is the minimization of a quadratic approximation to the objective function in a neighborhood, the trust-region, of the current iterate. The trust-region subproblem can be solved very efficiently when restricted to two dimensions. The `fminunc` algorithm identifies this subspace by an inexact solution for the Newton direction via preconditioned conjugate gradients (PCG). The Newton direction is given by the solution to the system $\mathbf{Hdp} = -\mathbf{g}$ where $\mathbf{Hdp}$ is the Hessian-vector

product and $\mathbf{g}$ is the gradient. If the current iterate is located where the Hessian is not positive definite, then if a direction of negative curvature is identified by PCG, it is followed to the edge of the trust-region. A proposed step is accepted if sufficient decrease is achieved. If decrease is not achieved, the trust-region radius is reduced and the subproblem is resolved. The iteration continues until the gradient of the objective function is reduced below a specified tolerance.[3] In the implementation the norm constraint is included as a quadratic penalty term in the objective function.

[3]Additional details about `fminunc` can be found in the MATLAB documentation.

Chapter 4

Markov chain Monte Carlo

In Chapter 1 we motivate the necessity for uncertainty quantification in the context of decision-making-under-uncertainty. In this chapter, we develop further the proposed paradigm shift to probabilistic descriptions of parametric uncertainty.

Many important decision-making processes are conducted based on single-point estimates. Such estimates neglect the always-present uncertainty in parameters and data. Not only should parametric inputs be characterized by complete probabilistic descriptions, but the outputs of models used to make decisions must properly propagate that uncertainty. The final decision-making step is manifested in a stochastic or robust optimization problem, depending on the context and the desire to mitigate risk. In this work, we focus on the first step in this process, namely, the identification of a probabilistic quantification of uncertain parametric inputs. In particular, we consider the parameter estimation problem for discretized field quantities ($N_p > 10^4$) in the form of a statistical inverse problem.

In Chapter 2 we introduce the statistical inverse problem and its key components: prior, likelihood, and posterior. We note that, in some sense, the posterior is the solution to the statistical inverse problem but that the analytics are not sufficient to gain insight into the distribution or make it useful for design under uncertainty. Instead, we would like to compute moments of the posterior, e.g. the mean and variance. These metrics give us a better sense of how the posterior is distributed throughout parameter space. Markov chain Monte Carlo (MCMC) sampling provides

a method by which to draw samples from the posterior distribution. From these samples, we may then compute estimates of these moments.

We begin the discussion of MCMC by describing the basic ideas. The standard Metropolis-Hasting (M-H) implementation is described and its satisfaction of detailed balance observed. After highlighting some of the disadvantages of M-H, we turn to adaptive techniques. We outline four adaptive methods including delayed rejection, adaptive Metropolis, Langevin sampling, and componentwise adaptation. In Section 4.4 the potential benefits of reduced-order MCMC are outlined and methodology explained. Section 4.5 explores connections to techniques from large-scale optimization and their potential extension to the probabilistic setting. Finally, in Section 4.6, we provide a few brief comments about the requirements of an MCMC solution to the statistical inverse problem from a broader perspective of design-under-uncertainty.

4.1 Basic Principles

MCMC is a powerful combination of two main concepts: Markov processes and Monte Carlo sampling.

A Markov process is defined by a transition kernel $\kappa(\mathbf{p}, \mathbf{q})$ – that is, a probabilistic description of a movement from $\mathbf{p}$ to $\mathbf{q}$. The transition is Markovian if the transitional kernel depends only on the current parameter $\mathbf{p}$ and the proposed parameter $\mathbf{q}$. A random walk is a set of parameters visited according to transitions by the kernel κ. With sufficiently many transitions, the random walk reaches the stationary distribution of the Markov process.

Monte Carlo sampling is commonly employed for numerical integration in high-dimensional spaces where quadrature suffers from combinatorial function evaluation requirements. If instead samples can be drawn from the integrand, or a weighted function thereof, the integral can be approximated numerically by the law of large numbers via the sample mean.

MCMC combines these two ideas to define a methodology for generating samples from an implicitly-known probability density. A random walk, tending toward sam-

ples of the posterior, through the parameter space is defined by a transition kernel. Moments of the posterior are approximated by discrete approximations to the continuous moments. Whereas Markov processes are defined by the transition kernel and the stationary distribution is the object of interest, MCMC sampling considers the inverse problem of determining the transition kernel whose stationary distribution is the posterior. This result is guaranteed only if the transition kernel satisfies the balance condition as we now describe.

Let $\mathbf{p} \in \mathcal{D}$ be the parameter in domain $\mathcal{D}$. Consider a Markov process with transition kernel $\kappa(\mathbf{p}, \mathbf{q})$ and transition probability $\eta(\mathbf{p})$, $0 \leq \eta(\mathbf{p}) < 1$. From an initial point we repeatedly flip the $\eta(\mathbf{p})$-coin and if heads, we transition via κ; and if tails, we stay put. We normalize the transition kernel to satisfy $\int_{\mathcal{D}} \kappa(\mathbf{p}, \mathbf{q})\mathbf{dq} = 1$. By evaluating the probability that a step of the process leads to a parameter in a subset $\mathcal{B}$ of the domain (see [14], pp. 169–171), we determine that

$$\psi(\mathbf{q}) = \int_{\mathcal{D}} \pi(\mathbf{p}) \left[1 - \eta(\mathbf{p})\right] \kappa(\mathbf{p}, \mathbf{q})\mathbf{dp} + \eta(\mathbf{q})\pi(\mathbf{q}) \tag{4.1}$$

where $\pi(\mathbf{q})$ is the probability that $\mathbf{q} \in \mathcal{B}$ and $\pi(\mathbf{p})$ is the probability of parameter $\mathbf{p}$. Since we wish $\pi(\mathbf{p})$ to be the stationary distribution, we are left with the problem of determining $\kappa(\mathbf{p}, \mathbf{q})$ such that $\psi(\mathbf{p}) = \pi(\mathbf{p})$. This requirement leads to the balance equation

$$\int_{\mathcal{D}} \pi(\mathbf{q})\zeta(\mathbf{q}, \mathbf{p})\mathbf{dp} = \int_{\mathcal{D}} \pi(\mathbf{p})\zeta(\mathbf{p}, \mathbf{q})\mathbf{dp} \tag{4.2}$$

where $\zeta(\mathbf{p}, \mathbf{q}) = \left[1 - \eta(\mathbf{p})\right] \kappa(\mathbf{p}, \mathbf{q})$. If this relation is satisfied then a Markov process defined by the transition kernel κ has stationary distribution $\pi(\mathbf{p})$.

4.2 Metropolis-Hastings

This observation leads directly to the Metropolis-Hastings [57, 39] algorithm. In particular, (4.2) is satisfied certainly if the integrands are equal, $\pi(\mathbf{q})\zeta(\mathbf{q}, \mathbf{p}) = \pi(\mathbf{p})\zeta(\mathbf{p}, \mathbf{q})$, when we have *detailed balance*. Let $\xi(\mathbf{p}, \mathbf{q})$ be a candidate generating kernel such that a proposal $\mathbf{q}$ is determined from the current parameter $\mathbf{p}$. Then

if we define $\zeta(\mathbf{p}, \mathbf{q}) = \alpha(\mathbf{p}, \mathbf{q})\xi(\mathbf{p}, \mathbf{q})$, with

$$\alpha(\mathbf{p}, \mathbf{q}) = \min\left\{1, \frac{\pi(\mathbf{q})\xi(\mathbf{q}, \mathbf{p})}{\pi(\mathbf{p})\xi(\mathbf{p}, \mathbf{q})}\right\}, \tag{4.3}$$

then $\zeta(\mathbf{p}, \mathbf{q})$ satisfies detailed balance. The flexibility of M-H stems directly from the unrestricted form of $\xi(\mathbf{p}, \mathbf{q})$ because detailed balance is enforced by α.

The M-H transition rule is summarized in Algorithm 3.

Algorithm 3 Metropolis-Hastings transition rule.

From the current parameter $\mathbf{p}$, propose a transition to $\mathbf{q}$ by $\xi(\mathbf{p}, \mathbf{q})$
Compute the acceptance ratio $\alpha(\mathbf{p}, \mathbf{q})$ according to (4.3)
Draw t from the uniform distribution $\mathcal{U}([0, 1])$
if $\alpha > t$ **then**
  Move to $\mathbf{q}$
else
  Remain at $\mathbf{p}$
end if

In theory, samples drawn from almost any reasonable candidate generating kernel will result in a Markov chain which tends toward samples from the distribution $\pi(\cdot)$ as the number of samples N_s approaches infinity. However, the rate of convergence of the empirical distribution to the actual distribution can vary greatly depending on the dimensionality of the state space, the regularity of the distribution, and the region of support of the posterior.

Like unsteady physics, MCMC has a transient and stationary regime. The transient is termed *burn-in* and represents the point in the chain where samples start to be collected from the stationary distribution of the Markov process. The burn-in is not sampling the posterior and therefore should be discarded before moments are computed. After that point, the chain enters the stationary regime where selected samples are representative of the posterior. One source of inefficiency of standard M-H algorithms stems from the disparate requirements of these two regimes.

Since the samples from the burn-in do not contribute to our knowledge of the posterior, we attempt to select a candidate generating kernel which minimizes this

transient phase. Without additional information,[1] we must begin the sampling process by either sampling from the prior or selecting randomly from the domain $\mathcal{D}$. The journey from the starting point to the support[2] of the posterior could be arduous indeed. It is often beneficial to increase the covariance of the candidate generating kernel during this phase. If the candidates are proposed too close together, the process may take very many samples to burn-in. This issue is addressed in part by Langevin sampling (see Section 4.3.2) where the proposal distribution is shifted in the direction of the gradient.

Once the chain reaches the stationary distribution, the candidate generating kernel must satisfy two opposing requirements: sufficiently explore the entire support while minimizing the number of samples in the chain. Consider a candidate generating kernel which results in small steps relative to the characteristic length scale of the support. Then, many steps will be accepted, but the chain will meander around the support slowly. Many samples will be required to explore it completely. On the other hand, if the candidate generating kernel proposes long steps, few proposals will be accepted and the chain remains stationary. The hope is that we can find the middleground which represents a compromise between exploring the posterior completely and accepting enough proposals. The caveat here is that if the candidate generating kernel is not properly chosen, the middleground may be infintesimally small or even non-existent. For example, if the posterior is highly anisotropic, the optimal length scale for a multivariate normal candidate generating proposal may still result in an inefficient sampler.

In summary, although the M-H algorithm provides a simple recipe by which to generate candidates from any probability distribution in theory, the practical implementation has many pitfalls. Based on the above observations, it stands to reason that an efficient sampler must be adaptive. At the very least, it should operate differently in the two regimes. The chain should move swiftly to the support of the

[1] Some practitioners solve a large-scale NLP to find a parameter with finite probability from which to start the chain. If one has the tools to solve the optimization problem efficiently, then the MAP estimate could be computed.

[2] To be more precise, we use the term *support* to define the region of the posterior where $\pi(\mathbf{p}) > \epsilon$ such that the isoprobability countour $\pi = \epsilon$ contains 99% of the density.

posterior to minimize burn-in. Then a balance must be struck between acceptance rate and sufficient exploration. Adaptive methods have been active area of research in MCMC for many years but have witnessed a resurgence in the last decade.

The statistics community is responsible in large part for these advances. These approaches typically treat the model as a black box, designing algorithms around data. Our models of interest arise from the discretization of PDEs and therefore contain significant structure of which we may take advantage. A brief discussion of methods by which the physics can be incorporated into the sampling process appears later in this chapter.

4.3 Adaptive MCMC

In light of the difficulties highlighted above in defining an efficient sampler, there has been much recent attention given to adaptive MCMC [36, 66, 65, 35, 52]. In this section we describe four adaptive techniques. These methods were tested for the groundwater inverse problem with varied level of success. The DRAM algorithm of Haario et al. [35] yielded the best practical performance and therefore was utilized to generate the results of Chapter 5.

4.3.1 Delayed rejection adaptive Metropolis (DRAM)

In this section we highlight one adaptive approach which we utilize to generate the results of Chapter 5. Delayed rejection adaptive Metropolis (DRAM) [35] combines local adaptivity through delayed rejection and global adaptivity by adaptive Metropolis. Delayed rejection allows multiple proposals per sample iteration while maintaining reversibility of the underlying Markov process. Adaptive Metropolis leverages the sample covariance to rotate the proposal distribution in an attempt to match anisotropy of the posterior. Although the resulting algorithm is adaptive, the efficiency of the sampler is sensitive to several parameters including the non-adaptive start-up period, the initial candidate generating kernel, the frequency of adaptivity, and the number of delayed rejection cycles. We now describe in detail delayed

rejection and adaptive Metropolis methods.

Delayed rejection

In M-H, we propose a new parameter from the candidate generating kernel and accept the proposal with certain probability, and reject otherwise. When we reject the proposal, the previous parameter is duplicated in the ensemble – that is, $\mathbf{p}_{k+1} = \mathbf{p}_k$. With delayed rejection [31], we instead propose again *without duplicating the previous state in the chain.* If we determine the acceptance probability appropriately in further proposal cycles, we can preserve reversibility of the resulting chain. This implies that the delayed rejection procedure can be discontinued at any time, i.e. when the candidate is eventually accepted or when a preset number of rejection cycles are performed.

Consider one MCMC sample. At a given parameter $\mathbf{p}$ we propose a candidate $\mathbf{q}_1$ from the candidate generating kernel $\xi_1(\mathbf{p}, \mathbf{q})$. We accept the proposal with probability

$$\alpha_1(\mathbf{p}, \mathbf{q}_1) = \min\left\{1, \frac{\pi(\mathbf{q}_1)\xi_1(\mathbf{q}_1, \mathbf{p})}{\pi(\mathbf{p})\xi_1(\mathbf{p}, \mathbf{q}_1)}\right\} = \min\left\{1, \frac{N_1}{D_1}\right\}. \tag{4.4}$$

This is the usual M-H approach with $\xi_1 = \xi$. We utilize the subscript to indicate the delayed rejection cycle.

If the proposal $\mathbf{q}_1$ is accepted, then the delayed rejection approach reverts to M-H for this iteration. If the proposal is rejected, instead of duplicating $\mathbf{p}$ in the chain and then reproposing, we repropose right away. Let $\mathbf{q}_2$ be the next proposal generated from the kernel $\xi_2(\mathbf{p}, \mathbf{q}_1, \mathbf{q}_2)$. Note that this candidate generating kernel can depend on previously rejected state $\mathbf{q}_1$. The proposed state $\mathbf{q}_2$ is then accepted with probability

$$\alpha_2(\mathbf{p}, \mathbf{q}_1, \mathbf{q}_2) = \min\left\{1, \frac{\pi(\mathbf{q}_2)[1-\alpha_1(\mathbf{q}_2, \mathbf{q}_1)]\xi_1(\mathbf{q}_2, \mathbf{q}_1)\xi_2(\mathbf{q}_2, \mathbf{q}_1, \mathbf{p})}{\pi(\mathbf{p})[1-\alpha_1(\mathbf{p}, \mathbf{q}_1)]\xi_1(\mathbf{p}, \mathbf{q}_1), \xi_2(\mathbf{p}, \mathbf{q}_1, \mathbf{q}_2)}\right\} = \min\left\{1, \frac{N_2}{D_2}\right\} \tag{4.5}$$

chosen to preserve the reversibility of the resulting MCMC chain.

In theory, one could perform many rejection cycles. In general, on rejection cycle

i, the acceptance probability $\alpha_i = \min\{1, N_i/D_i\}$ where

$$N_i = \pi(\mathbf{q}_i)\xi_i(\mathbf{q}_i, \ldots, \mathbf{q}_1, \mathbf{p}) \prod_{j=1}^{i-1} \xi_j(\mathbf{q}_i, \ldots, \mathbf{q}_{i-j})[1 - \alpha_j(\mathbf{q}_i, \ldots, \mathbf{q}_{i-j})] \tag{4.6}$$

and D_i satisfy the recursive formula

$$D_i = \xi_i(\mathbf{p}, \mathbf{q}_1, \ldots, \mathbf{q}_i)[D_{i-1} - N_{i-1}]. \tag{4.7}$$

Since the acceptance probabilities are chosen to preserve reversibility, we can perform any number of delayed rejection cycles. If at some point a candidate is accepted, we sample that parameter and continue to the next iteration. If we decide to discontinue the process, then we duplicate the previous parameter as we would in M-H and then continue on to the next iteration.

The delayed rejection concept is particularly useful when the chain is started in a region of low posterior probability. Outside of the support, it may be unlikely that a candidate generating kernel will propose a candidate that will be accepted. With delayed rejection, one can take advantage of knowledge of rejected parameters to locally tune the proposal to increase the chance of proposing a candidate that will be accepted. Without such adaptivity, it is possible for the chain to remain at the initial state for an extended number of samples thereby realizing a long burn-in period and an inefficient sampling process.

Adaptive Metropolis

Whereas delayed rejection is a locally adaptive method modifying the proposal distribution from one rejection cycle to the next, adaptive Metropolis attempts to modify the proposal distribution based on a global view of the samples in the existing chain. In a manner consistent with a M-H implementation, we begin with a multivariate normal distribution defined by covariance $\mathbf{\Gamma} \in \mathbb{R}^{N_p \times N_p}$. After an initial non-adaptive period ($n - 1$ samples), we utilize the samples in the chain to update the proposal

covariance such that

$$\mathbf{\Gamma} = \delta \left[\mathrm{cov}(x_0, \ldots, x_{n-1}) + \epsilon \mathbf{I} \right] \tag{4.8}$$

where $\delta = 2.4^2/N_p$ is a scaling parameter [30] which optimizes mixing properties for the multivariate normal case, $\mathbf{I} \in \mathbb{R}^{N_p \times N_p}$ is the identity matrix, and ϵ is a small constant chosen to ensure positive definiteness of $\mathbf{\Gamma}$.

Updating the covariance matrix does not represent significant computational overhead because it can be obtained by the recursion

$$\mathbf{\Gamma}_{n+1} = \frac{n-1}{n}\mathbf{\Gamma}_n + \frac{\delta}{n}\left[n\bar{\mathbf{p}}_{n-1}\bar{\mathbf{p}}_{n-1}^T - (n+1)\bar{\mathbf{p}}_n\bar{\mathbf{p}}_n^T + \mathbf{p}_n\mathbf{p}_n^T + \epsilon\mathbf{I} \right] \tag{4.9}$$

where $\mathbf{p}_n$ is the nth sample in the chain and $\bar{\mathbf{p}}_n = \frac{1}{n}\sum_{j=1}^{n} x_j$ is the sample mean up to and including sample n. The mean can be updated by a similar recursive formula

$$\bar{\mathbf{p}}_{n+1} = \frac{n\bar{\mathbf{p}}_n + \mathbf{p}_{n+1}}{n+1}. \tag{4.10}$$

Since the adaptation is based purely on empirical knowledge, the sampler is susceptible to over-tuning. If the adaptation is carried out at every iteration and from the very start of the sampling procedure, the proposal distribution may be tuned to a very small region of the posterior. At best, this will lead to slow exploration; in the worst case, the samples may remain in that region for the duration leaving other regions of significant posterior probability entirely unexplored. It is in part for this reason that there exists symbiosis between adaptive Metropolis and delayed rejection. The delayed rejection allows a variety of samples to be collected early in the sampling procedure, and then the proposal covariance can be updated according to samples from different regions of parameter space.

4.3.2 Langevin sampling

If we knew the posterior distribution explicitly, sampling from it directly would be the most efficient process — the ideal case. In general, we do not know the posterior density explicitly, but instead we can only evaluate the relative probabilities of a set

of parameters. Based on the ideal case, however, one may propose candidates from a local approximation to the posterior. Sampling based on the Langevin diffusion process uses gradient information to modify the proposal distribution.

The so-called Metropolis-adjusted Langevin algorithm (MALA) shifts the candidate generating kernel so it is centered about the point $\mathbf{p}_n + \frac{\delta^2}{2}\nabla \log \pi(\mathbf{p}_n)$. By shifting the proposal distribution in the direction of $\log \pi$, the chain tends towards larger values of the posterior. The candidate generating kernel is

$$\xi(\mathbf{p}, \mathbf{q}) = \frac{1}{2\delta^2} \exp\left\{ \frac{1}{\delta^2} \|\mathbf{q} - \boldsymbol{\mu}\|_2^2 \right\} \tag{4.11}$$

where δ is a chosen step size parameter and $\boldsymbol{\mu} = \mathbf{p}_n + \frac{\delta^2}{2}\nabla \log \pi(\mathbf{p}_n)$.

This concept can be useful for two reasons. Firstly, the burn-in transient can be reduced by avoiding many samples in a region of low posterior probability. Secondly, the shift toward the peak of the distribution can keep the chain from meandering too far from the most important sector of parameter space. On the other hand, if the peak of the posterior is not in the region of high posterior density (e.g. if the distribution has an isolated sharp peak), the chain may be shifted away from a region containing a significant portion of the posterior density. This could result in misrepresentative samples in the resulting chain.

Extensions of this method are possible by modifying the covariance matrix of the proposal distribution. This could be done by incorporating the adaptive Metropolis ideas above or by picking up second-order information from the analytical posterior probability.

4.3.3 Componentwise MCMC

Componentwise adaptation has also been suggested [36]. Although the resulting chain ceases to be Markovian it can be shown to remain ergodic and to converge to the correct posterior probability density. In the single component Metropolis-Hastings algorithm, the proposals are performed componentwise. In a given iteration, a one-dimensional proposal and acceptance/rejection cycle is performed in each dimension

of the parameter space. The method can be made adaptive by applying the adaptive Metropolis technique of Section 4.3.1 individually by components.

This algorithm may be particularly useful when the posterior evaluation can be computed efficiently when just one element of the parameter vector is changed. Otherwise, componentwise MCMC will likely lead to a greater number of total posterior evaluations, i.e. forward model solutions, and a resulting increase in CPU time. One may then consider blockwise updating where changes in a block of elements in the parameter vector are proposed at once. This may reduce the total number of forward model evaluations, but it raises new questions including how does one decide to group the components of the parameter.

4.4 Reduced parameter MCMC

Efficient MCMC sampling in high-dimensional parameter spaces remains an open challenge. Although the adaptive approaches described above represent significant strides in this area, the plethora of adaptive methods and combinations thereof in the literature is evidence that no generally applicable methodology has yet been discovered. With independent samples from the distribution, errors in statistics of interest converge like $N_s^{-1/2}$ with Monte Carlo sampling. The samples in MCMC are not independent and therefore the rate of convergence is decreased. In light of this fact, the adaptive measures are efforts to obtain an efficient sampler.

We submit that the difficulty in generating an efficient sampler is partially due to the high dimensionality of the parameter space in which the sampling is conducted. So instead we propose that the MCMC sampling take place in the low-dimensional parameter space defined by the reduced-order model of Chapter 3. We cast the prior and the likelihood into the reduced parameter space; therefore, the posterior is a function of $\mathbf{p}_r \in \mathbb{R}^{n_p}$ with $n_p \ll N_p$. Other approaches along these lines include coarse grid approximations [4] and coarse-scale model preconditioned MCMC [24].

We demonstrate in Chapter 5 that even traditional M-H algorithms will yield efficient samplers in the reduced space. Not only do we accelerate overall CPU time

by the reduction in state which speeds up forward model evaluation, the sampling process will also undergo cost reduction with the elimination of the adaptive measures i.e. computation of delayed rejection probabilities and empirical statistics.

In consideration of the fact that samples in reduced space can only occupy the span of the reduced basis, the reduced MCMC approach will only generate samples from that span. This is certainly a drawback but one we must accept to realize the benefit of an efficient sampling process. Although we cannot yet show theoretical results determining the inconsistencies between full- and reduced-order sampling and how they depend on the quality of the reduced basis, we can state that if the region of the support of the full-order posterior can be well-approximated by the span of a reduced set of parameters, the reduced MCMC results will match well the full-order MCMC results. Determining the inconsistencies and quantifying the error in this approximation is an important direction for further research.

4.5 Connections to large-scale optimization

In terms of future directions of research in adaptive MCMC, there may be insight to be drawn from large-scale optimization. The Langevin sampling of Section 4.3.2 can be viewed as a small step in this direction. In particular, the incorporation of second-order information in the proposal distribution could provide a significant speed-up in convergence as it does in the large-scale optimization setting with Newton's method [13]. With Hessian information one obtains a second-order accurate picture of the posterior probability density locally and can leverage that information to develop an efficient proposal scheme. Furthermore, if the end-use is to have a parametric representation of the posterior, the second-order information could be incorporated to extend the range of agreement between the parameterized density and the true density.

Some other concepts from large-scale optimization may also be useful when extended to the MCMC setting. Grid continuation and other homotopy schemes, interior-point methods, and subspace iterations are three examples.

In light of the extended burn-in period that accompanies even some adaptive measures in high-dimensional MCMC, efficient methods for determining a good starting location are critical. We encounter similar issues in large-scale optimization where the number of iterations can also depend highly on the initial iterate. To generate an initial iterate closer to the basin of attraction of an optimum, we consider a sequence of problems corresponding to different coarseness of discretization. We can begin with an optimization problem in just one- or two-dimensions and solve the problem by brute force. The grid is refined and the solution to the previous problem prolongated to the new grid. This iterative process can lead to a very good initial iterate for the optimization problem at the finest grid level. As a result, more optimization problems are solved but often the total cost is reduced. A similar approach could be applied to determining an initial sample in two ways. This methodology could be applied to solving for the MAP estimate in the optimization setting. In an alternative formulation, one could start the MCMC process in a coarse discretization and extrapolate the results to successively finer meshes.

Grid continuation is one example of homotopy schemes which are commonly employed to enhance the robustness and convergence properties of Newton's method in the solution of nonlinear systems of equations. The method boils down essentially to a temporary reparameterization of the problem – a method commonly used in numerical optimization to enhance the regularity of the objective function. Similar approaches could be applied adaptively in the MCMC setting to manage a reasonable acceptance rate.

Bound constraints also pose difficulty for MCMC sampling. In the general case, it is not feasible to draw proposals from bound-respecting distributions. For one-sided constraints, the combination of a shift and the natural logarithm can be employed – so candidates are generated from the log normal distribution; but for multiple-sided constraints, this author is unaware of such solutions. Without the ability to restrict proposals to the domain, the MCMC chain has a tendency to get stuck at the boundaries where many proposals can land outside of the domain, and therefore must be immediately rejected. These sequences lead to a very low overall acceptance rate

and poor mixing of the chain. To address this problem, ideas from interior-point methods may be extended to the MCMC setting. Interior-point algorithms are characterized by iterates which compromise two objectives: decreasing objective function and avoiding boundaries of the feasible set. This can be achieved by computing the central path and taking steps with a component in the direction of the central path. A similar approach could keep an MCMC chain away from boundaries in a constrained feasible set in the probabilistic setting as well.

When much of the posterior density occupies a reduced subspace of the parameter space, identification of that subspace will lead to the design of more efficient candidate generating kernels. The idea of solving subproblems in subspaces is key to inner-loop conjugate gradient solutions of the Newton direction in Newton-Krylov optimization routines. At each iteration, the Newton direction is obtained by solving approximately the linear system $\mathbf{Hv} = -\mathbf{g}$ where $\mathbf{H} \in \mathbb{R}^{N_p \times N_p}$ is the Hessian, $\mathbf{v} \in \mathbb{R}^{N_p}$ is the Newton direction and $\mathbf{g} \in \mathbb{R}^{N_p}$ is the gradient. The system is often solved inexactly using Krylov methods; then the Newton direction is restricted to the Krylov subspace. In the MCMC setting, a subproblem could be posed to identify the important subspace and generate a candidate based on that information.

4.6 Comment on MCMC requirements

We conclude this chapter with a brief statement about the requirements of an MCMC solution to the statistical inverse problem. In particular, an emphasis on development of methods which are applicable and efficient in both the Gaussian and non-Gaussian settings is critical. As we will see in a brief study in Chapter 5, the posterior of the groundwater inverse problem is non-Gaussian.[3] Much of the existing work in adaptive MCMC assumes multivariate normality of the posterior at least in part. To be widely applicable, proposed methods must incorporate techniques to efficiently sample irregular distributions as well.

[3]In fact, it closely resembles the Rosenbrock function – an inherently difficult shape for existing MCMC sampling techniques.

Furthermore, if the results of the statistical inverse problem are to be propagated through complex models and inputs to stochastic programs, the data compression which results from the characterization of the posterior by moments may be insufficient. Particularly for non-Gaussian densities, the mean may be an uninformative (or even misleading) statistic. It may correspond to a parameter of very low posterior probability. To obtain a more complete probabilistic picture, methods to parameterize the density should be considered. A host of response surface modelling techniques could be leveraged in this case. Approximations would then be aided by first-order and second-order information like that which would already be computed by Langevin sampling and in a subspace algorithm.

Chapter 5

Results

We now demonstrate the methodologies outlined in Chapter 3 and Chapter 4.

In Section 5.1, the greedy sampling results are presented. For cases in 1-D and 2-D, we investigate the resulting basis vectors in both parameter and state. The value of the objective at the solution of each greedy cycle is also presented. The resulting reduced-order models are tested against the full-order model for the forward problem when the parameter field is drawn from a Gaussian smoothness prior.

We then turn to the statistical inverse problem in Section 5.2 and study the prior, likelihood, and posterior for the 1-D case with just two degrees of freedom in the full-order problem. Certainly such a coarse mesh will introduce significant discretization error but the ability to visualize the probability density functions is paramount to understanding the difficulties of MCMC sampling.

Still in 1-D the number of degrees of freedom is increased and the reduced MCMC implementation studied. In particular, we explore the same probability density functions except in the reduced space. We hypothesize about the range of validity of the reduced implementation and highlight the justification for its use in solving the statistical inverse problem.

In Section 5.3 we move to the full problem size in 1-D. Results from the full-order and reduced-order implementation are presented and compared. We demonstrate that the reduced-order implementation leads to similar results in far less CPU time thereby showing that the combination of state and parameter reduction with sampling

in the low-dimensional subspace should be considered for large-scale statistical inverse problems.

Section 5.4 tests the same methodology in the 2-D case. The problem is not fundamentally different in that the degrees of freedom are vectorized independent of the dimensionality in physical space; however, 2-D and especially 3-D problems have higher computational cost for a forward solution than the 1-D case even with the same number of degrees of freedom. Therefore, these problems are both more realistic and a better-justified setting for the reduced implementation.

5.1 Model reduction

In this section, we demonstrate the model reduction algorithm described in Chapter 3. We plot the basis vectors obtained by the greedy sampling algorithm and demonstrate the reduction in objective with increasing number of greedy cycles.

Consider the 1-D problem of Chapter 2 with homogeneous boundary conditions and three superposed Gaussian sources (see Figure 5-1). Before we begin the greedy

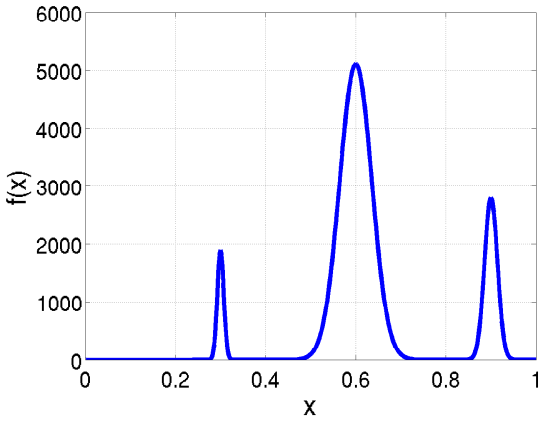


Figure 5-1: Three superposed Gaussian sources for the 1-D problem.

algorithm, we first sample the constant parameter distribution. Then, on the first cycle, we have a reduced-order model that is accurate over the span of constant parameters. The solution of the first greedy cycle problem yields the next parameter basis vector. The forward problem is solved using that parameter and the resulting

state vector is added to the state basis. In Figure 5-2, we plot the parameter and state basis vectors.

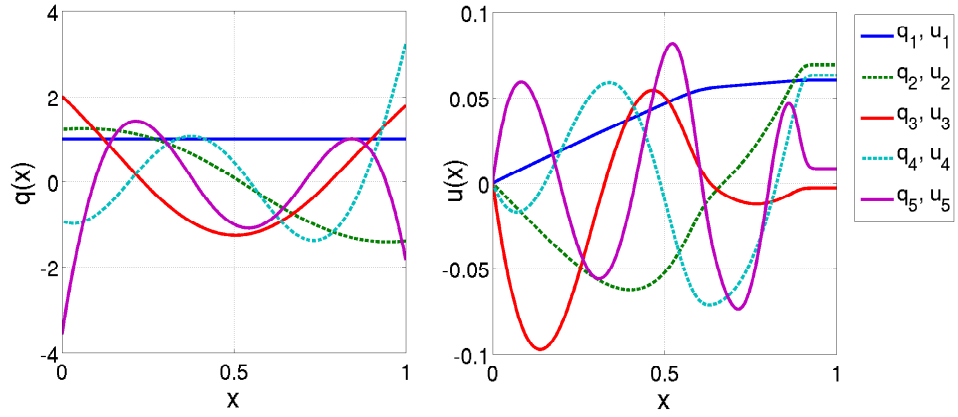


Figure 5-2: On the left, the orthonormalized parameter basis vectors. On the right, the orthonormalized state vectors.

The orthogonalization process is a Gram-Schmidt process in each case, but the parameter basis vectors are made orthonormal in the mass-matrix-weighted inner product $\langle \mathbf{p}, \mathbf{q} \rangle_M = \mathbf{p}^T \mathbf{M} \mathbf{q}$.

In the greedy sampling algorithm, we iteratively sample the parameter which maximizes the error between full-order and current reduced-order model outputs. However, the greedy optimization problem in this case is nonlinear and also non-convex; therefore, we cannot guarantee that we select such a parameter. Consequently, we do not expect to necessarily have a monotonically decreasing optimal objective cost with increasing greedy cycle. We should, however, observe a generally decreasing trend. Figure 5-3 plots the objective cost of the sampled parameter at each greedy cycle for the two test problems in 2-D.

Unfortunately, due to the non-convexity, we cannot make any guarantees about the worst-case error in the reduced-order model outputs based on this data. It is possible that there is a parameter for which the worst-case error is much greater than that shown here. However, in practice, we find that this algorithm is suitable; although it is heuristic, it has been shown that greedy sampling leads to a more efficient reduced-order model than those constructed by random sampling, Latin hypercube sampling

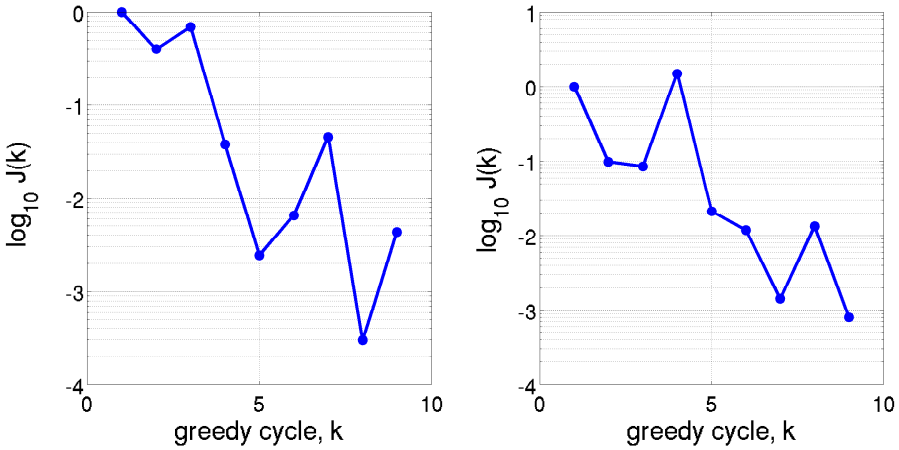


Figure 5-3: The objective of the sampled parameter with increasing greedy cycle for the 2-D test cases. On the left, $N = 59$; and on the right, $N = 494$.

(LHS), and uniform sampling [11].

We demonstrate the accuracy of the reduced-order model with increasing number of basis vectors in Figure 5-4. For three parameter fields drawn from the prior, we compute the model outputs using $n = 1, 2, \ldots, 10$ basis vectors. As the number of basis vectors increases, the parameter and its corresponding state become better resolved and the error decreases.

5.2 Preliminary study

We now turn to a preliminary study in the statistical inverse problem at full-order in 1-D. The purpose of this section is to build some intuition for why the MCMC sampling process is a major challenge for the groundwater flow inverse problem. The difficulties arise for two reasons: the inversion seeks a field variable which is discretized on the mesh and therefore has many degrees of freedom for refined meshes and the likelihood function has non-Gaussian contours. In fact, the contours resemble the Rosenbrock function, a notoriously challenging density for MCMC.

We begin with a very coarse discretization of the 1-D groundwater inverse problem: $N = 2$. With just two degrees of freedom, we can visualize the prior, likelihood, and posterior to build some intuition. To manage the positivity requirement on K, we

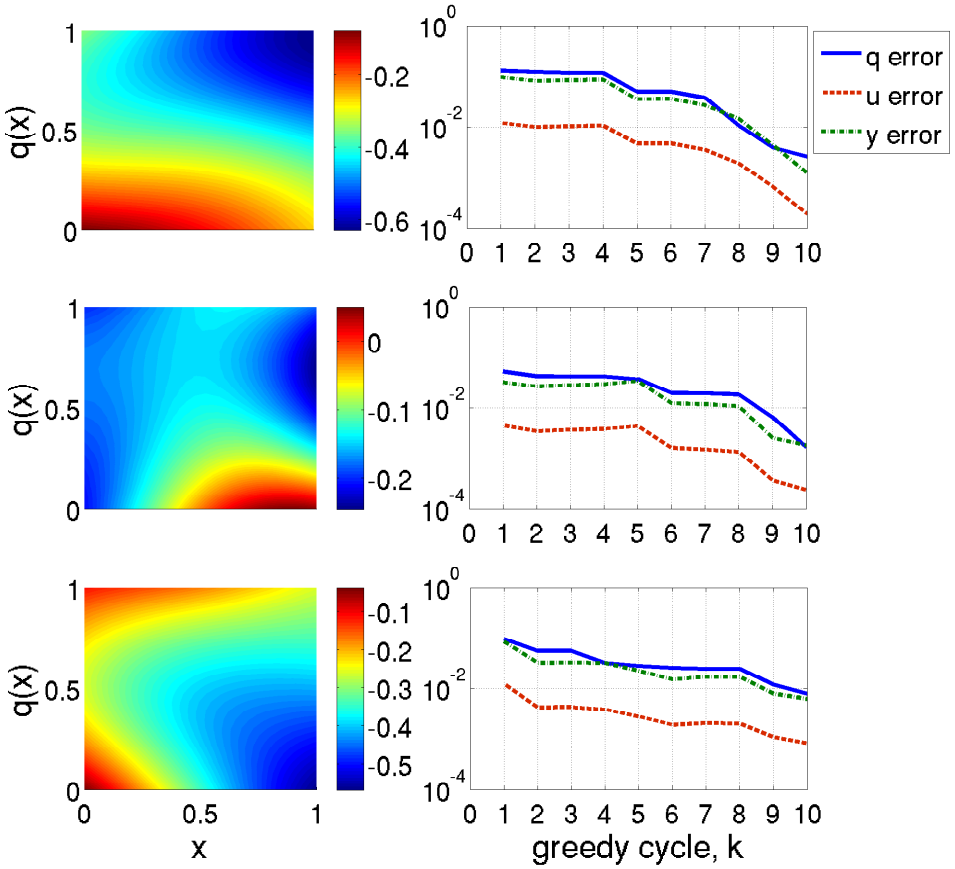


Figure 5-4: On the left, three parameters drawn from the $N = 494$ prior distribution. On the right, the error in parameter, $\|\mathbf{q} - \mathbf{Q}\mathbf{q}_r\|_{\mathbf{M}}$; error in state, $\|\mathbf{u} - \mathbf{V}\mathbf{u}_r\|_{\mathbf{M}}$; and error in output $\|\mathbf{y} - \mathbf{y}_r\|_2$.

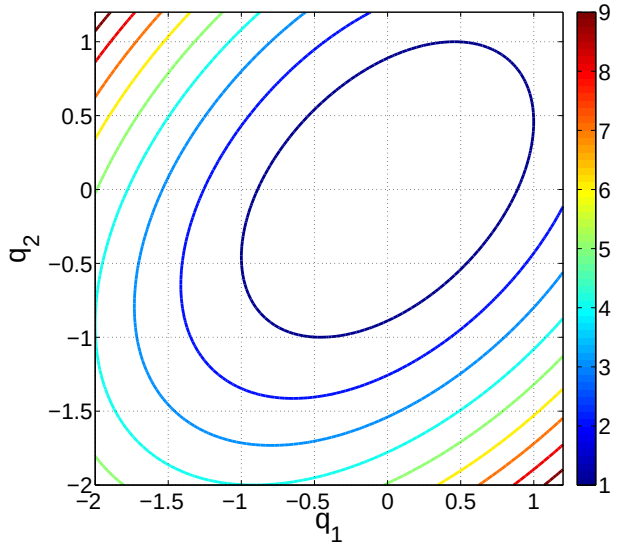
work instead with $q = \log K$ in the following. The parameter is discretized so that $q(x) = q_1(1-x) + q_2$ is defined by the pair (q_1, q_2) over the domain $\Omega = [0, 1]$. The state is discretized in the same manner. The parameter has free boundary conditions, but the state is required to satisfy $u_1 = 0$. This leaves u_2 as the only degree of freedom given a pair (q_1, q_2). When we solve for u_2 in the discrete form we find

$$u_2(q_1, q_2) = \frac{1 - \exp\{-q_1\}}{q_1} \exp\{q_1 + q_2\}, \quad q_1, q_2 > 0. \tag{5.1}$$

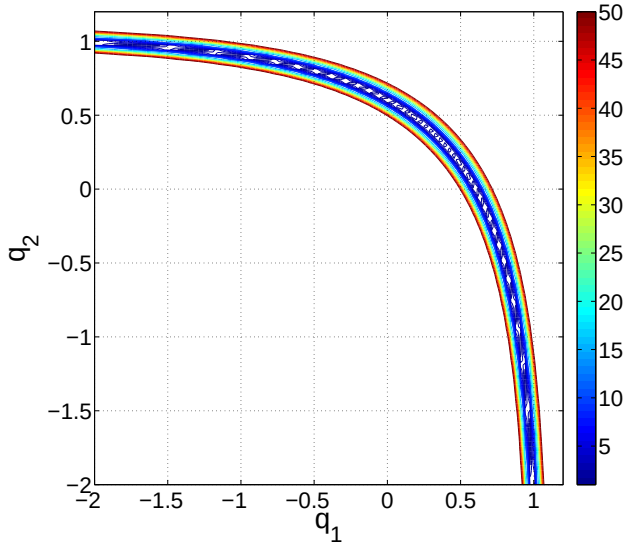
Take just one sensor output at $x = 1$ so that $y_1 = u_2$. With $s = 0.01$ and the likelihood $L(\mathbf{y}|\mathbf{p}) = \exp\{-\frac{1}{2s^2}\|\mathbf{y}(\mathbf{p}) - \mathbf{y}_d\|_2^2\}$ we obtain the likelihood contours in

Figure 5-5 (b). In Figure 5-5 (a) we plot the isoprobability contours of our Gaussian smoothness prior. It is clear that any weighted combination of these sets of contours which respects the curvature of the likelihood will contain this Rosenbrock-like form, and therefore pose several difficulties for the statistical inverse problem.

In Figure 5-6 we plot the contours of the posterior (left) and 100 000 samples obtained by DRAM (right). In this low-dimensional case, DRAM has little difficulty exploring the entirety of the support of the posterior. The first important point to note is that the mean (red dot) of the distribution lies in a region of infinitesimally small posterior probability. Therefore, the mean is a somewhat misleading statistic of the ensemble, and therefore the statistical inverse problem itself. The second difficulty for MCMC arises because the distribution is locally highly anisotropic. Many schemes propose candidates from Gaussian distributions because they are simple to define and random number generators are readily available. Unfortunately, Gaussian proposals will not provide an efficient sampler for these shapes of posterior. In practice, one would like to select the mean and variance of the Gaussian proposal to satisfy two criteria: the posterior is sufficiently explored and the acceptance rate attains a prescribed minimum. One cannot draw an ellipse on top of the posterior which envelopes the majority of the distribution without admitting a lot of area with very low probability. This implies that if one attempts to satisfy the first criterion, the second will not be met. Alternatively, one may draw a very small ellipse around one part of the distribution. In this case, the acceptance rate is good, but MCMC will require many samples to sufficiently explore the entire distribution.

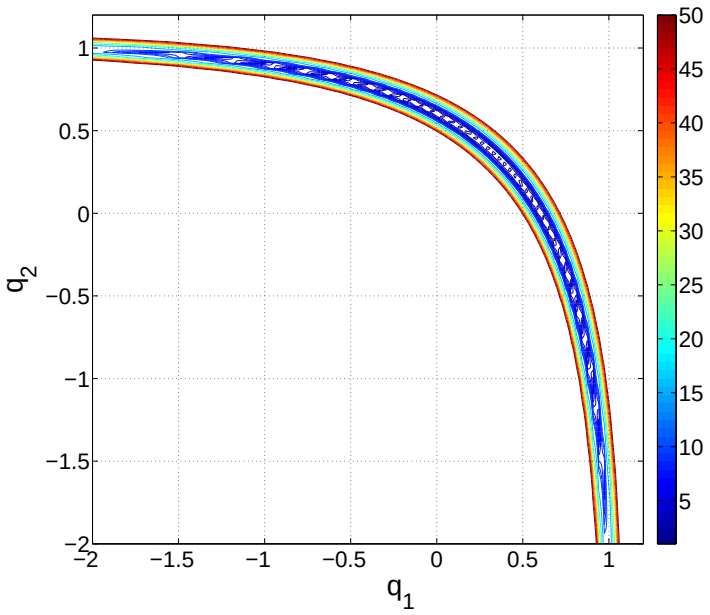


(a)

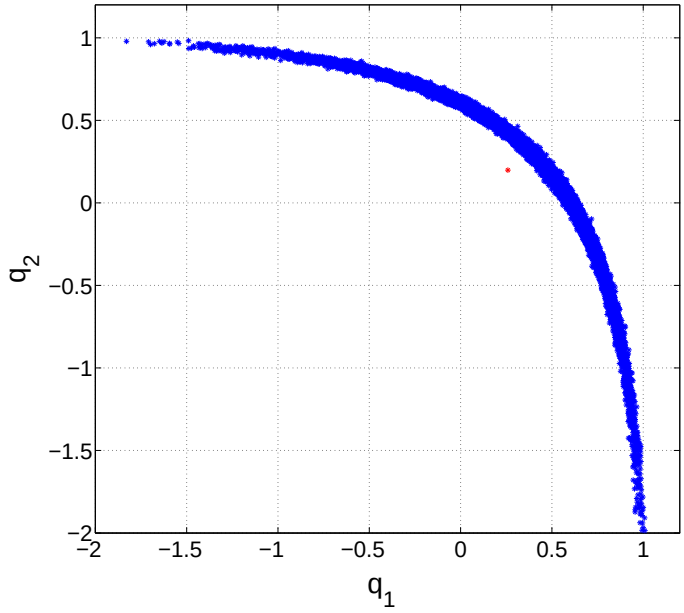


(b)

Figure 5-5: (a) Contours of $-2\log\gamma(\mathbf{q})$ in (q_1, q_2) space and (b) $-2\log L(\mathbf{y}_d|\mathbf{q})$ in (q_1, q_2) space.



(a)



(b)

Figure 5-6: (a) Contours of $-2\log\pi(\mathbf{q}|\mathbf{y}_d)$ in (q_1, q_2) space and (b) 100 000 samples using DRAM starting from the actual parameter field values $(q_1, q_2) = (0.5741, 0.0092)$. Red dot on right shows sample mean. Note that the mean has very low posterior probability. Here, DRAM does a very good job of exploring the entire support of the posterior.

We now consider similar aspects of the solution when the problem has three degrees of freedom, $N = 3$. We can visualize the MCMC samples and get a sense of how the distribution generalizes to higher dimensions. In Figure 5-7, we plot the MCMC chain along with the actual parameter coordinates and the ensemble mean. The distribution appears to be taking on a shape which is the multi-dimensional generalization of that seen in Figure 5-6 (a).

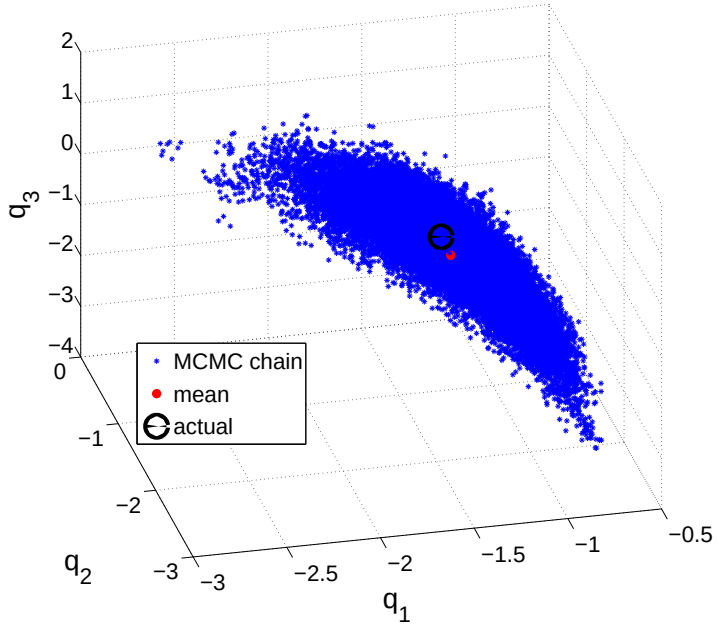


Figure 5-7: Actual, mean, and MCMC chain.

In Figure 5-8 we show the histogram of the MCMC chain components on the diagonal, and the projection into two dimensions of the chain on the off-diagonal. This corresponds to the distributions for two of the coordinates with the third marginalized. The distribution looks very similar to the two-dimensional posterior when the third degree of freedom is marginalized. In the other projections, the distribution is still far from Gaussian but at least is not locally highly anisotropic.

Reparameterizing the chain by the eigenvectors of the sample covariance results in the chain in Figure 5-9 and marginalizations in Figure 5-10.

Although the density is not Gaussian in this new parameterization, the isoprobability contours have become more regular for the third unknown. This may provide

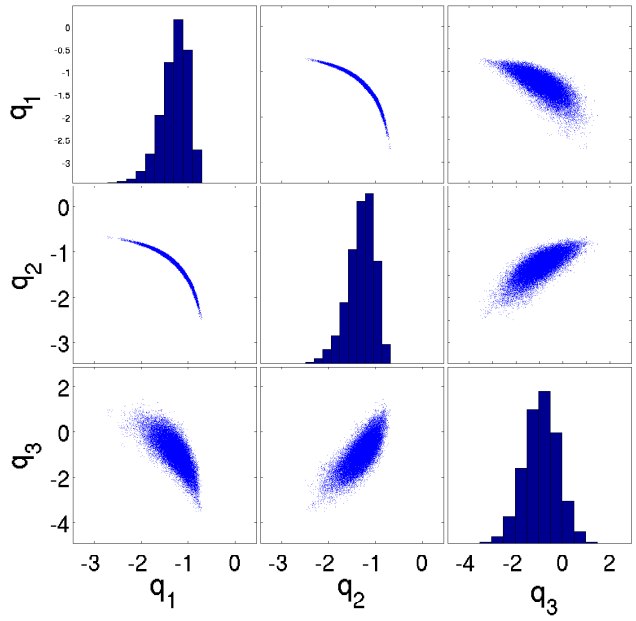


Figure 5-8: On diagonal, histogram of MCMC chain; off-diagonal, MCMC points projected onto plane.

some insight into the reason why we can utilize traditional Metropolis-Hastings algorithms in reduced MCMC. If the greedy algorithm selects important directions of the posterior, the density in reduced space may be more regular than it is when expressed in the original space. It is difficult to make this claim precise due to the nonlinearity between parameter and state, but our results support it.

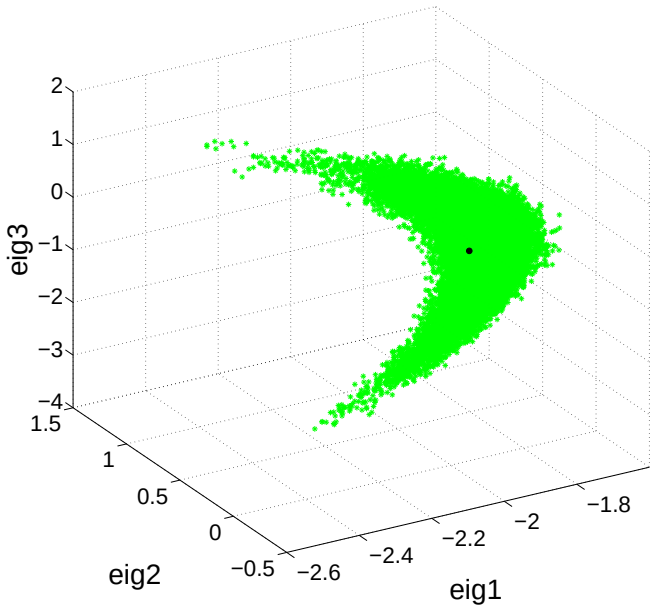


Figure 5-9: MCMC samples in the coordinate system defined by the eigenvectors of the sample covariance. The black dot is the mean.

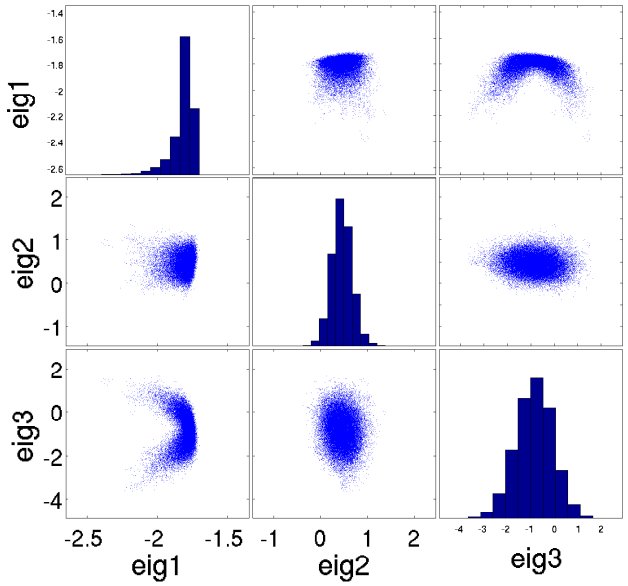


Figure 5-10: On the diagonal, histograms and on the off-diagonal, projected samples in eigenvector coordinates.

5.3 Results in 1-D

We now move to practical results in 1-D at two discretization levels, $N = 51$ and $N = 501$. In each case, we reduce to $n = 5$ and demonstrate that the reduced implementation yields accurate results in far less CPU time. The reduction in cost is achieved for two reasons: firstly, the reduction in state dimension leads to very small dense linear systems and, secondly, the MCMC sampling is conducted in reduced parameter space with the cheaper Metropolis-Hastings approach without adaptivity.

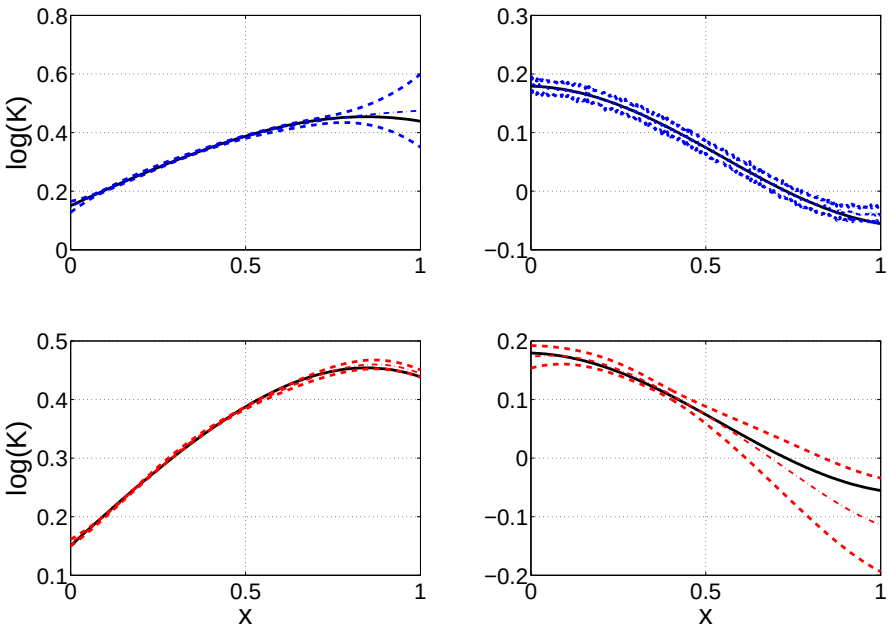


Figure 5-11: On the left, full- (top) and reduced-order (bottom) results for the $N = N_p = 51$ test case. On the right, full- (top) and reduced-order (bottom) results for the $N = N_p = 501$ test case. In each graph we plot the actual parameter field (solid), the MCMC mean estimate (dash-dot), and the $\pm 2\sigma$ envelope (dash). The reduced-order result for $N = 51$ predicts a narrower envelope than the full-order result, whereas the opposite is true for $N = 501$ where the reduced-order implementation predicts greater uncertainty in the right half of the domain. Both reduced-order results encapsulate the true parameter.

The method is tested for $N = N_p = 51$ and $N = N_p = 501$ degrees of freedom (DOFs) with reductions to $n = n_p = 5$ in each case. The data are obtained by generating a parameter field from the prior, propagating through the full-order model, and then adding noise drawn from $\mathcal{N}(\mathbf{0}, s^2\mathbf{I})$. We present results for the full-order and reduced-

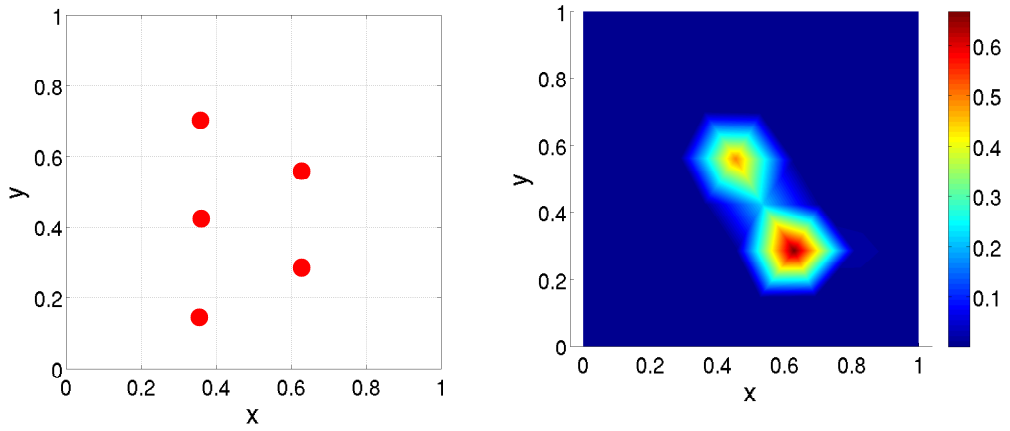
Table 5.1: Number of full-order and reduced-order degrees of freedom and offline, online, and total time required to generate the results of Figure 5-11 on a DELL Latitude D530 with Intel Core 2Duo at 2 GHz. The offline time denotes the CPU time required to solve five iterations of the greedy sampling problem with MATLAB's `fmincon`. In the MCMC simulation, the first 10% of the samples were discarded as burn-in. Two methods used: DRAM with four cycles of delayed rejection and adaptation every 100 steps (DRAM(4,100)) and vanilla Metropolis-Hastings (M-H) with no adaptivity.

N	N_o	n	Offline time (s)	Method	Samples	Online time (s)	Total time (s)
51	5	–	–	DRAM	500 000	1.0508×10^3	1.0508×10^3
51	5	5	3.9778×10^2	M-H	150 000	1.5988×10^2	5.5766×10^2
501	25	–	–	DRAM	500 000	2.4344×10^4	2.4344×10^4
501	25	5	7.2415×10^3	M-H	150 000	2.5345×10^2	7.4950×10^3

order MCMC solutions to the statistical inverse problem. For each full-order case, we use DRAM to perform the MCMC sampling; for reduced-order implementations we use traditional random walk Metropolis-Hastings. The parameters are hand-tuned to provide the best practical performance the author could achieve for each test case.

Figure 5-11 plots the results. The actual log hydraulic conductivity is plotted with the MCMC sample mean and the $\pm 2\sigma$ envelope for full- and reduced-order implementations. The reduced-order method ascertains the mean field on par with the full-order implementation. In the $N = 51$ case, the reduced-order results predict a more narrow envelope near $x = 1$. For $N = 501$, the reduced-order results indicate less certainty than the full-order results, but the match with full-order results is generally consistent. The remaining inconsistency may be corrected if more samples are collected in the reduced MCMC chain.

In Table 5.1, we present the CPU timings for the offline, online, and total cost of the full- and reduced-order implementations. The offline cost corresponds to the greedy sampling procedure to build the reduced-order model bases. These repeated large-scale NLP solutions dominate the total cost of the reduced implementation. By conducting the MCMC sampling process in the reduced parameter space and utilizing the reduced-order model for posterior evaluations we realize a speed up of about two orders of magnitude in the online phase. Although the overall computational savings is a factor of two or three, we note here that we have demonstrated the methodology on



(a) (b)

Figure 5-12: Sensor array (a) and source term (b) for the $N = 59$ test case with $N_o = 5$ sensors.

small model problems. The high-fidelity problems of interest have many more degrees of freedom; in that setting, the full-order MCMC is computationally intractable.

Although there is a substantial offline cost to construct the reduced-order model, the reduced-order MCMC sampling can be performed without adaptivity, which results in a significant reduction in sampling cost. We realize a factor of two speed-up in the $N = 51$ case and a factor of three in the $N = 501$ case; we expect this speed-up to improve as the number of full-order DOFs increases. Indeed the best setting for this methodology is for problems with $N > 10^4$ in multiple dimensions when a full-order forward solve becomes very expensive.

5.4 Results in 2-D

Consider now the 2-D model problem described in Chapter 2. We present results here for the full-order and reduced-order implementations for the problems with full-order dimension $N = 59$ and $N = 494$. For these problems, the unstructured meshes are composed of simplices and generated by `distmesh2d`.[1]

[1]Thanks to Professor Per-Olof Persson for the MATLAB mesher `distmesh2d` which is freely available at `http://math.mit.edu/~persson/mesh/`.

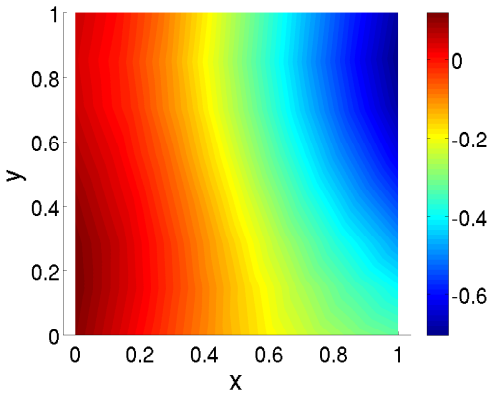


Figure 5-13: True, or actual, $q(x)$ for test case $N = 59$. This parameter field was drawn from the prior distribution.

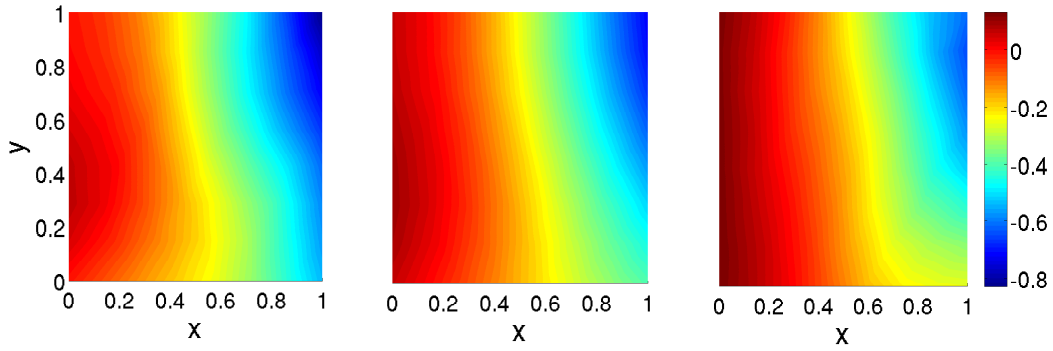


Figure 5-14: The results of the MCMC solution to the $N = 59$ test case. The $\bar{q} - 2\sigma$ and $\bar{q} + 2\sigma$ interval bounds on the left and right, respectively. The mean is shown in the center. Actual parameter field shown in Figure 5-13.

We begin with the $N = 59$ case. The domain has $N_o = 5$ sensors arranged into wells. See Figure 5-12 (a). Figure 5-12 (b) shows the source term $f(x)$. In Figure 5-13 we plot the actual parameter field – the one used to solve the forward problem and generate the noisy data. Figure 5-14 plots $\bar{q} - 2\sigma$, $\bar{q}$, and $\bar{q} + 2\sigma$ for the full-order implementation. It is these results to which we compare the same quantities from the reduced-order MCMC. We investigate the behavior for one reduced model of dimension $n=5$. The parameter basis vectors and the state basis vectors are plotted in Figure 5-15 on the left and right respectively.

With just five basis vectors, the reduced-order results presented in Figure 5-16 match very nicely the full-order results. The resulting interval defined by $(\bar{q} - 2\sigma, \bar{q} +$

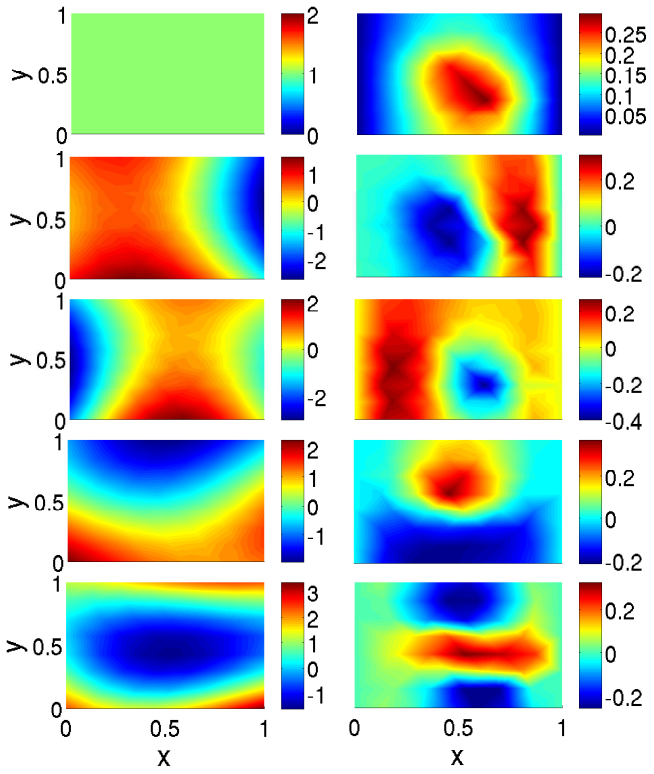


Figure 5-15: The parameter $q(x)$ (left) and state $u(x)$ (right) basis vectors obtained by the greedy sampling algorithm for the $N = 59$ test case.

2σ) is tighter than the same interval of the full-order results. This results is likely due to the fact that some of the full-order MCMC chain cannot be represented by a linear combination of the parameter basis vectors. As the number of basis vectors is increased we expect to capture all of the samples in the full-order MCMC chain. In practice, however, for large-scale models of interest, we cannot afford to compute the full-order chain; therefore, the reduced MCMC approach provides a method for uncertainty quantification in the statistical inverse problem — an attractive alternative to single-point estimation.

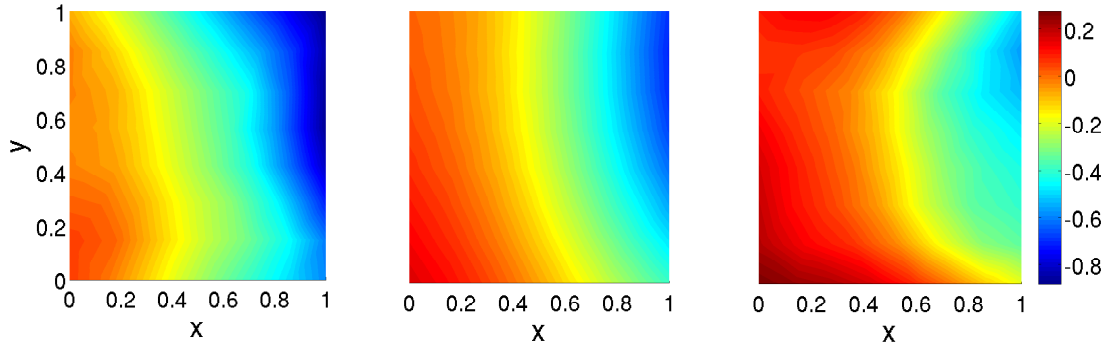
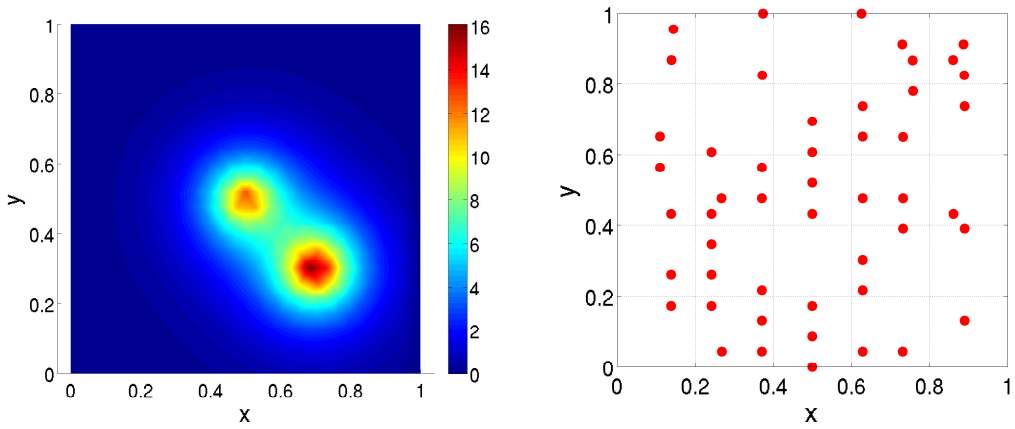


Figure 5-16: The results of the MCMC solution to the $N = 59$ test case with reduction to $n = 5$. The $\bar{q} - 2\sigma$ and $\bar{q} + 2\sigma$ interval bounds on the left and right, respectively. The mean is shown in the center. Actual parameter field shown in Figure 5-13.

We now turn to the $N = 494$ test case. The domain has $N_o = 49$ sensors arranged loosely into wells. The problem is forced by a superposition of Gaussian sources as shown in Figure 5-17 (a). The sensors are distributed according to Figure 5-17 (b).



(a) (b)

Figure 5-17: On the left, source term $f(x)$, and on the right, sensor distribution, for the $N = 494$ test case.

The data are produced by running a full-order forward model solution for a parameter field sampled from the prior.[2] The outputs are calculated and noise is added

[2]Evaluating our approach by generating synthetic data from the prior requires the assumption that our prior is accurate. Although in practice it may be difficult to formulate an appropriate prior, here we eliminate that source of error in order to evaluate the reduced-order implementation.

such that the error is Gaussian, i.e. $\mathbf{y}_d = \mathcal{N}(\mathbf{y}(\mathbf{p}), s^2\mathbf{I})$ with $s = 0.01$.[3] The true, or actual, parameter field is pictured in Figure 5-18.

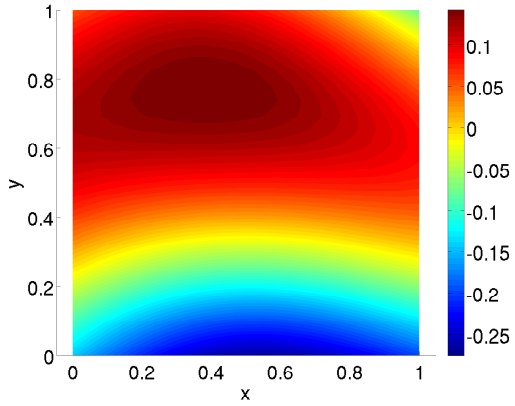


Figure 5-18: True, or actual, $q(x)$ for test case $N = 494$. This parameter field was drawn from the prior distribution.

First we present the results from the full-order implementation. It is important to recall here that, although we present the mean $\bar{q}$, it may not be an informative statistic for these types of non-Gaussian posteriors. More reliable in this case is the range between $(\bar{q} - 2\sigma, \bar{q} + 2\sigma)$. The results from the MCMC solution are plotted in Figure 5-19.

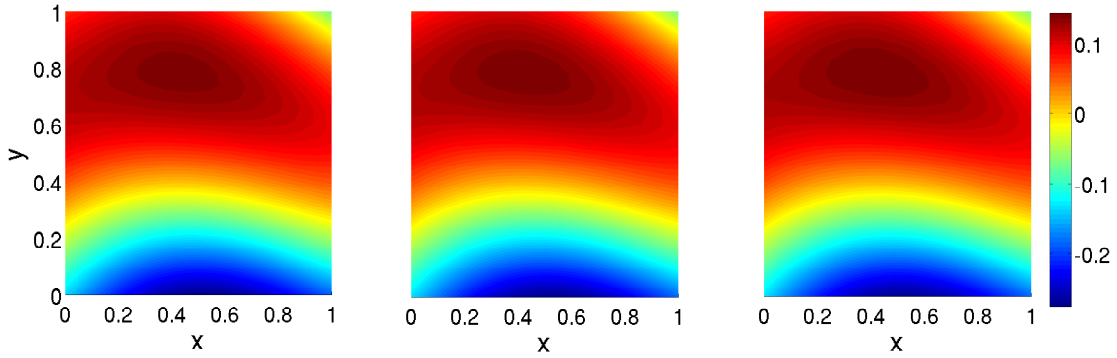


Figure 5-19: The results of the MCMC solution to the $N = 494$ test case. The $\bar{q} - 2\sigma$ and $\bar{q} + 2\sigma$ interval bounds on the left and right, respectively. The mean is shown in the center. Actual parameter field shown in Figure 5-18.

The full-order implementation identifies an interval in which the actual parameter

[3]For our parameter fields, the isoprobability contour containing 95% of the likelihood density corresponds to approximately 2% error in the sensor measurements.

field is fully contained. As discussed earlier, we do not expect the mean to match the parameter field. Here, the mean is qualitatively close to the actual parameter field, but certainly differs slightly. Although we do not have the ability to determine the true distribution here, we take these full-order MCMC results as truth and use them as a baseline to compare the reduced MCMC results.

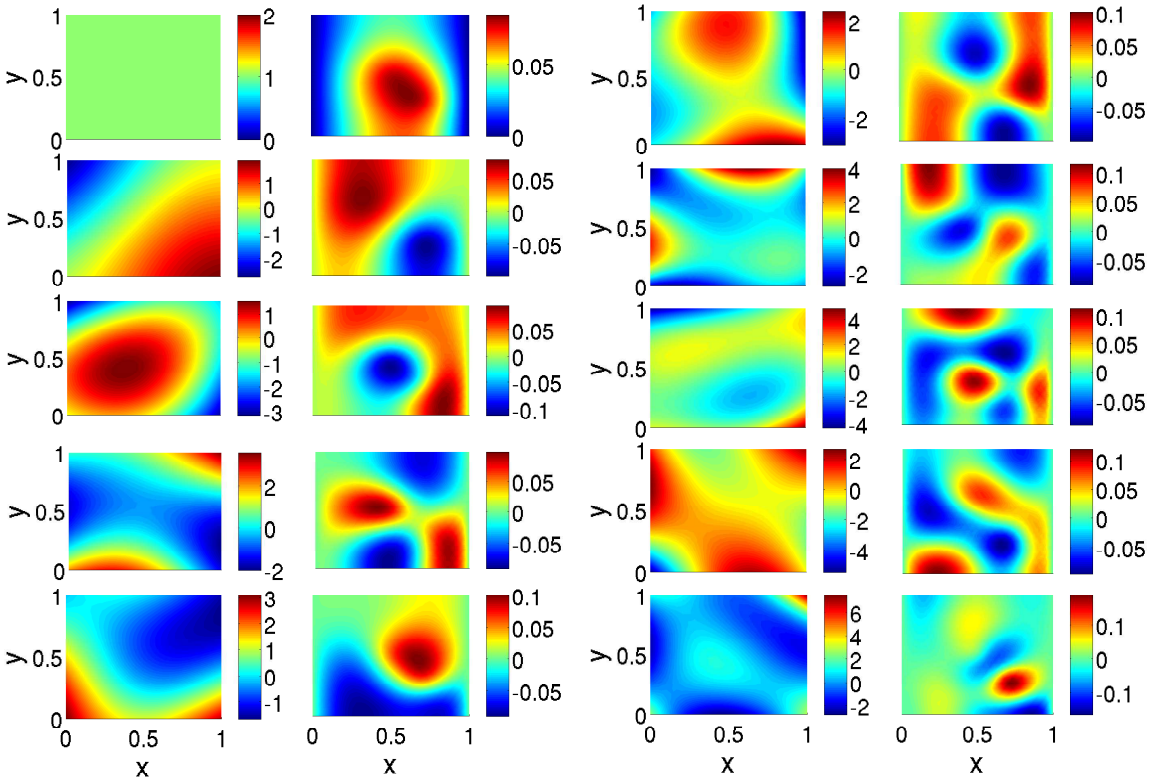


Figure 5-20: The parameter $q(x)$ (left) and state $u(x)$ (right) basis vectors obtained by the greedy sampling algorithm for the $N = 494$ test case. The first five basis vectors are pictured in the left column (a) and the last five on the right (b).

We now consider the reduced MCMC results. For a sequence of reduced-order model sizes, we plot the $\bar{q} \pm 2\sigma$ interval along with the mean itself as in Figure 5-19. We consider reduced-order models of size $n = 5$ and $n = 10$ and understand how the performance of the reduced implementation depends on the span of the reduced parameter space. For each set of results, we plot the projection of the actual parameter field onto the reduced subspace for reference. In particular, if our reduced subspace does not include the actual parameter, then we do not expect the proximity

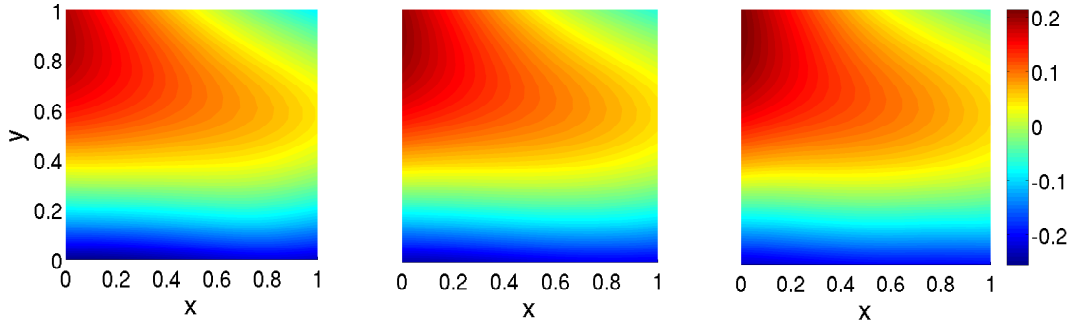
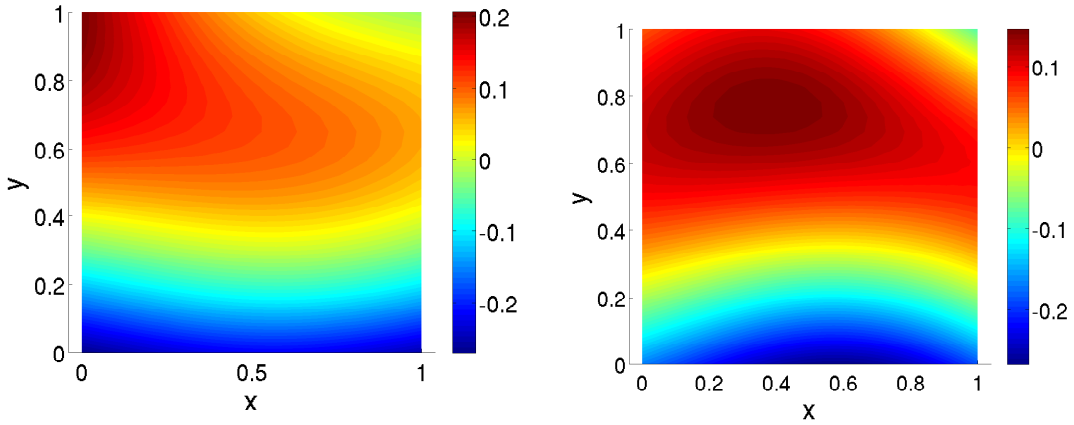


Figure 5-21: The results of the MCMC solution to the $N = 494$ test case with reduction to $n = 5$. The $\bar{q} - 2\sigma$ and $\bar{q} + 2\sigma$ interval bounds on the left and right, respectively. The mean is shown in the center. Actual parameter field shown in Figure 5-18. For reference, we show the projected actual parameter field below in Figure 5-22 (a).

of the results to be any closer than the true projection.

In Figure 5-20 we plot the parameter and state basis vectors in pairs. The first five pairs are shown in the first two columns (left) and the last five pairs in the two rightmost columns.



(a) (b)

Figure 5-22: The actual parameter field shown in Figure 5-18 projected onto the reduced parameter space for (a) $n = 5$ and (b) $n = 10$.

We now turn to the $n = 5$ case. The MCMC results are shown in Figure 5-21. Here, the reduced-order MCMC does not identify perfectly the actual parameter field, but it obtains a close shape and general trend as well as identifying the magnitude

closely. However, the sample variance is not large — note the proximity of $\bar{q} - 2\sigma$ and $\bar{q} + 2\sigma$. This is likely due to an inadequate subspace. If the span of the subspace does not include the posterior density then we will have errors in these predictions. It is important to note though that the solution here does correspond very closely with the actual parameter field when projected onto the reduced parameter space: see Figure 5-22 (a).

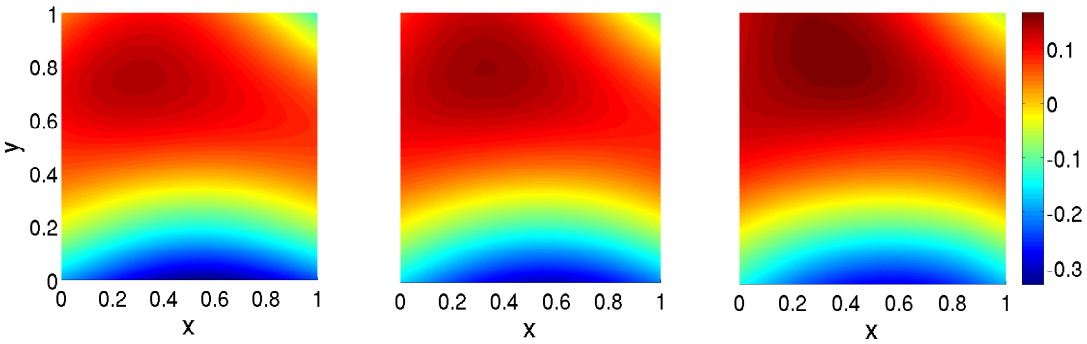


Figure 5-23: The results of the MCMC solution to the $N = 494$ test case with reduction to $n = 10$. The $\bar{q} - 2\sigma$ and $\bar{q} + 2\sigma$ interval bounds on the left and right, respectively. The mean is shown in the center. Actual parameter field shown in Figure 5-18. For reference, we show the projected actual parameter field below in Figure 5-22 (b).

In this case the MCMC sample mean matches well the actual parameter field. We hypothesize that this occurs because the basis vectors have identified the principal directions (in this case, fields) of the distribution. The effect is that the distribution becomes more Gaussian in reduced space. We cannot, however, guarantee that the interval $(q - 2\sigma, q + 2\sigma)$ is accurate. For the inclusion of this reduced MCMC into the decision-making under uncertainty framework we must understand how the error in the reduction produces errors in this interval. Based on observation in this work, it appears that the reduced implementation leads to a tighter interval than does the full-order MCMC. This is a drawback of the method. We would prefer a looser bound which could then be used in the robust optimization setting. If we base decisions on tight inaccurate bounds, we neglect a portion of the posterior density which may be important. Although this is far better than basing decisions on single-point estimates, we would like to do better.

Table 5.2: Number of full-order and reduced-order degrees of freedom and offline, online, and total time required to generate the results of Figure 5-11 on a DELL Latitude D530 with Intel Core 2Duo at 2 GHz. The offline time denotes the CPU time required to solve $n-1$ iterations of the greedy sampling problem with MATLAB's `fmincon`. In the MCMC simulation, the first 10% of the samples were discarded as burn-in. We used DRAM for the full-order implementation and Metropolis-Hastings (M-H) with no adaptivity for the reduced-order MCMC. For DRAM, it was more beneficial to run DRAM with more delayed rejection cycles and, in the $N = 494$ case, to discard the proposal adaptivity altogether.

N	N_o	n	Offline time (s)	Method	Samples	Online time (s)	Total time (s)
59	5	–	–	DRAM	500 000	1.1015×10^3	1.1015×10^3
59	5	5	2.0448×10^1	M-H	150 000	1.5664×10^2	1.7709×10^2
494	49	–	–	DRAM	500 000	1.6702×10^4	1.6702×10^4
494	49	5	4.6913×10^3	M-H	200 000	3.3169×10^2	5.0230×10^3
494	49	10	1.0384×10^4	M-H	200 000	4.9238×10^2	1.0876×10^4

Table 5.2 presents the CPU timings for the results in 2-D. We see similar results to those of Table 5.1. We recognize again that the offline cost, the greedy sampling procedure to build the reduced-order model, contributes almost entirely to the total time. The MCMC sampling process, on the other hand, is accelerated by almost two orders of magnitude. This data also demonstrates that a doubling from $n=5$ to $n=10$ reduced basis vectors results in an approximately 50% increase in the online time. Therefore, for relatively small reduced-order models,the online time will not grow quickly with increased fidelity of the reduced-order model.

Consider now the $n = 10$ case. Now we have more basis vectors and the actual parameter field can be better resolved. In Figure 5-22 (b) we plot the projected parameter field on this more enriched subspace. Compare this result with that of Figure 5-22. Since we can resolve the actual parameter field better with ten basis vectors, we expect the reduced MCMC results to be more accurate. The results are plotted in Figure 5-23.

5.5 Complexity analysis

We now consider the asymptotic complexity analysis for the full- and reduced-order implementations. We reiterate here that the CPU timing results presented herein

demonstrate some overall speedup for the reduced-order MCMC, but that we test the methodology for small model problems for which we can obtain full-order results. The models of interest have many more degrees of freedom and the full-order MCMC is computationally intractable. We show here that the reduced-order MCMC strategy will show significant savings over full-order MCMC in the large limit.

The asymptotic complexity is broken down for the offline and online portions, and then further by subtask in Table 5.3. We assume here that the number of MCMC samples scales quadratically with the number of parameter dimension [65]. In practice, adaptive samplers may be used in both scenarios to reduce the complexity of this task. As pointed out above, however, we expect to develop more efficient samplers for the reduced parameter dimension; therefore, we find additional savings for the reduced implementation.

Table 5.3: Complexity analysis for solution to the statistical inverse problem by MCMC sampling in full- and reduced-order. We assume here that the number of MCMC samples scales quadratically with the number of parameter dimension. Let k be a small bounded constant corresponding to the number of iterations required by an iterative solver. The number c is the average number of neighbors of a given node in the mesh.

		Operation	Full-order	Reduced-order
1.	Offline	greedy cycles		n
2.		subcycle iters		m
3.		gradient		$\mathcal{O}(N_p n_p + Nn)$
4.		Hessian-vector		$\mathcal{O}(N_p n_p + Nn)$
5.		Newton step		$\mathcal{O}(kN_p n_p + kNn)$
6.		solve $\mathbf{u}(\mathbf{p})$		$\mathcal{O}(cN_p + kN)$
7.	Subtotal			$\mathcal{O}(km(N_p n_p n + Nn^2))$
8.	Online	MCMC samples	$\mathcal{O}(N_p^2)$	$\mathcal{O}(n_p^2)$
9.		proposal	$\mathcal{O}(kN_p N)$	$\mathcal{O}(kn_p n^2)$
10.		prop. eval.	$\mathcal{O}(N_p)$	$\mathcal{O}(n_p)$
11.		post. eval.	$\mathcal{O}(3cN_p + kN)$	$\mathcal{O}(kn_p n + kn^2)$
12.	Subtotal		$\mathcal{O}(kNN_p^3)$	$\mathcal{O}(kn_p^3 n^2)$
13.	Total		$\mathcal{O}(kNN_p^3)$	$\mathcal{O}(km(N_p n_p n + Nn^2) + kn_p^3 n^2)$

If we assume that the parameter and state representations have the same dimensionality for full and reduced cases, i.e. $N = N_p$ and $n = n_p$, then the full-order cost is $\mathcal{O}(kN^4)$ and the reduced-order cost is $\mathcal{O}(kmNn^2 + kn^5)$. In the large limit, we

expect the number of reduced basis vectors required to capture the finite number of important directions to remain constant. That is, $\lim_{N\to\infty} n = c < \infty$. Therefore, for N sufficiently large, $Nn^2 > n^5$. This implies that the dominant cost of the reduced-order approach is the greedy sampling procedure, $\mathcal{O}(kmNn^2)$. The cost of the reduced approach scales linearly with the number of full-order degrees of freedom. In comparison with the full-order implementation, we have a speedup of three orders of magnitude in this large limit.

Suppose N is large, then with the full-order implementation we expect an approximate 16-fold increase in runtime when we double N. With the reduced-order model implementation, we expect m and n to be roughly constant with changes in N when N is large, thus we expect just a doubling in the total time for the reduced-order model implementation.

Chapter 6

Conclusion

In this chapter, we summarize the major findings of this work and present several interesting directions for future research. In Section 6.1 we review the process for reduced-order MCMC with reductions in parameter and state. Suggestions for future work are outlined in Section 6.2.

6.1 Summary

The solution to the statistical inverse problem is the first step in the decision-making under uncertainty framework outlined in Chapter 1. In this thesis, we consider the inverse problem for linear systems whose forward operator is linearly dependent on a field parameter. For such systems, high-fidelity models are very high-dimensional in both parameter and state space. The usual multi-query solution of the statistical inverse problem by MCMC is then no longer computationally tractable. Instead, we use a novel approach which seeks reduction in both parameter and state.

Whereas MCMC in high-dimensional spaces requires adaptive measures to yield reasonable solutions, traditional Metropolis-Hastings non-adaptive samplers can be utilized in the low-dimensional parameter space. The combination of speed-up obtained by foregoing adaptivity and the acceleration of forward evaluations resulting from the reduction in state yield a solution methodology which is very efficient. Furthermore, if the reduced subspaces contain the parameter fields of interest, the results

are also accurate. To include the appropriate parameter fields in the subspace, we develop our reduced-order model via the greedy sampling procedure of Chapter 3.

In the greedy sampling procedure, we iteratively sample parameter fields which maximize the error between the high-fidelity and current reduced-order model outputs. With a regularization by the prior, we sample parameters which are admitted by our prior belief and which are observable by the sensor outputs. The resulting reduced-order model spans the important subspace of the prior and is sensitive to those parameters which give rise to high-energy outputs.

The resulting reduced-order models are tested against full-order model solutions of the 1-D and 2-D groundwater inverse problems for mesh dimensions as large as 501 degrees of freedom. While reduced-order models could continue to perform past this limit, MCMC simulations with full-order models greater than $N = 500$ degrees of freedom require many hours of hand-tuning and many samples to obtain proper results. In the aforementioned tests, the reduced-order models accurately identify the posterior distribution within the span of the reduced basis and do so in far less CPU time than the full-order implementation.

6.2 Future Work

Many interesting future research directions have come out of this work. Fully understanding all three steps of the decision-making under uncertainty process – the statistical inverse problem, uncertainty propagation, and stochastic programming – is essential to defining the requirements of the solution to the statistical inverse problem. The linear Gaussian case is amenable to existing methods, but the groundwater inverse problem is inherently non-Gaussian and non-linear. For these problems, we need more robust methods for all three steps.

In this thesis, we have seen two major disadvantages of current MCMC practice for non-Gaussian posteriors. Firstly, some moments of the posterior will be uninformative statistics, for example, when the mean occupies a region of parameter space with very low posterior probability. When the posterior is non-Gaussian, an approximation by

a Gaussian may be very inefficient, especially when that uncertainty is propagated through a complex model. Secondly, the posterior may be very difficult to explore efficiently with existing MCMC methods. Even with adaptive methods, schemes required many hours of hand-tuning to turn out reasonable results. Both of these problems may be addressed by a more comprehensive parametric description of the posterior.

The reduced-order models have been selected to satisfy the prior belief and sample those parameters which excite the sensor outputs. Although these are two requirements of the important subspace, we neglect the difficulties of MCMC sampling when we select the reduced bases. For these small model problems there is not an issue, but for more complex time-dependent problems, we may require many more reduced basis vectors and therefore will need an efficient MCMC sampler in a space with several hundred dimensions. In this regard, it may be advantageous to choose the reduced-order bases to force the posterior distribution to be Gaussian. This sort of requirement could be included in the objective function of the greedy algorithm.

If efficient MCMC sampling is required by the final decision-making under uncertainty framework, there appears to be a wealth of knowledge from large-scale optimization which can be leveraged. The concepts which may be extended include, but are not limited to, grid continuation for obtaining better starting points, homotopy schemes to make more regular the isoprobability contours, interior-point methods to account for non-trivial boundary conditions, and subspace iterations for local adaptivity. Many of these ideas will work well in concert with reduced-order modelling. In addition, higher-order derivative information may be incorporated to both improve the efficiency of the sampling process and develop a wider range of applicability for a response surface model.

Finally, we must devise proper *a posteriori* error quantification for the reduced MCMC procedure. This is a daunting task considering the full-order solution may be completely intractable. However, it may be possible to provide some bounds or at least some rules of thumb to determine how many basis vectors one must include, for example. Pursuit of this goal would include a complete treatment of error quantifica-

tion for Galerkin projection reduced-order models in addition to statistical sampling errors. An empirical approach would require a thorough parameter study.

Appendix A

Regularization and Prior

In Chapter 2, we proposed a connection between choice of regularization functional in the deterministic setting and that of the prior distribution in the statistical setting. Here we make this statement more concrete by demonstrating that the NLP arising from the deterministic setting with $T_1(p)$ regularization is equivalent to that which we get from computing the MAP estimate in the statistical setting with Gaussian priors defined by the stiffness matrix.

We previously defined the T_1 regularization functional

$$T_1(p) = \int_\Omega \nabla p \cdot \nabla p \, d\Omega.$$

The deterministic inverse problem formulation leads to the following NLP

$$\mathbf{p}^* = \arg\min_{\mathbf{p}\in\mathcal{D}} \mathcal{J} = \frac{1}{2}\|\mathbf{y} - \mathbf{y}_d\|^2 + \frac{1}{2}\beta \mathbf{p}^T \mathbf{S}\mathbf{p}, \tag{A.1}$$

$$\text{subject to } \mathbf{f}(\mathbf{u};\mathbf{p}) = \mathbf{0}, \tag{A.2}$$

$$\mathbf{y} = \mathbf{y}(\mathbf{u}) \tag{A.3}$$

where we have discretized the regularization functional using the piecewise linear nodal basis functions defined in Chapter 2.

We now demonstrate that computing the MAP estimate from the statistical formulation leads to the same NLP. We utilize the Gaussian prior $\gamma_{\mathbf{p}}(\mathbf{p}) = \exp\left\{-\frac{1}{2}\beta \mathbf{p}^T \mathbf{S}\mathbf{p}\right\}$

and likelihood function $L(\mathbf{y}_d|\mathbf{p}) = \exp\left\{-\frac{1}{2}\|\mathbf{y}(\mathbf{p}) - \mathbf{y}_d\|_2^2\right\}$. The Bayesian formulation yields the posterior

$$\pi_{\mathbf{p}}(\mathbf{p}) \propto \exp\left\{-\frac{1}{2}\|\mathbf{y}(\mathbf{p}) - \mathbf{y}_d\|_2^2 - \frac{1}{2}\beta\mathbf{p}^T\mathbf{S}\mathbf{p}\right\}.$$

The MAP estimate is computed by solving the NLP

$$\begin{aligned}
\mathbf{p}^* &= \arg\max_{\mathbf{p}\in\mathcal{D}} \pi_{\mathbf{p}}(\mathbf{p}), \\
&= \arg\max_{\mathbf{p}\in\mathcal{D}} \exp\left\{-\frac{1}{2}\|\mathbf{y}(\mathbf{p}) - \mathbf{y}_d\|_2^2 - \frac{1}{2}\beta\mathbf{p}^T\mathbf{S}\mathbf{p}\right\}, \\
&= \arg\min_{\mathbf{p}\in\mathcal{D}} \frac{1}{2}\|\mathbf{y}(\mathbf{p}) - \mathbf{y}_d\|_2^2 + \frac{1}{2}\beta\mathbf{p}^T\mathbf{S}\mathbf{p}
\end{aligned}$$

where we used the positive semidefiniteness of $\mathbf{S}$. The last step follows from $\|\cdot\|_2^2 \geq 0$ and $\mathbf{p}^T\mathbf{S}\mathbf{p} \geq 0$. Note the equivalence to (A.1) as originally stated.

Appendix B

Positivity in reduced parameter space

In this appendix, we prove the connectedness of the region in reduced parameter space corresponding to the positivity constraints on the reconstructed parameter. Let $\mathbf{P} \in \mathbb{R}^{n \times m}$ be a basis for a subspace of $\mathbb{R}^n$ with $n > m$. Furthermore, let $\mathbf{p} \in \mathbb{R}^m$ be a vector of coefficients such that $\mathbf{Pp}$ is a discrete reconstruction of a parameter. We now state and prove the connectedness of the region $\mathcal{D} \subset \mathbb{R}^m$ such that for all $\mathbf{p} \in \mathcal{D}$ we have $\mathbf{Pp} > 0$.[1]

Theorem Let $\mathbf{P} \in \mathbb{R}^{n \times m}$. If there exists a vector $\mathbf{p} \in \mathbb{R}^m$ such that $\mathbf{Pp} > 0$, then $\mathbf{p}$ is contained in one connected region $\mathcal{D} \subset \mathbb{R}^m$.

Proof: The key to the proof is to interpret the matrix-vector product not as an expansion in the columns of $\mathbf{P}$ but rather as a set of inner products. Let $\mathbf{a}_i^T$ be the ith row of $\mathbf{P}$. Consider in turn the subspaces of $\mathbb{R}^m$ for which $\mathbf{v} \in \mathbb{R}^m$ satisfy $\mathbf{a}_i^T \mathbf{v} > 0$. Each $\mathbf{a}_i^T \mathbf{v} = 0$ defines a hyperplane in $\mathbb{R}^m$ passing through the origin; this hyperplane defines a halfspace in which $\mathbf{a}_i^T \mathbf{v} > 0$. We now consider a sequence of restrictions of $\mathbb{R}^m$. Let $\mathcal{D}_i = \bigcap_{j=1}^{i} \{\mathbf{v} \mid \mathbf{a}_j^T \mathbf{v} > 0\}$.

Without loss of generality we begin with $\mathbf{a}_1^T$, the first row of $\mathbf{P}$. In particular, $\mathbf{a}_1^T \mathbf{v} > 0$ defines a halfspace of $\mathbb{R}^m$. We have immediately $\mathcal{D}_1 = \{\mathbf{v} \mid \mathbf{a}_1^T \mathbf{v} > 0\}$. With

[1] We use the greater than sign here to indicate the conditional applied to every element of the resulting vector. That is, $\mathbf{v} > 0$ if and only if $v_i > 0$ for all i.

the inclusion of each row vector $\mathbf{a}_i^T$, the space gets more and more restrictive. Let this restriction process occur for the first $i < n$ row vectors of $\mathbf{P}$ and consider the next restriction. There are two possibilities for $\mathcal{D}_{i+1}$: either (1) $\mathcal{D}_{i+1} \subset \mathcal{D}_i$ or (2) $\mathcal{D}_{i+1} = \emptyset$. This bifurcation is general to the restriction at any step.

If $\mathcal{D}_{i+1}$ is empty, then there *does not* exist a vector $\mathbf{p} \in \mathbb{R}^m$ such that $\mathbf{Pp} > 0$ and the basis is insufficient for representation of positive quantities. If $\mathcal{D}_{i+1} \subset \mathcal{D}_i$ for every $i = 1, 2, \ldots, n-1$, then we have $\mathcal{D}_n = \mathcal{D}$ and for all $\mathbf{p} \in \mathcal{D}$ we have $\mathbf{Pp} > 0$ as desired.

We now demonstrate that if $\mathcal{D} \neq \emptyset$ that any $\mathbf{q} \notin \mathcal{D}$ violates $\mathbf{Pq} > 0$. For such a vector $\mathbf{q}$ we must have at least one index $i \in \{1, 2, \ldots, n\}$ such that $\mathbf{a}_i^T \mathbf{q} \leq 0$; therefore $\mathbf{Pq} \not> 0$. This demonstrates that there cannot exist another region $\mathcal{F} \in \mathbb{R}^m$ such that $\mathbf{p} \in \mathcal{F}$ satisfy $\mathbf{Pp} > 0$; and therefore, $\mathcal{D}$ is connected and the only set containing parameter vectors whose reconstruction satisfies the positivity constraint.

Appendix C

Grid continuation

In Chapter 3 it is pointed out that the greedy sampling procedure can become an especially challenging task when the regularization (alternatively, prior) term is outweighed by the mismatch in the objective function. We recommend grid continuation as an approach to mitigate the resulting difficulties. In this appendix, we highlight the procedure one might utilize to overcome these challenges. Please see [5] for further details.

Perhaps the most well-known analogy to grid continuation in the optimization setting is that of homotopy methods in Newton's method for solving nonlinear systems. Homotopy methods use continuous deformations of the original problem to form a sequence of subproblems. In the nonlinear systems root-finding setting, the problem is solved in series. First, the nonlinear function is altered greatly to achieve a more amenable system for Newton's method. Then Newton's method is converged on that subproblem. The solution to that subproblem is then utilized as the initial condition to the next problem in the sequence, which is driven closer and closer to the original problem in succession. The result is that the sequence of subproblems (hopefully) leads to a starting point for the original problem which resides in the basin of attraction of the root of the original system. If the homotopic sequence is chosen appropriately, Newton's method may only require a handful of iterations to converge for each subproblem.

Grid continuation represents a similar method in nonlinear programming. In some

ways, the method utilizes the simplest homotopy scheme available. We define several levels of discretization of the field over which we wish to optimize. In practice, we may have five meshes of varying fineness. It is usually best to select the coarsest mesh to be one on which the optimization problem can be solved inexpensively by brute force, to ensure a good start. Then we interpolate the solution onto the next finest mesh. This interpolation becomes a starting point for our optimizer at this level of discretization. That solution will be interpolated and used as an initial condition for the next mesh, and so on. The process continues until the optimization problem has been solved on the finest mesh in the hierarchy. Similarly to the homotopy schemes for Newton's method, the goal is to find a good initial guess for the optimization problem at each step. Once again, if the mesh discretizations are not drastically different from step to step, one can hope to require relatively few optimization iterations at each level. Although we cannot prove anything rigorous about this method in most cases, it has been shown to yield better results than attempting the full-scale optimization problem immediately from a random starting location [5].

Bibliography

[1] V. Akcelik, G. Biros, A. Draganescu, O. Ghattas, J. Hill, and B. van Bloemen Waanders. Dynamic data-driven inversion for terascale simulations: real-time identification of airborne contaminants. In *Proceedings of the 2005 ACM/IEEE conference on Supercomputing*, 2005.

[2] S.R. Arridge. Optical tomography in medical imaging. *Inverse Problems*, 15:R41–R93, 1999.

[3] S.R. Arridge and J.C. Hebden. Optical imaging in medicine: II. Modeling and reconstruction. *Physics of Medicine and Biology*, 42:841–853, 1997.

[4] S.R. Arridge, J.P. Kaipio, V. Kolehmainen, M. Schweiger, E. Somersalo, T. Tarvainen, and M. Vauhkonen. Approximation errors and model reduction with an application in optical diffusion tomography. *Inverse Problems*, 22:175–195, 2006.

[5] U.M. Ascher and E. Haber. Grid refinement and scaling for distributed parameter estimation problems. *Inverse Problems*, 17:571–590, 2001.

[6] Y.F. Atchadé. An adaptive version for the Metropolis adjusted Langevin algorithm with a truncated drift. *Methodology and Computing in Applied Probability*, 8:235–254, 2006.

[7] O. Bashir, K. Willcox, O. Ghattas, B. van Bloemen Waanders, and J. Hill. Hessian-based model reduction for large-scale systems with initial-condition inputs. *International Journal of Numerical Methods in Engineering*, 73:844–868, 2008.

[8] L.T. Biegler, J. Nocedal, and C. Schmid. A reduced Hessian method for large-scale constrained optimization. *SIAM Journal of Optimization*, 5(2):314–347, 1995.

[9] L.T. Biegler, J. Nocedal, C. Schmid, and D. Ternet. Numerical experience with a reduced Hessian method for large scale constrained optimization. *Computational Optimization and Applications*, 15:45–67, 2000.

[10] L. Borcea. Electrical impedance tomography. *Inverse Problems*, 18:R99–R136, 2002.

[11] T. Bui-Thanh. *Model-Constrained Optimization Methods for Reduction of Parameterized Large-Scale Systems*. PhD thesis, Massachusetts Institute of Technology, Department of Aeronautics and Astronautics, 2007.

[12] T. Bui-Thanh, K. Willcox, and O. Ghattas. Model reduction for large-scale systems with high-dimensional parametric input space. *SIAM Journal of Scientific Computing*, 30(6):3270–3288, 2008.

[13] C. Burstedde, O. Ghattas, J. Martin, and L. Wilcox. Uncertainty quantification in inverse problems with stochastic Newton MCMC. To be submitted, 2009.

[14] D. Calvetti and E. Somersalo. *Introduction to Bayesian Scientific Computing: Ten Lectures on Subjective Computing*. Surveys and Tutorials in the Applied Mathematical Sciences (2). Springer, 2007.

[15] J. Carrera, A. Alcolea, A. Medina, J. Hidalog, and L.J. Slooten. Multiresolution parameterization for geophysical inverse problems. *Hydrogeology Journal*, 13:206–222, 2005.

[16] W. Chen, L. Baghdasaryan, T. Buranathiti, and J. Cao. Model validation via uncertainty propagation and data transformations. *AIAA Journal*, 42(7):1406–1415, July 2004.

[17] W. Chen, R. Jin, and A. Sudjianto. Analytical variance-based global sensitivity analysis in simulation-based design under uncertainty. *Journal of Mechanical Design*, 127(5):875–886, September 2005.

[18] L-Y Chiao and W-T Liang. Multiresolution parameterization for geophysical inverse problems. *Geophysics*, 68(1):199–209, 2003.

[19] T.F. Coleman and Y. Li. On the convergence of reflective Newton methods for large-scale nonlinear minimization subject to bounds. *Mathematical Programming*, 67(2):189–224, 1994.

[20] T.F. Coleman and Y. Li. An interior, trust region approach for nonlinear minimization subject to bounds. *SIAM Journal on Optimization*, 6:418–445, 2005.

[21] D. Colton, H. Haddar, and M. Piana. The linear sampling method in inverse electromagnetic scattering theory. *Inverse Problems*, 19:S105–S137, 2003.

[22] D. Colton and A. Kirsch. A simple method for solving inverse scattering problems in the resonance region. *Inverse Problems*, 12:383–393, 1996.

[23] A.E. Deane, I.G. Kevrekidis, G.E. Karniadakis, and S.A. Orszag. Low-dimensional models for complex geometry flows: Application to grooved channels and circular cylinders. *Physics of Fluids*, 3(10):2337–2354, 1991.

[24] Y. Efendiev, T. Hou, and W. Luo. Preconditioning Markov chain Monte Carlo simulations using coarse-scale models. *SIAM Journal of Scientific Computing*, 28(2):776–803, 2006.

[25] H.W. Engl, M. Hanke, and A. Neubauer. *Regularization of Inverse Problems*. Springer, 1996.

[26] C.L. Epstein. *Mathematics of Medical Imaging*. Prentice Hall, 2003.

[27] L.D. Faddeyev and B. Seckler. The inverse problem in the quantum theory of scattering. *Journal of Mathematical Physics*, 4(1):72–104, 1963.

[28] P. Feldmann and R.W. Freund. Efficient linear circuit analysis by Pade approximation via the Lanczos process. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 14:639–649, 1995.

[29] K. Gallivan, E. Grimme, and P. Van Dooren. Padé approximation of large-scale dynamic systems with Lanczos methods. In *Proceedings of the 33rd IEEE Conference on Decision and Control*, December 1994.

[30] A. Gelman, G.O. Roberts, and W.R. Gilks. Efficient metropolis jumping rules. In J.M. Bernardo, J.O. Berger, A.F. David, and A.F.M. Smith, editors, *Bayesian Statistics*, volume 5, pages 599–608. Oxford University Press, 1996.

[31] P.J. Green and A. Mira. Delayed rejection in reversible jump metropolis-hastings. *Biometrika*, 88:1035–1053, 2001.

[32] M. Grepl and A. Patera. A posteriori error bounds for reduced-basis approximations of parametrized parabolic differential equations. *ESAIM Mathematical Modelling and Numerical Analysis*, 39:157–181, 2005.

[33] E. Grimme. *Krylov Projection Methods for Model Reduction*. PhD thesis, University of Illinois at Urbana-Champaign, Coordinated Science Laboratory, 1997.

[34] S. Gugercin and A. Antoulas. A survey of model reduction by balanced truncation and some new results. *International Journal of Control*, 77:748–766, 2004.

[35] H. Haario, M. Laine, A. Mira, and E. Saksman. DRAM: Efficient adaptive MCMC. *Statistics and Computing*, 16:339–354, 2006.

[36] H. Haario, E. Saksman, and J. Tamminen. Componentwise adaptation for high dimensional MCMC. *Computational Statistics*, 20:265–273, 2005.

[37] E. Haber. Quasi-Newton methods for large-scale electromagnetic inverse problems. *Inverse Problems*, 21:305–323, 2005.

[38] J. Hadamard. *Lectures on the Cauchy Problem in Linear Partial Differential Equations*. Yale University Press, 1923.

[39] W.K. Hastings. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57(1):97–109, April 1970.

[40] P. Holmes, J.L. Lumley, and G. Berkooz. *Turbulence, coherent structures, and dynamical systems and symmetry*. Cambridge University Press, 1996.

[41] M.G. Ierapetritou, J. Acevedo, and E.N. Pistikopoulos. An optimization approach for process engineering problems under uncertainty. *Computers and Chemical Engineering*, 20(6):703–709, 1996.

[42] J. Kaipio and E. Somersalo. Statistical inverse problems: Discretization, model reduction and inverse crimes. *Journal of Computational and Applied Mathematics*, 198:493–504, 2007.

[43] J.P. Kaipio, V. Kolehmainen, E. Somersalo, and M. Vauhkonen. Statistical inversion and Monte Carlo sampling methods in electrical impedance tomography. *Inverse Problems*, 16:1487–1522, 2000.

[44] M.C. Kennedy and A. O'Hagan. Bayesian calibration of computer models. *Journal of the Royal Statistical Society B.*, 63(3):425–464, 2001.

[45] P.K. Kitanidis. On the geostatistical approach to the inverse problem. *Advances in Water Resources*, 19(6):333–342, December 1996.

[46] J. Kittler and P.C. Young. A new approach to feature selection based on the karhunen-loeve expansion. *Pattern Recognition*, 5:335–352, 1973.

[47] O. Kwon, E.J. Woo, J-R Yoon, and J.K. Seo. Magnetic Resonance Electrical Impedance Tomography (MREIT): Simulation study of J-substitution algorithm. *IEEE Transactions on Biomedical Engineering*, 49(2):160–167, 2002.

[48] S. Lall, J.E. Marsden, and S. Glavaski. A subspace approach to balanced truncation for model reduction of nonlinear control systems. *International Journal of Robust and Nonlinear Control*, 12:519–535, 2002.

[49] A. Levin, V. Wieland, and J.C. Williams. The performance of forecast-based monetary policy rules under model uncertainty. *The American Economic Review*, 93(3):622–645, June 2003.

[50] J. Li and J. White. Low rank solution of Lyapunov equations. *SIAM Journal on Matrix Analysis and Applications*, 24(1):260–280, 2002.

[51] D.C. Liu and J. Nocedal. On the limited memory BFGS method for large scale optimization. *Mathematical Programming*, 45:503–528, 1989.

[52] J.S. Liu, F.Liang, and W.H. Wong. The multiple-try method and local optimization in Metropolis sampling. *Journal of the American Statistical Association*, 44(247):121–134, March 2000.

[53] A.K. Louis. Medical imaging: state of the art and future development. *Inverse Problems*, 8:709–738, 1992.

[54] Y.M. Marzouk, H.N. Najm, and L.A. Rahn. Stochastic spectral methods for efficient Bayesian solution of inverse problems. *Journal of Computational Physics*, 224:560–586, 2007.

[55] J. McFarland and S. Mahadevan. Multivariate significance testing and model calibration under uncertainty. *Computer Methods in Applied Mechanics and Engineering*, 197:2467–2479, 2008.

[56] J. McFarland, S. Mahadevan, V. Romero, and L. Swiler. Calibration and uncertainty analysis for computer simulations with multivariate output. *AIAA Journal*, 46(5):1253–1265, 2008.

[57] N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller, and E. Teller. Equations of state calculations by fast computing machines. *Journal of Chemical Physics*, 21(6):1087–1092, 1953.

[58] A.M. Michalak and P.K. Kitanidis. A method for enforcing parameter nonnegativity in bayesian inverse problems with an application to contaminant source identification. *Water Resources Research*, 39(2):7–1–7–14, 2003.

[59] M. Monde. Analytical method in inverse heat transfer problem using Laplace transform technique. *International Journal of Heat and Mass Transfer*, 43:3965–3975, 2000.

[60] B.C. Moore. Principal component analysis in linear systems: controllability, observability, and model reduction. *IEEE Transactions on Automatic Control*, 26(1):17–32, 1981.

[61] H.N. Najm. Uncertainty quantification and polynomial chaos techniques in computational fluid dynamics. *Annual Review of Fluid Mechanics*, 41:35–52, 2009.

[62] A.M. Obukhov. Statistically homogeneous fields on a sphere. *Usp. Mat. Navk.*, 2:196–198, 1947.

[63] M.N. Ozisik and H.R.B. Orlande. *Inverse Heat Transfer*. Taylor & Francis, 2000.

[64] T. Penzl. Algorithms for model reduction of large dynamical systems. *Linear Algebra and its Applications*, 415(2):322–343, June 2006.

[65] G.O. Roberts and J.S. Rosenthal. Optimal scaling for various Metropolis-Hastings algorithms. *Statistical Science*, 16(4):351–367, November 2001.

[66] G.O. Roberts and J.S. Rosenthal. Examples of adaptive MCMC. *Preprint*, 2008. Available at http://probability.ca/jeff/ftpdir/adaptex.pdf.

[67] N.V. Sahinidis. Optimization under uncertainty: state-of-the-art and opportunities. *Computers and Chemical Engineering*, 28:971–983, 2004.

[68] L. Sirovich. Turbulence and the dynamics of coherent structures. part 1: Coherent structures. *Quarterly of Applied Mathematics*, 45(3):561–571, October 1987.

[69] D.C. Sorensen and A.C. Antoulas. The Sylvester equation and approximate balanced reduction. *Linear Algebra and its Applications*, 351:671–700, 2002.

[70] K. Veroy and A. Patera. Certified real-time solution of the parametrized steady incompressible navier-stokes equations: Rigorous reduced-basis a posteriori error bounds. *International Journal for Numerical Methods in Fluids*, 47:773–788, 2005.

[71] K. Veroy, C. Prud'homme, D. Rovas, and A. Patera. A posteriori error bounds for reduced-basis approximation of parametrized noncoercive and nonlinear elliptic partial differential equations. In *Proceedings of the 16th AIAA Computational Fluid Dynamics Conference*, Orlando, FL, 2003.

[72] J. Wang and N. Zabaras. Hierarchical Bayesian models for inverse problems in heat conduction. *Inverse Problems*, 21:183–206, 2005.