ENHANCED LINEARIZED REDUCED-ORDER MODELS FOR SUBSURFACE FLOW SIMULATION

A THESIS
SUBMITTED TO THE DEPARTMENT OF ENERGY
RESOURCES ENGINEERING
OF STANFORD UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
MASTER OF SCIENCE

Jincong He
June 2010

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as partial fulfillment of the degree of Master of Science in Petroleum Engineering.

Louis J. Durlofsky Principal Adviser

Abstract

Many reservoir simulation applications, such as optimization of well settings and history matching, require a large number of runs. Using traditional simulators for such problems is often very expensive computationally. In this thesis we extend and enhance the trajectory piecewise linearization (TPWL) method for accurate and efficient reduced-order modeling for such applications. In this approach, the reservoir equations are linearized around previously simulated training runs and the high-dimension state space is projected into a low-dimension space using proper orthogonal decomposition (POD). With linearization and reduction, simulations that require hours to run can now be completed within a few seconds. TPWL does require overhead computations that correspond to the time required for a few full-order simulations.

In this work, both the accuracy and stability of the TPWL method are considered. A local resolution treatment, in which we preserve high resolution in well blocks and other important flow regions, is proposed to improve the accuracy of the method. Stability analysis shows that, particularly for cases with density differences between phases, the TPWL method can be unstable. The stability of the method can be quantified in terms of the spectral radius of an amplification matrix appearing in the TPWL model equation. Two different stabilizing methods are proposed. The first method seeks to minimize the spectral radius of the amplification matrix by choosing the optimized number of reduced variables. The second method stabilizes cases with density differences using a basis constructed from the same case without density differences. Both methods are tested on two reservoir problems of practical size with significant density differences. Results demonstrate that both methods are able to provide stabilized and accurate solutions for these cases.

The stabilized TPWL method is then implemented as a surrogate model within a generalized pattern search (GPS) optimization procedure. Two production optimization problems are considered. In the first problem, optimization of 36 bottom hole pressure (BHP) variables under linear constraints is performed on a model containing 4800 grid blocks. Comparison with the results from full-order simulations is possible in this case and demonstrates the accuracy and applicability of TPWL for this problem. In the second production optimization example, 54 BHP variables under nonlinear constraints are optimized for a model containing 20,400 grid blocks. The TPWL method is shown to provide a feasible solution with much improved net present value. The equivalent of only around 20 full-order training simulations are needed for this problem, even though the optimization requires more than 4000 function evaluations (which are provided by the TPWL model).

Finally, the potential use of TPWL for history matching is investigated. Preliminary results show that the TPWL approach is able to provide a reasonable approximation to the true solution even when the geological model for the test case is considerably different than that for the training case. This capability is very useful for history matching and suggests that further study of TPWL in this application area is warranted.

Contents

List of Tables

List of Figures

Acknowledgments

First of all, I would like to give my sincere gratitude to my advisor Prof. Louis Durlofsky, who not only provided instruction and supervision as an advisor, but also offered his insightful suggestions, kindly encouragement, and helpful advice to me as a colleague and a friend. I benefited a lot from all the discussions and interactions with him.

Furthermore, Jon Sætrom from Norwegian University of Science and Technology (NTNU) has been deeply involved with this work. Jon was a visiting researcher in the department for the 2008-2009 academic year and we have been collaborating on this topic since then. He developed many interesting ideas including the local resolution and the EDP method. He also brought in a perspective of statistics and an efficient working style. I learned a lot by working with him.

I would also like to thank Marco Cardoso, a former Ph.D. student, and Obi Isebor, a current Ph.D. student in the department, for their significant help on this work.

I am grateful to my colleagues and friends at Stanford, who make my life here more colorful. Special thanks go to my parents and my wife Qiwen Zhao, who support me in the past, present, and future.

Chapter 1

Introduction

Optimization and uncertainty quantification are essential components in many model-based design procedures. The associated computations, which typically require large numbers of simulations, can be extremely time-consuming if highly resolved models are used. This may be the case even if parallel processing is applied, as multiobjective optimization and optimization under uncertainty, in which simulations are performed over a large number of models, can quickly occupy multiple cores.

The use of reduced-order models provides a means for accelerating these simulations. Our interest here is in subsurface flow modeling, which includes simulation of oil reservoirs, aquifers, and carbon sequestration operations. Several reduced-order modeling procedures based on proper orthogonal decomposition have been previously applied within this context; see, e.g., [1–4]. For the nonlinear problems associated with oil reservoir simulation, the speedups achieved by these procedures were, however, relatively modest, at most about a factor of 10.

Trajectory piecewise linearization (TPWL) is a promising approach for model-order reduction that can potentially provide much larger speedups. TPWL represents new solutions of the governing equations in terms of linear expansions around previously simulated (saved) states. This requires that one or more training simulations be performed, during which the states and converged Jacobian matrices at each time step are saved. High degrees of efficiency are achieved by projecting the saved states and matrices into a low-dimensional subspace. This projection can be

accomplished in different ways. In the implementation discussed in this paper, the projection matrix is constructed by proper orthogonal decomposition (POD) of the saved states. The TPWL approach was first introduced by Rewienski and White [5, 6] for the modeling of nonlinear circuits and micromachined devices. Since then it has been applied in a number of disciplines including computational fluid dynamics [7], nonlinear heat-transfer modeling [8] and electromechanical systems [9, 10].

In recent work Cardoso and Durlofsky [11] applied trajectory piecewise linearization (TPWL) procedures for oil reservoir simulation. Systems involving two fluid components and two phases – oil and water – were considered. For test runs involving equal density fluids, TPWL results were shown to be in close agreement with reference (full-order) simulations for control schedules that were within the general range of those used in the training runs. Runtime speedups of a factor of 200-1000 were observed for the examples considered. For cases in which the fluid densities differed significantly, however, instabilities were observed in some runs. This can lead to inaccuracy or, in some instances, to the blowup of the TPWL solution. This is of concern as oil and water phases often display significant density differences in practical cases.

Our goal in this work is to enhance the TPWL procedure presented in [11] to address the limitations noted above. We proceed in two important directions. First, we introduce a localization treatment in which key grid blocks, such as those containing injection or production wells, are represented at full resolution; i.e., the states in these blocks are not projected into the reduced subspace. A missing point estimation procedure [12] is used to determine which grid blocks (in addition to well blocks) to represent explicitly. This localization will be shown to improve the accuracy of the overall TPWL representation and to have relatively little impact on run times (assuming the number of resolved blocks is not too large). The second enhancement is the use of stabilization procedures. Two such approaches are investigated. In one approach, a stabilized basis is determined in a preprocessing step in which various combinations of basis vectors are considered with the goal of minimizing the spectral radius of an amplification matrix that appears in the TPWL representation. In the other approach, the POD basis is determined based on simulations involving equal

density fluids. Accuracy is restored through use of the localization treatment. Both procedures will be shown to improve TPWL stability and performance significantly.

The stability of TPWL models has been studied previously and our development here builds on earlier work. Rather than use the traditional Galerkin projection, Bond and Daniel [13–15] computed a left projection matrix that stabilizes the system. This new left projection matrix is based on Lyapunov stability theory. However, if only stability is considered, the accuracy of the reduced-order model can degrade considerably. Thus, in [14], a stabilizing procedure was proposed where the left projection matrix is constrained by Lyapunov theory to guarantee stability and the difference between the basis matrix and this new left projection matrix is optimized to recover accuracy. Bui-Thanh et al. [16, 17] formulated a goal-oriented, model-constrained optimization problem to determine the optimal basis under Galerkin projection. This approach was also shown to improve stability. This method differs from that of [14] in that it retains the Galerkin projection and optimizes both the basis matrix and the left projection matrix. However, when both accuracy and stability are considered, both of these approaches can become very expensive computationally.

This thesis proceeds as follows. In Chapter 2, the equations for the subsurface flow of oil and water are presented, followed by a brief description of the finite volume approach used for their solution. The POD-based TPWL representation is then described. In Chapter 3 we present the local resolution approach in which key grid blocks are resolved explicitly. The enhanced accuracy provided by this treatment is illustrated with an example. The two approaches for stabilizing the TPWL representation are presented in Chapter 4. The impact of the local resolution and stabilization procedures is demonstrated through two examples, both of which contain $O(10^5)$ grid blocks, in Chapter 5. Next, in Chapter 6, the enhanced TPWL method is combined with a generalized pattern search (GPS) optimization technique and applied to two production optimization problems. Finally, in Chapter 7, we present conclusions and suggestions for future work.

Much of the work presented in this thesis was preformed in collaboration with Jon Sætrom, a Ph.D. student at NTNU (Trondheim, Norway) who spent the 2008-2009 academic year at Stanford as a visiting researcher. Specifically, Jon Sætrom

initiated the work on the local resolution method and developed the EDP method. The work on basis optimization and the use of TPWL for production optimization were performed by the author.

Chapter 2

Problem Formulation

The governing equations for oil-water flow and the basic TPWL formulation were presented in detail in [11]. For the sake of completeness, an abbreviated description is also included here.

2.1 Oil-Water Flow Equations

The equations governing the two-phase flow of oil and water in porous formations are derived by combining expressions for mass conservation with Darcy's law. Using the subscript j to designate component/phase (j=o for oil and w for water), these equations can be written as:

$$\frac{\partial}{\partial t}\left(\phi \rho_j S_j\right) - \nabla \cdot \left[\rho_j \lambda_j \mathbf{k}\left(\nabla p_j - \rho_j g \nabla D\right)\right] + q_j^w = 0, \tag{2.1}$$

where $\mathbf{k}$ is the (diagonal) absolute permeability tensor, $\lambda_j = k_{rj}/\mu_j$ is the phase mobility, with k_{rj} the relative permeability to phase j and μ_j the phase viscosity, p_j is phase pressure, ρ_j is the phase density, g is gravitational acceleration, D is depth, t is time, ϕ is porosity, S_j is saturation and q_j^w is the source/sink term. Eq. 2.1 is written slightly differently here than in [11]. Specifically, the source term q_j^w here differs by a constant factor of ρ_j^0 (where ρ_j^0 is the reference density of phase j) from

that in [11], and the definition of λ_j differs by a factor of ρ_j/ρ_j^0. The general oil-water model is completed by enforcing the saturation constraint ($S_o + S_w = 1$) and by specifying a capillary pressure relationship $p_c(S_w) = p_o - p_w$. Eq. 2.1 is nonlinear since functions of unknowns (e.g., $k_{rj}(S_j)$) multiply unknowns.

We take p_o and S_w to be primary variables (p_w and S_o can be immediately computed once these are known). Eq. 2.1 is solved numerically using a fully-implicit finite volume procedure. Discretized forms for all terms are discussed in [11, 18]. Basically, the flow from block to block is given by the interface transmissibility multiplied by the difference in block pressures, the accumulation term is handled using a first-order implicit (backward Euler) method, and the source term is treated using a well index representation, in which well rates are expressed in terms of an appropriately defined transmissibility times the pressure difference between the well block and the well. Using these representations, and defining $\mathbf{x} = [p_o, S_w]$ as the state vector and $\mathbf{u}$ as the well controls (in this case the wells are controlled by specifying bottom hole pressure or BHP), the discrete system for the fully-implicit formulation can be written as:

$$\mathbf{g}\left(\mathbf{x}^{n+1}, \mathbf{x}^n, \mathbf{u}^{n+1}\right) = \mathbf{A}\left(\mathbf{x}^{n+1}, \mathbf{x}^n\right) + \mathbf{F}\left(\mathbf{x}^{n+1}\right) + \mathbf{Q}\left(\mathbf{x}^{n+1}, \mathbf{u}^{n+1}\right). \tag{2.2}$$

Here $\mathbf{g}$ is the residual we seek to drive to zero, n and $n+1$ designate time level, and $\mathbf{A}$, $\mathbf{F}$ and $\mathbf{Q}$ are the discretized accumulation, flux and source/sink terms, respectively.

Typically, a full-order reservoir simulator is used to solve Eq. 2.2. Newton's method, with the Jacobian matrix given by $\partial\mathbf{g}/\partial\mathbf{x}$, is applied. This is computationally expensive because Eq. 2.2 can be highly nonlinear and practical models may contain on the order of 10^5 or 10^6 grid blocks. We now describe the application of the TPWL approach for the efficient solution of Eq. 2.2.

2.2 Solution of Discretized System using TPWL

The idea of trajectory piecewise linearization is to linearize the residual equation around states saved from previous (training) simulations. Here, at any given time, we linearize around a single point on the training trajectory. Methods that involve

weighted linearizations around multiple points also exist (e.g., [6]), though optimal weights can be difficult to determine and improper weighting can lead to stability problems.

Given the current state $\mathbf{x}^n$, we designate the closest saved state encountered during the training run as $\mathbf{x}^i$. To determine $\mathbf{x}^{n+1}$, we linearize Eq. 2.2 around the state $(\mathbf{x}^{i+1}, \mathbf{x}^i, \mathbf{u}^{i+1})$. This gives

$$\mathbf{g}^{n+1} = \mathbf{g}^{i+1} + \frac{\partial \mathbf{g}^{i+1}}{\partial \mathbf{x}^{i+1}} \left(\mathbf{x}^{n+1} - \mathbf{x}^{i+1}\right) + \frac{\partial \mathbf{g}^{i+1}}{\partial \mathbf{x}^{i}} \left(\mathbf{x}^{n} - \mathbf{x}^{i}\right) + \frac{\partial \mathbf{g}^{i+1}}{\partial \mathbf{u}^{i+1}} \left(\mathbf{u}^{n+1} - \mathbf{u}^{i+1}\right), \quad (2.3)$$

where $\mathbf{g}^{n+1} = \mathbf{g}(\mathbf{x}^{n+1}, \mathbf{x}^n, \mathbf{u}^{n+1})$ and $\mathbf{g}^{i+1} = \mathbf{g}(\mathbf{x}^{i+1}, \mathbf{x}^i, \mathbf{u}^{i+1})$. Here $\mathbf{g}^{i+1} = 0$ because it is the residual of the training simulation. Defining the Jacobian matrix as the derivative of the residual with respect to the state,

$$\mathbf{J}^{i+1} = \frac{\partial \mathbf{g}^{i+1}}{\partial \mathbf{x}^{i+1}}, \quad (2.4)$$

and using the fact that, upon solution $\mathbf{g}^{n+1} = 0$, Eq. 2.3 can be expressed as

$$\mathbf{J}^{i+1} \left(\mathbf{x}^{n+1} - \mathbf{x}^{i+1}\right) = -\left[\frac{\partial \mathbf{A}^{i+1}}{\partial \mathbf{x}^{i}} \left(\mathbf{x}^{n} - \mathbf{x}^{i}\right) + \frac{\partial \mathbf{Q}^{i+1}}{\partial \mathbf{u}^{i+1}} \left(\mathbf{u}^{n+1} - \mathbf{u}^{i+1}\right)\right]. \quad (2.5)$$

Given $\mathbf{x}^n$ and saved information, Eq. 2.5 allows us to linearly compute $\mathbf{x}^{n+1}$; i.e., no iteration is required. This equation is, however, still in a high-dimensional space.

In order to reduce the dimension of Eq. 2.5, we employ linear order reduction. This entails representing the state $\mathbf{x}$ in terms of a reduced state $\mathbf{z}$ and a basis matrix $\mathbf{\Phi}$ using

$$\mathbf{x} \approx \mathbf{\Phi} \mathbf{z}. \quad (2.6)$$

There are many ways to construct $\mathbf{\Phi}$ including optimal Hankel model [19, 20], balanced truncation [21], Krylov subspace methods [22], and proper orthogonal decomposition (POD) [23]. Within the context of TPWL, previous researchers have used Krylov subspace methods for electrical engineering applications [24] and POD for computational fluid dynamics applications [7]. The POD method was applied successfully for our problem in [11] and will be used again here.

In POD, the high-dimension space is represented by a set of orthogonal basis vectors, which are the singular vectors of a "snapshot" matrix $\mathbf{X}$. Each column of $\mathbf{X}$ is simply the pressure or saturation state at a particular time step saved from a training simulation. POD is optimal in the sense that it minimizes the mean squared reconstruction error for the snapshots [25]. Therefore, it is reasonable to assume that POD can represent the states of other (test) simulations if these states are somewhat similar to those encountered during training runs. In this work, consistent with [2, 11, 26], POD is used to reduce pressure and saturation separately. As discussed in [11], in the actual implementation of forming the pressure basis matrix, we use oil potential φ_o ($\varphi_o = p_o/\rho_o - gD$) rather than pressure. This was found to improve TPWL stability in [11].

If we denote the number of simulation grid blocks as N_c and the number of reduced pressure and saturation basis vectors as l_p and l_s respectively, then the dimension of the problem can be reduced from $2N_c$ to $l_p + l_s$. This is accomplished by expressing $\mathbf{x}$ in Eq. 2.5 using Eq. 2.6 and by premultiplying both sides of Eq. 2.5 by $\mathbf{\Phi}^T$ (also called Galerkin projection [27]), which gives

$$\mathbf{J}_r^{i+1}\left(\mathbf{z}^{n+1} - \mathbf{z}^{i+1}\right) = -\left[\left(\frac{\partial \mathbf{A}_r^{i+1}}{\partial \mathbf{x}^i}\right)_r \left(\mathbf{z}^n - \mathbf{z}^i\right) + \left(\frac{\partial \mathbf{Q}_r^{i+1}}{\partial \mathbf{u}^{i+1}}\right)_r \left(\mathbf{u}^{n+1} - \mathbf{u}^{i+1}\right)\right], \quad (2.7)$$

where

$$\mathbf{J}_r^{i+1} = \mathbf{\Phi}^T \mathbf{J}^{i+1} \mathbf{\Phi}, \left(\frac{\partial \mathbf{A}^{i+1}}{\partial \mathbf{x}^i}\right)_r = \mathbf{\Phi}^T \left(\frac{\partial \mathbf{A}^{i+1}}{\partial \mathbf{x}^i}\right) \mathbf{\Phi}, \left(\frac{\partial \mathbf{Q}^{i+1}}{\partial \mathbf{u}^{i+1}}\right)_r = \mathbf{\Phi}^T \left(\frac{\partial \mathbf{Q}^{i+1}}{\partial \mathbf{u}^{i+1}}\right). \quad (2.8)$$

Rearranging Eq. 2.7 we have

$$\mathbf{z}^{n+1} = \mathbf{z}^{i+1} - \left(\mathbf{J}_r^{i+1}\right)^{-1} \left[\left(\frac{\partial \mathbf{A}^{i+1}}{\partial \mathbf{x}^i}\right)_r \left(\mathbf{z}^n - \mathbf{z}^i\right) + \left(\frac{\partial \mathbf{Q}^{i+1}}{\partial \mathbf{u}^{i+1}}\right)_r \left(\mathbf{u}^{n+1} - \mathbf{u}^{i+1}\right)\right]. \quad (2.9)$$

Eq. 2.9 can be solved very efficiently for two reasons. First, as a result of linearization, the new reduced state $\mathbf{z}^{n+1}$ can be calculated directly from $\mathbf{z}^n$ without any iteration. Second, all of the terms in Eq. 2.9 are in low-dimension space, which means that the matrix operations are very fast. Specifically, the evaluation of Eq. 2.9 only involves

two matrix-vector products and four vector additions in the reduced space, which can theoretically be done in a fraction of a second for typical dimension of the reduced space (~ 300) and typical number of time steps (~ 300). Therefore, most of the total TPWL computation time is spent on training simulations and projection, which only need to be done once during preprocessing.

This completes our description of the basic TPWL formulation. See [11] for additional details and algorithms and a discussion of some implementation issues. In the following section we will present an application example of the basic TPWL method and discuss some of its limitations, which we will address in the next chapter.

2.3 Application Example: Reservoir Model 1

We now apply the basic TPWL procedure described in Section 2.2 to a realistic reservoir simulation model. The simulation model, shown in Figure 2.1, is part of the so-called Stanford VI geological model developed by [28]. The model represents a fluvial system with high-permeability channels embedded in a low-permeability background region. The portion of the model considered here contains 20,400 grid blocks (with $n_x = 30$, $n_y = 40$, $n_z = 17$, where n_x, n_y and n_z indicate the number of blocks in each coordinate direction). The dimension of the full-order problem is 40,800 (pressure and saturation unknowns in each grid block). In this model the fluid and rock compressibility and the capillary pressure between the two phases are neglected. We specify oil and water viscosities as $\mu_o = 3$ cp, $\mu_w = 0.5$ cp. The fluids are here specified to have equal densities ($\rho_o = \rho_w = 55$ lb/ft^3). The relative permeability relationships are given by

$$k_{ro}(S_w) = k_{ro}^0 \left(\frac{1 - S_w - S_{or}}{1 - S_{wr} - S_{or}} \right)^a, \quad k_{rw}(S_w) = k_{rw}^0 \left(\frac{S_w - S_{wr}}{1 - S_{wr} - S_{or}} \right)^b. \tag{2.10}$$

We set $k_{ro}^0 = k_{rw}^0 = 1$, $S_{wr} = S_{or} = 0.2$ and $a = b = 2$, as was used in [11].

To systematically investigate the performance of TPWL, we define test cases based on the level of perturbation from the training run. Shown on the left in Figure 2.2 is the bottom hole pressure (BHP) control schedule for the production wells in the

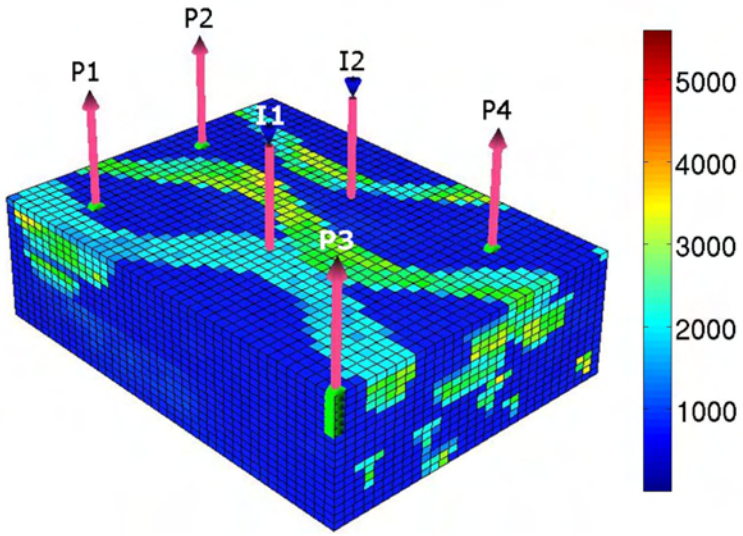


Figure 2.1: Model 1 with four production wells and two injection wells. Permeability in the x-direction (in mD) is shown.

training simulation, which we designate as $\mathbf{u}_{training}$. This schedule is generated randomly, with BHPs between 1000 psi and 3000 psi, and is updated every 200 days. On the right is a different (target) schedule also generated randomly with the same update frequency as the training schedule, though this schedule varies between 2000 psi and 4000 psi. Test cases are specified as a weighted combination of these two schedules; i.e.,

$$\mathbf{u}_{test} = (1 - \alpha)\mathbf{u}_{training} + \alpha\mathbf{u}_{target}, \tag{2.11}$$

where α represents the "distance" of the test controls from the training controls. When α is near zero, test cases are close to the training run and TPWL would be expected to provide accurate results. As α increases toward 1, test cases are further from the training run and larger errors are expected. For this example, the BHPs of the two injection wells are held constant at 6000 psi throughout the simulations.

In this work, errors are quantified in terms of the mismatch of the production rates (for both oil and water) and water injection rates between the full-order solution (Q_{full}) and TPWL simulations (Q_{tpwl}). For example, for oil production rate, the error

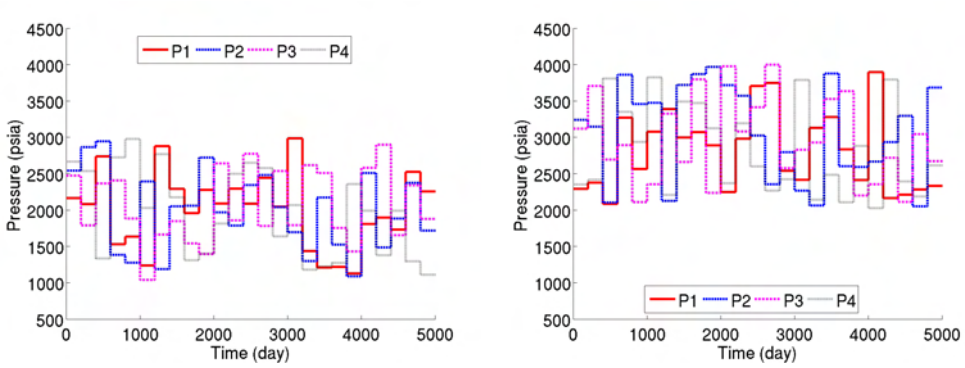


Figure 2.2: Training (left) and target (right) producer BHP schedules for Model 1.

for the jth well (E_o^j) is calculated as:

$$E_o^j = \frac{\int_0^T |Q_{o,full}^j - Q_{o,tpwl}^j| \, \mathrm{d}t}{\int_0^T Q_{o,full}^j}, \tag{2.12}$$

where subscript o designates oil and T is the total simulation time. The overall average error of the oil production rates, designated E_o, is computing by averaging E_o^j over all wells:

$$E_o = \frac{1}{n_w} \sum_{j=1}^{n_{pw}} E_o^j, \tag{2.13}$$

where n_{pw} is the total number of production wells. Similar expressions are used to compute average water production error and water injection error (E_w and E_i, respectively).

We now compare full-order simulation results, generated using Stanford's General Purpose Research Simulator GPRS [29, 30], with TPWL results. For the TPWL model we use $l_p = 40$ and $l_s = 60$. Results for $\alpha = 1.0$ for oil and water production rates are shown in Figure 2.3, while results for water injection rates appear in Figure 2.4. The errors in the TPWL solution for other levels of perturbation are shown in Table 2.1. Discrepancies are evident between the full-order and TPWL results especially for water production and water injection rates. In Chapter 3 these discrepancies are discussed in detail and a local resolution method is proposed to improve

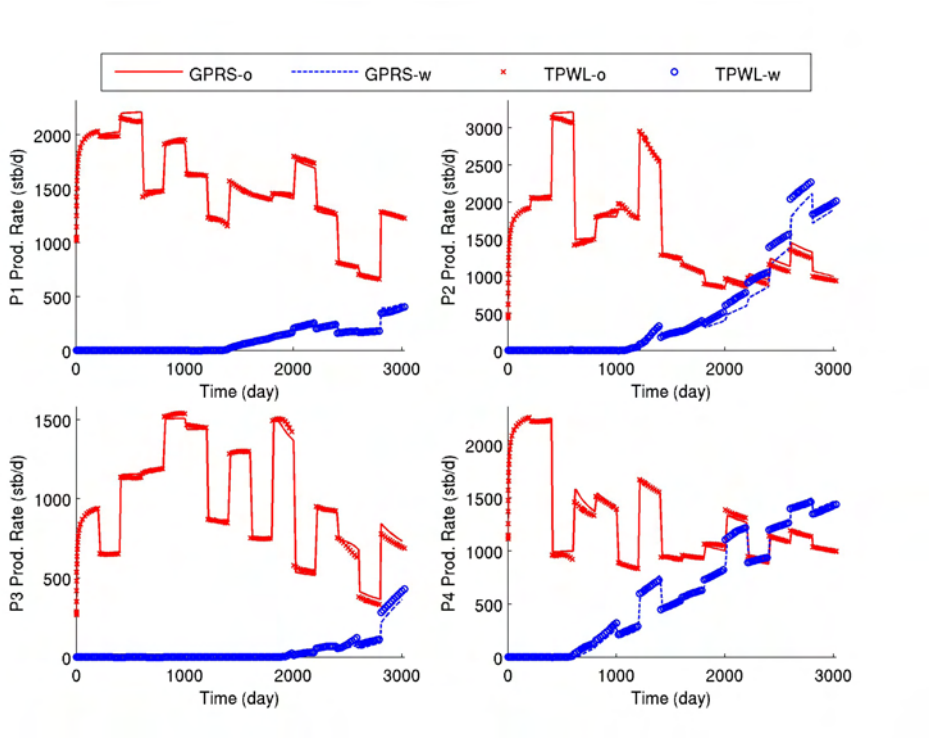


Figure 2.3: Model 1 production rates for $\alpha = 1.0$ using basic TPWL ($l_p = 40$, $l_s = 60$).

the accuracy. However, it should be noted here that this inaccuracy results from the use of $\alpha = 1.0$, which means that the test case is quite different from the training run. As shown in Table 2.1, the use of smaller values of α leads to TPWL results that are much more accurate.

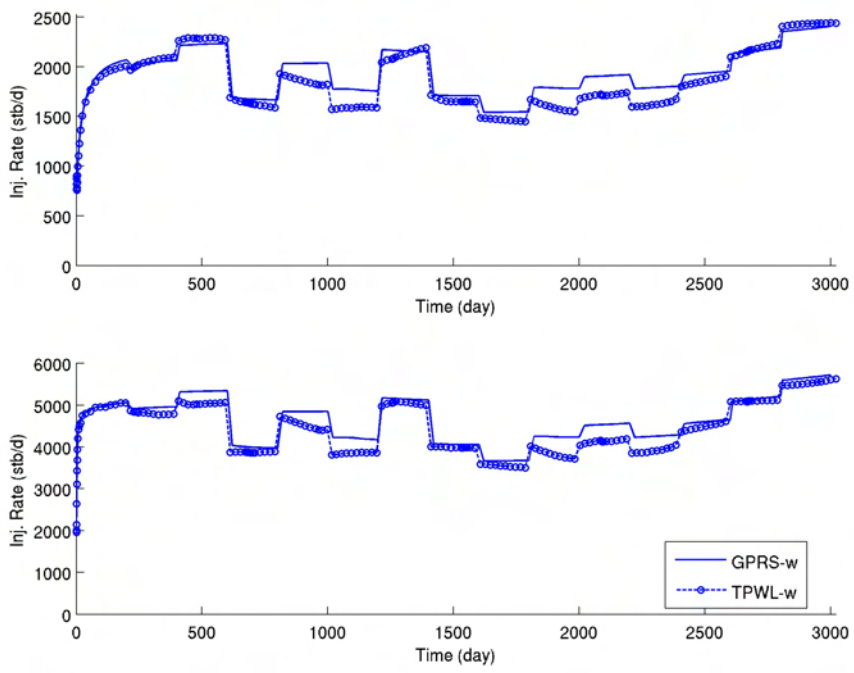


Figure 2.4: Model 1 injection rates for $\alpha = 1.0$ using basic TPWL ($l_p = 40$, $l_s = 60$).

Table 2.1: Relative error in TPWL solutions for various test schedules for Model 1 for basic TPWL with $l_p = 40$ and $l_s = 60$.

	$\alpha = 0$	$\alpha = 0.2$	$\alpha = 0.4$	$\alpha = 0.6$	$\alpha = 0.8$	$\alpha = 1.0$
TPWL(40,60), E_o	0.0039	0.0047	0.0071	0.0108	0.0144	0.0193
TPWL(40,60), E_w	0.0024	0.0110	0.0246	0.0441	0.0733	0.1266
TPWL(40,60), E_i	0.0121	0.0156	0.0209	0.0272	0.0354	0.0456

Chapter 3

Local Resolution TPWL

As shown in Section 2.3, the basic TPWL method can display inaccuracy for cases with large perturbations. In this chapter we will first analyze the problem of underfitting and overfitting. We then propose the local resolution method to enhance the accuracy of the basic TPWL.

3.1 Underfitting and Overfitting

Since $\mathbf{\Phi}$ is orthonormal, we have $\mathbf{z} = \mathbf{\Phi}^T\mathbf{x}$. POD will by construction minimize the mean squared reconstruction error of the training snapshots $(1/S)\Sigma_{i=1}^{S}|\mathbf{\Phi}(\mathbf{\Phi}^T\mathbf{x}^i) - \mathbf{x}^i|^2$, where S is the number of snapshots [25]. Thus, if we consider TPWL solutions based on the full-order model (Eq. 2.5) and the reduced-order model (Eq. 2.9) for $\mathbf{u}^n = \mathbf{u}^i$, and if we take the number of basis vectors stored in $\mathbf{\Phi}$ equal to the number of snapshots, both approaches will reproduce the training states exactly (for Eq. 2.9 we need to apply $\mathbf{x} = \mathbf{\Phi}\mathbf{z}$). However, this will not be the case for states from a new set of target well controls $\mathbf{u}^n \neq \mathbf{u}^i$.

The POD dimension reduction technique is based on the assumption that most of the variability in the snapshot matrices can be represented using a limited number of orthonormal basis vectors $\mathbf{\Phi} = [\boldsymbol{\phi}_1, \ldots, \boldsymbol{\phi}_l]$ [31]. We therefore expect that selecting too few basis vectors will potentially lead to the problem of model underfitting, which can result in large errors in the TPWL solution, both for $\mathbf{u}^n = \mathbf{u}^i$ and $\mathbf{u}^n \neq \mathbf{u}^i$. On the

other hand, including too many basis vectors will potentially cause model overfitting [32], leading to large errors in the TPWL solution for $\mathbf{u}^n \neq \mathbf{u}^i$. This can occur because in the basis matrix obtained from POD, the basis vectors that correspond to smaller eigenvalues are more subject to noise in the training snapshots. Thus, including them in $\mathbf{\Phi}$ can lead to a deterioration of the representation of $\mathbf{x}$ stored in $\mathbf{z}$.

These effects can be observed through an assessment of the sensitivity of the TPWL results to different numbers of basis vectors in $\mathbf{\Phi}$. Errors in injection rate (E_i) for various α are presented in Table 3.1. The smallest error at each value of α is shown in bold. The other errors (E_o and E_w) display generally similar behavior and are not shown. It is apparent from the table that selecting a small number of basis vectors ($l_p = 5$, $l_s = 5$) leads to large errors in the TPWL solution, particularly as α increases. Interestingly, however, error does not necessarily decrease monotonically as we include more basis vectors in $\mathbf{\Phi}$. Consistent with this, the use of the largest l_p and l_s does not provide the most accurate results for water injection for any value of α.

This is in part due to error introduced through overfitting. It is additionally because the POD reduction scheme focuses on the global reconstruction error, not the reconstruction error at the well blocks. Note also that the reason we do not observe a monotonic decrease in the error for the training schedule ($\alpha = 0$) is because, as noted earlier, we use oil potential rather than pressure snapshots to compute the pressure basis matrix. This results in improved accuracy in the TPWL solution for test schedules that differ from the training schedule [11], although it leads to an increase in the reconstruction error for the training schedule.

3.2 Description of Local Resolution Scheme

For our applications, we are mainly interested in the pressure and saturation at well locations because they directly affect injection and production rates, which are the key quantities needed for production optimization. To compute these quantities, we construct $\mathbf{x}_w^{n+1} = \mathbf{\Phi}_w \mathbf{z}^{n+1}$, where $\mathbf{\Phi}_w$ includes only the rows of $\mathbf{\Phi}$ corresponding to the grid blocks containing wells (subscript w here denotes well). Although we are

Table 3.1: Relative error in TPWL solutions for various test schedules for Model 1 with equal density. The notation TPWL(l_p,l_s) in this and subsequent tables denotes the numbers of pressure and saturation basis vectors retained in $\mathbf{\Phi}$. The smallest error at each value of α is shown in bold.

	$\alpha = 0$	$\alpha = 0.2$	$\alpha = 0.4$	$\alpha = 0.6$	$\alpha = 0.8$	$\alpha = 1.0$
TPWL(5,5), E_i	0.0513	0.1245	0.1521	0.2078	0.2602	0.2921
TPWL(20,30), E_i	0.0298	0.0355	0.0423	0.0494	0.0564	0.0664
TPWL(40,60), E_i	**0.0121**	**0.0156**	0.0209	0.0272	0.0354	0.0456
TPWL(70,90), E_i	0.0183	0.0177	**0.0187**	**0.0215**	**0.0281**	**0.0381**
TPWL(120,120), E_i	0.0326	0.0332	0.0351	0.0367	0.0383	0.0416

particularly interested in maintaining accuracy in $\mathbf{\Phi}_w$, the POD dimension reduction technique, by construction, minimizes the *global* reconstruction error of the training snapshots – it does not preferentially weight information at well locations. We can therefore expect reconstruction of the saturation and pressure at well locations to be suboptimal.

This effect is illustrated in Figure 3.1, where we plot the maximum reconstruction error (relative to the full-order solution) of the saturation and pressure snapshots at the injection and production wells for $\alpha = 1.0$. From this figure we see that reconstruction of saturation at the production wells is problematic, and that increasing the number of basis vectors included in $\mathbf{\Phi}$ can lead to a clear increase in the reconstruction error.

It is thus evident that reconstruction error at well locations can significantly impact the accuracy of TPWL results for production and injection rates. To eliminate reconstruction error at the well locations, we therefore propose a TPWL procedure in which selected key grid blocks are represented using the full-order (linearized) model. We let $\mathbf{x}_{LR}$ designate the full-order states for n_{LR} selected grid blocks and $\mathbf{x}_G$ the full-order states for the remaining grid blocks. Then, in place of Eq. 2.6, we write:

$$\begin{bmatrix} \mathbf{x}_{LR} \\ \mathbf{x}_G \end{bmatrix} \approx \begin{bmatrix} \mathbf{\Phi}_{LR} & 0 \\ 0 & \mathbf{\Phi}_G \end{bmatrix} \begin{bmatrix} \mathbf{z}_{LR} \\ \mathbf{z}_G \end{bmatrix}, \tag{3.1}$$

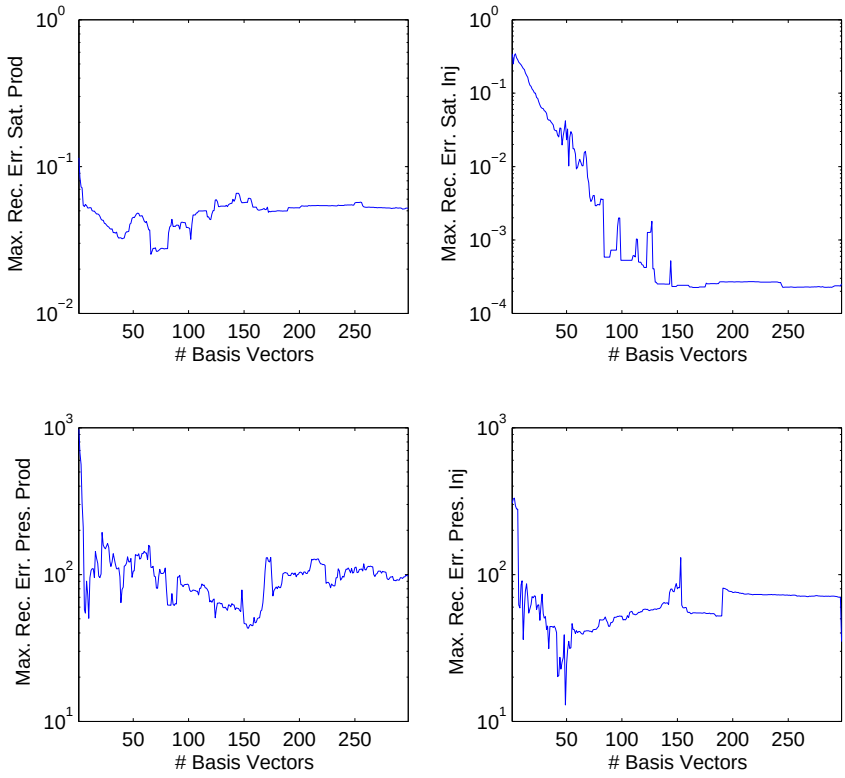


Figure 3.1: Log plot of the maximum reconstruction error for saturation and pressure at production and injection wells for Model 1 with $\alpha = 1.0$.

where $\mathbf{\Phi}_{LR}$ is taken to be the identity matrix (thus $\mathbf{z}_{LR} = \mathbf{x}_{LR}$). This means that the grid blocks associated with $\mathbf{x}_{LR}$ are not subject to reduction, so high local resolution is maintained. Note that, although the local and global grid blocks are decoupled in Eq. 3.1, they do couple in the TPWL scheme defined in Eq. 2.9 through $(\mathbf{J}_r^i)^{-1}$.

In our implementation, the n_{LR} locally-resolved blocks include the well blocks and possibly additional blocks that are important for the flow solution. To determine these additional blocks, we apply the missing point estimation (MPE) procedure suggested by [12]. In this approach we retain blocks that have the largest impact on the condition number of $\mathbf{\Phi}^T\mathbf{\Phi}$, which are blocks that strongly affect the flow solution. A computationally efficient algorithm for MPE is described in [4]. In our implementation, locally resolved blocks are determined separately for the saturation and pressure portions of $\mathbf{\Phi}$. The final set of n_{LR} locally-resolved blocks is the union of these two sets of blocks.

The local resolution method provides flexibility for improving the accuracy of the TPWL representation. As is evident in Figure 3.1 and Table 3.1, this cannot necessarily be accomplished by adding more basis vectors. Even if it could, the maximum number of basis vectors is limited by the number of snapshots. The local resolution method does not have these limitations, and the TPWL model thus defined approaches the full-order model as n_{LR} is increased. Thus, the local resolution method enables us to achieve a balance between accuracy and efficiency.

3.3 Numerical Results using TPWL(LR)

We now apply the local resolution TPWL scheme, designated TPWL(LR), to the case considered earlier (Model 1 with $\alpha = 1.0$). For these runs we use $l_p = 40$, $l_s = 60$ and $n_{LR} = 26$ (local resolution only at well blocks), which corresponds to a total of 152 unknowns. Results are shown in Figures 3.2 and 3.3. Comparing these results with those from the basic TPWL scheme using $l_p = 40$, $l_s = 60$ (Figures 2.3 and 2.4), we see that by eliminating the reconstruction error at wells, the TPWL results are improved significantly, particularly the water injection rates.

Results for flow rate errors for a range of α are shown in Table 3.2. Errors are

shown for TPWL(LR) (with $l_p = 40$, $l_s = 60$, $n_{LR} = 26$) as well as for the basic method for two different sets of (l_p, l_s). The smallest errors for each value of α are shown in bold. TPWL(LR) is the most accurate in all cases, and for some quantities the improvement over the basic TPWL is very significant. These results clearly demonstrate the efficacy of the use of local resolution within a TPWL model.

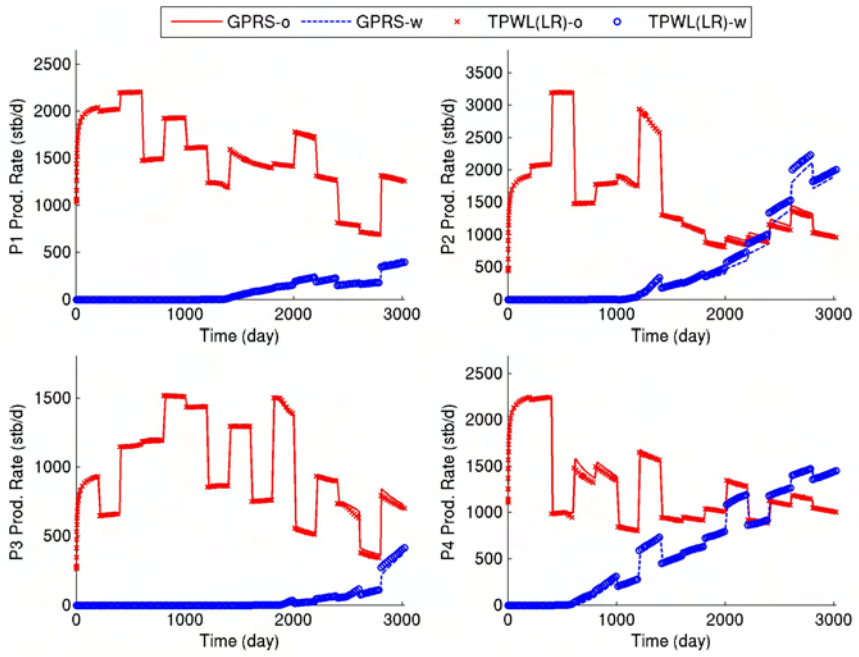


Figure 3.2: Model 1 production rates for $\alpha = 1.0$ using local resolution TPWL ($l_p = 40$, $l_s = 60$, $n_{LR} = 26$).

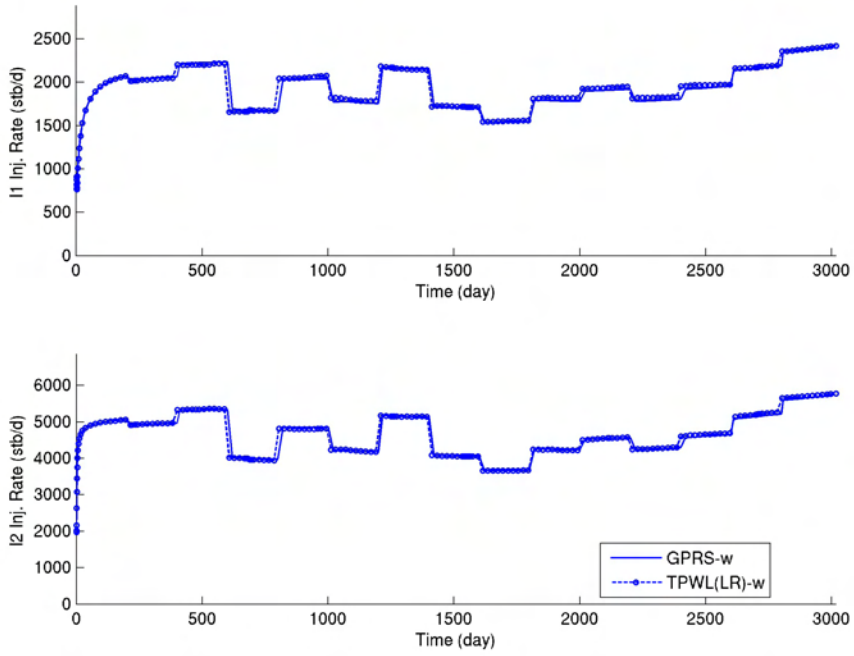


Figure 3.3: Model 1 injection rates for $\alpha = 1.0$ using local resolution TPWL ($l_p = 40$, $l_s = 60$, $n_{LR} = 26$).

Table 3.2: Relative error in TPWL solutions for various test schedules for Model 1. Basic TPWL and local resolution TPWL with $l_p = 40$, $l_s = 60$, $n_{LR} = 26$ are compared. The smallest error at each value of α is shown in bold.

	$\alpha = 0$	$\alpha = 0.2$	$\alpha = 0.4$	$\alpha = 0.6$	$\alpha = 0.8$	$\alpha = 1.0$
TPWL(40,60), E_o	0.0039	0.0047	0.0071	0.0108	0.0144	0.0193
TPWL(70,90), E_o	0.0099	0.0113	0.0154	0.0212	0.0281	0.0357
TPWL(LR), E_o	**0.0002**	**0.0020**	**0.0041**	**0.0065**	**0.0089**	**0.0121**
TPWL(40,60), E_w	0.0024	0.0110	0.0246	0.0441	0.0733	0.1266
TPWL(70,90), E_w	0.0128	0.0229	0.0529	0.1106	0.2135	0.4126
TPWL(LR), E_w	**0.0002**	**0.0090**	**0.0193**	**0.0350**	**0.0547**	**0.1032**
TPWL(40,60), E_i	0.0121	0.0156	0.0209	0.0272	0.0354	0.0456
TPWL(70,90), E_i	0.0183	0.0177	0.0187	0.0215	0.0281	0.0381
TPWL(LR), E_i	**0.0003**	**0.0013**	**0.0025**	**0.0037**	**0.0049**	**0.0067**

Chapter 4

Stability of TPWL Models

As demonstrated through extensive examples in [11, 26], the TPWL procedure can provide reasonable accuracy and robustness for cases with equal phase densities. It was, however, also reported in [11] that the method can become unstable when significant density differences between the two phases exist. In this chapter we will first show an example of this instability. We will then discuss stability criteria and present two methods for stabilizing TPWL.

4.1 Example Showing Instability of TPWL

We consider a model that is the same as that used in Chapter 3, except that now the densities for the two phases are different (here we set $\rho_o = 45$ lb/ft^3 and $\rho_w = 55$ lb/ft^3). Figure 4.1 (upper) displays the oil production rate for well P1 for $\alpha = 0.3$. Small spikes in the solution can be observed at around 500 days and 900 days. Shortly after 2000 days the solution becomes completely unstable and blows up. Results for all other wells display similar behaviors and are not shown here. Clearly, the method requires improvement if it is to be applied to problems of this type. We now consider the stability of the TPWL model.

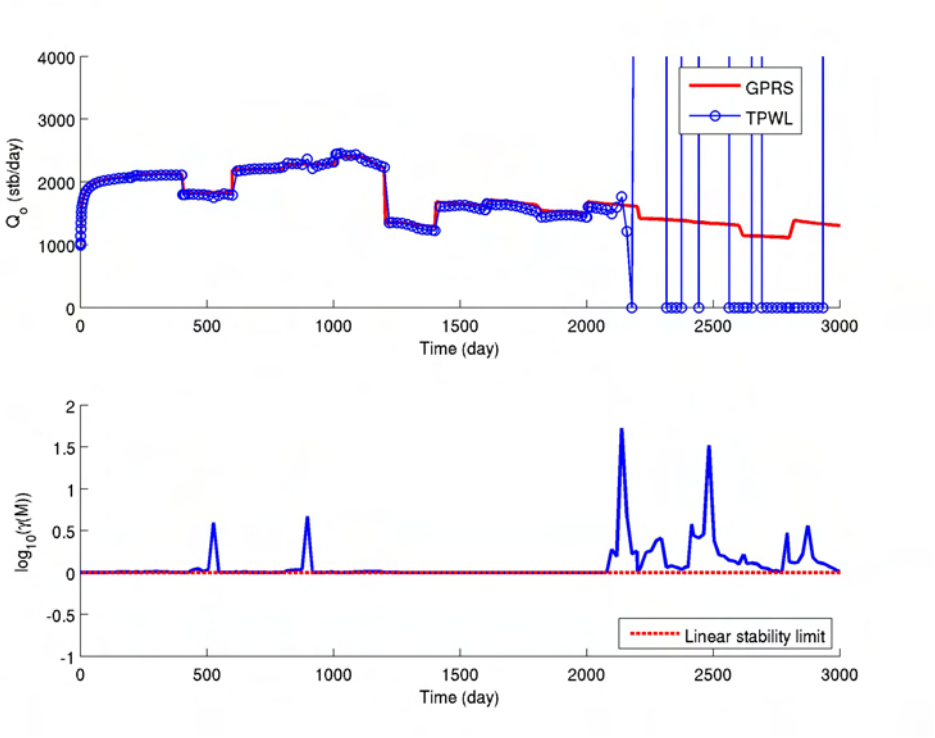


Figure 4.1: Upper: production rate for well P1 using basic TPWL with different phase densities (Model 1, $\alpha = 0.3$, $l_p = 70$, $l_s = 100$). Lower: log of spectral radius of amplification matrix.

4.2 Stability Analysis

We first consider the linearized full-order system, Eq. 2.5. This equation can be viewed as a piecewise linear discrete-time system,

$$\mathbf{x}^{n+1} = \mathbf{M}^{i+1}\mathbf{x}^n + \mathbf{b}^{i+1}, \tag{4.1}$$

where $\mathbf{b}$ is a vector that involves the source term and $\mathbf{M}$ is the amplification matrix given by

$$\mathbf{M}^{i+1} = -\left(\mathbf{J}^{i+1}\right)^{-1} \frac{\partial \mathbf{A}^{i+1}}{\partial \mathbf{x}^i}. \tag{4.2}$$

For a constant $\mathbf{M}$, the system defined in Eq. 4.1 is stable if and only if the spectral radius of $\mathbf{M}$ ($\gamma(\mathbf{M})$) is less than or equal to 1. Here, stability means that the error in $\mathbf{z}^n$ will not be amplified in $\mathbf{z}^{n+1}$.

As can be seen from Eq. 4.2, the amplification matrix $\mathbf{M}^{i+1}$ of Eq. 4.1 is the same as that for the training simulation. Therefore, the full-order linearized system (Eq. 4.1) displays the same numerical stability properties as the original system. This is not the case, however, for the reduced-order model. In this case, the counterpart to Eq. 2.9 is

$$\mathbf{z}^{n+1} = \mathbf{M}_r^{i+1}\mathbf{z}^n + \mathbf{b}_r^{i+1}, \tag{4.3}$$

where the amplification matrix is given by

$$\mathbf{M}_r^{i+1} = -\left(\mathbf{\Phi}^T\mathbf{J}^{i+1}\mathbf{\Phi}\right)^{-1}\mathbf{\Phi}^T\left(\frac{\partial \mathbf{A}^{i+1}}{\partial \mathbf{x}^i}\right)\mathbf{\Phi}. \tag{4.4}$$

For general matrices $\mathbf{J}^{i+1}$ and $\partial\mathbf{A}^{i+1}/\partial\mathbf{x}^i$, $\gamma(\mathbf{M}_r^{i+1})$ can be greater than 1 even when $\gamma(\mathbf{M}^{i+1}) \leq 1$. In general, only special choices of $\mathbf{\Phi}$ can maintain the stability of the system. We note that the spectral radius of $\mathbf{M}_r^{i+1}$ affects the stability of the linearized system at the time step when it is used. An isolated unstable $\mathbf{M}_r^{i+1}$ will amplify the error at a specific time step and may create a spike in the solution. The solution may still be able to recover if the perturbation is not too large and if subsequent $\mathbf{M}_r^{i+1}$ are stable. However, if we have several consecutive time steps with unstable $\mathbf{M}_r^{i+1}$, the error will amplify and the solution may blow up. Therefore, to ensure that the error

does not accumulate over time, it is necessary to require the linearized system to be stable for all time steps.

Figure 4.1 (lower) presents the log of the spectral radius of the amplification matrix at each time step. Instability occurs for $\log \gamma > 0$. It is apparent that the spikes and eventual blowup of the solution correspond to values of $\log \gamma$ that are significantly greater than zero and thus unstable. The isolated unstable points at around 500 days and 900 days produce small spikes in the solution, while the consecutive unstable points starting at around 2200 days lead to solution blowup. Thus it is clear that the use of the TPWL method with a POD basis and Galerkin projection can result in instability for this problem.

The loss of stability for reduced-order models, especially those based on POD and Galerkin projection, has been studied previously and several methods to enhance stability have been proposed [14, 17]. Two basic types of stabilization procedures have been considered. The idea of the first set of methods is to compute a left projection matrix that stabilizes the system, rather than use $\mathbf{\Phi}^T$ as in Galerkin projection. In [13–15], this new left projection matrix was based on Lyapunov stability theory. However, if only stability is considered, the accuracy of the reduced-order model can degrade considerably. If accuracy is also taken into account, a matrix optimization has to be solved to obtain an optimal left projection matrix, which is very expensive computationally for large systems. The other group of methods focuses on finding a basis other than POD that can guarantee stability under Galerkin projection. In [17], a goal-oriented, model-constrained optimization problem was formulated to determine the optimal $\mathbf{\Phi}$. However, the procedure involves calculation in the full-order space. Even when the basis matrix is parameterized by the snapshots, as in [17], the optimization can still be very computationally expensive.

Neither of the approaches described above maintains both POD and Galerkin projection while stabilizing the result. We will present two relatively efficient ways to stabilize TPWL in the following sections. Both apply a POD basis and Galerkin projection.

4.3 Stabilizing TPWL

4.3.1 Stabilization using Optimized Basis

The goal of this method is to improve stability while maintaining POD and Galerkin projection. The method is based on the observation that the stability of the reduced-order model is sensitive to l_p and l_s, which define the number of columns in $\mathbf{\Phi}$. The dependence of $\mathbf{M}_r$ on $\mathbf{\Phi}$ is evident in Eq. 4.4.

Figure 4.2 depicts $\log(\gamma(\mathbf{M}_r))$ for different values of l_p and l_s for two particular saved points, $i = 70$ and $i = 100$, for the problem with different phase densities defined in Section 4.1. In the figure, the lower bound of the color bar is the linear stability limit. Therefore the dark blue regions indicate combinations of l_p and l_s that give a stable linearized reduced system while other colors represent different levels of instability. It can be seen that the relation between stability and the number of basis vectors is somewhat random and that the use of more basis vectors does not necessarily lead to improved stability. This demonstrates that the widely used energy criterion (see, e.g., [26]) for selecting the number of basis vectors based on singular values may lead to stability problems. However, it can also be seen that for many of the (l_p, l_s) combinations, the spectral radius is less than 1. This means that if these combinations are used to generate the basis, the resulting reduced system will be stable.

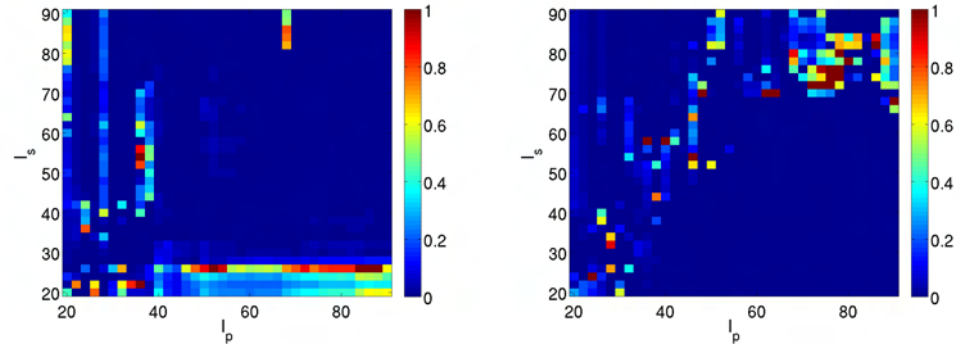


Figure 4.2: $\log_{10} \mathbf{M}_r^i$ for $i = 70$ and $i = 100$.

The idea of basis optimization is to define a range of l_p and l_s and to determine values (designated l_p^i and l_s^i) that minimize $\gamma(\mathbf{M}_r)$; i.e.,

$$(l_p^i, l_s^i) = \arg\min_{l_p^i, l_s^i} \gamma(\mathbf{M}_r^i). \tag{4.5}$$

This is accomplished using an exhaustive search over the allowable range of l_p and l_s (with prescribed increments in l_p and l_s). If we select different l_p and l_s for different time steps, which may be necessary in some cases, the reduced space changes in time. It is therefore necessary to map $\mathbf{z}$ in one reduced space to $\mathbf{z}$ in another reduced space. This is accomplished using

$$\mathbf{z}_\beta = \mathbf{\Phi}_\beta \mathbf{\Phi}_\alpha \mathbf{z}_\alpha, \tag{4.6}$$

where subscripts α and β indicate the two reduced spaces.

The optimized basis (OB) procedure is summarized in Algorithm 1. First, a range for l_p and l_s is specified. Then the spectral radius is calculated for selected combinations of l_p and l_s. The combination that gives the smallest γ is selected. This method takes into account both accuracy and stability. Accuracy is controlled approximately by the search range, and stability is improved by choosing the optimal (l_p, l_s) combinations.

The optimization procedure is reasonably efficient because $\mathbf{J}_r^i$ and $(\partial \mathbf{A}^{i+1}/\partial \mathbf{x}^i)_r$ only need to be calculated once, for $l_{p,max}$ and $l_{s,max}$. Denoting these matrices as $\mathbf{J}_{r,max}^i$ and $(\partial \mathbf{A}^{i+1}/\partial \mathbf{x}^i)_{r,max}$, $\mathbf{J}_r^i$ and $(\partial \mathbf{A}^{i+1}/\partial \mathbf{x}^i)_r$ for any other (l_p^i, l_s^i) combination are just submatrices of $\mathbf{J}_{r,max}^i$ and $(\partial \mathbf{A}^{i+1}/\partial \mathbf{x}^i)_{r,max}$ and can be extracted directly. Therefore the matrix operations inside the optimization loop are all in reduced space. Furthermore, the optimal (l_p^i, l_s^i) combinations only need to be determined once during the preprocessing and will not add to the runtime of TPWL. Thus this optimization does not overly affect the efficiency of the TPWL model.

This method differs from the two types of methods developed previously in that it maintains the advantages of POD and the Galerkin projection. Specifically, POD provides optimal accuracy in representing the snapshots and the Galerkin projection is straightforward and efficient. It is, however, important to note that the algorithm does not guarantee stability. Nevertheless, it does lead to significant improvements in

Algorithm 1: Selecting the Number of Basis Vectors to Achieve Stability.

Input: Training data obtained using GPRS. Search region for $l_p^i \in [l_{p,min}, l_{p,max}]$ and $l_s^i \in [l_{s,min}, l_{s,max}]$

```
[Φ, Λ] = SVD(X)
Keep first l_{p,max} + l_{s,max} components
Compute J^i_{r,max} and (∂A^{i+1}/∂x^i)_{r,max} based on l_{p,max} and l_{s,max} for all time steps i
for each time step do
    for l^i_p = l_{p,min} to l_{p,max} do
        for l^i_s = l_{s,min} to l_{s,max} do
            Extract columns 1,...,l_p, 1,...,l_s, of J^i_{r,max} and (∂A^{i+1}/∂x^i)_{r,max}
            Construct amplification matrix M^i_r
            Calculate γ(M^i_r)
    Select the optimal values l^i_p and l^i_s that minimize γ(M^i_r)
    Construct the basis Φ^i
```

stability and enables the solution of challenging problems with significant differences in density between phases.

Figure 4.3 displays the results when applying this technique to the problem described in Section 4.1. In this case a single set of optimized l_p and l_s was used ($l_p = 45$, $l_s = 60$); i.e., basis switching was not required. It is evident from Figure 4.3 that, after basis optimization, $\gamma(\mathbf{M}_r)$ is close to 1, which means that TPWL is stabilized. The oil production for well P1 is seen to be in reasonable agreement with the reference full-order results, with the solution blowup after 2000 days eliminated (compare with Figure 4.1). We note that the use of local resolution will act to further improve the accuracy of the TPWL solution.

It is of interest to note that spikes in $\gamma(\mathbf{M}_r)$ usually correspond to spikes in the condition number of $\mathbf{J}_r^{-1}$ (or $\mathbf{J}_r$). This is potentially of concern since $\mathbf{J}_r^{-1}$ appears not only in $\mathbf{M}_r$ but also in $\mathbf{b}_r$ in Eq. 4.3. Thus, even when the system is stable, spikes in the condition number of $\mathbf{J}_r^{-1}$ can still cause inaccuracy in the solution. In such cases, we may need to determine optimum (l_p, l_s) such that both $\gamma(\mathbf{M}_r)$ and the condition number of $\mathbf{J}_r^{-1}$ are minimized to assure solution stability and accuracy. This will be

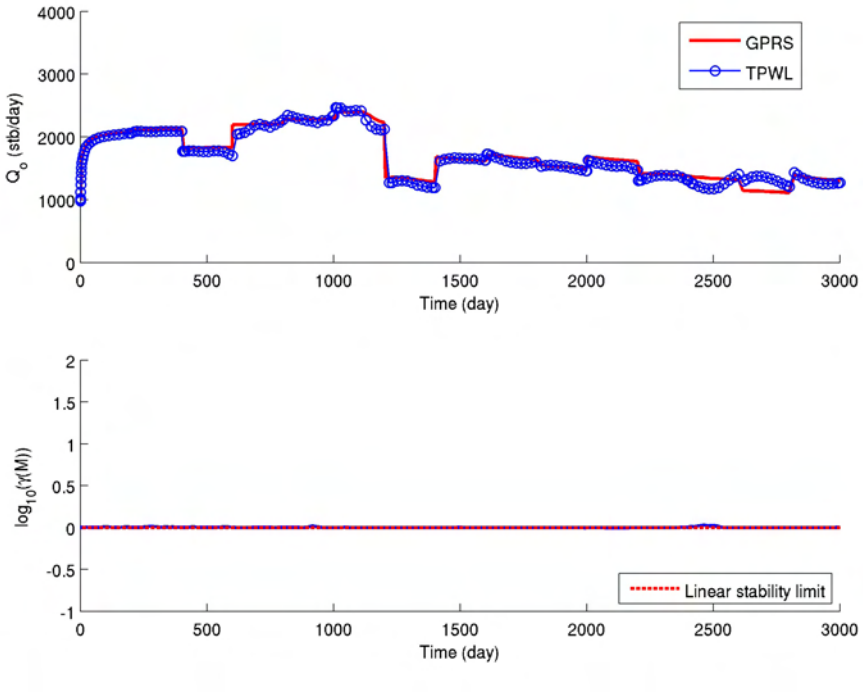


Figure 4.3: Upper: production rate for well P1 using TPWL with optimized basis for case with different phase densities (Model 1, $\alpha = 0.3$, $l_p = 45$, $l_s = 60$). Lower: log of spectral radius of amplification matrix.

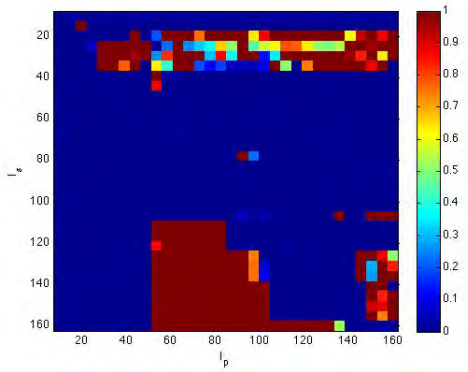
addressed in future work.

4.3.2 Stabilization using Modified Basis

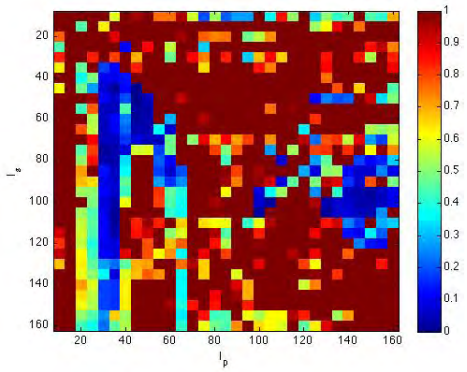
We now introduce a stabilization procedure in which $\mathbf{\Phi}$ is derived from a problem with better stability characteristics than the target problem with $\Delta\rho \neq 0$. This approach can be motivated with reference to stability maps. Figure 4.4(a) shows the maximum spectral radius for all saved points as a function of l_p and l_s for Model 1 with $\rho_o = \rho_w$. Here we see that the TPWL scheme is unstable only for a limited number of combinations of l_p and l_s. However, the stability map for the same reservoir model but with $\Delta\rho = 10$, shown in Figure 4.4(b), indicates only isolated regions where the TPWL model is stable. This motivates the construction of a model reduction scheme where the basis matrix is constructed using snapshots generated from a reservoir model with $\Delta\rho = 0$. The states and Jacobian matrices used in the actual TPWL solution (Eq. 2.9) are still from the specific problem of interest (with $\Delta\rho \neq 0$). We will refer to this procedure as the Equal Density Projection (EDP) scheme. A stability analysis of the EDP scheme for different values of l_p and l_s reveals that the regions where the TPWL solution is stable, seen in Figure 4.4(c), correspond to the stability regions in Figure 4.4(a). Thus the approach inherits the stability properties of the $\Delta\rho = 0$ TPWL model.

Because the EDP basis matrix is constructed using snapshots which are different from the states in the actual solution, the global reconstruction error for $\mathbf{z}^i$ will necessarily increase dramatically compared with any of the approaches considered above. However, combining the EDP scheme with local resolution TPWL effectively eliminates the reconstruction error at key locations such as well blocks.

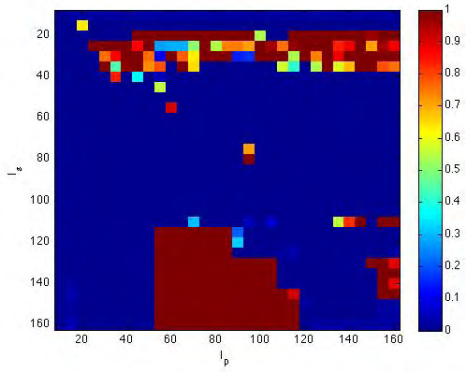
Figure 4.5 shows the oil and water production rates for Model 1 with $\Delta\rho = 10$ and $\alpha = 0.3$. Here we apply both EDP and LR and set $l_p = 70$, $l_s = 100$ and $n_{LR} = 27$ in the TPWL(EDP+LR) solution. Note that the use of $l_p = 70$, $l_s = 100$ in the basic TPWL method leads to instability (see Figure 4.1). As is evident in Figure 4.5, the TPWL(EDP+LR) scheme, by contrast, is able to provide a stable and reasonably accurate solution relative to the full-order results. This is further illustrated in the



(a) $\log_{10} \max\{\gamma(\mathbf{M}^i)\}$, $\Delta\rho = 0$, POD



(b) $\log_{10} \max\{\gamma(\mathbf{M}^i)\}$, $\Delta\rho = 10$, POD



(c) $\log_{10} \max\{\gamma(\mathbf{M}^i)\}$, $\Delta\rho = 10$, EDP

Figure 4.4: Maximum value for the spectral radius of the amplification matrix, $\mathbf{M}_r^i$, as a function of number of basis vector retained for Model 1 using (a) $\Delta\rho = 0$ with standard POD, (b) $\Delta\rho = 10$ with standard POD, and (c) $\Delta\rho = 10$ with EDP scheme.

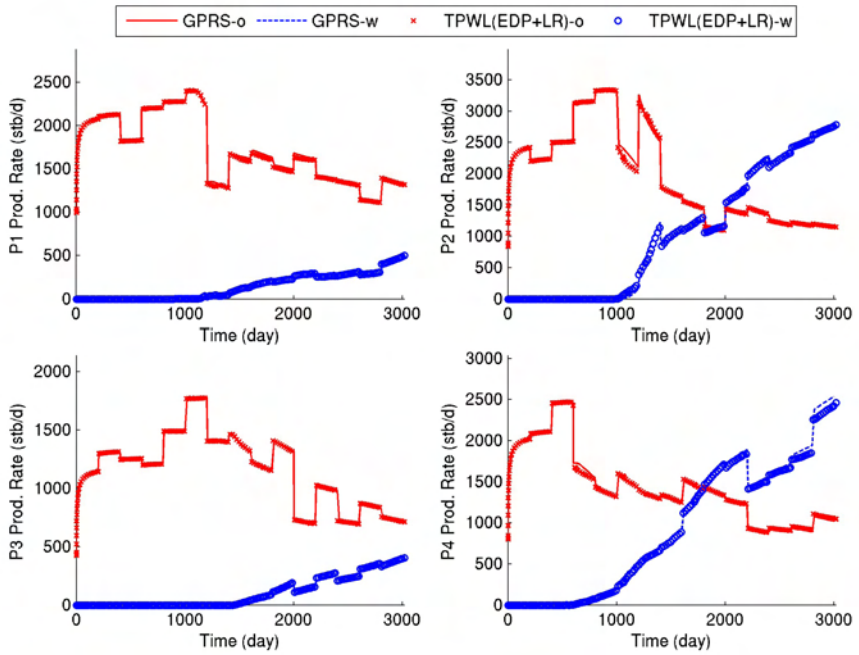


Figure 4.5: Model 1 production rates for $\alpha = 0.3$ using TPWL(EDP+LR) ($l_p = 70$, $l_s = 100$, $n_{LR} = 27$).

results for water injection rates shown in Figure 4.6.

We do not yet have a complete explanation for why the basis matrix generated from snapshots of the corresponding equal density case ($\mathbf{\Phi}_{ED}$) gives better stability than the basis matrix generated from snapshots of the actual (different density) case. Referring to this latter basis as $\mathbf{\Phi}_{DD}$, we observe that $\mathbf{\Phi}_{DD}$ tends to have more extreme values than $\mathbf{\Phi}_{ED}$. These appear to derive from the fact that changes between snapshots in cases with different densities are more localized than in cases with equal densities. This occurs because of the different physics in the two types of problems. In any event, the use of $\mathbf{\Phi}_{DD}$ containing these extreme values leads to high condition numbers in $\mathbf{J}_r$ and instability. On the other hand, $\mathbf{\Phi}_{ED}$ contains more evenly distributed values and as a result provides better model stability.

This argument is also consistent with results from the Random Projection (RP) scheme [33]. In the RP scheme, the pressure and saturation snapshots are replaced

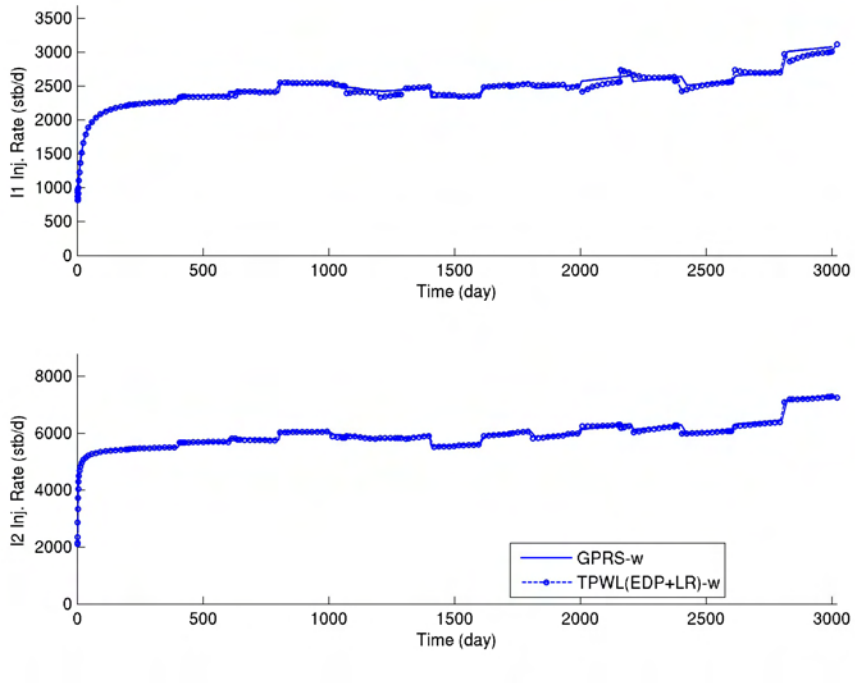


Figure 4.6: Model 1 injection rates for $\alpha = 0.3$ using TPWL(EDP+LR) ($l_p = 70$, $l_s = 100$, $n_{LR} = 27$).

with random vectors generated independently from a Gaussian probability distribution with zero mean and unit variance. The variance between snapshots is thus evenly distributed spatially. Orthonormalization of these vectors provides the basis matrix. Interestingly, this approach leads to a stable TPWL scheme for any combination of l_p and l_s. In numerous tests, we achieved stable TPWL results for all of the reservoir models considered in this paper, including those where the basic TPWL method exhibited instability. The accuracy of the RP scheme is, however, quite poor as it does not use any information from the snapshots. The EDP scheme can be viewed as an enhanced or supervised version of the RP scheme in the sense that, by using a particular set of snapshots, it achieves stability at the cost of accuracy. As noted above, accuracy is recovered through use of local resolution. We note finally that the accuracy of the RP scheme can also be improved using local resolution, but n_{LR} needs to be very large before adequate levels of accuracy are consistently achieved.

4.3.3 Summary

The two stabilization schemes proposed in this chapter have somewhat different features. The optimized basis method requires only one full-order simulation, though it does require basis optimization computations. In many cases it performs reasonably well without the use of local resolution. However, it has limited flexibility in the choice of l_p and l_s as they are determined based on stability properties. The EDP method, by contrast, requires two full-order simulations and it must be combined with local resolution to provide reasonable accuracy. It has more flexibility, however, in the choice of l_p and l_s. Thus both methods have advantages and limitations, and further application and development of both approaches appears to be warranted.

Chapter 5

Application of TPWL to Realistic Problems

In this chapter we apply the enhanced TPWL procedure to two realistic reservoir models. The models contain $O(10^5)$ grid blocks and the oil and water phases are of different densities. For these examples we combine the stabilization methods with the local resolution procedure, and present results for a range of perturbations α. The basic TPWL procedure has difficulty providing stable solutions for these cases.

5.1 Model 2: Upper Six Layers of SPE 10

The geological model used here is shown in Figure 5.1. The model, referred to as Model 2, comprises the upper six layers of the so-called SPE 10 geological model, developed by Christie and Blunt [34]. The model contains 79,200 grid blocks (with $n_x = 60$, $n_y = 220$, $n_z = 6$). This model was also studied in [11], where it was applied for an example with equal phase densities, for which accurate TPWL results were reported. The problems that can arise using the basic TPWL procedure for cases with unequal densities were also illustrated using this model in [11]. Model 2 includes four producers, which are perforated in the upper two layers, and two injectors, perforated in the lower two layers. The rock and fluid properties are the same as were used in Model 1 (see Section 2.3) except that here we set the density

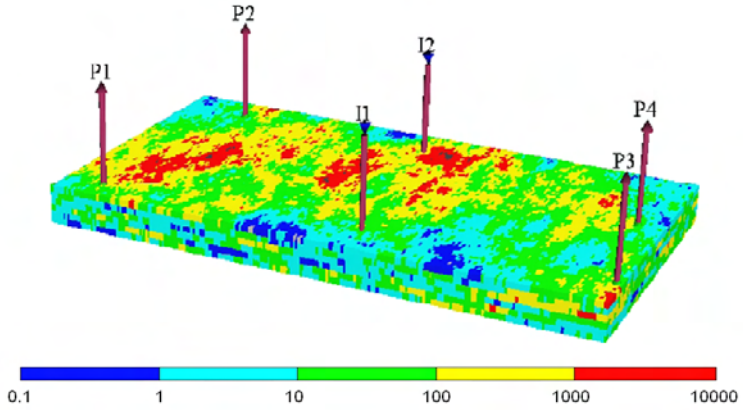


Figure 5.1: Upper six layers of the SPE 10 reservoir model (79,200 grid blocks) with four producers and two injectors. Permeability in x-direction (in mD) is shown.

for water to be ρ_w=60 lb/ft^3, which gives a larger density difference.

The performance of the TPWL procedure for this problem is again studied using perturbation tests. The training schedule and target BHP schedules for this case are shown in Figure 5.1. For producers, the training BHP schedules vary randomly between 1000 psi and 3000 psi and are changed every 200 days. The target BHP schedules vary randomly between 1000 psi and 4000 psi; they also change every 200 days. Clear differences are evident between the two schedules. For the injectors, the training BHP schedules are constant at 8000 psi while the target BHP schedules vary randomly between 7000 psi and 9000 psi. Again, input BHP schedules for test cases are generated as weighted combinations of the training and target schedules, as defined in Eq. 2.11. The training simulation is run for 5000 days and produces 311 pressure and saturation snapshots.

We apply both the optimized basis method and the EDP procedure to generate stable TPWL models. For the optimized basis method, we considered l_p in the range [40, 90] and l_s in the range [60, 90]. The search increment for both was 5. A basis with $l_p = 90$ and $l_s = 85$ was found to be stable for the entire simulation, so no basis switching was applied in this case. A total of 154 blocks were locally resolved. These include the 12 well blocks and 144 additional blocks determined by the MPE procedure.

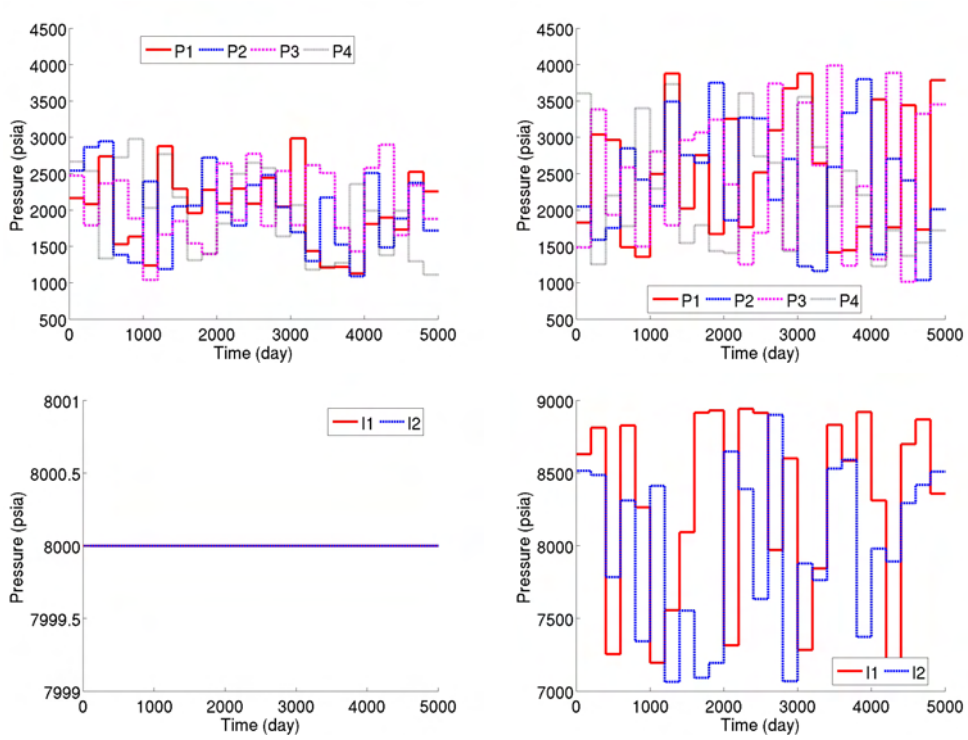


Figure 5.2: Training (left) and target (right) BHP schedules for producers and injectors for Model 2.

Test cases were run for values of α from 0 to 1. Results for oil and water production rates, for $\alpha = 0.5$, are shown in Figure 5.3. Water injection rates are presented in Figure 5.4. It is clear that the TPWL model performs reasonably well for this case. Although slight mismatches are observed for some quantities, the general level of accuracy of the TPWL solution is quite acceptable and the method is clearly stable.

We now present results for this case using the EDP method for stabilization. This requires that the training simulations be run twice – once using the actual densities and once using equal densities. The equal density run is used only to construct $\mathbf{\Phi}$; the saved states and matrices are from the run using the actual densities. To allow direct comparison with TPWL(OB), we also use $l_p = 90$, $l_s = 85$ and $n_{LR} = 154$.

Figures 5.5 and 5.6 display results for oil and water production rates and water injection rates for $\alpha = 0.5$. We again observe stability and a reasonable degree of

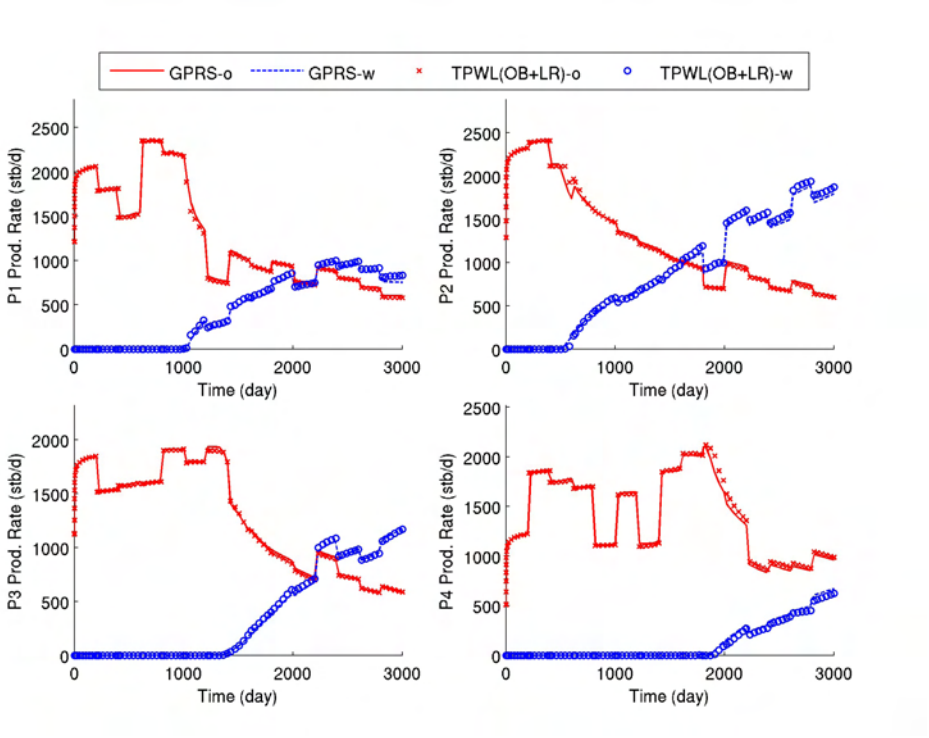


Figure 5.3: Model 2 production rates for $\alpha = 0.5$ using TPWL with optimized basis and local resolution ($l_p = 90$, $l_s = 85$, $n_{LR} = 154$).

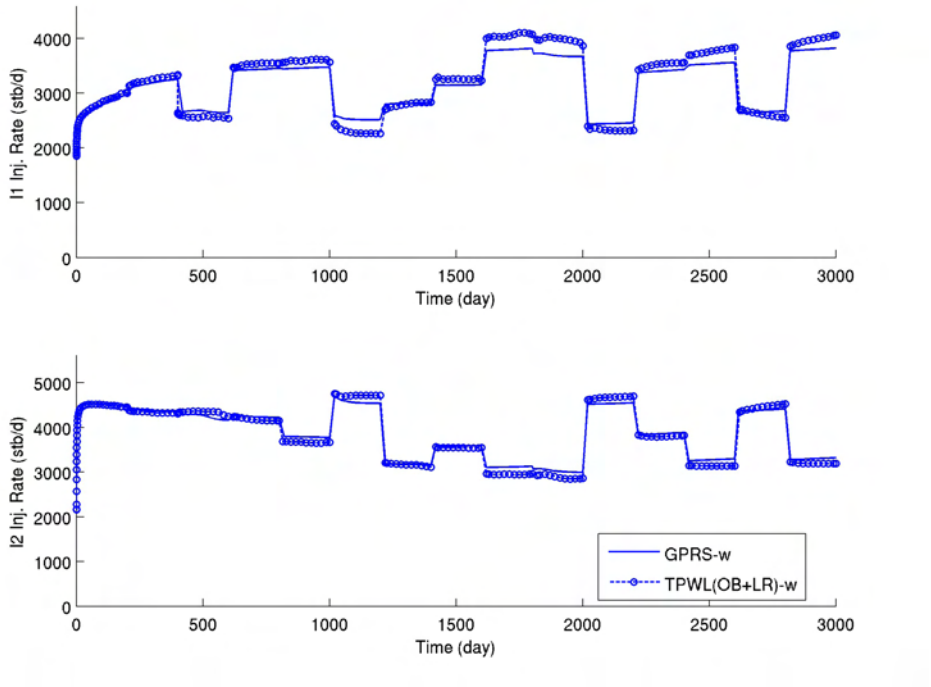


Figure 5.4: Model 2 injection rates for $\alpha = 0.5$ using TPWL with optimized basis and local resolution ($l_p = 90$, $l_s = 85$, $n_{LR} = 154$).

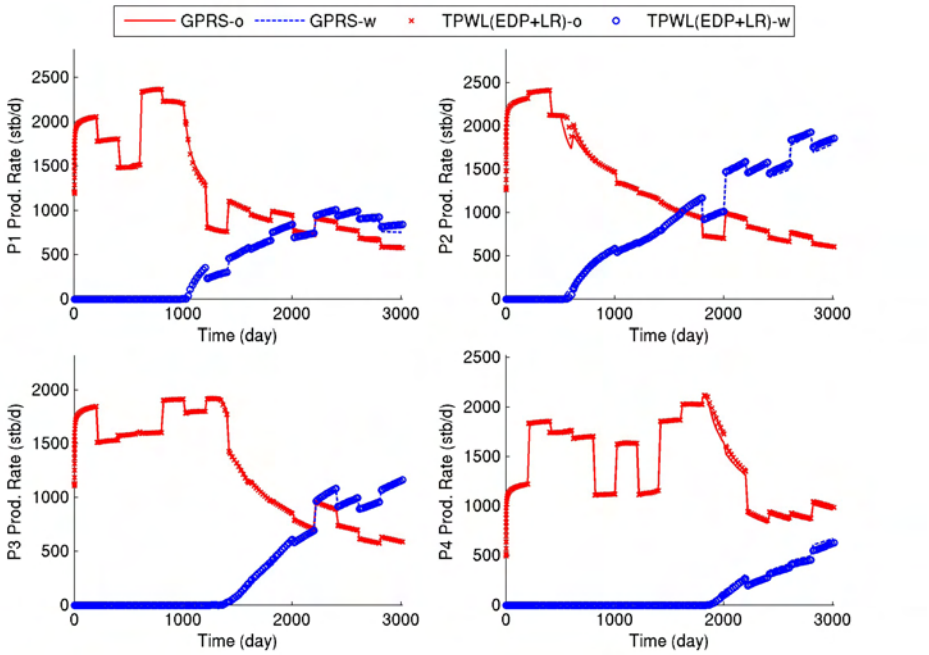


Figure 5.5: Model 2 production rates for $\alpha = 0.5$ using TPWL with EDP basis and local resolution ($l_p = 90$, $l_s = 85$, $n_{LR} = 154$).

accuracy in the TPWL results. In fact, these results are quite comparable to those using the optimized basis procedure, shown in Figures 5.3 and 5.4.

Table 5.1 presents errors for oil and water production rates and water injection rates for the two sets of TPWL solutions at five values of α. Errors for both methods increase consistently with α, as would be expected. The EDP TPWL method is slightly more accurate than TPWL with the optimized basis, though the magnitudes of the errors are quite comparable.

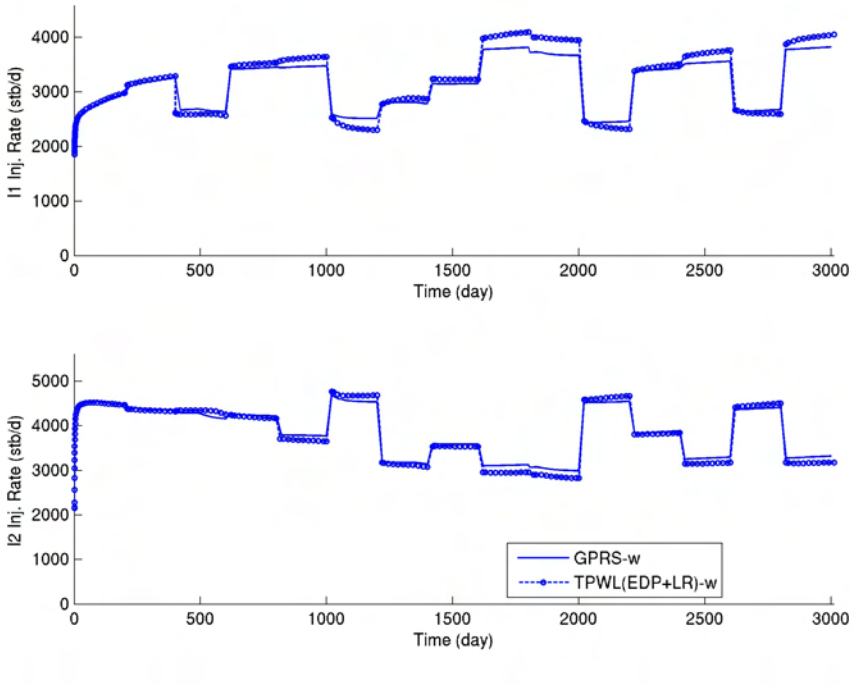


Figure 5.6: Model 2 injection rates for $\alpha = 0.5$ using TPWL with EDP basis and local resolution ($l_p = 90$, $l_s = 85$, $n_{LR} = 154$).

Table 5.1: Errors for TPWL(OB+LR) and TPWL(EDP+LR) for Model 2.

Method	$\alpha = 0.1$	0.3	0.5	0.7	0.9
TPWL(OB+LR), E_o	0.0029	0.0088	0.0142	0.0203	0.0263
TPWL(OB+LR), E_w	0.0116	0.0351	0.0582	0.0819	0.1063
TPWL(OB+LR), E_i	0.0065	0.0194	0.0321	0.0449	0.0585
TPWL(EDP+LR), E_o	0.0022	0.0067	0.0116	0.0166	0.0220
TPWL(EDP+LR), E_w	0.0097	0.0293	0.0495	0.0702	0.0912
TPWL(EDP+LR), E_i	0.0059	0.0174	0.0292	0.0409	0.0531

5.2 Model 3: Portion of Upper 30 Layers of SPE 10

We now consider a more complex case. This model, also extracted from the geological description in [34], contains 108,000 grid blocks ($n_x = 60$, $n_y = 60$, $n_z = 30$). The permeability distribution is shown in Figure 5.7. The model is referred to as Model 3 and includes four producers perforated in the upper 12 layers and two injectors perforated in the lower 12 layers. There are thus a total of 72 well blocks. Well indices were set to a specified value to avoid very large discrepancies in the well rates. The rock and fluid properties are the same as were used in Model 2; the density difference between phases is again 15 lb/ft^3. This case is challenging not only because of its size but also because it has more and thicker layers, which means that density-driven gravitational effects can be large.

The training and target BHP schedules are the same as were used for Model 2, as shown in Figure 5.1. The training simulation was run for 5000 days. A total of 314 pressure and saturation snapshots were saved.

We again present results using both stabilization methods. For the optimized basis method, l_p and l_s were both evaluated over the range [60, 120] with increments of 5. The optimal combination, $l_p = 90$ and $l_s = 90$, was found to be stable for the entire run, so basis switching was not required. Local resolution is applied only to the 72 well blocks. Therefore, the total dimension of the reduced space is 324, compared

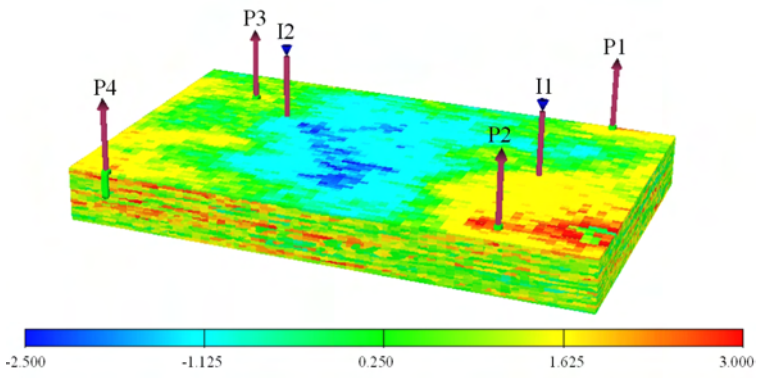


Figure 5.7: Portion of the SPE 10 reservoir model (108,000 grid blocks) with four producers and two injectors. Log of permeability in x-direction is shown.

with 204,000 for the full-order model.

Test cases were again run for α between 0 and 1. Figures 5.8 and 5.9 display results for production and injection rates for $\alpha = 0.5$. The results are stable and generally accurate and are of about the same quality as those presented for Model 2. This is very encouraging, as Model 3 represents a more challenging test case.

We also generated a TPWL model for this case using the EDP approach, again with $l_p = 90$, $l_s = 90$ and $n_{LR} = 72$. As indicated above, constructing this TPWL model requires that two training runs be performed. Results for production and injection rates for $\alpha = 0.5$ are shown in Figures 5.10 and 5.11. These results are again very comparable to those using the optimized basis method.

Errors for both sets of runs for a range of α are presented in Table 5.2. As was observed for Model 2, error increases with increasing α and the EDP method is slightly more accurate than the optimized basis method, though both provide comparable levels of accuracy.

The runtime for the full-order models for the two examples considered in this chapter were 30 minutes and 50 minutes respectively on an Opteron dual-core CPU. The TPWL models, by contrast, required at most a few seconds. Thus runtime speedups of $O(10^3)$ were achieved. We reiterate, however, that the overhead associated with TPWL is equivalent to about 1–2 full-order simulations.

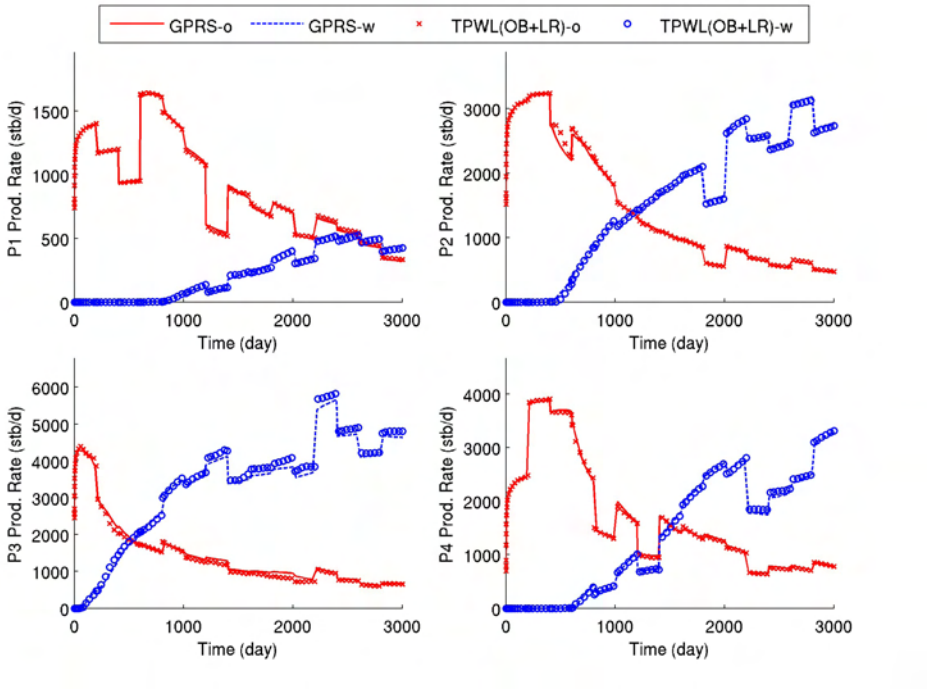


Figure 5.8: Model 3 production rates for $\alpha = 0.5$ using TPWL with optimized basis and local resolution ($l_p = 90$, $l_s = 90$, $n_{LR} = 72$).

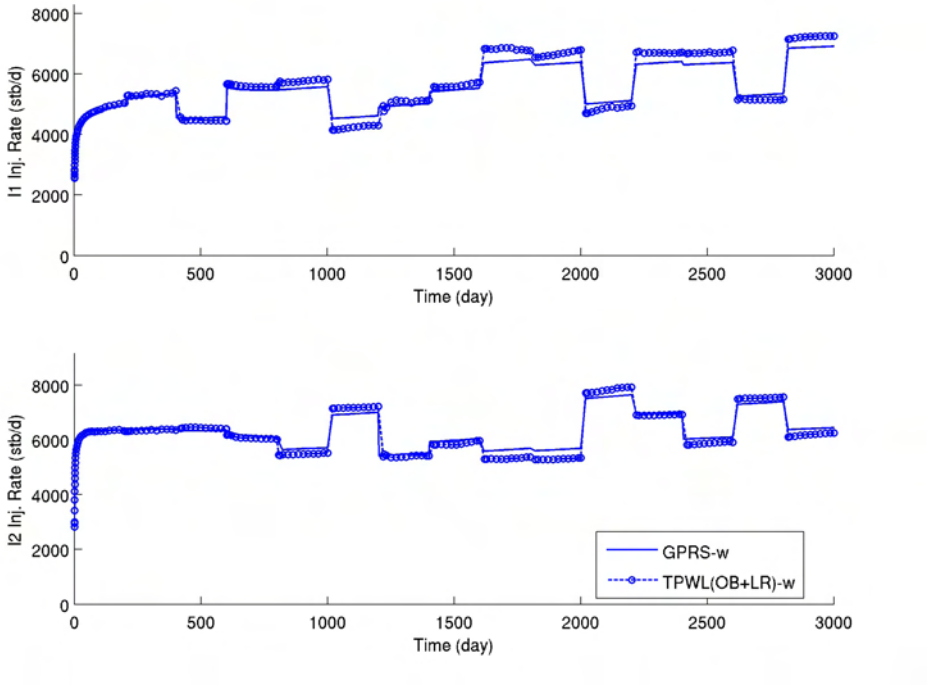


Figure 5.9: Model 3 injection rates for $\alpha = 0.5$ using TPWL with optimized basis and local resolution ($l_p = 90$, $l_s = 90$, $n_{LR} = 72$).

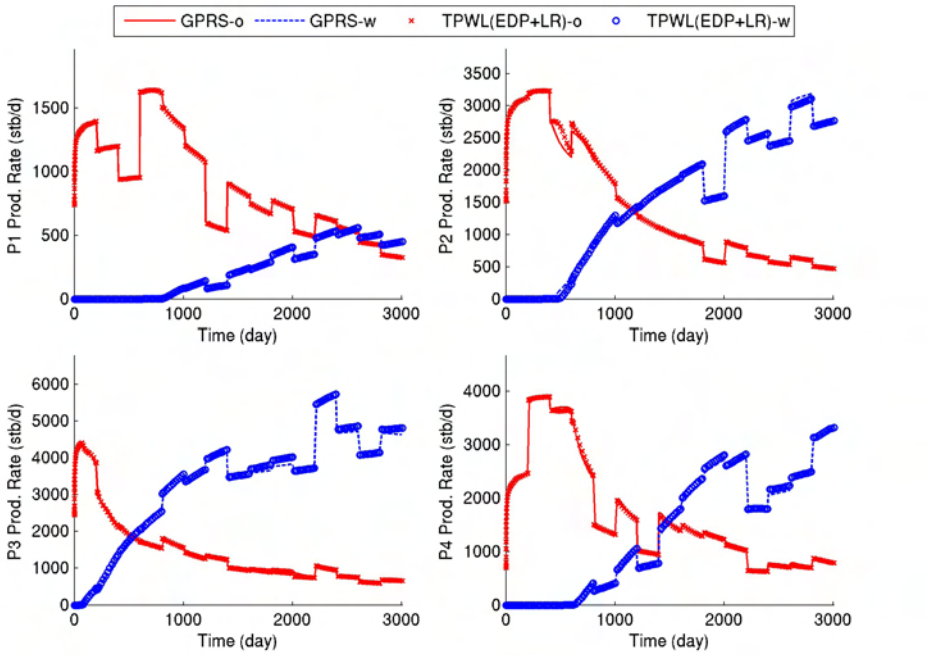


Figure 5.10: Model 3 production rates for $\alpha = 0.5$ using TPWL with EDP basis and local resolution ($l_p = 90$, $l_s = 90$, $n_{LR} = 72$).

Taken in total, the results presented in this chapter demonstrate that our enhanced TPWL procedures are able to provide stable results of reasonable accuracy for challenging reservoir simulation problems. This suggests that these approaches may indeed be applicable in computational optimization or uncertainty quantification procedures. In the next chapter, we apply the enhanced TPWL method to production optimization problems.

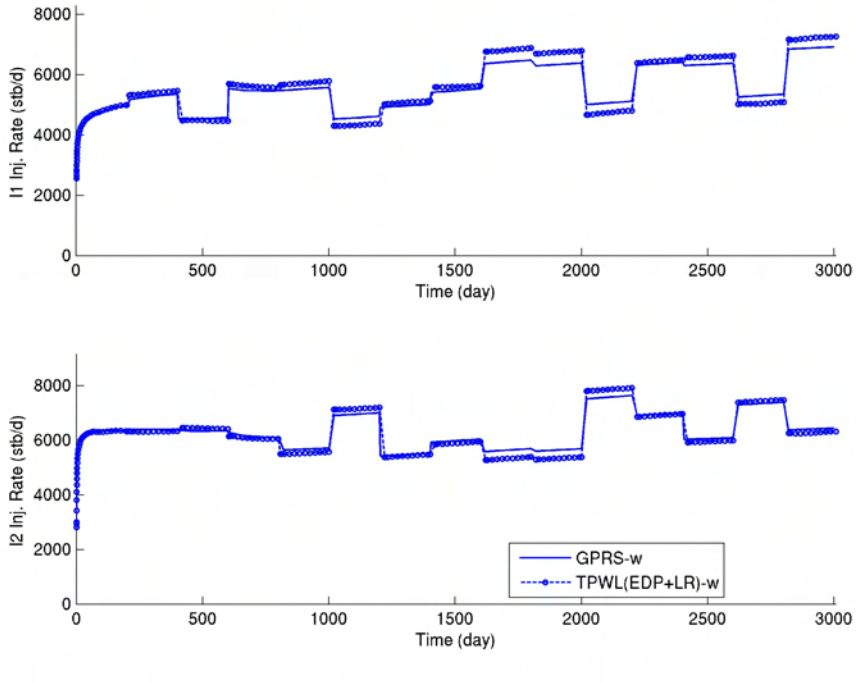


Figure 5.11: Model 3 injection rates for $\alpha = 0.5$ using TPWL with EDP basis and local resolution ($l_p = 90$, $l_s = 90$, $n_{LR} = 72$).

Table 5.2: Errors for TPWL(OB+LR) and TPWL(EDP+LR) for Model 3.

Method	$\alpha = 0.1$	0.3	0.5	0.7	0.9
TPWL(OB+LR), E_o	0.0044	0.0132	0.0222	0.0308	0.0398
TPWL(OB+LR), E_w	0.0063	0.0181	0.0298	0.0412	0.0526
TPWL(OB+LR), E_i	0.0064	0.0192	0.0326	0.0457	0.0595
TPWL(EDP+LR), E_o	0.0033	0.0098	0.0164	0.0229	0.0295
TPWL(EDP+LR), E_w	0.0043	0.0128	0.0217	0.0308	0.0403
TPWL(EDP+LR), E_i	0.0057	0.0173	0.0291	0.0401	0.0530

Chapter 6

Use of TPWL for Production Optimization

In this section we use TPWL as a surrogate within a generalized pattern search optimization algorithm. The optimization targets the maximization of net present value for oil production under water injection. We first discuss the generalized pattern search algorithm used for optimization, as well as our treatment of nonlinear constraints. We then describe the use of TPWL for production optimization, including the retraining of the model as the optimization proceeds. We next present optimization results for two example cases. We note that TPWL was used previously as a surrogate in gradient-based optimizations (with gradients computed numerically) in [11, 26]. That work did not consider systematic retraining or nonlinear constraints, both of which are applied here.

6.1 Direct Search Optimization with TPWL

Surrogate modeling is widely used for simulation-based optimization when the full-order (high-fidelity) model is computationally expensive to evaluate. A surrogate model should be computationally inexpensive and at least locally accurate. TPWL appears to be well suited for use as a surrogate as it is able to provide a reasonable approximation of the true solution within a reasonably sized neighborhood around

the training case.

The direct search method used here is generalized pattern search (GPS). GPS computes a sequence of points that approach an optimal point. The algorithm applies polling, which entails the evaluation of solutions defined by a stencil (aligned with the coordinates) in the search space. The central point of the stencil is the current (best) solution. If an improvement in the cost function is found, the stencil is shifted such that it is centered on the improved point. If an improved solution is not found, the stencil size is decreased. See [35] for more detail on GPS and [36] for application of GPS and related procedures to oil production optimization problems.

In Case 2 below we include nonlinear constraints in the optimization. To handle these constraints, GPS with an incremental penalty function is used. In this method, a modified objective function, which is a weighted combination of the original objective function and a penalty term that quantifies the violation of the constraint, is defined. The weighting of the penalty term is increased incrementally (a sequence of subproblems is solved) until an optimized solution satisfying the constraint is achieved.

Our approach for incorporating TPWL into GPS is depicted in Figure 6.1. We start by performing a training simulation with well BHPs defined by the initial guess. The states and Jacobian matrices are saved and the stabilized TPWL model is constructed using the basis optimization procedure described earlier. Then, the GPS optimization is started using the TPWL surrogate for function evaluations. After a specified number of function evaluations are performed, GPS is paused and a training simulation is run at the current best point (the specified number of function evaluations can vary during the course of the optimization). TPWL is then retrained at this point and GPS is resumed. It occasionally happens that, upon retraining, the objective function of the current point, evaluated using the full-order model, is suboptimal relative to that of the previous full-order solution. This inconsistency can occur when the TPWL solution loses accuracy because it is too far from the most recent training case. When this problem is detected, we restart the search from the previous retraining point and reduce the number of function evaluations until the next retraining. The size of the GPS mesh may also be reduced. We note that it should be possible to incorporate more sophisticated criteria, possibly based on mass balance

errors in the TPWL model (which are straightforward to compute), for retraining. Such procedures will be considered in future work.

6.2 Optimization Results

6.2.1 Production Optimization: Case 1

We first optimize a small problem to enable comparison of results using the surrogate procedure to those using the full-order reference solution. The reservoir model for this case, referred to as Model 4, comprises the first four layers of Model 1 (Model 1 is shown in Figure 2.1). Model 4 contains 4800 grid blocks. The rock and fluid properties and well locations are the same as for Model 1 in Section 4.1. In this case, we optimize the production well BHPs to maximize net present value (NPV) over five years (1800 days) of production. The BHP of each well is changed every 200 days. Thus there are nine control variables for each producer, giving a total of 36 control variables. Injection well BHPs are set to 6000 psi for the entire simulation. The oil price is specified to be \$80/bbl while the cost of water produced and injected are \$36/bbl and \$18/bbl, respectively. Water prices are set to be artificially high to limit the use of water. The bounds for the production well BHPs are 1000 psi and 3000 psi. Initially, the BHPs for the four production wells are set to 1500 psi for the entire production period.

The TPWL model here used basis optimization with basis switching. Typical TPWL parameter values were $l_p = 65, l_s = 75$ and $n_{LR} = 50$. The evolution of NPV with the number of simulations is shown in Figure 6.2 and summarized in Table 6.1. In the figure, the red curve represents the optimization results using the full-order simulation model while the blue curve presents results using the TPWL model. The circles indicate points where the TPWL surrogate model was retrained. It is evident that, using only 15 full-order training simulations, the TPWL guided optimization provides essentially the same result as was achieved using the full-order simulations. The total TPWL overhead in this case requires the equivalent of about another 10 training simulations. Thus the overall speedup for this example is about a factor of

100. Because our TPWL implementation is currently in Matlab, we expect that these speedups could be further improved through a careful C++ implementation.

6.2.2 Production Optimization: Case 2

We now consider a larger reservoir model, and therefore do not present a comparison with full-order optimization results. The reservoir model in this case is the full Model 1, shown in Figure 2.1, which contains 20,400 grid blocks. In this example we optimize the BHPs for the four production wells and the two injection wells. The objective function is again NPV over five years of production. The well controls are again changed every 200 days, so there are a total of 54 control variables. The bounds for injector BHPs are 5500 psi and 7500 psi, while those for producers are again 1000 psi and 3000 psi. A nonlinear constraint is also imposed on the optimization. This constraint requires that the water cut (fraction of water in the produced fluid) for all producers is less than 50% at all times.

The TPWL model again uses basis optimization and local resolution for this case. The optimization results for this case are shown in Figure 6.3. The blue curve indicates NPV and the green curve shows the constraint violation. At early iterations, NPV is improved and the water cut constraint is satisfied, but at later iterations the constraint is violated. As the penalty weight increases, the constraint violation decreases. Finally, after nearly 4000 function evaluations, a feasible solution is obtained with an NPV that is 34% greater than that of the initial guess. The circles and stars in Figure 6.3 indicate where retraining of the TPWL model is performed. A total of 12 full-order training simulations are required. The total TPWL overhead for this case is equivalent to around 8-10 full-order simulations. Thus the speedup relative to running GPS with the full-order model is around a factor of 200. We note that, because several days of computation would be required, we have not performed optimizations using the full-order model for this case.

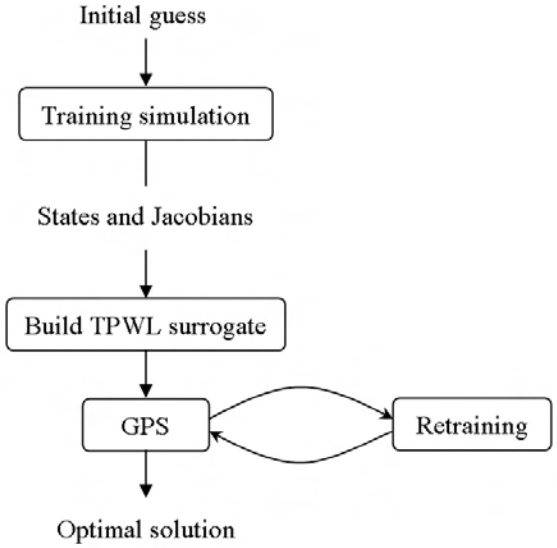


Figure 6.1: Flowchart for generalized pattern search with TPWL.

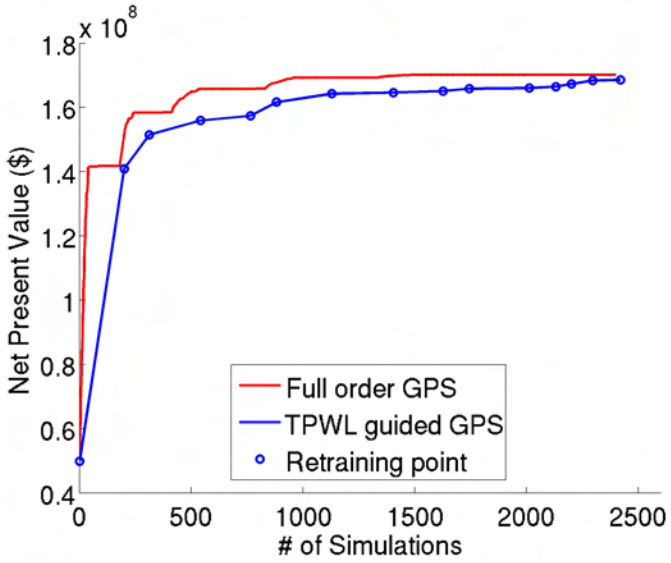


Figure 6.2: NPV evolution during optimization (Case 1).

Table 6.1: Optimization results for Case 1.

Method	Initial NPV ($\$10^6$)	Final NPV($\$10^6$)	# of full simulations
Full-order GPS	49.9	170.1	2500
TPWL guided GPS	49.9	169.0	15

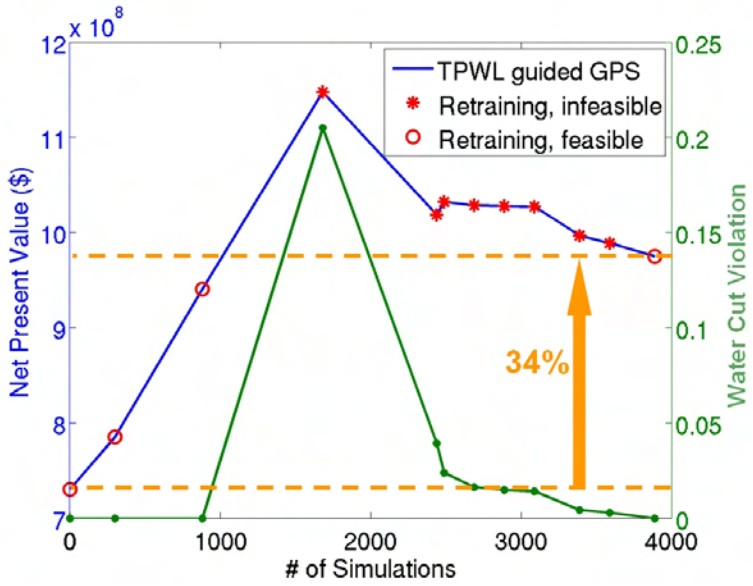


Figure 6.3: Evolution of NPV and constraint violation during optimization (Case 2).

Chapter 7

Summary and Future Work

7.1 Summary

In this thesis, we enhanced the performance of a recently developed approach for reduced-order modeling – the trajectory piecewise linearization (TPWL) procedure. In our TPWL implementation, reservoir equations are linearized around previously simulated training runs and the high-dimension state space is projected into a low-dimension space using proper orthogonal decomposition (POD). Both the accuracy and stability characteristics of this method were investigated and extended in this work.

To improve TPWL accuracy, a local resolution method was proposed to preserve high resolution in well blocks and important flow regions. The idea is to avoid dimension reduction in these regions by separating them from the rest of the model during the reduction. Results show that this method outperforms the basic TPWL when using the same number of reduced variables. The method also provides flexibility on the choice of the number of reduced variables to include in the TPWL representation.

An example with a significant density difference between phases was shown to lead to instability in the TPWL model. Stability analysis revealed that the stability of the TPWL method is related to the spectral radius of the amplification matrix. Two different stabilization methods were proposed. In the optimized basis (OB) method, we seek to minimize the spectral radius of the amplification matrix by choosing the

optimum number of reduced variables. The OB method does not guarantee stability, but it does improve TPWL stability and performance significantly. The equal density projection (EDP) method stabilizes cases with density differences by using the basis from cases without density differences. The EDP method requires one extra full-order simulation for stabilization. This method appears to be slightly more accurate than the OB approach. Both methods were further tested on two reservoir problems of practical size with significant density differences. Results showed that both methods are able to provide stabilized and accurate solutions for challenging cases.

The use of TPWL within a direct search optimization framework was also demonstrated. TPWL was used as a surrogate in the generalized pattern search method. Retraining is applied to update the TPWL model as the optimization proceeds. Optimization results for two example cases illustrated the applicability of TPWL as a surrogate within the generalized pattern search algorithm. Speedups of about 100 to 200 relative to optimizations using the full-order model were achieved.

7.2 Future Work

One direction for future work is to investigate stabilization methods that not only guarantee stability but also provide reasonable accuracy with low computational cost. Several possible candidates have been suggested. One approach is to construct a left projection matrix based on Lyapunov stability theory to guarantee stability, as is done in [14], and to then build accuracy back into the scheme in some way. So far guaranteed stability has been achieved, but accuracy is still an issue.

For optimization applications, future work should be directed toward formalizing the determination of when retraining should be performed. It would also be of interest to investigate whether it is more advantageous to have multiple training runs that cover the whole space rather than perform retraining along the search path.

The use of TPWL for history matching is also a promising direction for future work. As shown in the Appendix, TPWL is able to provide a reasonable approximation to the true solution even when the geology of the test case is considerably different from that of the training simulation. This feature can be utilized to provide

a search direction for minimizing the mismatch between simulated and actual data by adjusting the geological model. This capability could find use in history matching problems.

Appendix A

TPWL for History Matching

A.1 Introduction

History matching entails the updating of the geological model using new data, e.g., production data or time-lapse seismic data. It is often formulated as an optimization process in which the mismatch between observed and simulated data is minimized by modifying the geological model. In this Appendix, we present a preliminary approach for modeling the impact of perturbation of the geological model on the flow response using a TPWL formulation. This could eventually represent a key component of a history matching procedure based on TPWL.

A.2 TPWL Representation with Geological Variation

The TPWL representation for history matching can be developed in analogy to the representation for the production optimization problem. In production optimization problems, the control parameters are well controls, e.g., well BHPs. For history matching, the "controls" are geological parameters such as porosity, permeability or transmissibility. Therefore, just as we linearize the governing equation around training well BHPs for problems where well BHPs are to be varied, in the history

matching problem we linearize the equation around training geological parameters. Thus, in analogy to Eq. 2.3, for this case we have

$$\mathbf{g}^{n+1} = \mathbf{g}^{i+1} + \frac{\partial \mathbf{g}^{i+1}}{\partial \mathbf{x}^{i+1}} \left(\mathbf{x}^{n+1} - \mathbf{x}^{i+1}\right) + \frac{\partial \mathbf{g}^{i+1}}{\partial \mathbf{x}^{i}} \left(\mathbf{x}^{n} - \mathbf{x}^{i}\right) + \frac{\partial \mathbf{g}^{i+1}}{\partial \mathbf{m}_0} \left(\mathbf{m} - \mathbf{m}_0\right). \quad \text{(A.1)}$$

Here, $\mathbf{g}^{n+1} = \mathbf{g}(\mathbf{x}^{n+1}, \mathbf{x}^{n}, \mathbf{m})$ and $\mathbf{g}^{i+1} = \mathbf{g}(\mathbf{x}^{i+1}, \mathbf{x}^{i}, \mathbf{m}_0)$, where $\mathbf{m}_0$ is the parameter set for the training geology model and $\mathbf{m}$ is the parameter set for the test geology model. Following the procedure used to develop Eq. 2.9 in Chapter 2, for this case we have

$$\mathbf{z}^{n+1} = \mathbf{z}^{i+1} - \left(\mathbf{J}_r^{i+1}\right)^{-1} \left[\left(\frac{\partial \mathbf{A}^{i+1}}{\partial \mathbf{x}^{i}} \right)_r \left(\mathbf{z}^{n} - \mathbf{z}^{i}\right) + \left(\frac{\partial \mathbf{g}^{i+1}}{\partial \mathbf{m}_0^{i+1}} \right)_r \left(\mathbf{m} - \mathbf{m}_0\right) \right], \quad \text{(A.2)}$$

where

$$\left(\frac{\partial \mathbf{g}^{i+1}}{\partial \mathbf{m}_0^{i+1}} \right)_r = \mathbf{\Phi}^T \frac{\partial \mathbf{g}^{i+1}}{\partial \mathbf{m}_0^{i+1}}. \quad \text{(A.3)}$$

As can be seen in the equations above, an extra term, $\partial \mathbf{g}^{i+1}/\partial \mathbf{m}_0^{i+1}$, appears which is not in our earlier formulation. This term represents the derivative of the residual to the geological parameters. The specific form of this term depends on how the geological model is represented. GPRS has been modified to output this term in the required form.

The geological parameters $\mathbf{m}$ can be taken as permeability or transmissibility (we assume for now that porosity is not modified so it is not included in $\mathbf{m}$). The residual $\mathbf{g}$ of the governing equation is nonlinear in permeability $\mathbf{k}$ but linear in transmissibility $\mathbf{T}$. Therefore, by using $\mathbf{T}$ to represent $\mathbf{m}$ instead of $\mathbf{k}$, we can eliminate one of the second-order error terms, $\partial^2 \mathbf{g}/\partial \mathbf{m}^2$. Eq. A.2 is now written as

$$\mathbf{z}^{n+1} = \mathbf{z}^{i+1} - \left(\mathbf{J}_r^{i+1}\right)^{-1} \left[\left(\frac{\partial \mathbf{A}^{i+1}}{\partial \mathbf{x}^{i}} \right)_r \left(\mathbf{z}^{n} - \mathbf{z}^{i}\right) + \left(\frac{\partial \mathbf{g}^{i+1}}{\partial \mathbf{T}_0^{i+1}} \right)_r \left(\mathbf{T} - \mathbf{T}_0\right) \right]. \quad \text{(A.4)}$$

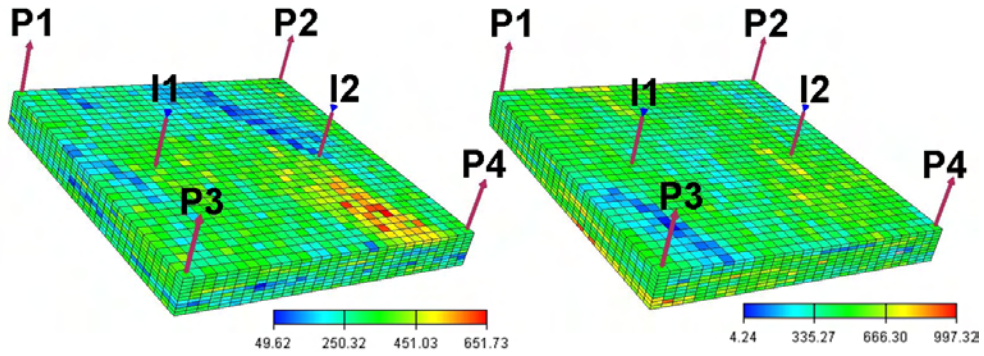


Figure A.1: Left: training geology; Right: target geology. Permeability in x-direction (in mD) is shown.

In Eq. A.4, $\partial \mathbf{g}^{i+1}/\partial \mathbf{T}_0^{i+1}$ is the derivative of the residual with respect to transmissibility and is output from the full-order simulator GPRS during the training run. Both $\mathbf{T}$ and $\partial \mathbf{g}^{i+1}/\partial \mathbf{T}_0^{i+1}$ are in high-dimension space. It may be advantageous to reduce their dimension, which will reduce the number of parameters needed for the history matching. In this work, however, we simply use the full transmissibility without reduction.

A.3 Simulation Results

The methodology based on Eq. A.4 is now tested using synthetic reservoir models generated by sequential Gaussian simulation in SGEMS [37]. The permeability distributions for the training and target geologies are shown in Figure A.1. The variogram model for both geologies is the same; the main direction of permeability correlation is 20^o from the y-direction. For the training geology the mean permeability is 320 mD while the variance is 80 mD. For the target geology the mean permeability is 480 mD and the variance is 120 mD. As can be seen in Figure A.1, the distributions are quite different for these two models. The well locations (evident in Figure A.1) and well BHP specifications are the same for the two models.

Consistent with our earlier approach, we designate $\alpha = 0$ for the training geology and $\alpha = 1$ for the target geology. Test cases are specified using a weighted combination of the two, as in Eq. 2.11. Figures A.2 to A.4 present results for $\alpha = 0.5$. The black

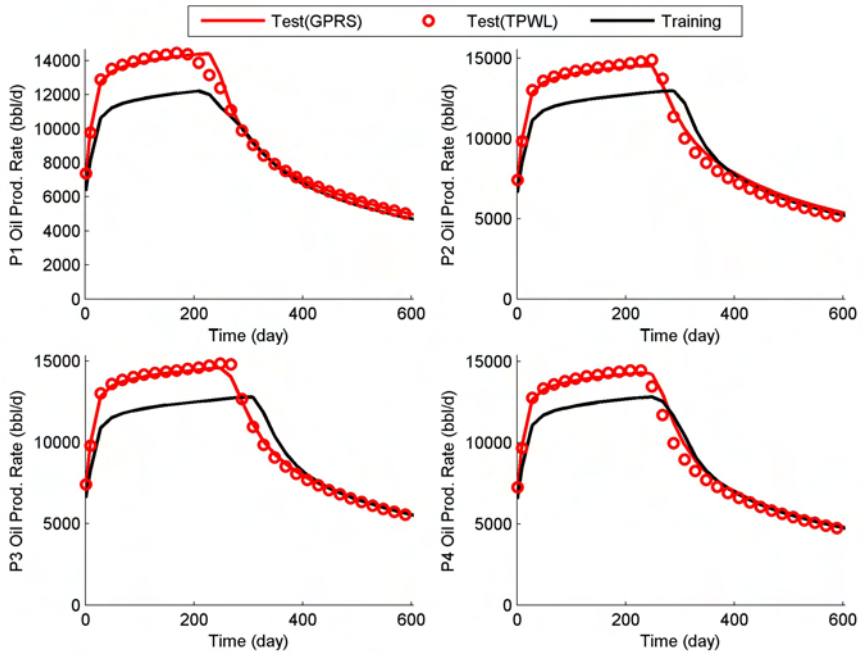


Figure A.2: Oil production rates for $\alpha = 0.5$.

line is the full-order GPRS solution for the training case, about which we linearize. The red line is the full-order solution for the test case we wish to match. It is evident that the training and test solutions differ from each other. The TPWL test results (red circles) are in reasonable agreement with the full-order results, particularly for oil production rates. This indicates that the first-order correction provided by TPWL is able to capture a large part of the difference between the training and test results. This feature is important for history matching as it can be used to identify the search direction for improvement in the objective function.

As mentioned above, in order to use this method for history matching, parameterization of the geology will also be needed to reduce the number of optimization parameters. This can be accomplished, for example, through use of principal component analysis (PCA), where $\mathbf{\Phi}_{geo}$ is the basis matrix of the geology and ξ is the set

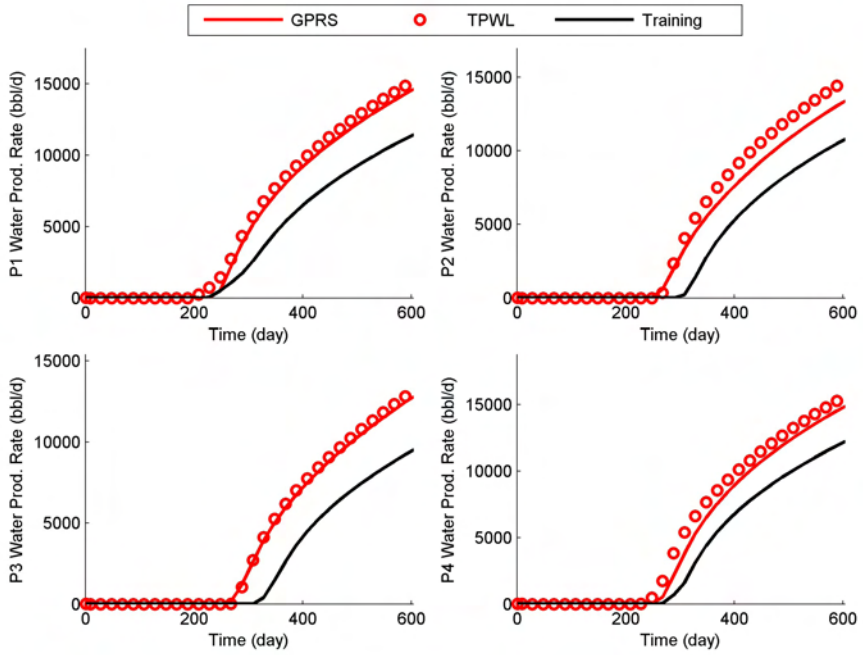


Figure A.3: Water production rates for $\alpha = 0.5$.

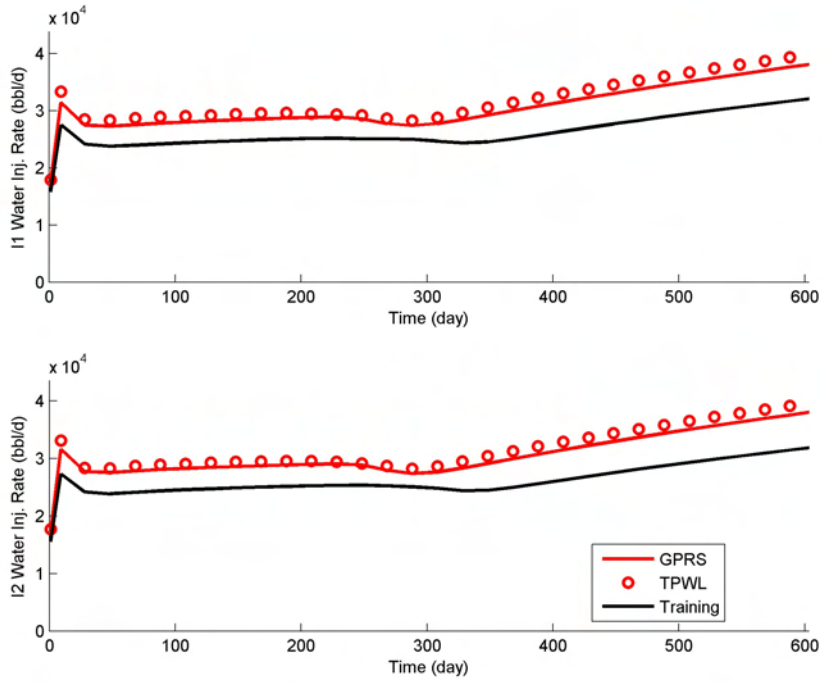


Figure A.4: Water injection rates for $\alpha = 0.5$.

of reduced geological parameters:

$$\mathbf{m} = \mathbf{\Phi}_{geo}\xi. \tag{A.5}$$

Nonlinear parameterizations, such as those based on kernel principal component analysis (KPCA), could also be applied. Such approaches can be used to approximately represent multipoint geostatistical models. See [38] for further discussion.

In future work, we plan to further develop the use of TPWL for history matching and to investigate the use of geological parameterizations within this context.

Bibliography

[1] P. T. M. Vermeulen, A. W. Heemink, C. B. M. T. Stroet, Reduced models for linear groundwater flow models using empirical orthogonal functions, Advances in Water Resources 27 (2004) 57–69.

[2] J. F. M. van Doren, R. Markovinović, J. D. Jansen, Reduced-order optimal control of water flooding using proper orthogonal decomposition, Computational Geosciences 10 (2006) 137–158.

[3] T. Heijn, R. Markovinović, J. D. Jansen, Generation of low-order reservoir models using system-theoretical concepts, SPE Journal 9 (2) (2004) 202–218.

[4] M. A. Cardoso, L. J. Durlofsky, P. Sarma, Development and application of reduced-order modeling procedures for subsurface flow simulation, International Journal for Numerical Methods in Engineering 77 (9) (2009) 1322 – 1350.

[5] M. J. Rewienski, A trajectory piecewise-linear approach to model order reduction of nonlinear dynamical systems, Ph.D. thesis, Massachusetts Institute of Technology (2003).

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

[7] D. Gratton, K. Willcox, Reduced-order, trajectory piecewise-linear models for nonlinear computational fluid dynamics, in: 34^{th} AIAA Fluid Dynamics Conference and Exhibit, Portland, Oregon, USA, 2004.

[8] Y. J. Yang, K. Y. Shen, Nonlinear heat-transfer macromodeling for MEMS thermal devices, Journal of Micromechanics and Microengineering 15 (2) (2005) 408–418.

[9] D. Vasilyev, M. Rewienski, J. White, Macromodel generation for BioMEMS components using a stabilized balanced truncation plus trajectory piecewise-linear approach, IEEE Transactions on Computer-aided Design of Integrated Circuits and Systems 25 (2) (2006) 285–293.

[10] B. N. Bond, L. Daniel, A piecewise-linear moment-matching approach to parameterized model-order reduction for highly nonlinear systems, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 26 (12) (2007) 2116–2129.

[11] M. A. Cardoso, L. J. Durlofsky, Linearized reduced-order models for subsurface flow simulation, Journal of Computational Physics 229 (3) (2010) 681–700.

[12] P. Astrid, A. Verhoeven, Application of least squares MPE technique in the reduced order modeling of electrical circuits, in: 17th International Symposium on Mathematical Theory of Networks and Systems, Kyoto, Japan, 2006.

[13] B. N. Bond, L. Daniel, Stabilizing schemes for piecewise-linear reduced order models via projection and weighting functions, in: Proceedings of the IEEE/ACM International Conference on Computer-Aided Design, San Jose, California, 2007, pp. 860–867.

[14] B. N. Bond, L. Daniel, Guaranteed stable projection-based model reduction for indefinite and unstable linear systems, in: Proceedings of the IEEE/ACM International Conference on Computer-Aided Design, San Jose, California, 2008, pp. 728–735.

[15] B. N. Bond, L. Daniel, Stable reduced models for nonlinear descriptor systems through piecewise-linear approximation and projection, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 28 (10) (2009) 1467–1480.

[16] T. Bui-Thanh, M. Damodaran, K. Willcox, Aerodynamic data reconstruction and inverse design using proper orthogonal decomposition, AIAA Journal 42 (8) (2004) 1505–1516.

[17] T. Bui-Thanh, K. Willcox, O. Ghattas, B. van Bloemen Waanders, Goal-oriented, model-constrained optimization for reduction of large-scale systems, Journal of Computational Physics 224 (2) (2007) 880–896.

[18] K. Aziz, A. Settari, Fundamentals of Reservoir Simulation, Elsevier Applied Science Publishers, 1986.

[19] V. Adamjan, D. Arov, M. Krein, Analytic properties of Schmidt pairs for a Hankel operator and the generalized Schur-Takagi problem, Math. USSR Sbornik 15 (1971) 31–73.

[20] M. Bettayeb, L. Silverman, M. Safonov, Optimal approximation of continuous-time system, in: Proceedings of the IEEE Conference on Decision and Control, Vol. 1, Albuquerque, NM, 1980.

[21] B. Moore, Principal component analysis in linear systems: Controllability, observability, and model reduction, IEEE Transactions on Automatic Control 26 (1981) 17–31.

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

[23] J. L. Lumley, Atmospheric turbulence and radio wave propagation, Journal of Computational Chemistry 23 (13) (1967) 1236–1243.

[24] R. Markovinović, J. D. Jansen, Accelerating iterative solution methods using reduced-order models as solution predictors, International Journal for Numerical Methods in Engineering 68 (5) (2006) 525–541.

[25] K. Pearson, On lines and planes of closest fit to points in space, Philosophical Magazine 2 (1901) 559–572.

[26] M. A. Cardoso, L. J. Durlofsky, Use of reduced-order modeling procedures for production optimization, to appear in SPE Journal, available online.

[27] G. Berkooz, E. Z. Titi, Galerkin projections and the proper orthogonal decomposition for equivariant equations, Physics Letter A 174(1-2) (1993) 94–102.

[28] S. A. Castro, A probabilistic approach to jointly integrate 3D/4D seismic production data and geological information for building reservoir models, Ph.D. thesis, Stanford University (2007).

[29] H. Cao, Development of techniques for general purpose simulators, Ph.D. thesis, Stanford University (2002).

[30] Y. Jiang, A flexible computational framework for efficient integrated simulation of advanced wells and unstructured reservoir models, Ph.D. thesis, Stanford University (2007).

[31] K. V. Mardia, J. T. Kent, J. M. Bibby, Multivariate Analysis, Academic Press, London, 1979.

[32] D. M. Hawkins, The problem of overfitting, Journal of Chemical Information and Computer Science 44(1) (2004) 1–12.

[33] S. S. Vempala, The Random Projection Method, American Mathematical Society, 2004.

[34] M. A. Christie, M. J. Blunt, Tenth SPE comparative solution project: A comparison of upscaling techniques, SPE Reservoir Evaluation & Engineering (4) (2001) 308 – 317.

[35] T. G. Kolda, R. M. Lewis, V. Torczon, Optimization by direct search: New perspectives on some classical and modern methods, SIAM Review 45(3) (2003) 385–482.

[36] D. Echeverría Ciaurri, O. J. Isebor, L. J. Durlofsky, Application of derivative-free methodologies to generally constrained oil production optimization problems, in:

International Conference on Computational Science (ICCS), Amsterdam, The Netherlands, 2010.

[37] N. Remy, A. Boucher, J. Wu, Applied geostatistics with SGeMS: A user's guide, Cambridge University Press, 2008.

[38] P. Sarma, L. J. Durlofsky, K. Aziz, Kernel principal component analysis for efficient, differentiable parameterization of multipoint geostatistics, Mathematical Geosciences 40(1) (2008) 3–32.