

Clemson University

TigerPrints

All Dissertations

Dissertations

8-2019

Stochastic Parameter Estimation of Poroelastic Processes Using Geomechanical Measurements

Alexander Charles Hanna
Clemson University

Follow this and additional works at: https://tigerprints.clemson.edu/all_dissertations

Recommended Citation

Hanna, Alexander Charles, "Stochastic Parameter Estimation of Poroelastic Processes Using Geomechanical Measurements" (2019). *All Dissertations*. 2478.
https://tigerprints.clemson.edu/all_dissertations/2478

This Dissertation is brought to you for free and open access by the Dissertations at TigerPrints. It has been accepted for inclusion in All Dissertations by an authorized administrator of TigerPrints. For more information, please contact kokeefe@clemson.edu.

STOCHASTIC PARAMETER ESTIMATION OF POROELASTIC PROCESSES USING GEOMECHANICAL MEASUREMENTS

A Dissertation
Presented to
the Graduate School of
Clemson University

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy
Environmental Engineering and Earth Science

by
Alexander Hanna
August 2019

Accepted by:
Dr. Lawrence Murdoch, Committee Chair
Dr. Stephen Moysey
Dr. Scott DeWolf
Dr. Brian Dean

Abstract

Understanding the structure and material properties of hydrologic systems is important for a number of applications, including carbon dioxide injection for geological carbon storage or enhanced oil recovery, monitoring of hydraulic fracturing projects, mine dewatering, environmental remediation and managing geothermal reservoirs. These applications require a detailed knowledge of the geologic systems being impacted, in order to optimize their operation and safety. In order to evaluate, monitor and manage such hydrologic systems, a stochastic estimation framework was developed which is capable of characterizing the structure and physical parameters of the subsurface. This software framework uses a set of stochastic optimization algorithms to calibrate a heterogeneous subsurface flow model to available field data, and to construct an ensemble of models which represent the range of system states that would explain this data.

Many of these systems, such as oil reservoirs, are deep and hydraulically isolated from the shallow subsurface making near-surface fluid pressure measurements uninformative. Near-surface strainmeter, tiltmeter and extensometer signals were therefore evaluated in terms of their potential information content for calibrating poroelastic flow models. Such geomechanical signals are caused by mechanical deformation, and therefore travel through hydraulically impermeable rock much more quickly. A numerical geomechanics model was therefore developed using Geocentric, which couples subsurface flow and elastic deformation equations to simulate geomechanical signals (e.g. pressure, strain, tilt and displacement) given a set of model parameters. A high-performance cluster computer performs this computationally expensive simulation for each set of parameters, and compares the simulation results to measured data in order to evaluate the likelihood of each model. The set of data-model comparisons are then used to estimate each unknown parameter, as well as the uncertainty of each parameter estimate. This uncertainty can be influenced by limitations in the

measured dataset such as random noise, instrument drift, and the number and location of sensors, as well as by conceptual model errors and false underlying assumptions.

In this study we find that strain measurements taken from the shallow subsurface can be used to estimate the structure and material parameters of geologic layers much deeper in the subsurface. This can significantly mitigate drilling and installation costs of monitoring wells, as well as reduce the risk of puncturing or fracturing a target reservoir. These parameter estimates were also used to develop an ensemble of calibrated hydromechanical models which can predict the range of system behavior and inform decision-making on the management of an aquifer or reservoir.

Table of Contents

Title Page	i
Abstract	ii
List of Tables	vi
List of Figures	vii
1 Development of a cloud computing framework for model calibration and decision support in the geosciences	3
1.1 Abstract	3
1.2 Introduction	4
1.3 Methods	11
1.4 Results and Discussion	19
1.5 Conclusion	31
2 Numerical proof-of-feasibility of using geomechanical measurements to estimate poroelastic parameters	32
2.1 Abstract	32
2.2 Introduction	33
2.3 Background	34
2.4 Methods	39
2.5 Results and Discussion	44
2.6 Conclusion	51
3 Using borehole strain tensor measurements to locate and characterize subsurface heterogeneities in the Bartlesville formation in Avant field, Oklahoma . . .	52
3.1 Abstract	52
3.2 Introduction	53
3.3 Background	54
3.4 Methods	55
3.5 Results and Discussion	60
3.6 Conclusion	65
4 Combining wellbore strainmeter and pressure transducer data for improved reservoir characterization	66
4.1 Abstract	66
4.2 Introduction	66
4.3 Methods	67
4.4 Results and Discussion	76
4.5 Conclusion	83

5	Using geomechanical data to image heterogeneous subsurface flow systems . . .	84
5.1	Abstract	84
5.2	Introduction	85
5.3	Methods	85
5.4	Results and Discussion	91
5.5	Conclusion	100
	Bibliography	103

List of Tables

1.1	Transmissivity and storativity estimates for the drawdown data in Walton, 1970 [63] and McElwee, 1980 [41] are shown. Inversion results are also shown using the Voronoi, Gaussian KDE, and MCMC approaches to likelihood integration (Section 1.3.3.1). Uncertainties marked with an asterisk (*) were calculated using Equation 1.19, and were not drawn from the references. The analytical estimate was calculated by choosing the best-fitting model and evaluating the derivative at that point in the parameter space.	27
2.1	Summary of instruments and instrument locations. Fluid pressures (p_f), displacement components (u_x, u_y, u_z), strain components ($\varepsilon_x, \varepsilon_y, \varepsilon_z$) and tilt components ($\nabla_z u_x, \nabla_z u_y$) measured at different instrument positions.	43
2.2	Estimates of permeability (κ), bulk modulus (K) and Poisson's ratio (ν) are shown in terms of mean (μ), error (δ) and standard deviation (2σ), with various instrument types included in the analysis. The synthetic datasets use a 2% noise level taken from the instrument locations specified in Table 2.1, and include various combinations of pressure (P), displacement (D), tilt (T), and strain (S) data.	47
2.3	Estimates of permeability (κ), bulk modulus (K) and Poisson's ratio (ν) are shown in terms of mean (μ), error (δ) and standard deviation (2σ) with various instrument locations included in the analysis. The synthetic datasets use a 2% noise level taken from all instrument types specified in Table 2.1, and include various combinations of the injection (inj) and observation (obs) wells.	47

List of Figures

1.1	Illustration of Pareto optimality. a.) Results from three solutions plotted with respect to objective functions f_1 and f_2 . Colored rectangles indicate region of lower Pareto rank. b.) Pareto ranking used to characterize model-data misfit for a set of 26 simulations with different parameter sets. Blue dashed line represents the 'true' Pareto front.	6
1.2	Latin Square	11
1.3	The simulated annealing process is illustrated. The prior model ($p(\theta)$) is shown at high temperatures (T) while the likelihood model ($\mathcal{L}(\theta)$) begins to dominate at lower temperatures.	12
1.4	Entity-Relationship diagram for the inversion database. Blue boxes indicate tables in the database, while lines indicate the cardinality relationships between tables. . . .	13
1.5	Illustration of two methods of estimating volume of the representative parameter space of each simulation. Color represents sparsity, or the volume of empty space surrounding each point. The (A) Voronoi tessellation method computes intersections between Voronoi polygons and the edges of the parameter space, and uses these intersection points to compute exact volumes. The (B) Gaussian KDE method instead fits a uni-modal Gaussian distribution to the points.	17
1.6	Multi-chain MCMC results are shown for two parameters (θ_1, θ_2) as a function of iteration, using an interpolation-based surrogate function as the fitting objective. . .	18
1.7	Comparison of Gradient Descent and Markov chain Monte Carlo for the sphere test function. Color indicates the value of the objective function while black dots indicate paths taken by Gradient Descent and Markov chain Monte Carlo.	20
1.8	Comparison of Gradient Descent and Markov Chain Monte Carlo for the Ackley test function. Color indicates the value of the objective function while black dots indicate paths taken by Gradient Descent and Markov chain Monte Carlo.	21
1.9	A: Rosenbrock function is shown in 3-dimensional space. The height of the surface (z-axis value) represents the output of the Rosenbrock function, while the color of the surface represents the base-10 Log of the Rosenbrock function. This highlights the narrowness and sharpness of the banana-shaped valley characteristic of the Rosenbrock test function. B: 10 chains of gradient descent, using a total of 5156 function evaluations. Zoomed inset shows zig-zag pattern characteristic of gradient descent. C: 10 chains of Markov chain Monte Carlo, using a total of 3897 function evaluations.	22
1.10	Visualization of Dropwave function.	23
1.11	Comparison of (A) Gradient Descent, (B) Markov Chain Monte Carlo, and (C) Genetic Algorithm for the Drop wave test function. Grayscale color indicates the value of the objective function while colored dots indicate paths taken by Gradient Descent and Markov chain Monte Carlo.	23
1.12	Visualization of both Binh-Korn functions, f_1 and f_2 . Local minima are at A and A'. Pareto front represented by red dashed line.	24

1.13	Comparison of Markov Chain Monte Carlo (A,B,D) and Gradient Descent (C) results using the weighted sum approach for subfigures A ($\omega_1 : \omega_2 = 0.25$) and B ($\omega_1 : \omega_2 = 4.0$), and C ($\omega_1 : \omega_2 = 1.0$), and the Pareto rank approach for subfigure D. Red dashed lines indicate the true Pareto front.	25
1.14	(A) Measured drawdown from a pumping test [63] is shown in red. Simulation results are shown in gray while the 50 best-fitting curves are highlighted in black. (B) Model likelihoods are plotted as color, as a function of storativity and transmissivity. . . .	27
1.15	Parameter estimates and uncertainties for transmissivity and storativity are made as functions of measurements error, using the Voronoi, Gaussian KDE, MCMC, and analytical error propagation approaches. Red dashed line indicates the estimate of the measurement error in the source dataset. Note that no parameter estimates are shown for the analytical error propagation method, as it does not estimate the parameter itself but only the uncertainty. The analytical error propagation uncertainty is also linearly dependent on the measurement error σ_n , and therefore is not subject to the stochastic variation shown by the other methods.	28
1.16	A). Map view layout of the fictional pumping test field site. B). Pareto graph shows Observation Well 1 data-model misfits plotted against their corresponding Observation Well 2 misfits at each noise level. C). Fictional 'measured' drawdown is shown (1% random noise) as red dots, compared to 5000 proposed simulations (gray lines). Pareto optimal models are highlighted in black. D). Data-model misfits are shown as a function of storativity and transmissivity.	29
1.17	A). Drawdown as a function of time is shown for observation well 1, both with and without the secondary well pumping. B). Fictional 'measured' drawdown is shown (1% random noise) as red dots, compared to 5000 proposed simulations (gray lines). Pareto optimal models are highlighted in black. C). Pareto rank of each simulation is shown as a function of transmissivity and storativity. D. Pareto graph shows Observation Well 1 data-model misfits plotted against their corresponding Observation Well 2 misfits.	30
2.1	Illustration of Pareto optimality. Blue dashed line represents the 'true' Pareto front. Rank 1 solutions (blue dots) represent the best currently-available approximation of the Pareto front.	37
2.2	Example of a mapping between a 2-dimensional parameter space and a 3-dimensional objective space.	38
2.3	Geometry and mesh of conceptual model for numerical poroelastic simulation. . . .	40
2.4	Illustration of parallel model calibration process. A hypothetical model M_{true} is run through the forward model to generate a synthetic dataset, d_{true} . Alternatively, field observations can be collected d_{measured} . A series of random models (M_i) are then generated based on the prior model (mean μ_0 , variance σ_0) and evaluated by the forward model to produce simulated datasets (d_i) which are then compared to the measured or synthetic dataset. These comparisons are used by a variety of inverse algorithms including Monte Carlo (MC), Gradient Descent (GD), Markov Chain Monte Carlo (MCMC), space-filling algorithms such as Latin Hypercube or Voronoi method (Sparse), genetic algorithms (NSGA-II, SPEA2). These algorithms use data-model comparisons to produce iteratively improved models and thereby reducing uncertainties and providing a better characterization of the posterior distribution (mean μ_i , variance σ_i).	42
2.5	Pareto optimal data fits are shown (top row) for fluid pressure, tilt, radial strain, and vertical strain at the observation well in the center of the injection formation. Simulation results are shown in gray while synthetic data measurements are shown in red. Data-model errors are presented as a function of Young's modulus and conductivity (middle row), and as a function of Young's modulus and Poisson's ratio (bottom row).	45

2.6	Parameter estimates and uncertainties are presented as a function of the Gaussian noise added to the synthetic data. Red error bars indicate separate runs of the parameter sampling (~ 4000 simulations each). Each gray error bars re-uses the same sample of ~ 4000 simulations, with different levels of Gaussian noise applied to the synthetic data.	46
2.7	Preliminary model calibration results. (A) Comparison of synthetic data and simulated responses of the pressure breakthrough curve for the deep observation well. (B) Data-model misfits (L_2 -norm) for pressure sensor in the deep observation well as a function of permeability and porosity. Dashed line indicates the set of best-fit solutions. (C) Probability density function based on best-fit solutions.	49
2.8	Preliminary model calibration results. (A) Comparison of synthetic data and simulated responses of the pressure breakthrough curve for the deep observation well. (B) Data-model misfits (L_2 -norm) for pressure sensor in the deep observation well as a function of permeability and porosity. Dashed line indicates the set of best-fit solutions. (C) Probability density function based on best-fit solutions.	49
2.9	Mean and standard deviation of the pressure of 171 plausible models is plotted as a function of radial distance from the injection site and vertical distance above the lower confining unit.	50
2.10	Mean and standard deviation of the displacement magnitude of 171 plausible models is plotted as a function of radial distance from the injection site and vertical distance above the lower confining unit.	51
3.1	Location and conceptual model of the field site for 1A Shut-in test. Cross-section on top shows stratigraphy with contrasts in permeability (circle) and locations of well 1A and Gladwin strainmeter at AVN2. Location map of field site with wells included in the model.	56
3.2	Computational mesh in plan view for entire model (left) and finely meshed region around well (inset).	57
3.3	Measured and simulated strains as functions time. Effects of wells 1 and 2 simulated individually (red and green), and their superposition (magenta), along with effects of both wells simulated simultaneously (dotted), and compared to field data (black). (a) NS strain, (b) EW strain and (c) shear strain.	58
3.4	Measured NS Strain (red) and high-order polynomial fit (blue) as functions of time.	60
3.5	Strain as a function of time for 1A Shut-in Test. Measured strain data (red), simulation results (gray). Best-fitting simulation results in black. Models with permeable confining layer included.	62
3.6	Strain as a function of time for 1A Shut-in Test. Measured strain data (red), simulation results (gray). Best-fitting simulation results in black. Models with permeable confining layer excluded.	62
3.7	The weighted error (Eq 3.1) of each simulation as a function of (a) the confining unit permeability and lens permeability, and (b) the confining unit permeability and confining unit bulk modulus. Blue points correspond to the black lines in Figure 3.5. Note that this plot projects a 9-dimensional parameter space to a 2-dimensional plane, and therefore some of the variation shown is due to the other 7 parameters.	63
3.8	Data-model misfits for pairs of strain components (Pareto plots) with (a) East-West strain against North-South, (b) East-West against shear strain, and (c) North-South strain against shear for 1A Shut-in test. Yellow dots indicate models where the confining unit has a high permeability (greater than 10^{-15} m ²), gray dots indicate models where the confining unit has a lower permeability, and black dots indicate models with low confining unit permeability and also fit the measured data within the selected error bounds (corresponding to black lines in Figure 3.6).	63

3.9	Histograms of the best-fitting parameters for low permeability confining unit from 1A-Shut-in test. Permeability as log base 10 m ² . Bulk moduli are in GPa.	64
3.10	(A). Lens geometries of the best-fitting models are shown with well locations for comparison. Note that these lenses correspond to the black lines in Figure 3.5. (B). Lens thickness isopach developed by Grand Resources Oil company.	64
4.1	Active injection wells in the Avant oil field are shown in map view. Geological conceptual model is shown in cross section.	68
4.2	Outer, coarse mesh is shown with inset finely meshed region.	69
4.3	The injection rate over time and total injected volume over time are shown for both the measured flow, and the smooth function approximating the measured flow. . . .	70
4.4	A histogram of simulation runtimes is shown.	72
4.5	Illustration of parallel model calibration workflow. Inversion data is stored in MySQL database, and Python code accesses this data to produce simulation input files and store them in an AWS S3 bucket. Compute nodes retrieve these input files, run them and produce output files, which are also stored in the S3 bucket and later retrieved by the database.	73
4.6	Entity-Relationship diagram for the inversion database. Blue boxes indicate tables in the database, while lines indicate the cardinality relationships between tables. . . .	74
4.7	Measured NS Strain is shown compared to a high-order polynomial fit of the data. .	75
4.8	Simulation results from the circular parameterization inversion are compared to their corresponding field measurements (red lines) for three components of strain, and three pressure transducer locations. Simulation results are shown in gray, while best-fitting results are highlighted in black.	76
4.9	Simulation results from the elliptical parameterization inversion are compared to their corresponding field measurements (red lines) for three components of strain, and three pressure transducer locations. Simulation results are shown in gray, while best-fitting results are highlighted in black.	77
4.10	Data-model misfits are shown for all Geocentric simulations, for several pairs of objectives. Red dots indicate circular lens simulations, blue dots indicate elliptical lenses. . . .	78
4.11	Parameter values for circular parameterization are shown as scatter plots in upper right and lower left corners, where color represents a weighted sum summarizing how well all six datasets (upper right) are fitted, and how well the three strain datasets are fitted (lower left). Blue points represent the best-fit models. Histograms along diagonal show the parameter distribution of best-fitting models, according to both the strain alone (S), and pressure and strain combined (P+S).	81
4.12	Parameter values for elliptical parameterization are shown as scatter plots in upper right and lower left corners, where color represents a weighted sum summarizing how well all six datasets (upper right) are fitted, and how well the three strain datasets are fitted (lower left). Blue points represent the best-fit models. Histograms along diagonal show the parameter distribution of best-fitting models, according to both the strain alone (S), and pressure and strain combined (P+S).	82
4.13	Best-fitting lens geometries are shown for both circular and elliptical parameterizations. . . .	83
5.1	Active injection wells in the Avant oil field are shown in map view. Geological conceptual model is shown in cross section.	86
5.2	Outer, coarse mesh is shown with inset finely meshed region.	87
5.3	The injection rate over time and total injected volume over time are shown for both the measured flow, and the smooth function approximating the measured flow. . . .	88

5.4	Illustration of parallel model calibration workflow. Inversion data is stored in MySQL database, and Python code accesses this data to produce simulation input files and store them in an AWS S3 bucket. Compute nodes retrieve these input files, run them and produce output files, which are also stored in the S3 bucket and later retrieved by the database.	89
5.5	Three stage heterogeneity inversion process uses a circular assumption (A), elliptical assumption (B), followed by an irregularly shaped heterogeneity (C). Gray circles represent the Latin Hypercube search of the entire space, while green circles represent the genetic algorithm result, and blue circles represent the best-fitting models possible under the circular assumption. Gray ellipses represent the second-stage Latin Hypercube search of the elliptical parameter space, supplemented by the previous best-fitting circles from the first stage. Red ellipses represent the best-fitting ellipses found using NSGA-II. Black line represents the jagged, irregularly shaped heterogeneity arrived at by NSGA-II in the third stage.	91
5.6	Data fits are shown for three components of strain at the the AVN2 observation well, and for pressure data at wells 27, 29 and 60. Simulation results are shown in gray while best-fit simulations are highlighted in black. Measured field data shown as red dots.	92
5.7	Data-fitting tradeoffs are presented for each pair of objectives.	93
5.8	Results from two simulations (a) and (b) using irregular lens geometries for the inversion of the October 2017 injection test at well 9A. Each group shows the lens geometry in map view in the left corner. Parameters are shown in upper right corner, where red line represents the parameter value used in the particular simulation, and blue bars are the distribution of 1,886 best-fit models. Lower row shows data fit as black line, with gray lines representing 1,886 best-fit models.	94
5.9	Weighted average lens geometry is shown in map view for both the object based (circular and elliptical) lenses (left) and the irregular shaped lenses (right). The 1,886 best-fit models are selected and given a rank according to their weighted data misfit.	95
5.10	Strain components at AVN (North-South, East-West, shear) as functions of time for the November 9A injection test. Vertical strain at well AVN3, areal strain at AVN4. Simulation results are shown in gray and best-fit simulations are in black. Measured field data shown as red dots.	96
5.11	Pressure readings at wells 27, 29 and 60 as functions of time for the November 9A injection test. Simulation results are shown in gray and best-fit simulations are in black. Measured field data shown as red dots.	97
5.12	Tilt components at AVN3 and AVN4 as functions of time for the November 9A injection test. Simulation results are shown in gray and best-fit simulations are in black. Measured field data shown as red dots.	97
5.13	Differences in predicted strains are shown as a function of position. For each mesh element, the greatest difference is computed between any two of the selected best-fit models (corresponding to 5.10, 5.11). East-West (a), North-South (b), shear (c), vertical (d) and areal (e) strain components are shown. Results are displayed in map view with the horizontal slice defined at a depth of 30 meters, the approximate depth of the currently installed strainmeters.	98
5.14	Maximum fluid pressure (a) and maximum difference in fluid pressure (b) are shown as a function of position in map view. For each mesh element the maximum pressure represents the highest pressure indicated by any of the selected best-fit models, while the maximum difference represents the greatest pressure difference between any two best-fit models. Horizontal slice is located in the vertical center of the permeable lens.	98

5.15	Strain components at AVN (North-South, East-West, shear) as functions of time for the July 2018 9A injection test. Vertical strain at well AVN3, areal strain at AVN4. Simulation results are shown in gray and best-fit simulations are in black. Measured field data shown as red dots.	99
5.16	Pressure readings at wells 27, 29 and 60 as functions of time for the July 2018 9A injection test. Simulation results are shown in gray and best-fit simulations are in black. Measured field data shown as red dots.	100
5.17	Tilt components at AVN3 and AVN4 as functions of time for the July 2018 9A injection test. Simulation results are shown in gray and best-fit simulations are in black. Measured field data shown as red dots.	100

Preface

The overall purpose of this dissertation is to evaluate how well strain tensor measurements taken within the shallow subsurface can be used to estimate the structure and material properties of remote geologic layers at greater depths, and to develop a reservoir calibration tool that uses minimally invasive data collection methods to characterize a geomechanical system and forecast its future behavior. New instruments were available, capable of measuring the elastic strain response to well injection in a deep confined formation. A field test was therefore conducted evaluating the capabilities of these sensors as well as the information value of the data they provide. Sensors were installed in the Avant field, an oil field north of Tulsa, Oklahoma. A realistic simulation of this pumping experiment required a computationally intensive 3D hydromechanical solver, so parallelized inversion software was developed to run many iterations of this simulation until convergence to a calibrated hydromechanical model.

During the course of this doctoral research five journal article manuscripts were prepared for publication, and are presented here as chapters of this dissertation.

In Chapter One, a novel parameter estimation software called Distributed Multiobjective Stochastic Optimization Framework (DMSOF) is developed and tested against benchmark inverse problems with well known solutions and convergence properties. These include four benchmarks taken from the inversion literature as well as one example problem from the hydrogeologic literature.

In Chapter Two, this parameter estimation software is used to analyze synthetic data, which is produced by a finite element multiphysics solver. This provides proof-of-concept and indicates that when signal to noise and conceptual model issues are accounted for, strain data measured in the field can be sufficient to infer geomechanical parameters.

In Chapter Three, strain data measured in the field is used to infer reasonable values describing the location, size and material properties of a small oil reservoir. This reservoir is located at

a depth of approximately 500 meters and is stressed by pressurization by an injection well, causing elastic deformation which is measured by strainmeters at a shallower depth of 30 meters.

In Chapter Four, data from a second field test is analyzed and compared to the conventional approach of using in-situ pressure transducers to calibrate a reservoir model. While including both strain and pressure data yields lower uncertainties, we find that strain measurements alone predict a very similar set of reservoir geometries (ie location, size) and material properties.

In Chapter Five, both pressure and strain are used to infer fine details about the range of possible geometries of a subsurface formation, and these detailed models are used to accurately predict geomechanical system behavior under later pumping conditions.

Chapter 1

Development of a cloud computing framework for model calibration and decision support in the geosciences

1.1 Abstract

The solutions to many problems in the geosciences involve a model calibration or parameter estimation step, where numerical simulations of a physical process are compared to measured data to infer the value of some parameter that is difficult to observe directly. This becomes challenging when the necessary physical simulations are computationally intensive. To help overcome these challenges, a cloud computing tool has been developed for model calibration and decision support called Distributed Multiobjective Stochastic Optimization Framework (DMSOF), which runs several commonly used inversion algorithms in parallel allowing them to share results and thus explore the parameter space more effectively. This framework also includes a job scheduler that uses cloud services to distribute computational effort over multiple disparate high-performance and high-throughput computing resources, allowing many forward model evaluations to be run simulta-

neously. The code uses a modular, object-oriented approach that allows other researchers to readily incorporate their own inverse algorithms or forward models. This inversion approach is tested using several benchmark inversion problems as well as some more familiar geoscience inverse problems.

1.2 Introduction

Many problems in geoscience and environmental engineering involve some form of parameter estimation or model calibration. This can require applying computationally intensive physical simulations to a set of possible geologic models, and comparing each set of simulation results with real world field observations. These comparisons allow one to determine which geologic models best match available data, and to iteratively build an improved set of geologic models until the full range of models that explain the observable data have been identified.

As no analytical solution or multiphysics partial differential equation solver can perfectly represent the full complexity of a real-world environmental system, whichever analysis or software chosen will necessarily introduce some conceptual model error by simplifying or neglecting a physical process, approximating a physical process at a lower spatial or temporal resolution, or assuming some level of homogeneity over a heterogeneous domain [45].

In an inversion problem with multiple data-fitting objectives, this conceptual model error often causes a non-uniqueness or trade-off issue such that a compromise can be made to fit one dataset or another relatively well, but no single simulation result can explain the various datasets simultaneously. A common solution to this problem is to choose a set of weighting coefficients that allow one to sum the data-model misfits into a singular value representing the relative quality of each geologic model, and to then minimize this value using single-objective inverse methods such as gradient descent [69] or Markov chain Monte Carlo [29]. However, as this weighting approach tends to obscure the extent of the non-uniqueness problem, it is often beneficial to explore the full range of potential optimal data-fitting compromises that could be made in order to evaluate the validity of a conceptual model.

Inversion techniques are available that have multi-objective variants designed to explore trade-off regions, including Markov chain Monte Carlo (MCMC) [62], genetic algorithms [16, 71], simulated annealing [53], neural networks [54] and particle swarm optimization [55]. Each technique has strengths and weaknesses, with some being ideal for the properties of a particular inverse problem

while others might fail to converge, or converge quickly to an incorrect solution. In order to converge properly, many of these methods also require a long sequence of simulations run in series, rather than many simulations run simultaneously in parallel. As this may not be feasible when the runtime of the simulation is very long, it is often more beneficial to develop surrogate models, computationally cheap functions that approximate the response of a more detailed model. Rather than computing a long-runtime simulation and comparing it to measured data, a surrogate model takes the set of long-runtime simulations that have already been completed and uses interpolation to approximate the data-model misfit at any given point in the parameter space. This faster surrogate model approach makes more traditional inversions such as MCMC feasible for computationally intensive inversions.

In this paper, a model calibration and job scheduling framework is developed to run computationally-intensive simulations in massive parallel in a heterogeneous, high-throughput computing environment. Multiple inversion methods can be run in parallel, and results can be shared between them to allow their various strengths to reinforce one another. After thoroughly exploring the trade-offs between objectives, surrogate models are then developed based on these simulation results and used to run a second, much faster inversion which can run for a long enough series of iterations to assess convergence and uncertainty according to conventional metrics.

1.2.1 Mapping of Parameter Space to Objective Space

The inversion process can be visualized as searching an n -dimensional parameter space by investigating a series of points within the space according to a particular search strategy. Each point in the search represents a unique geologic model which must be run through a physics simulation so that the simulation results may be compared to observed data. Since n is the number of unknown parameters, the size of the space being searched increases exponentially with respect to n . At each step of this search, each model is evaluated according to a vector of m data misfits (often the L_2 norm between measured and predicted data), where m is the number of data-fitting objectives. Some search strategies are designed for single-objective optimization, and therefore require a single value rather than a vector of data misfits. By multiplying each data misfit (E_i) by a weighting coefficient (ω_i) and then summing each term to compute a single value

$$E = \sum_i^m E_i \omega_i \quad (1.1)$$

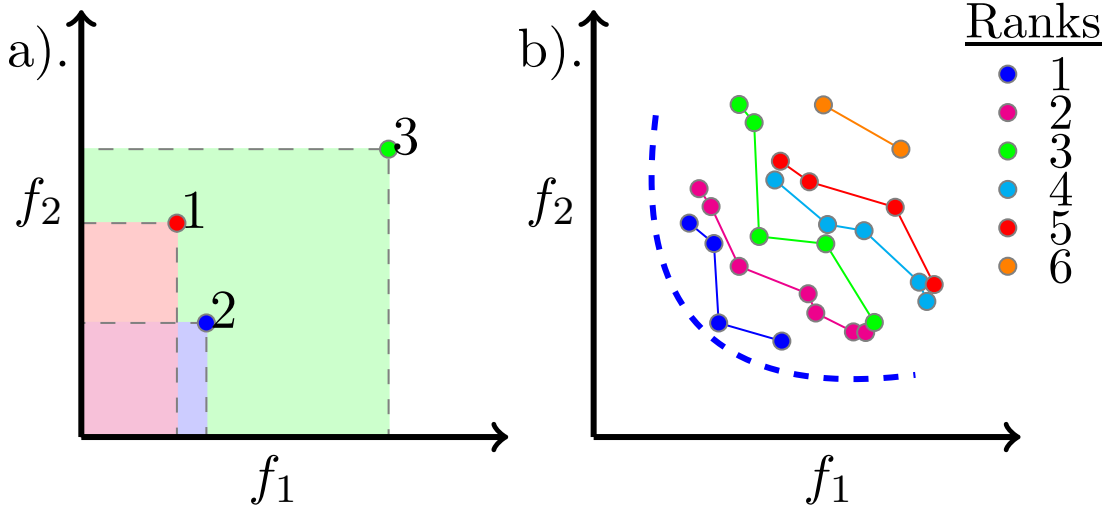


Figure 1.1: Illustration of Pareto optimality. a.) Results from three solutions plotted with respect to objective functions f_1 and f_2 . Colored rectangles indicate region of lower Pareto rank. b.) Pareto ranking used to characterize model-data misfit for a set of 26 simulations with different parameter sets. Blue dashed line represents the 'true' Pareto front.

quantifying the data-model misfit (E). These search strategies can readily be used for multi-objective optimization. These weighting parameters ω_i represent a normalizing factor related to the data error as well as an implicit assumption about the information value of each dataset. However, choosing these weighting parameters can be arbitrary, and can unduly influence the convergence behavior of the inversion.

An alternative approach is to use the concept of Pareto optimality, which allows one to consider each objective's data-model misfit separately. A solution is Pareto optimal when compared with other solutions if no other solution in the group is superior to it in terms of all of the objective functions. For example, in the multiobjective minimization problem presented in Figure 1.1a, solution 1 has a lower value in terms of objective f_1 , but a higher value in objective f_2 . Solution 2, by contrast, has a lower value in terms of objective f_2 , but a higher value in objective f_1 . Therefore neither solution is superior to the other in the Pareto optimality sense. However, solution 3 is not Pareto optimal, because both solutions 1 and 2 have lower f_1 and f_2 values. The group of solutions that are not dominated by any other solution are denoted rank 1, solutions that are dominated only by rank 1 solutions are denoted rank 2, and so on. In Figure 1.1b, several ranks are identified and color-coded. The rank 1 solutions represent the best approximation of the Pareto front.

As the number of unknown parameters increases, the size of the n -dimensional parameter

space increases dramatically, and therefore the number of simulations necessary to thoroughly explore the parameter space quickly becomes impractical. Overcoming this 'curse of dimensionality' requires the integration of high performance computing resources with an efficient search strategy. Since each search strategy has its own advantages and disadvantages, one often cannot know which method is ideal for solving a given inverse problem until after one has solved it.

Each search strategy strikes a balance between 'exploration' and 'exploitation'. Exploration refers to searching the entire parameter space as thoroughly as possible in order to find promising new models. Exploitation refers to searching a narrower space around the most promising models identified so far, refining and improving upon them. By coupling several of these methods into one model calibration process, the parameter space can be explored thoroughly while the most promising models can be carefully refined.

1.2.2 Inversion Algorithms

This inversion framework uses a modular programming approach and implements a variety of standard inversion algorithms, which are described below.

1.2.2.1 Monte Carlo

At each iteration, Monte Carlo uses the prior model to generate a set of parameters to be evaluated. The prior model is a statistical model representing any initial knowledge about the parameter value. It may be a uniform distribution over the range of physically meaningful values (i.e., porosity between 0% and 100%), or a gaussian distribution centered about what is judged to be the most likely value, with a variance representative of uncertainty in site conditions or otherwise judged to be large enough to be sure of bounding the true value. The Monte Carlo approach is resilient against premature convergence because it is not biased by results from previous simulations, but also slow to converge because it does not learn from previous simulations to search more thoroughly around areas of interest.

1.2.2.2 Gradient Descent

At the initial iteration, gradient descent [69] uses the prior model to generate a random point within the parameter space. A set of points adjacent to this point is then evaluated in order to

estimate the Jacobian of the forward model. The Jacobian [Equation 1.2] is made up of the partial derivatives of each of the n parameters with respect to each the m objectives,

$$\mathbf{J} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \frac{\partial f_1}{\partial x_3} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \frac{\partial f_2}{\partial x_3} & \cdots & \frac{\partial f_2}{\partial x_n} \\ \frac{\partial f_3}{\partial x_1} & \frac{\partial f_3}{\partial x_2} & \frac{\partial f_3}{\partial x_3} & \cdots & \frac{\partial f_3}{\partial x_n} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \frac{\partial f_m}{\partial x_1} & \frac{\partial f_m}{\partial x_2} & \frac{\partial f_m}{\partial x_3} & \cdots & \frac{\partial f_m}{\partial x_n} \end{bmatrix}, \quad (1.2)$$

where each gradient is estimated using a central difference Taylor approximation,

$$\frac{\partial f_i}{\partial x_j} \approx \frac{f_i(\mathbf{x} + \delta_j) - f_i(\mathbf{x})}{(\mathbf{x} + \delta_j) - \mathbf{x}}, \quad (1.3)$$

as the reference model is perturbed by a small value $+\delta$ for each of the n parameters. The Jacobian is then used to determine the down-gradient direction, and an appropriate step is taken in that direction. This process continues until the gradient is below a given threshold, indicating a local minimum has been found. Once local convergence has been detected, the prior model is again used to generate a random point within the parameter space, and the gradient descent process continues. By taking a series of steps directly down gradient, gradient descent can very quickly find a nearby local minimum. However, for inverse problems where multiple local minimums exist, gradient descent often misses the global minimum. By repeating the gradient descent process using a number of different, random starting points within the parameter space, many local minimums can be found making it more likely that the true global minimum is identified. This method can also be very computationally intensive for high-dimensionality problems, as computing the Jacobian requires $n + 1$ function evaluations per step. It is also up to the user to select the step size, as well as the finite difference offset values (δ).

1.2.2.3 Markov Chain Monte Carlo

The Markov Chain Monte Carlo (MCMC) algorithm also implicitly moves down-gradient, but in a stochastic fashion. At each iteration, the MCMC chain takes a random step in the parameter space, compares the error at the new step location to the error at the initial location, and probabilistically selects the superior model. If the newer model has a lower error, it is accepted. If the

new model has a higher error, it is accepted with a probability proportional to the ratio between the old and new errors (Metropolis-Hastings criteria, [42]). The algorithm then takes another random step in the parameter space, and continues. Therefore, an MCMC chain is capable of occasionally moving 'up-gradient', allowing it to leave a small local minima and explore the remainder of the parameter space.

The MCMC accept/reject decision requires a single error value to compare each step to its predecessor, whereas a complex multi-objective inversion problem may involve a separate data-model misfit for each instrument installed in the field, potentially meaning dozens or hundreds of objectives. Therefore in this implementation of MCMC, two methods are included to deal with multiobjective optimization problems. One method uses the weighted sums approach (Equation 1.1), where each error is weighted by the estimated noise in the data [40]. The other method sorts all simulations run so far by Pareto rank, and uses the respective ranks of the two MCMC candidates as the basis for the accept/reject decision. Examples of these two methods are shown in Figure 1.13.

1.2.2.4 Genetic Algorithms

Genetic algorithms [16, 71, 61] select the most promising models identified so far, recombine them randomly using a process called crossover, and then perturb the recombined models using a mutation operator. This heuristic approach to parameter estimation mimics the way evolution naturally selects the most capable members of a population for reproduction, gradually shifting the entire population towards a more robust, adaptive gene pool. Most genetic algorithms are designed for single-objective optimization, using a single value to represent a models 'fitness', which regulates how the selection, crossover and mutation operators function. However, several multi-objective genetic algorithms exist that use the concept of Pareto optimality (see Section 1.2.1, Figure 1.1) to govern these operators.

Similar to populations in nature, genetic algorithms require genetic diversity to perform well. The Non-dominated Sorting Genetic Algorithm (NSGA-II) [16] encourages a diverse set of models by imposing a penalty on simulations that are too similar to each other in terms of their position along the Pareto front. It accomplishes this by identifying the models nearest neighbor on either side, along each of the m objective space dimensions. It then computes the 'crowding distance', or average distance to these $2m$ neighbors, and uses this distance as a criteria for the selection operator. This prevents the population from crowding too closely around the first Pareto

optimal model that presents itself, but instead investigating the entire Pareto front as thoroughly and evenly as possible.

The Strength Pareto Evolutionary Algorithm (SPEA2) algorithm [72, 71] uses the concept of *elitism* to ensure that Pareto optimal solutions are not inadvertently rejected due to the stochastic nature of the selection process. The algorithm maintains an external population of all non-dominated solutions that have been identified so far. At each generation, this external population is combined with the current population and included in the selection operator. A Pareto rank based fitness is assigned, and used in the selection, crossover and mutation operators. Similar to NSGA-II, a k^{th} nearest neighbor approach is used to penalize clustering along the Pareto front.

1.2.2.5 Space-filling algorithms

Space-filling algorithms are designed to explore the parameter space as uniformly as possible, without referring to how previous simulation results mapped to the objective space. This helps avoid issues of premature convergence, or neglecting large volumes of the parameter space that might contain the true global minimum.

The simplest space-filling approach is called a grid search, where the parameter space is subdivided into a uniform grid and each combination of parameters is simulated. While this brute-force approach is simple to implement and minimizes the risk of premature convergence, it can be computationally infeasible for high-dimensionality parameter spaces or for forward models with a long runtime.

The Latin hypercube [52, 31] is an n -dimensional generalization of the Latin square, a combinatorial puzzle where n different symbols are arranged in a table such that each symbol appears in exactly one column and exactly one row. The Latin hypercube inversion method subdivides the parameter space into a regular grid of arbitrary grid size, and then samples from this grid such that each parameter is allowed to vary with respect to all other parameters. Latin hypercube methods are commonly used in experimental design where a large number of unknowns are being investigated using a limited number of experiments or simulations.

Centroidal Voronoi tessellation [46] uses computational geometry to produce an arbitrary number of uniformly distributed points throughout the parameter space. This approach is mathematically exact and efficient to compute in low-dimensionality inverse problems, but can be computationally challenging in high-dimensionality parameter spaces, it can

1	2	3	4
2	4	1	3
3	1	4	2
4	3	2	1

Figure 1.2: Latin Square

be more efficient to relax this precision by simply generating a large number of proposed models, comparing their distances with all existing models, and selecting the one with the greatest distance to its nearest neighbor.

1.2.2.6 Simulated Annealing

The simulated annealing inversion algorithm is a variant of MCMC, which begins by making accept/reject decisions based on a statistical model that represents any prior knowledge of the system, and slowly shifts to sampling from the more data-driven likelihood model [39, 25].

This method uses a weighting approach [Eq 1.4] using a coefficient, T or “temperature”, to control how quickly information from the likelihood model is introduced into the inversion. When T is high, the prior model $p(\theta)$ is weighted more heavily, and as T approaches zero, the likelihood function $\mathcal{L}(\theta)$ begins to guide the inversion more.

$$p(\theta|d) = p(\theta)^T + \mathcal{L}(\theta)^{1-T} \quad (1.4)$$

The technique is named after the metallurgical process of annealing, where a metal is heated and then cooled slowly in order to avoid the formation of defects in the crystalline structure of the metal. Similarly, the weighting parameter T must be reduced slowly in order to avoid latching on to a local minimum and overly-exploiting it to the exclusion of any other unexplored optima, analogous to a crystalline defect.

1.3 Methods

1.3.1 Computational Workflow

A computational workflow was developed which implements a wide range of inversion algorithms with a common format and namespace, allowing the necessary forward model evaluations

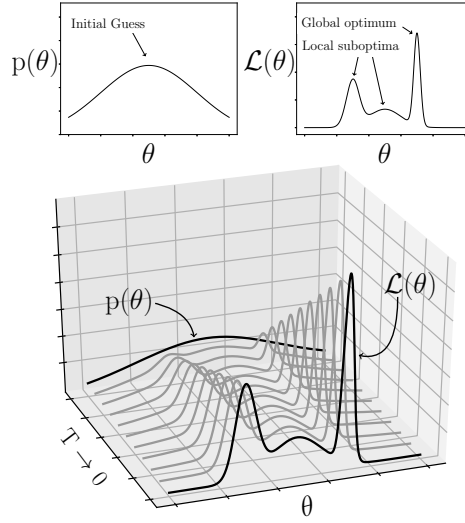


Figure 1.3: The simulated annealing process is illustrated. The prior model ($p(\theta)$) is shown at high temperatures (T) while the likelihood model ($\mathcal{L}(\theta)$) begins to dominate at lower temperatures.

to be run in a highly distributed, heterogeneous computing environment while storing results in the centralized namespace for later use.

This code takes advantage of the Amazon Web Services (AWS) cloud computing service. An Elastic Compute Cloud (EC2) instance is used as the head node, which houses a MySQL relational database [Figure 1.4] on a solid state drive providing high read-write speeds and non-volatile storage for information about the parameterization of the inverse problem, the prior models for each parameter, and various other pieces of information specific to each inversion algorithm (step sizes, population sizes, number of MCMC chains, etc).

The inversion code has each been implemented in Python using a modular, object-oriented approach that mirrors the structure of the database [Figure 1.4] with one class definition per database table. Classes and tables are synchronized using SQLAlchemy, a popular object-relational mapping software. This approach allows the use of external Python libraries such as numpy and scipy (math and statistics functions), matplotlib (graphics and visualization), and boto (AWS file transfer and job management).

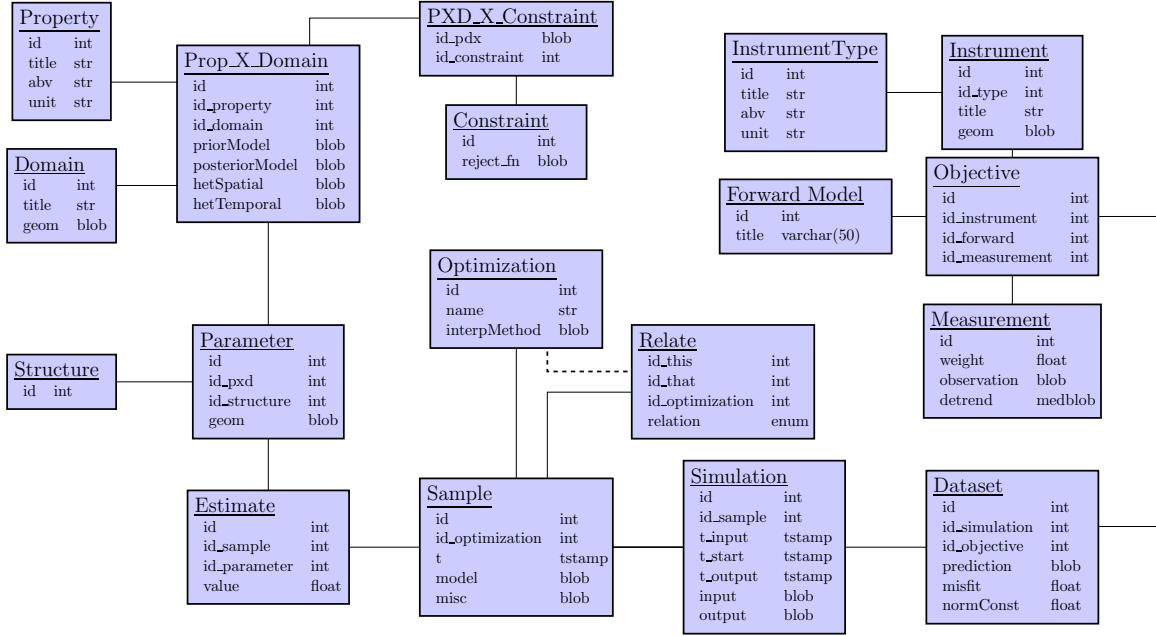


Figure 1.4: Entity-Relationship diagram for the inversion database. Blue boxes indicate tables in the database, while lines indicate the cardinality relationships between tables.

An abstract class (**Optimization**) contains general definitions for all the basic operations that any inversion algorithm must be able to perform, while each specific inversion algorithm inherits these definitions and adds some distinct code detailing how these operations are to be performed.

The physical properties of interest (i.e, density, permeability, compressibility) are listed in the **Property** table, and the physical domains of interest (i.e., distinct rock layers) are listed in the **Domain** table. The **Prop_X_Domain** table defines a particular **Property** of a particular **Domain** using a statistical distribution, the prior model. Two or more **Prop_X_Domains** may be related by the **Constraint** table, which contains a rejection function that tests whether a particular combination of parameter values is feasible for the forward model. This allows one to define the feasible parameter space as any arbitrary shape rather than simply as an n -dimensional hypercube. The **Parameter** table lists the unknown parameters in the particular inverse problem. It is conceptually similar to the **Prop_X_Domain** table, but remains a distinct table to allow for the special case of a tomography or imaging problem, where many parameters may describe a single pixel or voxel of a given structure but all have a common underlying statistical distribution. Where differing sets of prior information (statistical distributions) exist are available for different regions in the field site, they can then be assigned to the corresponding **Domains**.

The types of instruments installed in the field (i.e., pressure sensor, temperature probe, humidity etc) are listed in the **InstrumentType** table while specific instruments are listed in the **Instrument** table with their name and location. In general, each instrument corresponds to a distinct entry in the **Objective** table, however when there are multiple forward models available to attempt to simulate the behavior of an **Instrument**, there may be multiple **Objective** entries for a single **Instrument**. An **Objective** may relate to the **Measurement** table, where synthetic measurement is stored or where actual measurements from the field are periodically uploaded.

Once these database tables are populated, the inversion process can begin. The basic function of the **Optimization** object is to create an entry in the **Sample** table, create one entry in the **Estimate** table for each **Parameter**, use these estimates to build a set of input files in a format appropriate for the physics solver being used, then create an entry in the **Simulation** table and stores the input files in this table. The input files are then uploaded to the Amazon Simple Storage Service (S3) where it can be retrieved by any remote machine with both the appropriate credentials and the S3 object key, a string of characters which uniquely identifies that particular set of input files. The object key is then posted on the Amazon Simple Queue Service (SQS), another AWS service which maintains the integrity of the simulation queue. A variety of remote machines can then periodically check this queue to see if any new input files are available for evaluation, without the risk that any two compute nodes will be served the same S3 object key and thus duplicating one another's work or causing table locking by attempting to modify the same database entry simultaneously. Each compute node then uses their unique S3 object key to download the appropriate set of input files, uses them to run a (perhaps computationally intensive) physical simulation, and performs any necessary post-processing to extract the expected sensor responses from the larger output file. Since sensors are generally relatively sparse compared to the scope of the simulated region, the post-processed summary file is generally much smaller. The raw simulation output files are then uploaded to long-term storage in an appropriate S3 bucket (or discarded if the storage cost is judged prohibitive), while the simulated sensor data file is sent to short-term storage in a separate S3 bucket so it may be later downloaded by the head node. The head node then stores the simulated sensor predictions in the **Dataset** table. The prediction is compared to the field data stored in the **Measurement** table to arrive at a prediction-observation misfit.

Once prediction-observation misfits are available, each specific implementation of the **Optimization** object uses a unique strategy to interpret data-model misfits and assemble progressively

better input files. These strategies are described in detail in the following section. Many of these strategies involve relating one `Sample` to another, so these relationships are defined in the `Relate` table.

1.3.2 Implementation of Inverse Methods

This code implements each of the inverse methods in Section 1.2.2, modifying them for the multi-objective case as well as allowing them to run in parallel and incorporate one another’s simulation results. The space-filling algorithm begins by filling the parameter space as evenly as possible, then shifts to filling in the gaps left by other methods. The Markov chain Monte Carlo and gradient descent methods have options to use either the Pareto rank or weighted sum as the accept/reject criteria. The genetic algorithm implemented is a variant of NSGA-II which draws simulation results from other methods into its population at each iteration, and also places an optional clustering penalty in the parameter space to avoid run too many simulations that are too similar to each other. Similarly, the simulated annealing method includes an adjustable clustering penalty. This combination of methods is an effective way to quickly and efficiently explore a parameter space with a long-runtime forward model.

1.3.3 Interpretation of simulation results

Inversion results can be visualized by plotting each measured dataset on the same plot as the simulated results [Figure 1.14a]. Line colors can be used to highlight models with the best data-model misfits, allowing one to make a preliminary evaluation of whether any simulations have yet been found that fit the data well. When the dimensionality of the parameter space is relatively low, results can be visualized by calculating the data-model misfit for each simulation and plotting them as a function of the parameter values [Figures 1.9,1.11,1.13,1.14] in order to evaluate whether the shape of the error surface is beginning to resolve.

When a sufficient set of samples of the parameter space is available and some stopping criteria is reached (see Section 1.3.3.2), statistical inference methods can be used to estimate the values of the model parameters as well as quantify the uncertainty of those estimates. The Bayesian approach to statistical inference is often used to make this estimate. Bayesian inference begins with a prior probability distribution $p(\theta)$, a statistical model constructed using any prior knowledge about

the parameters in question. This prior belief is then combined with new information using Bayes theorem,

$$p(\theta|d) = p(\theta) \frac{p(d|\theta)}{p(d)}. \quad (1.5)$$

Bayes theorem relates the posterior distribution $p(\theta|d)$ to the prior distribution $p(\theta)$ via the likelihood function $\mathcal{L}(\theta|d)$. Put another way, Bayes theorem says that our belief in a given model after accounting for new data is proportional to our belief before accounting for the data, and to the degree of compatibility between the data and the model. This compatibility is often quantified using the probability density function of a Gaussian distribution

$$\mathcal{L}(\theta|d) = \frac{p(d|\theta)}{p(d)} \propto \frac{1}{\sqrt{2\pi\sigma_n^2}} \exp\left(-\frac{[f(\theta) - d]^2}{2\sigma_n^2}\right), \quad (1.6)$$

where the the measured data, d is compared to $f(\theta)$, a function or simulation representing what the numerical physics simulation predicts the data would be for the model θ . The difference between the simulated and measured data is then compared to the variance σ_n , which represents the random measurement error inherent in the measured data.

1.3.3.1 Parameter Space Integration

When integrated over the parameter space, these probability densities should be equal to one [Eq 1.7].

$$\int_V p(\theta|d) dV = 1 \quad (1.7)$$

However, because they are scattered at semi-random locations throughout the parameter space, performing this integration can be challenging especially in high-dimensionality parameter spaces. Several methods were implemented for performing this integration [Figure 1.5]. The first method uses a Python computational geometry library to perform a Voronoi tessellation on the set of evaluated points in the parameter space, thus assembling a convex polygon about each one that defines its nearest-neighbor space. The `qhull` library is then used to compute the area (2d), volume (3d), or hypervolume (4d+) of each polygon. Where the polygon extends beyond the feasible space selected, the polygon is first truncated by computing the intersection of the polygon with the

planes defining the feasible space. While generally only very low-probability points near the edge of the feasible space require truncation, many have enormous volumes which can skew the numerical integration.

The Voronoi tessellation, truncation and convex hull computations become impractical at high dimensionalities, so a less exact but more efficient method has been implemented for estimating the hypervolume that each point represents [Figure 1.5]. Using a Python library function, the Gaussian KDE (kernel density estimation) of the set of points is computed, and the hypervolume is treated as being inversely proportional to the point density.

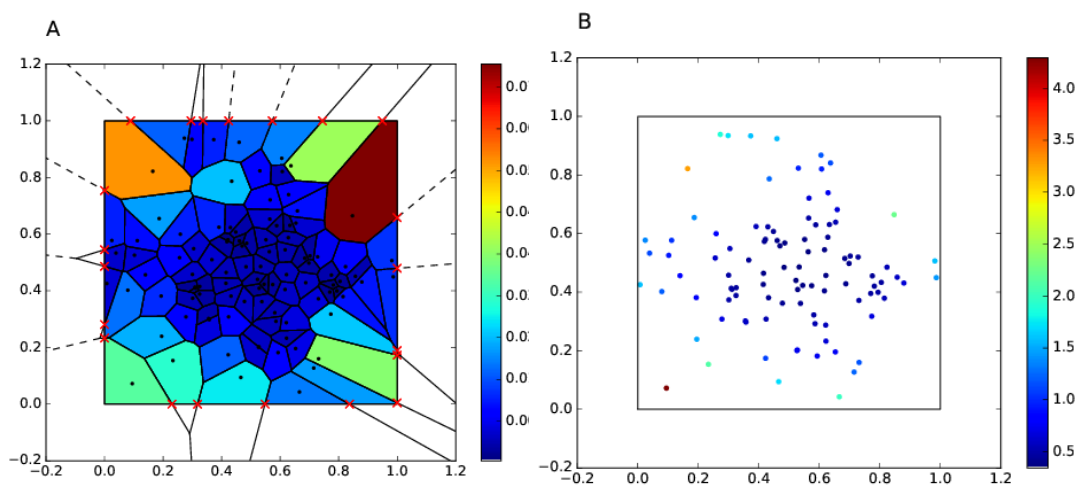


Figure 1.5: Illustration of two methods of estimating volume of the representative parameter space of each simulation. Color represents sparsity, or the volume of empty space surrounding each point. The (A) Voronoi tessellation method computes intersections between Voronoi polygons and the edges of the parameter space, and uses these intersection points to compute exact volumes. The (B) Gaussian KDE method instead fits a uni-modal Gaussian distribution to the points.

For dealing with the very large parameter spaces necessary for tomography and imaging problems, a third integration method [Figure 1.6] has been implemented based on MCMC (see Section 1.2.2.3). While MCMC can be effective for optimizing short-runtime forward models, such as the benchmark functions shown in Section 1.4.1, its sequential nature makes it difficult to parallelize effectively. This makes it impractical to run enough MCMC steps to optimize a computationally-intensive forward model. However, when run for a sufficient number of iterations, MCMC has strong convergence properties that are useful in approximating high-dimensional integrals [26].

Therefore, the set of parameter values and likelihoods (Equation 1.6) are exported to a

compute node and the linear interpolation of this parameter to likelihood mapping is used as a surrogate objective function. A random point in the parameter space is first selected, and interpolation is used to estimate the likelihood at that location. A random step is then taken, interpolation is again used to estimate the likelihood at that point, and the accept/reject decision is made based on these respective likelihoods.

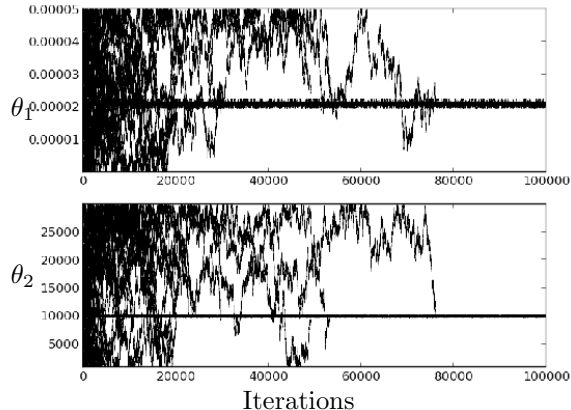


Figure 1.6: Multi-chain MCMC results are shown for two parameters (θ_1, θ_2) as a function of iteration, using an interpolation-based surrogate function as the fitting objective.

For cases where the interpolation step is many orders of magnitude faster than running the forward model itself, a many-core compute node can be used to run several hundred MCMC chains for enough iterations to converge in a reasonable amount of compute time. Running many chains allows one to take advantage of parallelization, as well as ensuring that a unique solution is found.

1.3.3.2 Convergence Criteria

Many of the most effective parameter space sampling algorithms are heuristic, and therefore lack the statistical rigor necessary to objectively determine when convergence has been reached. The MCMC approach has a broad range of effective convergence diagnostics available in the literature [23, 24, 30], however these diagnostics require many iterations of MCMC in order to meet the convergence criteria. Using interpolation as a surrogate objective function, many MCMC chains can be run for many iterations until these diagnostic convergence criteria are met. This allows one to sample the posterior distribution and develop statistical interpretations of the inverse problem.

The challenge then is deciding whether a given sampling of the parameter space is sufficient

to represent the real-world system. This problem can be approached using the bootstrap method, by using a series of subsamples of some number n of the simulations and run the analysis using these n simulation results only. The mean and variance of the resulting estimates and uncertainties can then be computed. If the variance is low, this indicates that any n of the simulations will produce the same results. By performing this for many values of n , one can examine the sensitivity of the results to the number of simulations and therefore decide whether additional simulations are likely to improve understanding of the system.

1.4 Results and Discussion

A series of optimizations are run using benchmark functions common in the inversion literature. Field data from a well-studied pump test is then used to compare this methods parameter estimates and uncertainties to previously published analyses. An inversion is performed on synthetic pump test data with varying levels of random numerical error introduced, and the same inversion is then performed introducing a systematic error in the form of a flawed conceptual model.

1.4.1 Benchmark Functions

The strengths and weaknesses of gradient descent and MCMC are first compared using the sphere, Ackley, and Rosenbrock functions. Genetic algorithms are then compared to the gradient-sensitive methods using the dropwave function. The Binh-Korn function, a multi-objective inverse problem, is then used to compare the weighting coefficient approach to the Pareto approach. For simplicity, each test problem is presented as a function f of a vector of input parameters \mathbf{x} . Where the global minimum is explicitly defined, it is presented as \mathbf{x}_0 .

1.4.1.1 Sphere

The first and simplest test function is the sphere function [Eq 1.8]. This function is ideal for gradient-based approaches because it contains a single minimum, and the down-gradient direction is oriented toward the minimum at every point in the parameter space. In Figure 1.7, the Gradient Descent chains move in a straight line directly toward the global minimum while the MCMC chains meander back and forth in the correct direction, requiring more steps to reach the minimum. However, once the MCMC chains arrive at the global minimum, they continue a random walk around

the vicinity of the minimum and provide one with better information about uncertainty.

$$f(\mathbf{x}) = \sum_{i=1}^n (x_i - x_{0,i})^2 \quad (1.8)$$

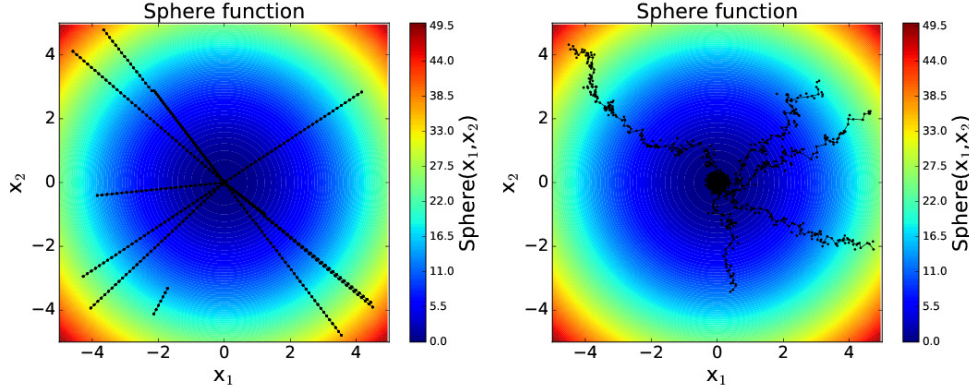


Figure 1.7: Comparison of Gradient Descent and Markov chain Monte Carlo for the sphere test function. Color indicates the value of the objective function while black dots indicate paths taken by Gradient Descent and Markov chain Monte Carlo.

1.4.1.2 Ackley

The Ackley function [1] contains many local minima, and one central global minimum.

$$f(\mathbf{x}) = -20 \cdot \exp\left(-\frac{1}{5}\sqrt{x_1^2 + x_2^2}\right) - \exp\left(\frac{1}{2}[\cos(2\pi x_1) + \cos(2\pi x_2)]\right) + e + 20 \quad (1.9)$$

This function [Eq 1.9] shows an advantage of MCMC over Gradient Descent, as each Gradient Descent chain quickly falls into the nearest local minima and then proceeds to solve for that particular global minimum more and more precisely. In this case, only a chain that happens to begin very close to the global minimum will ever reach the global minimum. By contrast, MCMC chains can spend many iterations exploring near a particular global minimum, but will eventually leave it to explore other minima. This process may take many iterations, and is largely controlled by the user-chosen step size. If the step size is too small, then climbing out of a local minima may require many 'unlikely' steps in a row, and therefore require many iterations before the chain climbs entirely out of the local minima. If the step size is too large, MCMC may leap from one local minima to

another without exploring inside the local minima.

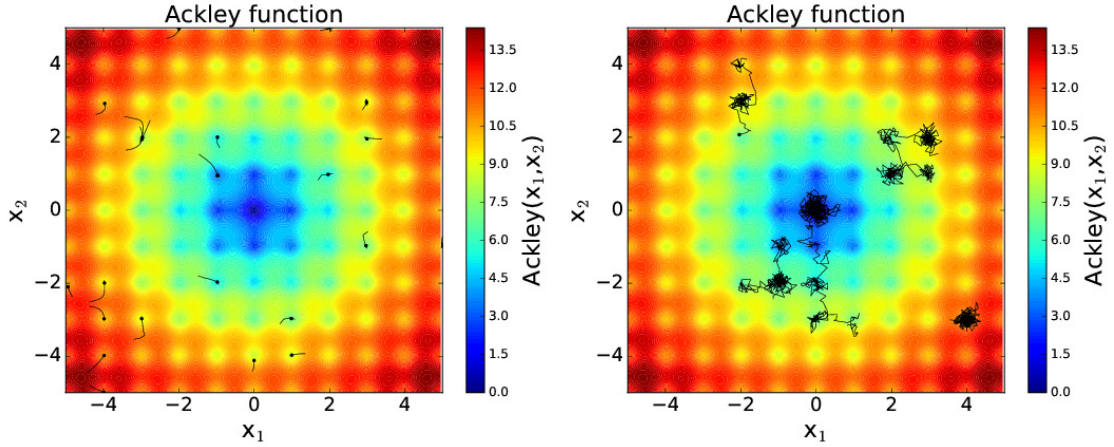


Figure 1.8: Comparison of Gradient Descent and Markov Chain Monte Carlo for the Ackley test function. Color indicates the value of the objective function while black dots indicate paths taken by Gradient Descent and Markov chain Monte Carlo.

1.4.1.3 Rosenbrock

The Rosenbrock function [Eq 1.10]

$$f(\mathbf{x}) = \sum_{i=1}^{n-1} \left[100 (x_{i+1} - x_i^2)^2 + (x_i - 1)^2 \right] \quad (1.10)$$

contains a large, steep trough-shaped feature with the axis of the trough oriented parallel to the y-axis. Near the origin is a long, narrow, roughly banana-shaped valley. The global minima is inside this at, at $\mathbf{x} = (1, 1)$. Because of how sharply defined the valley is, the gradient is very sharp and therefore rarely points in the actual direction of the global minima. As a result, most gradient-based optimizations quickly find the valley but then zig-zag back and forth inside it, requiring many steps to reach the global minima and only minimally exploring the overall shape of the valley. By contrast, MCMC does tend to find the global minima as well as define the overall shape of the valley.

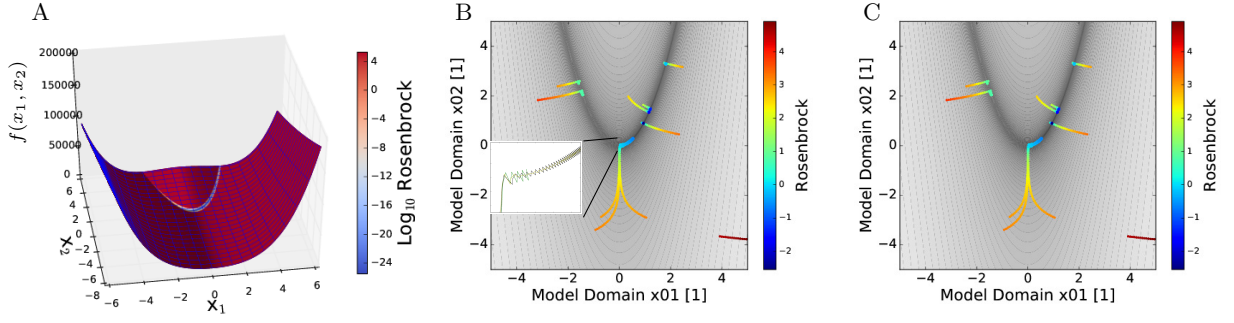


Figure 1.9: A: Rosenbrock function is shown in 3-dimensional space. The height of the surface (z-axis value) represents the output of the Rosenbrock function, while the color of the surface represents the base-10 Log of the Rosenbrock function. This highlights the narrowness and sharpness of the banana-shaped valley characteristic of the Rosenbrock test function. B: 10 chains of gradient descent, using a total of 5156 function evaluations. Zoomed inset shows zig-zag pattern characteristic of gradient descent. C: 10 chains of Markov chain Monte Carlo, using a total of 3897 function evaluations.

1.4.1.4 Drop Wave

The drop wave function [Eq 1.11]

$$f(\mathbf{x}) = -\frac{1 + \cos\left(12\sqrt{x_1^2 + x_2^2}\right)}{0.5(x_1^2 + x_2^2) + 2} \quad (1.11)$$

is reminiscent of a drop of water on a pond, where sinusoidal waves expand out radially and are attenuated by distance. The global minimum is a small region centered around the origin ($x = 0, y = 0$), with concentric rings of alternating minima and maxima. This makes the inverse problem very challenging since the true global minima occupies a very small region of the parameter space compared to the other minima, and is only marginally lower in terms of the objective. Additionally, the true global minima is not connected to the other minima, and therefore gradient-sensitive methods tend to experience premature convergence and become trapped in the first ring that they encounter.

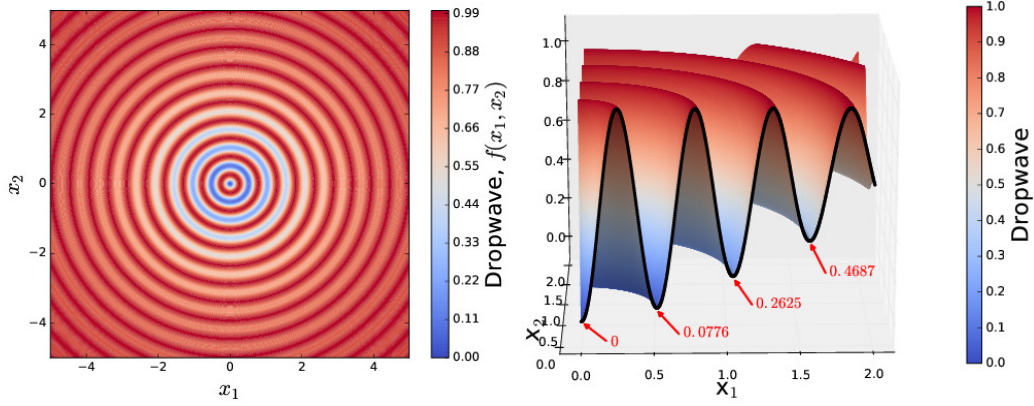


Figure 1.10: Visualization of Dropwave function.

The gradient-descent, MCMC and NSGA-II methods [Figure 1.11] were run for approximately 8,000 simulations each. The gradient descent chains quickly converge to the nearest ring and do not explore any further. The MCMC chains perform better, accomplishing a marginal amount of lateral exploration along their nearest ring. However, the genetic algorithm performs much better, exploring the space much more thoroughly and finding the global minimum.

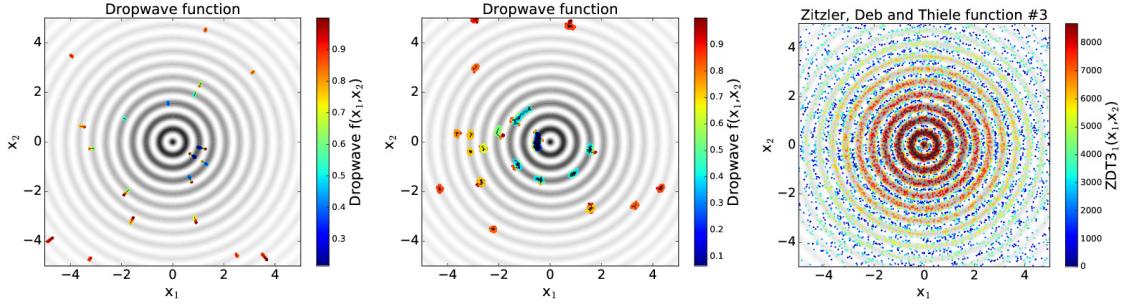


Figure 1.11: Comparison of (A) Gradient Descent, (B) Markov Chain Monte Carlo, and (C) Genetic Algorithm for the Drop wave test function. Grayscale color indicates the value of the objective function while colored dots indicate paths taken by Gradient Descent and Markov chain Monte Carlo.

1.4.1.5 Binh-Korn

Multiobjective inversion problems are introduced using the Binh-Korn function [6], which attempts to optimize two simple quadratic equations [Eqs 1.12, 1.13] with different vertices.

$$f_1(\mathbf{x}) = 4x_1^2 + 4x_2^2 \quad (1.12)$$

$$f_2(\mathbf{x}) = (x_1 - 5)^2 + (x_2 - 5)^2 \quad (1.13)$$

Since these functions have different local minima, no unique solution exists to this problem. The goal therefore is to approximate the Pareto front and identify all solutions which lie between the two optima.

An MCMC optimization was run using three different criteria as the minimized objective. The first two use the weighted sum approach [Equation 1.1] to combine the Binh-Korn function's two objectives. Two different sets of weighting coefficients are used, where the weight ratio favors one objective over the other by a factor of 4. In these two cases, the MCMC chains converge on a single point along the Pareto front, but do not explore the rest of the Pareto front. A third MCMC optimization is then run using the Pareto rank rather than the weighted sum as the objective to be minimized. In this case, the chains generally approached their nearest intersection with Pareto front, and then tended to do a random walk along the Pareto front. This gives one a much more complete understanding of the range of viable solutions to the Binh-Korn problem.

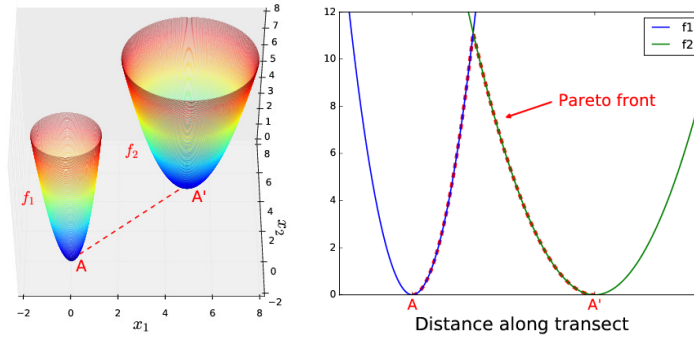


Figure 1.12: Visualization of both Binh-Korn functions, f_1 and f_2 . Local minima are at A and A'. Pareto front represented by red dashed line.

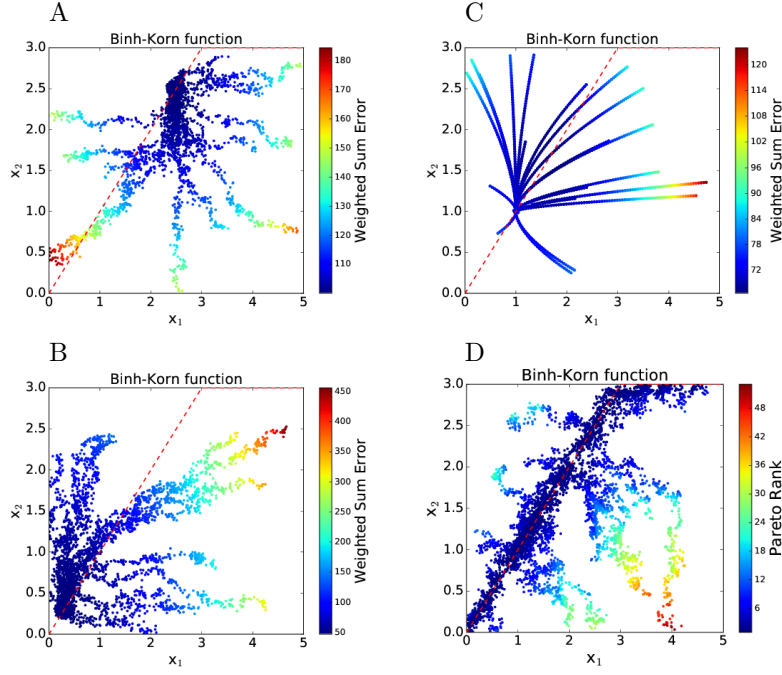


Figure 1.13: Comparison of Markov Chain Monte Carlo (A,B,D) and Gradient Descent (C) results using the weighted sum approach for subfigures A ($\omega_1 : \omega_2 = 0.25$) and B ($\omega_1 : \omega_2 = 4.0$), and C ($\omega_1 : \omega_2 = 1.0$), and the Pareto rank approach for subfigure D. Red dashed lines indicate the true Pareto front.

1.4.2 Groundwater Model: Theis Solution

1.4.2.1 Analysis of Real Pumping Test Data

The Theis solution [58] models the response of the potentiometric surface to a well being pumped at a constant rate. This solution assumes an infinite, homogeneous confined reservoir of uniform thickness, and treats the well as a line source. The model space can be treated as one or two-dimensional due to radial symmetry about the line source. Drawdown s of the potentiometric surface is computed using the Theis equation,

$$s = \frac{Q}{4\pi T} W\left(\frac{r^2 S}{4Tt}\right), \quad (1.14)$$

where Q is the pumping rate out of the well, T is the transmissivity of the formation, S is the storativity of the formation, r is the radial distance from the well, and t is time. The exponential integral function $W(u)$ defines the shape of the drawdown cone around the well, while the pumping

rate and transmissivity control the magnitude of the cone.

$$W(u) = \int_u^\infty \frac{e^{-y}}{y} dy \quad (1.15)$$

$$W(u) \approx -\gamma - \ln(u) + u - \frac{u^2}{2! \cdot 2} + \frac{u^3}{3! \cdot 3} - \dots + (-1)^{n+1} \frac{u^n}{n! \cdot n} \quad (1.16)$$

$$u = \frac{r^2 S}{4Tt} \quad (1.17)$$

In field sites where the Theis assumptions represent a reasonable approximation of the geology, this equation can be used to estimate transmissivity and storativity based on drawdown measurements in an observation well near a pumped well. An example pumping test dataset is drawn from the literature, and an optimization is run on it using MCMC, Gradient Descent, Voronoi-based space-filling, and NSGA-II. This dataset was used by [63] to estimate transmissivity and storativity using a graphical method [36], and later by [41] using a method based on sensitivity analysis. In this analysis the derivatives of the Theis equation with respect to transmissivity and storativity,

$$\frac{\partial s}{\partial T} = -\frac{S}{T} + \frac{Q}{4\pi T^2} \exp\left(-\frac{r^2 S}{4Tt}\right), \quad \frac{\partial s}{\partial S} = -\frac{Q}{4\pi TS} \exp\left(-\frac{r^2 S}{4Tt}\right), \quad (1.18)$$

were used to construct an expression for the gradient of the error function, which was then used to perform gradient descent and estimate the parameters. Taking this analysis one step further, these derivatives are used to construct the uncertainties of the estimates (σ_S, σ_T) made by [63] and [41], and compared to the estimates and uncertainties.

$$\sigma_s^2 = \left(\frac{\partial s}{\partial T}\right)^2 \sigma_T^2, \quad \sigma_s^2 = \left(\frac{\partial s}{\partial S}\right)^2 \sigma_S^2 \quad (1.19)$$

$$\frac{1}{n_m - 1} \sum_{i=1}^{n_m} [s_m - s(T, S, t_i)]^2 = \frac{1}{n_m - 1} \sum_{i=1}^{n_m} \left(-\frac{S}{T} + \frac{Q}{4\pi T^2} \exp\left(-\frac{r^2 S}{4Tt_i}\right)\right)^2 \sigma_T^2 \quad (1.20)$$

$$\frac{1}{n_m - 1} \sum_{i=1}^{n_m} [s_m - s(T, S, t_i)]^2 = \frac{1}{n_m - 1} \sum_{i=1}^{n_m} \left(-\frac{Q}{4\pi TS} \exp\left(-\frac{r^2 S}{4Tt_i}\right)\right)^2 \sigma_S^2 \quad (1.21)$$

These uncertainties [Table 1.1] reflect the gradient of the data-model misfit in the region around a given solution and do not account for any uniqueness in the inverse problem, which is apparent from the range of plausible parameters shown in the map of likelihood [Figure 1.14b].

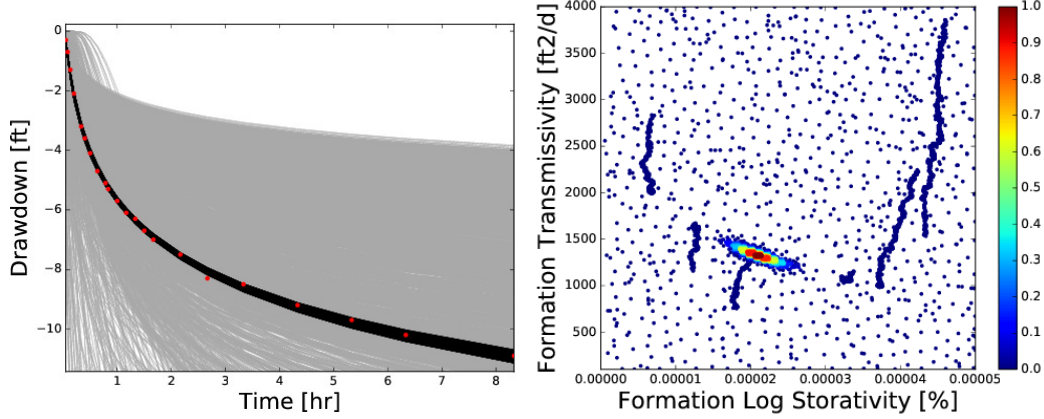


Figure 1.14: (A) Measured drawdown from a pumping test [63] is shown in red. Simulation results are shown in gray while the 50 best-fitting curves are highlighted in black. (B) Model likelihoods are plotted as color, as a function of storativity and transmissivity.

	$T(\text{gpd}/\text{ft})$		$S(\text{ratio})$	
Walton, 1970	10100	$\pm 266^*$	$2.00\text{e-}5$	$\pm 39.42\text{e-}7^*$
McElwee, 1980	9910	$\pm 257^*$	$2.10\text{e-}5$	$\pm 40.77\text{e-}7^*$
Voronoi	9970	± 202	$2.08\text{e-}5$	$\pm 8.66\text{e-}7$
Gaussian KDE	10009	± 154	$2.05\text{e-}5$	$\pm 7.36\text{e-}7$
MCMC	9992	± 213	$2.08\text{e-}5$	$\pm 8.97\text{e-}7$
Analytical	9898	± 117	$2.14\text{e-}5$	$\pm 18.93\text{e-}7$

Table 1.1: Transmissivity and storativity estimates for the drawdown data in Walton, 1970 [63] and McElwee, 1980 [41] are shown. Inversion results are also shown using the Voronoi, Gaussian KDE, and MCMC approaches to likelihood integration (Section 1.3.3.1). Uncertainties marked with an asterisk (*) were calculated using Equation 1.19, and were not drawn from the references. The analytical estimate was calculated by choosing the best-fitting model and evaluating the derivative at that point in the parameter space.

In order to examine the sensitivity of this inversion technique to the amount of random noise in the data, the best-fitting model ($S = 2.11 \times 10^{-5}$, $T = 9924$ gpd/ft) is chosen and 100 different levels of random Gaussian noise are added to the resulting drawdown curve, ranging from 0.1% to 50%. These 100 realizations are then used to perform each of the parameter estimation methods available for this case: Voronoi integration, Gaussian KDE integration, MCMC, and analytical error propagation using Equation 1.19. This error analysis shows that all the methods provide similar parameter estimates and uncertainties up to a measurement error of approximately 0.01–0.1 feet.

Above 0.1 feet, the Gaussian KDE finds an estimate that is slightly closer to the truth, and a lower uncertainty. The analytical error propagation approach appears to underestimate parameter uncertainties at lower measurement errors, then rapidly diverges and over-estimates the uncertainty at high measurement errors. The pumping test dataset [63] indicates that measurements were made to the nearest 0.1 feet, however based on the data misfit between this data and the best-fitting Theis equation model a measurement error of $\sigma_n = 0.00905$ ft is estimated.

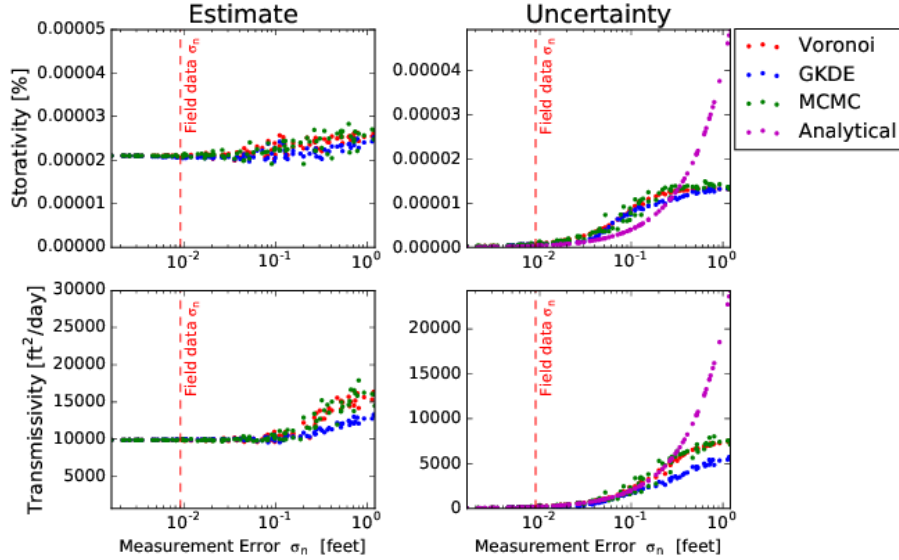


Figure 1.15: Parameter estimates and uncertainties for transmissivity and storativity are made as functions of measurements error, using the Voronoi, Gaussian KDE, MCMC, and analytical error propagation approaches. Red dashed line indicates the estimate of the measurement error in the source dataset. Note that no parameter estimates are shown for the analytical error propagation method, as it does not estimate the parameter itself but only the uncertainty. The analytical error propagation uncertainty is also linearly dependent on the measurement error σ_n , and therefore is not subject to the stochastic variation shown by the other methods.

1.4.2.2 Analysis of synthetic data, random numerical noise

The Theis equation was then used to construct a fictitious dataset for a multi-objective inversion. In this scenario there are two observation wells where drawdown is being measured and two injection wells [Figure 1.16a] with only the nearer injection well being pumped initially. The Theis equation is used to calculate drawdowns [Figure 1.16b], and various levels of random numerical noise are added. A separate inversion is then run for each noise level, and the data-model misfits are plotted as a function of storativity and transmissivity [Figure 1.16c] in order to examine how well the

two dataseries agree in terms of their information content. In this case, while the level of parameter uncertainty scales with the amount of randomly distributed data noise, the parameter estimates themselves remain relatively consistent. Plotting the two objectives against one other [Figure 1.16], data-model misfits fall roughly along a 1:1 line indicating that there is minimal tradeoff in the objective space.

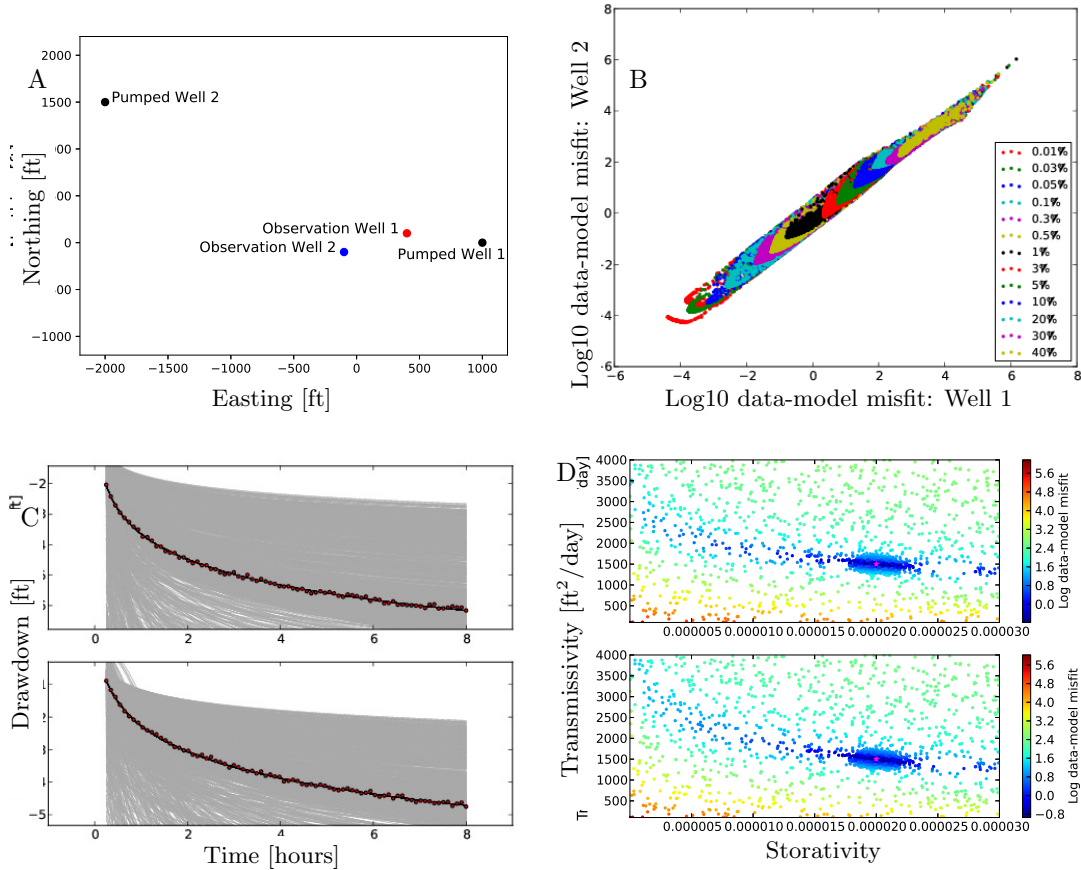


Figure 1.16: A). Map view layout of the fictional pumping test field site. B). Pareto graph shows Observation Well 1 data-model misfits plotted against their corresponding Observation Well 2 misfits at each noise level. C). Fictional 'measured' drawdown is shown (1% random noise) as red dots, compared to 5000 proposed simulations (gray lines). Pareto optimal models are highlighted in black. D). Data-model misfits are shown as a function of storativity and transmissivity.

1.4.2.3 Analysis of synthetic data, conceptual model error

In order to introduce systematic error in the data, a fictional dataset is constructed where the second well begins pumping halfway through the experiment, and is treated as an unknown

process contaminating the data [Figure 1.17a]. Therefore, the forward model used to fit this data does not include data from the second well. In this case a relatively minor systematic bias introduces non-uniqueness that prevents one from inferring the correct parameter values. Plotting the misfits against each other shows that rather than an "L"-shape with a point approaching the origin, data-model misfits are arranged along a diagonal form oriented normal to the origin, which indicating that a tradeoff relationship exists such that data from one well cannot be fit better without fitting data from the other worse.

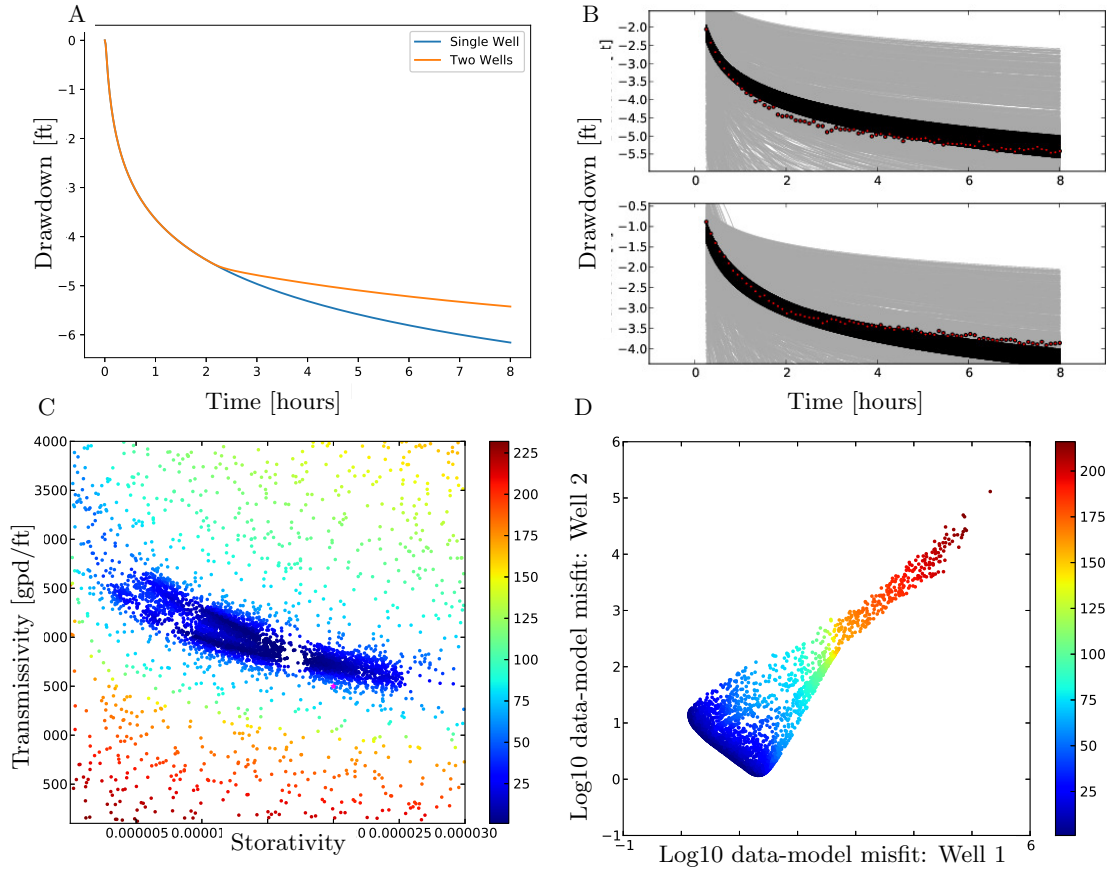


Figure 1.17: A). Drawdown as a function of time is shown for observation well 1, both with and without the secondary well pumping. B). Fictional 'measured' drawdown is shown (1% random noise) as red dots, compared to 5000 proposed simulations (gray lines). Pareto optimal models are highlighted in black. C). Pareto rank of each simulation is shown as a function of transmissivity and storativity. D). Pareto graph shows Observation Well 1 data-model misfits plotted against their corresponding Observation Well 2 misfits.

1.5 Conclusion

This inversion framework distributes computational effort over various computing resources, running a variety of standard inversion algorithms in parallel to efficiently explore a parameter space while avoiding premature convergence. When forward model run times are too long to allow for solid convergence using a sequential algorithm such as MCMC, interpolation of the existing simulation results can be used to develop a surrogate model that can run quickly enough to meet MCMC convergence diagnostics in a reasonable amount of time. Whether this surrogate model is sufficiently similar to reality can then be evaluated by using the bootstrapping method as well as a certain amount of professional intuition about the physical problem of interest. This modular programming approach will allow for other researchers to readily adapt their own forward models and inversion algorithms so they may take advantage of better parallelization and job scheduling.

Chapter 2

Numerical proof-of-feasibility of using geomechanical measurements to estimate poroelastic parameters

2.1 Abstract

Carbon injection projects introduce several risk-assessment challenges, including leakage and escape of carbon dioxide through a fracture or derelict well, the reactivation of dormant faults, topographic subsidence/uplift, or contamination of existing groundwater resources. In order to help assess and manage these risks, a stochastic estimation framework has been developed which is capable of characterizing the physical parameters of a formation during injection operations. A numerical forward model developed using Geocentric couples the Darcy flow and elastic deformation equations to compute a set of geomechanical signals (e.g. pressure, strain, tilt and displacement) given a set of model parameters. A set of stochastic optimization algorithms are then used to iteratively generate a sequence of parameter estimates, and a high performance cluster computer efficiently evaluates this computationally expensive forward model for each set of parameters. The set of converged parameter estimates is then used to find the best estimate and uncertainty of each parameter, as well as the uncertainty of each expected value as required by the measurement limitations (noise, model error, spatial/temporal constraints) imposed on the dataset. It is shown

that synthetic geomechanical measurements taken from a target formation, either from a nearby observation well or the injection well itself, can be used to accurately and efficiently estimate the physical parameters of that formation. This experiment also demonstrates that measurements taken in an overlying confining unit can be used to estimate the parameters of the target reservoir. This parameter sensitivity indicates that measurements in the confining unit can significantly mitigate drilling and instrument installation costs as well as reduce the risk of puncturing the confining unit.

2.2 Introduction

Carbon sequestration is one proposed method of reducing greenhouse gas emissions in order to mitigate the effect of carbon dioxide on climate change, ocean acidification and human health [15]. Carbon injection is the process of capturing carbon dioxide, compressing it into a super-critical fluid, and injecting it at high pressures into a deep, hydraulically confined geologic structure such as a saline aquifer, coal bed or depleted oil or natural gas reservoir. As supercritical carbon is injected into a target formation, a pressure wave propagates through the permeable formation layer at a relatively high velocity and begins diffusing through the impermeable cap rock layer much more slowly. In response to those pressure changes, the pore skeleton experiences elastic deformation and in turn exerts a stress on the cap rock. Unlike the diffusive pressure wave, this elastic response propagates quickly through the solid skeleton of the cap rock. This suggests the possibility of measuring the elastic response to injection from within either the cap rock or target formation, and using these measurements to infer the physical properties of the subsurface.

Previous studies have found that measurements of the strain and deformation fields can be used to infer some parameters of the subsurface. Seismic methods have been developed which relate seismic velocities to CO₂ saturation [65], and help identify geologic structures that can potentially act as an escape conduit [66]. Pressure and displacement datasets (either measured or synthetic) have been combined with stochastic methods such as Monte Carlo or Ensemble Kalman filtering [4], or deterministic approaches such as least-squares estimation [19] or Newton conjugate gradient descent [32, 35]. Tiltmeters installed at the Earth’s surface or in boreholes have also been combined with datasets such as pressure or displacement to study fractured reservoirs or karst systems [34], often employing Monte Carlo [49, 50, 37] or linear least squares [59] approaches. Satellite interferometry (InSAR) has also been used to fit vertical displacements using geomechanical flow models [11, 14],

or to integrate surface deformation data with borehole fluid pressure [35] and production data [12]. These studies have used linear optimization approaches [60], as well as genetic algorithms [38] and data assimilation [73]. Borehole strainmeters have also been used in the past to estimate rock properties by using a grid search approach to fit an analytical solution to strain data collected during a pumping test [3]. However, this effort used an analytical solution which may not be sufficient to describe geologic heterogeneities in a real field site.

The carbon sequestration process poses several operational risks including subsidence and uplift, fault reactivation, and hydraulic fracturing resulting in leakage and escape of stored CO₂. Mitigation of these risks will require careful site assessment and close monitoring in order to manage operations in a safe and economical manner.

The goal of this paper is to demonstrate the feasibility of using data from geomechanical instruments such as tilt meters and strain meters to estimate the poroelastic properties of a confined reservoir as it undergoes fluid injection. To this end, a forward model has been developed which couples the fluid flow and elastic deformation equations to compute a set of predicted geomechanical signals given a set of reservoir parameters. A set of stochastic optimization methods is then used to perform a series of inversions on this forward model. As these stochastic methods require many simulation runs in order to perform adequately, high-performance computing methods were used to distribute simulation runs over many computational nodes on a cluster computer. Inversion of this synthetic dataset is used to evaluate how well strain measurements could constrain reservoir properties.

2.3 Background

This project builds upon concepts from the fields of finite element simulation of geomechanics, and statistical model calibration.

2.3.1 Forward Model for Geomechanical Response

Poroelastic simulations couple two important dynamics: deformation of the solid pore skeleton in response to changes in the stress tensor, and flow of pore fluid in response to changes in pore pressure and deformation of the solid. These two processes are highly coupled; mechanical deformation of the pore skeleton induces changes in the pore pressure driving flow, and the changes

in pore pressure alters the stress on the pore skeleton. The Biot poroelasticity model [7] couples these processes using four constitutive equations. One constitutive equation specifies conservation of momentum of the fluid for a single phase during flow in a porous medium [Eq 2.1] based on the pressure head formulation of Darcy's law,

$$\vec{\mathbf{v}} = -\frac{1}{\rho_f g} \kappa \cdot \nabla p \quad (2.1)$$

where $\vec{\mathbf{v}}$ is the fluid flow vector, ρ_f is the fluid density, \mathbf{g} is the magnitude of acceleration due to gravity, κ is the hydraulic conductivity tensor, and ∇p is the pressure gradient. The second equation specifies conservation of mass of the fluid

$$-\nabla \cdot \mathbf{q} = \frac{1}{M} \frac{\partial p}{\partial t} + \alpha \frac{\partial \varepsilon_{kk}}{\partial t} \quad (2.2)$$

where \mathbf{q} is the volumetric flux, M is the Biot modulus, α is the Biot-Willis coefficient, and ε_{kk} is the volumetric strain. The third constitutive equation specifies conservation of momentum of the solid,

$$-\nabla \cdot \sigma = \rho_t \mathbf{g}, \quad (2.3)$$

where σ is the total strain tensor, ρ_t is the total density, and \mathbf{g} is gravity. The fourth equation specifies a small-strain elastic deformation equation [Eq 2.4] based on Hooke's law,

$$\sigma - S_0 = \mathbf{C} : (\varepsilon - \varepsilon_0) - \alpha_B p_f \mathbf{I} \quad (2.4)$$

S_0 is the initial strain, \mathbf{C} is the elasticity tensor and is a function of the Young's modulus (E) and Poisson ratio (ν), ε is the stress and ε_0 is the initial stress.

By defining a set of plausible values for the mechanical (i.e., E , ν , α_B etc) and hydrogeologic (K , ϕ) parameters above, a set of geologic structures can be defined. Using a finite element approach, one can then solve the resulting partial differential equation and simulate how pore pressure, deformation and strain evolve through time at given points in the subsurface.

2.3.2 Parameter Space and Pareto Optimality

The goal of the parameter estimation process is to find sets of parameter values that best fit a set of measured data, with the assumption that the parameters that best fit the data are

representative of the true subsurface. In the case of reservoir characterization problems however, it is often the case that the inverse problem is underdetermined or that many contradictory geologic models would explain the observable data. In addition many separate sets of measured data are often available, meaning the parameter estimation process must satisfy many distinct and sometimes conflicting objectives. These objectives might include minimizing data misfits for the pore pressure, deformation and strain misfits at a number of distinct points in the subsurface representing distinct observation wells. The objective space is therefore an m -dimensional space where each dimension represents one of the m data-fitting objectives. The goal of parameter estimation is therefore to identify the models that minimize the data-model misfit of each objective, or are as near as possible to the origin of the objective space.

One approach to working with large, complex objective spaces [40] is to select a vector of weighting coefficients typically related to the calibrated measurement error of the instrument, and use these weights to combine the data-model misfits into a single value using equation 2.5,

$$e_w = \sum_i^k e_i \omega_i, \quad e_i = \left| (d - \hat{d})^2 \right|, \quad (2.5)$$

where e_w is the weighted error, e_i is the L_2 norm, or the sum of squared errors between the measured and synthetic data, and ω_i is the weighting coefficient. Selecting the optimal weighting coefficients can be challenging, and requires one to implicitly decide which data objectives are most reliable and informative about the unknown parameters. One common approach is to divide by the range, variance, or measurement error (σ_d^2) of the measured data (Eqn 2.6) in order to normalize the data errors and produce a single, unitless measure of how well the parameters explain the data.

$$\omega_i = \frac{1}{\sigma_d^2} \quad (2.6)$$

Many inversion algorithms require a single error value to compare one solution to another, including the gradient descent and MCMC algorithms described below. In contrast, some multi-objective optimization algorithms instead use the concept of Pareto optimality [48] to treat each objective separately, allowing these algorithms to efficiently explore tradeoff relationships between data misfits. A specific set of parameter values is Pareto optimal when compared with other solutions if no other solution is superior to it in terms of all objectives. For example, in the multiobjective minimization problem presented in Figure 2.1a, solution 1 has a lower value in terms of objective f_1 ,

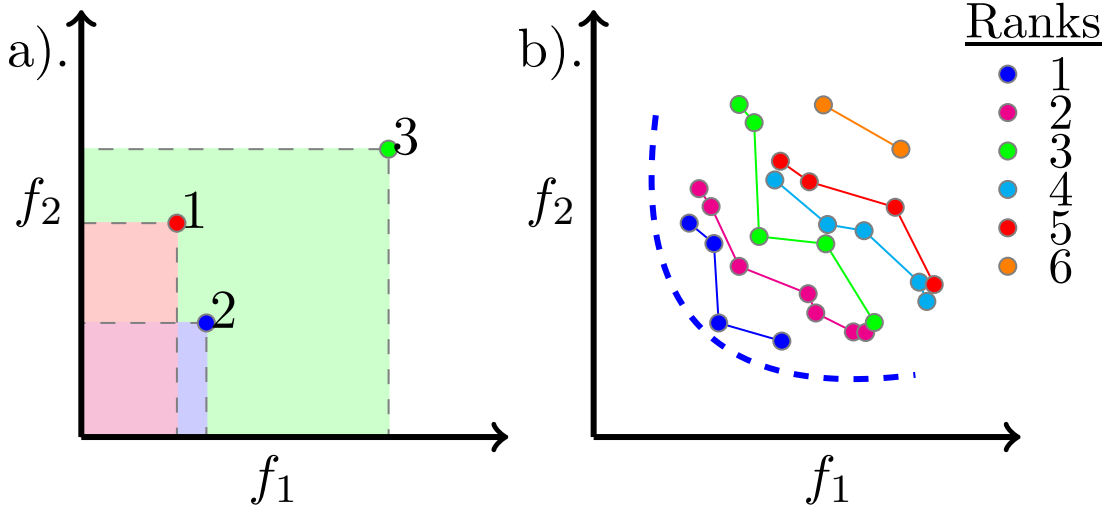


Figure 2.1: Illustration of Pareto optimality. Blue dashed line represents the 'true' Pareto front. Rank 1 solutions (blue dots) represent the best currently-available approximation of the Pareto front.

but a higher value in objective f_2 . Solution 2, by contrast, has a lower value in terms of objective f_2 , but a higher value in objective f_1 . Therefore neither solution is superior to the other in the Pareto optimality sense. However, solution 3 is not Pareto optimal, because both solutions 1 and 2 have lower f_1 and f_2 values. The group of solutions that are not dominated by any other solution are denoted rank 1. Solutions that are dominated only by rank 1 solutions are denoted rank 2, and so on. In Figure 2.1b, several ranks are identified and color-coded.

2.3.3 Parameter Space Search Methods

The parameter space is an n -dimensional space, where n is the number of unknown parameters in the reservoir model. Thus each point in the parameter space represents a unique model, whose likelihood can be investigated by evaluating the numerical simulation and comparing the results to measured data in order to find its position in the objective space.

The simplest way to define the parameter space is using a uniform distribution over the range of possible values for each of the n parameters, and constructing an n -dimensional hypercube. However, it is often possible to reduce the complexity of the problem by using additional information to define a smaller feasible space with an arbitrary shape. This shape can be deterministic, based on an analytical relationship between two of the parameters (as in the case of Equation 2.10), or it can be statistical based on empirical observations [Figure 2.7].

In order to define the mapping [Figure 2.2] between the parameter space and objective space, there are a number of distinct search methods for sampling the parameter space. Each search strategy strikes a balance between 'exploration' and 'exploitation'. Exploration refers to searching the entire parameter space as thoroughly as possible, in order to identify promising new models. Exploitation refers to searching a narrower space around the most promising models identified so far, refining and improving upon them. By running multiple search algorithms in parallel and allowing them to share results at each iteration, exploration is balanced with exploitation.

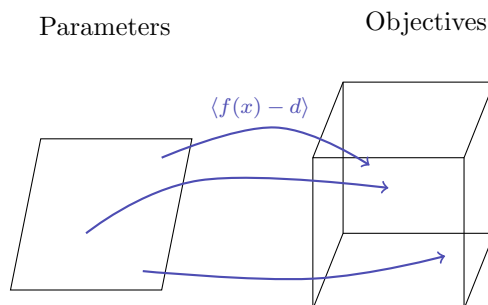


Figure 2.2: Example of a mapping between a 2-dimensional parameter space and a 3-dimensional objective space.

The Monte Carlo method [43] uses the prior model (based on any pre-existing knowledge about the problem) to sample the parameter space at random. It has a simple, efficient implementation that searches the space thoroughly without sampling too densely in one space.

The gradient descent method [69] also begins by sampling a random point, but then samples a grid of neighboring points in order to estimate the Jacobian matrix, or the gradient of the objective functions with respect to the parameters. It then uses this gradient information to take a step in the parameter space, and begins populating a new grid of neighboring points. Gradient descent converges very efficiently on low-dimensionality problems, however because each step requires $2n + 1$ simulations in order to evaluate the Jacobian, it tends to become impractical when the number of unknown parameters is high. It also tends to converge prematurely on local minima if they exist.

The Markov chain Monte Carlo (MCMC) method [42, 29] also begins at a random point and takes a series of down-gradient steps. However, rather than estimating the gradient, MCMC

simply takes a series of random steps, accepting or rejecting each step based on the change in error. If a step reduces the error, it is accepted. If it increases the error, it is accepted with a probability proportional to the change in error. This approach is more efficient for high-dimensionality problems since it requires only one simulation per step. It tends to provide slower convergence than gradient descent, but is also capable of escaping local minima and avoiding premature convergence.

The sparsest point method uses computational geometry to identify points in the parameter space that have been most sparsely explored by the other methods. This tends to disproportionately explore less promising regions of the parameter space, but prevents the neglect of large unexplored spaces.

Multiobjective genetic algorithms (NSGAII [16] and SPEA2 [71]) collect the broad range of unique solutions available, and use the concept of Pareto optimality to select the best models. By making random permutations of these selected models, they select the best attributes from the existing population and recombine them to produce improved models. These populations tend to quickly swarm around the best solutions found so far, then quickly abandon them if and when better models are found.

2.4 Methods

2.4.1 Forward Model

The numerical forward model uses the geomechanical partial differential equation solver Geocentric [67, 68] to perform simulations of a reservoirs poroelastic response to pumping. The domain is cylindrical with a radius of 30 km [Figure 2.3a] and an axis of symmetry defined about a central injection well. The model consists of three horizontal layers, a permeable formation (100 m thickness) between a lower confining layer (100 m thickness) and an upper confining layer (1,000 m thickness).

Axial symmetry is used to minimize computational effort, with the inner corner of the model space representing the injection well as a line source. We prescribe a no-flow condition along the upper and lower boundary and a specified fluid pressure ($p_f = 0$) along the outer circumferential boundary. The well is screened within the permeable reservoir and has a prescribed pressure that starts at steady state ($p_f = 0$) and quickly ramps to 1 MPa over 2 hours.

The lower boundary is a roller boundary allowing radial or circumferential displacement

but not vertical displacement. The upper boundary can deform freely. The outer circumferential boundary is also a roller boundary, allowing circumferential or vertical displacement but not radial displacement.

The mesh is coarsest near the outer boundaries where gradients are expected to be very small, and becomes denser near the injection well. Radial symmetry is used to minimize computation time, thus flow and its accompanying geomechanical response are computed for only a wedge of the cylinder [Figure 2.3b].

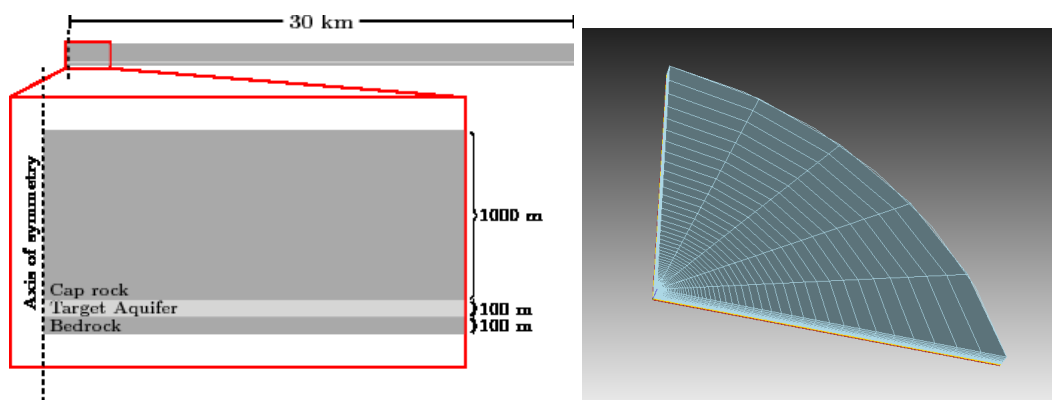


Figure 2.3: Geometry and mesh of conceptual model for numerical poroelastic simulation.

Geocentric has no licensing restrictions, so many instances of it were run on many separate machines allowing a linear speedup limited only by the computational resources available. Using 100 high-end nodes (8 cores, 16 GB RAM), a sustained throughput of roughly 4,000 Geocentric simulations per day was achieved. As conceptual models become more complex requiring heterogeneous structures, multiphase flow, plastic deformation or nonisothermal effects, the average runtime will increase dramatically. Therefore, in order to be effective for near-real-time decision-making and monitoring of carbon sequestration operations, an optimization approach is required that is capable of converging reliably with a minimal number of function evaluations.

2.4.2 Synthetic model

We first select an arbitrary model with typical property values for a sandstone formation and shale confining unit. This model is then treated as the 'true' model which defines the synthetic data. A set of instruments are defined with unique locations and levels of measurement noise. The forward model is then used to generate a timeseries of synthetic data for each instrument, and

random noise is used to represent measurement error.

Real data tends to have a characteristic baseline level of noise, and a scale-dependent component of noise that increases with the magnitude of the process being measured. However, simply scaling data noise to the magnitude of the measurement yields unrealistic behavior in cases where the measured property starts at zero and fluctuates between large positive and negative values. The noise model therefore represents the effects of hysteresis using the cumulative sum of the differences between time steps, producing a noise function that starts at the baseline value and smoothly increases with the magnitude of the total change in the data.

2.4.3 Computational Workflow

We initialize the model calibration process by parameterizing the problem and defining the unknown parameters of interest. For each unknown, a statistical distribution is built based on prior knowledge about the problem. These statistical distributions are then used to generate a set of initial geologic models. A while loop is run until convergence criteria are achieved. At each iteration, the newest set of geologic models are evaluated to generate predicted timeseries for each of the instruments at each installation. These results are then compared with measured or synthetic datasets, and the data/model misfits are used to improve statistical distributions (see Section 2.4.4). Each search algorithm then uses its own unique strategy to generate a new set of geologic models. This approach allows one to share simulation results between separate search strategies. Since each strategy has its unique advantages and disadvantages, coupling these processes allows one to integrate the thorough exploration of some strategies with the careful exploitation of others.

2.4.4 Parameter Estimation

We use a statistical inference method based on Bayes theorem to summarize the simulation results and infer statistics of the posterior probability of the model parameters. A data likelihood function $\rho(\mathbf{d}|\mathbf{m})$ is used to relate a prior probability density function $\rho(\mathbf{m})$ to a posterior probability density function $\rho(\mathbf{m}|\mathbf{d})$.

$$\rho(\mathbf{m}|\mathbf{d}) = \rho(\mathbf{d}|\mathbf{m}) \frac{\rho(\mathbf{m})}{\rho(\mathbf{d})} \quad (2.7)$$

By integrating the posterior probability density function over the volume of the parameter

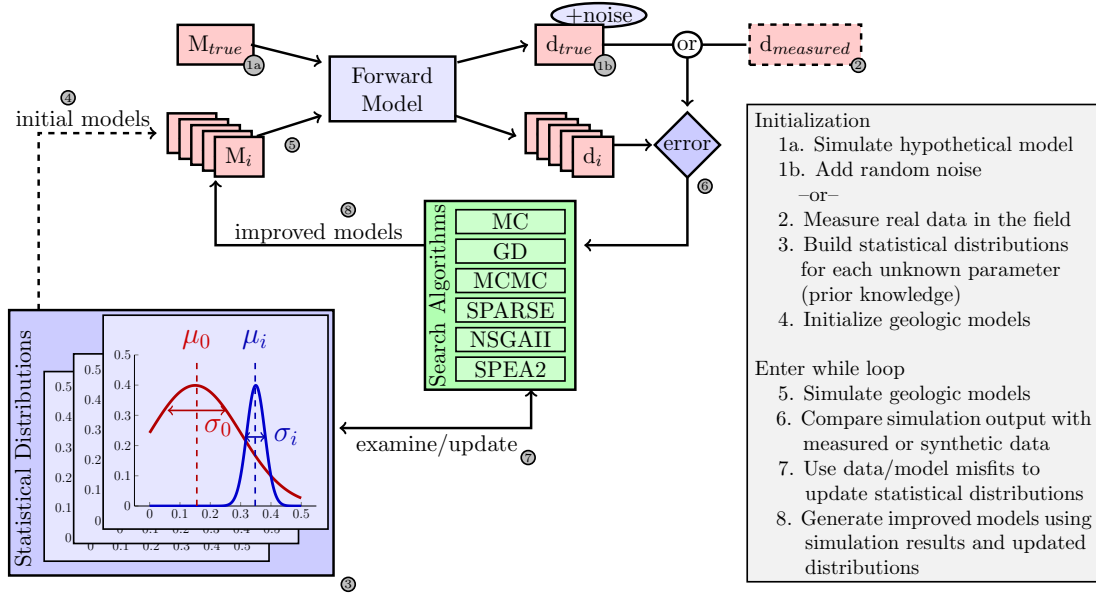


Figure 2.4: Illustration of parallel model calibration process. A hypothetical model M_{true} is run through the forward model to generate a synthetic dataset, d_{true} . Alternatively, field observations can be collected $d_{measured}$. A series of random models (M_i) are then generated based on the prior model (mean μ_0 , variance σ_0) and evaluated by the forward model to produce simulated datasets (d_i) which are then compared to the measured or synthetic dataset. These comparisons are used by a variety of inverse algorithms including Monte Carlo (MC), Gradient Descent (GD), Markov Chain Monte Carlo (MCMC), space-filling algorithms such as Latin Hypercube or Voronoi method (Sparse), genetic algorithms (NSGA-II, SPEA2). These algorithms use data-model comparisons to produce iteratively improved models and thereby reducing uncertainties and providing a better characterization of the posterior distribution (mean μ_i , variance σ_i).

space, the posterior probability $\rho(\mathbf{m}|\mathbf{d})$ can be directly evaluated for any point sampled, and therefore the expected value (mean) and uncertainty (standard deviation) of the model parameters as implied by the data fits can be computed.

$$\mu = \int m \rho(m|d) dm, \quad (2.8)$$

$$\sigma^2 = \int (m - \mu)^2 \rho(m|d) dm, \quad (2.9)$$

In order to evaluate the integral at low dimensionalities, an n -dimensional voronoi grid is constructed with centroids at each of the existing samples. The volume $V(\mathbf{m})$ of each voronoi cell is then computed using an n -dimensional convex hull algorithm [2], and the volume of each cell is

multiplied by its probability density $\rho(\mathbf{m}|\mathbf{d})$. The product $V(\mathbf{m}) \cdot \rho(\mathbf{m}|\mathbf{d})$ is then summed across all cells to arrive at the integral. At higher dimensionalities, stochastic approximations of the integral may be more practical.

2.4.5 Experimental Design

The experimental design uses a set of instrument types and locations described in Table 2.1, composed of four synthetic instruments located in two wells, one being the injection well and the other an observation well 1000 meters from the injection. Both wells have one instrument located in the vertical center of the formation (depth of 1050 meters), and another in the shallow cap rock (depth of 150 meters). Fluid pressure, displacements, and strains are computed at each of these locations. However, in a real field scenario, displacement measurements will only be possible near the injection well while only strains will be observable in an observation well far from injection. Therefore, these datasets are neglected in the analysis phase of the inversion.

For each case, poroelastic parameters of the formation are estimated including hydraulic permeability κ , drained bulk modulus of the solid K , Poisson’s ratio ν , porosity ϕ and grain compressibility K_s . The properties of the confining unit are assumed known with a permeability of 10^{-19} m², bulk modulus of 10 GPa, Poisson’s ratio of 0.25, porosity of 0.20, and grain compressibility of 42.9 GPa (quartz). Using the mesh, boundary conditions and initial conditions described in Section 2.4.1, a synthetic pumping experiment is performed with injection pressure of 1 MPa for 10^8 seconds or approximately 3 years. A hysteretic scale-dependent noise model is then applied to represent measurement error.

Table 2.1: Summary of instruments and instrument locations. Fluid pressures (p_f), displacement components (u_x, u_y, u_z), strain components ($\varepsilon_x, \varepsilon_y, \varepsilon_z$) and tilt components ($\nabla_z u_x, \nabla_z u_y$) measured at different instrument positions.

	Position		Instrument Types								
	r [m]	z [m]	p_f	u_x	u_y	u_z	ε_x	ε_y	ε_z	$\nabla_z u_x$	$\nabla_z u_y$
Injection well, deep	0.1	150	✓	✓	✓	✓				✓	✓
Injection well, shallow	0.1	1050	✓	✓	✓	✓				✓	✓
Observation well, deep	1000	150	✓				✓	✓	✓	✓	✓
Observation well, shallow	1000	1050	✓				✓	✓	✓	✓	✓

First, a single inversion is run using all available datasets, estimating for only three parameters (Young’s modulus, conductivity and Poisson’s ratio). This allows one to evaluate the overall value of the combined datasets.

Three separate inversions are then run with three different noise levels of 2%, 7%, and 21% to investigate the sensitivity of the inversion to measurement error. Additionally, a fourth inversion with a noise level of 1% is run during the parameter sampling and simulation phase. After these simulations have been run, the noise function is re-applied at many different noise levels and the parameter estimation [Section 2.4.4] procedure is re-evaluated for each noise level. This allows the trend shown in Figure 2.6 (gray bars) to be generated, while using the fixed-noise inversions (red bars) to verify that this trend is not an artifact of the sampling and simulation phase of the method, but represents real sensitivity to the noise in the data.

A series of inversions are then run incorporating various limited combinations of instruments type and location, to evaluate the optimal locations and instrument types for monitoring of a carbon injection project.

The role that parameter tradeoff plays in the inversion is then investigated by attempting to estimate all five poroelastic parameters of the formation, permeability κ , drained bulk modulus of the porous solid K , bulk modulus of the solid grains K_s , Poisson's ratio ν , and porosity ϕ . For this case the properties of the confining unit are assumed to be known. *A priori* analytical constraints are imposed on the parameters to reduce the size of the feasible space. The Biot coefficient α_B is a function of the bulk modulus of the porous skeleton and the bulk modulus of the solid grains [Equation 2.10], and cannot physically be less than zero. In practice, for earth materials the Biot coefficient is generally greater than 0.5, so a large range of combinations of K and K_s can be neglected from the analysis.

$$\alpha_B = 1 - \frac{K}{K_s} \quad (2.10)$$

$$1/M = \frac{\alpha_B - \phi}{K_s} + \frac{\phi}{K_f} \quad (2.11)$$

2.5 Results and Discussion

2.5.1 Base Case

The first inversion ran for approximately ~6,000 simulations, and 60 unique sets of model parameters [Figure 2.5] were found that explain all available data. Applying numerical integration

of the posterior distribution yielded an estimated Young's modulus value of 10.18 ± 0.12 GPa, hydraulic conductivity of $10^{-5.98 \pm 0.18}$ m/s, and Poisson's ratio of 0.29 ± 0.11 . For comparison, the 'true' synthetic model used a Young's modulus of 10.2 GPa, hydraulic conductivity of 10^{-6} m/s and a Poisson's ratio of 0.25.

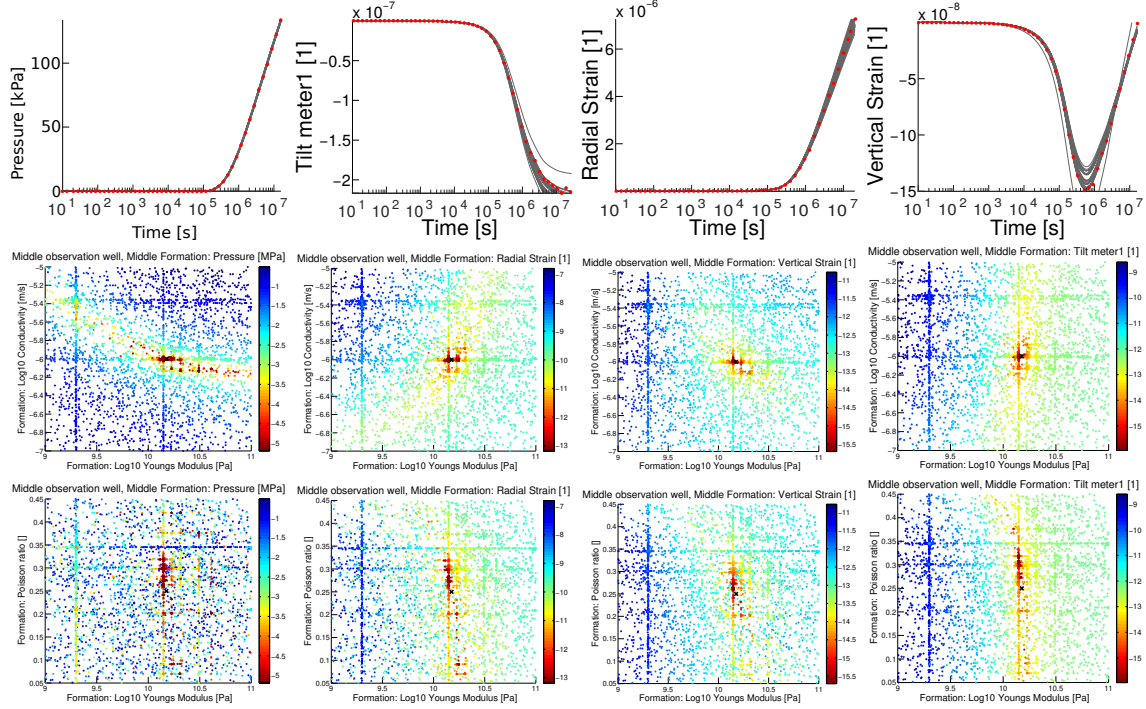


Figure 2.5: Pareto optimal data fits are shown (top row) for fluid pressure, tilt, radial strain, and vertical strain at the observation well in the center of the injection formation. Simulation results are shown in gray while synthetic data measurements are shown in red. Data-model errors are presented as a function of Young's modulus and conductivity (middle row), and as a function of Young's modulus and Poisson's ratio (bottom row).

2.5.2 Sensitivity to Data Noise

We find that parameter estimates exhibit moderate sensitivity to the magnitude of the random measurement noise in the data. The estimate of permeability shows the least sensitivity to random noise, while Young's modulus and Poisson's ratio are more sensitive. In each case while the uncertainty in the parameter estimate begins to increase dramatically at around 10% measurement error, the expected value does not deviate from the true value until around 20-30% measurement error.

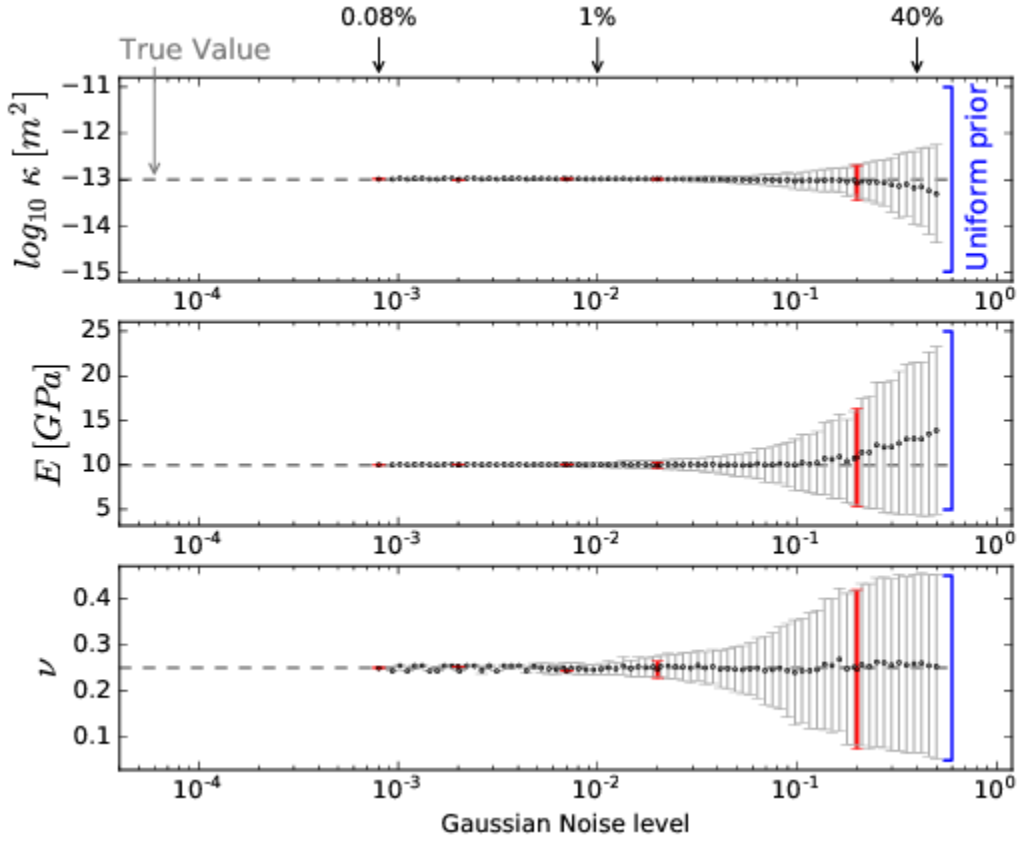


Figure 2.6: Parameter estimates and uncertainties are presented as a function of the Gaussian noise added to the synthetic data. Red error bars indicate separate runs of the parameter sampling (~ 4000 simulations each). Each gray error bars re-uses the same sample of ~ 4000 simulations, with different levels of Gaussian noise applied to the synthetic data.

2.5.3 Sensitivity to Instrument Type and Location

The results indicate that pressure measurements by themselves are least effective at estimating the three parameters, whereas displacement, strain and tilt measurements have similar information value [Tables 2.2 and 2.3].

However, combining pressure with displacement, strain or tilt improves these parameter estimates, and all four data types improves the estimate. Deep instruments installed within the formation have a slight edge over shallow instruments. However, in general combining multiple sensor locations provides the most information. As models become more complex and heterogeneous, the value of multiple data components in multiple locations should increase dramatically. Therefore, since puncturing the confining unit and installing instruments within the formation carries a

significant cost and risk, it is generally more effective to install a greater number of geomechanical instruments in the shallow subsurface.

Table 2.2: Estimates of permeability (κ), bulk modulus (K) and Poisson’s ratio (ν) are shown in terms of mean (μ), error (δ) and standard deviation (2σ), with various instrument types included in the analysis. The synthetic datasets use a 2% noise level taken from the instrument locations specified in Table 2.1, and include various combinations of pressure (P), displacement (D), tilt (T), and strain (S) data.

Instrument	Permeability [$\log_{10} \text{ m}^2$]			Bulk Modulus [GPa]			Poisson’s Ratio		
	μ	δ	2σ	μ	δ	2σ	μ	δ	2σ
P	-12.9923	-0.059%	0.132	10.6912	6.912%	2.876	0.2170	-13.218%	0.122
D	-12.9999	-0.001%	0.079	10.0318	0.318%	0.644	0.2491	-0.375%	0.025
S	-12.9975	-0.019%	0.088	10.0110	0.110%	0.706	0.2488	-0.495%	0.032
T	-12.9998	+0.002%	0.059	10.0523	0.523%	0.979	0.2506	+0.259%	0.026
P+D	-12.9982	-0.014%	0.067	10.0383	0.383%	0.595	0.2493	-0.288%	0.023
P+S	-12.9965	-0.027%	0.074	10.0173	0.173%	0.630	0.2491	-0.341%	0.028
P+T	-12.9985	-0.011%	0.052	10.0371	0.371%	0.806	0.2504	+0.151%	0.021
All	-12.9989	-0.009%	0.038	10.0332	0.332%	0.355	0.2503	+0.125%	0.010
True Value	-13			10			0.25		

Table 2.3: Estimates of permeability (κ), bulk modulus (K) and Poisson’s ratio (ν) are shown in terms of mean (μ), error (δ) and standard deviation (2σ) with various instrument locations included in the analysis. The synthetic datasets use a 2% noise level taken from all instrument types specified in Table 2.1, and include various combinations of the injection (inj) and observation (obs) wells.

Instrument	Permeability			Bulk Modulus			Poisson’s Ratio		
	μ	δ	2σ	μ	δ	2σ	μ	δ	2σ
Inj	-13.0068	+0.053%	0.088	10.0388	+0.388%	0.642	0.2498	-0.066%	0.028
Inj, deep	-12.9972	-0.021%	0.084	10.0252	+0.252%	1.039	0.2499	-0.027%	0.026
Inj, shallow	-13.0009	+0.007%	0.054	10.0411	+0.411%	0.440	0.2507	+0.272%	0.013
Obs	-13.0002	+0.002%	0.074	10.0715	+0.715%	0.821	0.2492	-0.334%	0.024
Obs, deep	-12.9935	-0.050%	0.095	9.9987	-0.013%	1.117	0.2485	-0.604%	0.029
Obs, shallow	-12.9965	-0.027%	0.058	10.0290	+0.290%	0.658	0.2497	-0.126%	0.017
All deep	-13.0021	+0.016%	0.051	10.0373	+0.373%	0.427	0.2497	-0.113%	0.017
All shallow	-12.9959	-0.032%	0.063	9.9985	-0.015%	0.788	0.2493	-0.284%	0.020
All	-12.9989	-0.009%	0.038	10.0332	+0.332%	0.355	0.2503	+0.125%	0.010
True Value	-13			10			0.25		

2.5.4 Parameter Tradeoffs

When estimating five unknown parameters simultaneously, synthetic geomechanical dataset is found to be insufficient to uniquely constrain a single, unimodal solution. This occurs because of tradeoffs between parameters. Initially, the parameter space is sampled using the space-filling algorithm only. These simulations were then compared to the first 10^6 seconds (~ 11 days) of data measured and it was observed that the pressure wave from injection had begun to reach the instruments in the deep observation well [Figure 2.7a]. By plotting these data-model misfits as a

function of the permeability and porosity, a parameter tradeoff can be observed [Figure 2.7b] that allows one to combine early pressure data with a physical understanding of the problem to develop an additional parameter constraint, thereby reducing the size of the feasible space of the inverse problem. Since the pressure travels as a diffusive wave the arrival of the pressure wave in the deep observation well is linearly related to both the permeability κ and porosity ϕ ,

$$D = K/S, \quad K = \frac{\phi\kappa}{\mu} \nabla P, \quad S \leq \phi \quad (2.12)$$

The inversion then began to incorporate other parameter search methods, therefore constrained within the smaller set of parameter combinations that agree with the pressure wave arrival. Note that plots such as Figure 2.7b show a projection of a 5-dimensional space onto a 2-dimensional plane. Therefore Figure 2.7b indicates that regardless of any variation in the other 3 parameters, permeability and porosity must be along the dashed line in order to explain the data. This trend was found to be consistent across all 36 datasets (9 instrument types at 4 distinct locations, see Table 2.1). Therefore a simple probability density function [Figure 2.7c] is defined, imposing the requirement that future models be drawn much more frequently from within this region of the parameter space.

This approach mirrors the way that a real long-term experiment might operate. In practice, simulations can begin running once the controlled parameters of a field experiment are known (ie pumping rate, well locations, sensor calibration etc), but the search of the parameter space may not be guided by information content of the data until much later in the field experiment. As additional simulation results become available, parameter constraints may be modified according to the judgment of the modeller and based on new information as it becomes available.

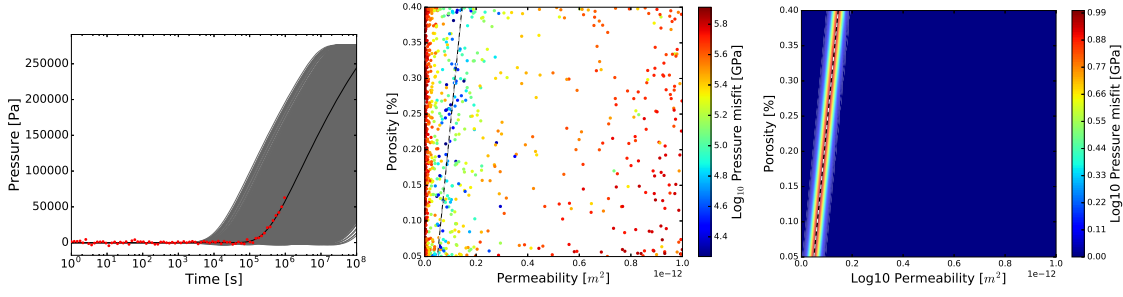


Figure 2.7: Preliminary model calibration results. (A) Comparison of synthetic data and simulated responses of the pressure breakthrough curve for the deep observation well. (B) Data-model misfits (L_2 -norm) for pressure sensor in the deep observation well as a function of permeability and porosity. Dashed line indicates the set of best-fit solutions. (C) Probability density function based on best-fit solutions.

In the case of this five-parameter inversion, geomechanical data was unable to constrain parameters to a unique solution. Instead a 5-dimensional tradeoff space emerges which can be roughly visualized according to Figure 2.8. By selecting the set of models (171 models) that fit all available datasets within a given tolerance (ie $2\sigma_n$), this tradeoff space can be characterized as a 5-dimensional polygon. While this polygon extends over a broad region of the parameter space, in this case it is constrained to approximately 2 millionths of a percent of the volume of the feasible model space.

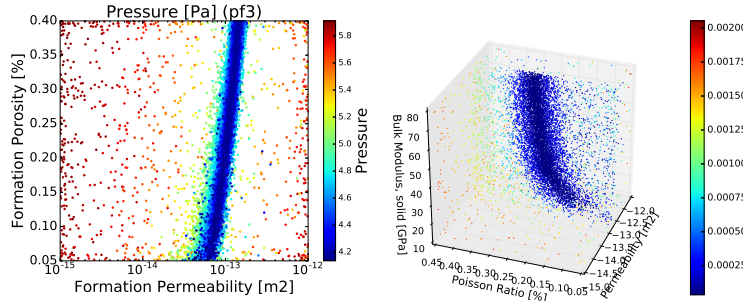


Figure 2.8: Preliminary model calibration results. (A) Comparison of synthetic data and simulated responses of the pressure breakthrough curve for the deep observation well. (B) Data-model misfits (L_2 -norm) for pressure sensor in the deep observation well as a function of permeability and porosity. Dashed line indicates the set of best-fit solutions. (C) Probability density function based on best-fit solutions.

In a real field operation, this set of plausible models could be used for risk assessment and planning purposes. Based on the set of 171 models that agree closely with the available data, one may evaluate how and where they differ from each other, and use this information to predict the

range of possible system behaviors as well as to identify optimal locations for additional instruments for long-term monitoring. For example in Figures 2.9, the mean and standard deviation of these 171 pressure signals show that at day 183 the models have a higher variance in the formation in the range of 2 km from the injection site. Therefore one might expect to improve parameter constraints using data from a pressure sensor at that location. It can also be observed from Figure 2.10 that the caprock near the injection site would be an ideal place for additional displacement instruments. Once the parameters are constrained as much as possible, one might then run simulations of various scenarios for pumping pressure and duration, and evaluate the probability of the formation exceeding a given hydraulic fracturing pressure.

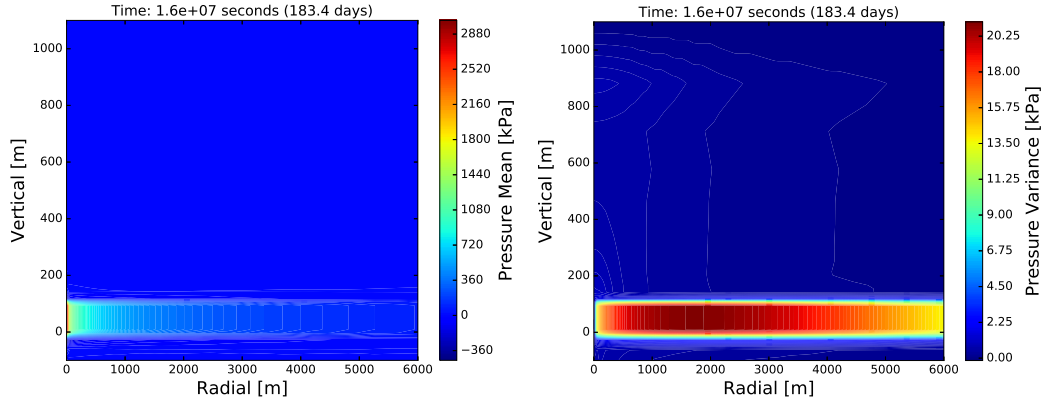


Figure 2.9: Mean and standard deviation of the pressure of 171 plausible models is plotted as a function of radial distance from the injection site and vertical distance above the lower confining unit.

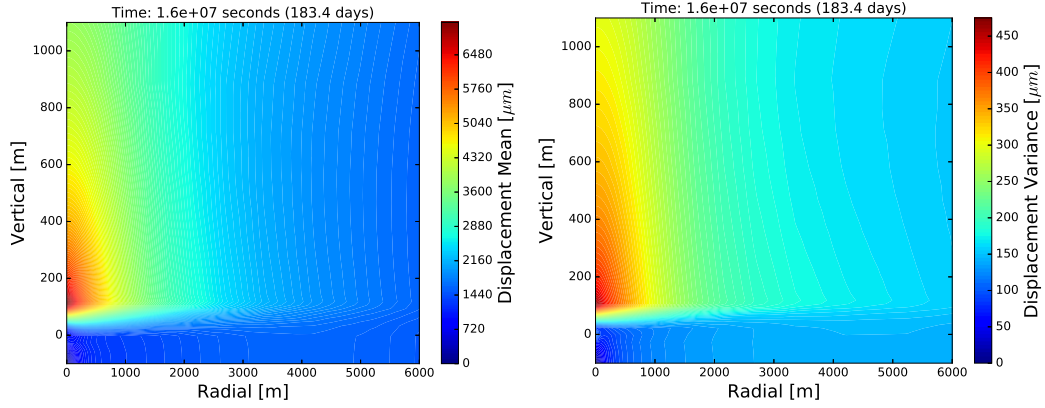


Figure 2.10: Mean and standard deviation of the displacement magnitude of 171 plausible models is plotted as a function of radial distance from the injection site and vertical distance above the lower confining unit.

2.6 Conclusion

This study finds that geomechanical measurements of pressure, displacement, tilt and strain can be used to estimate parameters within a poroelastic system such as a carbon storage reservoir. While the relative information value of these data types is highly dependent on the particular geomechanical problem, it is found that pressure measurements alone are generally less informative than tilt, displacement and strain. Placing instruments inside the target formation is also more informative than in the shallow confining unit, however rarely enough so as to justify the added cost and risk associated with drilling through the confining unit of a carbon capture and storage project. In cases where geomechanical field data is insufficient to conclusively estimate poroelastic parameters, it can still often reduce the parameter space dramatically allowing for improved risk assessment as well as informing installation of new instruments at more informative sites.

Chapter 3

Using borehole strain tensor measurements to locate and characterize subsurface heterogeneities in the Bartlesville formation in Avant field, Oklahoma

3.1 Abstract

Assessing reservoir properties frequently relies on measurements from wells drilled directly to the depth of interest, penetrating reservoir or formation. A less intrusive method of evaluating such reservoirs is therefore field-tested to demonstrate proof-of-concept of numerical results from a previous study. Previous work based on numerical simulations demonstrated that in situ strain measurements collected from shallow sensors could be used to monitor production and estimate reservoir properties. This study validates these theoretical findings using a field test at the Avant oil field in Oklahoma.

A shut-in test was performed by pumping produced water into a permeable reservoir at

a fixed rate until steady state conditions were achieved, then briefly shutting-off injection for one week. This pumping test was monitored using three components of the strain tensor at a nearby shallow well (30 m depth, 1 km away). The data were then used to infer the position, size and composition of the confined reservoir using a stochastic optimization technique was then applied to simulate approximately 50,000 possible geomechanical models of the subsurface.

This field test confirmed that near-surface strainmeter data is sensitive to geologic processes and structures at deeper depths. While a unique solution was not found, an ensemble of models which explain the data adequately were constructed and used to dramatically reduce model uncertainties.

3.2 Introduction

Measuring strain and understanding geomechanical systems is important for a wide range of applications. For example, geomechanics has been used for site assessment and monitoring the geologic injection of compressed carbon dioxide for carbon capture and storage [13] and enhanced oil recovery [70, 33, 51], optimization of hydraulic fracturing operations [20], geothermal energy production [21], and water production from aquifers driven by fracture flow [5]. Understanding responses to induced strain also has value in risk assessment studies as high-pressure injection into a confined formation can cause human-induced microseismicity through fault reactivation [10, 47, 44], and displacement due to fracture behavior or aquifer compaction can cause damage to built infrastructure [22] and may open a preferential flow pathway allowing pore fluid to move between previously isolated reservoirs and potentially carrying contaminants into a fresh-water aquifer.

Being able to predict and manage strain responses requires an understanding of the geomechanical processes and parameters in a particular setting in order to estimate subsurface properties. A number of inversion methods have been applied to geomechanical datasets. One study used an Ensemble Kalman filter to fit a numerical geomechanics simulator to measured pore pressures and well production data [12]. Other studies have combined measured or synthetic pressures with displacement data using stochastic methods, such as Monte Carlo or Ensemble Kalman filtering [4], or deterministic approaches such as least-squares estimation [19] or Newton conjugate gradient descent [32, 35]. Tiltmeters installed at the Earth’s surface or in boreholes have also been combined with datasets such as pressure or displacement to study fractured aquifers or karst systems [34], often employing Monte Carlo [49, 50, 37] or linear least squares [59] approaches. Satellite interferometry

has also been used to fit vertical displacements to geomechanical flow models [11, 14]. These studies have used linear approaches [60], as well as genetic algorithms [38] and data assimilation [73]. Bore-hole strainmeters have also been used to estimate rock properties by using a grid search approach to fit an analytical solution to strain data collected during a pump test [3].

While a variety of geomechanical datasets have been used in inverse problems, heterogeneous 3D numerical simulations have thus far proven too computationally intensive to calibrate to strain data. In this study near-surface strain is measured during active oilfield operations, and parallelized inverse methods are used to determine how sensitive these data are to stresses in deep confined reservoirs, and whether it can be used to infer the structure and composition of the subsurface. In particular this study investigates the sensitivity of geomechanical parameters to measurement errors such as measurement noise and instrument drift, and to whether sensitivity to geomechanical parameters of interest is less than or greater than sensitivity to unpredictable processes outside of the conceptual model. This study also examines the existence, uniqueness, and computational tractability of solutions to the resulting inverse problem. This field test is a follow up of previous inversions [Section 2.5], where a similar inverse methodology was applied to establish proof-of-concept using synthetic data. These results showed that non-uniqueness issues are common in geomechanical inverse problems, so it is expected that real-world data will reduce the range of possibilities dramatically but may not find a unique, unimodal solution.

3.3 Background

The strain response of a solid, porous material under stress by a pressurized pore fluid is governed by a two-way coupling between the mechanical deformation of the solid medium, and the diffusive flow of the pore fluid. Terzaghi’s theory of 1D consolidation [57, 56] modelled the compaction of a porous saturated material by assuming that both the solid and fluid components were incompressible. A general theory of poroelasticity [7, 8] was later developed to account for compressibility, and was incorporated into the general framework of continuum mechanics [64]. The continuum mechanics approach contrasts with fracture behavior, where a discrete discontinuity in the porous material allows for displacement along a plane accomodating stress in some areas while concentrating it in others.

3.4 Methods

This study involved a field component, forward modelling component, and inverse modelling component. A set of strainmeters and tiltmeters were installed and calibrated, and used to monitor strain during a scheduled well shut-in. A forward model was developed and adjusted for optimal accuracy and efficient use of computational resources. An inverse method using a combination of Latin Hypercube and NSGA-II genetic algorithms was implemented, and used to identify a set of models that would explain the measured field data.

3.4.1 Shut-In Experiment

Our field site is an active oil field near Avant, Oklahoma. The target reservoir in this study is the Bartlesville Formation, a sandstone layer approximately 30 meters thick, at a depth of approximately 500 m and confined by layers of shale. The disconnected lenses of high permeability material directly underlying the Bartlesville and penetrating the lower shale are understood to have been deposited under fluvial conditions. These high-permeability lenses are promising locations for oil exploration and are important re-injection locations for produced water.

This field experiment included 5 active wells. Wells 1A, 4A, 731 and 25-1W [Figure 5.1] are screened in the Bartlesville formation and underlying high-permeability lenses, but cased throughout the confining unit. Injection had occurred at a relatively constant rate (1500 barrels/day) in each of these wells for several weeks, resulting in an approximate steady-state condition. Well 1A was then shut-in for one week, with some of the excess flow rate being partially redistributed to the other three injection wells (200 barrels/day increase at well 4A, 200 barrels/day at well 731, and 400 barrels/day at well 25-1W). When injection resumed at Well 1A, the reservoir began to recover and return to its previous steady-state condition. A Gladwin strainmeter was installed in a shallow well (AVN2) at a depth of 30 meters, and the North-South, East-West, and shear strain were recorded for the duration of both the shut-in and recovery.

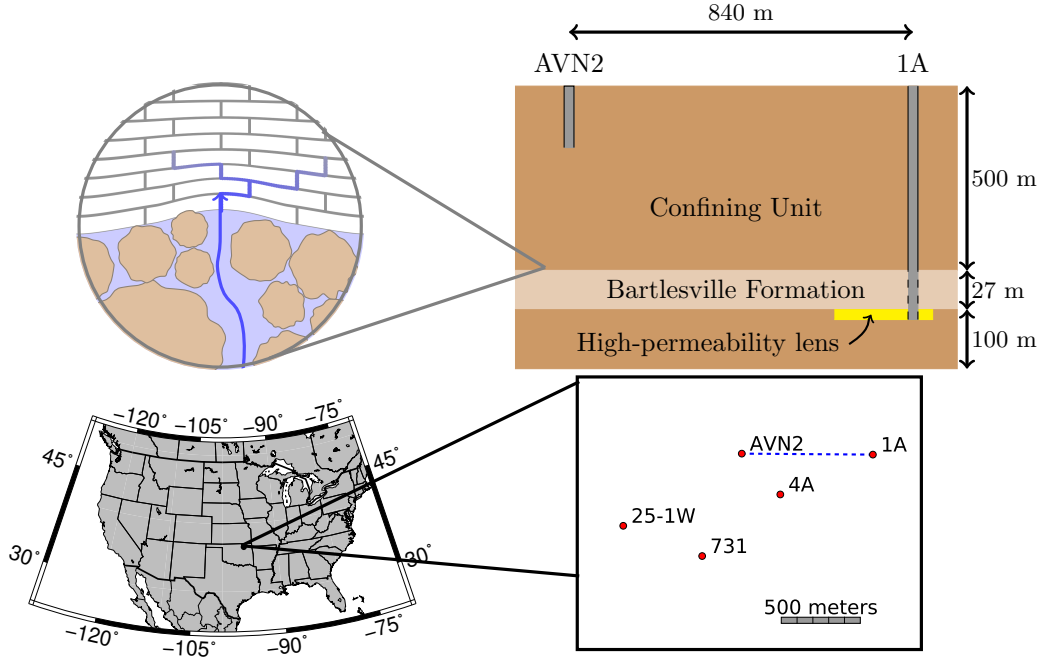


Figure 3.1: Location and conceptual model of the field site for 1A Shut-in test. Cross-section on top shows stratigraphy with contrasts in permeability (circle) and locations of well 1A and Gladwin strainmeter at AVN2. Location map of field site with wells included in the model.

3.4.2 Forward Modelling

The forward model is a poroelastic finite element partial differential equation solver called Geocentric [67, 68], which simulates fluid flow and rock deformation. For this analysis four horizontal layers are included in the model [Figure 5.1], an upper (500 m thick) and lower (100 m thick) confining unit, a permeable confined unit representing the Bartlesville sandstone (27 m thick), and a thin lens underlying the Bartlesville formation to represent the high-permeability fluvial lenses (5 m thick). A 16 km by 16 km model space is defined, where the outer region is coarsely meshed and a smaller interior region (4km by 4km) is much more finely meshed [Figure 3.2]. The side boundaries are defined by fixed fluid pressure and a mechanical roller for strain, meaning that the displacement normal to the boundary is zero. The lower boundary is specified to have zero flux, and a mechanical roller for strain. The upper boundary also specifies no flow and zero total stress.

Well geometries are defined explicitly as a cylindrical shell with a smaller, concentric cylinder inside. The volume of the outer shell represents the casing and has a bulk modulus and permeability approximating that of steel ($E = 200$ GPa, $\kappa = 10 \times 10^{-99}$ m²), while the inner core represents the

inner diameter of the well and has as low a bulk modulus and permeability as is computationally feasible ($K = 3 \text{ GPa}$, $\kappa = 1 \times 10^{-8} \text{ m}^2$). The well casing is screened within the permeable lens, and has a permeability of $\kappa = 1 \times 10^{-8} \text{ m}^2$. A flux is specified along the top surface of the inner cylinder, corresponding to the measured volumetric flow injected into each well.

The computational mesh consists of free quadrilateral elements over a region approximately 4 km across embedded in a larger domain of rectangular elements [Figure 3.2]. The free quad elements wrap around the wells. The mesh is swept vertically to create prismatic elements in 3D.

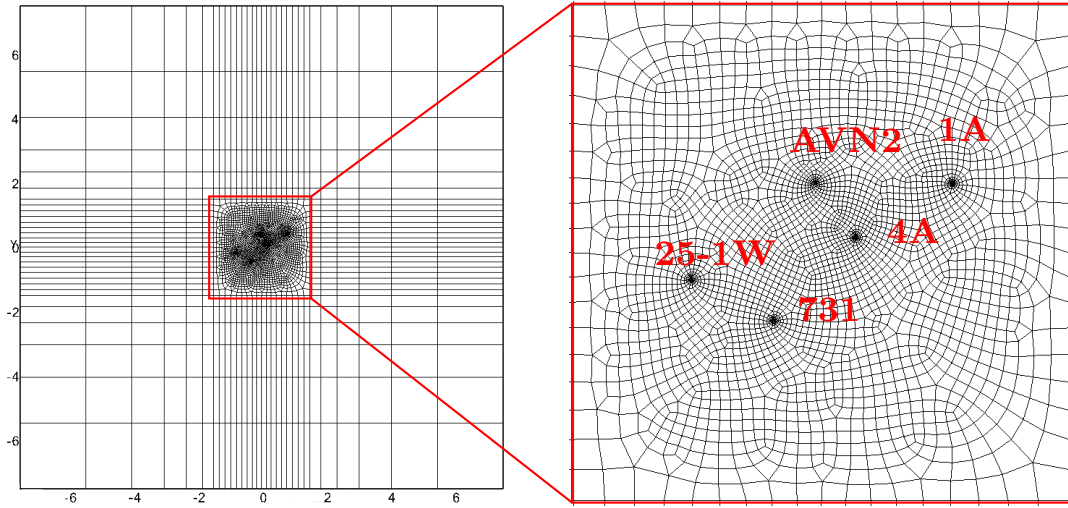


Figure 3.2: Computational mesh in plan view for entire model (left) and finely meshed region around well (inset).

The shut-in test at well 1A involved four separate injection wells and one observation well, so the mesh is refined around these five locations [Figure 3.2]. Simulating all four injection wells simultaneously was computationally intensive, and therefore the strain response of each well was simulated individually and the joint response was obtained by applying superposition to sum the contributions from all four wells. The validity of the superposition was tested by simulating two wells jointly using a mesh refined at three points (2 injection wells and one observation well). Those same two wells were then simulated separately using a mesh refined at only two points (injection well and observation well). Results from the two separate simulations were superimposed and compared to results from the simulation that included both wells together. The results [Figure 3.3] are nearly identical, but the superposition simulation ran much faster. As a result, the analysis was conducted using superposition of strains from individual wells.

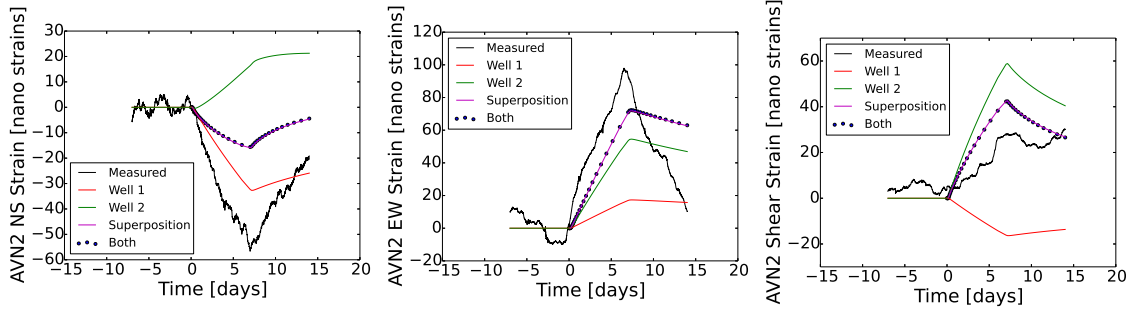


Figure 3.3: Measured and simulated strains as functions time. Effects of wells 1 and 2 simulated individually (red and green), and their superposition (magenta), along with effects of both wells simulated simultaneously (dotted), and compared to field data (black). (a) NS strain, (b) EW strain and (c) shear strain.

As the true field site potentially involves heterogeneities and preferential flow pathways at a very fine scale, a very large number of free parameters would be needed in order to fully describe the field site, resulting in a computationally intractable data-fitting problem. A set of simplifying assumptions were therefore made in order to approximate the field site using a manageable number of parameters. These include assumptions that the confining unit, lens and Bartlesville formation are internally homogeneous, that their Poisson's ratios and porosities are fixed and known ($\nu = 0.25, \phi = 0.2$), that their grain moduli are each that of quartz ($K_s = 42.9$ GPa), and that the pore fluid was single-phase water ($K_f = 2.15$ GPa). The domain is modelled as a continuous hydromechanical volume with no faults or fractures to accommodate or concentrate strain. The lens was assumed to be circular, and intersecting with the 1A injection well. Allowing for these assumptions, a data-fitting problem is therefore constructed with nine unknown parameters, namely the permeability (κ) and bulk moduli (K) of the lens, Bartlesville formation and confining units, and the geometry of the lens in terms of its centroid location (x, y) and radius (r).

3.4.3 Inverse Methods

A two-step inversion approach was used to identify an ensemble of reservoir properties that produce simulated data that fit the field observations. First, a Latin Hypercube was used to explore the parameter space as uniformly as possible while minimizing correlation between the initial samples of the model parameters. The simulation results obtained with these samples was then used in a multi-objective genetic algorithm (NSGA-II), which uses a Pareto ranking algorithm to explore the

tradeoffs between multiple data-fitting objectives for each of the three components of measured strain at AVN2. In order to perform the necessary simulations quickly, an object-relational programming approach was used to couple the inversion code to a diverse set of compute nodes on a university cluster computer, Open Science Grid (OSG), and Amazon Web Services (AWS).

Since the three measured components of strain (North-South, East-West, shear) represent three potentially contradictory data-fitting objectives, NSGA-II was modified to run as a series of sub-populations, alternating between fitting the tradeoff between the North-South and East-West strain components, then the North-South and shear components, then the East-West and shear components, then fitting all three components. After 19,134 models were run under the Latin Hypercube, and an 28,773 additional models were run using NSGA-II (191,628 simulations total, ie 4 single-well simulations per sample of model parameters), the marginal data-fitting improvement of each iteration was found to have dropped off significantly, and a variety of models were identified that closely fit all three datasets.

3.4.4 Data Analysis

These three contradictory strain components are visualized using a Pareto front approach, where each simulation is presented in terms of data-model misfits with one objective plotted against another. A singular value representing the overall data fit of each simulation is then calculated as the sum [Eq 3.1] of the squared prediction error for each of the strain time series, where the errors are weighted by the magnitude of the data noise (σ_n). The data noise is estimated by fitting a high-order polynomial to the time series data [Figure 3.3], and taking the sum of squared errors between the noisy data and the smooth polynomial. The weighted error of each simulation is plotted as a function of two parameters, cycling through the 36 possible pairs of 9 parameters ($\frac{9!}{2!7!}$) to characterize the error structure of the 9-dimensional parameter space being explored [two examples are shown in Figure 3.7].

$$WS = \sum_{i=1}^3 \frac{\sum (d_i - f_i(m))^2}{\sigma_{n,i}} \quad (3.1)$$

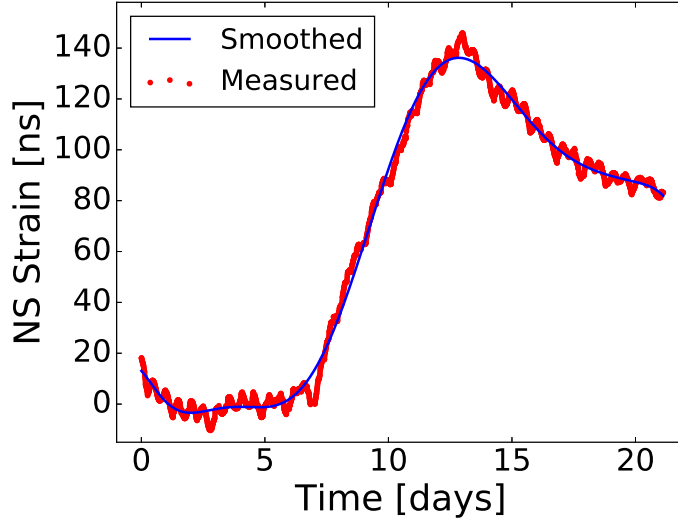


Figure 3.4: Measured NS Strain (red) and high-order polynomial fit (blue) as functions of time.

3.5 Results and Discussion

Of the 47,907 models evaluated, 5,490 unique sets of parameters were found which could explain all three available datasets [Figure 3.5] within the error bounds of EW strain $\varepsilon_{EW} \pm 25 \text{ n}\varepsilon$, NS strain $\varepsilon_{NS} \pm 15 \text{ n}\varepsilon$, shear strain $\varepsilon_{xy} \pm 10 \text{ n}\varepsilon$. These error bounds were selected to avoid overfitting of conceptual model noise. The best-fitting models, highlighted in black in Figures 3.5, 3.6, 3.8 have errors falling between the data noise $\sigma_{n,i}$ and these upper thresholds.

The distribution of simulations with small error is multi-modal when the confining unit is relatively permeable in the range of $k = 10^{-14}$ to 10^{-13} m^2 , for all values of the lens permeability (blue dots on right side of Figure 3.7). Another region of low error (blue dots) occurs where the confining unit permeability is in the range of $10^{-17} < k < 3 \times 10^{-16} \text{ m}^2$, and the permeability of the lens is relatively high ($3 \times 10^{-13} < k < 10^{-12} \text{ m}^2$). This distribution indicates that a reasonable data fit can occur for a relatively impermeable confining only if the lens itself is permeable, but if the confining unit is permeable then the fit is insensitive to the lens permeability.

The inversion identified a small set of parameters that fit the NS and EW strains well, as indicated by a Pareto front with an “L” shape (yellow points on the lower left of Figure 3.8a) oriented toward the origin, indicating that models have been found which explain both signals well. Similar shapes also occur for the other parameter sets (yellow points in Figures 3.8b and 3.8c) indicating small sets of parameters that fit the data with little tradeoff.

These results are encouraging, but they include parameter sets with a confining unit permeability that is fairly high (the band on the right side of Figure 3.7). The confining unit is shale and the permeability is expected to be low, so the parameter sets with high confining unit permeability are inconsistent with the geologic conditions in our prior understanding of the site. As a result, parameter sets with a high permeability confining unit were subsequently omitted. The Pareto front that requires the confining unit to be relatively impermeable is more rounded and sparse than when all the data are included (black dots on Figure 3.8). This indicates more tradeoff and a poorer fit to the data when the confining unit permeability is low.

One reason for a decrease in the data fit occurs when injection resumes on day 14 [Figure 3.5]. The observed strain decreases sharply at this time, but the simulated strain with an impermeable confining unit decreases only slightly. The mismatch that occurs when injection resumes on day 14 appears to be the reason why the models with an impermeable confining unit fit more poorly than the models with a more permeable confining unit.

Assuming that the permeability of the confining unit is low constrains the problem to a less multi-modal set of 661 solutions. The resulting lens geometries and physical properties are shown in Figures 3.10 and 3.9. These results indicate the log permeability of the lens is 2×10^{-13} to 1.5×10^{-12} m² (200 mD to 1500 mD), Bartlesville sandstone is 1×10^{-14} to 1×10^{-12} m² (10 mD to 1000 mD), and the confining unit is 3×10^{-17} to 1×10^{-15} m² (0.03 mD to 1 mD). The bulk modulus of the lens is $E=2$ GPa to 4 GPa, the Bartlesville sand is $E=12$ GPa to 16 GPa and the confining unit is $E=20$ GPa to 25 GPa [Figure 3.9].

The extent of the lens intersected by well 1A is 800 m to 1000 m and the centroid is to the southwest of well 1A [Figure 3.10a]. For comparison, the isopach map of the lens thickness [Figure 3.10b] suggests that the lens extends approximately 500 m to the northeast, but the extent to west is limited to less than 100 m.

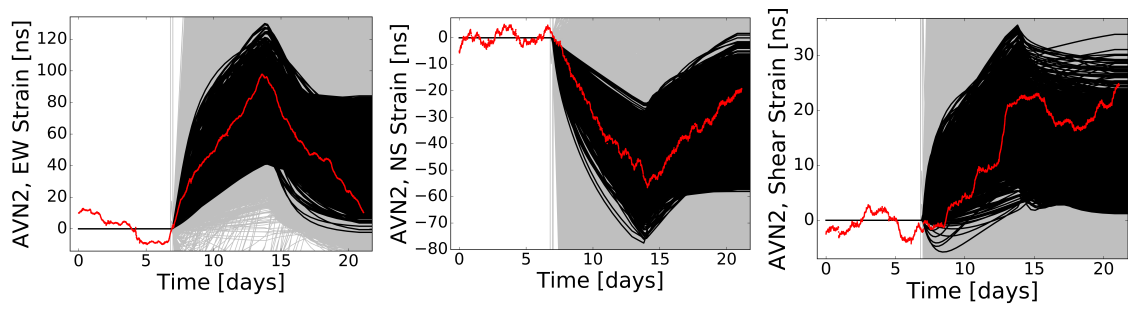


Figure 3.5: Strain as a function of time for 1A Shut-in Test. Measured strain data (red), simulation results (gray). Best-fitting simulation results in black. Models with permeable confining layer included.

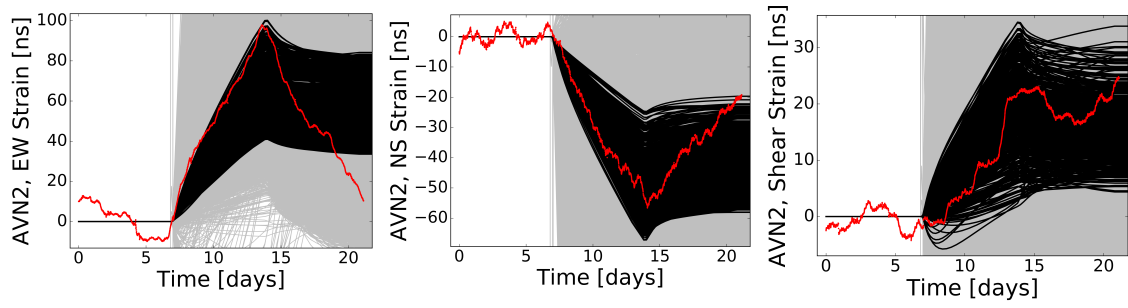


Figure 3.6: Strain as a function of time for 1A Shut-in Test. Measured strain data (red), simulation results (gray). Best-fitting simulation results in black. Models with permeable confining layer excluded.

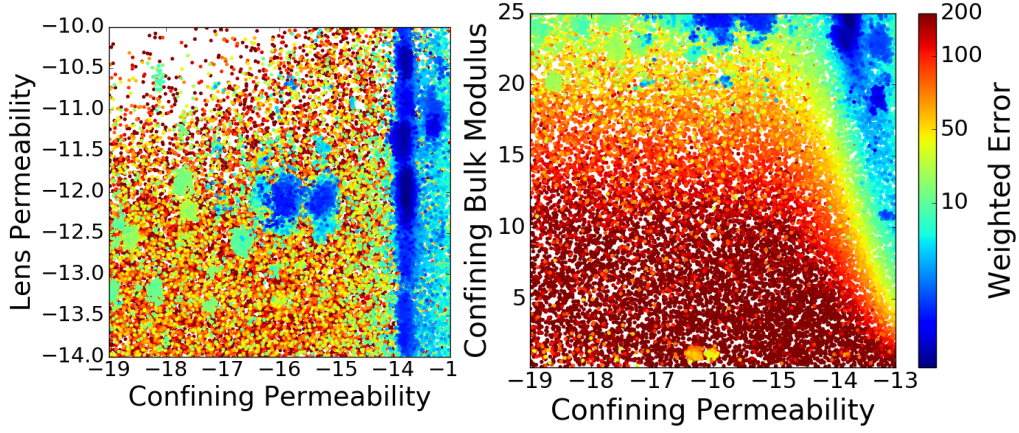


Figure 3.7: The weighted error (Eq 3.1) of each simulation as a function of (a) the confining unit permeability and lens permeability, and (b) the confining unit permeability and confining unit bulk modulus. Blue points correspond to the black lines in Figure 3.5. Note that this plot projects a 9-dimensional parameter space to a 2-dimensional plane, and therefore some of the variation shown is due to the other 7 parameters.

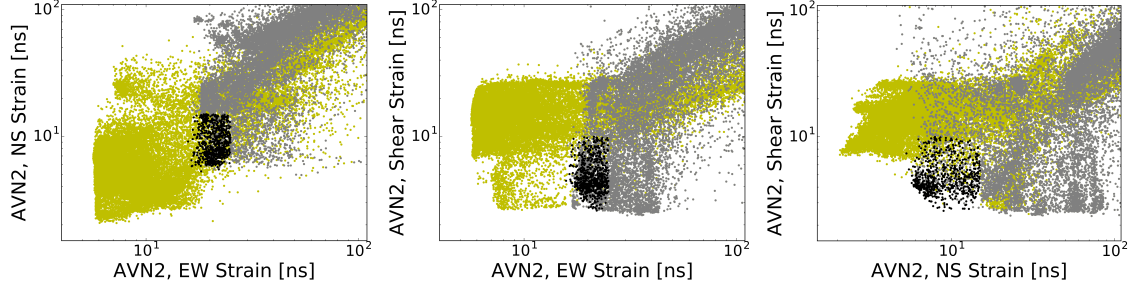


Figure 3.8: Data-model misfits for pairs of strain components (Pareto plots) with (a) East-West strain against North-South, (b) East-West against shear strain, and (c) North-South strain against shear for 1A Shut-in test. Yellow dots indicate models where the confining unit has a high permeability (greater than 10^{-15} m^2), gray dots indicate models where the confining unit has a lower permeability, and black dots indicate models with low confining unit permeability and also fit the measured data within the selected error bounds (corresponding to black lines in Figure 3.6).

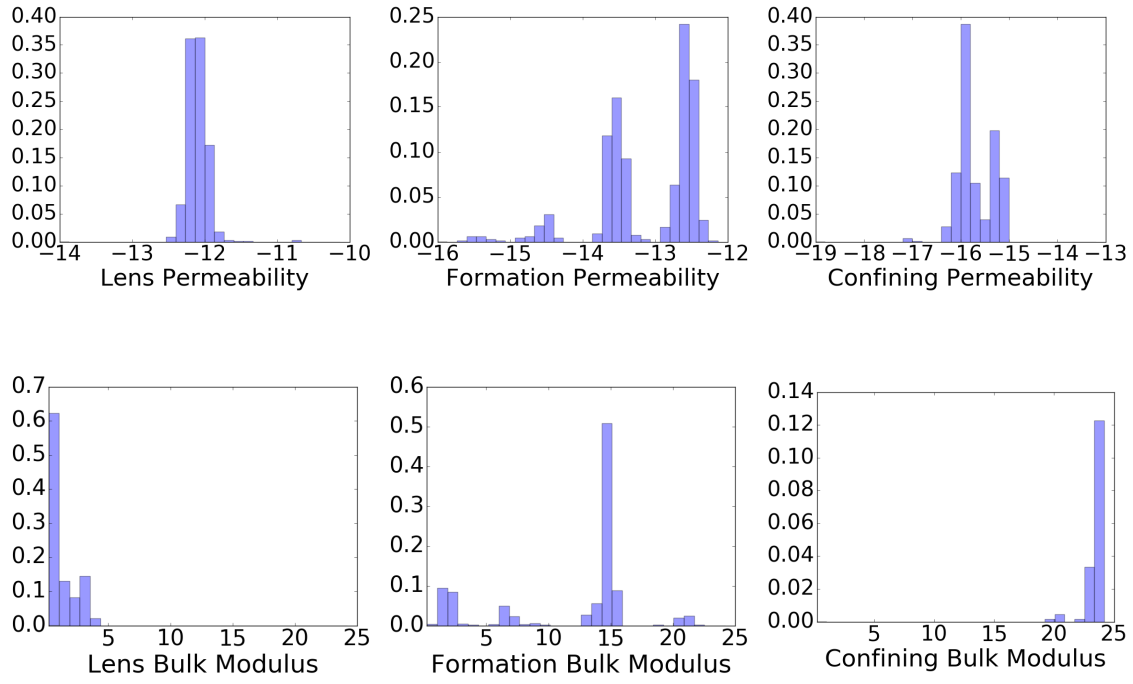


Figure 3.9: Histograms of the best-fitting parameters for low permeability confining unit from 1A-Shut-in test. Permeability as log base 10 m^2 . Bulk moduli are in GPa.

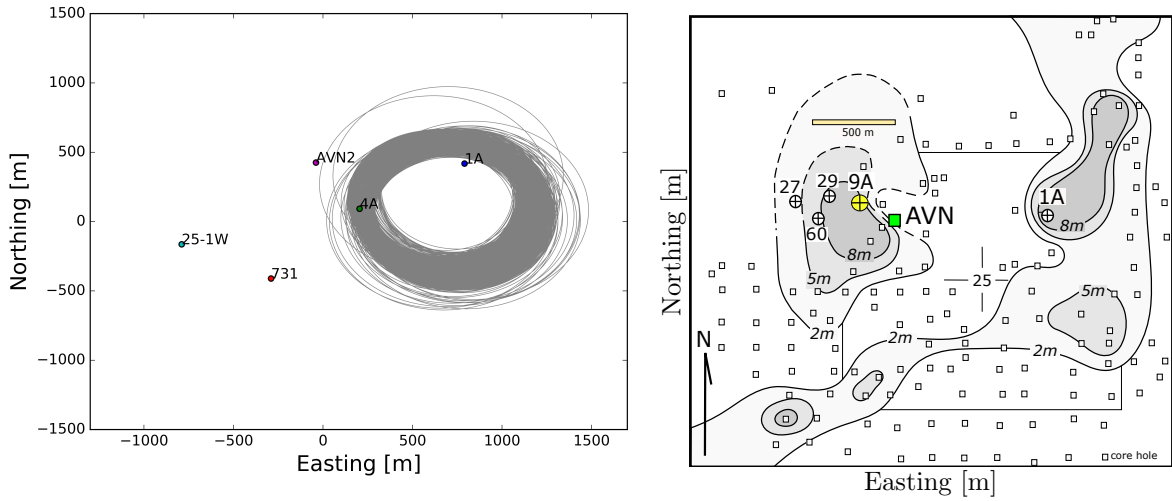


Figure 3.10: (A). Lens geometries of the best-fitting models are shown with well locations for comparison. Note that these lenses correspond to the black lines in Figure 3.5. (B). Lens thickness isopach developed by Grand Resources Oil company.

3.6 Conclusion

In this study, a numerical proof-of-concept was successfully field tested and it was demonstrated that shallow strainmeter data is sensitive to the geometry and physical properties of distant, deep, hydraulically confined geologic structures. While non-uniqueness issues and data-fitting tradeoffs remain, this type of data can dramatically reduce model uncertainties while requiring only inexpensive wells completed in the shallow subsurface. This can allow stakeholders to develop an improved understanding of a geologic formation, and determine ideal drilling locations and pumping schedules.

Chapter 4

Combining wellbore strainmeter and pressure transducer data for improved reservoir characterization

4.1 Abstract

In order to infer the structure and material properties of a deep confined reservoir, measurements from borehole strainmeters completed in the shallow (30 m depth) subsurface were combined with pressure transducer data completed in the deeper (500 m) confined reservoir. While including pressure data in the analysis dramatically reduces uncertainties, strain data alone is found to be highly sensitive to the subsurface parameters and has a very similar information content. In addition, drilling and installation of the strainmeter observation well is much cheaper and safer since the pressurized reservoir does not need to be punctured.

4.2 Introduction

Projects involving exploration of oil and natural gas, mine dewatering, geologic carbon storage and disposal of produced water require detailed knowledge about the geologic structure and composition of the various rock layers at a given field site. The parameters of interest in these systems

can influence both the fluid pressure at depth as well as the mechanical strain and deformation in the shallow subsurface [27, 28]. However, collecting data on fluid pressure carries a greater cost and risk since the drilling process is more involved for deeper wells, and puncturing into a confined formation releasing fracking fluids into surficial drinking water aquifers, or releasing stored carbon from a geologic storage operation. In order to mitigate these risks, it is critical to develop methods of characterizing the subsurface using minimally invasive wells.

As a pressurized fluid moves through a porous medium, it influences the mechanical response of the medium. Simultaneously, the elastic behavior of the medium can impact the fluid flow by changing the pore diameters and tortuosity of the pore skeleton itself, as well as the propagation of pressure throughout the system.

In this study, borehole strainmeter and pressure transducer data are combined to characterize the subsurface, and their information value for site assessment and monitoring of geologic reservoirs is compared.

An important and unique aspect of this work compared to prior studies is that the strainmeter is installed in a shallow borehole well above a target formation, while pressure transducers are completed within the target formation itself.

4.3 Methods

In the following sections, the field experiment and data collection process are described as well as the approach to numerically approximating the field site’s hydromechanical behavior. Data-fitting methods are then discussed in terms of both the particular inverse methods employed and the computational workflow needed to accomplish the hydromechanical simulation runs.

4.3.1 Site Description and Pumping Experiment

An active oil field near Avant, Oklahoma was selected as the study area. This field site includes a sandstone layer called the Bartlesville formation, with a thickness of approximately 30 meters and at a depth of approximately 500 m. This layer is confined by shale layers above and below, and is underlain by permeable lenses which are used for oil exploration and development, and re-injection of produced water.

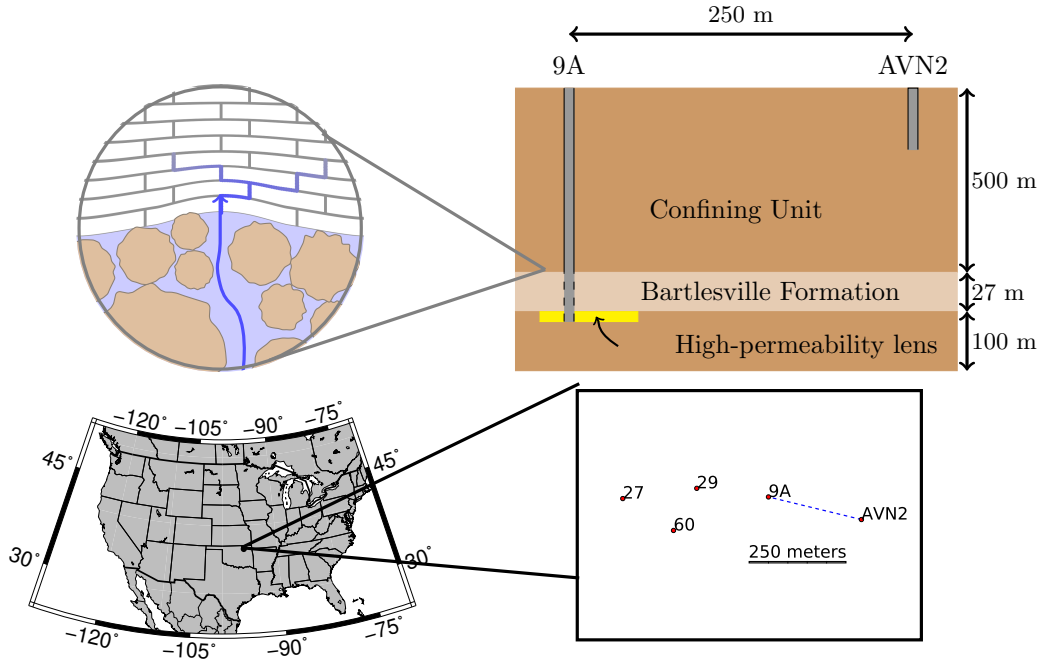


Figure 4.1: Active injection wells in the Avant oil field are shown in map view. Geological conceptual model is shown in cross section.

This field experiment included 1 active injection well and 4 observation wells [Figure 5.1]. A Gladwin strainmeter was installed in a shallow well at AVN2 (depth of 30 meters) and measured strain along the East-West, North-South, and vertical axes. Pressure was measured in wells 27, 29, and 60, which are cased throughout the confining unit and screened in the high-permeability lenses. Well 9A was pumped at a varying flow rate for approximately six days before flow was stopped, after which recovery was monitored for seven more days.

4.3.2 Forward Model

Hydromechanical response to injection is simulated using Geocentric, a poroelastic finite element partial differential equation solver [67, 68]. The mesh is composed of four layers [Figure 5.1] representing the upper confining unit (500 m thick), Bartlesville formation (27 m thick), permeable lenses (5 m thick) and lower confining unit (100 m thick).

The permeable lenses are laterally discontinuous, while the other layers are uniform throughout the NS-EW plane. The model space is 16 km by 16 km, with the inner region (4 km by 4 km) meshed finely compared to the outer region [Figure 4.2]. The mesh elements are made up of free

quadrilateral elements, which are swept vertically forming hexahedral elements. The side boundaries are assigned a constant fluid pressure and a mechanical roller condition ensuring that they cannot move normal to the boundary. The lower boundary is specified as a zero flux condition, and a fixed vertical position allowing it to deform along the xy-plane only. The upper boundary has a no-flow condition and fixed zero total stress.

Injection and observation wells are meshed as a pair of concentric cylinders of radius 2.0 m and 1.8 m. The outer shell has the material properties appropriate for steel casing ($E = 200$ GPa, $\kappa = 10 \times 10^{-99} \text{ m}^2$), while the interior of the well has material properties that approximate free air as closely as the solver will efficiently allow ($E = 3$ GPa, $\kappa = 10 \times 10^{-8} \text{ m}^2$). Within the permeable lens, the steel casing is screened and therefore has a permeability of $\kappa = 10 \times 10^{-8} \text{ m}^2$. The injected volumetric flux is specified along the top of the inner cylinder.

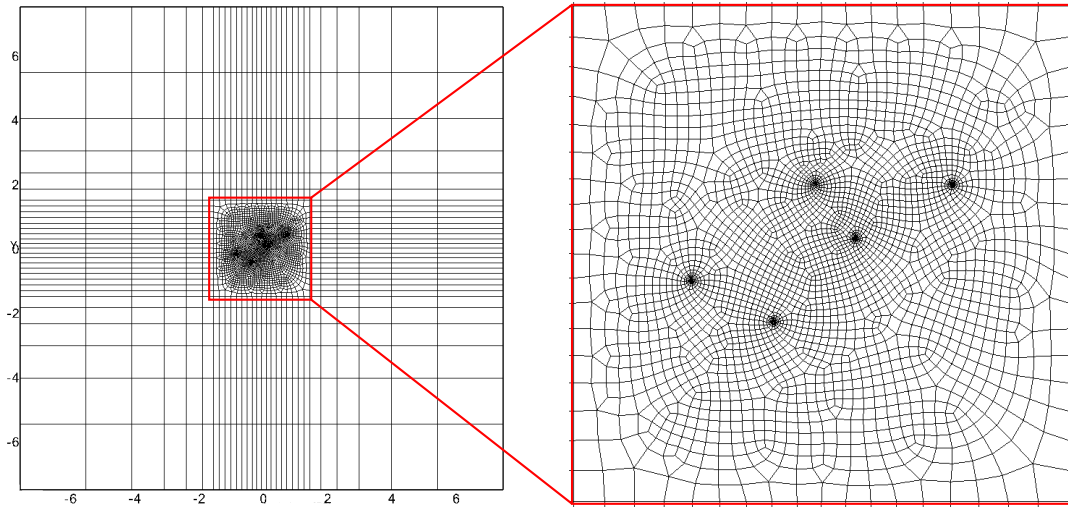


Figure 4.2: Outer, coarse mesh is shown with inset finely meshed region.

As the variations in flow rate were at a much finer timescale than would be reasonable for the hydromechanical simulation, a smooth function was constructed to closely approximate the volume of water injected at any given moment [Figure 5.3b] without precisely matching the instantaneous injection rate history [Figure 5.3a].

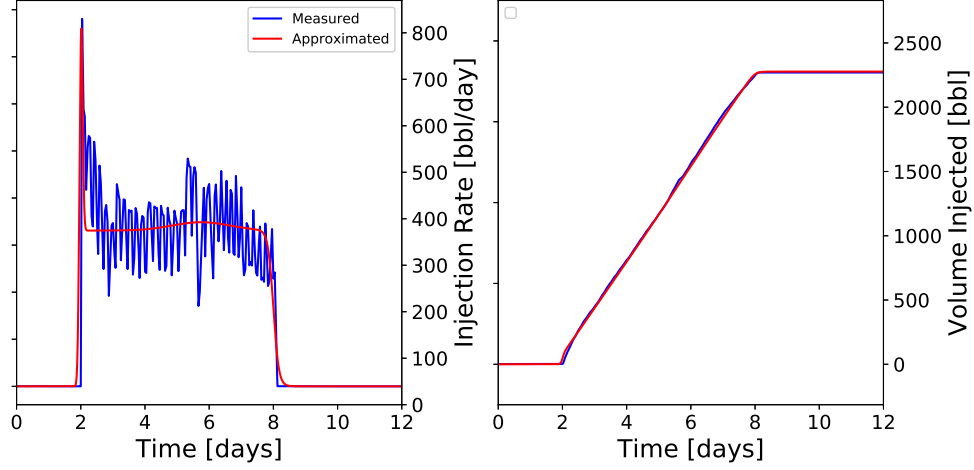


Figure 4.3: The injection rate over time and total injected volume over time are shown for both the measured flow, and the smooth function approximating the measured flow.

4.3.3 Inverse Algorithms

The goal of this inversion is to use the 6 available timeseries of strain at AVN2 and pressure data at wells 27, 29 and 60 to make inferences about the structure and composition of the subsurface. Three homogeneous poroelastic materials are used to represent the confining unit, Bartlesville formation, and permeable lens. It is also assumed there are no faults or fractures in any of the layers, that the bulk modulus of the solid grains is $K_s = 42.9$ GPa (quartz), and that the pore fluid is water only with no air, oil or natural gas present ($K_f = 2.15$ GPa). It is also assumed that the lens was either circular or elliptical, and could thus be parameterized using an x,y centroid location and either a radius or a rotational angle, semi-major and semi-minor axis. Unknown parameters in the model therefore include the permeability, bulk modulus, porosity and Poisson's ratio of these three materials, as well as the geometric parameters of the lens, for a total of 15 parameters in the circular lens case and 17 in the elliptical lens case.

The optimization proceeded in stages. The circular lens case was analyzed first, the results of which were used as a starting point for the elliptical scenario. A Latin Hypercube was used to sample the parameter space ($n=12,046$) as evenly as possible while minimizing any correlations between parameters. This helps prevent the inversion from neglecting large, unexplored regions of the parameter space where the true model might lie. Next, a multi-objective genetic algorithm (NSGA-II, [16]) was used to search for optimal parameter sets to fit the data.

Due to the potential for sensor drift evident in the data, and the many simplifying assumptions built into the conceptual model, substantial tradeoff between data-fitting objectives is possible. For example, the inversion might easily find models which explain one component of strain, but strongly contradict the others. Or the inversion might find models that fit the pressure data nicely but contradict the strain data. To mitigate the inherent difficulty of exploring a 15-dimensional parameter space using 6 potentially contradictory datasets, the standard NSGA-II algorithm was modified to run many subpopulations in parallel, with roughly a third of them attempting to strike a balance between fitting only two of the datasets at a time, another third attempting to satisfy three data-fitting objectives at a time, and another third attempting to fit all six datasets simultaneously. This breaks the inverse problem into many smaller, more solvable problems, and allows good solutions to the easier problems to be continuously fed into the six-objective inversion as candidate solutions.

After running 10,017 simulations using the NSGA-II approach and the 15-parameter circular lens model, a second Latin Hypercube search ($n=1,219$) was then performed using the 17-dimensional parameter space implied by an elliptical lens. The best-fitting circular lenses were then converted to an elliptical parameterization, and used to supplement this second Latin Hypercube sweep of the parameter space. When these simulations were complete, the NSGA-II algorithm was continued for an additional 20,178 simulations.

4.3.4 Distribution of Computational Effort

Stochastic model calibration can require on the order of 10s to 100s of thousands of simulations, with each simulation in this study requiring an average of 20 minutes to run [Figure 4.4] on a 4-core, cpu-optimized machine. An efficient workflow [Figure 5.4] was therefore needed to run the simulations in parallel as much as possible, distributing this computational effort over a wide variety of compute nodes.

The approach is cloud based with a head node located on Amazon Web Services that manages the overall progress of the optimization. The head node constructs simulation input files and distributes them to compute nodes. To this end, a MySQL relational database [Figure 4.6] was developed which organizes the incoming field data, simulation inputs and outputs, and lineage relationships between simulations. An Amazon S3 (Simple Storage Service) bucket is used to store these input files and make them available online to a heterogeneous set of compute nodes, while Amazon

SQS (Simple Queue Service) maintains the integrity of the distributed queue and ensures that no two compute nodes begin simulating the same input file. The composition of compute nodes varies depending on availability, but are drawn from Clemson University's cluster computer (Palmetto), Open Science Grid (OSG), and Amazon Web Services (AWS). The database also houses the measured field data, and includes information about the parameterization of the inverse problem, prior statistical distributions of the parameters, and tuning parameters for the inversions. As simulations are completed, the incoming simulation results are integrated into the overall inversion and used to inform the next set of input files to be simulated.

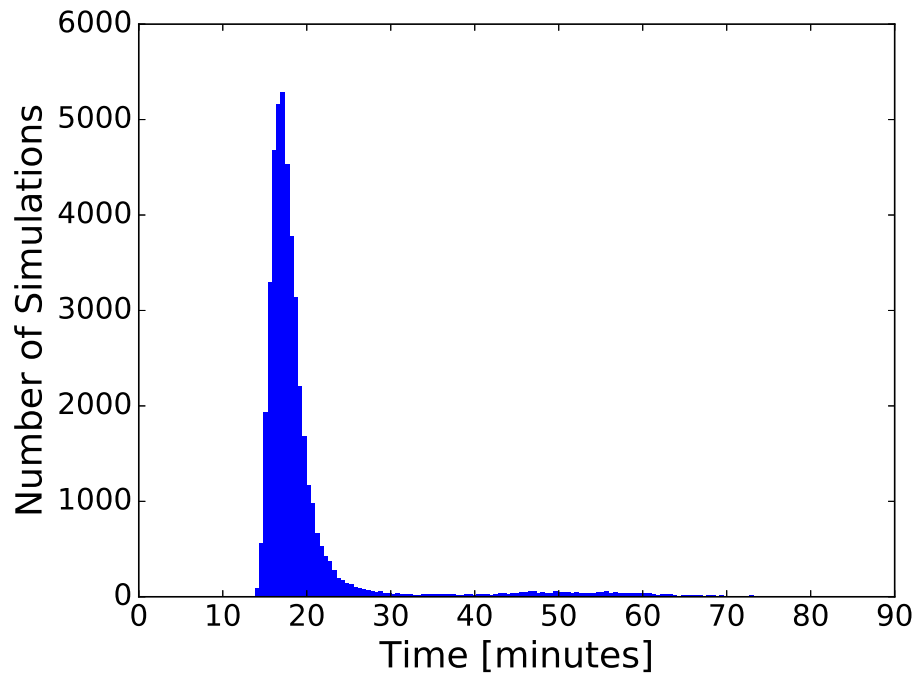


Figure 4.4: A histogram of simulation runtimes is shown.

- MySQL: Structured data
- Python:
 - Inverse methods
 - Input file assembly
 - Data transfer
 - Post-processing
 - Visualization
- SQS/S3: Cloud storage

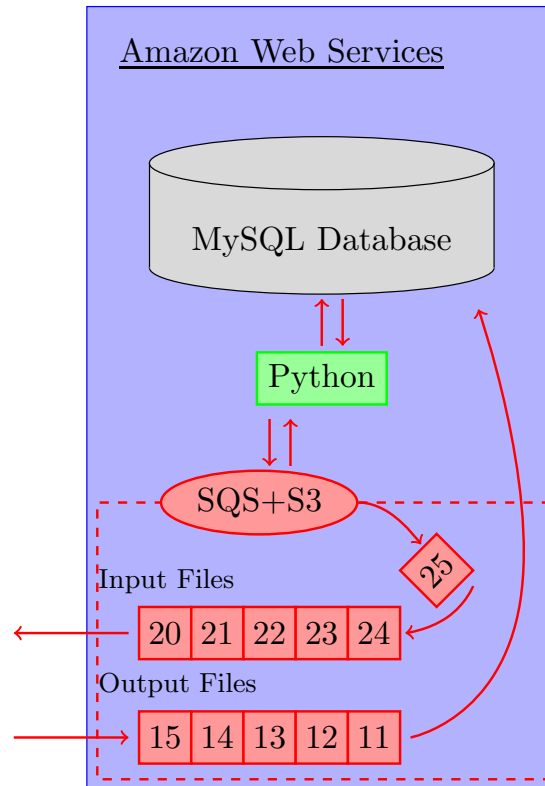
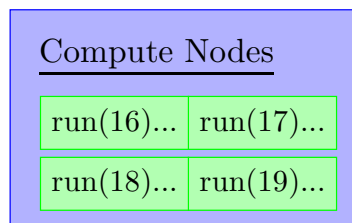


Figure 4.5: Illustration of parallel model calibration workflow. Inversion data is stored in MySQL database, and Python code accesses this data to produce simulation input files and store them in an AWS S3 bucket. Compute nodes retrieve these input files, run them and produce output files, which are also stored in the S3 bucket and later retrieved by the database.

While the full mapping between the 15–17 dimensional parameter space and 6-dimensional objective space is impossible to examine graphically, it can be simplified in several ways to draw out the important dynamics. To summarize the overall data fit of each simulation as a function of its model parameters, the weighted sum of the six misfits weighted by the magnitude of their data noise (σ_n) is computed. The data noise is estimated by fitting a high-order polynomial to the data [Figure 4.7], and taking the sum of squared errors between the noisy data and the smooth polynomial. Therefore a weighted sum of 6.0 would indicate that the model fit all six measured datasets at least as well as the smooth polynomial.

$$WS = \sum_{i=1}^6 \frac{\sum_{j=0}^6 (d_{i,j} - f_{i,j}(m))^2}{\sigma_{n,i}} \quad (4.1)$$

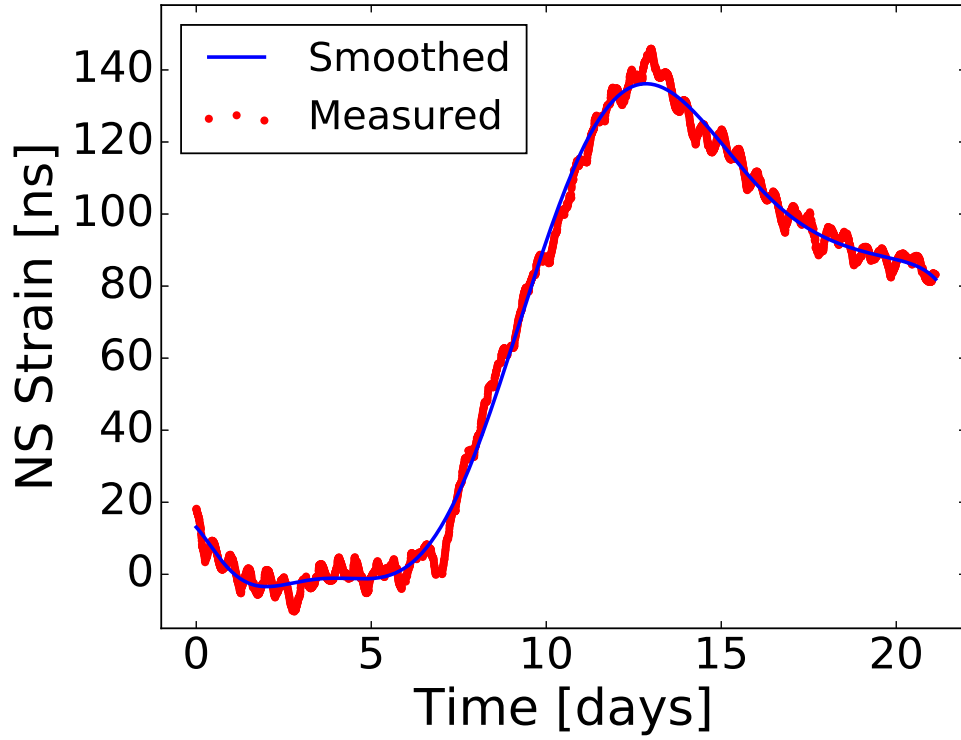


Figure 4.7: Measured NS Strain is shown compared to a high-order polynomial fit of the data.

4.4 Results and Discussion

In both the circular [Figure 4.8] and elliptical [Figure 4.9] cases, a set of models were found which fit all available datasets within a selected margin of error. Many different models were found early in the inversion which would fit all three strain components, but the algorithm had a much harder time finding models which also explain the pressure signals [Figure 4.8]. In general, the models which fit the strain data had too low of a pressure response, possibly indicating that the lens volume was too large to pressurize the entire region, and that a circular geometry would not allow a small enough lens that still contains all three pressure observations wells.

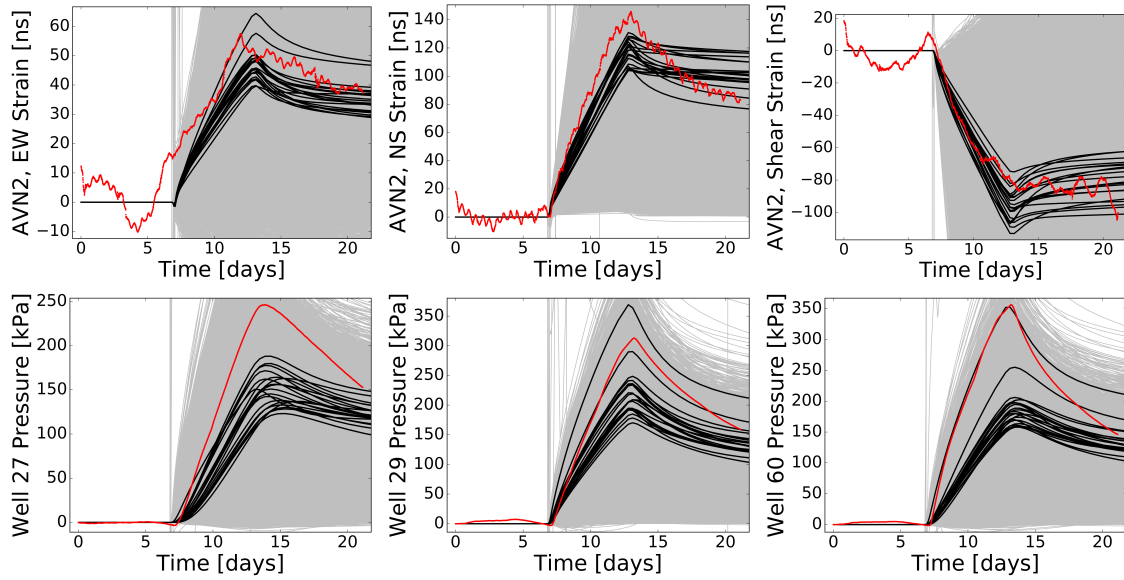


Figure 4.8: Simulation results from the circular parameterization inversion are compared to their corresponding field measurements (red lines) for three components of strain, and three pressure transducer locations. Simulation results are shown in gray, while best-fitting results are highlighted in black.

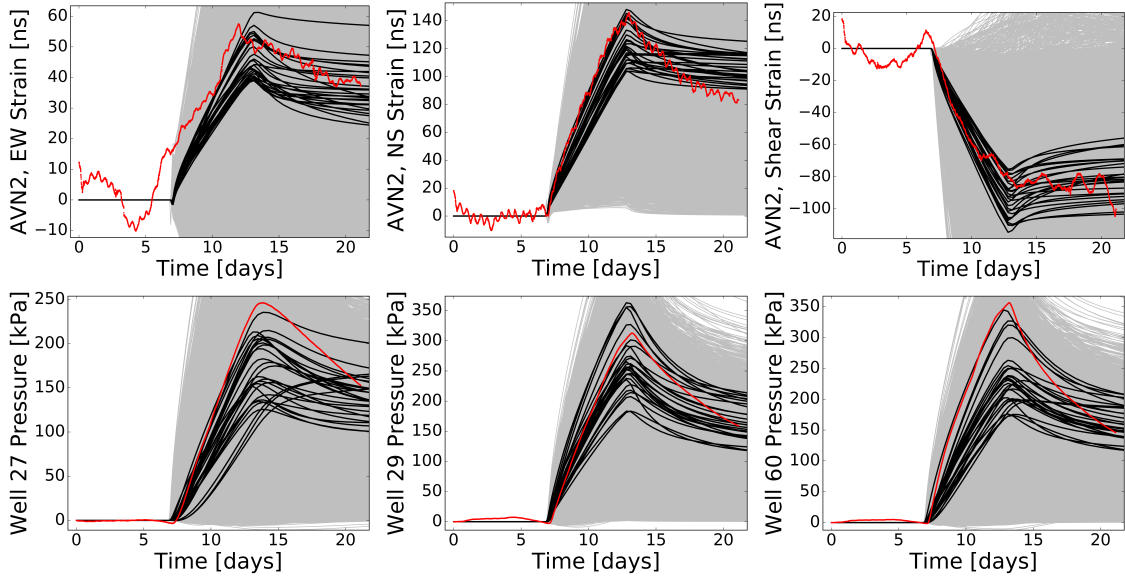


Figure 4.9: Simulation results from the elliptical parameterization inversion are compared to their corresponding field measurements (red lines) for three components of strain, and three pressure transducer locations. Simulation results are shown in gray, while best-fitting results are highlighted in black.

While the tradeoff between all six objectives is impossible to examine graphically, 2-dimensional tradeoffs can be examined by selecting two objectives at a time and plotting the corresponding data-model misfits against each other [Figure 4.10]. For example the tradeoff between East-West and vertical strain components shows an “L” shape near the origin, indicating minimal tradeoff, or that many simulations were found which fit both the East-West and vertical strain timeseries equally well. By contrast North-South and vertical strain show a stronger tradeoff near the origin, meaning that beyond a certain point (10–12 nanostrain), fitting one strain component marginally better requires fitting the other marginally worse.

The NSGA-II algorithm is explicitly designed to identify simulations along the leading edge of this tradeoff surface by making random combinations and mutations of these “equally good” parameter sets, filling in the gaps. However the tradeoffs involving pressures were often much more difficult to sample densely. This may indicate that pressure is much more sensitive to these parameters, such that a slight mutation to a model along the tradeoff surface would result in a very different pressure signal while leaving the strain data reasonably unchanged.

While the circular and elliptical parameterizations have similar performance in terms of

fitting strains, the elliptical parameterization was better able to balance the tradeoff between strain and pressure objectives. This may be because the elliptical lens allowed for a smaller overall lens volume while still encompassing all four observation wells.

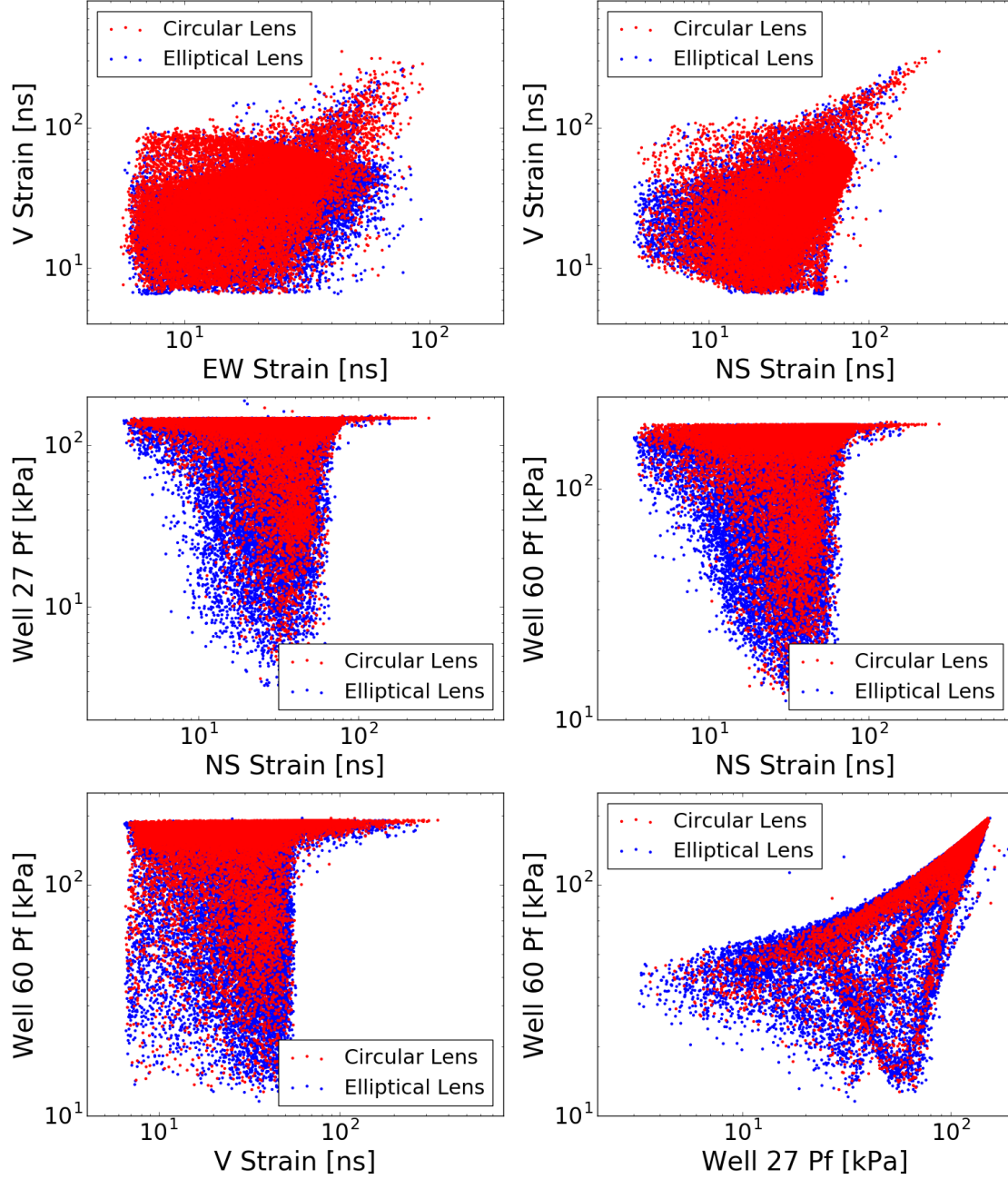


Figure 4.10: Data-model misfits are shown for all Geocentric simulations, for several pairs of objectives. Red dots indicate circular lens simulations, blue dots indicate elliptical lenses.

The weighted sum of data misfits is plotted as a function of pairs of model parameters in Figure 4.11. In order to compare the value of pressure and strain data, the data misfits calculated using only the strain signals are shown in the lower portion of the figure, whereas the misfits calculated using both the pressure and strain signals are shown in the upper portion of the figure. Note that the porosity and Poisson’s ratio were observed to be the least sensitive parameters, and therefore are not shown. Similarly, the geometric parameters were the most well-constrained and exhibited minimal tradeoff, and are therefore presented separately in map view [Figure 4.13]. The formation, lens and confining units permeabilities each were shown to have a single, unimodal range that explains all six datasets well [Figure 4.11, diagonal].

These results indicate that a \log_{10} permeability (m^2) of approximately -18 to -17 for the confining unit, -16 to -15 for Bartlesville sandstone, and -13.5 to -12.5 for the lens. The confining unit shows a bulk modulus of approximately 6-8 GPa. However, the bulk modulus of the lens and formation exhibit a multi-modal distribution. This multi-modal character may indicate that the data is simply less sensitive to the formation and lens moduli, or that a sparsely-explored tradeoff surface exists between the compressibilities and permeabilities. This tradeoff surface may have been sparsely-explored because it is topologically complex, because it represents a very narrow sub-region within the parameter space, or simply because greater sensitivities to the material permeabilities and lens geometry tended to dominate the inversion.

Strain data alone [Figure 4.11, (diagonal)] yield parameter distributions with modes that are similar to what we observe when pressure is considered [Figure 4.12, (diagonal)]. While the pressure data was better able to find unique solutions with less uncertainties, near-surface strain data was much safer and less expensive to collect as the observation point was much shallower and did not intersect the target formation. In a hydraulic fracturing or carbon sequestration operation, being able to study and monitor a target formation without puncturing an observation well into it can be invaluable.

While the measurement noise of these strainmeters themselves is much smaller than the strain signals observed here, it is important to note that conceptual model errors are likely considerable. There is therefore a danger of overfitting model noise. In future work, it will be important to run similar pumping tests in a repeated fashion in order to distinguish random drift from the characteristic signal produced by a given pumping schedule.

In addition, improved hydromechanical solvers may be needed in order to effectively simulate

the full range of physical processes that may be influencing near-surface strain data. For example, while this analysis assumes a single-phase fluid, it may be that multi-phase flow is necessary to adequately explain all of the data. Likewise, fracture behavior may account for a component of the signals that we see here, concentrating stress in unexpected areas while releasing it in others. Another critical step in improving our ability to characterize the subsurface will be the integration of additional strainmeters. Since including more complex physics simulations will necessarily increase the number of unknown parameters, adding more objectives will be critical in keeping geomechanical inversions computationally tractable.

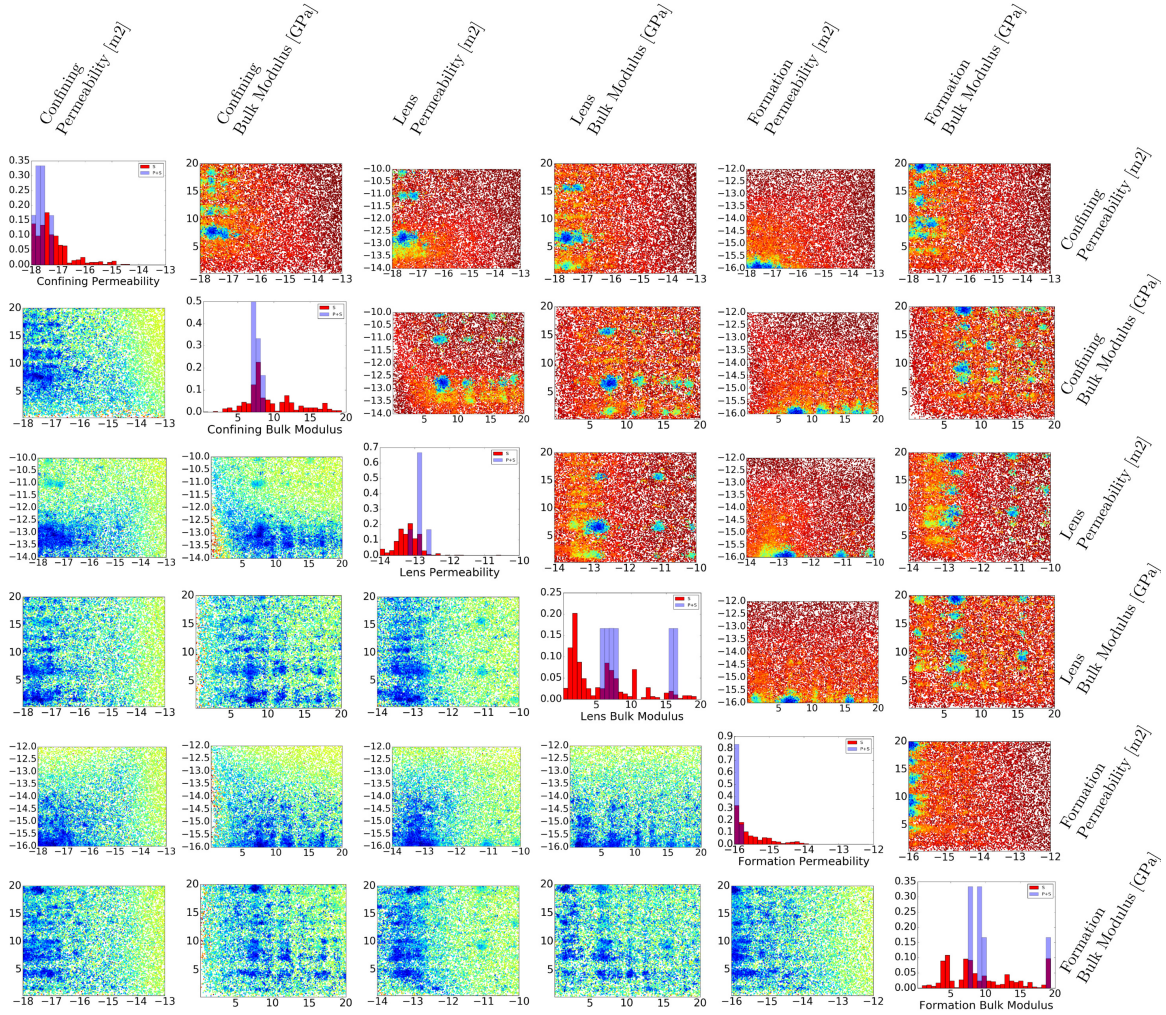


Figure 4.11: Parameter values for circular parameterization are shown as scatter plots in upper right and lower left corners, where color represents a weighted sum summarizing how well all six datasets (upper right) are fitted, and how well the three strain datasets are fitted (lower left). Blue points represent the best-fit models. Histograms along diagonal show the parameter distribution of best-fitting models, according to both the strain alone (S), and pressure and strain combined (P+S).

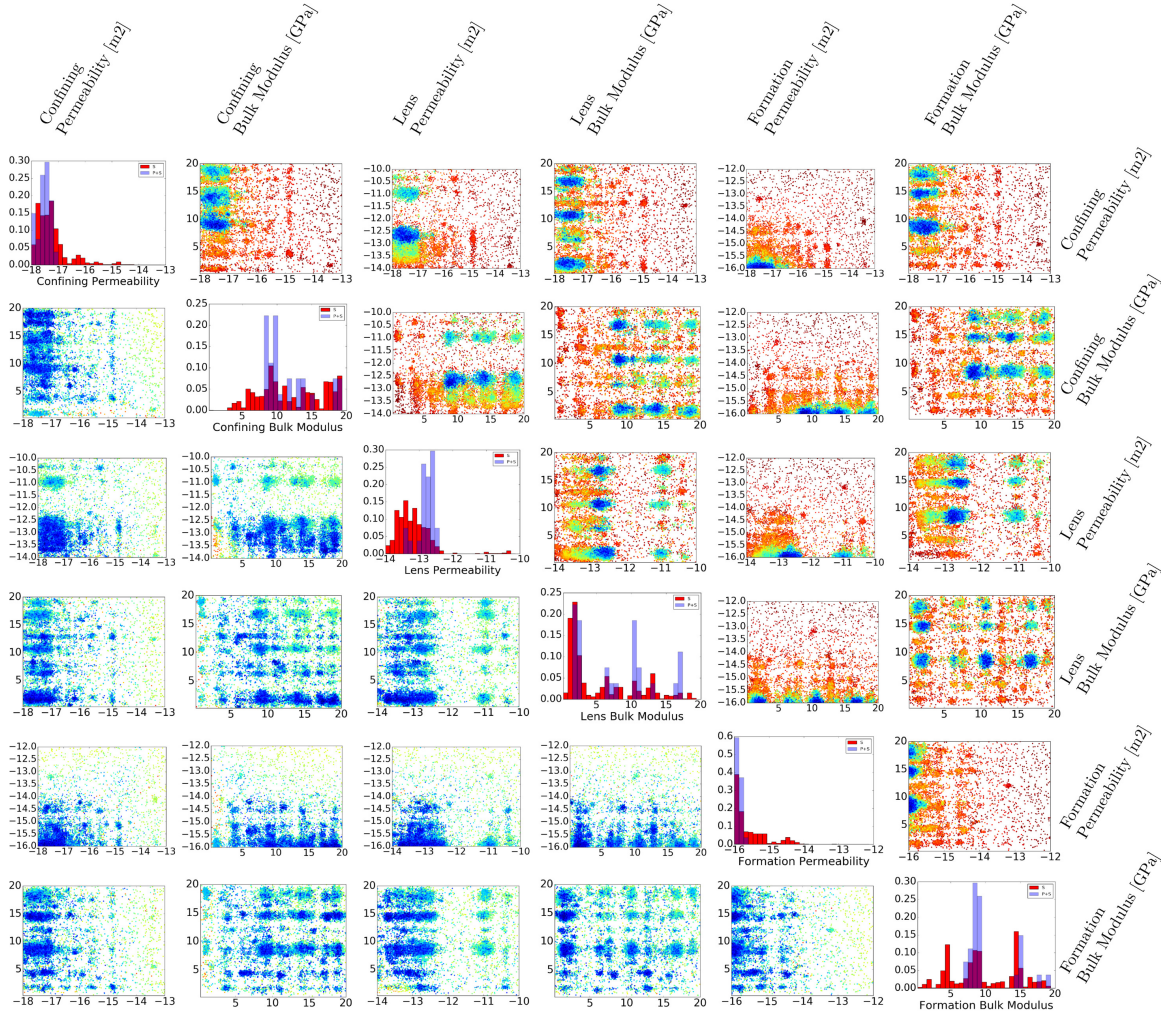


Figure 4.12: Parameter values for elliptical parameterization are shown as scatter plots in upper right and lower left corners, where color represents a weighted sum summarizing how well all six datasets (upper right) are fitted, and how well the three strain datasets are fitted (lower left). Blue points represent the best-fit models. Histograms along diagonal show the parameter distribution of best-fitting models, according to both the strain alone (S), and pressure and strain combined (P+S).

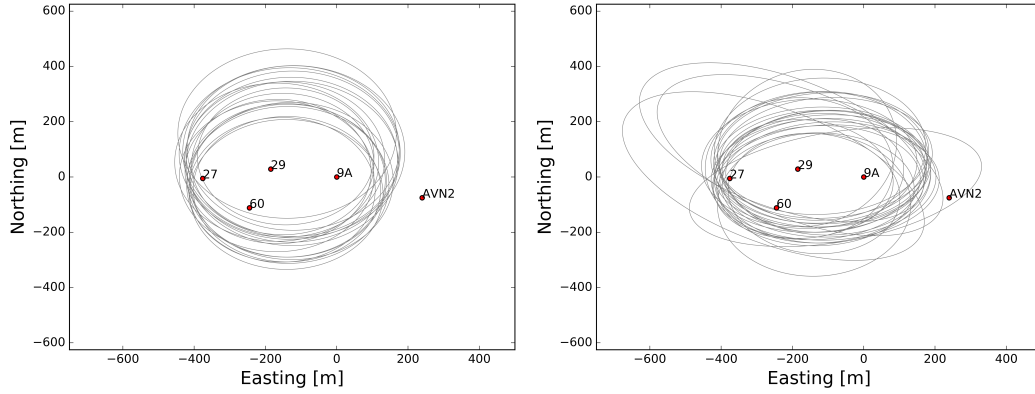


Figure 4.13: Best-fitting lens geometries are shown for both circular and elliptical parameterizations.

4.5 Conclusion

In this study, pressure and strain data were used to determine the geometry and properties of rock layers in the subsurface. While pressure data was taken within the target formation, strain data was measured from the shallow subsurface allowing one to much less invasively study the subsurface. While strain data taken from the shallow subsurface was sufficient to locate and characterize a potentially oil-bearing permeable lens, installing deeper observation wells in the target formation allowed for much lower uncertainties in estimating of the reservoir properties.

Both strain and pressure data were shown to have a strong sensitivity to the shape of the lens, and therefore the results obtained are biased by any geometric assumptions made by the modeller. In future work, these geometric assumptions will be obviated using a dynamic parameterization that allows the lens shape to vary during the optimization rather than assuming a lens shape.

Chapter 5

Using geomechanical data to image heterogeneous subsurface flow systems

5.1 Abstract

The structure and material properties of a confined permeable lens were studied using in-situ pressure transducer data combined with remote strainmeter taken in the shallow subsurface. A novel inversion approach was used to fit these datasets using Geocentric, a hydromechanical finite element solver.

The role of simplifying assumptions in inversion was investigated by initially using a fixed parameterization which made strong assumptions about the lens shape, and by then successively relaxing these assumptions. It was initially assumed that the permeable lens was circular, and an inversion was performed to find the approximate location, size and material properties of the lens. The lens was then assumed elliptical, and a second, more focused inversion was performed to find the long axis and orientation while fine-tuning the location, size and material properties. The lens shape was then allowed to vary dynamically and an image was constructed of the most likely lens shape as well as its material properties.

5.2 Introduction

Real subsurface flow systems are highly heterogeneous, and minor variations in the configuration of different rock layers can alter flow patterns significantly. Structures such as braided river channels to fracture networks can create preferential flow pathways at many different scales. Any model parameterization capable of allowing for all these possibilities would require many (ie, millions) unknown parameters, and the resulting inverse problem would be computationally challenging or potentially intractable. It is therefore common to make simplifying assumptions that limit the potential shapes of heterogeneous rock formations, reducing the effective size of the parameter space and allowing one to construct a tractable inverse problem.

One approach to implementing these simplifying assumptions is the object-based method [17], where heterogeneities are specified as easily parameterizable geometric shapes. For example a flat, circular heterogeneity can be parameterized using only a few parameters, an x,y,z centroid location and a radius. Another approach is the pilot point method [18], where arbitrarily-shaped heterogeneities are constructed from a set of points and then adapted over many iterations until some data-fitting objective is achieved. This method allows for a wider variety of geometries and may better explain the behavior of real-world systems, but requires more free parameters and therefore a longer runtime. The pilot point method is an example of a dynamic parameterization [9], since the parameterization describing the system is variable throughout the course of the inversion and is itself treated as one of the unknowns to be solved for.

In this study a hierarchical approach is used to combine these two approaches, solving the inverse problem in its most simplified form before feeding the best solutions forward and using them as initial guesses for progressively more challenging forms of the inverse problem. The forward model [67, 68] uses the concept of poroelasticity [57, 7, 64] to fit geomechanical signals measured in the shallow subsurface during injection of a confined permeable rock formation of unknown geometry. The approach has potential applications in a wide range of industries such as oil and natural gas development, mining and geothermal energy.

5.3 Methods

This study involved the design of a conceptual model and pumping schedule for the site, implementation of a numerical model to represent the hydromechanical behavior of the field site,

and software development of the inversion algorithm.

5.3.1 Site Description and Pumping Experiment

We conducted this experiment at an active oil field near Avant, Oklahoma. The local geology include a 30-meter thick layer of sandstone called the Bartlesville formation, at a depth of approximately 500 m. The sandstone formation has shale confining layers above and below, and is underlain by permeable lenses thought to be formed by a braided stream system. These lenses have wells installed for the purpose of oil exploration and for injection of produced water.

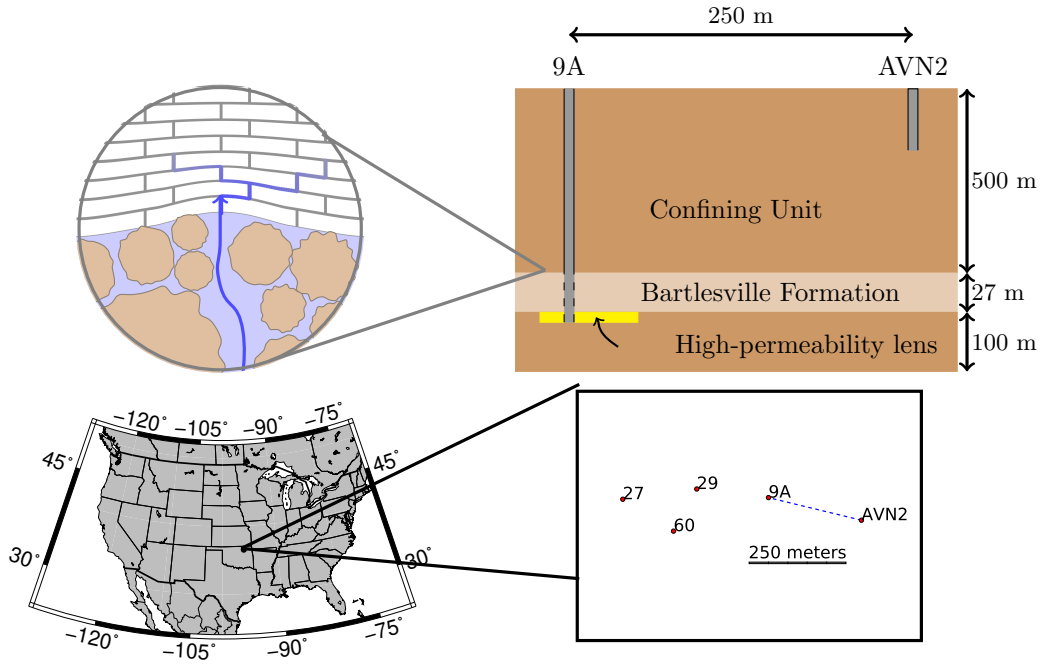


Figure 5.1: Active injection wells in the Avant oil field are shown in map view. Geological conceptual model is shown in cross section.

The field site had one injection well (9A) which was completed within the permeable lens, one observation well (AVN2) completed to a depth of 30 m and instrumented with a Gladwin strainmeter, and three additional observation wells (27, 29, 60) completed in the permeable lens and used to monitor fluid pressure. The Gladwin strainmeter recorded the East-West, North-South and vertical strain components. The injection well was performed at well 9A with a variable flow rate for approximately six days before the injection was stopped for a seven day recovery.

5.3.2 Forward Model

The numerical forward model uses a finite element software called Geocentric [67, 68] to simulate the poroelastic response to the injection schedule. The model space is 16 km by 16 km, with the inner region (4km by 4km) meshed finely while the outer region is coarser [Figure 5.2]. The finite element mesh consists of free quadrilateral elements defined over the inner region and rectangular elements over the outer region [Figure 5.2], which were swept vertically forming 3D hexahedral mesh elements. Wells are meshed explicitly using a vertical cylinder with material properties approximating steel ($E = 200$ GPa, $\kappa = 10 \times 10^{-99} \text{ m}^2$), and an interior cylinder with a lower bulk modulus and permeability ($K = 3$ GPa, $\kappa = 1 \times 10^{-8} \text{ m}^2$). The well is screened (permeability $\kappa = 1 \times 10^{-8} \text{ m}^2$) where it intersects with the permeable lens.

Flux equal to the measured volumetric injection rate is specified along the top of the well. Other boundary conditions are defined along the outer side boundaries assuming a fixed fluid pressure and a roller boundary for strain that allows the boundaries to deform parallel to the surface, but not normal to the plane. This explicitly prevents the outer boundaries from expanding or contracting. The lower boundary is specified as a zero-flux boundary for flow, and a roller boundary preventing vertical deformation. The upper boundary is assigned zero total stress and a no-flow condition. The model is divided into four layers [Figure 5.1] which represent the upper (500 m thick) and lower (100 m thick) confining units, Bartlesville formation (27 m thick), and permeable layers (5 m thick).

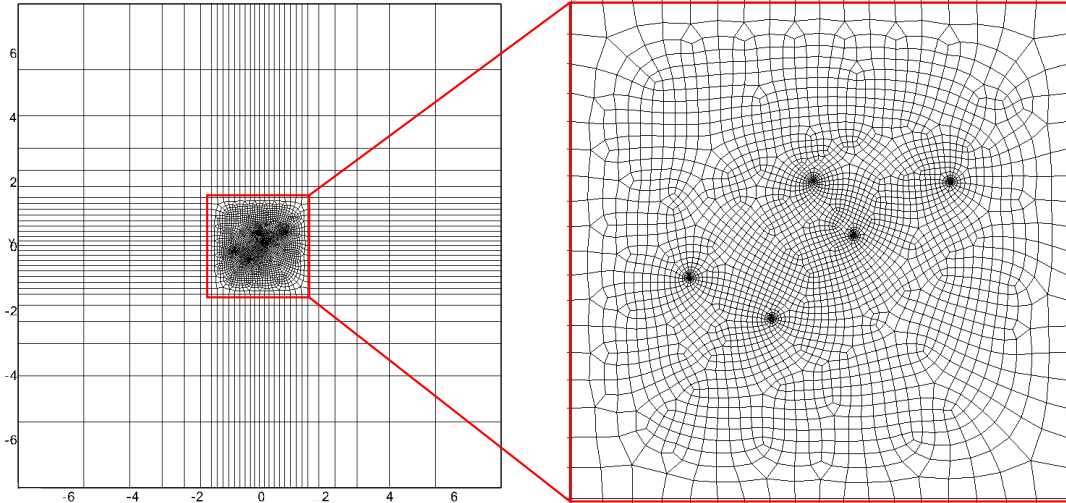


Figure 5.2: Outer, coarse mesh is shown with inset finely meshed region.

An algebraic function [Figure 5.3] was developed which approximates the injection rate. The measured flow rate fluctuated substantially over small time periods, making it impractical to simulate enough timesteps to precisely represent the injection rate. However, this algebraic function very closely matches the cumulative volume of water injected.

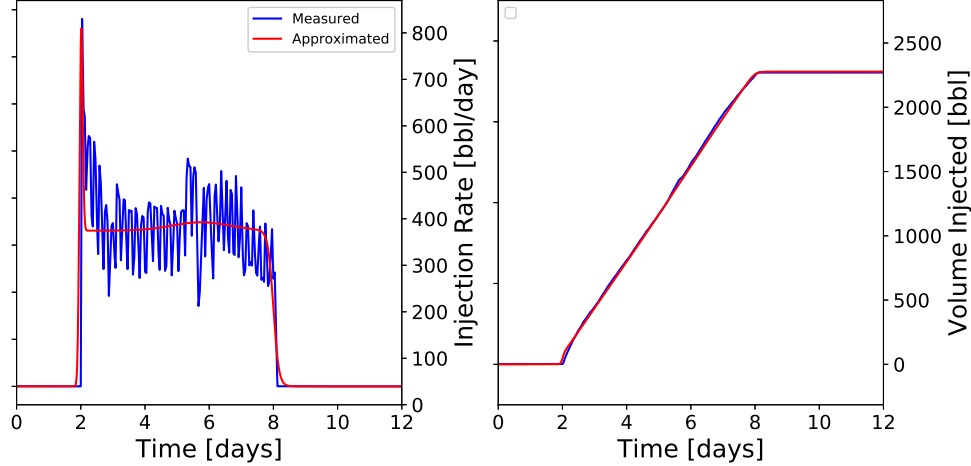


Figure 5.3: The injection rate over time and total injected volume over time are shown for both the measured flow, and the smooth function approximating the measured flow.

5.3.3 Parallelization of Computational Effort

In total, this inversion involved 24,125 simulations requiring 30,756 core hours, and would have taken nearly a year on a typical four-core desktop machine. In order to reduce computation time, this effort was distributed over many compute nodes. A MySQL relational database [Figure 4.6] was used to house the measured field data, and to store the information the inversion algorithms need in order to run. This includes prior information about the field as well as the model assumptions and controlled variables such as the well pumping rate and observation well locations. It also includes information about the geometry of each heterogeneity being considered. This database stores the input and output files necessary for simulation, and makes them available through the web to computational nodes. These nodes were drawn from a combination of the university cluster computer and Amazon Web Services (AWS). As each simulation is completed, the resulting strain signals are returned to the database and compared to the measured data, and used to inform the next set of simulation input files.

the inversion.

Initially the lens is assumed to be a flat, circular inclusion immediately below the Bartlesville formation and therefore the geometry is described using three parameters: the centroid coordinates (x and y), and the radius. A Latin Hypercube (LHC) algorithm is used ($n=13,150$) to sample the parameter space as evenly as possible while minimizing any correlations between parameters. These simulation results are obtained from these sampled parameter sets then fed into a multi-objective genetic algorithm (NSGA-II, [16]). The genetic algorithm identifies the best models and creates random permutations and mutations of them to iteratively improve the population of models until a set of best-fitting models can be identified. The initial LHC step provides the genetic algorithm with a highly diverse initial population that approximates the entire parameter space as closely as possible.

A modified variant of the NSGA-II algorithm was used, running several instances of the standard NSGA-II algorithm in parallel such that each instance attempted to optimize a different combination of parameters. A third of these instances attempted to explore the tradeoff between only two objectives at a time, while another third focuses on three at a time, and the remaining third attempted to fit all six objectives simultaneously. As the actual six-dimensional tradeoff surface may have been too topologically difficult to explore at once, this approach allowed good solutions from the two- and three-objective inversions to feed into the more challenging six-objective inversion. An additional 31,525 simulations using the circular lens assumption are run, and the best-fitting ($n=1,294$) circular lens geometries and material properties are identified. These circular lenses are re-parameterized to the elliptical parameterization, and a second Latin Hypercube search ($n=22,213$) of the elliptical parameter space is run. An additional 17,295 simulations are then run using the NSGA-II algorithm.

We then applied a novel, dynamically parameterized variation of NSGA-II which allowed lens shape to vary on a pixel-by-pixel basis rather than assuming any particular geometric shape [Figure 5.5]. At each iteration, a best-fitting lens is selected and either a random “chunk”, or set of 2-7 contiguous mesh elements, along the outer edge of the lens is added or removed, allowing the lens to deviate from its original circular or elliptical shape. If this deviation in shape proves to fit the field data better, it will be more likely to be selected in future iterations and be further modified in shape. After enough iterations, a set of irregularly shaped lenses emerges which each explain the data equally well. This ensemble can then be used to examine trends and commonalities between

best-fitting models. This approach would normally be computationally intractable since the number of effective free parameters is proportional to the mesh density of the finite element model. However by using the results of the previous circular and elliptical inversions as starting models, this inversion can be initialized very close to the best viable solution.

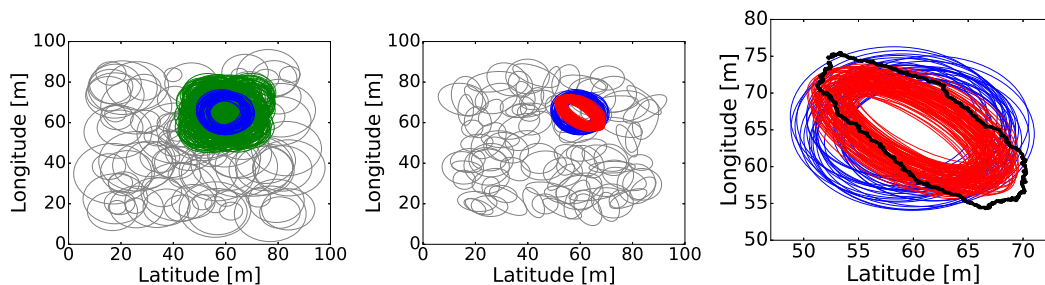


Figure 5.5: Three stage heterogeneity inversion process uses a circular assumption (A), elliptical assumption (B), followed by an irregularly shaped heterogeneity (C). Gray circles represent the Latin Hypercube search of the entire space, while green circles represent the genetic algorithm result, and blue circles represent the best-fitting models possible under the circular assumption. Gray ellipses represent the second-stage Latin Hypercube search of the elliptical parameter space, supplemented by the previous best-fitting circles from the first stage. Red ellipses represent the best-fitting ellipses found using NSGA-II. Black line represents the jagged, irregularly shaped heterogeneity arrived at by NSGA-II in the third stage.

5.4 Results and Discussion

We were able to find a set of models which fit the data well [Figure 5.6], with the elliptical lenses showing a marginal improvement over circular lenses, and irregular lenses showing a much more significant improvement. In particular the irregular lens performs much better in terms of optimizing tradeoffs between pressure and strain data [Figure 5.7]. This might indicate the circular and elliptical lenses were too high in volume while remaining large enough to encompass the pressure observation wells. By contrast the irregular lenses can extend to the necessary observation points while being smaller overall.

Of the approximately 1,886 models that fit the data within a selected error margin [Figure 5.6], two examples were chosen for Figure 5.8. This figure shows a plot of the irregular lens geometry, data fits and material properties. A weighted average of the best-fitting 1,886 models was also plotted, where each model was weighted according to how well it fit the data [Figure 5.9]. In most lenses a small, poorly connected heterogeneity was observed extending northward. This may indicate

the presence of a second distinct compartment hydraulically separated from the lens, which would fit the fluvial geologic history of this rock layer.

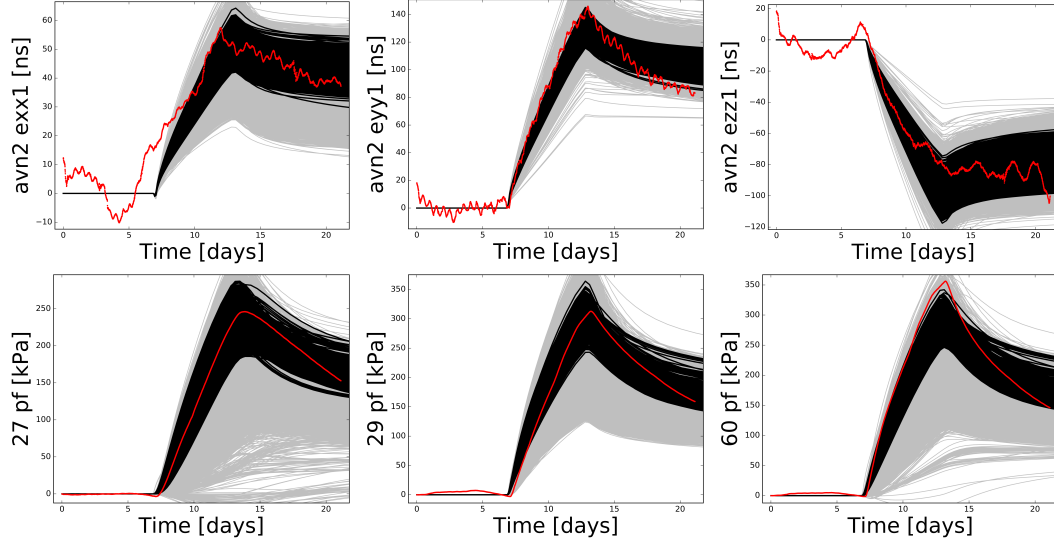


Figure 5.6: Data fits are shown for three components of strain at the the AVN2 observation well, and for pressure data at wells 27, 29 and 60. Simulation results are shown in gray while best-fit simulations are highlighted in black. Measured field data shown as red dots.

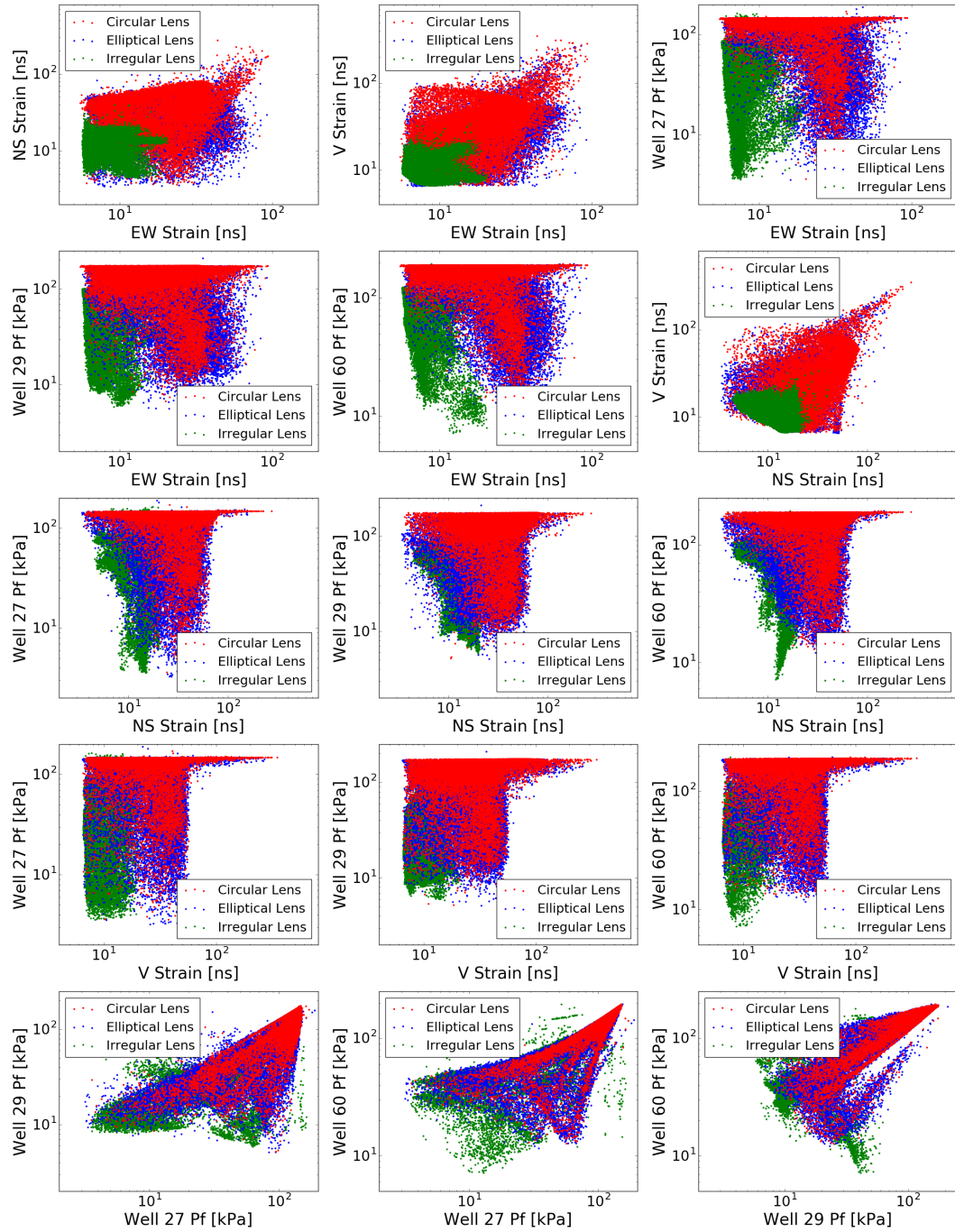


Figure 5.7: Data-fitting tradeoffs are presented for each pair of objectives.

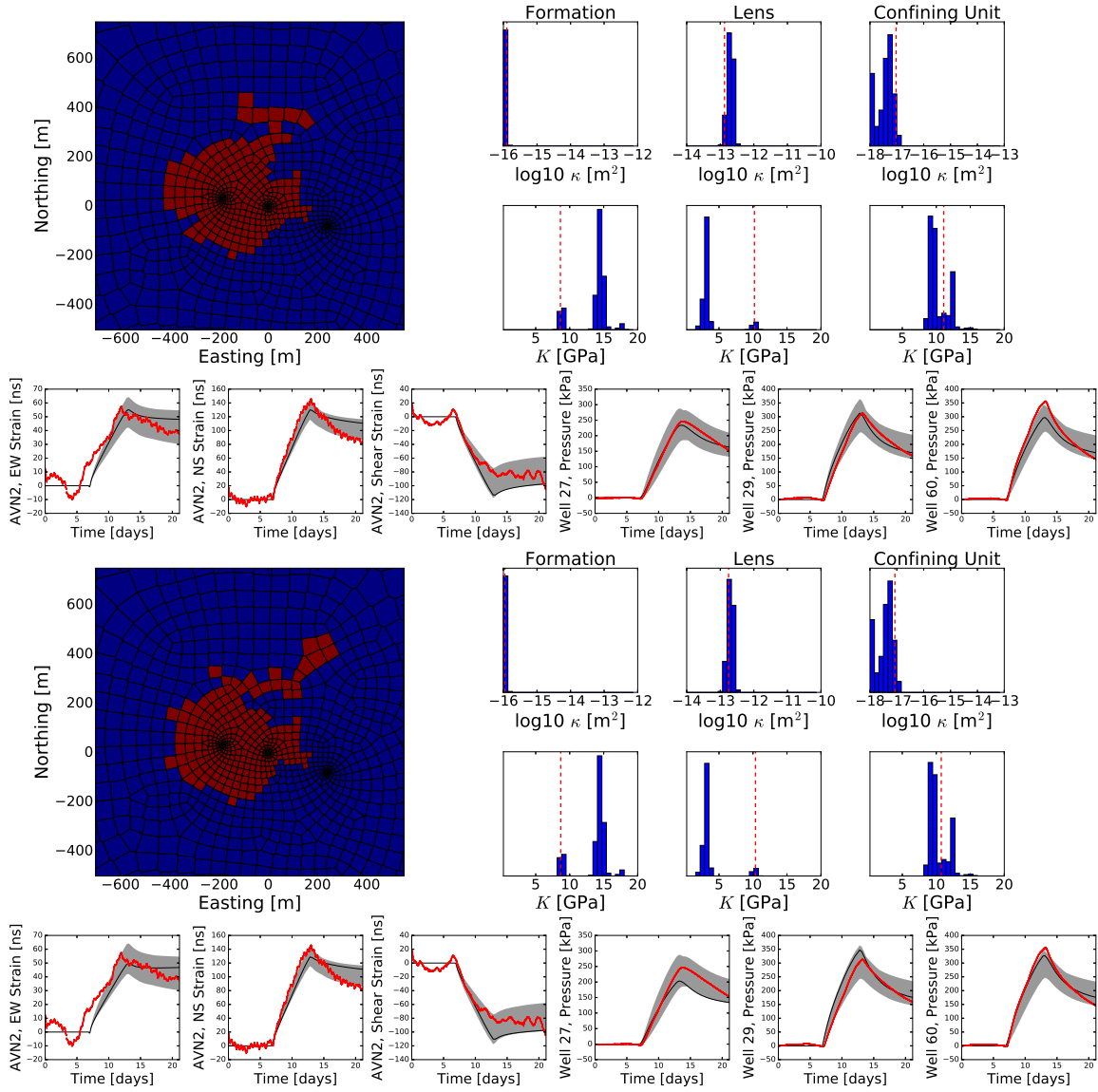


Figure 5.8: Results from two simulations (a) and (b) using irregular lens geometries for the inversion of the October 2017 injection test at well 9A. Each group shows the lens geometry in map view in the left corner. Parameters are shown in upper right corner, where red line represents the parameter value used in the particular simulation, and blue bars are the distribution of 1,886 best-fit models. Lower row shows data fit as black line, with gray lines representing 1,886 best-fit models.

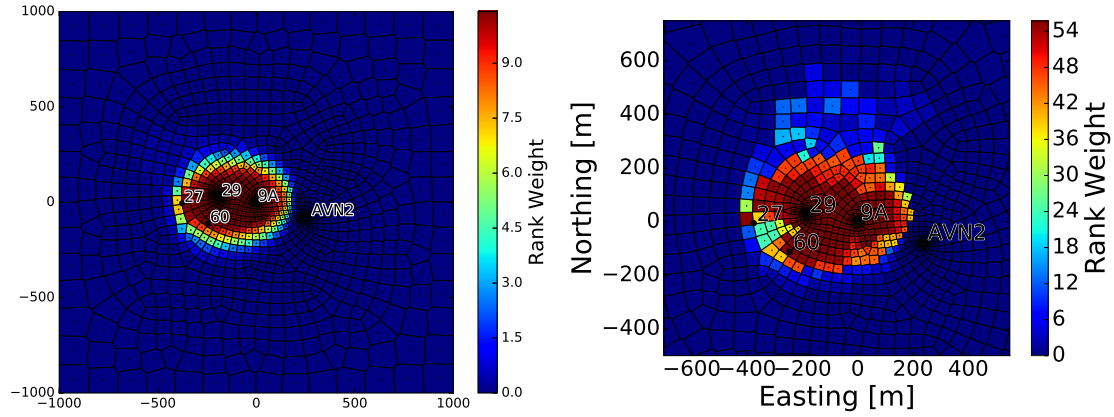


Figure 5.9: Weighted average lens geometry is shown in map view for both the object based (circular and elliptical) lenses (left) and the irregular shaped lenses (right). The 1,886 best-fit models are selected and given a rank according to their weighted data misfit.

One important application of this inversion approach is in risk assessment and decision support. By using field data to calibrate a hydromechanical conceptual model, a large range of models can be constructed which differ from one another in any number of ways, but closely agree in that they explain all of the available field data within some selected error margin. By then simulating the behavior of these hydromechanical models under different pumping scenarios, the range of possible strain responses can be predicted given a proposed injection schedule, and these predictions can be used to inform decision-making.

In this section the 1,886 best-fit models from the injection test were re-run using the pumping schedule for a second pumping test of the same well and reservoir system. This set of simulations were generally able to predict the strain and pressure responses quite well [Figures 5.10,5.11], but failed to predict the tilt responses [Figure 5.12] perhaps due to the short duration of the test and high level of sensor drift observed.

Rather than simply isolating individual sensor responses, these models can also be used to predict the full pressure and strain fields. Predictions of the strain field can be useful when deciding where to position additional sensors by providing an approximation of the likely information value of hypothetical well locations. While the selection of best-fit models will likely closely agree with one another in areas near existing strainmeter locations (because that is the criteria used to select them), they may disagree with one another substantially in locations where data is not yet available. By placing additional sensors at such locations, the set of best-fit models can be narrowed down

and used to better refine our knowledge of the subsurface. For example in Figure 5.13 the East-West, North-South, vertical and areal strain components agree closely with one another in the area around the AVN2 sensor installation. However these models show strong disagreement in an area approximately 200-300 meters northwest of AVN2, with a difference in magnitude on the order of 50-70 nanostrains. The shear strain component indicates different areas of maximum sensitivity, but has a much lower magnitude that may not be resolvable given the degree of sensor drift and conceptual model error.

Predictions of the pressure field can be useful when deciding an ideal injection schedule, selecting flow rates and durations to dispose of as much produced water as possible while minimizing the risk of hydraulic fracturing. In Figure 5.14, the maximum pressure predicted by any of the best-fitting models can give managers a quantification of this risk. The maximum difference in pressures can indicate locations in the field where pressure transducer data would be most valuable.

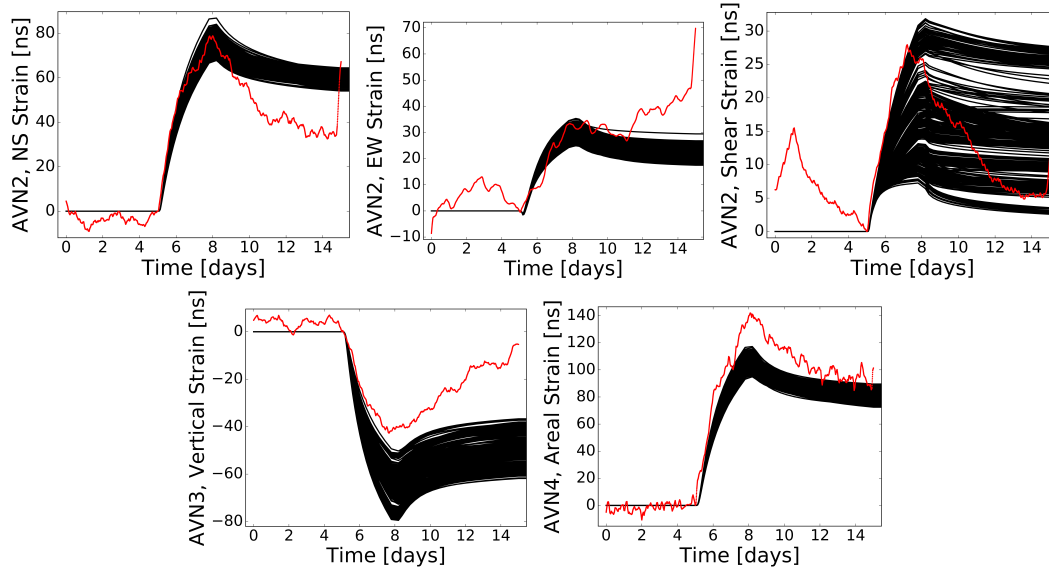


Figure 5.10: Strain components at AVN (North-South, East-West, shear) as functions of time for the November 9A injection test. Vertical strain at well AVN3, areal strain at AVN4. Simulation results are shown in gray and best-fit simulations are in black. Measured field data shown as red dots.

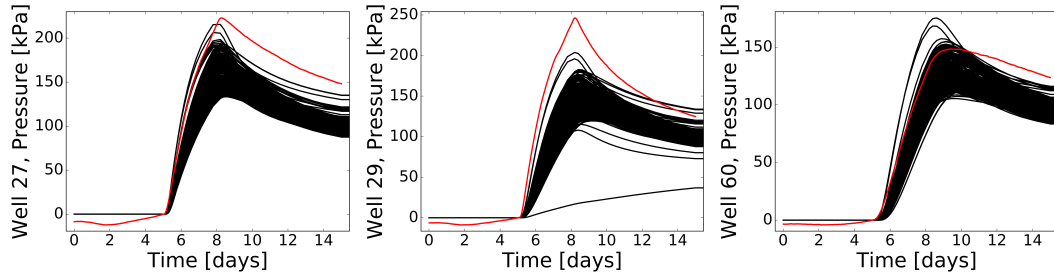


Figure 5.11: Pressure readings at wells 27, 29 and 60 as functions of time for the November 9A injection test. Simulation results are shown in gray and best-fit simulations are in black. Measured field data shown as red dots.

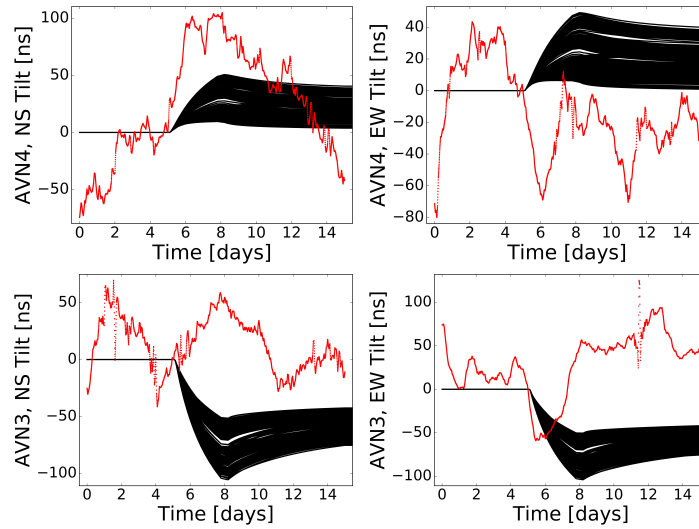


Figure 5.12: Tilt components at AVN3 and AVN4 as functions of time for the November 9A injection test. Simulation results are shown in gray and best-fit simulations are in black. Measured field data shown as red dots.

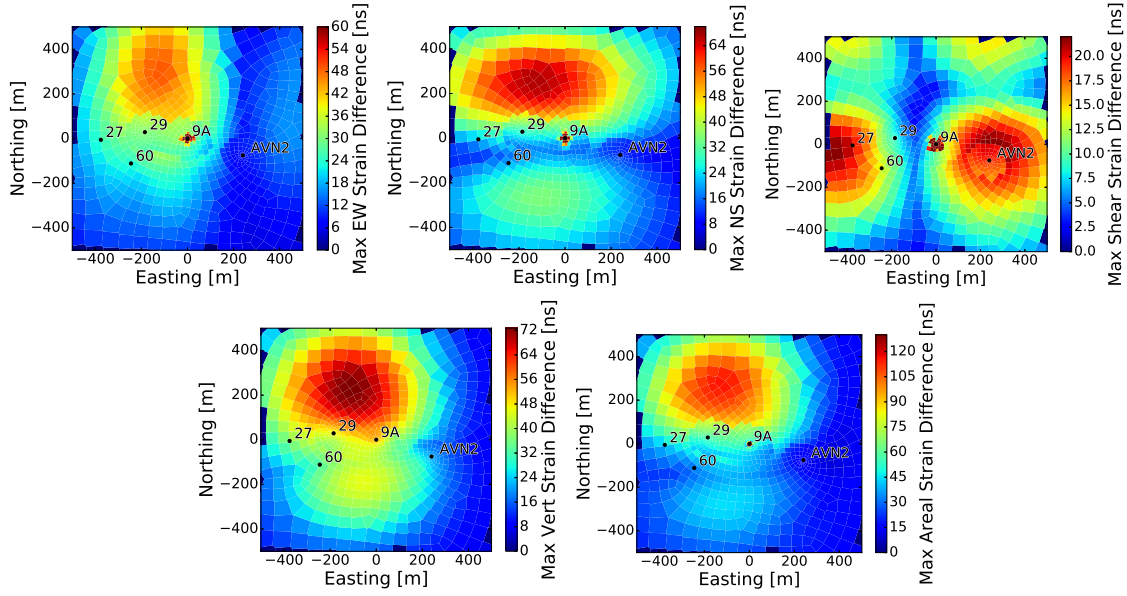


Figure 5.13: Differences in predicted strains are shown as a function of position. For each mesh element, the greatest difference is computed between any two of the selected best-fit models (corresponding to 5.10, 5.11). East-West (a), North-South (b), shear (c), vertical (d) and areal (e) strain components are shown. Results are displayed in map view with the horizontal slice defined at a depth of 30 meters, the approximate depth of the currently installed strainmeters.

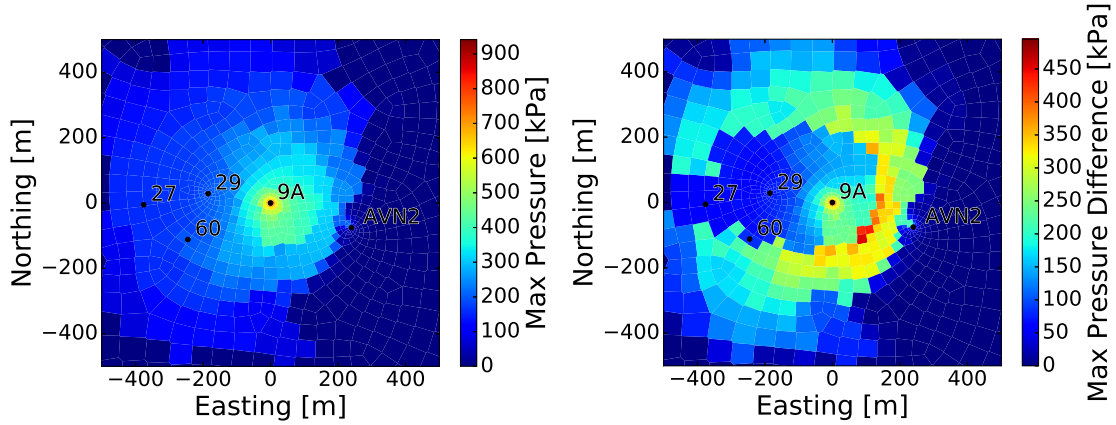


Figure 5.14: Maximum fluid pressure (a) and maximum difference in fluid pressure (b) are shown as a function of position in map view. For each mesh element the maximum pressure represents the highest pressure indicated by any of the selected best-fit models, while the maximum difference represents the greatest pressure difference between any two best-fit models. Horizontal slice is located in the vertical center of the permeable lens.

A third, much longer injection experiment was then run and simulations of this pumping experiment were evaluated in order to predict the strain, pressure and tilt behavior for the longer

pumping schedule. These simulations generally fit strains and pressures [Figures 5.15,5.16] well for the first few days, then systematically overestimated both strains and pressure at later times. This may be due to the longer injection time, as in this case the diffusive pressure response is able to move further out and interact with geologic structures further from the injection well. For the shorter pumping tests, these structures may not have impacted the system much at all as the pressure wave never reached them. In future work it may be beneficial to conduct repeated injection tests at various durations, in order to develop models that explain both the long-injection and short-injection field observations. This would allow one to establish the general trends of the larger geologic setting while also resolving fine details in the immediate vicinity of pumping wells.

As with the second test, tilts were predicted poorly [Figure 5.17] and may require further study in order to calibrate to filter out random noise.

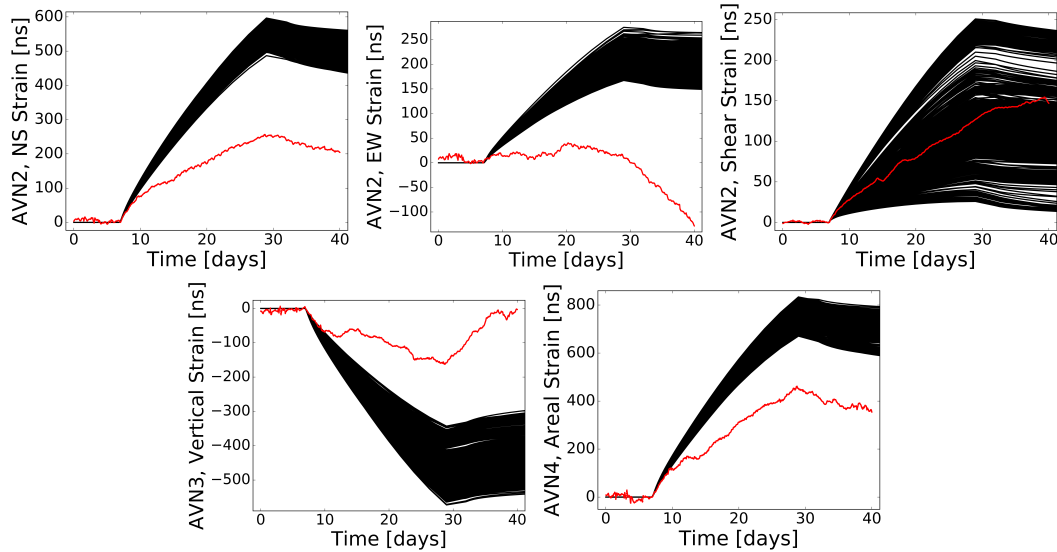


Figure 5.15: Strain components at AVN (North-South, East-West, shear) as functions of time for the July 2018 9A injection test. Vertical strain at well AVN3, areal strain at AVN4. Simulation results are shown in gray and best-fit simulations are in black. Measured field data shown as red dots.

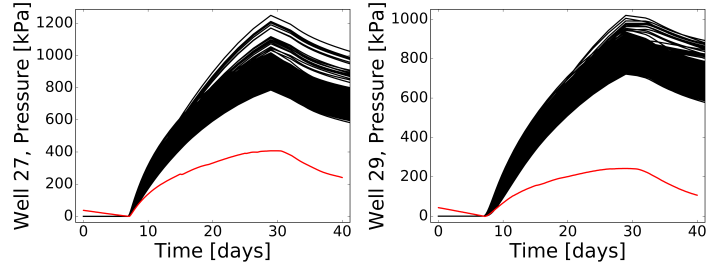


Figure 5.16: Pressure readings at wells 27, 29 and 60 as functions of time for the July 2018 9A injection test. Simulation results are shown in gray and best-fit simulations are in black. Measured field data shown as red dots.

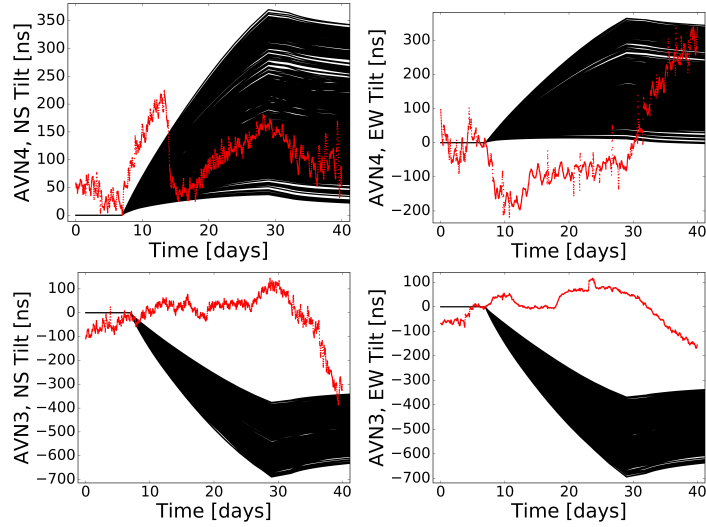


Figure 5.17: Tilt components at AVN3 and AVN4 as functions of time for the July 2018 9A injection test. Simulation results are shown in gray and best-fit simulations are in black. Measured field data shown as red dots.

5.5 Conclusion

In-situ pressure measurements within a confined permeable formation (500 m depth) were combined with remote strainmeter data measured in shallow wells (30m depth) to infer the geologic structure and material properties of the subsurface. The influence of simplifying assumptions was examined using three stages, one where the confined formation was assumed to be circular, one where it was elliptical, and another where it was allowed to take any irregular shape necessary to optimally fit the data.

While more computationally intensive, the irregular-lens assumption was found to yield the

best data fits. However, the circular and elliptical assumptions allowed for a more thorough and efficient initial exploration of the parameter space. By combining the three methods the parameter space was explored thoroughly and the general size and location of the permeable formation determined, allowing a more detailed subsequent calibration of the lens shape.

Summary

In this dissertation, a novel parallelized inversion and job-scheduling software was developed in Python, using an object-relational approach to store the large, highly structured datasets involved. This software was tested using benchmark problems as well as applications in the field. This software has a variety of potential applications in hydrologic model calibration, as it allows coupling of multiple forward models using very different programming languages and modelling approaches.

A geomechanical model was developed to represent an oil field in Avant, Oklahoma. This model uses two-way coupling between groundwater flow and mechanical rock deformation to represent the strain response of a pressurized subsurface reservoir. A series of pumping tests were then conducted at this oil field in order to cause a mechanical stress which produced a measurable strain signal in the shallow subsurface. It was found that this near-surface strain data is sensitive to material properties of the deeper subsurface as well as small, distant details in geologic structure. This strain data has similar information content to in-situ pressure measurements, and therefore would be an ideal method to less-invasively study and monitor an environmental remediation project, hydraulic fracking site, enhanced oil recovery operation or geothermal energy field.

While nonuniqueness was found to be common in geomechanical inverse problems, this software allows one to construct an ensemble of subsurface models that explain the near-surface strain data. This ensemble can be used to predict the range of possible future system behaviors. Incorporation of additional datasets and multiphysics solvers would likely reduce the uncertainty in this range of behaviors, and produce a more accurate and reliable reservoir calibration tool.

Bibliography

- [1] D.H. Ackley. *A Connectionist Machine for Genetic Hillclimbing*. The Kluwer International Series in Engineering and Computer Science. Kluwer Academic Publishers, Boston, 1987.
- [2] C Bradford Barber, David P Dobkin, David P Dobkin, and Hannu Huhdanpaa. The quickhull algorithm for convex hulls. *ACM Transactions on Mathematical Software (TOMS)*, 22(4):469–483, 1996.
- [3] Andrew J Barbour and Frank K Wyatt. Modeling strain and pore pressure associated with fluid extraction: The pathfinder ranch experiment. *Journal of Geophysical Research: SolidEarth*, 119(6):5254–5273, 2014.
- [4] Domenico Bau, Massimiliano Ferronato, Giuseppe Gambolati, Pietro Teatini, and Ayman Alzraiee. Ensemble smoothing of land subsidence measurements for reservoir geomechanical characterization. *International Journal for Numerical and Analytical Methods in Geomechanics*, 39:207–228, 2015.
- [5] Brian Berkowitz. Characterizing flow and transport in fractured geological media: A review. *Advances in water resources*, 25(8):861–884, 2002.
- [6] To Thanh Binh and Ulrich Korn. Mobes: A multiobjective evolution strategy for constrained optimization problems. In *The Third International Conference on Genetic Algorithms (Mendel97)*, volume 25, pages 176–182, 1997.
- [7] Maurice A Biot. General theory of three-dimensional consolidation. *Journal of Applied Physics*, 12(2):155–164, 1941.
- [8] Maurice A Biot. Mechanics of deformation and acoustic propagation in porous media. *Journal of Applied Physics*, 33(4):1482–1498, 1962.
- [9] Thomas Bodin and Malcolm Sambridge. Seismic tomography with the reversible jump algorithm. *Geophysics Journal International*, 178:1411–1436, April 2009.
- [10] Frédéric Cappa and Jonny Rutqvist. Modeling of coupled deformation and permeability evolution during fault reactivation induced by deep underground injection of co 2. *International Journal of Greenhouse Gas Control*, 5(2):336–346, 2011.
- [11] Pascal Castellazzi, Richard Martel, Devin L Galloway, Laurent Longuevergne, and Alfonso Rivera. Assessing groundwater depletion and dynamics using grace and insar: Potential and limitations. *Groundwater*, 54:768–780, November-December 2016.
- [12] Haibin Chang, Yan Chen, and Dongxiao Zhang. Data assimilation of coupled fluid flow and geomechanics using the ensemble kalman filter. *Society of Petroleum Engineers Journal*, 15, June 2010.

- [13] Laura Chiaramonte, Mark D Zoback, Julio Friedmann, and Vicki Stamp. Seal integrity and feasibility of co2 sequestration in the teapot dome eor pilot: geomechanical site characterization. *Environmental Geology*, 54(8):667–1675, 2008.
- [14] Rattanasuda Cholathat, Linlin Ge, and Xiaojing Li. Monitoring geologic carbon sequestration with radar remote sensing. In W. Short and I. Cairns, editors, *Australian Space Science Conference Series*, pages 187–198, 2010.
- [15] R.K. Pachauri Core Writing Team and L.A. Meyer (eds.). Ipcc, 2014: Climate change 2014: Synthesis report. Technical report, 2014.
- [16] Kalyanmoy Deb. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197, 2002.
- [17] Clayton V Deutsch and Libing Wang. Hierarchical object-based stochastic modeling of fluvial reservoirs. *Mathematical Geology*, 28(7):857–880, 1996.
- [18] John Doherty. Ground water model calibration using pilot points and regularization. *Groundwater*, 41(2):170–177, 2003.
- [19] Jing Du and Jon E Olson. A poroelastic reservoir model for predicting subsidence and mapping subsurface pressure fronts. *Journal of Petroleum Science and Engineering*, 30:181–197, 2001.
- [20] Pengcheng Fu, Scott M Johnson, and Charles R Carrigan. An explicitly coupled hydrogeomechanical model for simulating hydraulic fracturing in arbitrary discrete fracture networks. *International Journal for Numerical and Analytical Methods in Geomechanics*, 37(14):2278–2300, 2013.
- [21] Pengcheng Fu, Scott M Johnson, Yue Hao, and Charles R Carrigan. Fully coupled geomechanics and discrete flow network modeling of hydraulic fracturing for geothermal applications. In *The 36th Stanford geothermal workshop*, volume 35, 2011.
- [22] Devin L Galloway and Thomas J Burbey. Regional land subsidence accompanying groundwater extraction. *Hydrogeology Journal*, 19(8):1459–1486, 2011.
- [23] Andrew Gelman and Donald B. Rubin. Inference from iterative simulation using multiple sequences. *Statistical Science*, 7(4):457–511, 1992.
- [24] John Geweke. Evaluating the accuracy of sampling-based approaches to the calculation of posterior moments. *Bayesian Statistics*, 7(4):457–511, 1992.
- [25] Peter L Green. Bayesian system identification of a nonlinear dynamical system using a novel variant of simulated annealing. *Mechanical Systems and Signal Processing*, 52–53:133–146, 2015.
- [26] Heikki Haario, Eero Saksman, and Johanna Tamminen. Componentwise adaptation for high dimensional mcmc. *Computational Statistics*, 20(2):265–273, 2005.
- [27] Alexander C. Hanna, Stephen M.J. Moysey, and Lawrence C. Murdoch. Numerical proof-of-feasibility of using geomechanical measurements to estimate poroelastic parameters. *Geomechanics for Energy and the Environment*, 2019.
- [28] Alexander C. Hanna, Stephen M.J. Moysey, and Lawrence C. Murdoch. Using borehole strain tensor measurements to locate and characterize subsurface heterogeneities in the bartlesville formation in avant field, oklahoma. *Geophysical Research Letters*, 2019.
- [29] W. K. Hastings. Monte carlo sampling methods using markov chains and their applications. *Biometrika*, 57(1):97–109, 1970.

- [30] Philip Heidelberger and Peter D Welch. Simulation run length control in the presence of an initial transient. *Operations Research*, 31(6):1109–1144, 1983.
- [31] J C Helton and F J Davis. Latin hypercube sampling and the propagation of uncertainty in analyses of complex systems. *Reliability Engineering and System Safety volume =*, July 2003.
- [32] Marc A Hesse and Georg Stadler. Joint inversion in coupled quasi-static poroelasticity. *Journal of Geophysics Research: Solid Earth*, 119:1425–1445, 2014.
- [33] Bruce Hill, Susan Hovorka, and Steve Melzer. Geologic carbon storage through enhanced oil recovery. *Energy Procedia*, 37:6808–6830, 2013.
- [34] David B Hisz, Lawrence C Murdoch, and Leonid N Germanovich. A portable borehole extensometer and tiltmeter for characterizing aquifers. *Water Resources Research*, 49:7900–7910, 2013.
- [35] Marco A. Iglesias and Dennis McLaughlin. Data inversion in coupled subsurface flow and geomechanics models. *Inverse Problems*, 28(11), 2012.
- [36] C.E. Jacob. The flow of water in an elastic artesian aquifer. *Transactions, American Geophysical Union*, 21:574–586, 1940.
- [37] Thomas Jacob, Jean Chery, Frederic Boudin, and Roger Bayer. Monitoring deformation from hydrologic processes in a karst aquifer using long-baseline tiltmeters. *Water Resources Research*, 46, 2010.
- [38] M Yusup Nur Khakim, Takeshi Tsuji, and Toshifumi Matsuoka. Geomechanical modeling for insar-derived surface deformation at steam-injection oil sand fields. *Journal of Petroleum Science and Engineering*, 96-97:152–161, October 2012.
- [39] S Kirkpatrick, C D Gelatt, and M P Vecchi. Optimization by simulated annealing. *Science*, 220:671–680, May 1983.
- [40] R Timothy Marler and Jasbir S Arora. The weighted sum method for multi-objective optimization: new insights. *Structural and multidisciplinary optimization*, 41(6):853–862, 2010.
- [41] Carl D McElwee. Theis parameter evaluation from pumping tests by sensitivity analysis. *Groundwater*, 18:56–60, 1980.
- [42] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, and A. H. Teller. Equations of state calculations by fast computing machines. *The Journal of Chemical Physics*, 21(6):1087–1092, 1953.
- [43] N. Metropolis and S. Ulam. The monte carlo method. *Journal of the American Statistical Association*, 44:335–341, 1949.
- [44] Inga Moeck, Grzegorz Kwiatek, and Günter Zimmermann. Slip tendency analysis, fault reactivation potential and induced seismicity in a deep geothermal reservoir. *Journal of Structural Geology*, 31(10):1174–1182, 2009.
- [45] Jens Christian Refsgaard, Jeroen P Van der Sluijs, James Brown, and Peter Van der Keur. A framework for dealing with uncertainty due to model structure error. *Advances in water resources*, 29(11):1586–1597, 2006.
- [46] Vicente J Romero, John V Burkardt, Max D Gunzburger, and Janet S Peterson. Initial application and evaluation of a promising new sampling method for response surface generation: centroidal voronoi tessellation. In *44th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, 2008.

- [47] Jonny Rutqvist, Antonio P Rinaldi, Frédéric Cappa, and George J Moridis. Modeling of fault reactivation and induced seismicity during hydraulic fracturing of shale-gas reservoirs. *Journal of Petroleum Science and Engineering*, 107:31–44, 2013.
- [48] Jong-hyun Ryu, Sujin Kim, and Hong Wan. Pareto front approximation with adaptive weighted sum method in multiobjective simulation optimization. In *Winter Simulation Conference*, pages 623–633. Winter Simulation Conference, 2009.
- [49] Jonathan Schuite, Laurent Longuevergne, Olivier Bour, Frederick Boudin, Stephane Durand, and Nicolas Lavenant. Inferring field-scale properties of a fractured aquifer from ground surface deformation during a well test. *Geophysical Research Letters*, 42(24), 2015.
- [50] Jonathan Schuite, Laurent Longuevergne, Olivier Bour, Nicolas Guiheneuf, Matthew W Becker, Matthew Cole, Thomas J Burbey, Nicolas Lavenant, and Frederick Boudin. Combining periodic hydraulic tests and surface tilt measurements to explore in situ fracture hydromechanics. *Journal of Geophysical Research: Solid Earth*, 2017.
- [51] Mahmood Shakiba, Shahab Ayatollahi, and Masoud Riazi. Investigation of oil recovery and co₂ storage during secondary and tertiary injection of carbonated water in an iranian carbonate oil reservoir. *Journal of Petroleum Science and Engineering*, 137:134–143, 2016.
- [52] Michael Stein. Large sample properties of simulations using latin hypercube sampling. *Technometrics*, 29(2):143–151, 1987.
- [53] B Suman and P Kumar. A survey of simulated annealing as a tool for single and multiobjective optimization. *Journal of the Operational Research Society*, 57:1143–1160, 2006.
- [54] Riccardo Taormina and Kwok wing Chau. Data-driven input variable selection for rainfallrunoff modeling using binary-coded particle swarm optimization and extreme learning machines. *Journal of Hydrology*, 529:1617–1632, 2015.
- [55] Riccardo Taormina and Kwok wing Chau. Neural network river forecasting with multi-objective fully informed particle swarm optimization. *Journal of Hydroinformatics*, 17:99–113, 2015.
- [56] Karl Terzaghi. *Theoretical soil mechanics*. Chapman And Hali, Limited John Wiler And Sons, Inc, 1944.
- [57] Karl Terzaghi et al. *Erdbaumechnik auf bodenphysikalischer grundlage*. 1925.
- [58] Charles V Theis. The relation between the lowering of the piezometric surface and the rate and duration of discharge of a well using ground-water storage. *Transactions, American Geophysical Union*, 16:519–524, 1935.
- [59] Donald Wyman Vasco, Alessandro Ferretti, and Fabrizio Novali. Reservoir monitoring and characterization using satellite geodetic data: Interferometric synthetic aperture radar observations from the krechba field, algeria. *Geophysics*, 73(6), 2001.
- [60] Donald Wyman Vasco, Alessandro Ferretti, and Fabrizio Novali. Reservoir monitoring and characterization using satellite geodetic data: Interferometric synthetic aperture radar observations from the krechba field, algeria. *Geophysics*, 73(6), 2008.
- [61] Jasper A. Vrugt, Hoshin V. Gupta, Luis A. Bastidas, Willem Bouten, and Soroosh Sorooshian. Effective and efficient algorithm for multiobjective optimization of hydrologic models. *Water Resources Research*, 39(8), 2003.

- [62] Jasper A. Vrugt and Cajo J. F. ter Braak. Dream(d): An adaptive markov chain monte carlo simulation algorithm to solve discrete, noncontinuous, and combinatorial posterior parameter estimation problems. *Hydrology and Earth System Sciences*, 15:3701–3713, 2011.
- [63] W. C. Walton. *Groundwater Resource Evaluation*. McGraw Hill Co, December 1970.
- [64] Herbert F. Wang. *Theory of linear poroelasticity with applications to geomechanics and hydrogeology*. Princeton Series in Geophysics. Princeton University Press, December 2000.
- [65] Zhijing Wang, Michael E. Cates, and Robert T. Langan. Seismic monitoring of a co₂ flood in a carbonate reservoir; a rock physics study. *Geophysics*, 63(5):1604–1617, October 1998.
- [66] D. J. White. Geophysical monitoring of the weyburn co₂ flood: Results during 10 years of injection. *Energy Procedia*, 4:3628–3635, 2011.
- [67] Josh A. White and R. I. Borja. Stabilized low-order finite elements for coupled solid-deformation/fluid-diffusion and their application to fault-zone transients. *Computational Methods in Applied Mechanical Engineering*, 197:4353–4366, 2008.
- [68] Josh A. White and R. I. Borja. Block-conditioned newton krylov solvers for fully coupled flow and geomechanics. *Computers and Geosciences*, 15:527–541, 2011.
- [69] William T. Vetterling Brian P. Flannery William H. Press, Saul A. Teukolsky. *Numerical Recipes in C++: The art of scientific computing*. Cambridge University Press, 2nd edition, 2002.
- [70] Wei Yu, Hamid Reza Lashgari, Kan Wu, and Kamy Sepehrnoori. Co₂ injection for enhanced oil recovery in bakken tight oil reservoirs. *Fuel*, 159:354–363, 2015.
- [71] E. Zitzler, M. Laumanns, and L. Thiele. Spea2: Improving the strength pareto evolutionary algorithm. Technical Report 103, Computer Engineering and Networks Laboratory (TIK), ETH Zurich, Zurich, Switzerland, 2001.
- [72] E. Zitzler and L. Thiele. An evolutionary algorithm for multiobjective optimization: The strength Pareto approach. Technical Report 43, Computer Engineering and Communication Networks Lab (TIK), Swiss Federal Institute of Technology (ETH), Zürich, Switzerland, 1998.
- [73] Claudia Zoccarato. *Data Assimilation in Geomechanics: Characterization of Hydrocarbon Reservoirs*. PhD thesis, Universita Degli Studi di Padova, 2016.