Goal-oriented inference: Theoretical foundations and application to carbon capture and storage

by

Chad Lieberman

Submitted to the Department of Aeronautics & Astronautics
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2013

Author . . . .
Department of Aeronautics & Astronautics
May 15, 2013

Certified by . . . .
Karen Willcox
Professor, Department of Aeronautics & Astronautics
Thesis Supervisor

Certified by . . . .
Youssef Marzouk
Associate Professor, Department of Aeronautics & Astronautics
Committee Member

Certified by . . . .
Tommi Jaakkola
Professor, Department of Electrical Engineering & Computer Science
Committee Member

Accepted by . . . .
Eytan H. Modiano
Professor, Department of Aeronautics & Astronautics
Chair, Graduate Program Committee

Goal-oriented inference: Theoretical foundations and application to carbon capture and storage

by

Chad Lieberman

Submitted to the Department of Aeronautics & Astronautics
on May 15, 2013, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Abstract

Many important engineering problems require computation of prediction output quantities of interest that depend on unknown distributed parameters of the governing partial differential equations. Examples include prediction of concentration levels in critical areas for contamination events in urban areas and prediction of trapped volume of supercritical carbon dioxide in carbon capture and storage. In both cases the unknown parameter is a distributed quantity that is to be inferred from indirect and sparse data in order to make accurate predictions of the quantities of interest. Traditionally parameter inference involves regularization in deterministic formulations or specification of a prior probability density in Bayesian statistical formulations to resolve the ill-posedness manifested in the many possible parameters giving rise to the same observed data. Critically, the final prediction requirements are not considered in the inference process.

Goal-oriented inference, on the other hand, utilizes the prediction requirements to drive the inference process. Since prediction quantities of interest are often very low-dimensional, the same ill-posedness that stymies the inference process can be exploited when inference of the parameter is undertaken solely to obtain predictions. Many parameters give rise to the same predictions; as a result, resolving the parameter is not required in order to accurately make predictions. In goal-oriented inference, we exploit this fact to obtain fast and accurate predictions from experimental data by sacrificing accuracy in parameter estimation.

When the governing models for experimental data and prediction quantities of interest depend linearly on the parameter, a linear algebraic analysis reveals a dimensionally-optimal parameter subspace within which inference proceeds. Parameter estimates are inaccurate but the resulting predictions are identical to those achieved by first performing inference in the full high-dimensional parameter space and then computing predictions. The analysis required to identify the parameter subspace reveals inefficiency in experiment and sources of uncertainty in predictions, which can also be utilized in experimental design. Linear goal-oriented inference is demonstrated on a model problem in contaminant source inversion and prediction.

In the nonlinear setting, we focus on the Bayesian statistical inverse problem formulation where the target of our goal-oriented inference is the posterior predictive probability density function representing the relative likelihood of predictions given the observed experimental data. In many nonlinear settings, particularly those involving nonlinear partial differential equations, distributed parameter estimation remains an unsolved problem. We circumvent estimation of the parameter by establishing a statistical model for the joint density of experimental data and predictions using either a Gaussian mixture model or kernel density estimate derived from simulated experimental data and simulated predictions based on parameter samples from the prior distribution. When experiments are conducted and data are observed, the statistical model is conditioned on the observed data, and the posterior predictive probability density is obtained. Nonlinear goal-oriented inference is applied to a realistic application in carbon capture and storage.

Thesis Supervisor: Karen Willcox
Title: Professor, Department of Aeronautics & Astronautics

Committee Member: Youssef Marzouk
Title: Associate Professor, Department of Aeronautics & Astronautics

Committee Member: Tommi Jaakkola
Title: Professor, Department of Electrical Engineering & Computer Science

Acknowledgments

Although my name appears alone at the top of the cover page, this work would not have been possible without the support of many people.

I cannot begin to express in words my gratitude to my advisor for her balance of steadfast encouragement and kind compassion. No matter how terrible I was feeling about the progress of my research, meeting with Karen made me feel better. At the same time, she pushed me to achieve what I thought at times would be impossible. And I'm so glad she did. I have learned so much from working with Karen for the last seven years in a variety of capacities from UROP to RA to TA to Crosslinks Co-Founder. They are lessons that I will carry with me for the rest of my life.

I would also like to thank the other members of my thesis committee. Youssef and Tommi both provided terrific insight and asked important questions throughout my research, and this thesis is much better for it. My gratitude is also due to Lior Horesh and Bart van Bloemen Waanders for their comments and questions on the manuscript, which have also served to improve it tremendously.

This thesis would not have been possible without a strong support structure outside of work as well. I cannot thank my family enough, who I know would have supported me in any outcome, but with whom I am so proud to share this achievement. I appreciate their compassion and understanding and their tolerance for the roller coaster of emotions that always tracked my research progress so well. Also important to my success was the friendship of those in ACDL, who have really become like a second family to me. Finally, my gratitude is owed to my PKS brothers, both at MIT and across the world, who always help me keep things in perspective.

This work was supported in part by the MIT Center for Computational Engineering Kambourides Fellowship, the Air Force Office of Scientific Research MURI Program (Program Manager Dr. F. Fahroo), the Department of Energy Applied Mathematics Program (Grant DE-FG02-08ER25858, Program Manager A. Landsberg), and the DiaMonD MMICC Center (DOE Award Number 021551-001, Program Manager A. Landsberg).

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Many years of research have focused on the development of algorithms for estimating distributed parameters of mathematical models for physical systems. Models are frequently based on partial differential equations (PDEs). Experiments are performed, and parameters must be estimated from observed data [7]. In these cases, inference is typically severely ill-posed due to the relatively high-dimensionality of parameters compared to that of the observed data [38]. Problem formulations are either deterministic or statistical, and both require special attention to the ill-posedness and the extensive computational resources required to estimate the parameter [75, 79]. Recognizing that estimation of parameters is a step in pursuit of making predictions,[1] and subsequently decisions, we establish goal-oriented inference, a novel approach to estimation when the goal is to obtain accurate predictions from data without regard for accuracy in parameter estimation. In short, we refocus resources toward estimating the predictions, the true target of the parameter estimation. By exploiting the low-dimensionality of the map from data to predictions, we expose weaknesses in the experimental design, discover the major sources of uncertainty in the predictions, and circumvent the most expensive online computations, making feasible pseudo real-time deployment.

Motivation for goal-oriented inference is discussed at greater length in section 1.1.

[1] It should be noted that there are fields where parameter estimation is the goal and predictions either do not exist or are not apparent. These settings usually have a flavor of scientific discovery rather than the engineering context we treat in this work.

In section 1.2 we define the key terminology of the work. Current practice in the estimation of distributed parameters is discussed in section 1.3. We give particular attention to the challenges of PDE-constrained inverse problems in section 1.4. In section 1.5 we highlight the recent advances and trends in the research community to focus on quantities of interest, the predictions in the context of goal-oriented inference.

1.1 Motivation for goal-oriented inference

When predictions depend on a system with an unknown distributed parameter, a typical approach is to perform parameter inference on collected data before passing the estimate to the prediction simulation. Data informative about the parameter are first observed. An inference procedure, deterministic or statistical, is then employed to make an estimate of the parameter based on the observed data and an assumed model. The parameter estimate is then used as an input to a simulation that will make predictions.

The inference step is ill-posed and computationally very expensive. In deterministic formulations of the inverse problem, a solution is determined by regularizing an objective function minimizing the mismatch between observed data and model-predicted data [44]. The field of regularization theory has developed to address ill-posedness [27, 40]. While the solution of such inverse problems is very well understood compared to its statistical counterpart, the computational cost for PDE-constrained problems still prohibits pseudo real-time solution, limiting the applicability of this strategy.

Bayesian statistical formulations treat ill-posedness in the inverse problem by specifying a prior distribution over the parameter space, a representation of one's belief in the relative likelihood of different parameters before data are observed [72, 45]. In the limited data context, however, this prior distribution will not be forgotten through Bayesian updates; its effect remains even after all data are processed [74]. There are many computational challenges of Bayesian inference of distributed parameters for models described by PDEs. Markov chain Monte Carlo (MCMC) methods are a popular technique for exploring the posterior distribution of the Bayesian inference [80, 54, 58, 42, 76, 13, 33]. However, MCMC requires efficient exploration of parameter space and many samples, meaning many PDE solves. Efficient exploration is challenging in high-dimensional parameter spaces and many PDE solves makes the computation intractable for the pseudo real-time setting.

These challenges can be addressed by incorporating the final objectives of the inference into the problem statement. In many engineering applications, typically parameter estimation is not the goal but rather a critical intermediate step in making predictions of the system under different operating conditions. We propose a new approach to the parameter estimation problem that focuses on accuracy in those predictions. We find that exploiting the low-dimensional map from observed data to predictions allows us to circumvent many of the challenges mentioned above, making many high-dimensional problems amenable to real-time prediction in the deterministic setting and tractable prediction in the statistical setting.

1.2 Terminology and scope

There are two central components to the developments in goal-oriented inference: an experimental process by which data are obtained, and a prediction process yielding the target of our estimation.

An *experimental process* is a physical system, or model thereof, as well as an observation paradigm given by experimental design, that produces data depending on the existing, but unknown, parameter. The data are corrupted by noise, which we will regularly model as additive Gaussian. In this work, the data will be simulated using a model of a physical system; in practice, one would perform the experiments on the real world. Our applications are governed by PDEs. Therefore, the experimental process will consist of the composition of the numerical solution of a system of PDEs, determining state variables from a given parameter, and an observation operator, which yields uncorrupted data from the state variables.

A *prediction process* is a physical system, or model thereof, that yields an estimate of a quantity of interest given a specified value of model parameter. Although such a process usually is not modelled perfectly, we will assume there is no noise in the output. Like the experimental process, the prediction process will typically also consist of the composition of a PDE operator and an observation operator. The PDE need not be the same as the experimental process; however, the two must be linked by a consistent description of the unknown parameter.

A block diagram of the two processes, connected by the parameter, is shown in Figure 1-1. Goal-oriented inference will involve the exploitation of information content in observed data to make estimation of the prediction.

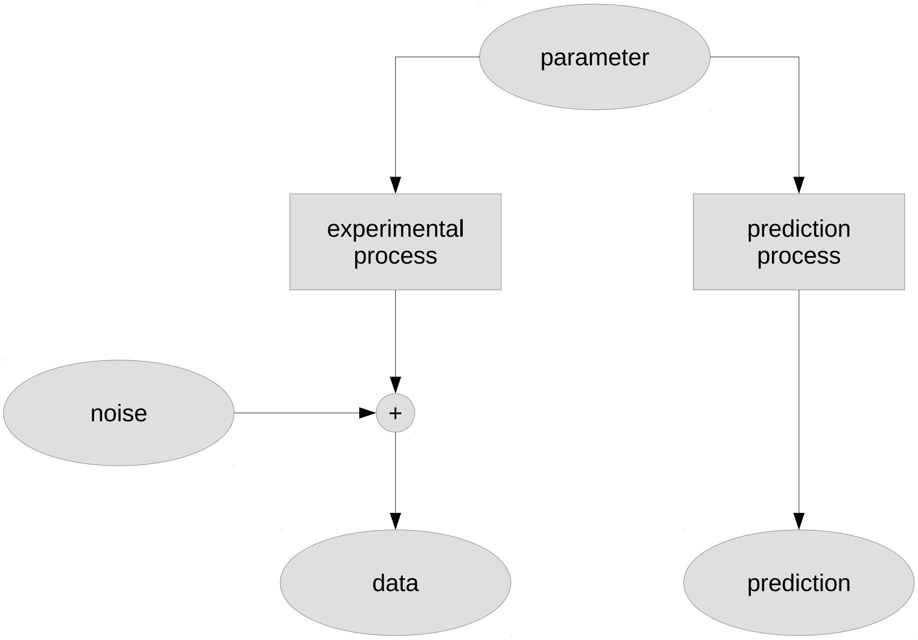


Figure 1-1: The experiment and prediction processes both take as input the shared parameter. The output of the experimental process is corrupted by noise to yield the observed data. The output of the prediction process yields the prediction quantity of interest.

For the purposes of this work, we do not consider model uncertainty in either the experimental or prediction processes. We focus solely on the uncertainty, or lack of information, about the parameter resulting from the unobservability of the processes.

For both processes, this is a result of a PDE operator with collapsing spectrum and observation with significant sparsity.

The models underlying the two processes are assumed to have outputs that vary smoothly with the parameter. We do not consider problems with discontinuities or parameters taking integer values. Parameters reside as a field in 1-, 2-, or 3-d in continuous form or as a vector of modal coefficients in an n-d Euclidean space in discretized form where continuity is enforced by the modal functions.

The data collection process for parameter estimation can be sequential or batch. In sequential estimation, the parameter estimate is updated with each new measurement. In batch processing, all of the data are collected first, then the parameter estimation problem is solved. We will focus on the batch processing of data in this work. Goal-oriented inference can be extended to the sequential processing of data, but we do not undertake that task here. When experiments are completed, no additional data will be obtained before predictions are to be made.

1.3 Current practice in identification of distributed parameters

We are concerned primarily with distributed parameter systems, often governed by a set of partial differential equations (PDEs) modeling the relevant physics in space and time. These systems have infinite-dimensional parameters whose spatial discretization leads to high-dimensional vector forms. Some examples include contaminant identification and carbon dioxide sequestration. The contaminant problem is governed by a convection-diffusion equation with the unknown initial contaminant concentration as the parameter. In carbon dioxide sequestration, governed by two-phase flow in porous media, permeability and porosity of the subsurface are unknown parameters. Parameters must be estimated to understand the behavior of the system, to utilize the system accurately in simulation, and to design or control the system.

Distributed parameter systems are often estimated by reconciling indirect observations with mathematical models of the system. For the problems of interest, it is not feasible, or necessarily useful, to observe the unknown parameter directly. For example, in the contaminant identification problem, the parameter only exists in the past as the initial concentration; in the carbon dioxide sequestration application, some core samples may be taken but they are expensive to obtain, undermine the existing structure of the subsurface, and only provide localized information.

A key challenge of estimating distributed parameter systems is the high dimensionality of the unknown parameter. In the continuous description, the parameter is a scalar, vector, or even tensor field quantity defined everywhere in the domain. By the process of discretization in space, we arrive at a mathematical model suitable for computer implementation but with many (sometimes hundreds of thousands or even tens of millions) of unknown parameters. It is rarely, if ever, the case that every component of the high-dimensional parameter can be inferred from available data. The quantity of experimental data is often orders of magnitudes smaller than the number of unknown parameters. This is the essence of one form of ill-posedness in inverse problems [38].[2]

This ill-posedness is addressed primarily by two classes of formulations of inverse problem: regularized deterministic formulations and statistical formulations. We discuss each in turn below. In what follows we will often refer to *inverse* problems in the deterministic setting and *inference* in the statistical setting to be consistent with established nomenclature. They are different methods for answering essentially the same question: How do we estimate the high-dimensional parameter given low-dimensional data?

1.3.1 Regularized deterministic inverse problems

Deterministic inverse problem formulations are generally PDE-constrained optimization problems with an objective function involving the mismatch between observed

[2] In our applications, there is insufficient data based purely on a dimensional argument. In many applications, however, even with copious data, the parameter may not be uniquely determined. It depends on the amount of independent information contained in the data.

data and model-predicted experimental outputs. A parameter that drives this mismatch below the noise level of the data is a strong candidate to be the true parameter of the system. As mentioned before, many parameters may result in such a small mismatch. In order to improve the mathematical posedness and numerical robustness, the inverse problem is usually regularized by penalizing or restricting the parameters in a manner that does not depend on experimental data.

Regularization techniques can be categorized by two forms: subspace regularization and penalty regularization. Subspace-regularized formulations search for parameter estimates confined to a well-defined subspace. For penalty regularization, a term is added to the objective function that positively contributes to the objective function more for parameters exhibiting some undesirable characteristics (e.g., sharp interfaces, large difference from a nominal value, etc.).

In linear inverse problems, a subspace regularization may be based on the truncated singular value decomposition (TSVD) of the experimental observation operator. Effectively, coefficients of parameter modes informed by experimental data are estimated; the component of the parameter in the orthogonal complement is taken to be zero. The goal-oriented inference approach we present in Chapter 3 will also give rise to a subspace regularization, where the subspace is chosen to properly balance information content in the experimental data with requirements for the predictions.

Regularization is more generally imposed by adding a penalty term to the objective function. Tikhonov regularization is one common approach where a suitably-defined norm of the difference between the parameter and a nominal value is balanced against the mismatch [27]. The nominal value is chosen to bias the parameter estimate toward an expected parameter. The norm is selected to either admit or penalize against parameters with certain characteristics, i.e., smoothness, total integral, or sharp gradients.

One disadvantage of the deterministic formulation is that the parameter estimate in the end is a single field quantity. In particular, we have no measure of the uncertainty in that solution. In contrast, uncertainty quantication follows naturally from the statistical formulation of the inverse problem.

1.3.2 Statistical inference problems

Bayesian statistical inference formulations model the unknown parameter as a random variable (or in the case of distributed quantities as we have here, a random field). The result of the Bayesian inference is a posterior distribution over the parameters given observed experimental data. Critical components of the formulation include specification of a prior distribution over the parameters a priori and the definition of the likelihood function expressing the relative likelihood of experimental data given the parameter.

Given the specifications of prior and likelihood, Bayes's rule gives the posterior distribution (up to a normalizing constant — the evidence) by the product. Typically the forward model will be included in the likelihood function so that the posterior distribution is given implicitly by the parameter. Although we can evaluate the posterior (up to the normalizing constant) for any parameter, it is desired instead to sample from the posterior distribution. This can be achieved in theory by Markov chain Monte Carlo (MCMC) methods.

MCMC techniques aim to implicitly construct a Markov chain over the parameter space whose invariant distribution is the posterior of the inference problem. It is achieved by defining an acceptance probability based on evaluations of the posterior and the relative likelihood of proposed samples and then accepting or rejecting in accordance with that probability. An important benefit of MCMC is that the posterior need only be known up to a normalizing constant, since it cancels out in the acceptance ratio. Therefore, computing the evidence is not necessary – a show-stopping computational burden in many cases.

Since the introduction of the Metropolis-Hastings random walk MCMC in the 1950s [59], many new MCMC techniques have been developed to tackle a wide variety of challenges presented by applications of statistical inference in many settings [57, 26, 22, 34]. Despite the advances, inference in high-dimensional[3] parameter spaces remains unconquered territory. Most methods fail to achieve a suitable automatic

[3] Methods have not been demonstrated consistently for spaces with even hundreds of dimensions, whereas our parameter spaces can exceed hundreds of thousands.

compromise between exhaustive exploration enabled by longer steps from sample to sample and a significant acceptance ratio improved by taking shorter steps from sample to sample. For a detailed review of the state of the art in MCMC techniques, please refer to [51, 18, 61].

In the end, even well-tuned proposal distributions will result in MCMC implementations requiring hundreds of thousands of posterior evaluations, each of which requires a solution to a PDE in applications of interest.

1.4 Challenges of PDE-constrained distributed parameter inverse problems

Applications of interest are typically constrained by PDEs, an additional challenge on top of those mentioned above. Every time a set of outputs must be computed based on a given parameter, the PDE must be solved. For many applications, such a solution could require days of computing time even on the most advanced supercomputers in the world. Where pseudo real-time inversion is required, problems quickly become intractable.

In practice this difficulty is frequently addressed by building a surrogate model. A surrogate model is a computationally inexpensive counterpart to the complete PDE solution but that seeks to maintain the integrity of the input-output map of interest. There are many approaches for deriving a surrogate model coming from many different research communities. Approaches can be divided into intrusive and non-intrusive methods. Intrusive methods modify the governing equations directly while non-intrusive methods only require sample input-output pairs to build the surrogate model.

Intrusive methods largely consist of schemes of model reduction using projections of the governing equations onto lower-dimensional manifolds like moment-matching [24, 30, 36], proper orthogonal decomposition [6, 12, 71, 49], and reduced basis methods [66, 78, 77, 16]. While in the linear setting some algorithms yield reduced models without requiring a solution of the PDE, for nonlinear problems, some (usually guided) sampling of the parameter space and corresponding solutions is required. Intrusive methods typically provide a surrogate model for the maps from parameter/input to state, while non-intrusive methods often focus solely on the parameter/input to output map without regard for the governing equations or state at all.

Non-intrusive methods treat the model or physical system as a black box, establishing a scheme for producing a predicted output from parameter input, a so-called response surface. A set of experimental design points, parameters at which to interrogate the model or physical system, is determined to produce data from which to establish the response surface. Methods include regressions of linear and nonlinear type as well as a suite of interpolatory approaches including Kriging [63] and Gaussian processes [68] and their variants. Such methods are physics-agnostic and are popular in data-driven approaches typical in machine learning and statistics.

Many of the intrusive methods and all of the non-intrusive methods require an experimental design procedure to determine the parameters at which full physics models or real systems will be interrogated. There are many well-established approaches for conducting this experimental design process in parameter spaces with a handful of dimensions; however, in very high-dimensional parameter spaces, as we have with the discretization of distributed parameters here, most of those methods become intractable. They simply do not scale acceptably with parameter dimension.

Some successful approaches have involved targeting outputs of interest. For example, the construction of reduced basis models using greedy sampling of parameter/input space has been used in the context of inverse problems [9, 53]. These approaches typically only focus on the outputs of interest, but that treats only half of the problem — the experimental process. The prediction process is disregarded under the pretense that the parameter can be properly inferred. To dismiss this critical part of the inference-for-prediction problem is to leave out the driving facet: the final prediction that called for the parameter estimation in the first place.

This thesis considers the entirety of the process of establishing prediction estimates based ultimately on the observed data from experiments. We do not focus individually on developing surrogate models for the experimental and prediction processes separately; instead, we develop a surrogate model for the inference process such that the final predictions are accurate.[4]

1.5 Recent focus on goal-oriented methods

Recently there has been a significant thrust in the research community to emphasize the goal-oriented computation of quantities of interest. In finite elements, mesh adaptation and error estimation have both been driven by the need to accurately estimate low-dimensional quantities.

Consider the computational fluid dynamics approach of predicting the lift of an airfoil at certain angle of attack in uniform freestream flow in 2–D. One approach is to model the complete physics and represent them on a uniformly resolved finite element mesh in an attempt to accurately calculate the state of the flow everywhere in the computational domain. Once the pressures are obtained around the airfoil, they can be integrated to determine the lift. Researchers have realized that this approach for estimating the lift on an airfoil is wasteful and inefficient. If the lift is determined by integrating the pressures around the airfoil, why is it that we must resolve the state of the flow everywhere in the domain?

Adjoint techniques, which have also become essential tools in PDE-constrained optimization, are employed in error estimation of output quantities of interest as well. Based on linearization, error estimation via the adjoint permits the approximation of errors without computing a *truth* solution at all. For mesh adaptation, the adjoint solution can be localized to give a measure of the sensitivity of the error in output prediction as a means to guide selective refinement and coarsening procedures [67]. The adjoint gives a computationally efficient approach to calculate output sensitivities for high-dimensional parameters. In contrast, direct sensitivity calculations based on perturbations of each parameter (like finite differencing) are more expensive in this

[4] The accuracy of predictions is based on a given well-defined inference procedure to infer the parameter using the complete physics model.

context. Since the computation of the lift of the airfoil is the goal, it makes sense to modify the mesh to obtain accuracy in that calculation. Resolving the pressures far downstream of the airfoil is not necessary.

In the goal-oriented inference analog, we are tasked with inferring a parameter that is also a distributed quantity like the pressure living on a computational mesh. While in the mesh adaptivity case we solve for the pressure field to compute the lift on the airfoil as the quantity of interest, in the goal-oriented inference setting, we require the parameter estimate in order to compute our prediction output quantity of interest. It stands to reason then that with careful study of the problem we could avoid resolving the entirety of the parameter and instead focus on the components informed by our experimental data and required to accurately predict our output quantities of interest.

1.6 Research objectives

The primary objective of this research is to provide foundational work in goal-oriented inference. In many engineering applications, the inference of distributed parameters is one step in a process resulting in predictions of low-dimensional quantities of interest. We exploit this fact to understand sources of uncertainty in predictions, identify uninformative experiments, and achieve online efficiency in making predictions by trading off accuracy in parameter estimation. In the linear setting, we establish theoretical guarantees; for nonlinear problems, we have guarantees in the infinite sample limit. Both approaches are demonstrated on problems paramount to curbing anthropogenic effects on the environment. More specifically, the research objectives are:

- to formulate goal-oriented inference so that predictions are the driving factors in the inference process;

- to develop a set of goal-oriented inference procedures companion to well-established parameter identification algorithms;

- to establish offline analysis tools to guide experimental design and expose sources of prediction uncertainty for linear problems;

- to derive theoretical guarantees on the prediction accuracy of goal-oriented inference for linear problems;

- to demonstrate linear goal-oriented inference on a model problem in contaminant identification and prediction;

- to develop a practical algorithm that extends goal-oriented inference to nonlinear problems;

- and to demonstrate nonlinear goal-oriented inference in performing a probabilistic risk assessment in carbon capture and storage (CCS).

1.7 Thesis outline

This thesis is organized as follows. In Chapter 2 we introduce a teleological approach to identification of distributed parameters, looking forward to final objectives to inform the inference process. Chapters 3 and 4 are devoted to problems with outputs that are linear in the unknown parameter. In Chapter 3 we develop the foundational algorithm for the truncated singular value decomposition approach to goal-oriented inference, provide the relevant theory and analysis, then extend to other popular inverse problem formulations. In Chapter 4 we demonstrate the techniques on a model problem in contaminant identification and prediction. Chapters 5 and 6 extend goal-oriented inference to nonlinear problems with a statistical formulation. In Chapter 5 we develop an algorithm for learning the joint density between potentially observed data and predictions using a sampling scheme in combination with a Gaussian mixture model representation. The mechanics for conditioning on data are provided. In Chapter 6 we apply the method to a model problem in carbon capture and storage. We provide a summary and conclusions of the work in Chapter 7.

Chapter 2

A teleological approach to identification of distributed parameters

The process of utilizing experimental data to estimate unknown parameters is central to many important problems in science and engineering. Inference problems arise in medical imaging [5], geophysics [17], meteorology and oceanography [47], heat transfer [2], electromagnetic scattering [41], and electrical impedance tomography [3], among many other disciplines.

2.1 Parameter identification in the context of predictions

Many inverse problems are ill-posed; the data do not determine a unique solution. Inference approaches, therefore, rely on the incorporation of prior information. In deterministic formulations [27], this prior information is often manifested as a form of regularization. In Bayesian statistical formulations [75], the prior information is used to formulate a prior distribution reflecting the belief in probable parameter values. As a result, the distinction becomes blurred between inferred parameter modes informed by data and modes influenced largely or wholly by prior information. Without careful design of prior information, data-independent information can overshadow the information contained in the limited data collected from experiments. Although ill-posedness will always be an issue to some extent in limited data settings, in this thesis we show that it is possible to partially circumvent the deleterious effects of the use of regularizers or prior information by incorporating end goals.

While in some cases estimation of unknown parameters is the end goal, there are many engineering processes where parameter estimation is one step in a multi-step process ending with design. In such scenarios, engineers often define output quantities of interest to be optimized by the design. In consideration of this fact, we propose a goal-oriented approach to inference that accounts for the output quantities of interest. Generally, in an abstract sense, our experimental data are informative about certain modes in the parameter space and another set of modes in parameter space are required to accurately estimate the output quantities of interest. Our philosophy is to understand the relationship between these two sets of modes and to modify our approach to inference based on that information. The goal-oriented inference method involves identifying parameter modes that are both informed by experiment and also required for estimating output quantities of interest. In what follows, we refer to the output quantities of interest as predictions although predictions need not be outputs of a system but instead could be, for example, the evaluation of the objective function in a design optimization problem. We call it the *inference-for-prediction* (IFP) method.

The two decompositions of parameter space based on the experimental process and the prediction process are shown notionally in Figure 2-1. We consider abstractly the decomposition into experimentally-observable and experimentally-unobservable modes on the left side of Figure 2-1. Experimentally-observable modes are informed by the experimental data while experimentally-unobservable modes are not. The analogous decomposition for prediction is shown on the right side of Figure 2-1. It is informative to explore the combinations of observable modes from the two processes. We explicitly identify here three types of parameter modes. Modes informed by experiment and required for prediction are targeted by the IFP method. Modes that

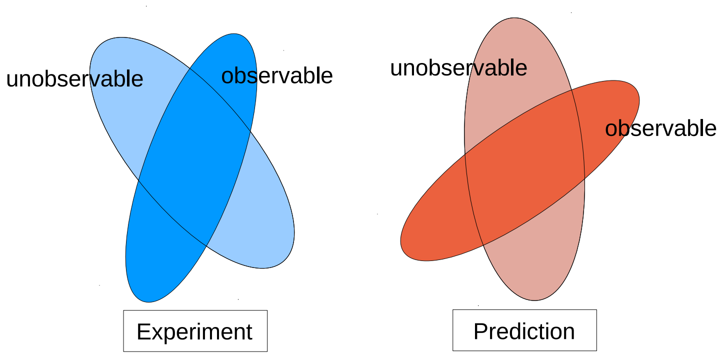


Figure 2-1: The two separate decompositions of the parameter space based on the experiment (left) and prediction (right). Note that the unobservable and observable spaces are orthogonal complements of each other in $\mathbb{R}^n$; the intersection contains only the zero vector.

are informed by experiment but not required for prediction represent inefficiencies in the experimental data acquisition. Finally, modes that are required for prediction but are uninformed by experiment lead to uncertainty in the prediction and may guide future experimentation.

There are many advantages to this way of thinking including computational efficiency in the inference step, enabling deployment on lightweight, portable devices (e.g., smartphones and laptops) in the field; understanding of the effects of regularization and prior information on predictions; identification of inefficiencies in experimental data acquisition to focus efforts on data informative about modes required for predictions; and understanding of vulnerabilities in predictions. In the nonlinear setting, this approach not only makes real-time prediction possible, but makes tractable a class of problems in statistical inference for prediction that are currently infeasible.

2.2 Decomposition of offline analysis and online inference-for-prediction

Consider the partition of the experiment-infer-predict process into offline analysis and online prediction. The offline and online segments of the process are divided by the data acquisition process. In the offline phase, we exploit the mathematical structure within the experiment and prediction processes (i.e., their dependence on parameter) to perform analysis and automatically construct a reduced model for the inference-for-prediction process. In the online phase, we perform experiments and acquire data, then utilize that data to make predictions based on the model constructed in the offline phase.

2.2.1 Offline analysis

Once the experiments and prediction requirements are defined, and before experiments are conducted and data observed, the mathematical structure of the experiment-infer-predict process is exploited to generate a reduced model for inference-for-prediction.

In the linear setting, the analysis tools are naturally linear algebraic. Through an eigendecomposition, a joint measure of experiment and prediction observability determines a basis for the lowest-dimensional subspace of parameters that gives exact predictions in the online stage. The decomposition exposes the relationships between parameter modes informed by the experimental data and modes required to make accurate predictions. This analysis yields identification of experimental inefficiencies based on experiments that provide information about parameter modes that are irrelevant for prediction. The process also establishes the primary sources of uncertainty based on the modes uninformed by experimental data but required for prediction. This information could be used to iterate on the experimental design before proceeding to data acquisition.

In the nonlinear setting, the linear algebraic analysis tools are no longer applicable globally; however, we can still identify and exploit the mathematical structure in the experiment and prediction processes. This can be achieved by learning the joint density between data and predictions. The density can be represented by a Gaussian mixture model (GMM) whose parameters are fit by sampling from a prior distribution on the parameter. Typically the density estimation problem is infeasible in the inverse problem since it would need to be constructed over parameters and data. Here, however, where we have the dimensional compression via the prediction process, our prediction and data typically occupy a space of modest dimension: density estimation is feasible. In this case we can also evaluate the experimental design by investigating, for example, a measure of the expected information gain in the posterior predictive density.

2.2.2 Online inference-for-prediction

After the analysis is performed and the inference-for-prediction model constructed, experiments are conducted and data are acquired. The task online is then to utilize the data to make accurate predictions.

In the linear setting, inference-for-prediction will take place in a low-dimensional subspace of the parameter space. The solution to the inference problem will be the coefficients in a basis expansion in this space. The basis can be computed offline since it is data independent. Since the prediction problem is also linear, we also are able to compute the prediction corresponding to each basis function, subsequently forming a basis for the prediction. Therefore, when data are collected, obtaining the prediction estimate requires only to determine the coefficients of the basis expansion and compute the weighted combination of previously-obtained predictions. The online computations can be performed in real-time.

For the statistical approach in nonlinear problems, the result of the online inference-for-prediction is the posterior predictive, a probability density over predictions representing the state of belief given observed data. After building the GMM representing the joint density between data and predictions in the offline phase, the online phase requires only the conditioning of the GMM at the observed data. There are two parts to the conditioning process: (i) each mixture component must be conditioned individually and (ii) the weights of each component must be recalculated based on the relative marginal likelihood of the observed data. Each of these steps is analytic due to the convenient form of the GMM.

2.3 Procedural timeline

The goal of our procedure is to obtain accurate real-time predictions when they depend on unknown parameters implicitly informed by observed data. In applications of interest, it will be intractable to perform parameter identification and subsequent prediction online in the linear setting, and may be completely intractable to perform parameter identification at all in the nonlinear setting. The time scales on which we need predictions are orders of magnitude less than the time it would take to resolve the parameter accurately, even using state-of-the-art supercomputers.

As an example, consider the contaminant prediction problem. A contaminant is released in an urban environment. It advects and diffuses. We make measurements of the contaminant concentration from sensors sparsely distributed throughout the domain. Using the time series data from the sensors, what level of contaminant will there be near a critical building in a later time interval? We could not expect that a state-of-the-art inverse problem solver would have the initial condition of the release identified within thirty minutes. By exploiting the goal of predicting the contaminant level in a given area at a specified time, we can make predictions in real-time if we pay an up front computational cost to perform the analysis described above.

It is important for this process that there be sufficient time in the offline phase to perform the necessary analysis. The analysis can proceed as soon as the experiments and prediction are defined so that the maps from parameter to data and parameter to predictions are well defined. It must be completed before the data are observed. In this work we do not treat adaptive or sequential experiment-infer-predict processes; however, many of the ideas could be extended to this context.

2.4 Abstract problem statement

Goal-oriented inference is the task of estimating a parameter from data for the purpose of making accurate predictions. We will now outline the abstract problem statement which we will treat in the linear and nonlinear cases in later chapters.

Let $\mathcal{P}$ be the unknown parameter. Experiments are defined to produce outputs $\mathcal{Y}_d = \mathcal{M}_e(\mathcal{P}) + \mathcal{E}$ based on the experiment model $\mathcal{M}_e$ and noise $\mathcal{E}$. Predictions $\mathcal{Y}_p = \mathcal{M}_p(\mathcal{P})$ are based on the prediction model $\mathcal{M}_p$.

Let $\mathcal{I} : \mathcal{Y}_d \to \mathcal{P}$ be a well-defined inference procedure so that $\hat{\mathcal{P}} = \mathcal{I}(\mathcal{Y}_d)$ is the resulting parameter estimate. The objective of parameter estimation is to minimize $\|\mathcal{P} - \hat{\mathcal{P}}\|$ in a suitable norm. The estimated parameter can then be passed as input to the prediction model to obtain outputs $\hat{\mathcal{Y}}_p = \mathcal{M}_p(\hat{\mathcal{P}})$ that are assumed to be noiseless.[1]

On the other hand, the objective of goal-oriented inference is to obtain accurate predictions, i.e., to minimize $\|\tilde{\mathcal{Y}}_p - \hat{\mathcal{Y}}_p\|$, where $\tilde{\mathcal{Y}}_p$ is the prediction obtained by goal-oriented inference and $\hat{\mathcal{Y}}_p$ is the prediction obtained by first estimating the parameter $\hat{\mathcal{P}}$ using the inference algorithm $\mathcal{I}$ defined above.

We will see in Chapter 3, in the case when $\mathcal{M}_e$ and $\mathcal{M}_p$ are linear functions of the parameter, that we can drive the prediction error $\|\tilde{\mathcal{Y}}_p - \hat{\mathcal{Y}}_p\|$ to zero (i.e., replicate the predictions obtained from standard inference techniques) for some popular inverse problem formulations. For the nonlinear statistical setting in Chapter 5, we will demonstrate that this error can be driven below a specified tolerance with sufficient a priori sampling of the parameter space.

[1] It is possible to extend this work to situations where there is a statistical model for the predictions. We do not treat that case here; instead, we assume the prediction model can be trusted.

Chapter 3

Linear goal-oriented inference

The fundamental principles of goal-oriented inference have now been established. In this chapter we will begin the technical development of solution methodology for goal-oriented inference problems in the context of linear experimental and linear prediction processes. Such processes can arise from inherently linear problems or from the suitable linearization of nonlinear systems. While this is a restrictive assumption, the developments of this chapter provide the critical substantive foundation for goal-oriented inference. Without proper theory in the linear case, we certainly cannot expect the ideas to apply to more challenging nonlinear scenarios.

This chapter is organized as follows. In section 3.1 we present background in control-theoretic concepts and balanced truncation model reduction, both fundamental ideas to be exploited in our solution to the linear goal-oriented inference problem. The inference-for-prediction (IFP) method is established in section 3.2 as the solution to the linear goal-oriented inference problem for the truncated singular value decomposition (TSVD) approach to parameter identification. Theoretical guarantees are provided in section 3.3 where we prove important properties of the IFP method including prediction exactness and the dimensional optimality of the IFP subspace. Finally, in section 3.4 we provide the extensions of the IFP method to Tikhonov-regularized and Gaussian statistical inverse problem formulations.

3.1 Background

In this section we provide background material in control-theoretic concepts and balanced truncation model reduction that will be the basis for the development of our goal-oriented inference approach. When developing mathematical models we often describe systems by state equations where the states are defined to be physical quantities (e.g., position, velocity, momentum, etc.). When those state equations are coupled with an output equation, the physically meaningful state vector is not always the minimum dimension vector that defines the system for the purposes of predicting the output. Model reduction is the term used to describe the act of reducing such a system to a lower-dimensional description that maintains integrity in output predictions, typically over a desired range of inputs to the system.

Balanced truncation is one systematic method for performing model reduction on a linear time-invariant system. The determination of a new state vector depends on two control-theoretic concepts. Controllability of a state refers to the input energy required to drive a system to zero from that state. Observability of a state refers to the output energy associated to that state. Independently, the most controllable (observable, respectively) modes are the eigenvectors of the controllability (observability, respectively) gramian corresponding to larger eigenvalues. The goal of balanced truncation model reduction is to obtain a reduced state vector composed of modes that exceed a certain threshold on a joint measure of controllability and observability known as Hankel singular values.

We will find later that a useful analogy can be drawn between the state of a large-scale model and the parameter in our goal-oriented inference problem. This is illustrated in Figure 3-1. Namely, the state equation of a system determines the time evolution of the state due to input to the system. In the goal-oriented inference context, the parameter estimate is determined through the solution of an inverse problem based on experimental data. Furthermore, the output equation determines the outputs given a state in the dynamical system context. In the goal-oriented inference context, the prediction quantity of interest is determined by the parameter

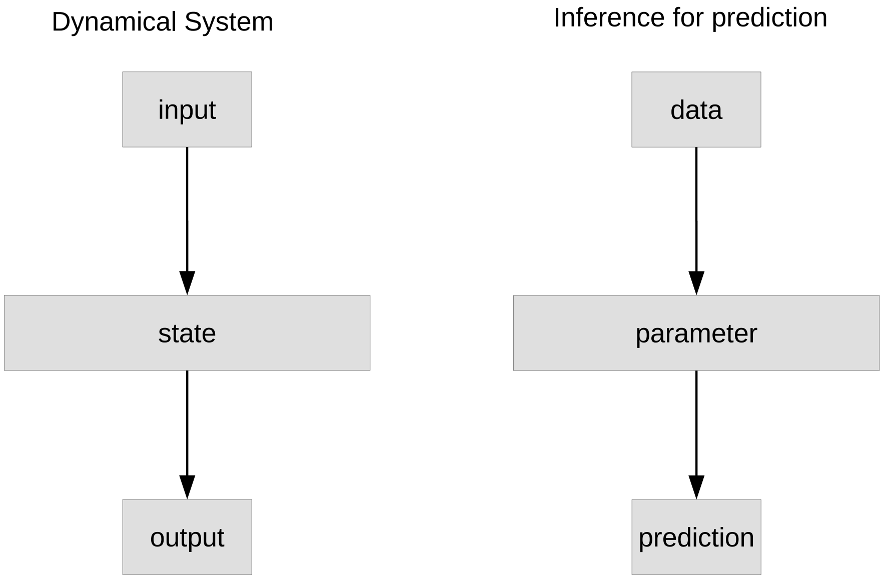


Figure 3-1: Comparison of dynamical systems (left) and inference for prediction (right). State and parameter are high-dimensional; however, inputs/data and outputs/prediction are low-dimensional.

estimate. In both (dynamical system, goal-oriented inference) situations, we have low-dimensional input (input, data), high-dimensional model description (state, parameter), and low-dimensional output quantities of interest (outputs, predictions). We will show later that the balanced truncation methodology used to determine a reduced system has an analog in goal-oriented inference.

For the following exposition, consider the time-invariant discrete-time linear system

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k, \quad k = 0, 1, \ldots, \tag{3.1}$$

$$\mathbf{y}_k = \mathbf{C}\mathbf{x}_k, \quad k = 0, 1, \ldots, \tag{3.2}$$

where $\mathbf{x}_k \in \mathbb{R}^n$ is the state at time step k, $\mathbf{A} \in \mathbb{R}^{n \times n}$, $\mathbf{B} \in \mathbb{R}^{n \times n_i}$, $\mathbf{u}_k \in \mathbb{R}^{n_i}$ is the input at time step k, $\mathbf{C} \in \mathbb{R}^{n_o \times n}$, and $\mathbf{y}_k \in \mathbb{R}^{n_o}$ is the output at time step k. The system has given initial condition $\mathbf{x}_0$. We assume that (3.1) is stable; i.e., the spectral radius $\rho(\mathbf{A}) < 1$.

3.1.1 Controllability and observability

Controllability and observability are two important properties of the system (3.1)–(3.2) [46]. The information contained within them is exploited in balanced truncation model reduction, as we describe in section 3.1.2.

A measure of the *controllability* $L_c(\mathbf{x})$ of a state $\mathbf{x}$ is the minimum input energy required to drive the system to zero when initialized at $\mathbf{x}_0 = \mathbf{x}$; i.e.,

$$L_c(\mathbf{x}) = \min_{\mathbf{u}_k,\, \forall k} \sum_{k=0}^{\infty} \|\mathbf{u}_k\|^2, \quad \text{s.t. } \mathbf{x}_0 = \mathbf{x},\ \lim_{k \to \infty} \mathbf{x}_k = \mathbf{0}.$$

Let $\mathbf{P} = \sum_{k=0}^{\infty} \mathbf{A}^k \mathbf{B} \mathbf{B}^\top (\mathbf{A}^\top)^k \in \mathbb{R}^{n \times n}$ be the controllability gramian [32]. The system (3.1)–(3.2) is *controllable* if $\mathbf{P}$ is full rank. Then we may write $L_c(\mathbf{x}) = \mathbf{x}^\top \mathbf{P}^{-1} \mathbf{x}$. If more energy is required to drive the system to zero, the state is less controllable.

A measure of the *observability* $L_o(\mathbf{x})$ of a state $\mathbf{x}$ is the total output energy generated by the unforced ($\mathbf{u}_k = \mathbf{0}$, $\forall k$) system initialized at $\mathbf{x}_0 = \mathbf{x}$; i.e.,

$$L_o(\mathbf{x}) = \sum_{k=0}^{\infty} \|\mathbf{y}_k\|^2 = \sum_{k=0}^{\infty} \|\mathbf{C}\mathbf{A}^k \mathbf{x}\|^2.$$

Let $\mathbf{Q} = \sum_{k=0}^{\infty} (\mathbf{A}^\top)^k \mathbf{C}^\top \mathbf{C} \mathbf{A}^k \in \mathbb{R}^{n \times n}$ be the observability gramian [32]. Then the observability associated to a state $\mathbf{x}$ is $L_o(\mathbf{x}) = \mathbf{x}^\top \mathbf{Q} \mathbf{x}$. The system (3.1)–(3.2) is *observable* if $\mathbf{Q}$ is full rank. If the evolving unforced system produces larger output, the initial state is more observable.

The controllability and observability gramians are usually computed as solutions to the Stein equations

$$-\mathbf{P} + \mathbf{A}\mathbf{P}\mathbf{A}^\top = -\mathbf{B}\mathbf{B}^\top, \qquad -\mathbf{Q} + \mathbf{A}^\top \mathbf{Q} \mathbf{A} = -\mathbf{C}^\top \mathbf{C},$$

respectively.

3.1.2 Balanced truncation model reduction

A projection-based reduced model of the system (3.1)–(3.2) is given by

$$\begin{aligned}\hat{\mathbf{x}}_{k+1} &= \hat{\mathbf{A}}\hat{\mathbf{x}}_k + \hat{\mathbf{B}}\mathbf{u}_k, \quad k = 0, 1, \ldots, \\ \hat{\mathbf{y}}_k &= \hat{\mathbf{C}}\hat{\mathbf{x}}_k, \qquad\qquad k = 0, 1, \ldots,\end{aligned}$$

where $\hat{\mathbf{A}} = \mathbf{U}^\top\mathbf{A}\mathbf{V} \in \mathbb{R}^{m\times m}$, $\hat{\mathbf{B}} = \mathbf{U}^\top\mathbf{B} \in \mathbb{R}^{m\times n_i}$, $\hat{\mathbf{C}} = \mathbf{C}\mathbf{V} \in \mathbb{R}^{n_o\times m}$, $\hat{\mathbf{x}}_k \in \mathbb{R}^m$ is the reduced state at time step k, and $\hat{\mathbf{y}}_k \in \mathbb{R}^{n_o}$ is the output of the reduced model at time step k. The left basis $\mathbf{U} \in \mathbb{R}^{n\times m}$ and right basis $\mathbf{V} \in \mathbb{R}^{n\times m}$ span subspaces of dimension $m \ll n$ and satisfy $\mathbf{U}^\top\mathbf{V} = \mathbf{I}$.

Balanced truncation model reduction is one method for selecting the left and right bases [60]. Conceptually, balanced truncation can be understood in two distinct steps. The first step is a similarity transformation to describe the state space of the system (3.1)–(3.2) in a way that balances each coordinate direction's combined measure of controllability and observability. In particular, the controllability and observability gramians of the transformed system are diagonal and equal. The second step is truncation, whereby only some of the states in the transformed model are retained. For example, any coordinate directions having zero combined measure of controllability and observability can be truncated without affecting the system's input-output behavior.

We first identify the similarity transformation. The balanced truncation left and right bases can be obtained using general matrix factors of the controllability and observability gramians [10, 50]. For purposes of exposition, we will assume the factors are square. Let $\mathbf{P} = \mathbf{S}\mathbf{S}^\top$ and $\mathbf{Q} = \mathbf{R}\mathbf{R}^\top$. Consider the similarity transformation defined by $\mathbf{T} = \mathbf{\Sigma}^{-1/2}\mathbf{M}^\top\mathbf{R}^\top$ and $\mathbf{T}^{-1} = \mathbf{S}\mathbf{N}\mathbf{\Sigma}^{-1/2}$ where $\mathbf{M}\mathbf{\Sigma}\mathbf{N}^\top$ is the singular value decomposition (SVD) of $\mathbf{R}^\top\mathbf{S}$. The transformed system

$$\begin{aligned}\tilde{\mathbf{x}}_{k+1} &= \mathbf{T}\mathbf{A}\mathbf{T}^{-1}\tilde{\mathbf{x}}_k + \mathbf{T}\mathbf{B}\mathbf{u}_k, \quad k = 0, 1, \ldots, \\ \mathbf{y}_k &= \mathbf{C}\mathbf{T}^{-1}\tilde{\mathbf{x}}_k, \qquad\qquad\quad k = 0, 1, \ldots,\end{aligned}$$

has diagonal and equal controllability and observability gramians,

$$\mathbf{T}\mathbf{P}\mathbf{T}^\top = \mathbf{T}^{-T}\mathbf{Q}\mathbf{T}^{-1} = \boldsymbol{\Sigma}.$$

The coefficients $\sigma_1, \sigma_2, \ldots, \sigma_n$ on the diagonal of $\boldsymbol{\Sigma}$ are known as the Hankel singular values, which represent a joint measure of the controllability and observability of the modes in the transformed system. The second step is the truncation of the transformed state $\tilde{\mathbf{x}} \in \mathbb{R}^n$ to $\hat{\mathbf{x}} \in \mathbb{R}^m$. The truncation eliminates the least controllable and observable modes of the system based on the Hankel singular values.

Algorithm 1 Balanced truncation model reduction left and right bases

```
1: Compute the first $m$ normalized eigenvectors $\boldsymbol{\psi}_i$ of $\mathbf{S}^\top\mathbf{Q}\mathbf{S}$ with corresponding eigenvalues $\sigma_i^2$, i.e.
       $\mathbf{S}^\top\mathbf{Q}\mathbf{S}\boldsymbol{\psi}_i = \sigma_i^2\boldsymbol{\psi}_i, \quad \|\boldsymbol{\psi}\|^2 = 1, \ \boldsymbol{\psi}_i^\top\boldsymbol{\psi}_j = \delta_{ij}, \ i = 1, 2, \ldots, m.$
2: Compute the first $m$ left eigenvectors $\boldsymbol{\phi}_i$ of $\mathbf{R}^\top\mathbf{P}\mathbf{R}$ also having eigenvalues $\sigma_i^2$, i.e.
       $\boldsymbol{\phi}_i = \sigma_i^{-1}\boldsymbol{\psi}_i^\top\mathbf{S}^\top\mathbf{R}, \quad i = 1, 2, \ldots, m.$
3: Then define the left and right bases
       $\mathbf{U} = \mathbf{R}\left[\begin{array}{ccc}\sigma_1^{-1/2}\boldsymbol{\phi}_1^\top & \cdots & \sigma_m^{-1/2}\boldsymbol{\phi}_m^\top\end{array}\right], \qquad \mathbf{V} = \mathbf{S}\left[\begin{array}{ccc}\sigma_1^{-1/2}\boldsymbol{\psi}_1 & \cdots & \sigma_m^{-1/2}\boldsymbol{\psi}_m\end{array}\right].$
```

In practice the balanced truncation reduced model can be obtained by directly identifying left and right bases $\mathbf{U}$ and $\mathbf{V}$ by Algorithm 1. Although balanced truncation is not optimal, there exist bounds on the $\mathcal{H}_\infty$-norm of the error system related to the truncated Hankel singular values [60]. There exist algorithms to obtain a balanced reduced model via approximate computation of the gramians for large-scale systems [50, 37].

In balanced truncation model reduction, analysis of the controllability and observability gramians leads to a reduced system of equations accounting for both the state and output equations. For goal-oriented inference, we have very similar goals. We wish to analyze the behaviors of the experimental process and prediction process to generate a reduced description of the inversion and prediction sequence. In balanced truncation, we maintain integrity of the input-output relation. For goal-oriented inference, we will maintain integrity of the data-prediction relation.

3.2 Inference-for-prediction (IFP) method

Let $\boldsymbol{\mu} \in \mathbb{R}^q$ be an unknown parameter defining a system of interest. We assume that q is large, i.e., there are many more parameters to infer than experiments we can afford to perform or predictions we wish to make. Let $\mathbf{O}_e \in \mathbb{R}^{r \times q}$ be the linear observation operator representing the (usually indirect) measurement process taking the parameter space to the space of experimental observables of dimension $r < q$. We write experimental outputs $\mathbf{y}_e = \mathbf{O}_e \boldsymbol{\mu}$. In many instances it will be appropriate to model sensor error in which case we obtain $\mathbf{y}_d = e(\mathbf{y}_e, \epsilon)$ for some error model e and a measure of error ϵ. Our formulation will utilize the experimental output matrix $\mathbf{O}_e$, but our algorithms will be data-independent and therefore admit any form of the error model e. For many applications of interest, $\mathbf{O}_e$ will be the composition of a PDE operator and an observation operator. Likewise, the prediction operator $\mathbf{O}_p \in \mathbb{R}^{s \times q}$ is analogous to $\mathbf{O}_e$, but instead measures prediction output quantities of interest in the space of dimension $s < r$. We write prediction $\mathbf{y}_p = \mathbf{O}_p \boldsymbol{\mu}$.[1]

In section 3.2.1 we define experiment and prediction observability and the associated gramians. In section 3.2.2 we state the assumptions, give important definitions, and establish the IFP property. We conclude with an algorithm for obtaining a basis for efficient inversion. Finally, in section 3.2.3 we discuss the numerical implementation of the algorithm and analyze the computational complexity.

[1] Note that the physics underlying the PDE operators (if present) in $\mathbf{O}_e$ and $\mathbf{O}_p$ need not be the same. Typically experimental conditions will differ from operational conditions, and our method admits that naturally.

3.2.1 Experiment and prediction observability

Experiment and prediction observability extend the concept of observability of linear systems described in section 3.1.1 to the goal-oriented inference setting.

A measure of the *experiment observability* of a parameter $\boldsymbol{\mu}$ is given by the experimental output energy associated to it. We define $L_e(\boldsymbol{\mu}) = \|\mathbf{y}_e\|^2 = \|\mathbf{O}_e\boldsymbol{\mu}\|^2$. Consequently, the experiment observability gramian $\mathbf{H}_e = \mathbf{O}_e^\top\mathbf{O}_e$ can be defined since $L_e(\boldsymbol{\mu}) = \boldsymbol{\mu}^\top\mathbf{H}_e\boldsymbol{\mu}$. Since the experiment observability gramian is symmetric and positive semi-definite it admits the decomposition $\mathbf{H}_e = \mathbf{V}_e\mathbf{L}_e\mathbf{V}_e^\top$ where $\mathbf{V}_e \in \mathbb{R}^{q\times r}$ is orthogonal and $\mathbf{L}_e \in \mathbb{R}^{r\times r}$ is diagonal with positive entries. The columns of $\mathbf{V}_e$ are eigenvectors of $\mathbf{H}_e$ with corresponding eigenvalues on the diagonal of $\mathbf{L}_e$. When we solve the inverse problem, the pseudoinverse $\mathbf{H}_e^\dagger = \mathbf{V}_e\mathbf{L}_e^{-1}\mathbf{V}_e^\top$ and its matrix factor $\mathbf{G}_e = \mathbf{V}_e\mathbf{L}_e^{-1/2}$ will play an important role.

Let $\mathbf{V}_{e^\perp}$ be an orthogonal basis whose range is the orthogonal complement to the range of $\mathbf{V}_e$. Then any parameter $\boldsymbol{\mu}$ can be decomposed as $\boldsymbol{\mu} = \mathbf{V}_e\mathbf{V}_e^\top\boldsymbol{\mu} + \mathbf{V}_{e^\perp}\mathbf{V}_{e^\perp}^\top\boldsymbol{\mu}$. The first term influences the data observed for parameter $\boldsymbol{\mu}$ while the second term produces exactly zero experimental output. When we utilize data to infer the parameter, the unobservable component (second term) of $\boldsymbol{\mu}$ is determined by incorporating data-independent information through regularization and prior distribution in the deterministic and statistical approaches, respectively.

A measure of the *prediction observability* of a parameter $\boldsymbol{\mu}$ is given by the prediction output energy associated to it. Define $L_p(\boldsymbol{\mu}) = \|\mathbf{y}_p\|^2 = \|\mathbf{O}_p\boldsymbol{\mu}\|^2$. The prediction observability gramian $\mathbf{H}_p = \mathbf{O}_p^\top\mathbf{O}_p$ then follows since $L_p(\boldsymbol{\mu}) = \boldsymbol{\mu}^\top\mathbf{H}_p\boldsymbol{\mu}$. It is also symmetric and positive semi-definite and therefore has a decomposition $\mathbf{H}_p = \mathbf{V}_p\mathbf{L}_p\mathbf{V}_p^\top$ analogous to $\mathbf{H}_e$ above. Similarly, any parameter $\boldsymbol{\mu}$ can be decomposed as $\boldsymbol{\mu} = \mathbf{V}_p\mathbf{V}_p^\top\boldsymbol{\mu} + \mathbf{V}_{p^\perp}\mathbf{V}_{p^\perp}^\top\boldsymbol{\mu}$. The first component will pass through to predictions $\mathbf{y}_p$ and is therefore necessary to accurately estimate; on the other hand, the second component is in the kernel of $\mathbf{O}_p$ and will therefore not contribute to $\mathbf{y}_p$. Thus, the second component need not be accurately estimated, or even estimated at all, to achieve accurate estimates of $\mathbf{y}_p$.

An algorithm to perform goal-oriented inference should exploit these decompositions and the relationships between them.

3.2.2 IFP algorithm

The IFP method will lead to an experiment and prediction balanced basis for inference spanning the low-dimensional subspace of the parameter space that will result in replication of the predictions obtained by a traditional approach to the linear inverse problem. In this section we will treat the truncated singular value decomposition approach to the linear inverse problem, and we will extend the method to the Tikhonov-regularized inverse problem and Gaussian statistical inverse problem in section 3.4.

We begin with a truncated singular value decomposition (TSVD) approach to the linear inverse problem. The inverse problem uses data $\mathbf{y}_d$ and knowledge of $\mathbf{O}_e$ to estimate the unknown parameter $\boldsymbol{\mu}$. In many applications however the inverse problem is ill-posed due to the vast null space of $\mathbf{O}_e$. This difficulty is usually overcome by regularization. In this section, we consider a form of subspace regularization by seeking a solution only in the row space of $\mathbf{O}_e$. In section 3.4.1 we will consider regularization in the form of a penalty in the objective function.

Let $\mathbf{PSV}^\top = \mathbf{O}_e$ be the SVD with $\mathbf{P} \in \mathbb{R}^{r\times r}$, $\mathbf{S} \in \mathbb{R}^{r\times q}$, and $\mathbf{V} \in \mathbb{R}^{q\times q}$. Let $\mathbf{V}_e \in \mathbb{R}^{q\times r}$ and $\mathbf{V}_e^\perp \in \mathbb{R}^{q\times(q-r)}$ span the row space and null space of $\mathbf{O}_e$, respectively, such that $\mathbf{V} = [\mathbf{V}_e, \mathbf{V}_e^\perp]$. Let $\mathcal{V}_e \subset \mathbb{R}^q$ be the r-dimensional subspace spanned by the columns of $\mathbf{V}_e$. The TSVD approach searches for $\boldsymbol{\mu} \in \mathcal{V}_e$ that reproduces the observed data with minimal error in ℓ_2-norm. That is,

$$\boldsymbol{\mu}^{\mathrm{TSVD}} = \arg\min_{\boldsymbol{\mu}\in\mathcal{V}_e} \frac{1}{2}\|\mathbf{y}_d - \mathbf{O}_e\boldsymbol{\mu}\|_2^2. \tag{3.3}$$

The first-order optimality condition for (3.3) is obtained by imposing the subspace constraint and setting the first-derivative of the objective function to zero; i.e.,

$$\mathbf{V}_e^\top\mathbf{O}_e^\top\mathbf{O}_e\mathbf{V}_e\mathbf{a} = \mathbf{V}_e^\top\mathbf{O}_e^\top\mathbf{y}_d, \tag{3.4}$$

where $\mathbf{a} \in \mathbb{R}^r$ is the vector of modal coefficients in the expansion $\boldsymbol{\mu}^{\text{TSVD}} = \mathbf{V}_e\mathbf{a}$.

Substituting the reduced eigendecomposition of $\mathbf{O}_e^T\mathbf{O}_e = \mathbf{H}_e$ and noting that $\mathbf{V}_e^T\mathbf{V}_e = \mathbf{I}$, (3.4) reduces to $\mathbf{a} = \mathbf{L}_e^{-1}\mathbf{V}_e^\top\mathbf{O}_e^\top\mathbf{y}_d$. Therefore, the TSVD parameter estimate is given by

$$\boldsymbol{\mu}^{\text{TSVD}} = \mathbf{V}_e\mathbf{L}_e^{-1}\mathbf{V}_e^\top\mathbf{O}_e^\top\mathbf{y}_d,$$

involving the well-known pseudoinverse $\mathbf{H}_e^\dagger = \mathbf{V}_e\mathbf{L}_e^{-1}\mathbf{V}_e^\top$. In the traditional two-step approach, this estimate of $\boldsymbol{\mu}$ would then be utilized in simulation to predict output quantities of interest

$$\mathbf{y}_p^{\text{TSVD}} = \mathbf{O}_p\boldsymbol{\mu}^{\text{TSVD}}.$$

It is precisely these prediction outputs that the IFP method will reproduce.

The following derivation of the IFP basis resembles, and in fact was inspired by, balanced truncation model reduction [60]. It is not necessary however for there to exist an underlying state space system to use the IFP method; it is sufficient to have models only for the experiment and prediction operators $\mathbf{O}_e$ and $\mathbf{O}_p$.

Before stating the basis generation algorithm, we first define the key property of the IFP method.

Property 1. *A parameter estimate $\boldsymbol{\mu}^*$ has the IFP property if it results in prediction equal to that of the prediction resulting from the TSVD parameter estimate; i.e.,* $\mathbf{y}_p(\boldsymbol{\mu}^*) = \mathbf{O}_p\boldsymbol{\mu}^* = \mathbf{y}_p^{TSVD}$.

Our goal is to find an s-dimensional subspace $\mathcal{W} \subset \mathbb{R}^q$ such that the IFP solution

$$\boldsymbol{\mu}^{\text{IFP}} = \arg\min_{\boldsymbol{\mu}\in\mathcal{W}} \frac{1}{2}\|\mathbf{y}_d - \mathbf{O}_e\boldsymbol{\mu}\|_2^2 \qquad (3.5)$$

has Property 1. For now, we assume that such a subspace exists.

We will also utilize an assumption regarding the geometry of the experiment and prediction observable subspaces.

Assumption 1. *We will assume throughout that rank*$(\mathbf{V}_p^\top\mathbf{V}_e) = s$.

Assumption 1 codifies the requirement that the experiments contain at least some information about at least one required prediction quantity of interest. It will almost always be the case that this assumption is valid. If it is not (i.e., there is some experiment that provides no information about any required prediction), that experiment can be removed from the process without effect to either the traditional parameter inference formulation or its goal-oriented counterpart. If Assumption 1 holds, we know that the IFP subspace will have dimension s. If $\text{rank}(\mathbf{V}_p^\top \mathbf{V}_e) < s$, the true rank will be exposed implicitly in our algorithm. If $\text{rank}(\mathbf{V}_p^\top \mathbf{V}_e) = 0$, then our algorithm breaks down appropriately, indicating that none of the experiments provide information about any of the required predictions.

We now define the IFP subspace.

Definition 1. *An IFP subspace is an s-dimensional subspace $\mathcal{W}$ such that the solution $\boldsymbol{\mu}^{IFP}$ to (3.5) has Property 1 independent of the data $\mathbf{y}_d$.*

The definition of an IFP basis follows naturally.

Definition 2. *Any basis $\mathbf{W} \in \mathbb{R}^{q\times s}$ is an IFP basis if its columns span an IFP subspace $\mathcal{W}$.*

We now present an algorithm for obtaining an IFP basis $\mathbf{W}$ (we prove it in section 3.3.1) that simultaneously diagonalizes the projected experiment and prediction observability gramians. Although the simultaneous diagonalization is not necessary to replicate the TSVD predictions (any basis for $\mathcal{W}$ will do), it does provide a measure by which further reduction can be performed if desired.

Algorithm 2 IFP Basis Generation for TSVD approach

```
1: Define $\mathbf{G}_e = \mathbf{V}_e \mathbf{L}_e^{-1/2}$.
2: Compute the reduced eigendecomposition $\boldsymbol{\Psi}\boldsymbol{\Sigma}^2\boldsymbol{\Psi}^\top$ of $\mathbf{G}_e^\top \mathbf{O}_p^\top \mathbf{O}_p \mathbf{G}_e$.
3: Define $\mathbf{W} = \mathbf{G}_e \boldsymbol{\Psi}\boldsymbol{\Sigma}^{-1/2}$.
```

The singular values on the diagonal of $\boldsymbol{\Sigma}$ are analogous to the Hankel singular values of balanced truncation. They represent a joint measure of the experiment and prediction observability. While Algorithm 2 identifies a low-dimensional basis for the IFP subspace, it is possible to truncate further (cf. second step of balanced truncation). In this basis, eliminating columns of $\mathbf{W}$ from the right is analogous to removing the least experiment and prediction observable modes according to the joint measure reflected by the singular values.

Using the IFP basis from Algorithm 2, the optimality condition of (3.5) becomes

$$\boldsymbol{\mu}^{\text{IFP}} = \mathbf{W}(\mathbf{W}^\top \mathbf{H}_e \mathbf{W})^{-1}\mathbf{W}^\top \mathbf{O}_e^\top \mathbf{y}_d = \mathbf{W}\boldsymbol{\Sigma}\mathbf{W}^\top \mathbf{O}_e^\top \mathbf{y}_d. \tag{3.6}$$

Thus, when the IFP parameter estimate is computed online, it does not require even the inversion of a small s-by-s matrix, but rather just the application of $\mathbf{W}\boldsymbol{\Sigma}\mathbf{W}^\top \mathbf{O}_e^\top \in \mathbb{R}^{n\times r}$, which can be precomputed, to the observed data $\mathbf{y}_d$.

3.2.3 IFP implementation and computational complexity

Algorithm 2 has two major computational steps. In Step 1 we require the eigendecomposition of the experiment observability gramian $\mathbf{H}_e = \mathbf{O}_e^\top \mathbf{O}_e$, which has rank r. Step 2 involves an eigendecomposition of a matrix of rank s.

For Step 1 efficient implementation should include a code to perform the matrix-vector product $\mathbf{H}_e\mathbf{v}$ as efficiently as possible, both in terms of storage and operation cost. Often times, particularly when the governing equations are given by PDEs, this implies a matrix-free implementation. For problems of interest, the action $\mathbf{H}_e\mathbf{v}$ is one forward and one adjoint solution starting from initial condition $\mathbf{v}$ [14]. Let $\alpha(n)$ be the cost of one time step of the PDE (depending on the mesh DOFs n) and K_e be the number of time steps until the last data are collected. Then the forward and adjoint solutions cost $2\alpha(n)K_e$. Since the experiment observability gramian has rank r, an iterative eigenvalue solver will typically require approximately $2r\alpha(n)K_e$ to obtain the eigendecomposition, assuming the eigenvalues are well separated [23]. Note here that r is independent of n and that $\alpha(n) \sim n$ if appropriate preconditioners are used. If all operations are performed iteratively, the storage requirements should not exceed a small constant number of parameter vectors and therefore scale linearly with q. However, we do assume here that we store the r eigenvalues and eigenvectors for a total storage cost of $(q+1)r = qr + r$.

The computation in Step 2 contains two parts. First, the implementation should include a matrix-free code for computing $\mathbf{H}_p\mathbf{v} = \mathbf{O}_p^\top \mathbf{O}_p\mathbf{v}$. Second, a rank s eigendecomposition must be computed. The code for the prediction observability gramian will also manifest in forward and adjoint solves, although the final time of the simulation $K_p > K_e$. Thus, the cost is approximately $2\alpha(n)K_p$ for each matrix-vector product. This computation sits inside the eigendecomposition which iteratively utilizes a code representing the product $\mathbf{G}_e^\top \mathbf{O}_p^\top \mathbf{O}_p \mathbf{G}_e \mathbf{v} = \mathbf{L}_e^{-1/2}\mathbf{V}_e^\top \mathbf{H}_p \mathbf{V}_e \mathbf{L}_e \mathbf{v}$. Each such product requires in order (from right to left) r scalar products, nr scalar products, r q-vector sums, $2\alpha(n)K_p$ for the action of $\mathbf{H}_p$, r q-vector inner products, and finally another r scalar products. That is a total cost of $2(4q+2)r\alpha(n)K_p$ for each matrix-vector product $\mathbf{G}_e^\top \mathbf{O}_p^\top \mathbf{O}_p \mathbf{G}_e \mathbf{v}$. Since this matrix has rank s, we can expect approximately s such iterations giving a total cost for Step 2 of approximately $2(4q+2)rs\alpha(n)K_p$. There is negligible additional storage required at this step since the storage of the eigenvectors $\mathbf{V}_e$ will dominate. We do store the resulting s eigenvalues and r-dimensional eigenvectors for a storage cost of $(r+1)s$.

If we combine the cost of the first two steps and then account for the final matrix multiplication to obtain $\mathbf{W}$, we have a total operation cost of approximately $2r\alpha(n)K_e + 2(4q+2)rs\alpha(n)K_p + qrs^2$ and total storage cost of approximately $(q+1)r + (r+1)s + qs$. While the IFP method may be more computationally expensive than traditional inference procedures for the solution of one-off inverse problems; the benefits of the IFP method are three-fold. First, if data are collected repeatedly under the same experimental observation operator, then the cost of determining the IFP basis can be amortized over the experiments. Second, the IFP basis encodes important information about the process relating inference and prediction, in particular through an analysis of the range of the IFP basis as it compares to the ranges of the gramians of the experiment and prediction processes. The study of this relationship could play a role in determining the effects of regularization and in designing future experiments. Lastly, and perhaps most importantly, since the IFP method has data-independent theory, it is feasible to move all of this computation offline and utilize only the resulting basis in an online deployment. This offline-online decomposition of cost can make the IFP approach efficient for inverse problem solutions on lightweight, portable devices in the field.

3.3 IFP Linear Theory

In this section we develop the relevant theory for the IFP method in linear inverse problems. We first prove that the basis generated by Algorithm 2 leads to Property 1. We give a geometric interpretation of the IFP method in this setting. Finally, the section is concluded with a proof of dimensional optimality, showing that there is no subspace of dimension less than s that produces a solution of (3.5) that has Property 1, and a proof of the uniqueness of the IFP subspace.

3.3.1 Prediction exactness

We first show that the basis generated by Algorithm 2 defines an IFP subspace, i.e., that the solution to (3.5) has Property 1.

Theorem 1. *Algorithm 2 leads to a basis* $\mathbf{W}$ *whose columns span an IFP subspace. Therefore, the solution* $\boldsymbol{\mu}^{\mathrm{IFP}}$ *to (3.5) with* $\mathcal{W} = \mathit{range}(\mathbf{W})$ *has Property 1.*

Proof. Let $\mathbf{U} = \mathbf{O}_p^\top \mathbf{O}_p \mathbf{G}_e \boldsymbol{\Psi} \boldsymbol{\Sigma}^{-3/2}$ and note that the columns of $\mathbf{U}$ are a basis for the row space of $\mathbf{O}_p$. Thus, any two parameter estimates $\boldsymbol{\mu}_1$ and $\boldsymbol{\mu}_2$ satisfying $\mathbf{U}^\top(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2) = \mathbf{0}$ will have the same prediction $\mathbf{O}_p\boldsymbol{\mu}_1 = \mathbf{O}_p\boldsymbol{\mu}_2$. We will now show $\mathbf{U}^\top(\boldsymbol{\mu}^{\mathrm{IFP}} - \boldsymbol{\mu}^{\mathrm{TSVD}}) = \mathbf{0}$. Substituting the optimality conditions for the TSVD and IFP optimization problems, we find

$$\mathbf{U}^\top(\boldsymbol{\mu}^{\mathrm{IFP}} - \boldsymbol{\mu}^{\mathrm{TSVD}}) = \mathbf{U}^\top(\mathbf{W}(\mathbf{W}^\top \mathbf{H}_e \mathbf{W})^{-1}\mathbf{W}^\top \mathbf{O}_e^\top - \mathbf{V}_e \mathbf{L}_e^{-1} \mathbf{V}_e^\top \mathbf{O}_e^\top)\mathbf{y}_d. \tag{3.7}$$

By construction, we have

$$\mathbf{U}^\top \mathbf{W} = \boldsymbol{\Sigma}^{-3/2} \boldsymbol{\Psi}^\top \mathbf{G}_e^\top \mathbf{O}_p^\top \mathbf{O}_p \mathbf{G}_e \boldsymbol{\Psi} \boldsymbol{\Sigma}^{-1/2} = \boldsymbol{\Sigma}^{-3/2} \boldsymbol{\Sigma}^2 \boldsymbol{\Sigma}^{-1/2} = \mathbf{I},$$

where we have used the orthonormality of $\mathbf{\Psi}$ and the eigendecomposition from Algorithm 2. Furthermore, the basis $\mathbf{W}$ satisfies the relation

$$\begin{aligned}\mathbf{W}^\top\mathbf{H}_e\mathbf{W} &= \mathbf{\Sigma}^{-1/2}\mathbf{\Psi}^\top\mathbf{G}_e^\top\mathbf{V}_e\mathbf{L}_e\mathbf{V}_e^\top\mathbf{G}_e\mathbf{\Psi}\mathbf{\Sigma}^{-1/2},\\ &= \mathbf{\Sigma}^{-1/2}\mathbf{\Psi}^\top\mathbf{L}_e^{-1/2}\mathbf{V}_e^\top\mathbf{V}_e\mathbf{L}_e\mathbf{V}_e^\top\mathbf{V}_e\mathbf{L}_e^{-1/2}\mathbf{\Psi}\mathbf{\Sigma}^{-1/2} = \mathbf{\Sigma}^{-1/2}\mathbf{\Psi}^\top\mathbf{\Psi}\mathbf{\Sigma}^{-1/2} = \mathbf{\Sigma}^{-1}.\end{aligned}$$

Using these facts, (3.7) reduces to

$$\mathbf{U}^\top(\boldsymbol{\mu}^{\text{IFP}} - \boldsymbol{\mu}^{\text{TSVD}}) = (\mathbf{\Sigma}\mathbf{W}^\top\mathbf{O}_e^\top - \mathbf{U}^\top\mathbf{V}_e\mathbf{L}_e^{-1}\mathbf{V}_e^\top\mathbf{O}_e^\top)\mathbf{y}_d.$$

We have now $\mathbf{\Sigma}\mathbf{W}^\top\mathbf{O}_e^\top = \mathbf{\Sigma}^{1/2}\mathbf{\Psi}^\top\mathbf{G}_e^\top\mathbf{O}_e^\top$ and

$$\mathbf{U}^\top\mathbf{V}_e\mathbf{L}_e^{-1}\mathbf{V}_e^\top\mathbf{O}_e^\top = \mathbf{\Sigma}^{-3/2}\mathbf{\Psi}^\top\mathbf{G}_e^\top\mathbf{O}_p^\top\mathbf{O}_p\mathbf{G}_e\mathbf{G}_e^\top\mathbf{O}_e^\top = \mathbf{\Sigma}^{1/2}\mathbf{\Psi}^\top\mathbf{G}_e^\top\mathbf{O}_e^\top.$$

This demonstrates that $\mathbf{U}^\top(\boldsymbol{\mu}^{\text{IFP}} - \boldsymbol{\mu}^{\text{TSVD}}) = (\mathbf{\Sigma}^{1/2}\mathbf{\Psi}^\top\mathbf{G}_e^\top\mathbf{O}_e^\top - \mathbf{\Sigma}^{1/2}\mathbf{\Psi}^\top\mathbf{G}_e^\top\mathbf{O}_e^\top)\mathbf{y}_d = \mathbf{0}$, and therefore proves that $\mathbf{y}_p^{\text{IFP}} = \mathbf{y}_p^{\text{TSVD}}$ for all data $\mathbf{y}_d$. □

Note that Theorem 1 holds irrespective of data $\mathbf{y}_d$. The implication is that the IFP method inherits many of the characteristics of the inference formulation. The method does not change the estimation methodology but rather circumvents some of the superfluous computation in the context of predictions. In particular, the IFP method inherits the sensitivity to noise of the TSVD approach, in this case.

An analogous approach can be utilized for any inverse problem formulation that has the form of a filter, of which TSVD is an example [39]. Let $\mathbf{O}_e = \mathbf{U}_e\mathbf{\Sigma}_e\mathbf{V}_e^\top$ be the reduced SVD of the experimental operator. Then the filtered parameter estimate can be written as $\boldsymbol{\mu}^{\text{filt}} = \mathbf{V}_e\mathbf{\Phi}\mathbf{\Sigma}_e^{-1}\mathbf{U}_e^\top\mathbf{y}_d$ for the filter weighting matrix $\mathbf{\Phi} \in \mathbb{R}^{r\times r}$. The IFP approach extends readily to this context. We demonstrate in section 3.4.1 that this is the case for Tikhonov-regularized inverse problems.

3.3.2 Geometric interpretation

Of particular interest is the geometry of the approach. The solution $\boldsymbol{\mu}^{\text{IFP}}$ is obtained as the oblique projection of $\boldsymbol{\mu}^{\text{TSVD}}$ based on the projector $\mathbf{\Pi} = \mathbf{G}_e\mathbf{\Psi}\mathbf{\Psi}^\top\mathbf{G}_e^\top\mathbf{H}_e$. That is to say, $\boldsymbol{\mu}^{\text{IFP}} = \mathbf{\Pi}\boldsymbol{\mu}^{\text{TSVD}}$ independent of the data. We show first that $\mathbf{\Pi}$ is an oblique projector.

Theorem 2. *The matrix* $\mathbf{\Pi} = \mathbf{G}_e\mathbf{\Psi}\mathbf{\Psi}^\top\mathbf{G}_e^\top\mathbf{H}_e$ *is an oblique projector.*

Proof. We first show that $\mathbf{\Pi}$ is a projector, and then we establish that its range and null space are not orthogonal complements. We have

$$\mathbf{\Pi}^2 = \mathbf{G}_e\mathbf{\Psi}\mathbf{\Psi}^\top\mathbf{G}_e^\top\mathbf{H}_e\mathbf{G}_e\mathbf{\Psi}\mathbf{\Psi}^\top\mathbf{G}_e^\top\mathbf{H}_e = \mathbf{G}_e\mathbf{\Psi}\mathbf{\Psi}^\top\mathbf{\Psi}\mathbf{\Psi}^\top\mathbf{G}_e^\top\mathbf{H}_e = \mathbf{G}_e\mathbf{\Psi}\mathbf{\Psi}^\top\mathbf{G}_e^\top\mathbf{H}_e = \mathbf{\Pi}.$$

Since $\mathbf{\Pi}^2 = \mathbf{\Pi}$, $\mathbf{\Pi}$ is a projector. An orthogonal projector has orthogonal range and null spaces. Any projector that is not an orthogonal projector is an oblique projector. Therefore, it suffices for us to obtain a vector $\mathbf{v} \in \mathbb{R}^q$ such that $(\mathbf{\Pi}\mathbf{v})^\top(\mathbf{v} - \mathbf{\Pi}\mathbf{v}) \neq 0$ to show that $\mathbf{\Pi}$ is an oblique projector, since $\mathbf{\Pi}\mathbf{v} \in \text{range}(\mathbf{\Pi})$ and $\mathbf{v} - \mathbf{\Pi}\mathbf{v} \in \text{null}(\mathbf{\Pi})$.

We assume that $\mathbf{L}_e \neq \mathbf{I}$ in general; if it is, then $\mathbf{\Pi}$ is an orthogonal projector. Let $\mathbf{z} \in \mathbb{R}^r$ be chosen such that $\mathbf{\Psi}^\top\mathbf{z} = \mathbf{0}$ but that $\mathbf{\Psi}^\top\mathbf{L}_e\mathbf{z} \neq \mathbf{0}$. Define then $\mathbf{v} = \mathbf{V}_e\mathbf{L}_e^{1/2}\mathbf{z}$. Then if we write out the expression above, we find

$$(\mathbf{\Pi}\mathbf{v})^\top(\mathbf{v} - \mathbf{\Pi}\mathbf{v}) = \mathbf{v}^\top\mathbf{V}_e\mathbf{\Lambda}\mathbf{V}_e^\top\mathbf{v} - \mathbf{v}^\top\mathbf{V}_e\mathbf{\Lambda}\mathbf{\Lambda}^\top\mathbf{V}_e^\top\mathbf{v} \tag{3.8}$$

where $\mathbf{\Lambda} = \mathbf{L}_e^{1/2}\mathbf{\Psi}\mathbf{\Psi}^\top\mathbf{L}_e^{-1/2}$. Based on our choice of $\mathbf{v}$ above, we find that the first term on the right hand side vanishes; i.e.,

$$\mathbf{v}^\top\mathbf{V}_e\mathbf{\Lambda}\mathbf{V}_e^\top\mathbf{v} = \mathbf{z}^\top\mathbf{L}_e\mathbf{\Psi}\mathbf{\Psi}^\top\mathbf{z} = \mathbf{0}.$$

The second term on the right hand side of (3.8) can be rewritten as $\|\mathbf{\Lambda}^\top\mathbf{V}_e^\top\mathbf{v}\|^2 = \|\mathbf{L}_e^{-1/2}\mathbf{\Psi}\mathbf{\Psi}^\top\mathbf{L}_e\mathbf{z}\|^2 \geq 0$. Since we chose $\mathbf{z}$ such that $\mathbf{\Psi}^\top\mathbf{L}_e\mathbf{z} \neq \mathbf{0}$, we have that the second term is positive. This implies that we have found a $\mathbf{v}$ such that $(\mathbf{\Pi}\mathbf{v})^\top(\mathbf{v} - \mathbf{\Pi}\mathbf{v}) \neq 0$, and therefore $\mathbf{\Pi}$ is an oblique projector. □

Theorem 3. *The parameter estimate $\boldsymbol{\mu}^{\mathrm{IFP}}$ obtained using the IFP method is the oblique projection under $\mathbf{\Pi}$ of the TSVD solution $\boldsymbol{\mu}^{\mathrm{TSVD}}$.*

Proof. Using the basis $\mathbf{W}$ obtained by Algorithm 2 the IFP solution is computed as

$$\boldsymbol{\mu}^{\mathrm{IFP}} = \mathbf{W}(\mathbf{W}^\top \mathbf{H}_e \mathbf{W})^{-1} \mathbf{W}^\top \mathbf{O}_e^\top \mathbf{y}_d.$$

Since $\mathbf{O}_e^\top(\mathbf{y}_d - \mathbf{O}_e \boldsymbol{\mu}^{\mathrm{TSVD}}) = \mathbf{0}$, we have

$$\boldsymbol{\mu}^{\mathrm{IFP}} = \mathbf{W}(\mathbf{W}^\top \mathbf{H}_e \mathbf{W})^{-1} \mathbf{W}^\top \mathbf{H}_e \boldsymbol{\mu}^{\mathrm{TSVD}}.$$

Recalling that $\mathbf{W}^\top \mathbf{H}_e \mathbf{W} = \mathbf{\Sigma}^{-1}$ and that $\mathbf{W} = \mathbf{G}_e \mathbf{\Psi} \mathbf{\Sigma}^{-1/2}$, we have

$$\boldsymbol{\mu}^{\mathrm{IFP}} = \mathbf{G}_e \mathbf{\Psi} \mathbf{\Psi}^\top \mathbf{G}_e^\top \mathbf{H}_e \boldsymbol{\mu}^{\mathrm{TSVD}} = \mathbf{\Pi} \boldsymbol{\mu}^{\mathrm{TSVD}}.$$

□

It can be shown directly that, for $\boldsymbol{\mu}^{\mathrm{TSVD}} \in \mathrm{null}(\mathbf{O}_p)$, the projection is zero, as expected. If the resulting parameter estimate has no prediction observable components, it must lead to zero prediction. One can also show that, under Assumption 1, if $\boldsymbol{\mu}^{\mathrm{TSVD}} \notin \mathrm{null}(\mathbf{O}_p)$, the projection is nonzero, meaning that a nonzero prediction will result.

3.3.3 Dimensional optimality of the IFP subspace

It is natural to ask if the IFP subspace of dimension s is the subspace of minimum dimension such that the solution of (3.5) has Property 1. We now show that there does not exist a $\tilde{s}$-dimensional subspace $\tilde{\mathcal{W}}$ for $\tilde{s} < s$ such that the solution of (3.5) has Property 1.

Theorem 4. *The IFP subspace $\mathcal{W}$ is the subspace of minimum dimension such that the solution to (3.5) has Property 1.*

Proof. In view of Assumption 1, the predictable component of the TSVD solution is obtained from the data by the operation

$$\boldsymbol{\mu}_p^{\mathrm{TSVD}} = \mathbf{V}_p\mathbf{V}_p^\top\boldsymbol{\mu}^{\mathrm{TSVD}} = \mathbf{V}_p\mathbf{V}_p^\top\mathbf{V}_e\mathbf{L}_e^{-1}\mathbf{V}_e^\top\mathbf{O}_e^\top\mathbf{y}_d$$

where the matrix $\mathbf{V}_p\mathbf{V}_p^\top\mathbf{V}_e\mathbf{L}_e^{-1}\mathbf{V}_e^\top\mathbf{O}_e^\top$ transforming data $\mathbf{y}_d$ to $\boldsymbol{\mu}_p^{\mathrm{TSVD}}$ has rank s. Let $\tilde{s} < s$ and $\tilde{\mathbf{W}} \in \mathbb{R}^{q\times\tilde{s}}$ be a basis for any $\tilde{s}$-dimensional subspace $\tilde{\mathcal{W}}$. Based on the IFP formulation the matrix from data $\mathbf{y}_d$ to predictable component of the IFP estimate $\boldsymbol{\mu}_p^{\mathrm{IFP}} = \mathbf{V}_p\mathbf{V}_p^\top\boldsymbol{\mu}^{\mathrm{IFP}}$ is $\mathbf{V}_p\mathbf{V}_p^\top\tilde{\mathbf{W}}(\tilde{\mathbf{W}}^\top\mathbf{H}_e\tilde{\mathbf{W}})^{-1}\tilde{\mathbf{W}}^\top\mathbf{O}_e^\top$. In order for $\boldsymbol{\mu}_p^{\mathrm{IFP}} = \boldsymbol{\mu}_p^{\mathrm{TSVD}}$ for arbitrary $\mathbf{y}_d$ it must be the case that $\mathbf{V}_p\mathbf{V}_p^\top\mathbf{V}_e\mathbf{L}_e^{-1}\mathbf{V}_e^\top\mathbf{O}_e^\top = \mathbf{V}_p\mathbf{V}_p^\top\tilde{\mathbf{W}}(\tilde{\mathbf{W}}^\top\mathbf{H}_e\tilde{\mathbf{W}})^{-1}\tilde{\mathbf{W}}^\top\mathbf{O}_e^\top$. However, we know that the matrix on the left has rank s and the matrix on the right has rank $\tilde{s} \neq s$, establishing a contradiction. Therefore, a basis permitting Property 1 must have dimension at least s. Thus, an IFP subspace, which has dimension s, is dimensionally optimal. □

3.3.4 Uniqueness of the IFP subspace

Clearly the basis for the IFP subspace is not unique, but we show in this section that the subspace is indeed unique. The algorithm provided above is designed to provide a balanced basis (with respect to experiment and prediction observability) of this subspace.

Theorem 5. *Let $\mathcal{V}_e$ be the subspace spanned by the columns of $\mathbf{V}_e$. Consider a decomposition of $\mathcal{V}_e$ into two $\mathbf{H}_e$-orthogonal subspaces $\mathcal{W}$ and $\mathcal{U}$ of dimensions s and $r - s$ respectively, where $\mathcal{U}$ is an $\mathbf{H}_p$-orthogonal subspace. The solution of (3.5) with the subspace $\mathcal{W}$ has Property 1; therefore $\mathcal{W}$ is an IFP subspace.*

Proof. Let $\mathbf{W} \in \mathbb{R}^{q\times r}$ and $\mathbf{U} \in \mathbb{R}^{q\times(r-s)}$ be bases for the subspaces $\mathcal{W}$ and $\mathcal{U}$ respectively. Since we impose that $\mathcal{W} + \mathcal{U} = \mathcal{V}_e$, we can represent the TSVD solution as $\boldsymbol{\mu}^{\mathrm{TSVD}} = \mathbf{W}\mathbf{a} + \mathbf{U}\mathbf{b}$ for coefficient vectors $\mathbf{a} \in \mathbb{R}^s$ and $\mathbf{b} \in \mathbb{R}^{r-s}$. Rewriting the TSVD parameter estimate, we obtain matrix equations for the unknown coefficient vectors

$$\begin{bmatrix} \mathbf{W}^\top \mathbf{H}_e \mathbf{W} & \mathbf{W}^\top \mathbf{H}_e \mathbf{U} \\ \mathbf{U}^\top \mathbf{H}_e \mathbf{W} & \mathbf{U}^\top \mathbf{H}_e \mathbf{U} \end{bmatrix} \begin{bmatrix} \mathbf{a} \\ \mathbf{b} \end{bmatrix} = \begin{bmatrix} \mathbf{W}^\top \mathbf{O}_e^\top \mathbf{y}_d \\ \mathbf{U}^\top \mathbf{O}_e^\top \mathbf{y}_d \end{bmatrix}. \tag{3.9}$$

Imposing the $\mathbf{H}_e$-orthogonality of $\mathbf{W}$ and $\mathbf{U}$ yields a decoupled block-diagonal system with corresponding TSVD parameter estimate

$$\boldsymbol{\mu}^{\text{TSVD}} = \mathbf{W}(\mathbf{W}^\top \mathbf{H}_e \mathbf{W})^{-1} \mathbf{W}^\top \mathbf{O}_e^\top \mathbf{y}_d + \mathbf{U}(\mathbf{U}^\top \mathbf{H}_e \mathbf{U})^{-1} \mathbf{U}^\top \mathbf{O}_e^\top \mathbf{y}_d.$$

The corresponding prediction is then

$$\begin{aligned} \mathbf{y}_p &= \mathbf{O}_p \mathbf{W}(\mathbf{W}^\top \mathbf{H}_e \mathbf{W})^{-1} \mathbf{W}^\top \mathbf{O}_e^\top \mathbf{y}_d + \mathbf{O}_p \mathbf{U}(\mathbf{U}^\top \mathbf{H}_e \mathbf{U})^{-1} \mathbf{U}^\top \mathbf{O}_e^\top \mathbf{y}_d, \\ &= \mathbf{O}_p \mathbf{W}(\mathbf{W}^\top \mathbf{H}_e \mathbf{W})^{-1} \mathbf{W}^\top \mathbf{O}_e^\top \mathbf{y}_d, \end{aligned}$$

where the second term vanishes due to the $\mathbf{H}_p$-orthogonality of $\mathcal{U}$, which is equivalent to $\mathbf{O}_p \mathbf{U} = \mathbf{0}$. What remains is the prediction resulting from the IFP estimate $\boldsymbol{\mu}^{\text{IFP}} = \mathbf{W}(\mathbf{W}^\top \mathbf{H}_e \mathbf{W})^{-1} \mathbf{W}^\top \mathbf{O}_e^\top \mathbf{y}_d$, showing that $\mathcal{W}$ is an IFP subspace. □

It is straightforward to show that the balanced IFP basis of Algorithm 2 defines an IFP subspace that has these properties. The corresponding subspace $\mathcal{U}$ is defined by a basis $\mathbf{U} = \mathbf{V}_e \mathbf{L}_e^{-1/2} \tilde{\boldsymbol{\Psi}}$ where $\tilde{\boldsymbol{\Psi}} \in \mathbb{R}^{r \times (r-s)}$ is defined such that $\boldsymbol{\Psi}^\top \tilde{\boldsymbol{\Psi}} = \mathbf{0}$.

Having specified the general conditions for an IFP subspace $\mathcal{W}$, we are now in a position to prove uniqueness.

Theorem 6. *If Assumption 1 holds, then the IFP subspace is unique.*

Proof. Let $\mathcal{V}_e = \text{range}(\mathbf{H}_e)$ and $\mathcal{V}_p = \text{range}(\mathbf{H}_p)$. Consider first the identification of the subspace $\mathcal{U}$. We have that $\mathcal{U} = \mathcal{V}_e \mathbf{B}$ for a full rank matrix $\mathbf{B} \in \mathbb{R}^{r \times (r-s)}$ imposing the dimensionality and that $\mathcal{U} \subset \mathcal{V}_e$. The $\mathbf{H}_p$-orthogonality of $\mathcal{U}$ then implies that $\mathcal{V}_p^\top \mathcal{V}_e \mathbf{B} = 0$. Since we have forbidden the orthogonality of $\mathcal{V}_e$ and $\mathcal{V}_p$ in any way via Assumption 1, $\mathcal{V}_p^\top \mathcal{V}_e \subset \mathbb{R}^r$ where $\dim(\mathcal{V}_p^\top \mathcal{V}_e) = s$. Therefore, $\mathcal{V}_p^\top \mathcal{V}_e$ has an $(r-s)$-dimensional null space, making unique the space spanned by the columns of $\mathbf{B}$. Now define $\mathcal{W} = \mathcal{V}_e\mathbf{A}$ for full rank matrix $\mathbf{A} \in \mathbb{R}^{r\times s}$. The $\mathbf{H}_e$-orthogonality of $\mathcal{W}$ and $\mathcal{U}$ implies that $\mathbf{B}^\top\mathbf{L}_e\mathbf{A} = \mathbf{0}$ where $\mathbf{L}_e \in \mathbb{R}^{r\times r}$ is a reduced diagonal matrix of the eigenvalues of $\mathbf{H}_e$. The product $\mathbf{B}^\top\mathbf{L}_e \in \mathbb{R}^{(r-s)\times r}$ has a null space of dimension s. Consequently, the space spanned by the columns of $\mathbf{A}$ is unique. Therefore, the subspace $\mathcal{W}$ is uniquely determined. □

3.4 Extensions of the IFP method

We extend the IFP method to Tikhonov-regularized inverse problems and Gaussian statistical inverse problems in this section. It is shown that only a small modification to the IFP method above is necessary to apply the goal-oriented approach to these cases.

3.4.1 Tikhonov-regularized inverse problem

Another method for regularizing ill-posed inverse problems requires adding a penalty term to the objective function [27]. The idea is to select the parameter that most closely matches the experimental data while also conforming to a certain extent with an a priori structural preference. The main effect is a modification of the experiment observability gramian in the algorithm.

A Tikhonov-regularized inverse problem [27] has the form

$$\boldsymbol{\mu}^{\mathrm{TR}} = \arg\min_{\boldsymbol{\mu}\in\mathbb{R}^q} \frac{1}{2}\|\mathbf{y}_d - \mathbf{O}_e\boldsymbol{\mu}\|_2^2 + \frac{1}{2}\|\mathbf{R}\boldsymbol{\mu}\|_2^2 \tag{3.10}$$

where we assume the regularization parameter weighting the two terms has been incorporated into the regularization matrix $\mathbf{R}$. For these formulations, the experiment observability gramian becomes $\mathbf{O}_e^\top\mathbf{O}_e + \mathbf{R}^\top\mathbf{R}$ where $\mathbf{R}^\top\mathbf{R}$ is assumed to fill at least the null space of $\mathbf{O}_e^\top\mathbf{O}_e$ making the problem (3.10) well-posed.

The optimality condition for the Tikhonov-regularized inverse problem (3.10) is given by

$$(\mathbf{O}_e^\top\mathbf{O}_e + \mathbf{R}^\top\mathbf{R})\boldsymbol{\mu}^{\mathrm{TR}} = \mathbf{O}_e^\top\mathbf{y}_d,$$

where we assume that $\mathbf{R}$ is chosen such that $\text{null}(\mathbf{O}_e^\top \mathbf{O}_e + \mathbf{R}^\top \mathbf{R}) = \{\mathbf{0}\}$. The solution of (3.10) is then given by

$$\boldsymbol{\mu}^{\text{TR}} = (\mathbf{O}_e^\top \mathbf{O}_e + \mathbf{R}^\top \mathbf{R})^{-1} \mathbf{O}_e^\top \mathbf{y}_d,$$

and the associated prediction is

$$\mathbf{y}_p^{\text{TR}} = \mathbf{O}_p \boldsymbol{\mu}^{\text{TR}}.$$

We will now show that we can modify the IFP method for the TSVD approach in section 3.2.2 to once again replicate the predictions without inverting for all of the parametric modes. The key here is a modification to the experiment observability gramian. In particular, we have $\mathbf{H}_e = \mathbf{O}_e^\top \mathbf{O}_e + \mathbf{R}^\top \mathbf{R}$. Given an IFP subspace $\mathcal{W}$, we obtain the IFP solution

$$\boldsymbol{\mu}^{\text{IFP}} = \arg \min_{\boldsymbol{\mu} \in \mathcal{W}} \frac{1}{2} \|\mathbf{y}_d - \mathbf{O}_e \boldsymbol{\mu}\|_2^2 + \frac{1}{2} \|\mathbf{R}\boldsymbol{\mu}\|_2^2. \tag{3.11}$$

However, the IFP basis $\mathbf{W}$ is now obtained by Algorithm 3.

Algorithm 3 IFP Basis Generation for Tikhonov-regularized approach

```
1: Define $\mathbf{G}_e = \mathbf{V}_e \mathbf{L}_e^{-1/2}$ where $\mathbf{V}_e \mathbf{L}_e \mathbf{V}_e^\top$ is the eigendecomposition of $\mathbf{O}_e^\top \mathbf{O}_e + \mathbf{R}^\top \mathbf{R}$.
2: Compute the reduced eigendecomposition $\boldsymbol{\Psi} \boldsymbol{\Sigma}^2 \boldsymbol{\Psi}^\top$ of $\mathbf{G}_e^\top \mathbf{O}_p^\top \mathbf{O}_p \mathbf{G}_e$.
3: Define $\mathbf{W} = \mathbf{G}_e \boldsymbol{\Psi} \boldsymbol{\Sigma}^{-1/2}$.
```

The eigendecomposition in Step 1 of Algorithm 3 will lead to square eigenvector matrices $\mathbf{V}_e \in \mathbb{R}^{n \times n}$ since $\mathbf{H}_e$ is full rank by design of $\mathbf{R}$, which increases both the operation and storage cost of the algorithm. However, since $\mathbf{R}$ is specified, the cost of each matrix-vector product $\mathbf{H}_e \mathbf{v}$ should not be significantly greater than the cost for the unregularized experiment observability gramian in the TSVD approach in section 3.2.2.

Theorem 7. *The predictions $\mathbf{y}_p^{\text{IFP}}$ arising from the IFP solution $\boldsymbol{\mu}^{\text{IFP}}$ of (3.11) with basis $\mathbf{W}$ defined by Algorithm 3 are identical to the Tikhonov-regularized predictions $\mathbf{y}_p^{\text{TR}}$.*

Proof. The proof is exactly the same as the proof of Theorem 1. Both Algorithm 2 and Algorithm 3 work with the eigendecomposition of $\mathbf{H}_e$, which has been suitably redefined for the Tikhonov-regularized inverse problem here. □

3.4.2 Linear Gaussian statistical inverse problem

One way to account for uncertainty in prior knowledge and uncertainty in sensor measurements is through a statistical formulation of the inverse problem. In this section, we demonstrate how the IFP methodology can be extended to the statistical setting using a Bayesian approach with a Gaussian prior and Gaussian likelihood. The solution to the statistical inverse problem is a random variable and therefore has a distribution, which in this case is also Gaussian due to the linearity. That distribution over the parameter is then propagated through to the prediction resulting in a distribution over predictions that we refer to as the *posterior predictive*. Instead of finding a single estimate of the predictions, we will determine a mean and covariance estimate. The mean estimate is obtained by the IFP method for a specific Tikhonov-regularized inverse problem; i.e., the procedure discussed in section 3.4.1 is all that is required. We show that the covariance estimate can be obtained at minimal additional cost through matrix multiplications involving the IFP basis $\mathbf{W}$ and singular value matrix $\boldsymbol{\Sigma}$.

Let $\boldsymbol{\mu} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Gamma}_0)$ be the multivariate Gaussian random variable with mean $\mathbf{0}$ and covariance $\boldsymbol{\Gamma}_0$ representing our prior knowledge of the unknown parameter.[2] We assume that the measurements we make are corrupted by independent additive Gaussian errors $\boldsymbol{\epsilon} = \mathbf{y}_d - \mathbf{O}_e\boldsymbol{\mu} \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I})$ with zero mean and variance σ^2.

Given that the map from parameters to experimental outputs is linear, by Bayes's rule, we write the posterior estimate of the parameter

$$\boldsymbol{\mu}|\mathbf{y}_d \sim \mathcal{N}(\boldsymbol{\mu}_\pi, \boldsymbol{\Gamma}_\pi)$$

[2] The method readily admits priors with nonzero mean. Both the traditional approach and IFP method would then target the deviation from the mean; the covariance remains unchanged.

where

$$\begin{aligned}\boldsymbol{\mu}_\pi &= \sigma^{-2}(\boldsymbol{\Gamma}_0^{-1} + \sigma^{-2}\mathbf{O}_e^\top \mathbf{O}_e)^{-1}\mathbf{O}_e^\top \mathbf{y}_d, \qquad (3.12)\\ \boldsymbol{\Gamma}_\pi &= (\boldsymbol{\Gamma}_0^{-1} + \sigma^{-2}\mathbf{O}_e^\top \mathbf{O}_e)^{-1}.\end{aligned}$$

Recall however that we are interested only in the statistics of the prediction arising from simulations utilizing this parameter; that is, the posterior predictive

$$\mathbf{y}_p|\mathbf{y}_d \sim \mathcal{N}(\mathbf{O}_p\boldsymbol{\mu}_\pi, \mathbf{O}_p\boldsymbol{\Gamma}_\pi\mathbf{O}_p^\top).$$

It is this posterior predictive distribution $\mathbf{y}_p|\mathbf{y}_d$ that will be replicated by the IFP method.

We will now show that the IFP approach can obtain the posterior predictive. First note that the posterior predictive mean is obtained as the solution to a Tikhonov-regularized inverse problem.

Theorem 8. *The posterior predictive mean* $\mathbf{O}_p\boldsymbol{\mu}_\pi$ *is obtained by solving (3.11) with* $\mathbf{W}$ *generated by Algorithm 3 where* $\mathbf{R}$ *is chosen such that* $\mathbf{R}^\top\mathbf{R} = \sigma^2\boldsymbol{\Gamma}_0^{-1}$.

Proof. We first rewrite the Tikhonov-regularized inverse problem (3.10) to account for the measurement error and prior knowledge; i.e., we search for the parameter

$$\begin{aligned}\boldsymbol{\mu}^* &= \arg\min_{\boldsymbol{\mu}\in\mathbb{R}^q} \frac{1}{2}\|\mathbf{y}_d - \mathbf{O}_e\boldsymbol{\mu}\|_2^2 + \frac{1}{2}\sigma^2\boldsymbol{\mu}^\top\boldsymbol{\Gamma}_0^{-1}\boldsymbol{\mu},\\ &= \arg\min_{\boldsymbol{\mu}\in\mathbb{R}^q} \frac{1}{2\sigma^2}\|\mathbf{y}_d - \mathbf{O}_e\boldsymbol{\mu}\|_2^2 + \frac{1}{2}\boldsymbol{\mu}^\top\boldsymbol{\Gamma}_0^{-1}\boldsymbol{\mu}. \qquad (3.13)\end{aligned}$$

We now show that this is precisely the posterior mean. The first-order optimality condition of (3.13) is given by

$$(\boldsymbol{\Gamma}_0^{-1} + \sigma^{-2}\mathbf{O}_e^\top\mathbf{O}_e)\boldsymbol{\mu}^* = \sigma^{-2}\mathbf{O}_e^\top\mathbf{y}_d$$

whose solution is $\boldsymbol{\mu}^* = \sigma^{-2}(\boldsymbol{\Gamma}_0^{-1} + \sigma^{-2}\mathbf{O}_e^\top\mathbf{O}_e)^{-1}\mathbf{O}_e^\top\mathbf{y}_d$. This is equal to $\boldsymbol{\mu}_\pi$ given in (3.12). The remainder of the proof is completely analogous to that of Theorem 7. □

The following theorem states that the posterior predictive covariance can be recovered by a matrix multiplication involving the IFP basis $\mathbf{W}$ and the diagonal matrix of singular values $\mathbf{\Sigma}$ already computed in the posterior predictive mean obtained above.

Theorem 9. *The posterior predictive covariance can be obtained as a matrix multiplication involving the IFP basis* $\mathbf{W}$ *and singular values* $\mathbf{\Sigma}$ *from Algorithm 3 since*

$$\mathbf{O}_p\mathbf{\Gamma}_\pi\mathbf{O}_p^\top = \mathbf{O}_p\mathbf{W}\mathbf{\Sigma}\mathbf{W}^\top\mathbf{O}_p^\top.$$

Proof. The posterior predictive covariance is given by

$$\mathbf{O}_p\mathbf{\Gamma}_\pi\mathbf{O}_p^\top = \mathbf{O}_p(\mathbf{\Gamma}_0^{-1} + \sigma^{-2}\mathbf{O}_e^\top\mathbf{O}_e)^{-1}\mathbf{O}_p^\top. \tag{3.14}$$

Recall that $\mathbf{H}_e = \mathbf{\Gamma}_0^{-1} + \sigma^{-2}\mathbf{O}_e^\top\mathbf{O}_e$ is full rank; therefore, $\mathbf{H}_e^{-1} = \mathbf{V}_e\mathbf{L}_e^{-1}\mathbf{V}_e^\top$ and $\mathbf{V}_e\mathbf{L}_e^{-1}\mathbf{V}_e^\top = \mathbf{G}_e\mathbf{G}_e^\top$. Substituting into (3.14) we find

$$\mathbf{O}_p\mathbf{\Gamma}_\pi\mathbf{O}_p^\top = \mathbf{O}_p\mathbf{G}_e\mathbf{G}_e^\top\mathbf{O}_p^\top. \tag{3.15}$$

Since $\mathrm{range}(\mathbf{\Psi})^\perp \subset \mathrm{null}(\mathbf{O}_p\mathbf{G}_e)$ and $(\mathbf{I} - \mathbf{\Psi}\mathbf{\Psi}^\top)$ is the orthogonal projector onto $\mathrm{range}(\mathbf{\Psi})^\perp$, we have $\mathbf{O}_p\mathbf{G}_e(\mathbf{I} - \mathbf{\Psi}\mathbf{\Psi}^\top) = \mathbf{0}$ and therefore $\mathbf{O}_p\mathbf{G}_e = \mathbf{O}_p\mathbf{G}_e\mathbf{\Psi}\mathbf{\Psi}^\top$. Substituting into (3.15),

$$\mathbf{O}_p\mathbf{\Gamma}_\pi\mathbf{O}_p^\top = \mathbf{O}_p\mathbf{G}_e\mathbf{\Psi}\mathbf{\Psi}^\top\mathbf{G}_e^\top\mathbf{O}_p^\top.$$

Inserting the identity $\mathbf{\Sigma}^{-1/2}\mathbf{\Sigma}\mathbf{\Sigma}^{-1/2} = \mathbf{I}$, we obtain

$$\mathbf{O}_p\mathbf{\Gamma}_\pi\mathbf{O}_p^\top = \mathbf{O}_p\mathbf{G}_e\mathbf{\Psi}\mathbf{\Sigma}^{-1/2}\mathbf{\Sigma}\mathbf{\Sigma}^{-1/2}\mathbf{\Psi}^\top\mathbf{G}_e^\top\mathbf{O}_p^\top = \mathbf{O}_p\mathbf{W}\mathbf{\Sigma}\mathbf{W}^\top\mathbf{O}_p^\top.$$

□

The posterior predictive covariance does not require then the extra computational effort of inverting an n-by-n matrix as it would in a typical approach. For the price of the computations to determine the IFP basis, we get both the mean and the covariance of the posterior predictive without significant additional computational cost.

The directions in parameter space that contribute to the variability of the posterior predictive are only those directions that are prediction-observable. Each direction's contribution to the variability depends on a tradeoff first between the prior and the likelihood (i.e., based on experimental observability) and second by prediction observability. All of these directions are contained in the IFP subspace, consistent with the prediction exactness property. If any of the directions were outside of the IFP subspace, underpredicting the variability in posterior predictive would be unavoidable.

Chapter 4

Application of linear goal-oriented inference to contaminant prediction

In this chapter we apply the algorithms of linear goal-oriented inference developed in the previous chapter to a model problem in contaminant identification and prediction. When the parameter is taken to be the initial contaminant release and the governing equations are those of advection-diffusion, the contaminant concentration field at a later time is a linear function of the initial condition. With linear experimental observations and linear prediction output quantities of interest, the assumptions of linear goal-oriented inference are satisfied.

The governing equations are established in section 4.1. In sections 4.2 and 4.3 we define the experimental and prediction observation operators. For the predictions we use a few different examples corresponding to some logical desirable output quantities of interest for this application. The discretization and numerical simulation is discussed in section 4.4. Finally we give discussion and results of the numerical experiments in section 4.5.

4.1 Governing equations

Let $\mathbf{z} = (z_1, z_2)$ be the spatial coordinates of a 2-D rectangular domain $\Omega = \{(z_1, z_2) \mid 0 \leq z_1 \leq 1, 0 \leq z_2 \leq 0.4\}$. Denote by $\partial\Omega$ the boundary of that domain. Let $c(\mathbf{z}, t) : \Omega \times \mathbb{R}_+ \to \mathbb{R}_+$ be the contaminant concentration at $\mathbf{z}$ and time t where $\mathbb{R}_+ = [0, \infty)$. We prescribe ambient wind velocity $\mathbf{u} = (1.5, 0.4)$ constant throughout the domain. Let the diffusivity $\kappa = 0.02$ also be constant. Given initial condition $c_0(\mathbf{z}) = c(\mathbf{z}, 0)$, the contaminant evolves in time according to the advection-diffusion equation

$$\begin{aligned} \frac{\partial c}{\partial t} &= \kappa \nabla^2 c - \mathbf{u} \cdot \nabla c, \quad && \mathbf{z} \in \Omega, t > 0, && (4.1) \\ \nabla c \cdot \mathbf{n} &= 0, && \mathbf{z} \in \partial\Omega, t > 0, && (4.2) \end{aligned}$$

where $\nabla = (\frac{\partial}{\partial z_1}, \frac{\partial}{\partial z_2})$ and $\mathbf{n}$ denotes the outward-pointing unit normal on each of the four segments of $\partial\Omega$.

4.2 Experimental process

The experimental outputs $\mathbf{y}_e(t) = (y_{e_1}(t), y_{e_2}(t), \ldots, y_{e_{n_s}}(t))$ at time t are given by localized integrals of the contaminant concentration, i.e.,

$$y_{e_i}(t) = \int_\Omega c(\mathbf{z}, t) \exp\left\{-\frac{1}{2\sigma_e^2}\|\mathbf{z} - \mathbf{z}_i\|^2\right\} d\mathbf{z}, \quad i = 1, 2, \ldots, n_s, \tag{4.3}$$

where $\mathbf{z}_i$ is the location of the ith sensor, $\sigma_e = 0.01$ is a measure of the sensing radius for all sensors, $\|\cdot\|$ represents the Euclidean norm in $\mathbb{R}^2$, and n_s is the number of sensors distributed in the domain. Contaminant concentration readings are available only at discrete times $t = t_0, t_1, \ldots, t_{n_r}$ where n_r is the number of readings. In what follows, we will denote the concatenation of experimental outputs as

$$\mathbf{y}_e = \left[\begin{array}{cccc} \mathbf{y}_e^\top(t_0) & \mathbf{y}_e^\top(t_1) & \cdots & \mathbf{y}_e^\top(t_{n_r}) \end{array} \right]^\top. \tag{4.4}$$

Then, $\mathbf{y}_e \in \mathbb{R}^r$ where $r = n_s n_r$. In our numerical experiments, we use eight sensors. The domain and sensor locations are shown in Figure 4-1. The sensors are placed in the domain with knowledge of the synthetic initial contaminant concentration but are not chosen with consideration for any of the outputs of interest. Both the IFP and traditional approaches utilize the same sensor configuration. We make measurements at time instants $t = \Delta t, 2\Delta t, \ldots, 30\Delta t$ where $\Delta t = 5 \times 10^{-3}$.

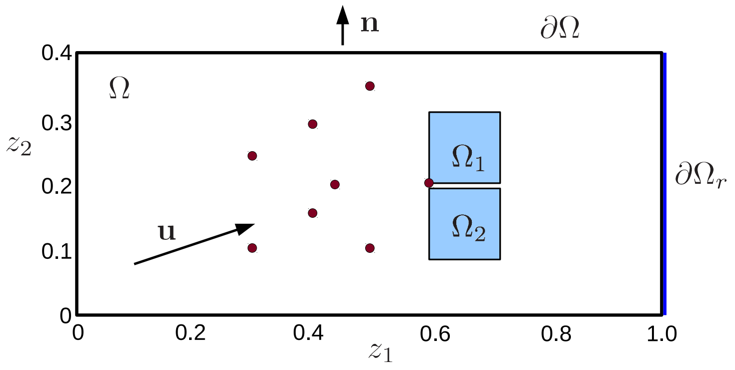


Figure 4-1: The domain and the eight sensor center locations.

4.3 Prediction process

For the numerical experiments we compare prediction outputs from the three traditional methods to their respective IFP implementations. We define three time-dependent prediction outputs of interest and two scalar prediction outputs.

Let $\partial\Omega_r = \{(z_1, z_2) \mid z_1 = 1,\, 0 < z_2 < 0.4\}$ denote the right boundary of the domain. One prediction output quantity of interest is the total contaminant propagating outward through this boundary as a function of time in the interval $60\Delta t \leq t \leq 70\Delta t$, i.e.,

$$y_{p_0}(t) = \int_{\partial\Omega_r} c(\mathbf{z}, t)\mathbf{u} \cdot \mathbf{n}_r \, dz_2, \quad 60\Delta t \leq t \leq 70\Delta t,$$

where $\mathbf{n}_r = (1, 0)$ is the outward-pointing unit normal for the right boundary.

We define a second prediction output of interest that is the total contaminant contained within a box on the interior of the domain. Let $\Omega_1 = \{(z_1, z_2) \mid 0.6023 \leq$

$z_1 \leq 0.6932,\, 0.2000 \leq z_2 \leq 0.2909\}$ and define

$$y_{p_1}(t) = \int_{\Omega_1} c(\mathbf{z}, t) dz_1\, dz_2, \quad 25\Delta t \leq t \leq 50\Delta t.$$

For the demonstration of the IFP methodology in the statistical setting, we will use two scalar prediction output quantities of interest. Let $\Omega_2 = \{(z_1, z_2) \mid 0.6023 \leq z_1 \leq 0.6932,\, 0.1000 \leq z_2 \leq 0.1909\}$; see Figure 4-1. Define

$$y_{p_2}(t) = \int_{\Omega_2} c(\mathbf{z}, t) dz_1\, dz_2, \quad 25\Delta t \leq t \leq 50\Delta t.$$

Our third and fourth prediction outputs of interest are the time-integrated quantities

$$y_{p_3} = \int_{t=25\Delta t}^{50\Delta t} y_{p_1}(t)\, dt \qquad \text{and} \qquad y_{p_4} = \int_{t=25\Delta t}^{50\Delta t} y_{p_2}(t)\, dt. \tag{4.5}$$

The IFP method utilizes the experimental data to infer those components of the parameter that are relevant for predicting the output quantities of interest. Our numerical experiments generate synthetic data by prescribing an initial condition that is a sum of Gaussian plumes, i.e.,

$$c_0(\mathbf{z}) = \sum_{i=1}^{5} \frac{1}{\alpha_i \sqrt{2\pi}} \exp\left\{ -\frac{1}{2\alpha_i^2} \|\mathbf{z} - \mathbf{z}_i\|^2 \right\}, \tag{4.6}$$

where the standard deviations α_i and centers $\mathbf{z}_i$, $i = 1, 2, \ldots, 5$ are given in Table 4.1.

The initial condition is pictured in Figure 4-2. For reference we present four snapshots of the contaminant concentration in the domain at times $t = 10\Delta t, 30\Delta t, 50\Delta t, 70\Delta t$ in Figure 4-3. The synthetic data is corrupted by noise for our experiments by adding random errors distributed normally with zero mean and variance $\sigma^2 = 0.01$.

The goal of our IFP method is not to obtain the true output of interest based on the true parameter but rather to match the prediction obtained by employing any of the traditional inference formulations discussed above. In other words, we are not proposing to improve accuracy of inference but rather to exploit final goals to make existing inference methods more efficient online and more transparent with respect

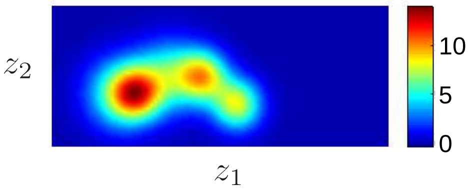
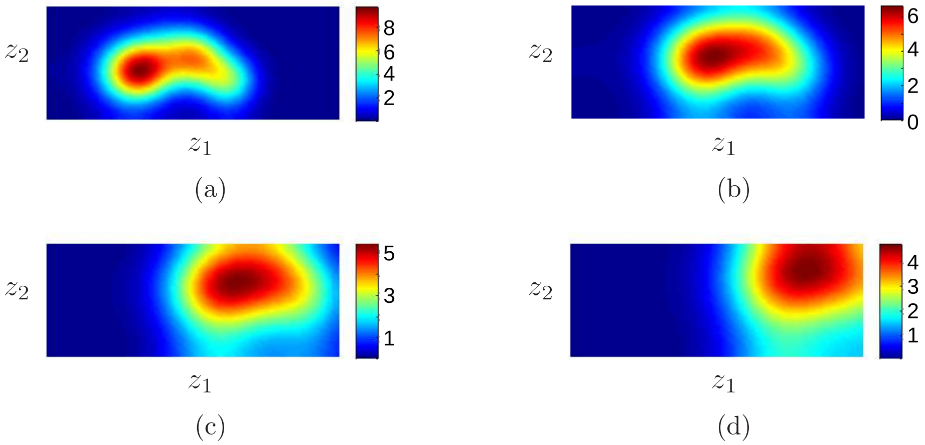


Figure 4-2: The initial contaminant concentration $c_0(\mathbf{z})$ used to generate synthetic data for the numerical experiments.

Table 4.1: Standard deviations and center locations for the five Gaussian plumes summed to form the initial condition (4.6) used to generate the synthetic data for the numerical experiments. The initial condition is pictured in Figure 4-2.

i	1	2	3	4	5
α_i	0.07	0.05	0.07	0.05	0.05
z_{1_i}	0.20	0.25	0.35	0.45	0.55
z_{2_i}	0.15	0.15	0.20	0.20	0.12

to injected prior information.



(a)

(b)

(c)

(d)

Figure 4-3: The evolution of the contaminant whose initial concentration is shown in Figure 4-2 at time steps (a) $t = 10\Delta t$, (b) $t = 30\Delta t$, (c) $t = 50\Delta t$, and (d) $t = 70\Delta t$.

4.4 Numerical simulation

For numerical simulation we discretize the continuous formulation (4.1)–(4.2) in space and time. We discretize in space by the finite element method (FEM) using a regular simplicial mesh with 44 and 88 elements each on the short and long boundary edges, respectively. The mesh has 7744 elements and 4005 nodes. We use a linear nodal basis to approximate the numerical solution and the parameter; i.e., we have $q = 4005$ parameter unknowns. The numerical instability due to the advection term is treated by a streamline upwind Petrov-Galerkin (SUPG) correction [15]. The semi-discrete equation is time-stepped by Crank-Nicolson leading to a system of the form (3.1) with $\mathbf{u}_k = \mathbf{0}\, \forall k$.

The integral computations for calculating the experimental outputs and the prediction output are also approximated by the discrete solution. For the experimental outputs, the integral is computed using a mass-matrix-weighted inner product between the rapidly decaying Gaussian sensor in the integrand of (4.3) and the solution vector $\mathbf{x}_k$ at time step k. The prediction output quantity of interest is estimated by using a midpoint rule in time and the linear nodal basis leads to a midpoint integration rule in space as well. In both experiment and prediction, the outputs are linear functions of the initial condition. For example, define $\mathbf{C}_e$ such that the experimental outputs (4.4) are given by

$$\mathbf{y}_e(k\Delta t) = \mathbf{C}_e \mathbf{x}_k, \qquad k = 1, 2, \ldots, 30.$$

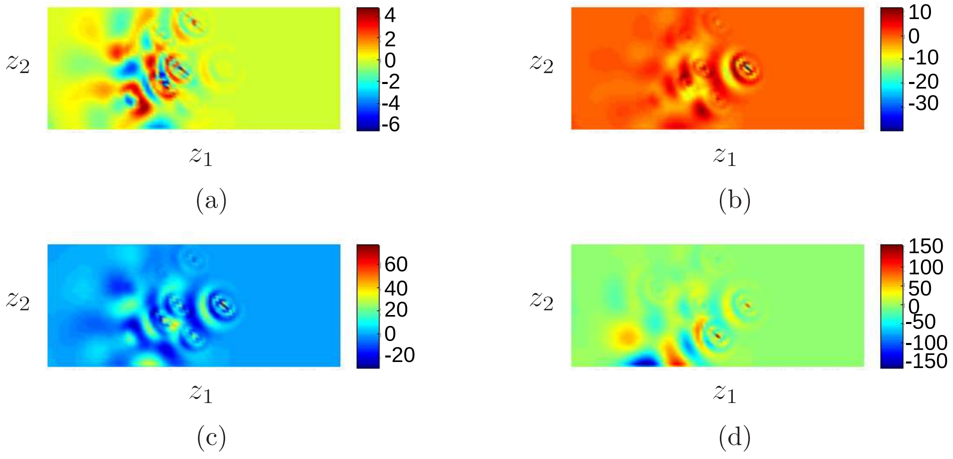
Let $\boldsymbol{\mu} = \mathbf{x}_0$, then $\mathbf{y}_e = \mathbf{O}_e \boldsymbol{\mu}$ where $\mathbf{O}_e = \begin{bmatrix} (\mathbf{C}_e\mathbf{A})^\top & (\mathbf{C}_e\mathbf{A}^2)^\top & \cdots & (\mathbf{C}_e\mathbf{A}^{30})^\top \end{bmatrix}^\top$. Similarly, define $\mathbf{C}_{p_0}$ such that $\mathbf{y}_{p_0}(k\Delta t) = \mathbf{C}_{p_0}\mathbf{x}_k$ for $k = 60, \ldots, 70$ then $\mathbf{y}_{p_0} = \mathbf{O}_p \boldsymbol{\mu}$ where $\mathbf{O}_p = \begin{bmatrix} (\mathbf{C}_{p_0}\mathbf{A}^{60})^\top & (\mathbf{C}_{p_0}\mathbf{A}^{61})^\top & \cdots & (\mathbf{C}_{p_0}\mathbf{A}^{70})^\top \end{bmatrix}^\top$. For the other outputs of interest, we just have to redefine $\mathbf{O}_p$ appropriately.

4.5 Results for numerical experiments

In this section we present results for the 2-D advection-diffusion application described in the preceding section. We will demonstrate the IFP methodology in each of the three inverse problem formulations described above: TSVD, Tikhonov-regularized, and Gaussian statistical. All of the problems are implemented in MATLAB and utilize the built-in LAPACK eigenvalue solver.

4.5.1 TSVD approach

The IFP method was applied in the context of the TSVD approach to the initial condition problem described above. In this case, we focus on the time-dependent output $y_{p_0}(t)$. Similar results are obtained for all of the outputs.



(a) (b) (c) (d)

Figure 4-4: The first four modes (a)–(d) of the IFP basis $\mathbf{W}$ for the TSVD approach.

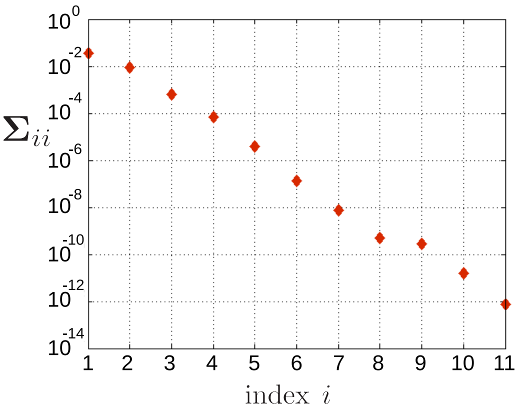


Figure 4-5: The singular values on the diagonal of $\mathbf{\Sigma}$ reflecting the joint measure of experiment and prediction observability for the TSVD approach.

Algorithm 2 is implemented to obtain the IFP basis $\mathbf{W} \in \mathbb{R}^{q \times s}$ whose first four modes are plotted in Figure 4-4. The high frequency characteristics are inherited from the eigenmodes $\mathbf{V}_e$. For this problem, there are 240 experimental outputs (8 concentration sensors over 30 time steps) and there are eleven prediction outputs (right side flux over 11 time steps). Although $\mathbf{H}_e$ mathematically has rank 240, the reduced eigendecomposition reveals that it can be approximated almost exactly (with respect to error in the Frobenius norm) by a rank-54 matrix; this is due to the numerical implementation and tolerance in the eigensolver. The singular values $\boldsymbol{\Sigma}_{ii}$ indicate that there is a subspace of dimension $s = 11$ for which there will be no information loss in the inference-to-prediction process; therefore, the IFP method yields a basis $\mathbf{W} \in \mathbb{R}^{4005\times 11}$. Decay of the singular values (see Figure 4-5) indicate that further truncation to fewer than eleven modes is possible; the IFP solution would then not result in exact predictions, but the error incurred by truncating the last three or four modes would be very small.

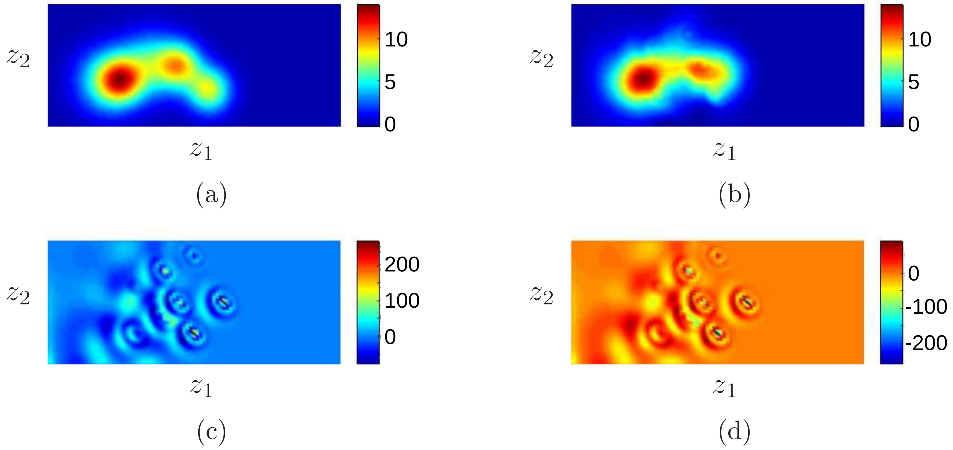


Figure 4-6: The (a) real initial condition, (b) TSVD estimate, (c) IFP estimate, and (d) error $\boldsymbol{\mu}_e = \boldsymbol{\mu}^{\text{TSVD}} - \boldsymbol{\mu}^{\text{IFP}}$. In Figure 4-7 we show the propagation of $\boldsymbol{\mu}_e$ to the output time steps.

We now turn to the results of the inversion. In Figure 4-6 we plot (a) the real initial condition, (b) the TSVD estimate, (c) the IFP estimate, and (d) the difference or error $\boldsymbol{\mu}_e = \boldsymbol{\mu}^{\text{TSVD}} - \boldsymbol{\mu}^{\text{IFP}}$. It is important to recall here that the IFP approach targets prediction outputs and is not designed to accurately infer the unknown parameter. Clearly the traditional inference method is more proficient at that.

What is relevant though is the propagation of the error $\boldsymbol{\mu}_e$ to the prediction output $y_{p_0}(t)$. If the IFP estimate $\boldsymbol{\mu}^{\text{IFP}}$ results in the same predictions as the TSVD estimate $\boldsymbol{\mu}^{\text{TSVD}}$ as the theory claims, then we expect that the error initial condition

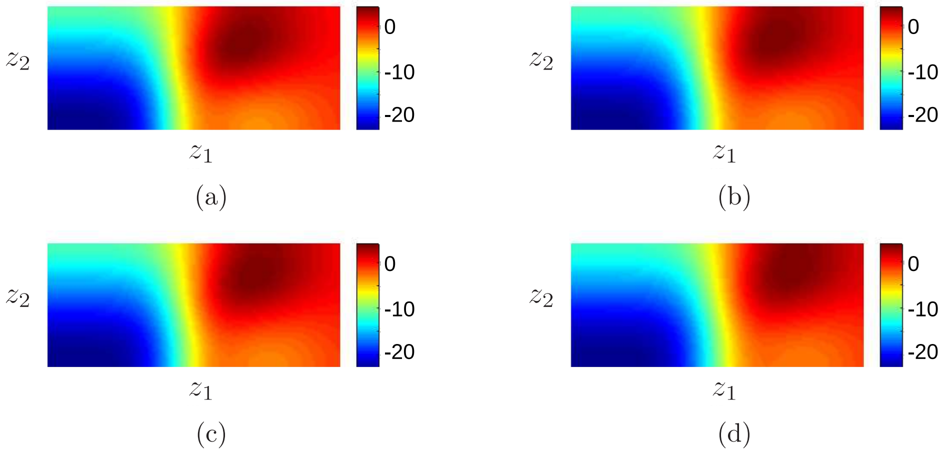


Figure 4-7: The propagation of $\boldsymbol{\mu}_e$ according to the advection-diffusion equation to the prediction output y_{p_0} at time steps (a) $t = 60\Delta t$, (b) $t = 64\Delta t$, (c) $t = 65\Delta t$, and (d) $t = 70\Delta t$. The integrated flux through the right boundary is negligible.

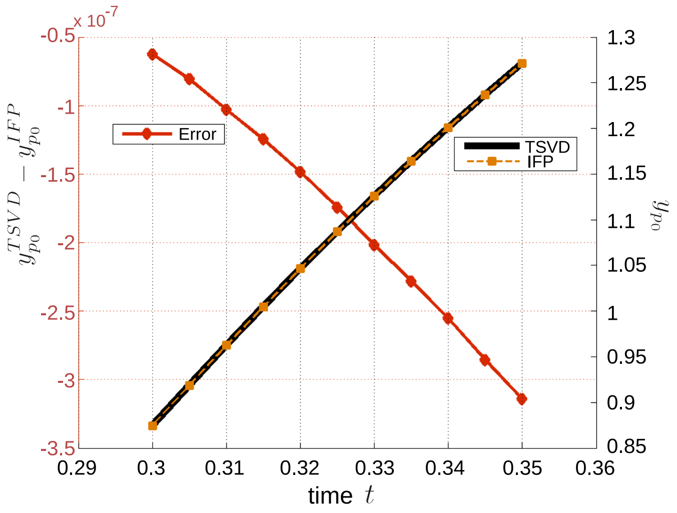


Figure 4-8: The (left ordinate axis) error between the prediction outputs from the TSVD and IFP approaches (red, diamonds) and the (right ordinate axis) predictions themselves based on TSVD (black, solid) and IFP (orange, dashed, squares) approaches.

$\boldsymbol{\mu}_e$ will lead to zero prediction. In Figure 4-7 we plot snapshots of the evolving error field beginning with initial condition $\boldsymbol{\mu}_e$ at four time steps within the prediction time region $t \in [60\Delta t, 70\Delta t]$. It can be seen that the error propagation leads to negligible flux through the right boundary, as the theory predicts.

In Figure 4-8 we plot the prediction outputs for both the TSVD and IFP approaches, as well as the error in the outputs. The prediction output curves are

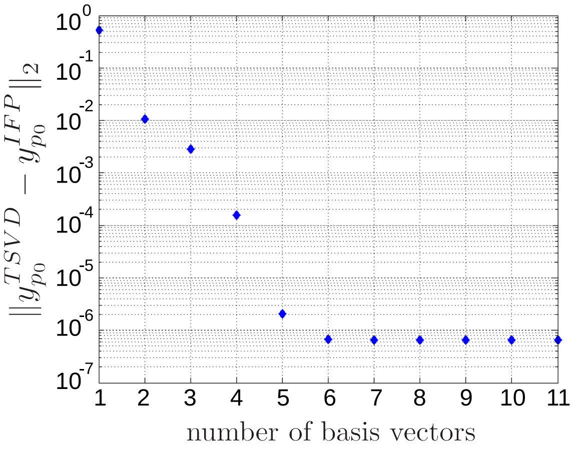


Figure 4-9: The error in prediction outputs $\|y_{p_0}^{\mathrm{TSVD}} - y_{p_0}^{\mathrm{IFP}}\|_2$ between the TSVD and IFP predictions vs. the number of IFP basis vectors included in $\mathbf{W}$.

directly on top of each other and the error is seven orders of magnitude less than the output predictions themselves. The error is not identically zero due to the numerical approximations, e.g., in the eigenvector solver, where tolerances are used.

Although our results above do not involve further truncation from the original IFP basis in $s = 11$ dimensions, we show in Figure 4-9 the error in prediction outputs as it varies with the number of basis vectors included in the IFP estimate. The error is significant if one includes just a few basis vectors, but as soon as six vectors are included the error drops to 10^{-6}.

In the next section, we demonstrate the approach for a Tikhonov-regularized inverse problem.

4.5.2 Tikhonov-regularized approach

In the Tikhonov-regularized approach, we define the matrix $\mathbf{R}$ implicitly by setting the diagonal matrix $\left(\mathbf{R}^\top\mathbf{R}\right)_{jj} = 0.1\lambda_{\min}(1 + (99j/4004))$ where $\lambda_{\min}$ is the smallest nonzero eigenvalue of the experiment observability gramian. This spreads the eigenvalues of $\mathbf{R}^\top\mathbf{R}$ evenly between approximately 0.0660 and 6.6028. We focus in this section on the output $y_{p_1}(t)$ defined above. Results are similar for the other outputs of interest.

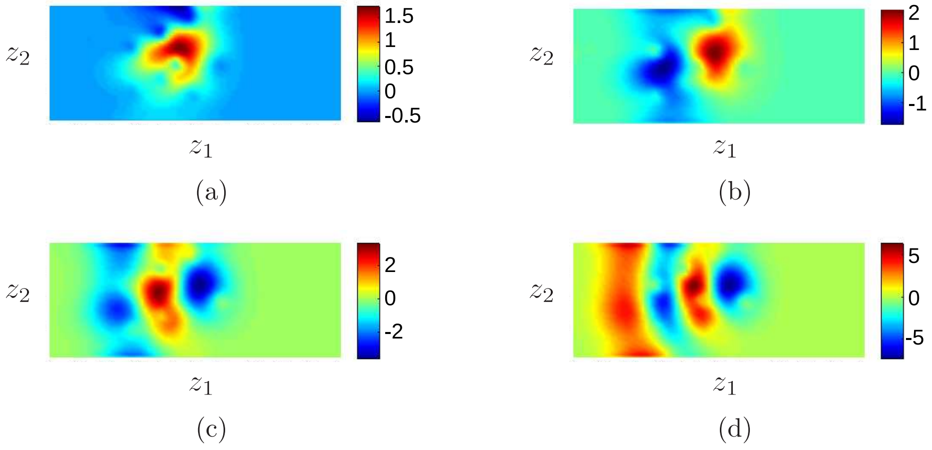


Figure 4-10: The first four modes (a)–(d) of the IFP basis $\mathbf{W}$ for the Tikhonov-regularized approach.

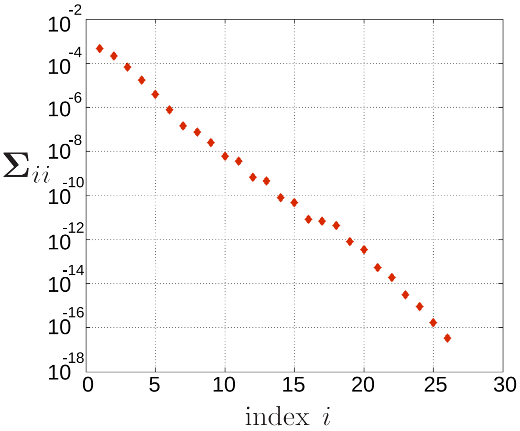


Figure 4-11: The singular values on the diagonal of $\mathbf{\Sigma}$ reflecting the joint measure of experiment and prediction observability for the Tikhonov-regularized inverse problem approach.

For this experiment, we find that $s = 26$ is the dimension of the IFP basis and here $r = q = 4005$ since the regularization fills the null space of the experiment observability gramian in the sense that $\mathbf{O}_e^\top \mathbf{O}_e + \mathbf{R}^\top \mathbf{R}$ is full rank. The first four basis modes are plotted in Figure 4-10 and the singular values are shown in Figure 4-11.

In Figure 4-12 we show the (a) real initial condition, (b) Tikhonov-regularized (TR) estimate, (c) IFP estimate, and the (d) error $\boldsymbol{\mu}_e = \boldsymbol{\mu}^{\text{TR}} - \boldsymbol{\mu}^{\text{IFP}}$. The evolution of the error $\boldsymbol{\mu}_e$ through the advection-diffusion equation is shown in Figure 4-13 for four time steps in the temporal range of the predictions $t \in [25\Delta t, 50\Delta t]$. The

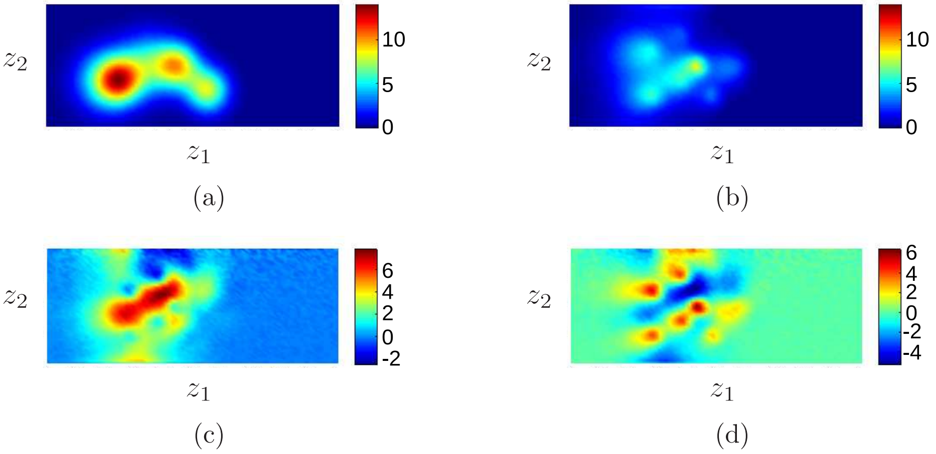


Figure 4-12: The (a) real initial condition, (b) Tikhonov-regularized (TR) estimate, (c) IFP estimate, and (d) error $\boldsymbol{\mu}_e = \boldsymbol{\mu}^{\mathrm{TR}} - \boldsymbol{\mu}^{\mathrm{IFP}}$. In Figure 4-13 we show the propagation of $\boldsymbol{\mu}_e$ to the output time steps.

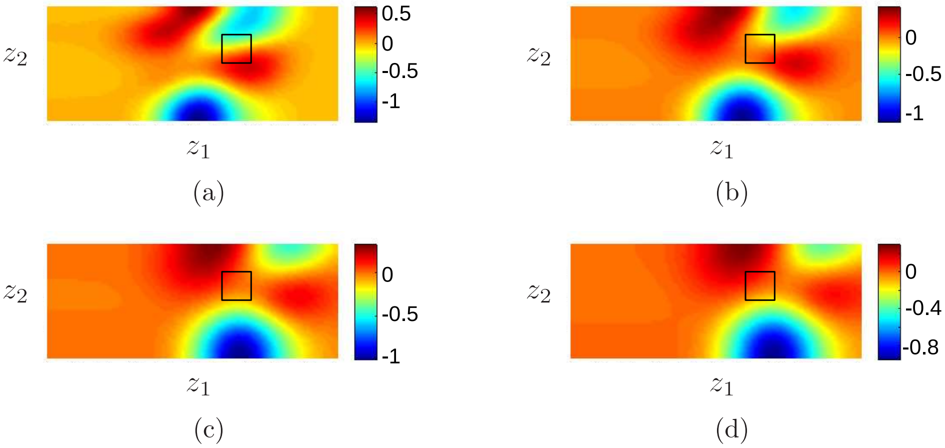


Figure 4-13: The propagation of $\boldsymbol{\mu}_e$ according to the advection-diffusion equation to the prediction output y_{p_1} at time steps (a) $t = 25\Delta t$, (b) $t = 35\Delta t$, (c) $t = 45\Delta t$, and (d) $t = 50\Delta t$. The average concentration inside of Ω_1 (box) is negligible for all time steps $t \in [25\Delta t, 50\Delta t]$.

average contaminant concentration in the subdomain Ω_1 is zero for all time steps in the prediction period. This is consistent with the proposition that the initial condition estimate based on TR and initial condition estimate based on IFP will result in the same predictions.

We show the error in predictions and the predictions themselves in Figure 4-14. The errors are again many orders of magnitude smaller than the predictions and the

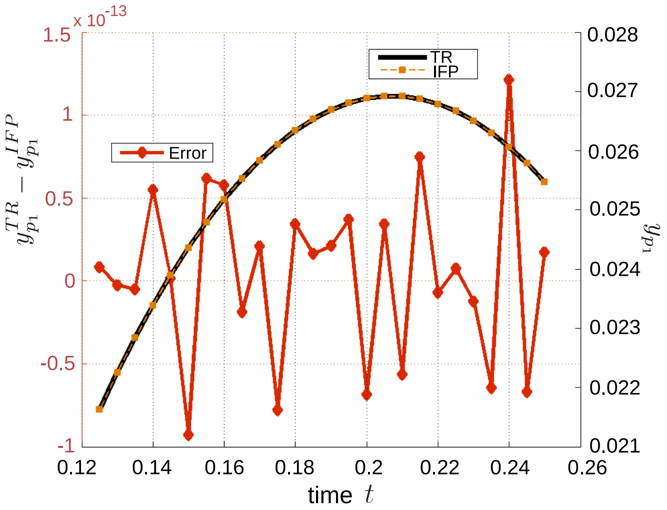


Figure 4-14: The (left ordinate axis) error between the prediction outputs from the TR and IFP approaches (red, diamonds) and the (right ordinate axis) predictions themselves based on TR (black, solid) and IFP (orange, dashed, squares) approaches.

predictions themselves lie directly on top of each other. This result is consistent with the theory presented in the preceding sections.

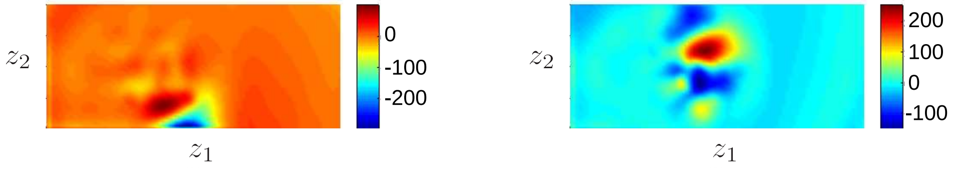
4.5.3 Gaussian statistical approach

For the statistical approach, we specify a prior distribution on the parameter $\boldsymbol{\mu}$. We use a multivariate normal prior with mean zero and covariance matrix $\boldsymbol{\Gamma}_0$ with (i, j)th element given by

$$\boldsymbol{\Gamma}_{0_{ij}} = a \exp\left\{ \frac{-\|\mathbf{z}_i - \mathbf{z}_j\|_2^2}{2b^2} \right\} + c\mathbf{I}, \quad 1 \le i, j, \le n$$

with constants $a = 0.001$, $b = 0.5$, and $c = 0.1$. We assume the sensors are corrupted by additive Gaussian noise that is i.i.d. each with zero mean and variance $\sigma^2 = 0.01$. For this numerical experiment we focus on the scalar outputs y_{p_3} and y_{p_4} defined in (4.5). We present results here for the posterior predictive mean and posterior predictive covariance; however, more attention is given to the covariance since the mean computation is analogous to the Tikhonov-regularized problem in the preceding section.

In this case again the prior distribution affects every mode of the parameter so that $r = q = 4005$. On the other hand, there are only two scalar outputs of interest so we find that the IFP basis has dimension $s = 2$. In Figure 4-15 we plot these two basis vectors. The singular values are $\mathbf{\Sigma}_{11} = 2.855 \times 10^{-4}$ and $\mathbf{\Sigma}_{22} = 1.390 \times 10^{-4}$.



(a) (b)

Figure 4-15: The only two modes (a) and (b) of the IFP basis $\mathbf{W}$ for the Gaussian statistical approach.

The results are presented in Table 4.2 and Figure 4-16. The estimated posterior predictive means and covariances are nearly identical having componentwise errors many orders of magnitude smaller than the values themselves. Once again, the numerical results reconfirm the theory. Inverting for just two modes of the parameter is sufficient to exactly obtain the posterior predictive distribution. In Figure 4-16 we plot equiprobability contours of the posterior predictive distribution from the (a) traditional and (b) IFP approaches.

Table 4.2: Means and covariances for the prediction outputs. In each cell, we list the result from the traditional approach, the result from the IFP approach, and the absolute value of the error. Equiprobable contours for the associated probability density functions are pictured in Figure 4-16.

			covariance	
		mean	y_{p_3}	y_{p_4}
y_{p_3}	Traditional Approach	5.1983E-1	1.9376E-8	2.0276E-9
	IFP	5.1983E-1	1.9376E-8	2.0276E-9
	Error	2.3309E-11	1.3235E-23	4.6736E-23
y_{p_4}	Traditional Approach	3.0017E-1	2.0276E-9	8.1433E-8
	IFP	3.0017E-1	2.0276E-9	8.1433E-8
	Error	1.0047E-10	4.6736E-23	7.9409E-23

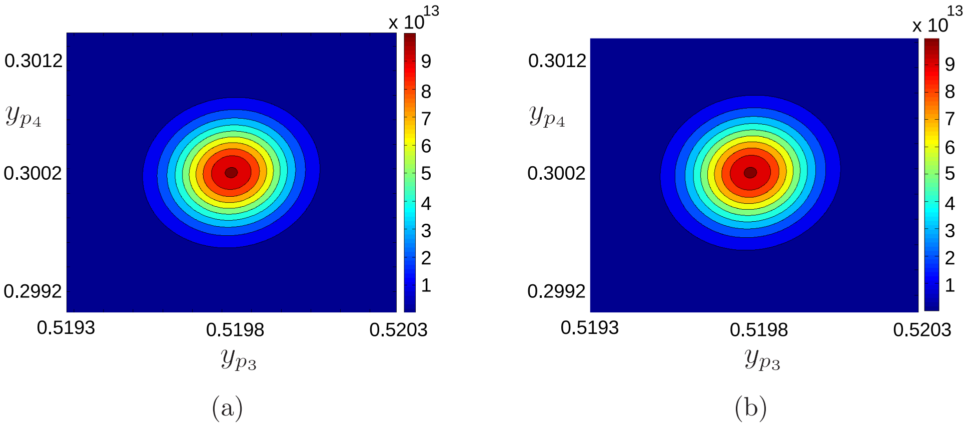


Figure 4-16: Contour plots of the joint probability density function over the outputs (y_{p_3}, y_{p_4}) for the (a) traditional approach and the (b) IFP approach. The means and covariance matrices corresponding to both approaches are also given in Table 4.2.

Chapter 5

Nonlinear goal-oriented inference in the statistical setting

Extending linear algorithms to the nonlinear setting is often achieved by repeated linearization of the nonlinearity. In the context of inference, however, this approach is unsuccessful. Critically, the point about which one desires to linearize the nonlinearity is the unknown parameter — that which itself is to be identified in the first place. Approaches based on iterating between linearizing the nonlinearity and solving the associated linear goal-oriented inference problem are possible but come with no guarantees that convergence will imply accuracy in predictions.

Although the theory of the linear case does not extend to nonlinear problems, the concept of driving the inference approach based on prediction requirements can be fruitfully extended. In this chapter we will focus on the statistical approach to inference for generally nonlinear systems. In this context, we present a new approach for circumventing the significant challenge of solving the nonlinear parameter inference problem in high-dimensions, particularly when the model equations are given by PDEs.

In section 5.1 we define the problem statement from the standpoint of generally nonlinear models for experimental and prediction processes and in the context of a statistical formulation. We describe the key challenges to parameter inference in this setting in section 5.2 as motivation for our approach. In section 5.3 we describe the details of the approach to goal-oriented inference in the nonlinear setting, where we focus on learning the joint density of experimental data and predictions in the form of a Gaussian mixture model (GMM). Posterior predictive distributions are obtained from the model by conditioning the GMM on the observed data. Finally, we present the kernel density estimation approach as a way of validating the GMM result.

5.1 Problem statement

We now present the formal problem statement for nonlinear goal-oriented inference in the statistical setting. We begin by writing the statement for the parameter inference problem and then propagating the posterior through to the prediction. This posterior predictive distribution is the target of our goal-oriented inference.

Let $\mathbf{f}_e(\boldsymbol{\mu}) : \mathbb{R}^q \to \mathbb{R}^r$ be a general nonlinear function representing the experimental process mapping parameter to expected observational data. The function will usually embed a PDE operator and an observation operator. In the carbon capture and storage example presented in the next chapter, $\mathbf{f}_e$ corresponds to the solution of the single-phase flow equations and observation of bottom hole pressure in some wells. We make observations $\mathbf{y}_d = \mathbf{f}_e(\boldsymbol{\mu}) + \boldsymbol{\epsilon}$ where $\boldsymbol{\epsilon}$ is assumed to be a multivariate normal noise such that $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \sigma_n^2 \mathbf{I})$ with noise variance σ_n^2.

Let $p(\boldsymbol{\mu})$ be the prior probability density function of the parameter. In Bayesian statistical approaches to parameter inference, the prior encompasses knowledge of the parameter before data are observed. There are several approaches for choosing the prior density and it has been the focus of significant controversy in the community [11, 19, 31]. Recall that our goal in this work is not to improve the existing parameter inference approaches but rather to reformulate them for the goal-oriented inference context. Therefore, we will simply take $p(\boldsymbol{\mu})$ as given, assuming that a routine to generate samples from the prior is available. For our numerical experiments, we will assume that $\boldsymbol{\mu}$ is a lognormal random process with exponential covariance kernel.

With the additive Gaussian noise model assumed above, our likelihood function $L(\boldsymbol{\mu}; \mathbf{y}_d)$ comes directly from the relation $\boldsymbol{\epsilon} = \mathbf{y}_d - \mathbf{f}_e(\boldsymbol{\mu}) \sim \mathcal{N}(\mathbf{0}, \sigma_n^2 \mathbf{I})$. In words, the likelihood represents the ratio (as a function of $\boldsymbol{\mu}$) of the posterior to the prior. For parameters that are more likely to have generated the observed data, the posterior undergoes a more significant update from the prior. Other likelihood models are possible, but this is a typical choice in the PDE-constrained statistical inference literature.

The posterior probability density function $\pi(\boldsymbol{\mu}|\mathbf{y}_d)$ can be readily expressed by Bayes's rule, i.e.,

$$\pi(\boldsymbol{\mu}|\mathbf{y}_d) \propto L(\boldsymbol{\mu};\mathbf{y}_d)p(\boldsymbol{\mu}). \tag{5.1}$$

The evidence that would appear in the denominator of Bayes's rule that scales to a proper density is unknown. Fortunately, it is not needed in order to draw samples from the posterior via Markov chain Monte Carlo (MCMC). Computation of the evidence itself is usually at least as computationally challenging as evaluating moments of the posterior by sampling since such a computation involves integration over the very high-dimensional parameter space.

The posterior (5.1) is the solution to the Bayesian statistical inference problem. It represents the new probabilistic description of the parameter after the prior is updated based on observed data. Unfortunately, the posterior is not immediately useful in its current form: since the likelihood embeds the nonlinear forward model $\mathbf{f}_e(\boldsymbol{\mu})$, the posterior is not written explicitly in terms of the parameter, and therefore, even the form of the posterior distribution is not readily apparent. What we can do, in theory, is generate samples from the posterior using MCMC. We discuss the challenges of MCMC, particularly in high-dimensional parameter spaces, in the next section.

We now introduce the prediction process $\mathbf{f}_p(\boldsymbol{\mu}) : \mathbb{R}^q \to \mathbb{R}^s$, a measurable function from parameters to prediction output quantity of interest. The function is often a composition of an output functional and a PDE operator. For the carbon capture and storage example, $\mathbf{f}_p$ represents the calculation of trapped carbon dioxide volume under a given injection scenario governed by a vertical equilibrium approximation of the two-phase flow equations.

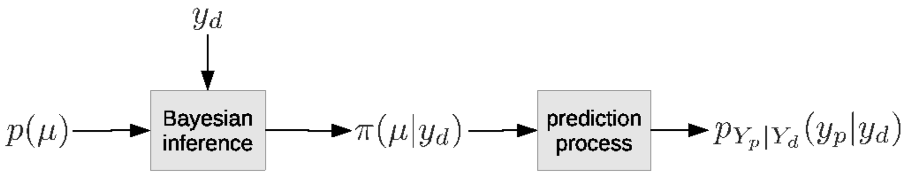


Figure 5-1: Block diagram of Bayesian statistical inference for prediction. The prior $p(\boldsymbol{\mu})$ is specified over the parameters representing the a priori relative likelihood of parameters. Data are observed and incorporated by Bayes's rule to yield the posterior $\pi(\boldsymbol{\mu}|\mathbf{y}_d)$. The posterior is then pushed forward through the prediction process to obtain the posterior predictive $p_{Y_p|Y_d}(y_p|\mathbf{y}_d)$.

The ultimate goal of our inference is to obtain the posterior predictive probability density function $p_{Y_p|Y_d}(\mathbf{y}_p|\mathbf{y}_d)$ which represents the push forward of the posterior measure $\pi(\boldsymbol{\mu}|\mathbf{y}_d)$ through the function $\mathbf{f}_p$. It is the representation of our estimate of the prediction given the choice of prior and accounting for the observed data. In the Bayesian paradigm, our estimate of the prediction is itself a random variable and therefore is characterized by a distribution representing the uncertainty in the estimate. If one could solve[1] the parameter inference problem by sampling from the posterior, one can obtain corresponding samples of the posterior predictive by passing each sample through the prediction process. The resulting samples can then be used to calculate moments of the posterior predictive, or since the prediction dimension is very low, one can even fit a density (using kernel density estimation, e.g.) to the provided samples to visualize the complete probability density function. A block diagram of the entire process is shown in Figure 5-1.

The goal-oriented inference method will obtain the resulting posterior predictive probability density function without inferring the parameter itself. Furthermore, the density will be obtained online in real-time; that is, when the data are observed, the posterior predictive can be obtained immediately without further sampling or expensive PDE solves. In the best case scenario using the traditional approach, the data are observed and then samples are generated from the posterior, if possible, where each proposal sample requires a full PDE solve (experimental process) and

[1] For the applications of interest, very high-dimensional parameter spaces and nonlinear PDE models render MCMC intractable in most cases.

each accepted sample requires another PDE solve (prediction process). This leads to a long delay between data observation and obtaining the posterior predictive. In most cases, the traditional approach of first performing statistical inference will be intractable, severing the path to obtain prediction estimates.

In the next section we discuss the challenges of parameter inference in more detail. As in the linear setting, we hope to convince the reader that extending the inference problem statement to include the ultimate goal of predictions actually serves to make the problem easier rather than more challenging, and in many cases, tractable where it otherwise would not be.

5.2 Motivation for goal-oriented inference in the nonlinear statistical setting

Statistical inference of distributed parameters in nonlinear problems is typically tackled using MCMC. In this section we highlight the daunting challenges facing this approach today. This serves as motivation for bypassing the inference of the parameter whenever it is not necessary. In the goal-oriented inference context, estimation of the parameter is just a means to obtain predictions. Therefore, we will circumvent the high-dimensional parameter inference and instead *infer* predictions directly from data.

MCMC [59] is a well-known approach for sampling from the posterior defined in the previous section. The goal is to implicitly construct a Markov chain whose invariant distribution is the posterior. When the chain is converged, samples from the chain are samples of the posterior. This is achieved by using an acceptance/rejection scheme based on proposing the next sample of the chain from a proposal distribution. The acceptance probability is determined such that detailed balance is satisfied and therefore the chain must converge to the posterior in the limit. There have been hundreds of variants of MCMC proposed in the literature; however, the method is still intractable in high-dimensional parameter spaces, particularly when the forward model has significant nonlinearity. One example is the carbon capture and storage application in the next chapter. In these cases, the traditional statistical inference of the parameter is intractable. Therefore, it would be impossible to obtain prediction estimates using that approach. With goal-oriented inference, on the other hand, we target estimation of low-dimensional prediction quantities of interest by essentially integrating out the parameter and focusing on the relationship between observed data and predictions. The intractability of parameter inference is circumvented and we obtain probabilistic prediction estimates from experimental data.

In addition to the challenges associated with sampling from the posterior, the computational cost makes infeasible the traditional approach of online parameter inference and subsequent forward uncertainty propagation to obtain prediction estimates. The posterior depends explicitly on the observed experimental data; therefore, this process must take place online. The evaluation of the acceptance ratio requires a PDE solve in the experimental process. The number of solves is guaranteed to be larger than the number of samples obtained (due to rejections), therefore the online cost of parameter inference alone will result in massive delays in prediction estimates. Furthermore, to obtain predictions will require another set of PDE solves of the prediction process, one for each of the parameter samples from the posterior. This renders real-time prediction infeasible.

5.3 Joint density estimation by Gaussian mixture model

The key to goal-oriented inference for nonlinear problems in the statistical setting is to exploit the low-dimensionality of experimental data and predictions. Although we did not entirely circumvent parameter inference in the linear case (we learned its restriction to the IFP subspace), here we propose to essentially integrate out the parameter itself, instead focusing entirely on the conditional relationship between predictions and experimental data.

Consider the concatenation $\mathbf{f}(\boldsymbol{\mu}) = [\mathbf{f}_e^\top(\boldsymbol{\mu}) + \boldsymbol{\epsilon}^\top, \mathbf{f}_p^\top(\boldsymbol{\mu})]^\top : \mathbb{R}^q \to \mathbb{R}^{r+s}$ of the data and prediction models. Let $p_{Y_d,Y_p}(\mathbf{y}_d, y_p)$[2] be the joint density of data and predictions given by the push forward of the prior through $\mathbf{f}(\boldsymbol{\mu})$. Our goal is to use samples to learn the joint density $p_{Y_d,Y_p}(\mathbf{y}_d, y_p)$ in the offline phase.

Once the joint density is learned, we move to the online phase where we conduct the experiment. Let $\tilde{\mathbf{y}}_d$ represent the data that are observed after the experiment is conducted, as opposed to $\mathbf{y}_d$, which represents the random variable associated to data that may be observed. When the real data $\tilde{\mathbf{y}}_d$ are observed, we can obtain the conditional density $p_{Y_p|Y_d}(y_p; \tilde{\mathbf{y}}_d)$ analytically from the learned joint density. That conditional density is precisely the posterior predictive that is the objective of our goal-oriented inference.

5.3.1 Sampling the parameter space

Similarly to many other statistical inference methods, we assume that the prior is designed so that the parameter can be efficiently sampled. Since our method is designed for, and our applications typically involve, distributed parameters, we describe the sampling procedure of the random field in this section. In particular, we describe the representation of the prior random field as a truncated Karhunen-Loeve (KL) expansion [48, 56]. For the carbon capture application undertaken in the next chapter, the permeability field is given by a lognormal random field. Our discussion here focuses on the representation of $\log \mu$.

We begin with the definition of a positive definite covariance function $C(\vec{x}_1, \vec{x}_2) : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ where d is the physical dimension of the domain and $\vec{x}_1$ and $\vec{x}_2$ are two points within the domain. The covariance function describes the correlation between the value of the parameter at $\vec{x}_1$ and the value of the parameter at $\vec{x}_2$. In practice, it is usually selected from a set of well-established choices. The parameters of the covariance function are selected based on the expected amplitude and correlation

[2] For the remainder of the discussion, we will assume $s = 1$; i.e., the prediction output quantity of interest is a scalar. The method is not restricted to such cases, but frequently it will be a scalar, and this makes the exposition and presentation cleaner.

length in the field, typically as given by experts (in our case, geologists). We aim to generate samples from a Gaussian random field $g(\vec{x};\xi)$ with zero mean and covariance function given by $C(\vec{x}_1,\vec{x}_2)$. The random field is given by $g(\vec{x};\xi)=\sum_{i=1}^{\infty}\sqrt{\beta_i}\xi_i v_i(\vec{x})$ where $(\beta_i, v_i(\vec{x}))$ are the eigenpairs of the covariance function and $\xi_i\sim\mathcal{N}(0,1)$ iid.

In practice, the domain is discretized and our parameter is represented as piecewise constant; we will refer to the discrete approximant as $\mathbf{g}(\xi)$. As is typical practice, we will calculate discrete eigenfunctions of the covariance function by forming the Gram matrix $\mathbf{K}\in\mathbb{R}^{n_{\text{el}}\times n_{\text{el}}}$ where n_{el} is the number of elements in our computational domain. Let $\vec{x}_i$ be the centroid of element i. Then the Gram matrix is given by

$$\mathbf{K}=\begin{bmatrix} C(\vec{x}_1,\vec{x}_1) & C(\vec{x}_1,\vec{x}_2) & \cdots & \cdots & C(\vec{x}_1,\vec{x}_{n_{\text{el}}}) \\ C(\vec{x}_2,\vec{x}_1) & C(\vec{x}_2,\vec{x}_2) & \cdots & & \vdots \\ \vdots & & \ddots & & \vdots \\ \vdots & & & \ddots & \\ C(\vec{x}_{n_{\text{el}}},\vec{x}_1) & \cdots & \cdots & & C(\vec{x}_{n_{\text{el}}},\vec{x}_{n_{\text{el}}}) \end{bmatrix}. \tag{5.2}$$

Once the Gram matrix is formed, we calculate the eigenvalue decomposition $\mathbf{K}=\mathbf{V}\mathbf{\Lambda}\mathbf{V}^{\top}$ where $\mathbf{V}=\begin{bmatrix}\mathbf{v}_1 & \mathbf{v}_2 & \cdots\mathbf{v}_{n_{\text{el}}}\end{bmatrix}$, $\mathbf{\Lambda}=\text{diag}(\lambda_1,\lambda_2,\ldots,\lambda_{n_{\text{el}}})$, $\mathbf{v}_i$ and λ_i are the ith eigenvector and eigenvalue with the ordering $\lambda_1\geq\lambda_2\geq\cdots\geq\lambda_{n_{\text{el}}}\geq 0$. The discretized random field is given by

$$\mathbf{g}(\xi)=\sum_{i=1}^{n_{\text{el}}}\xi_i\sqrt{\lambda_i}\mathbf{v}_i \tag{5.3}$$

still with $\xi_i\sim\mathcal{N}(0,1)$ iid.

Typically it is not necessary to retain all of the modes in the expansion. In practice we truncate the expansion after sufficient decay of the eigenvalues based on the ratio

$$\nu(m)=\frac{\sum_{i=1}^{m}\lambda_i}{\sum_{i=1}^{n_{\text{el}}}\lambda_i}. \tag{5.4}$$

In the application we will truncate at $m=\arg\min\nu(m)$ where $\nu(m)>0.98$, thereby retaining 98% of the *energy* in the representation.

A sample of the parameter will therefore be generated by sampling m iid standard normal random variables ξ_i and calculating the expansion

$$\boldsymbol{\mu} = \exp\left\{\sum_{i=1}^{m} \xi_i \sqrt{\lambda_i}\mathbf{v}_i\right\} \tag{5.5}$$

so that $\boldsymbol{\mu}$ is a lognormal random vector with zero mean and covariance $\mathbf{K}$, the discrete approximant of $\mu(\vec{x})$, the infinite-dimensional lognormal random field with zero mean function and covariance function $C(\vec{x}_1, \vec{x}_2)$.

5.3.2 Gaussian mixture models

For each of the parameter samples drawn from the prior, we simulate corresponding experimental data and prediction output. Let N_s be the total number of prior samples. For each sample we evaluate the experimental process and add simulated noise to generate synthetic measurements. Analogously, for each sample, we evaluate the prediction. As a result we obtain a set of ordered pairs $(\mathbf{y}_d^i, y_p^i)$ for $i = 1, \ldots, N_s$.[3] Ostensibly, these are samples from the joint density $p_{Y_d,Y_p}(\mathbf{y}_d, y_p)$ of experimental data and predictions.[4] From these data we propose to learn the joint density as a Gaussian Mixture Model (GMM). Then when data are observed, we simply condition the GMM on the given data to obtain the posterior predictive density as desired. In this section, we describe the construction of the GMM. Other density estimation techniques and other mixture models can also be used in this context. We select the GMM because of its convenient conditioning properties, allowing us to obtain real-time predictions.

A Gaussian mixture model [65] is a generative representation of a probability density function that generalizes the k-means clustering algorithm [73] to probabilistic

[3] As with any machine learning algorithm, one benefits greatly by perusing the data in advance, e.g., by plotting two-way marginals. Using the results, one should attempt to construct monotonic and differentiable transformations to make the data as normal as possible. The algorithm can then be applied to the transformed data, and the results can be transformed back appropriately. We will apply such transformations in the next chapter; here, we assume the data is already in a form amenable to our methods.

[4] It should be noted that this process is embarrassingly parallel; that is, the evaluation of experimental and prediction processes for each parameter sample can be performed completely independently. Therefore, optimal parallel scaling is possible.

clusters. Let X be a random variable distributed according to the density $p_X(x)$. Let $N(x;\mu,\Sigma)$ be the probability density function of a normal random variable with mean μ and covariance Σ. A Gaussian mixture model approximates

$$p_X(x) \approx \hat{p}_X(x) = \sum_{i=1}^{n_c} \alpha_i N(x;\mu_i,\Sigma_i), \qquad \sum_i \alpha_i = 1, \;\; \alpha_i \geq 0, \forall i, \tag{5.6}$$

where n_c is the number of components in the mixture model. The coefficients α_i are considered prior probabilities over the n_c clusters. One can therefore think of the mixture model in a generative manner. To draw a sample from X first select the cluster by sampling from the probability mass function corresponding to α_i. Then, given the mean μ_i and covariance Σ_i of that cluster, draw a sample from the corresponding multivariate normal.

For the moment consider the number of clusters to be fixed. The estimation problem then becomes one of determining the means and covariances of each of the components in the mixture. Typically this is achieved by choosing the parameters that maximize the likelihood of the data. Let x_i for $i = 1, \ldots, N_s$ be samples of X. Then, we have

$$\alpha_i, \mu_i, \Sigma_i = \arg\max \prod_{j=1}^{N_s} \hat{p}_X(x_j). \tag{5.7}$$

The component weights and component parameters are obtained by the well-known expectation-maximization (EM) algorithm [25]. We give a brief description of the algorithm here. Let $\theta_i = \{\mu_i, \Sigma_i\}$ be the unknown parameters of component i in the mixture model. Begin with an initial setting of the unknown parameters θ_i and weights α_i. Then calculate the membership weights

$$w_{ji} = \frac{\alpha_i N(x_j;\theta_i)}{\sum_{k=1}^{n_c} \alpha_k N(x_j;\theta_k)}, \qquad \forall i, \; \forall j \tag{5.8}$$

corresponding to the data point at x_j and component i. This corresponds to the E-step. For the M-step, we calculate the new component weights and component parameters

$$\alpha_i = \frac{1}{N_s} \sum_{j=1}^{N_s} w_{ji}, \quad \forall i, \tag{5.9}$$

$$\mu_i = \frac{1}{n_c} \sum_{j=1}^{N_s} w_{ji} x_j, \quad \forall i, \tag{5.10}$$

$$\Sigma_i = \frac{1}{n_c} \sum_{j=1}^{N_s} w_{ji} (x_j - \mu_i)(x_j - \mu_i)^\top, \quad \forall i, \tag{5.11}$$

in that order. The E and M steps are iterated until the likelihood is no longer changing from iteration to iteration, within a given tolerance.

5.3.3 Selection of the number of components

The GMM is an expressive statistical model, meaning that with sufficient components it could fit any set of data. In an extreme case we could fit a component to every data point. As with many learning algorithms, there is a danger of overfitting the data; in this case, the resulting model would not generalize well to unobserved samples. In practice, selection of the number of components in the mixture model can have a significant effect on the performance of the resulting statistical model. There have been proposed several methods to automatically determine the appropriate number of components.

One approach is to select an information criterion, which adds a regularization in the optimization (5.7). As a consequence, we maximize the likelihood of the observed data but with a penalty for including more components in the mixture model. Two common choices are the Bayesian information criterion (BIC) and Akaike information criterion (AIC) [1]. Let k be the number of parameters to be estimated in the statistical model. Then BIC adds a penalty of the form $k \ln N_s$ to the objective function. The number of data points are explicitly accounted for in this form. In contrast, the AIC uses a regularization that scales only with k. In what follows we use an approach based on BIC [29] since it accounts explicitly for the limited number of available data in our applications.

5.3.4 Evaluation of the posterior predictive online

Once the GMM is learned in the offline stage, i.e., before data are observed, we progress to the online stage of the process. The experiments are performed and data are collected. It remains only to condition the model on the observed data to obtain the posterior predictive density. Using the GMM, this process is straightforward as we now describe.

Let $\hat{p}_{Y_p,Y_d}(y_p, \mathbf{y}_d) = \sum_{k=1}^{n_c} \alpha_k N(y_p, \mathbf{y}_d; \mu_k, \Sigma_k)$ be the GMM we built in the offline stage. When we condition on observed data $\tilde{\mathbf{y}}_d$, we will obtain yet another GMM, this time over the prediction variables y_p only. Let $\hat{p}_{Y_p|Y_d}(y_p, y_d) = \sum_{k=1}^{n_c} \beta_k N(y_p; \tilde{\mathbf{y}}_d, \mu_{k|Y_d}, \Sigma_{k|Y_d})$ be the resulting GMM with positive component weights β_k that sum to unity, means $\mu_{k|Y_d}$, and covariances $\Sigma_{k|Y_d}$.

The new parameters are obtained as follows. Let

$$\mu_k = \begin{bmatrix} \bar{y}_{k_p} \\ \bar{\mathbf{y}}_{k_d} \end{bmatrix}, \qquad \Sigma_k = \begin{bmatrix} \Sigma_{k_{p,p}} & \Sigma_{k_{p,d}} \\ \Sigma_{k_{d,p}} & \Sigma_{k_{d,d}} \end{bmatrix} \tag{5.12}$$

be the decomposition of the component means and covariances into the parts corresponding to the prediction variables (subscript p) and data variables (subscript d). The parameters of the conditional GMM are then given by

$$\beta_k = \frac{\alpha_k (2\pi)^{-n_e/2} |\Sigma_{k_{d,d}}|^{-1/2} \exp\left\{-\frac{1}{2}(\tilde{\mathbf{y}}_d - \bar{\mathbf{y}}_{k_d})^\top \Sigma_{k_{d,d}}^{-1} (\tilde{\mathbf{y}}_d - \bar{\mathbf{y}}_{k_d})\right\}}{\sum_{m=1}^{n_c} \alpha_m (2\pi)^{-n_e/2} |\Sigma_{m_{d,d}}|^{-1/2} \exp\left\{-\frac{1}{2}(\tilde{\mathbf{y}}_d - \bar{\mathbf{y}}_{m_d})^\top \Sigma_{m_{d,d}}^{-1} (\tilde{\mathbf{y}}_d - \bar{\mathbf{y}}_{m_d})\right\}}, \tag{5.13}$$

$$\mu_{k|Y_d} = \bar{y}_{k_p} + \Sigma_{k_{p,d}} \Sigma_{k_{d,d}}^{-1} (\tilde{\mathbf{y}}_d - \bar{\mathbf{y}}_{k_d}), \tag{5.14}$$

$$\Sigma_{k|Y_d} = \Sigma_{k_{p,p}} - \Sigma_{k_{p,d}} \Sigma_{k_{d,d}}^{-1} \Sigma_{k_{d,p}}. \tag{5.15}$$

The conditional density can then be visualized, samples can be drawn, or moments computed.

5.3.5 Limitations

There are several limitations to this approach, although a few are shared by many other methods as well. Two important limitations are that of model misspecification and extension to higher dimensional data and prediction space, which we discuss in this section.

Model misspecification can take several forms in this setting. A prior for which the true unknown parameter has very low likelihood could lead to a data realization that is far away from simulated data samples. In this case, the density estimation scheme is required to extrapolate to the observed data as opposed to interpolating between them. This can lead to large errors in the posterior predictive density. However, it should be noted that it would be straightforward to detect when such a situation occurs, modify the prior in some way to account for it, and rebuild the GMM for the joint density.

Another form of model misspecification can be an unrealistic noise model in the likelihood function. In that case, the posterior predictive density is likely to reflect much lower variability in prediction than would be predicted with a more representative noise model. The same issues can also arise with observed data outside the regions adequately sampled in the offline stage, again leading to inaccuracy in predictions.

Lastly, model misspecification can also occur because of errors in the experimental or prediction processes themselves. Large errors in the posterior predictive can be manifested in this case as well.

It should be noted that many other techniques also suffer from model misspecification issues. In particular, a more traditional approach to the solution of the statistical inference problem would also be subject to errors in prior, noise model, and forward model prescription alike. These techniques would not, however, suffer from the issues of extrapolation that we have here, which, to some extent, exacerbate the situation. Instead, they work online and therefore can refer to the statistical model when the data are known.

Besides model misspecification, there are also impediments to extending the method to higher dimensional combined data and prediction spaces, i.e., the space where the mixture model is defined. The number of parameters to learn in the mixture model depends linearly on the number of components but scales roughly like the square of the number of dimensions (assuming anisotropic covariances are permitted). Density estimation therefore becomes more and more challenging as the dimensionality of the joint space increases. In applications of interest, however, it is likely that the prediction space will remain very low-dimensional, and that the dimensionality of the experimental data space may grow. For that context, it may be possible to break the algorithm into two steps: first, to identify some small number of important features in the data, and second, to learn the joint density between that feature space and the predictions. This will introduce another level of approximation but may be an adequate path forward for growing data spaces.

One important open question is the dependence of the accuracy of the method on the dimensionalities of the parameter, data, and prediction spaces as well as on the number of samples used to learn the mixture model. Intuitively, we expect the dimensionality of the parameter space to play an insignificant role since it is only the parameter's influence in the data and prediction spaces that are relevant for the joint density; that is, the information contained in the high-dimensional parameter space is collapsed into a much lower dimensional space. There is also an open question regarding the significance of the dimensionality of data and prediction spaces, separately, in the accuracy of the resulting posterior predictive density. One would like to believe that the dimensionality of the data space would play a less significant role since the final posterior predictive density is defined over just the prediction variables, but the conditional density itself (for unrealized data) is a function of both data and predictions. Therefore, it seems unavoidable that the dimensionality of the combined space will be the dominant consideration in such analyses. One way to alleviate this issue would be to combine our suggested approach with an online algorithm; however, the real-time capabilities would be necessarily sacrificed in doing so. In the next chapter we will explore numerically the effect of the total number of offline samples on the accuracy of the resulting posterior predictive density for several realizations of data.

5.4 Kernel conditional density estimation

In order to check the results from the GMM, we will compare to a kernel density estimate [70, 64] of the conditional density. For these purposes, we employ the Nadaraya-Watson conditional density estimator

$$\hat{p}_{Y_p|Y_d}(y_p; \mathbf{y}_d) = \frac{\sum_{i=1}^{N_s} \theta(y_p - y_p^i; h_p)\theta(\|\mathbf{y}_d - \mathbf{y}_d^i\|; h_d)}{\sum_{i=1}^{N_s} \theta(\|\mathbf{y}_d - \mathbf{y}_d^i\|; h_d)} \tag{5.16}$$

where $\theta(\Delta; h)$ is the Gaussian kernel function with length scale h [35]. We obtain h_p and h_d by minimizing a cross-validation estimate of the integrated squared error.

Define I to be the integrated squared error

$$I = \frac{1}{2} \int_{Y_p} \int_{Y_d} (\hat{p}_{Y_p|Y_d}(y_p; \mathbf{y}_d) - p_{Y_p|Y_d}(y_p; \mathbf{y}_d))^2 p_{Y_d}(\mathbf{y}_d)\, d\mathbf{y}_d\, dy_p. \tag{5.17}$$

Expanding out the terms, what remains is $I = \frac{1}{2}I_1 - I_2$ where

$$I_1 = \int_{Y_p} \int_{Y_d} \hat{p}_{Y_p|Y_d}(y_p; \mathbf{y}_d)^2 p_{Y_d}(\mathbf{y}_d)\, d\mathbf{y}_d\, dy_p, \tag{5.18}$$

$$I_2 = \int_{Y_p} \int_{Y_d} \hat{p}_{Y_p|Y_d}(y_p; \mathbf{y}_d) p_{Y_p,Y_d}(y_p, \mathbf{y}_d)\, d\mathbf{y}_d\, dy_p. \tag{5.19}$$

Let $J = \{1, \dots, N_s\}$ and define $\hat{p}_{J\setminus i}(y_p; \mathbf{y}_d)$ to be the Nadaraya-Watson conditional density estimator computed by excluding data point i. Then, we have leave-one-out cross validation Monte Carlo estimates

$$\hat{I}_1 = \frac{1}{N_s} \sum_{i=1}^{N_s} \int_{Y_p} \hat{p}_{J\setminus i}\left(y_p; \mathbf{y}_d^i\right)^2\, dy_p, \qquad \hat{I}_2 = \frac{1}{N_s} \sum_{i=1}^{N_s} \hat{p}_{J\setminus i}\left(y_p^i; \mathbf{y}_d^i\right). \tag{5.20}$$

Owing to the Gaussian kernel functions, we have the further simplification

$$\hat{I}_1 = \frac{1}{N_s} \sum_{i=1}^{N_s} \frac{\sum_{j\neq i} \sum_{k\neq i} \theta(\|\mathbf{y}_d^i - \mathbf{y}_d^j\|; h_d)\theta(\|\mathbf{y}_d^i - \mathbf{y}_d^k\|; h_d)\theta(y_p^k - y_p^j; \sqrt{2}h_p)}{\left(\sum_{j\neq i} \theta(\|\mathbf{y}_d^i - \mathbf{y}_d^j\|; h_d)\right)^2}. \tag{5.21}$$

We select h_p and h_d by minimizing $\hat{I} = \frac{1}{2}\hat{I}_1 - \hat{I}_2$.

We will check our solution using the GMM approach by evaluating the kernel density estimate at the observed experimental data $\tilde{\mathbf{y}}_d$ for a variety of samples of observed data. We will plot the densities for comparison.

5.5 Demonstration on model problems

We now demonstrate the above approaches on a few model problems before proceeding to the carbon capture and storage application in the next chapter. The goal here is to verify the approach on problems where the associated parameter inference problem is tractable. In this manner we can directly compare the goal-oriented approach to the traditional approach of first estimating the parameter and then propagating uncertainty forward to the prediction output quantity of interest.

5.5.1 Linear model

Consider linear models for both experiment and prediction, i.e., where we would normally apply the techniques from linear goal-oriented inference. Let the parameter $\mu \in \mathbb{R}^2$ have iid standard normal prior distributions. Our experimental outputs are defined by the function

$$\mathbf{y}_e = \begin{bmatrix} 2 & 3 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix}. \tag{5.22}$$

We make observations of data $\mathbf{y}_d = \mathbf{y}_e + \boldsymbol{\epsilon}$ where $\epsilon \sim \mathcal{N}(0, \sigma^2\mathbf{I})$ with $\sigma = 0.05$. The goal is to predict $y_p = \mu_1 + \mu_2$.

The solution to this problem is analytic owing to the normality and linearity. We compare the analytic solution to three approaches to making prediction. We use the GMM, the KDE, and lastly, we solve the inference problem using MCMC and propagate the resulting samples through the prediction, forming a kernel density estimate on those prediction samples. The results are shown in Figure 5-2.

For the KDE and GMM we use 50,000 samples of the joint density. We take 500,000 steps in the MCMC chain, retaining 9091 of them after discarding burn-in and removing correlated samples. All of the results are very accurate, with the KDE performing worst, but still with small error. The results from MCMC and the GMM both coincide with the truth solution, which we know analytically. In this case the GMM is able to fit the samples with just one mixture component. In general one may require many components and this affects both the computational complexity and the accuracy. As the number of samples of the joint density increases, the KDE will also converge to the truth. Here the GMM has the unfair advantage that it is able to exactly capture the true joint density since it is just a single multivariate Gaussian.

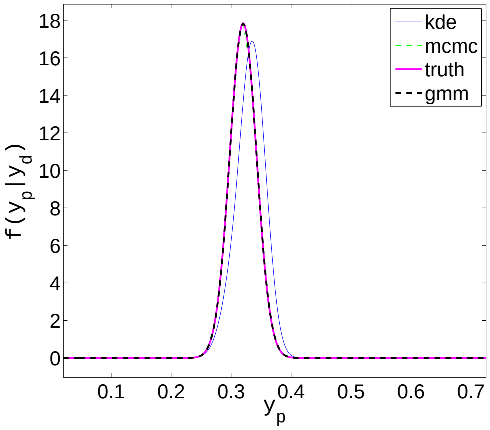


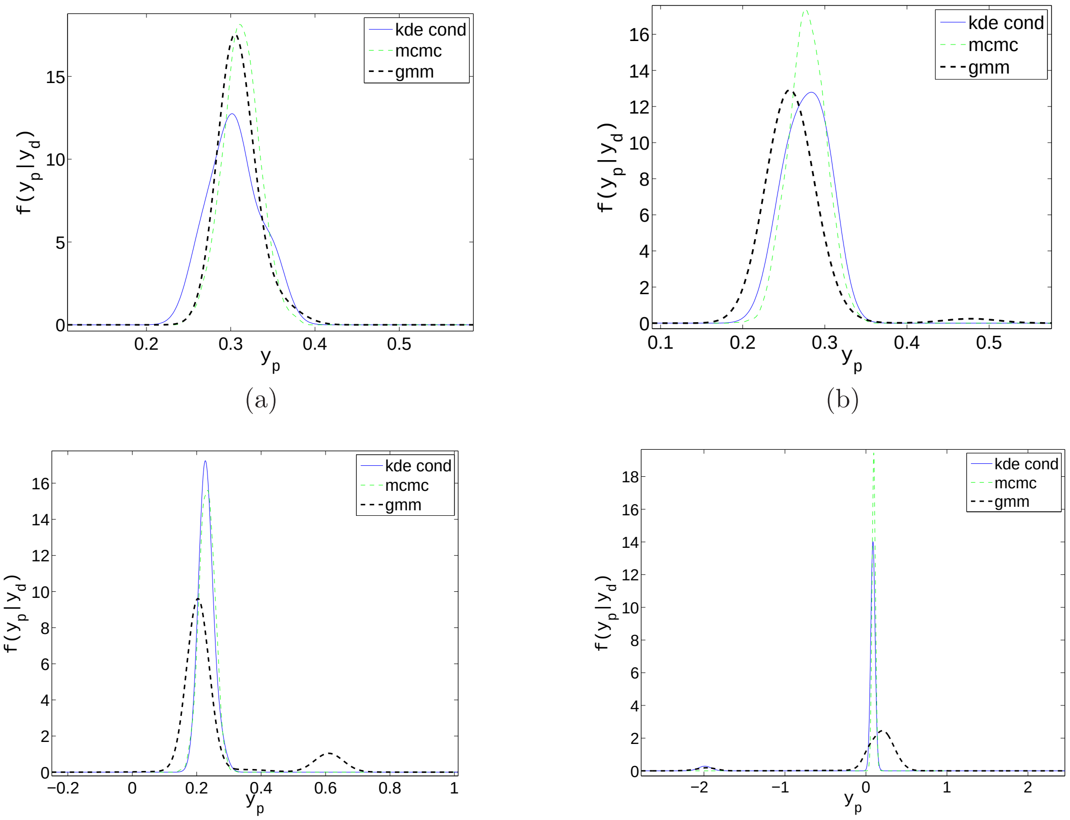
Figure 5-2: Results from the linear model for both experiment and predictions. The GMM and MCMC results both coincide with the truth result. The KDE is slightly off but still doing well.

5.5.2 Nonlinear model

We now introduce nonlinearity in the experimental process through an additional term whose effect we can dial in by adjusting a parameter λ, i.e.,

$$\mathbf{y}_e = \begin{bmatrix} 2 & 3 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix} + \lambda \begin{bmatrix} \mu_1^2 + \mu_2^2 \\ \mu_1^2 - \mu_2^2 \end{bmatrix}. \tag{5.23}$$

We use the same procedures for the approaches we used in the linear model example above. In this case, we do not have the analytic solution with which to compare. The results are plotted in Figure 5-3 for a variety of choices of the parameter λ. We consider the KDE results to be *truth* in this case due to its asymptotic convergence properties.



(a) (b) (c) (d)

Figure 5-3: Posterior predictive densities for choices of the nonlinear parameter (a) $\lambda = 0.1$, (b) $\lambda = 0.5$, (c) $\lambda = 1$, and (d) $\lambda = 3$.

As the parameter λ is increased, the experimental process becomes increasingly nonlinear (i.e., the nonlinear term becomes more pronounced). This is manifested in the results primarily by multimodality in the prediction. The MCMC approach fails to locate the other mode when it exists since it tends to get stuck in the first mode it finds.[5] The GMM, on the other hand, seems to identify a spurious extra mode in Figure 5-3(c). The GMM is susceptible to such behavior particularly where the observed data correspond to cuts through the joint density in regions with relatively few samples (i.e., low marginal likelihood of data). These regions are influenced by components of the mixture model that have been centered around more densely

[5] It should be noted that this drawback of the random walk Metropolis-Hastings form of MCMC has been addressed in the literature (see, e.g., [22]). We present the MCMC result here just as a reference; it's online cost is prohibitive in the goal-oriented inference context irrespective of algorithm.

packed samples in the joint density.

In the next chapter, we tackle the carbon capture and storage problem where both the experimental and prediction processes are nonlinear functions of the parameter. We will explore solutions based on KDE and GMM for obtaining posterior predictive densities. Unfortunately, the current MCMC technology is not feasible in the number of parameters we have in the application, so we cannot reasonably compare to such results.

Chapter 6

Application of nonlinear goal-oriented inference to carbon capture and storage

In this chapter we apply the nonlinear goal-oriented inference approach described in the previous chapter to a realistic application in carbon capture and storage. Sequestering carbon emissions in the subsurface is one method for curbing anthropogenic effects. Knowledge of the subsurface parameters (e.g., permeability and porosity) is essential to making accurate predictions of plume migration and trapping volumes. The subsurface parameters are field quantities and can only be measured indirectly by making sparse readings of pressures at boreholes for some experimental conditions. We utilize the goal-oriented inference approach to establish prediction estimates directly from observed data from the experiment.

We describe the carbon capture and storage application in section 6.1 and provide reference for the numerical implementation of the physics. In section 6.2 we define the geometry of a candidate aquifer and the associated computational domain. The random field defining the permeability is discussed in section 6.3. In sections 6.4 and 6.5 we establish the governing equations for the experiment and prediction processes, respectively. The experiments are governed by single-phase flow in porous media while the predictions depend on a vertical equilibrium approximation of the two-phase flow equations. Finally, in section 6.6 we give discussion of the numerical results employing the nonlinear goal-oriented inference procedure of the previous chapter.

6.1 Application description and numerical implementation

Supercritical carbon dioxide is typically injected in saline aquifers in the subsurface. The fluid is buoyant with respect to the resident brine; therefore, it floats and migrates along the caprock of the aquifer. Where the aquifer geometry is suitable, the fluid can be captured in pockets underneath the caprock. Remaining carbon dioxide continues to migrate. Of primary importance in such scenarios is the percentage of carbon dioxide effectively trapped in the injection and migration process over a given period of time. The dynamics of the plume depend heavily on the permeability in the aquifer, the target of parameter inference in the application. Determining the feasibility for injection of a candidate aquifer would typically involve performing experiments in the form of hydraulic interference tests to infer the permeability field. The estimate can then be used as input to an injection simulation to predict trapped volume percentage to evaluate different injection scenarios and ultimately to make decisions.

The computational tasks involving the geometry, numerical solution, and visualization of the experiment and prediction processes for the carbon capture and storage application are performed using SINTEF's Matlab Reservoir Simulation Toolbox (MRST) [52]. The governing equations for the experiment and prediction processes are discretized using the mimetic finite difference method. For the experiment process we solve the single-phase flow equations in 3-D. On the other hand, for the two-phase flow governing the migration of the carbon dioxide plume, we use the vertical equilibrium (VE) module that ships with MRST. MRST can be found at `www.sintef.no/Projectweb/MRST/` and is freely available under the GNU General Public License, well maintained, and updated with each new distribution of Matlab.

6.2 Computational domain

The computational domain is a hexahedral discretization of a synthetic, but realistic, saline aquifer. The domain is pictured in Figure 6-1. The aquifer occupies a one kilometer by one kilometer ground area and varies in thickness from approximately 30m to 80m as a function of x and y location. The aquifer's top surface contains both high and low frequency variations, and the aquifer itself has a 20m fault. We have made an effort to include the most challenging aspects of realistic candidate aquifers. The domain has 30,000 cells.

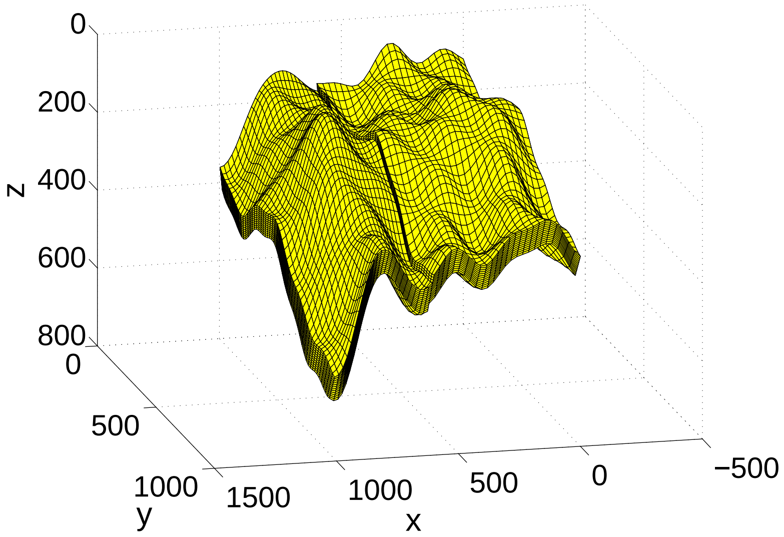


Figure 6-1: The computational domain representing the candidate aquifer.

We use the full 3-D domain for the experiment process where we solve the single-phase flow equations under given injection and production scenario. For the prediction process, however, we will enlist the vertical equilibrium approximation and use only the top grid, which contains 2500 quadrilateral cells. The top surface grid is pictured in Figure 6-2.

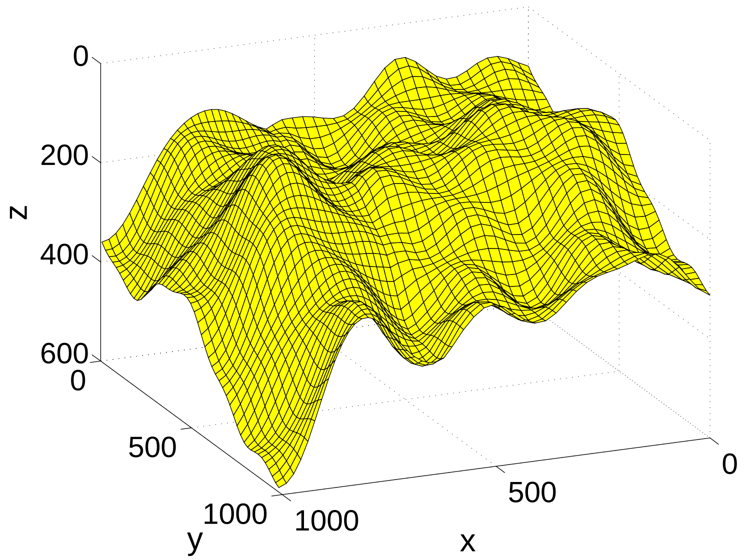


Figure 6-2: The computational domain for the prediction process; the top grid of the 3-D computational domain.

6.3 Permeability field

The parameter in this goal-oriented inference problem is the permeability field in the aquifer. Let $\underline{\mu}(x, y, z) : \mathbb{R}^3 \rightarrow \mathbb{R}^{3\times 3}$ be the permeability tensor field. We will assume that the tensor is anisotropic and has the form

$$\underline{\mu}(x, y, z) = \mu(x, y, z) \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0.1 \end{bmatrix} \tag{6.1}$$

where $\mu(x, y, z)$ is the parameter field.

We model the permeability as a lognormal random field with $\mu(x, y, z; \xi) = \exp g(x, y, z; \xi)$ where $g(x, y, z; \xi)$ is a Gaussian random field. We specify zero mean function and covariance function

$$C(\vec{x}_1, \vec{x}_2) = b \exp\left\{\frac{1}{L}\|\vec{x}_1 - \vec{x}_2\|\right\} \tag{6.2}$$

where $\vec{x} = (x, y, z)$, b is the amplitude, and L is the correlation length scale. In this application we use $L = 400$ and $b = 5$, which results in samples of permeability that vary by four to five orders of magnitude. This exponential kernel has algebraically diminishing eigenvalues (compared to the exponentially decreasing eigenvalues of the squared exponential covariance kernel, e.g.), which makes the moral dimension of the parameter space still very large.

As mentioned in the previous chapter, we discretize the random field and represent it as piecewise constant with the permeability varying from cell to cell. To sample approximately from the random field $g(x, y, z; \xi)$ we first construct the Gram matrix $\mathbf{K}$ by evaluating the covariance function at all pairs of centroids of the cells. We then perform the eigenvalue decomposition of the resulting matrix and retain modes corresponding to the highest 98% of the total energy as determined by the eigenvalues. The eigenvalue decay is shown in Figure 6-3. The first eight eigenvectors are pictured in Figure 6-4. Eight random samples of the log permeability are shown in Figure 6-5.

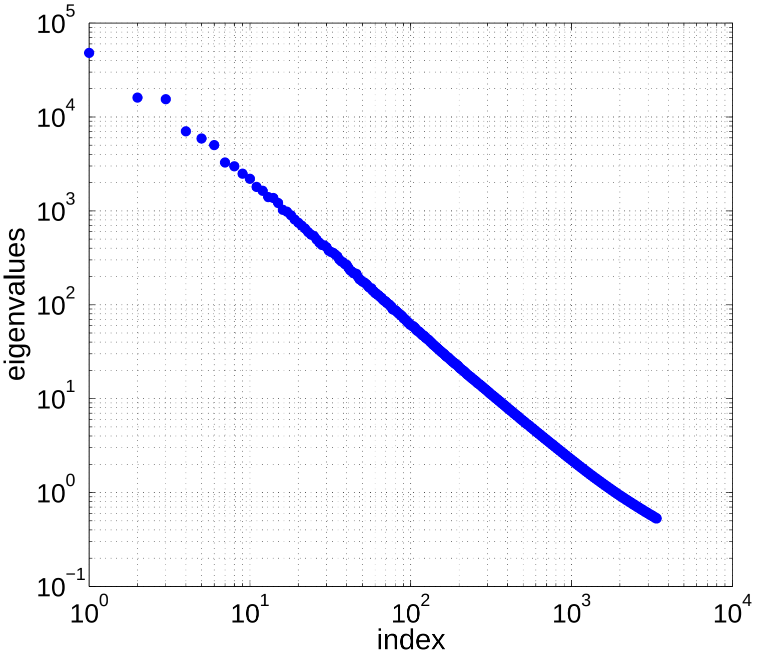
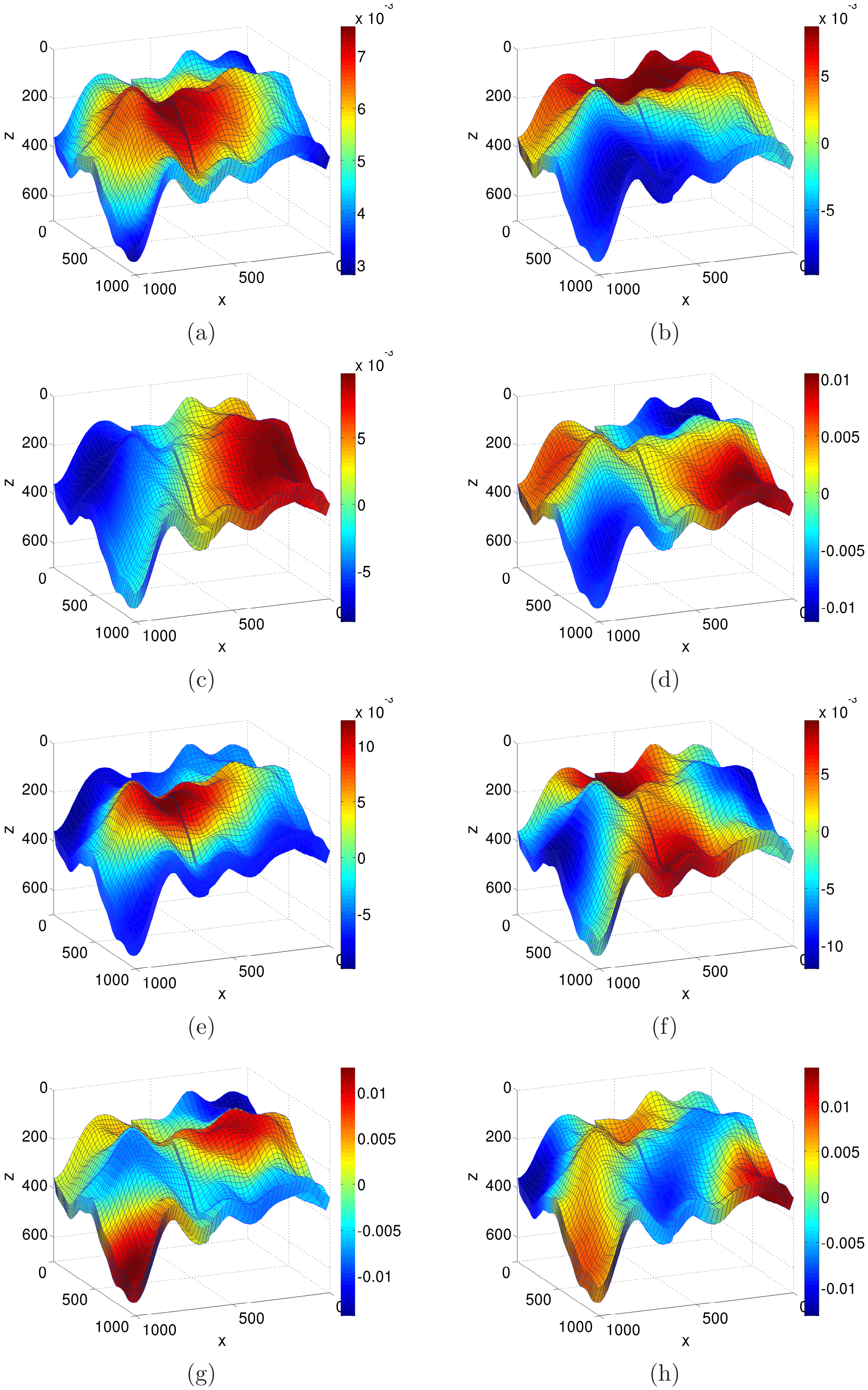
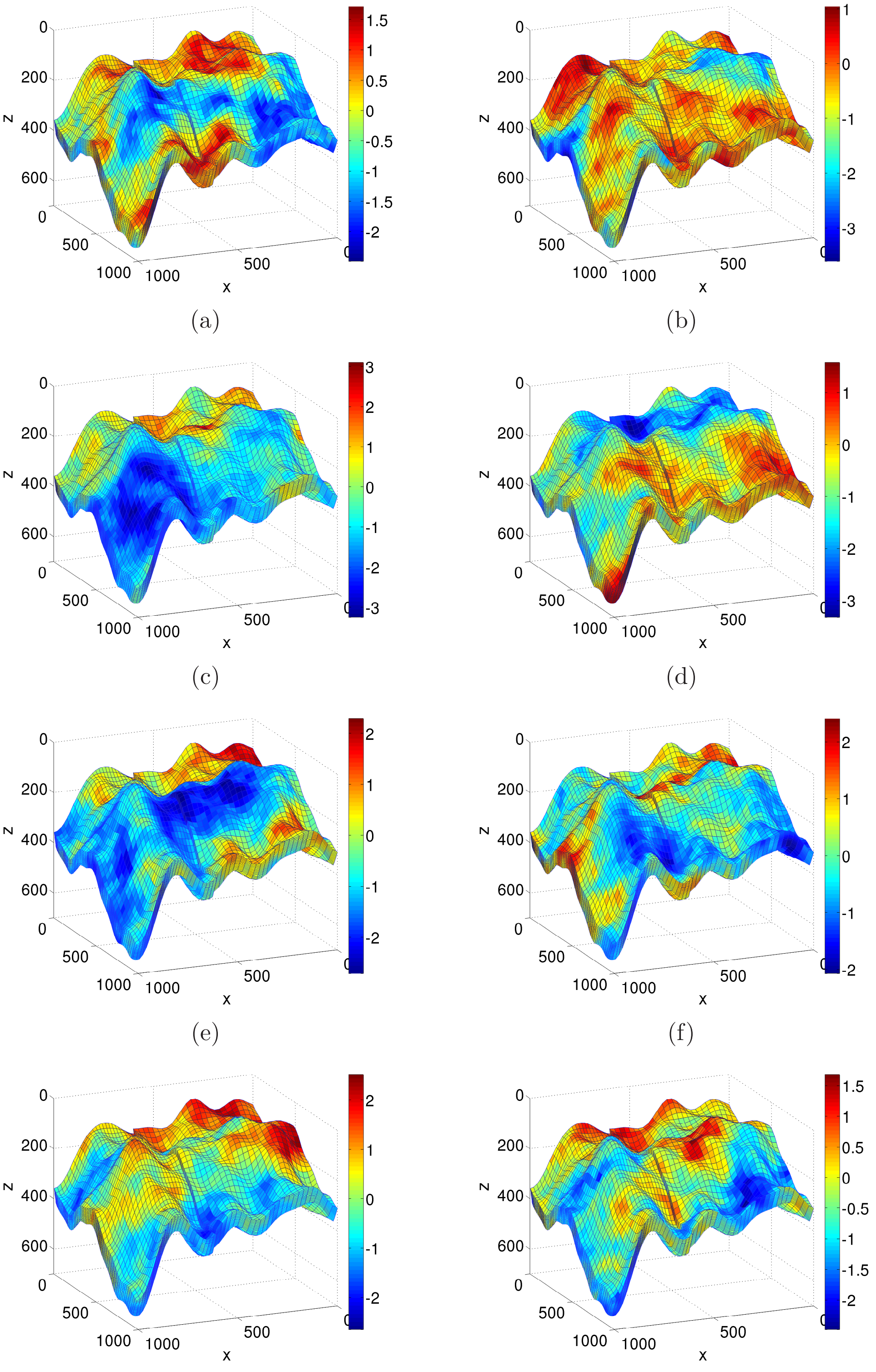


Figure 6-3: Eigenvalues of the Gram matrix with exponential covariance kernel. (Only the retained 3,383 eigenvalues pictured.)



(a) (b) (c) (d) (e) (f) (g) (h)

Figure 6-4: The first eight eigenmodes of the Gram matrix.



(a) (b) (c) (d) (e) (f) (g) (h)

Figure 6-5: Eight samples of the log permeability.

6.4 Experiment process

We now define the experiment process for the carbon capture and storage application. The experiment process is the steady state single-phase flow in the aquifer under given injection and production rates at five injection wells controlled by bottomhole pressure and three production wells controlled by rate. The outputs are the bottomhole pressures at each of the production wells.

Table 6.1: The positions and completions of the injection and production wells for the experimental process.

Label	Completion Top (x, y, z)	Completion Bottom (x, y, z)
I1	(62.352, 410.000, 135.592)	(63.178, 410.000, 150.430)
I2	(224.8412, 150.000, 194.957)	(223.716, 150.000, 209.392)
I3	(525.123, 729.999, 306.420)	(528.343, 729.998, 322.413)
I4	(784.941, 330.000, 164.954)	(785.166, 330.000, 180.140)
I5	(415.082, 70.001, 205.000)	(413.352, 70.002, 218.782)
P1	(396.613, 689.999, 201.606)	(399.512, 689.999, 215.890)
P2	(275.624, 50.001, 108.887)	(273.756, 50.001, 124.071)
P3	(587.946, 230.000, 225.926)	(587.421, 230.000, 241.260)

Let Ω be the computational domain with boundary faces $\delta\Omega$. We assume no flow boundary conditions (i.e., homogeneous Neumann at each boundary face). The pressure is fixed to 300 bar at the bottom of each of the injection wells. The production wells extract fluid at a rate of 3 m^3/day. The governing equation in the domain outside of the wells (which is solved using a Peaceman well model) is given by conservation of mass and Darcy flow, i.e.,

$$-\nabla \cdot (\mu \nabla u) = q, \qquad \vec{x} \in \Omega \tag{6.3}$$

where u is the global pressure and q corresponds to the sources/sinks at the injection and production wells.

For each sample of the permeability field, we solve (6.3) using MRST and extract the bottomhole pressure at the production wells. That is, we have $\mathbf{f}_e = [u(\vec{x}_1; \boldsymbol{\mu}), u(\vec{x}_2; \boldsymbol{\mu}), u(\vec{x}_3; \boldsymbol{\mu})]^\top$ where the production wells extend down to the cell whose centroids are at $\vec{x}_1$, $\vec{x}_2$, and $\vec{x}_3$, respectively. An example solution showing

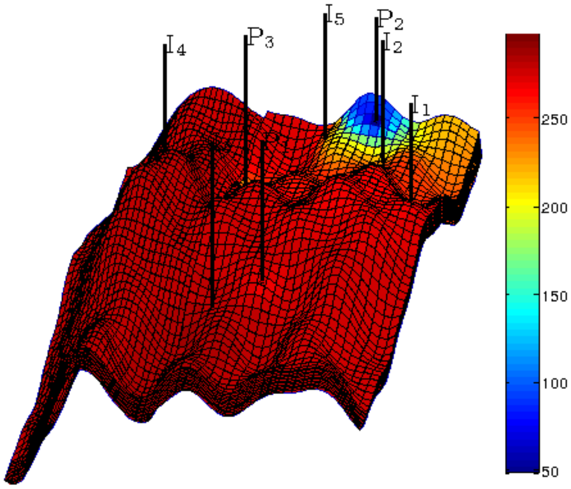
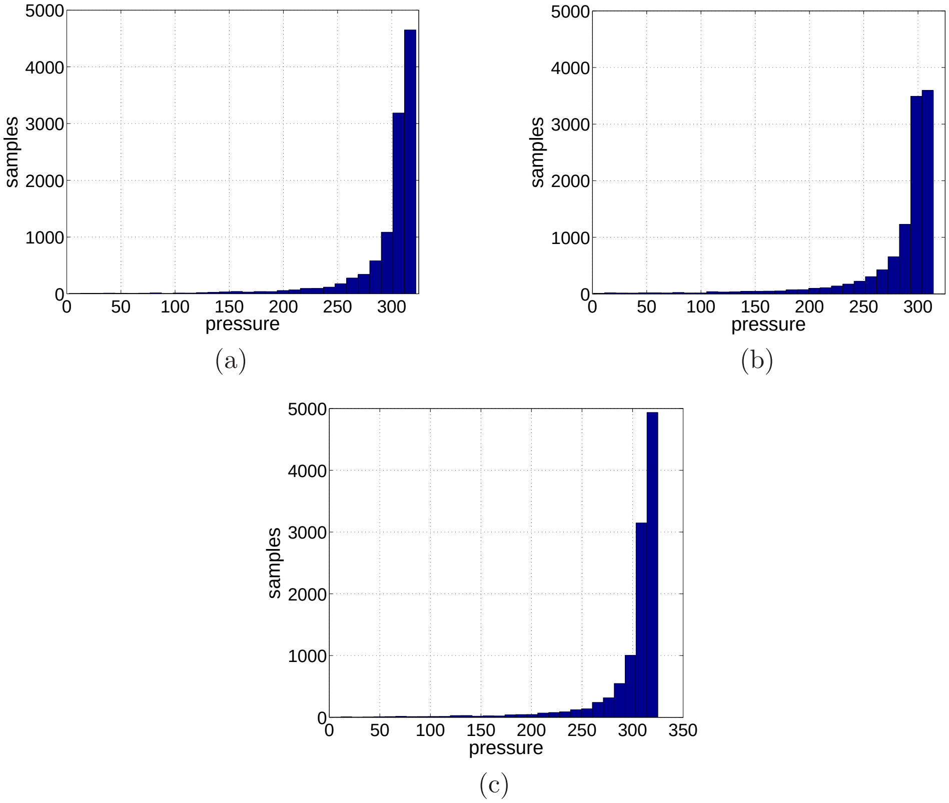


Figure 6-6: An example solution of the pressure (bar) in the experiment process. Five injection and three production wells pictured.

the injection and production wells is pictured in Figure 6-6. The locations of the injection and production wells are given in Table 6.1.

We simulate noise in the data by sampling from the additive Gaussian noise model with zero mean and standard deviation $\sigma_n = 2$ bar. Histograms of the marginal likelihood of the data for each experimental output are shown in Figure 6-7.



(a)

(b)

(c)

Figure 6-7: For 11,075 samples of the parameter, marginal likelihood of the data for the three experimental outputs, bottomhole pressure at production wells (a) P1, (b) P2, and (c) P3.

6.5 Prediction process

The prediction process is given by the two-phase flow of supercritical carbon dioxide and the resident brine in the aquifer. We make the common assumption that the flow is incompressible and that the two phases are immiscible. Furthermore, we will neglect capillary pressure in the model. For the development of the governing equations and the vertical equilibrium approximation, we follow [55]. For a more in-depth reference on the vertical equilibrium approximation and other modeling approaches, see [62].

Let φ be the porosity (assumed constant) in the aquifer, p_n and p_w be the pressures of the carbon dioxide (non-wetting) and brine (wetting), $\mathbf{v}_n$ and $\mathbf{v}_w$ the corresponding velocities, and S_n and S_w be the corresponding saturations. Mass conservation is then expressed by the PDEs

$$\varphi \frac{\partial S_i}{\partial t} + \nabla \cdot \mathbf{v}_i = q_i, \qquad \mathbf{v}_i = -\lambda_i(S)\mu(\nabla p_i - \rho_i \mathbf{g}), \quad i = n, w, \tag{6.4}$$

where ρ_i is the phase density, q_i is the phase source volume rate, $\lambda_i(S)$ is the phase mobility as a function of the saturation, and $\mathbf{g}$ is the gravitational vector. Define now a global pressure p and total velocity $\mathbf{v}$ to obtain

$$\varphi \frac{\partial S}{\partial t} + \nabla \cdot f(S)(\mathbf{v} + \lambda_w(S)\mu\Delta_\rho \mathbf{g}) = q_n, \tag{6.5}$$

$$\mathbf{v} = -\mu\lambda_t(S)(\nabla p - (f(S)\rho_n + (1 - f(S))\rho_w)\mathbf{g}), \tag{6.6}$$

$$\nabla \cdot \mathbf{v} = q_t, \tag{6.7}$$

where $\lambda_t = \lambda_n + \lambda_w$ is the total mobility, $f = \lambda_n/\lambda_t$ is the fractional mobility, $\Delta_\rho = \rho_n - \rho_w$ is the density difference of the two phases, and $q_t = q_n + q_w$ is the total volume rate of source.

Let H be the total height of the aquifer and h be the height of the carbon dioxide plume. Define $s = h/H$ to be the relative height of the CO2 plume as a function of position $\vec{x} = (x, y)$ and time t. If we vertically average (6.5), we obtain

$$\varphi H(\vec{x})\frac{\partial s}{\partial t} + \nabla_{||} \cdot \Big(\tilde{f}(s, \vec{x})\mathbf{v}_{ve} + \tilde{f}_g(s, \vec{x})(\mathbf{g}_{||}(\vec{x}) + \nabla p_c(s, \vec{x})\Big) = q_n(x, y), \tag{6.8}$$

$$\nabla_{||} \cdot \mathbf{v}_{ve} = q_t(\vec{x}), \tag{6.9}$$

$$\mathbf{v}_{ve} = -\tilde{\lambda}_t(s, \vec{x}) \left(\nabla_{||}p_t - (\tilde{f}(s, \vec{x})\rho_n + (1 - \tilde{f}(s, \vec{x}))\rho_w)\mathbf{g}_{||}(\vec{x}) + \frac{\tilde{\lambda}_w}{\tilde{\lambda}_t}\nabla_{||}p_c(s, \vec{x})\right), \tag{6.10}$$

where the notation $\mathbf{a}_{||}$ indicates the component of the vector $\mathbf{a}$ parallel to the top surface of the aquifer, $p_t(\vec{x})$ is the global pressure at the top surface. Since we disregard capillary forces, we have $p_c(s, \vec{x}) = H(\vec{x})g_{\perp}\Delta_{\rho}s$ where $g_{\perp}$ is the component of the gravity vector perpendicular to the top surface.

Heterogeneities in the medium are preserved in the vertical equilibrium approximation by defining modified mobilities and fractional flow functions

$$\tilde{\lambda}_n = \int_0^{sH(\vec{x})} \frac{k_n}{\nu_n}\mu dz, \quad \tilde{\lambda}_w = \int_{sH(\vec{x})}^{H(\vec{x})} \frac{k_w}{\nu_w}\mu dz, \quad \tilde{f} = \frac{\tilde{\lambda}_n}{\tilde{\lambda}_n + \tilde{\lambda}_w}, \quad \tilde{f}_g = \tilde{\lambda}_w\tilde{f}, \tag{6.11}$$

where k_i and ν_i are the relative permeability and viscosity, respectively, of phase i. The evaluation of the relative permeabilities in (6.11) depends on whether the reservoir is locally undergoing drainage or imbibition. Let s_i^{res} be the residual saturation of phase i. When the aquifer is undergoing imbibition, i.e. $s_{max} > s$ where s_{max} is the maximum historical saturation, then the relative permeabilities are evaluated at $1 - s_w^{res}$ for k_n and $1 - s_n^{res}$ for k_w; otherwise, they are evaluated at unit saturation.

We simulate the injection of supercritical carbon dioxide at one well at a rate of 500m^3/day for 50 years followed by the migration that takes place over the following 450 years. The resulting plume is pictured in Figure 6-8. The prediction quantity of interest is the total volume of trapped carbon dioxide. This corresponds to the portion of the fluid that has been successfully sequestered under the caprock and is no longer mobile, thereby no longer presenting a risk of leakage through faults or improperly sealed wells in other parts of the subsurface. A histogram of the marginal likelihood of the predictions is shown in Figure 6-9.

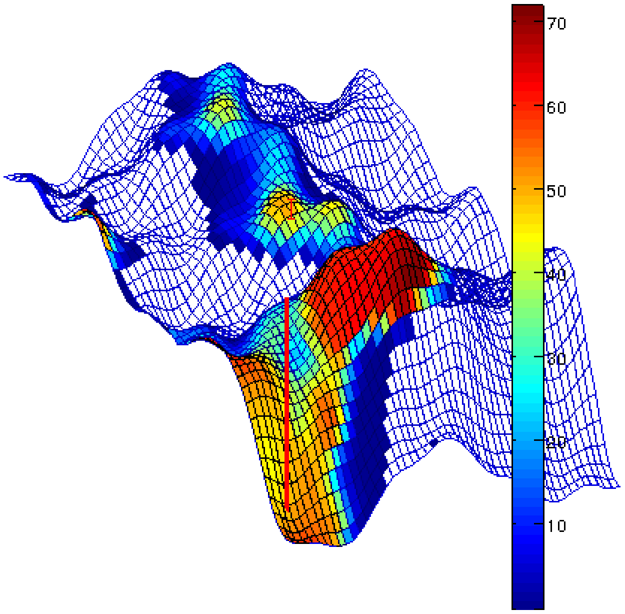


Figure 6-8: The height (m) of the CO2 plume after 50 years of injection at 500m^3/day followed by 450 years of migration.

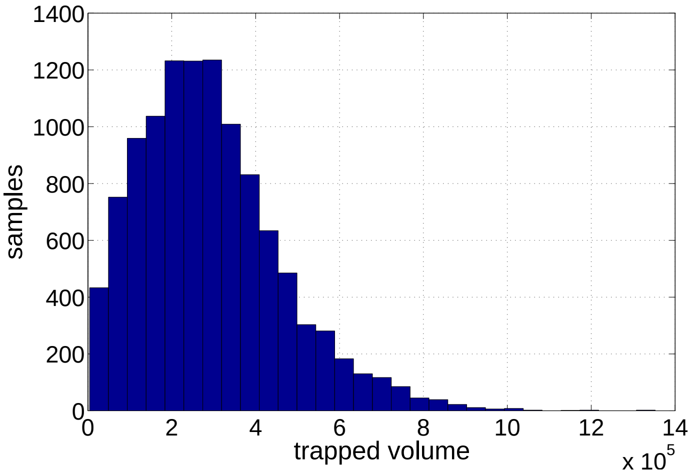


Figure 6-9: Marginal likelihood of the prediction of trapped volume for 11,075 samples of the parameter.

6.6 Discussion and numerical results

We now undertake the task of learning the conditional density of prediction given experimental data and performing numerical experiments to validate our approach. We begin first by visualizing and then transforming the data with monotonic and differentiable functions in section 6.6.1. In section 6.6.2 we evaluate the performance of our approach by simulating true experimental data and validating our approach with a kernel density estimate of the posterior predictive. Finally, in section 6.6.3, we study numerically the effect on the results of the number of parameter samples used to build the GMM in the offline stage.

6.6.1 Transforming the data

We begin by inspecting the data further, a recommended first step before employing any machine learning algorithm. Any additional insight one can gain from perusing the data can help to inform model building or algorithmic choices. The raw data is shown in Figure 6-10 in the form of pairwise marginals.

From the figure, it is clear that each component of the experimental data and the prediction would benefit from logarithmic transformation. Therefore, we perform the transformations

$$\mathbf{y}_d \leftarrow \ln(330 - \mathbf{y}_d), \qquad y_p \leftarrow \ln y_p, \tag{6.12}$$

which are both differentiable and monotonic. The barrier value at 330 bar was determined based upon inspection of the raw data.

6.6.2 Numerical results

Using the transformed data shown above, we learn a GMM for the joint density using the Matlab code developed in [28, 29]. We choose a maximum of 30 components and the code automatically chooses the number of components to balance the maximization of likelihood of data and the Bayesian information criterion (BIC) [21]. Given the 11,075 samples of the joint density we use, the algorithm settles on a GMM for

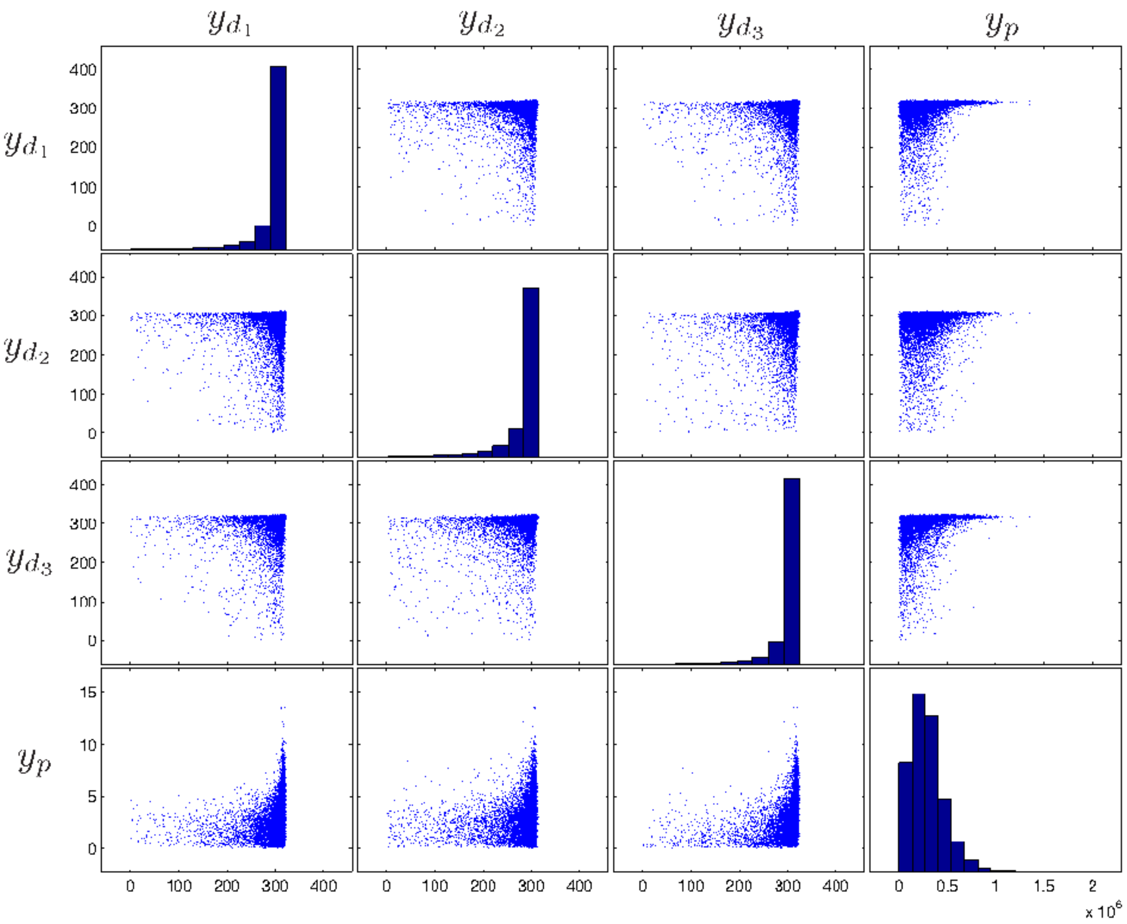


Figure 6-10: Pairwise marginal data and histograms. The first three rows (columns) correspond to the experimental data, and the last row (column) corresponds to the prediction output. It is clear that all components would benefit from certain logarithmic transformations.

the joint density that has 15 components. As a verification tool, we use a kernel density estimate (KDE) as described in the previous chapter. The KDE is obtained using the Kernel Density Estimation Toolbox for MATLAB [43].

For the numerical experiments, we select a permeability field at random from the prior distribution. We assume this random sample to be truth, i.e., the true permeability field in the subsurface. Experimental data are simulated by solving the single-phase pressure equation given the injection and production settings specified in section 6.4. Given the simulated noisy data, we carry out the online process of goal-oriented inference: the previously learned joint density is conditioned at the observed data to obtain the posterior predictive density. The raw observed data and

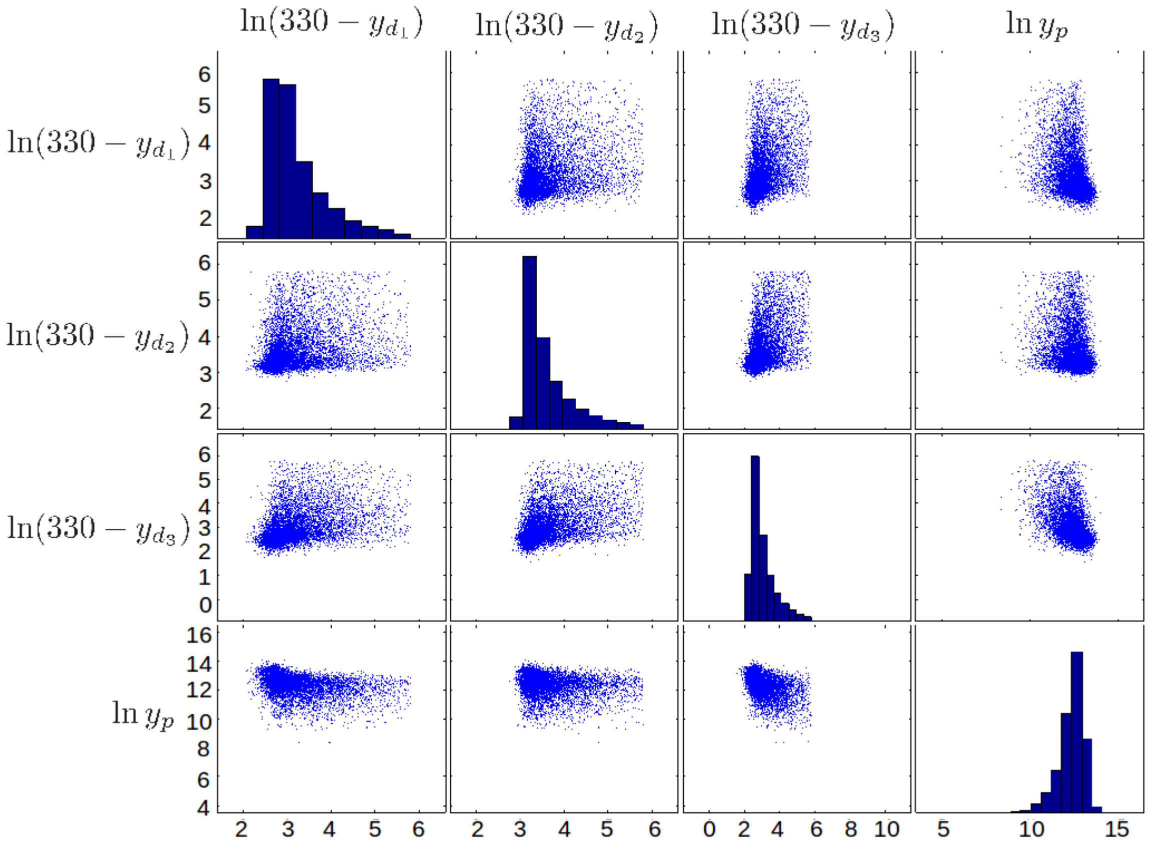


Figure 6-11: Pairwise marginal data and histograms after the transformations of (6.12) have been applied.

predictions based on the truth sample of permeability are given in Table 6.2.

Table 6.2: For the six assumed truth samples of the permeability field, we give the figure where the posterior predictive is plotted, the raw observed data, and the true prediction.

Figure	y_{d_1} (bar)	y_{d_2} (bar)	y_{d_3} (bar)	y_p (m^3)
6-12(a)	282.124	294.804	293.153	1.294×10^5
6-12(b)	294.865	298.440	295.787	3.416×10^5
6-12(c)	298.654	294.842	279.562	2.795×10^5
6-12(d)	291.663	288.973	293.773	1.994×10^5
6-12(e)	278.557	296.358	298.237	3.764×10^5
6-12(f)	298.061	273.317	293.704	2.948×10^5

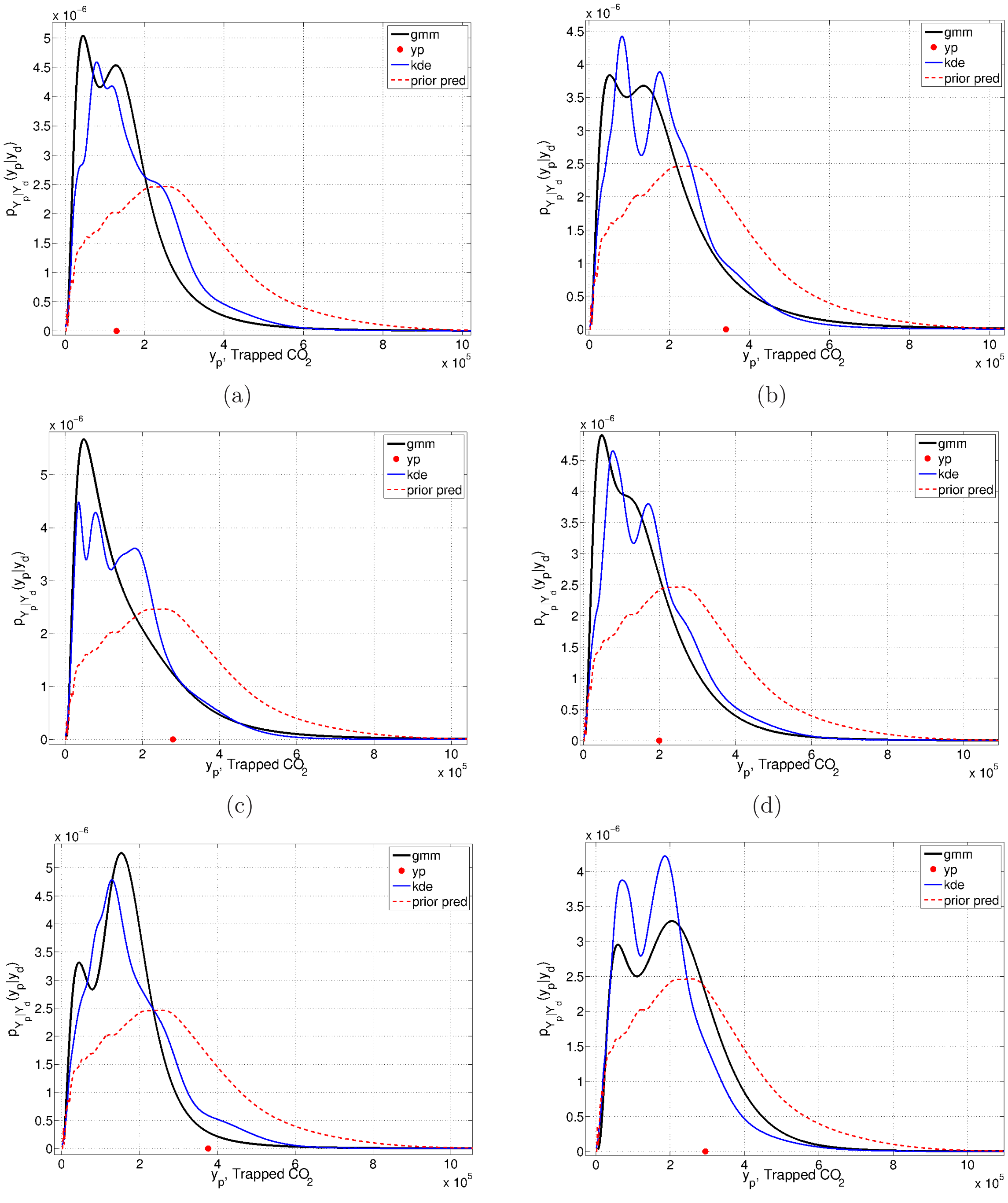
Figure 6-12 presents the posterior predictive by both GMM and KDE for some truth samples of the permeability field. For reference, we also show the prior predictive density as obtained by a KDE over all samples of prediction from the offline phase. For each case, we also plot the prediction quantity of interest obtained by simulating the prediction process for the given truth sample. Note that this is a deterministic quantity; however, we expect it to appear within the significant support of the posterior predictive with high probability.

In each of the results we notice trends similar to those observed in the model problems in section 5.5. The KDE generally produces results with greater total variation since it is more sensitive to the locality of samples of the joint density near where data are observed. On the other hand, the GMM tends to smooth the result since it involves only 15 Gaussian components, each with greater kernel width than the kernel from KDE, which has as many components as original samples. In all cases, the GMM and KDE are in general agreement, particularly in the manner with which the posterior predictive density differs from the prior predictive density. It is this update in information based on the observed data which is critical to our goal-oriented inference.

The posterior predictive densities represent our updated belief in the relative likelihood of trapped volume of carbon dioxide under the pumping scenario described in section 6.5 given our prior specification on the parameter and the observed data from experiments. We have a full probabilistic description of the trapped volume enabled by focusing on the relationship between experimental data and the prediction quantity of interest, all in an application where best practices in statistical inference of the parameter would have been insufficient. In practice these results would feed forward to a decision-making process where it would be determined if sufficient volume of the carbon dioxide would be trapped in the aquifer to proceed with the injection.

6.6.3 Effect of number of offline samples

In this section we investigate numerically the effect of the number of offline samples of parameter on the resulting prediction accuracy. We learn separate GMMs using 100 samples, 1000 samples, and all 11075 samples and compare the posterior predictive results against those obtained by KDE for several realizations of the data. The results are shown in Figure 6-13.



(a)

(b)

(c)

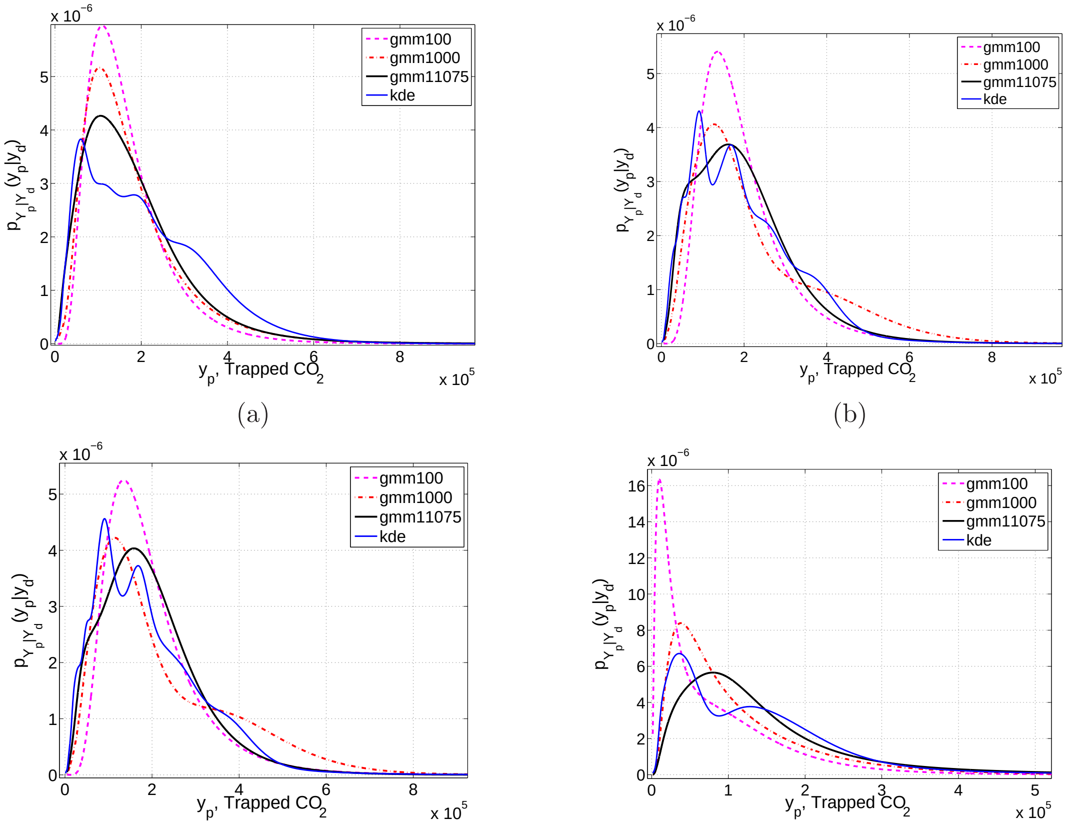
(d)

(e)

(f)

Figure 6-12: Posterior predictive densities for samples of the permeability field. Results from the GMM and KDE are both presented. The prior predictive density and true prediction are given as reference. Observed data values are given in Table 6.2.

The trend is expected. The approximation of the posterior predictive density by GMM appears to improve with more offline samples. Since we assume that the prior



(a)

(b)

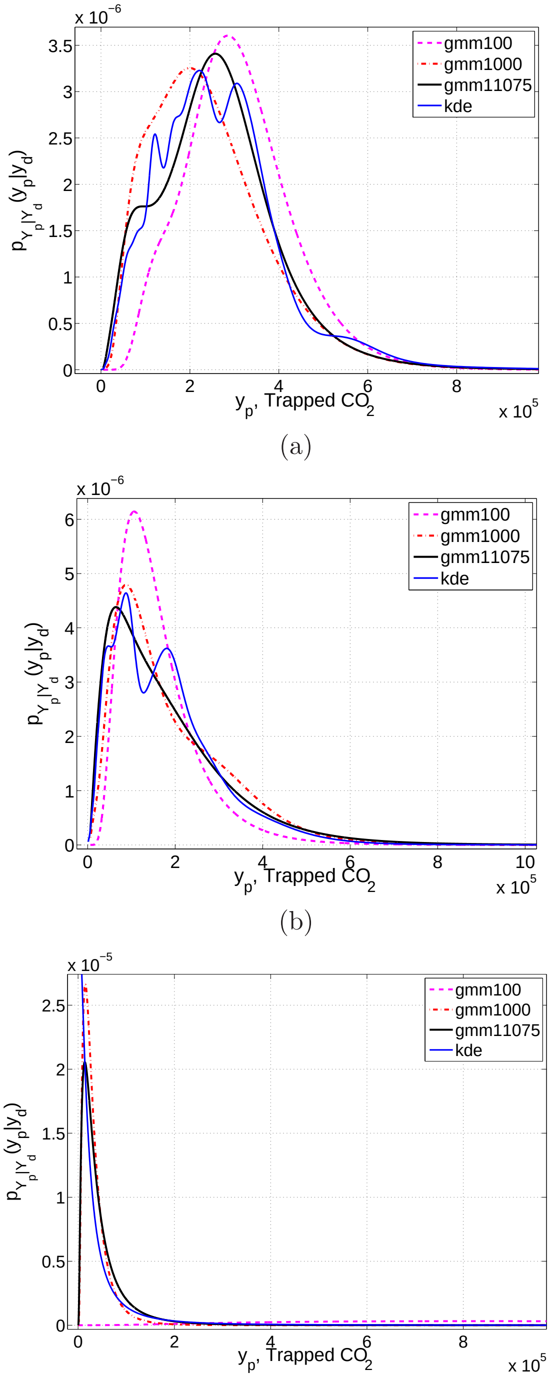
(c)

(d)

Figure 6-13: Posterior predictive densities for four samples of the permeability field representing the true permeability synthetically. Results from GMMs learned using 100 samples, 1000 samples, and 11075 samples offline are compared to solutions obtained using KDE.

is efficient to sample and we build the GMM before experiments are performed, we recommend that as much data are acquired as possible to build the GMM. If data are difficult or expensive to obtain for some reason, it should be noted that the accuracy of the GMM (as well as many other estimation procedures) will be affected significantly.

In order to study the dependency of the accuracy of the approach on proximity of the observed data point to the offline samples, we explore three cases where we artificially prescribe data. In the first case, we set the data to be in a high-density region of the samples for all of the mixture models we trained with different numbers of offline samples. For the second case, we select the data artificially to be proximal to many samples from the 1000-sample and 11075-sample GMMs but in a low-density region of the 100-sample GMM. Finally, in the last case, we select the data artificially to be in a low-density region of all of the GMMs. The results are shown in Figure 6-14.



(a)

(b)

(c)

Figure 6-14: Posterior predictive densities for three artificial realizations of data: (a) proximal to points used to train all GMMs, (b) proximal to points for many-sample GMMs but in low-density region of the 100-sample GMM, and (c) in low-density region of all GMMs. Results from GMMs learned using 100 samples, 1000 samples, and 11075 samples offline are compared to solutions obtained using KDE.

For the first case (see Figure 6-14(a)), there is general agreement of all of the GMMs with the KDE posterior predictive density. This is expected behavior since the model should be accurate in high sample density regions. In the second case (see Figure 6-14(b)), the GMM based on 100 samples is inaccurate since the observed data occurs in a low sample density region. Finally, in the last case (see Figure 6-14(c)), all of the posterior predictive density approximations fail. The observed data are well outside the offline samples. This reflects that these density estimations are susceptible to large errors when extrapolation is required. For this reason, it is critical that one attempt to detect such situations before trusting the resulting posterior predictive density.

Chapter 7

Summary and future work

In this final chapter we provide a summary of the work, enumerate the contributions of this thesis, and provide some thoughts on possible extensions.

7.1 Summary

We have introduced goal-oriented inference as a new approach to estimation of prediction quantities of interest in the context of unknown distributed parameters. Traditional approaches of parameter estimation followed by forward propagation are insufficient for enabling real-time online prediction computations and expend extensive computational resources on parameter inference when it is not necessary to do so.

We have developed a new approach in the linear setting that resulted in a set of goal-oriented inference algorithms companion to popular existing parameter inference algorithms. By letting the prediction requirements drive the parameter inference, an offline analysis of the experiment and prediction processes reveals a dimensionally-optimal subspace regularizer for the parameter inference. As a result the linear operator transforming observed data to predictions can be precomputed offline and stored, due to its low dimensionality. When observations are collected online, predictions can be obtained in real-time. Our method also has the benefit of revealing important properties of the inference in the context of predictions by identifying experimental inefficiency and modes of prediction uncertainty. This information could be used as inputs to an optimal experimental design. The linear approach was applied to a model problem in contaminant source identification and prediction where numerical results corroborated the theoretical findings.

In the nonlinear statistical setting we have developed a practical method for goal-oriented inference involving a priori parameter sampling and learning the joint density of predictions and experimental data. The method exploits the low-dimensionality of the product space of experimental data and predictions. The accuracy of the method depends on the ability to sample the parameter space and simulate experimental data and predictions; however, this can be completed entirely offline before the real data are observed, and the process is embarrassingly parallel meaning that one could obtain optimal parallel scaling on a distributed computing architecture. Once the joint density is learned, the experiment is performed, and the density is conditioned on the observed data to obtain the posterior predictive density in real-time. The approach was demonstrated on an important problem in carbon capture and storage, where the very high-dimensional parameter space puts the traditional approaches out of reach of state-of-the-art statistical inference techniques. Since we focus on prediction quantities of interest, we circumvent the issues associated with inferring the parameter itself; instead, we focus on the relationship between experimental data and prediction quantities of interest.

The contributions of this thesis are:

- formulation of goal-oriented inference, allowing predictions to drive the inference process;
- development of a set of goal-oriented inference procedures companion to well-established parameter identification algorithms;
- establishment of offline analysis tools to guide experimental design and expose sources of prediction uncertainty for linear problems;
- derivation of theoretical guarantees on the prediction accuracy of goal-oriented inference for linear problems;

- demonstration of linear goal-oriented inference on a model problem in contaminant identification and prediction;

- development of a practical algorithm that extends goal-oriented inference to nonlinear problems;

- and demonstration of nonlinear goal-oriented inference in performing a probabilistic risk assessment in carbon capture and storage (CCS).

7.2 Future work

One of the goals of this work was to establish goal-oriented inference as a new branch in the inference tree. We are hopeful that others will take up some of the possible extensions to this work. In this section we briefly describe a few possibilities in this direction.

The decompositions of experiment and prediction processes in the linear setting reveal information about how modes in parameter space affect experimental data and prediction outputs. In our developments, we have assumed the definition of the experimental process. One can easily envision extending the idea of goal-oriented inference to the experimental process as well. This would be consistent with the concept of allowing prediction requirements to drive the process of experiment in addition to the inference. Ideally one would choose experiments to align the experimentally observable modes with those of the prediction; however, in many cases, this may be physically impossible or too costly. An optimal experimental design formulation would account for the feasibility and cost of experiments in an attempt to align these spaces.

Extending the goal-oriented nature of the approach to experimental design may be accompanied by pushing beyond predictions and on to decisions as well. In this work, we focus on driving inference by prediction requirements, using predictions as a substitute for final decisions. However, particularly in the statistical setting, decisions would be derived from the posterior predictive density. If we encode that decision-making process and include it as part of the *prediction*, we expect to find that prediction requirements differ and may require less information about the unknown parameter. For example, if the decision about going ahead with a carbon capture strategy relies on the trapped carbon dioxide volume exceeding a certain value with given probability, then it is not necessary to resolve the posterior predictive density but rather just to obtain that probability accurately.

The goal-oriented inference approach for nonlinear problems in the statistical setting treats the experimental and prediction processes as black box models. It may be possible to improve the accuracy and/or efficiency of the approach by exploiting structure in these processes if they are known. In particular, exploiting sensitivity information of the experimental data and prediction outputs with respect to the parameter, information which varies as a function of the parameter itself, may be possible. Linear goal-oriented inference can be applied locally in parameter space, but transitioning between parametric descriptions is still an open problem. Some nonlinear model reduction concepts (e.g., trajectory piecewise linearization [69], interpolation on Grassmann manifolds [4], or empirical interpolation methods [8, 20]) may be fruitfully applied or adapted in this context.

Bibliography

[1] H. Akaike. A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19(6):716–723, 1974.

[2] O. Alifanov. *Inverse heat transfer problems*. Springer-Verlag, Berlin, Germany, 1994.

[3] H. Ammari, E. Bonnetier, Y. Capdeboscq, M. Tanter, and M. Fink. Electrical impedance tomography by elastic deformation. *SIAM Journal on Applied Mathematics*, 68(6):1557–1573, 2008.

[4] D. Amsallem and C. Farhat. Interpolation method for adapting reduced-order models and application to aeroelasticity. *AIAA Journal*, 46(7):1803–1813, 2008.

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

[6] P. Astrid, S. Weiland, K. Willcox, and T. Backx. Missing point estimation in models described by proper orthogonal decomposition. *IEEE Transaction on Automatic Control*, 53(10):2237–2251, 2008.

[7] H.T. Banks and K. Kunisch. *Estimation Techniques for Distributed Parameter Systems*. Birkhuser, 1989.

[8] M. Barrault, Y. Maday, N.C. Nguyen, and A.T. Patera. An 'empirical interpolation' method: application to efficient reduced-basis discretization of partial differential equations. *Numerical Analysis*, 339:667–672, 2004.

[9] 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 for Numerical Methods in Engineering*, 73(6):844–868, 2008.

[10] P. Benner, E.S. Quintana-Orti, and G. Quintana-Orti. Balanced truncation model reduction of large-scale dense systems on parallel computers. *Mathematical and Computer Modelling of Dynamical Systems*, 6(4):383–405, 2000.

[11] J. Berger. The case for objective Bayesian analysis. *Bayesian Analysis*, 1(3):385–402, 2006.

[12] G. Berkooz, P. Holmes, and J.L. Lumley. The proper orthogonal decomposition in the analysis of turbulent flows. *Annual Review of Fluid Mechanics*, 25:539–575, 1993.

[13] J. Besag, P. Green, D. Higdon, and K. Mengerson. Bayesian computation and stochastic systems. *Statistical Science*, 10(1):3–41, 1995.

[14] L.T. Biegler, O. Ghattas, M. Heinkenschloss, D. Keyes, and B. van Bloemen Waanders. *Real-Time PDE-Constrained Optimization*. SIAM, Philadelphia, PA, 2007.

[15] A. Brooks and T.J.R. Hughes. Streamline upwind/Petrov-Galerkin methods for advection-dominated flows. In *Proceedings of the Third International Conference on Finite Element Methods in Fluid Flows*, pages 283–292, 1980.

[16] T. Bui-Thanh, K. Willcox, and O. Ghattas. Model reduction for large-scale systems with high-dimensional parametric input space. *SIAM J. Sci. Comp.*, 30:3270–3288, 2008.

[17] C. Burstedde and O. Ghattas. Algorithmic strategies for full waveform inversion: 1d experiments. *Geophysics*, 74(6):WCC37–WCC46, 2009.

[18] B. Calderhead. *Differential geometric MCMC methods and applications*. PhD thesis, University of Glasgow, 2012.

[19] G. Casella. Objective Bayesian variable selection. *Journal of the American Statistical Association*, 101(473):157–167, 2012.

[20] S. Chaturantabut and D.C. Sorensen. Nonlinear model reduction via discrete empirical interpolation. *SIAM Journal on Scientific Computing*, 32(5):2737–2764, 2010.

[21] S. S. Chen and P. S. Gopalakrishnan. Clustering via the Bayesian information criterion with applications in speech recognition. *Proceedings of the 1998 IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2:645–648, May 1998.

[22] J. Christen and C. Fox. A general purpose sampling algorithm for continuous distributions (the t-walk). *Bayesian Analysis*, 5:263–281, 2010.

[23] J.K. Cullum and R.A. Willoughby. *Lanczos Algorithms for Large Symmetric Eigenvalue Computations: Theory*. SIAM, Philadelphia, PA, 2002.

[24] L. Daniel, O.C. Siong, L.S. Chay, K.H. Lee, and J. White. Multiparameter moment matching model reduction approach for generating geometrically parameterized interconnect performance models. *Transactions on Computer Aided Design of Integrated Circuits*, 23(5):678–693, May 2004.

[25] A. Dempster, N. Laird, and D. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, 39:1–38, 1977.

[26] Y. Efendiev, T. Hou, and W. Luo. Preconditioning Markov chain Monte Carlo simulations using coarse-scale models. *SIAM J. Sci. Comp.*, 28:776–803, 2006.

[27] H. Engl, M. Hanke, and A. Neubauer. *Regularization of Inverse Problems.* Kluwer Academic Publishers, Dordrecht, The Netherlands, 2000.

[28] M. Figueiredo and A. Jain. Gaussian Mixture Models for MATLAB. *Available at http://www.lx.it.pt/∼mtf/*, 2002.

[29] M. Figueiredo and A. Jain. Unsupervised learning of finite mixture models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24:381–396, 2002.

[30] K. Gallivan, E. Grimme, and P. Van Dooren. Padé Approximation of Large-Scale Dynamic Systems with Lanczos Methods. Proceedings of the 33rd IEEE Conference on Decision and Control, December 1994.

[31] A. Gelman. Objections to Bayesian statistics. *Bayesian Analysis*, 3(3):445–449, 2008.

[32] D. Georges. The use of observability and controllability gramians or functions for optimal sensor and actuator location in finite-dimensional systems. In *IEEE Proceedings of the 34th Conference on Decision and Control, New Orleans, LA*, December 1995.

[33] W. Gilks, S. Richardson, and D. Spiegelhalter. *Markov Chain Monte Carlo in Practice.* Chapman & Hall, 1996.

[34] M. Girolami and B. Calderhead. Riemann manifold Langevin and Hamiltonian Monte Carlo methods. *J. R. Stat. Soc. Ser. B Stat. Methodol.*, 73:123–214, 2011.

[35] J.G.D. Gooijer and D. Zerom. On conditional density estimation. *Statistica Neerlandica*, 57:159–176, 2003.

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

[37] S. Gugercin and A.C. Antoulas. A survey of model reduction by balanced truncation and some new results. *International Journal of Control*, 77(8):748–766, 2004.

[38] J. Hadamard. *Lectures on Cauchy's problem in linear partial differential equations.* Dover Publications, New York, 1923.

[39] P.C. Hansen. Analysis of discrete ill-posed problems by means of the L-curve. *SIAM Review*, 34(4), 1992.

[40] P.C. Hansen. *Rank-Deficient and Discrete Ill-Posed Problems: Numerical Aspects of Linear Inversion*. SIAM, Philadelphia, 1998.

[41] T. Hohage and S. Langer. Acceleration techniques for regularized Newton methods applied to electromagnetic inverse medium scattering problems. *Inverse Problems*, 26:074011, 2010.

[42] C. Huang, T. Hsing, N. Cressie, A. Ganguly, A. Protopopescu, and N. Rao. Bayesian source detection and parameter estimation of a plume model based on sensor network measurements. *Applied Stochastic Models in Business and Industry*, 26(4):331–348, 2006.

[43] A. Ihler and M. Mandel. Kernel Density Estimation Toolbox for MATLAB. *Available at http://www.ics.uci.edu/∼ihler/code/kde.html*, 2003.

[44] V. Isakov. *Inverse Problems for Partial Differential Equations*. Springer, 1998.

[45] J. Kaipio and E. Somersalo. *Statistical and Computational Inverse Problems*. Springer, 2005.

[46] R.E. Kalman. On the general theory of control systems. In *Proceedings of the First International Congress on Automatic Control*, pages 481–491, 1960.

[47] E. Kalnay. *Atmospheric Modeling, Data Assimilation, and Predictability*. Cambridge University Press, Cambridge, United Kingdom, 2003.

[48] K. Karhunen. Uber lineare methoden in der wahrscheinlichkeitsrechnung. *Ann. Acad. Sci. Fennicae. Seri. A. I. Math.-Phys.*, 37:1–79, 1947.

[49] K. Kunisch and S. Volkwein. Control of Burgers' equation by reduced order approach using proper orthogonal decomposition. *Journal of Optimization Theory and Applications*, 102:345–371, 1999.

[50] J. Li and J. White. Reduction of large circuit models via low rank approximate gramians. *International Journal of Applied Mathematics and Computer Science*, 11(5):101–121, 2001.

[51] F. Liang, C. Liu, and R. Carroll. *Advanced Markov Chain Monte Carlo Methods: Learning from Past Samples*. John Wiley & Sons, 2011.

[52] K.-A. Lie, S. Krogstad, I.S. Ligaarden, J.R. Natvig, H.M. Nilsen, and B. Skaflestad. Open source matlab implementation of consistent discretisations on complex grids. *Computational Geosciences*, 16(2):297–322, 2012.

[53] C. Lieberman, K. Fidkowski, K. Willcox, and B. van Bloemen Waanders. Hessian-based model reduction: large-scale inversion and prediction. *International Journal for Numerical Methods in Fluids*, 71(2):135–150, 2012.

[54] C. Lieberman, K. Willcox, and O. Ghattas. Parameter and state model reduction for large-scale statistical inverse problems. *SIAM Journal on Scientific Computing*, 32(5):2523–2542, 2010.

[55] I. Ligaarden and H.M. Nilsen. Numerical aspects of using vertical equilibrium models for simulating co2 sequestration. In *Proceedings of ECMOR XII – 12th European Conference on the Mathematics of Oil Recovery, Oxford, UK*, 1995.

[56] M. Loeve. *Probability Theory*. Springer-Verlag, 1978.

[57] J. Martin, L.C. Wilcox, C. Burstedde, and O. Ghattas. A stochastic Newton MCMC method for large-scale statistical inverse problems with application to seismic inversion. *SIAM J. Sci. Comp.*, 34(3):A1460–A1487, 2012.

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

[59] N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller, and E. Teller. Equations of state calculations by fast computing machines. *J. Chem. Phys.*, 21:1087–1092, 1953.

[60] B.C. Moore. Principal component analysis in linear systems: Controllability, observability, and model reduction. *IEEE Transactions on Automatic Control*, AC-26(1):17–31, August 1981.

[61] I. Murray. *Advances in Markov chain Monte Carlo methods*. PhD thesis, University College London, 2007.

[62] J.M. Nordbotten and M.A. Celia. *Geological Storage of CO2: Modeling Approaches for Large-Scale Simulation*. John Wiley & Sons, 2011.

[63] M.A. Oliver and R. Webster. Kriging: A method of interpolation for geographical information systems. *Int. J. Geographical Information Systems*, 4(3):313–332, 1990.

[64] E. Parzen. On estimation of a probability density function and mode. *The Annals of Mathematical Statistics*, 33(3):1065–1076, 1962.

[65] W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery. *Numerical Recipes: The Art of Scientific Computing*. Cambridge University Press, 2007.

[66] C. Prud'homme, D. Rovas, K. Veroy, Y. Maday, A.T. Patera, and G. Turinici. Reliable real-time solution of parameterized partial differential equations: Reduced-basis output bound methods. *Journal of Fluids Engineering*, 124:70–80, 2002.

[67] R. Rannacher. Adaptive Galerkin finite element methods for partial differential equations. *J. Comp. Appl. Math.*, 128(1–2):205–233, 2001.

[68] C.E. Rasmussen and C.K.I. Williams. *Gaussian Processes for Machine Learning.* MIT Press, 2006.

[69] M. Rewienski and J. White. A trajectory piecewise-linear approach to model order reduction and fast simulation of nonlinear circuits and micromachined devices. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 22(2):155–170, 2003.

[70] M. Rosenblatt. Remarks on some nonparametric estimates of a density function. *The Annals of Mathematical Statistics*, 27(3):832–837, 1956.

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

[72] C.R. Smith and W.T. Grandy Jr. *Maximum Entropy and Bayesian Methods in Inverse Problems.* Springer, 1985.

[73] H. Steinhaus. Sur la division des corps materiels en parties. *Bull. Acad. Polon. Sci.*, 4(12):801–804, 1957.

[74] A.M. Stuart. Inverse problems: A Bayesian perspective. *Acta Numerica*, 19:451–559, 2010.

[75] A. Tarantola. *Inverse Problem Theory and Methods for Model Parameter Estimation.* SIAM, Philadelphia, PA, 2005.

[76] L. Tierney. Markov chains for exploring posterior distributions. *The Annals of Statistics*, 22(4):1701–1728, 1994.

[77] 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.

[78] 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. AIAA Paper 2003-3847, Proceedings of the 16th AIAA Computational Fluid Dynamics Conference, Orlando, FL, 2003.

[79] C. Vogel. *Computational Methods for Inverse Problems.* SIAM, Philadelphia, 2002.

[80] G. Wang and S. Chen. Evaluation of a soil greenhouse gas emission model based on Bayesian inference and MCMC: Parameter identifiability and equifinality. *Ecological Modelling*, 253:107–116, 2013.