COPPE
Instituto Alberto Luiz Coimbra de
Pós-Graduação e Pesquisa de Engenharia
UFRJ

OPTIMIZATION AND CONTROL OF RESERVOIR MODELS USING SYSTEM IDENTIFICATION AND MACHINE LEARNING TOOLS

Luis Kin Miyatake

Dissertação de Mestrado apresentada ao Programa de Pós-graduação em Engenharia Elétrica, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Mestre em Engenharia Elétrica.

Orientador: Amit Bhaya

Rio de Janeiro
Julho de 2019

OPTIMIZATION AND CONTROL OF RESERVOIR MODELS USING SYSTEM IDENTIFICATION AND MACHINE LEARNING TOOLS

Luis Kin Miyatake

DISSERTAÇÃO SUBMETIDA AO CORPO DOCENTE DO INSTITUTO ALBERTO LUIZ COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE ENGENHARIA (COPPE) DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIAS EM ENGENHARIA ELÉTRICA.

Examinada por:

Prof. Amit Bhaya, Ph. D.

Prof. Luis Antonio Aguirre, Ph. D.

Prof. Ramon Romankevicius Costa, D. Sc.

Prof. Alexandre Anozé Emerick, Ph. D.

RIO DE JANEIRO, RJ – BRASIL
JULHO DE 2019

Miyatake, Luis Kin

Optimization and Control of Reservoir Models using System Identification and Machine Learning Tools/Luis Kin Miyatake. – Rio de Janeiro: UFRJ/COPPE, 2019.

XVII, 75 p.: il.; $29,7$cm.

Orientador: Amit Bhaya

Dissertação (mestrado) – UFRJ/COPPE/Programa de Engenharia Elétrica, 2019.

Referências Bibliográficas: p. 73 – 75.

1. System Identification. 2. Optimization. 3. Machine Learning. I. Bhaya, Amit. II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia Elétrica. III. Título.

Acknowledgement

Gostaria de agradecer, inicialmente, ao professor Amit Bhaya, meu orientador do programa de engenharia elétrica da Coppe/UFRJ. Obrigado pelas recomendações de matérias, pelas sugestões elegantes e por toda atenção concedida durante estes dois anos.

Também sou grato aos profissionais do Petrobras/Cenpes pelas sugestões de trabalho. Em especial, aos engenheiros Mario Campos, Emerick, Manuel Fragoso e Alex Teixeira, bem como ao gerente Ziglio pelo incentivo ao longo desta jornada.

À minha mãe, que sempre depositou grande confiança no meu esforço e força de vontade. À Isis, pela compreensão e bom humor. A Deus e a todos aqueles que me ajudaram até aqui.

Resumo da Dissertação apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências (M.Sc.)

OTIMIZAÇÃO E CONTROLE DE MODELOS DE RESERVATÓRIOS USANDO TÉCNICAS DE IDENTIFICAÇÃO DE SISTEMAS E APRENDIZADO DE MÁQUINA

Luis Kin Miyatake

Julho/2019

Orientador: Amit Bhaya

Programa: Engenharia Elétrica

Este trabalho desenvolve uma metodologia, usando conceitos de identificação de sistemas dinâmicos, para criar modelos substitutos (conhecidos como *proxy*) de reservatórios de óleo e gás considerando-se variáveis controladas, tais como vazões de líquido e injetadas. Os principais objetivos são previsão e otimização da produção.

Os métodos clássicos de ajuste de histórico consideram o ajuste de parâmetros de um modelo de simulação de fluxo em meios porosos. Em contraste, essa dissertação propõe avaliar o uso de modelos do tipo *proxy*, com dois enfoques diferentes: o primeiro é baseado puramente em entrada e saída, ao passo que o segundo leva em conta o espaço de estados de uma simulação numérica, ambos usando dados provenientes da simulação numérica de um modelo de reservatórios.

Seguindo o princípio da parcimônia, representações mais simples, tais como ARX e ARMAX, são avaliadas inicialmente para os modelos entrada-saída. Para modelos baseados em estados, realiza-se redução de dimensionalidade, através do método conhecido como POD (proper orthogonal decomposition). As matrizes de um modelo proxy linear são identificadas nos estados de dimensão reduzida, o que nos permite formular um problema de otimização, cuja função objetivo é maximizar uma função econômica VPL (valor presente líquido), como uma sequência de problemas do tipo programação linear, dentro de um arcabouço de um método de otimização baseado em região de confiança.

Algumas contribuições, mostradas ao longo dessa dissertação, incluem um método expedito para avaliação de incertezas, análise da adaptação dos coeficientes do filtro RLS (mínimos quadrados recursivos) em termos físicos para o problema, bem como insights sobre seleção de modelos e incorporação de conhecimento *a priori*.

Abstract of Dissertation presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Master of Science (M.Sc.)

OPTIMIZATION AND CONTROL OF RESERVOIR MODELS USING SYSTEM IDENTIFICATION AND MACHINE LEARNING TOOLS

Luis Kin Miyatake

July/2019

Advisor: Amit Bhaya

Department: Electrical Engineering

This dissertation develops a methodology to identify a dynamical system modeling an oil and gas reservoir, subject to production controls such as water injection rate and liquid production rate. The overall objectives are to improve production forecasts and decision making processes regarding the development of the field, such as future control strategies and "what-if" analyses considering different scenarios.

The classical history matching approach uses numerical simulation and tuning of geological parameters. In contrast, this dissertation proposes the use of a system identification approach to build two proxy models, one based on the input-output approach and the other on a state-space approach, both utilizing data that comes from a simulator used in industry. In accordance with the parsimony principle, simpler polynomial model structures such as ARX and ARMAX are used for the input-output model.

The linear state space model uses states coming from model simulation as its data for identification, and is subjected to model reduction using the proper orthogonal decomposition (POD) method. This linear state space reduced order proxy model is then used to formulate an optimal control problem, solved by transcription into a sequence of linear programs using a trust region algorithm, maximizing Net Present Value, which is an objective function representing the overall economic performance of the production process.

Additional significant contributions, developed in the course of this dissertation, include a fast method for uncertainty estimation, analysis of RLS coefficient adaptation to get physical insights into the correlation between producer and injector wells, as well as insights into model selection and incorporation of prior knowledge.

Contents

List of Figures **ix**

List of Tables **xv**

1 Introduction and Literature Review **1**

2 Discrete-time linear system models for identification: practical examples **7**

2.1 Introduction 7

2.2 Preliminaries on input-output discrete time models for identification . 8

2.2.1 ARX models 8

2.2.2 ARMAX models 8

2.3 Training algorithms for ARX and ARMAX models 9

2.3.1 Training an ARX model 9

2.3.2 ARMAX training procedure 10

2.4 Model Structure Selection 10

2.5 Estimation of Parameter Uncertainty 12

2.5.1 Covariance Matrix Estimation 13

2.5.2 Sampling on the boundary of the uncertainty ellipsoid 14

2.6 Online Learning 15

2.6.1 Recursive Least Squares 15

2.6.2 Practical Remarks about the Prior Knowledge in the RLS filters 18

2.7 Simulator-based identification and validation of reservoir models . . . 18

2.7.1 Choice of Features and Experimental Design 19

2.7.2 Results of ARX and ARMAX modeling 20

3 Proxy States Based Model **37**

3.1 Method of snapshots 37

3.2 Relation between Regularization and PCA 39

3.3 State Matrix Estimation 40

3.3.1 Output Identification 41

3.4 Proxy based Optimization 42

3.5 Results 44
3.5.1 Assessment of Proxy Model 44
3.5.2 Optimization Results 61

4 Conclusions and Future Work 70

Bibliography 73

List of Figures

1.1 Case 1: Input-Output schematics, where WIR (Water injection Rate) is a PRBS excitation and LPR (liquid production rate) is constant. As a result, BHP (bottom hole pressure) is a variable output, as is WC (water cut). 2

1.2 Case 2: Input-Output schematics, where WIR (Water injection Rate) is a PRBS excitation and BHP (bottom hole pressure) is constant. As a result, LPR (liquid production rate) is a variable output, as is WC (water cut). 2

1.3 Case 3: Input-Output schematics, where both WIR (Water injection Rate) and LPR (liquid production rate) are PRBS Excitation. As a result, BHP (bottom hole pressure) is a variable output, as is WC (water cut). 2

1.4 Case 4: Input-Output schematics, where both WIR (Water injection Rate) and BHP (bottom hole pressure) are PRBS Excitation. As a result, LPR (liquid production rate) is a variable output, as is WC (water cut). 2

2.1 Adaptation from MOSTAFA *et al.* [19]. The diagram shows how the data set is split between training and validation data sets. The resulting model g^{-} obtained by model selection procedure is the one which presents the smallest quadratic error in the validation dataset. 11

2.2 Water Injection Rates PRBS Excitation for System Identification. The Liquid Producer Rates are kept constant. 19

2.3 Water saturation map is represented during 14 years. It is worthwhile pointing out the time variant characteristics of the production system, which varies especially when water breakthrough occurs. 20

2.4 Reference log-permeability field, making evident the permeability path connectivity between INJ-01 and PRO-02/PRO-03 and INJ-02 and PRO-04/PRO-05. 21

2.5 Model Structure choice based on the Validation Set for the producer well PRO-01. 21

2.6 Chosen the model structure, training and validation data set are used for training. Residual analysis indicates that the error is a white noise for the producer well PRO-01. 22

2.7 Test set assessment comparing the forecast on the average case (in black) and Test Data Set. In green, 50 different parameter vectors with the same model structure were sampled on the Ellipsoid of Uncertainty considering 3 standard deviation for the producer PRO-01. . 23

2.8 Instead of sampling on the ellipsoid of uncertainty, Monte Carlo simulation is performed. The curves P-10, P-40, P-60 and P-90 are defined with 3000 simulations for the producer well PRO-01. 23

2.9 Step Response: BHP - PRO-01. Both INJ-01 and INJ-02 contribute for BHP increase for the producer well PRO-01 in the step response. . 24

2.10 Model Structure choice based on the Validation Set for the producer well PRO-02. 25

2.11 Chosen the model structure, training and validation data set are used for training. Residual analysis indicates that the error is not very different from a white noise for the producer well PRO-02. 26

2.12 Test set assessment comparing the forecast on the average case (in black) and Test Data Set. In green, 50 different parameter vectors with the same model structure were sampled on the Ellipsoid of Uncertainty considering 3 standard deviation for the producer well PRO-02. It is worthwhile mentioning the scale, which ranges from 0.90 to 0.94, indicating a good agreement between the uncertainty analysis forecast and the test data set. 26

2.13 Instead of sampling on the ellipsoid of uncertainty, Monte Carlo simulation is performed. The curves P-10, P-40, P-60 and P-90 are defined with 3000 simulations for the producer PRO-02. 27

2.14 Step Response for the producer wel PRO-02. INJ-01 has a meaningful contribution for water cut increase in PRO-02, whereas INJ-02 seems to have very little influence. 27

2.15 The graph shows 3 different choices for forgetting factors in the RLS filter. 12 months ahead prediction of water cut for the producer PRO-03 (in blue) is compared with the original time series, which comes from the model simulation. 28

2.16 RLS - Exogenous Parameters Adaptation for the producer PRO-03. Interestingly, we see a pattern change after the water breakthrough, which means the filter adapts quickly and is able to capture the real connectivity among the wells. 29

2.17 12 months ahead prediction of water cut for the producer PRO-04 (in blue) is compared with the original time series, which comes from the model simulation. Bigger adaptation is observed when water breakthrough occurs. 30

2.18 The evolution of the sum of exogenous parameters for each well for the producer PRO-04. Because the standard initial condition was chosen ($\theta = 0, P = 10^3 I$) the filter starts to learn from data when water breakthrough occurs. Until then, the output water cut is zero and there is no information to be learnt from. The parameters successfully indicate connectivity between INJ-02 and PRO-04. 31

2.19 Test set assessment comparing the forecast on the average case (in black) and Test Data Set. In green, 50 different models were sampled on the Ellipsoid of Uncertainty considering 3 standard deviation for the producer PRO-05. 32

2.20 Residual analysis indicates 1-period lag out of ACF bounds, which suggests that error could be modeled with a moving average (MA) component. This is why we attempt to model it evolving from ARX structure to ARMAX. 33

2.21 Test set assessment comparing the forecast on the average case (in black) and Test Data Set. In green, 50 different models were sampled on the Ellipsoid of Uncertainty considering 3 standard deviation for the producer PRO-05 using the ARMAX model. 34

2.22 In fact, ARMAX model presents suitable results in residual analysis, which means choosing $n_c = 1$ successfully models the error. 34

2.23 Comparison between ARX and ARMAX in the test data set. Despite its advantage of modeling the error, ARMAX presents results very similar to ARX in free simulation in the test data set. 35

2.24 Observed Data and Prior Knowledge Data Set. Prior Knowledge may come from different sources, such as a numerical model reservoir simulation or another well with a large history data set with similar characteristics. 35

2.25 12 months ahead prediction of the Field Oil Rate considering Standard RLS filter Initial Condition ($\theta = 0, P = 10^3 I$). 36

2.26 12 months ahead prediction of the Field Oil Rate considering Initial Condition from data assimilation of a prior knowledge data set, as depicted in 2.24. 36

3.1 Pseudo Random Binary Sequence (PRBS) Excitation considering both producer and injector wells. This graph represents LPR (liquid production rate) for all producer wells (from PRO-01 to PRO-06) and WIR (water injection rate) for all injector wells (INJ-01 and INJ-02). 46

3.2 Singular Value Decomposition - from the greatest singular value to the smallest. The dimensionality was reduced from 450 to 19 for water saturation and to 29 for block pressure, which simplifies the identification problem and prevents from overfitting. The energy retained ϵ (which defines the dimension of the POD-basis) is chosen based on the maps reconstruction assessment for both water saturation and block pressure maps, as shown in Figures 3.3 - 3.9. 48

3.3 The first row shows water saturation evolution according to the proxy model, the second according to the simulated model and the last row shows the difference between the two, for the case in which the box constraint allows zero (Δ_u^0) deviation from reference trajectory. 49

3.4 The first row shows water saturation evolution according to the proxy model, the second according to the simulated model and the last row shows the difference between the two, for the case in which the box constraint allows deviation ±100 barrels per day (Δ_u^{100}) from reference trajectory. 50

3.5 The first row shows water saturation evolution according to the proxy model, the second according to the simulated model and the last row shows the difference between the two, for the case in which the box constraint allows deviation ±200 barrels per day (Δ_u^{200}) from reference trajectory. 51

3.6 The first row shows water saturation evolution according to the proxy model, the second according to the simulated model and the last row shows the difference between the two, for the case in which the box constraint allows deviation ±300 barrels per day (Δ_u^{300}) from reference trajectory. 52

3.7 The first row shows block pressure (in psi) evolution according to the proxy model, the second according to the simulated model and the last row shows the difference between the two, for the case in which the box constraint allows zero deviation (Δ_u^0) from reference trajectory. 54

3.8 The first row shows block pressure (in psi) evolution according to the proxy model, the second according to the simulated model and the last row shows the difference between the two, for the case in which the box constraint allows a deviation of ±25 barrels per day (Δ_u^{25}) from reference trajectory. 55

3.9 The first row shows block pressure (in psi) evolution according to the proxy model, the second according to the simulated model and the last row shows the difference between the two, for the case in which the box constraint allows a deviation of ±50 barrels per day (Δ_u^{50}) from reference trajectory. 56

3.10 For all producer wells, the output variable is oil production rate. The figure compares the output of the proxy model with the simulated one. In green, the control (liquid production rate) resulted from the LP problem is shown in the case in which the box constraint allows zero deviation (Δ_u^{0}) from reference trajectory. 57

3.11 For all producer wells, the output variable is oil production rate. The figure compares the output of the proxy model with the simulated one. In green, the control (liquid production rate) resulted from the LP problem is shown in the case in which the box constraint allows a deviation of ±100 barrels per day (Δ_u^{100}) from reference trajectory. . 58

3.12 For all producer wells, the output variable is oil production rate. The figure compares the output of the proxy model with the simulated one. In green, the control (liquid production rate) resulted from the LP problem is shown in the case in which the box constraint allows a deviation of ±200 barrels per day (Δ_u^{200}) from reference trajectory. . 59

3.13 For all producer wells, the output variable is oil production rate. The figure compares the output of the proxy model with the simulated one. In green, the control (liquid production rate) resulted from the LP problem is shown in the case in which the box constraint allows a deviation of ±300 barrels per day (Δ_u^{300}) from reference trajectory. As the allowed deviation is larger, the proxy model tends to present larger deviations from the simulation model. 60

3.14 The first graph shows the rapid initial increase of the NPV due to the optimization algorithm achieving 9.1% overall gain. The second graph compares optimal cumulative oil production (blue curve) and initial solution (black curve). The third graph shows Total Liquid Production Rate attaining its upper bound (Liquid Capacity Constraint) and the fourth graph shows Total Injection Rate. 62

3.15 Figure shows the evolution of the trajectory controls (liquid production rate for producers PRO-01, PRO-02, PRO-03 and PRO-04) for all iterations in the trust region algorithm, from the initial condition (iter 1) to the final solution (iter 15). 64

3.16 Figure shows the evolution of the trajectory control (liquid production rate for producers PRO-05 and PRO-06 and water injection rate for injector INJ-01 and INJ-02) for all iterations in the trust region algorithm, from the initial condition (iter 1) to the final solution (iter 15). 65

3.17 For a liquid capacity constraint of 11350 BPD, the graph indicates the water cut for all producer wells. For most wells, the solution derived from the trust region algorithm results in water breakthrough being delayed, which explains the gains in field cumulative oil production and the net present value (NPV). 66

3.18 In the first graph, the sensitivity analysis shows the NPV Gain evolution when the Total Liquid Capacity constraint is increased. The second graph shows the total liquid production for different total liquid capacities. Interestingly, when the capacity is 14200 BPD, this constraint is not active during the entire period. 68

3.19 For all wells, both producers and injectors, the graph shows the trajectory controls (liquid production rate for producers and water injection rates for injectors) for different total liquid capacity constraints. Notice that the trajectory controls are not intuitively obvious, especially for different scenarios. 69

List of Tables

2.1 Range of perturbation 19

3.1 Liquid Rate Bounds (BPD) 61
3.2 Sensitivity Analysis 67

List of Symbols

$\bar{u}$	Current trajectory control
Δ_u	Box constraint in which the proxy model is valid around $\bar{u}$
ϵ	Percentage of energy retained for dimensionality reduction
λ	Forgetting factor in the RLS algorithm
ρ	Proxy Quality Control Index
$A(q)$	Auto regressive polynomial of ARX model
$B(q)$	Exogenous polynomial of ARX model
$C(q)$	Error polynomial of ARMAX model
E	Uncertainty ellipsoid corresponding to 1 standard deviation
E_{val}	Error calculated in the validation data set
LP	Linear Programming Problem
Q_{max}	Platform liquid capacity constraint
s_r	Slew Rate, a constraint which imposes smoothness in the trajectory control
U_r	Reduced matrix from POD dimensionality reduction procedure
X	Snapshot Matrix
ACF	Autocorrelation Function
ARMAX	Auto Regressive Moving Average Exogenous Model
ARX	Auto Regressive Exogenous Model
BHP	Bottom hole pressure
BPD	Barrels per day

ELS	Extended Least Squares
NPV	Net Present Value
POD	Proper Orthogonal Decomposition
PRBS	Pseudorandom Binary Sequence
WC	Water Cut

Chapter 1

Introduction and Literature Review

Reservoir modeling is concerned with the construction of a computer model of an oil and gas reservoir, with the aim of improving estimation of reserves, production forecasting and decision making processes regarding the development of the field, such as well placement, future control strategies and “what if” analyses considering different scenarios.

A reservoir model consists of grid blocks, which represent the physical space where the reservoir is located, and each grid block has parameters (porosity, permeability and so forth) and states (pressure, water saturation, oil saturation etc). The reservoir model solves a finite difference numerical scheme derived from a partial differential equation, which models the spatiotemporal evolution in the porous media, considering its different phases and compositions.

Typically, an oil and gas field has producer and injector wells. The producers produce liquid (oil and water) and gas, and the injectors inject water, gas or both in alternate cycles. The importance of the injectors is related to the maintenance of the reservoir pressure and oil displacement. If it were not for them, the average pressure in the reservoir would drop, and the oil recovery would worsen.

However, as the injectors start injecting water or gas, the water (or gas) front reaches the producer well. This moment is called water (or gas) breakthrough. Of course, this causes an increase in the water cut ratio (ratio between produced water rate and produced liquid rate), as well as gas oil ratio increase.

In terms of what can actually be measured, the output variables of a reservoir are generally oil, water and gas well rates, as well as the bottom hole pressure (BHP) and compositional contents. These variables, measured in hourly, daily or even monthly frequencies are modeled as outputs of suitable state variables, which evolve according to a dynamical system.

For the sake of identification, it is suitable to excite the reservoir system with

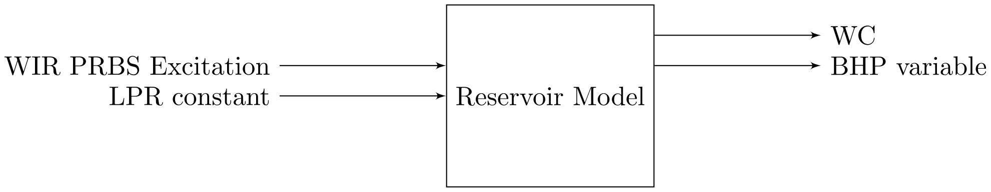

Figure 1.1: Case 1: Input-Output schematics, where WIR (Water injection Rate) is a PRBS excitation and LPR (liquid production rate) is constant. As a result, BHP (bottom hole pressure) is a variable output, as is WC (water cut).

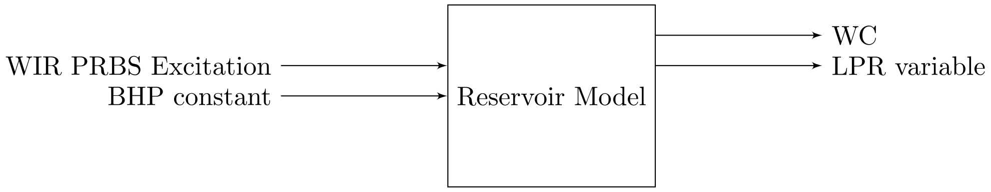

Figure 1.2: Case 2: Input-Output schematics, where WIR (Water injection Rate) is a PRBS excitation and BHP (bottom hole pressure) is constant. As a result, LPR (liquid production rate) is a variable output, as is WC (water cut).

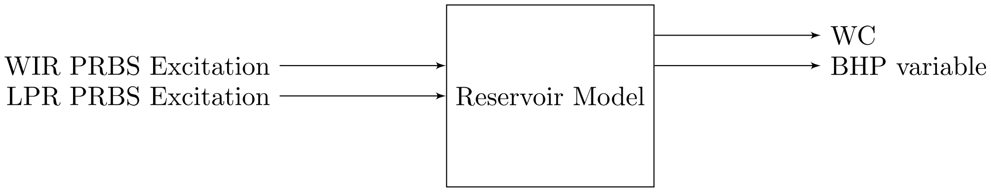

Figure 1.3: Case 3: Input-Output schematics, where both WIR (Water injection Rate) and LPR (liquid production rate) are PRBS Excitation. As a result, BHP (bottom hole pressure) is a variable output, as is WC (water cut).

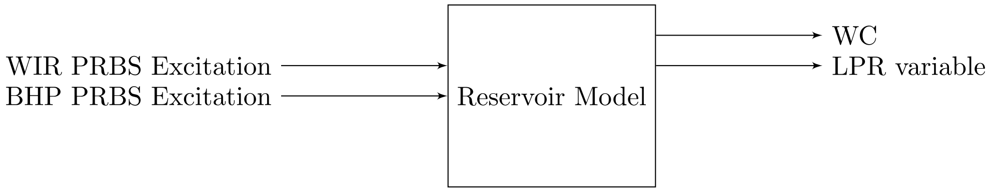

Figure 1.4: Case 4: Input-Output schematics, where both WIR (Water injection Rate) and BHP (bottom hole pressure) are PRBS Excitation. As a result, LPR (liquid production rate) is a variable output, as is WC (water cut).

a PRBS (pseudo random binary sequence) signal. The figures 1.1, 1.2, 1.3 and 1.4 show configurations of possible excitation scenarios, provided that model simulation considers as input variables either flow rate or bottom hole pressure for injector and producer wells.

System identification (SI) is a methodology to build mathematical models of a dynamical system based on measurements of input and output signals. System identification models tend to be simpler than their counterparts based on detailed models using the physics of the underlying phenomena followed by numerical simulation. When a long history of input-output measurements is available, SI models, despite their relative simplicity, can be accurate enough for the purposes of control and optimization (more details in ZHOU *et al.* [1]).

This work uses system identification concepts for reservoir modeling. Two different approaches are presented: the first one, discussed in chapter 2, estimates a transfer function between inputs and outputs, whereas the second one, discussed in chapter 3, tackles the development of a proxy model based on numerical simulation grid data, more likely to represent physical aspects of fluid flow through porous media and the geometry of the reservoir.

Typically, decline curve analysis (DCA), proposed by ARPS [2], and numerical reservoir simulation are the classical methods to forecast reservoir performance. DCA is based on parameter fitting of an empirical equation using measured production data. On the other hand, numerical reservoir simulation provides a mathematical description constrained to physical aspects, such as material and momentum balance.

The major difficulty in building a good numerical reservoir simulator is the fact that a lot of data are required. Moreover, rock and fluid characteristics data tend to present a great deal of uncertainty due to the lack of measurements, which are available only where samples of the rock are collected.

Provided that a numerical model simulation is built, history matching (HM) is the process of adjusting reservoir model parameters such that observed data (bottom hole pressure, water cut, gas oil ratio and so forth) are matched with the values provided by the simulator. Moreover, history matching is understood as a process of diminishing uncertainties as incoming observed data progressively reveal more information about the reservoir system.

Kalman Filter based HM methods are popular in the research community. Many improvements have been made, especially based on the ensemble Kalman Filter (EnKF), proposed in EVENSEN [3]. Other variations of the EnKF have been proposed, such as the Ensemble Smoother (ES) proposed by VAN LEEUWEN & EVENSEN [4] and the Ensemble Smoother with Multiple Data Assimilation (ES-MDA) by EMERICK & REYNOLDS [5].

On the other hand, concerning system identification theory, which disregards to a certain extent physical aspects of numerical model simulation, NEGASH *et al.* [6] and HOURFAR [7] propose a polynomial system identification approach for the waterflooding problem. The great advantage of this approach is that computations using polynomial transfer functions run much faster than numerical simulators, which may take hours, days or even weeks to complete a numerical simulation.

NEGASH *et al.* [6] considers UNISIM-I-M reservoir model, which was excited with a PRBS (pseudo random binary sequence) excitation for the injector rates for each well and the output consisted of the total oil production rate. This paper discusses model structure choice and its validation by using residual analysis and cross validation. Many possible candidates structures are considered, such as Frequency Impulse Response (FIR), Autoregressive Exogenous (ARX), Autoregressive moving average Exogenous (ARMAX), Output Error (OE) and Box-Jenkins (BJ).

HOURFAR [7] proceeds similarly in the case of 10th SPE-Model, performing system identification and assimilating the data by using a recursive-least squares approach with ARX structure. The results are particularly interesting for the RLS filter, in which the ARX parameters adapt every time step considered, approximating the non-linear system by a sequence of linear systems generated by the adaptive parameters. Moreover, HOURFAR [7] proposes a framework regarding data generation and model identification considering the Parsimony Principle, which aims to pick the simplest plausible model structure.

In chapter 2, this work aims to contribute to the application of system identification technique to reservoir modeling by:

- Providing uncertainty estimation, proposing a fast sampling technique based on the uncertainty ellipsoid.

- Giving physical insights about the correlation between producer and injector wells by analyzing coefficient adaptation in the RLS filter. The chosen reservoir case study presents permeability preferential paths among injector and producer wells, in which the connections are clearly defined.

- Discussing a model selection technique and how to evolve from the ARX to ARMAX based on autocorrelation of residuals function.

- Highlighting the importance of prior knowledge for history matching using a polynomial model, and assessing its quality considering 12 month ahead prediction.

- Evaluating the effect of initial conditions, and parameters such as forgetting factor on the performance of the RLS filter.

In the context of reservoir data driven modeling, it is worthwhile mentioning capacitance resistance methods (CRM), which are derived from a physical representation, but much less complex than numerical simulations. This class of methods represents liquid production rate as a function of production controls, such as BHP (bottom hole pressure) and WIR (water injection rate). Oil production rates are modeled empirically by power law fractional flow model (FFM), as described in WEBER [8]. WEBER [8] describes the history matching procedure as an optimization problem under constraints, addressed by CONOPT algorithm modeled in GAMS language.

Another surrogate model based on physical aspects is described by CARDOSO [9]. This work aims to perform model order reduction using TPWL/POD technique, in which POD (proper orthogonal decomposition) provides a function basis that is used for projecting the reservoir states variables into a low-dimensional subspace. Moreover, TPWL (trajectory piecewise linearization) approximates the states from numerical simulation by performing linear expansions around states previously simulated and saving them, as well as their jacobians, in order to speed up the full simulation until its end.

Inspired by these ideas and by AGUIRRE [10], this work performs system identification of a state space model of the reduced dynamical system, described in chapter 3. The advantage of this procedure is the fact that the linear state space model allows the use of an efficient linear programming based procedure, which simplifies the optimization of an associated objective function in the optimal control management problem.

Similarly to the history matching problem, ensemble based optimization is used to handle the optimal control management problem, which aims to maximize, for instance, the net present value of a field by proposing optimal production and injection trajectories controls. LORENTZEN *et al.* [11] proposes the so called EnKF-NPV and CHEN *et al.* [12] evolves it towards the EnOpt (ensemble optimization) method. EnOpt is an ensemble based method, which calculates the gradient approximation based on the ensemble sampled around the current trajectory control during the algorithm procedure, and performs linear search until a stopping criterion is reached.

Chapter 3 proposes to make use of the linear system, derived from a system identification procedure, in combination with a minor adaptation of the trust region optimization framework proposed in CONN *et al.* [13] and FRAGOSO [14], which uses a sequence of linear problems (LP). The solution is addressed sequentially by a LP solver in Python language, described in DIAMOND & BOYD [15], and more details about the slew rate constraint that aims to impose smoothness on the trajectory controls can be found in LÓPEZ [16]. To sum up, the objectives of chapter 3 are:

- To explain why model reduction can contribute to prevention of overfitting.

- To assess the quality of a linear state space for the problem of reservoir modeling and control.

- To solve a sequence of LP problems under a trust region optimization method, taking into account operational constraints.

- To assess the effectiveness of the proposed method in terms of the objective function gain and computational effort.

Chapter 2

Discrete-time linear system models for identification: practical examples

2.1 Introduction

The objective of this chapter is to propose a system identification procedure in order to provide a model possessing auto regressive and exogenous components capable of making predictions. For the purpose of trying to make predictions in reservoir management problem, the main question developed in this chapter is: are linear models good enough? Of course, by the Parsimony Principle (aka Occam's Razor), this is a key question that should be addressed. These questions can be phrased a little more specifically, in the context of this dissertation, as follows.

Even though reservoir dynamics are non linear, can linear models be suitable for forecasting? How fast can data based linear models adapt according to changes in reservoir dynamics, such as water flooding?

Models possessing process linear structures, such as ARX (auto-regressive exogenous) and ARMAX (auto-regressive moving average exogenous) are investigated at first. ARMAX modeling attempts to reduce or even eliminate the biased estimate that ARX models could suffer from.

Another important issue is uncertainty estimation. Reservoir models always have uncertainties in geological parameters, but in a data driven model there are no physical parameters. How can we quantify uncertainties in model forecasts?

Finally, in order to demonstrate validity, the methods proposed in this dissertation are applied to a simple reservoir model, which presents preferential permeability paths between injector and producer wells.

2.2 Preliminaries on input-output discrete time models for identification

2.2.1 ARX models

A Single Input Single Output ARX polynomial transfer function can be written as

$$A(q)y(t) = B(q)u(t - n_k) + e(t),$$

where

$$A(q) = \sum_{k=0}^{n_a} a_k q^{-k} = 1 + a_1 q^{-1} + a_2 q^{-2} + ... + a_{n_a} q^{-n_a}$$

and

$$B(q) = b_1 + b_2 q^{-1} + ... + b_{n_b} q^{-n_b+1},$$

where q^{-1} is the backward operator, $a_0 = 1$ and $e(t)$ is white noise.

For multiple input single output (MISO) models, n_b and n_k are vectors and the ith element of n_b and n_k corresponds to the ith input. For instance, $n_a = 2$, $n_b = [1, 2]$ and $n_k = [0, 3]$ represents

$$y(k) + a_1 y(k-1) + a_2 y(k-2) = b_1 u_1(k-0)$$

$$+c_1 u_2(k-3) + c_2 u_2(k-4) + e(k).$$

A special case is obtained when $n_b = [0, ..., 0]$, which reduces the ARX to an AR model structure. In this case, the exogenous component is not considered.

2.2.2 ARMAX models

ARMAX is similar to ARX with a MA (moving average) component which attempts to model the error. In ARX structure, $e(t)$ is white noise, which may not be true for a given dataset. In this case, a way to improve the model is by trying to model the coloured noise:

$$A(q)y(t) = B(q)u(t - n_k) + C(q)e(t), \tag{2.1}$$

where $c_0 = 1$ and

$$C(q) = \sum_{k=0}^{n_c} c_k q^{-k} = 1 + c_1 q^{-1} + ... + c_{n_c} q^{-n_c}.$$

2.3 Training algorithms for ARX and ARMAX models

This section recapitulates the basic least squares technique used to train (equivalently, identify or fit) ARX and ARMAX models.

2.3.1 Training an ARX model

For a given output $y(k)$ and features $\phi(k) = [y(k-1), y(k-2), ..., y(k-n_a), u(k-1), u(k-2), ..., u(k-n_b)]$, the estimated output can be expressed as

$$\hat{y}(k) = \phi(k)\theta.$$

The error is: $e(k) = y(k) - \hat{y}(k) = y(k) - \phi(k)\theta$.

The vector b and the matrix ψ are defined as:

$$b = \begin{pmatrix} y(k) \\ y(k-1) \\ y(k-2) \\ ... \end{pmatrix}, \psi = \begin{pmatrix} \phi(k) \\ \phi(k-1) \\ \phi(k-2) \\ ... \end{pmatrix}$$

With this definition, a linear model in θ is $\psi\theta = b$, which possibly has no solution, since it is just an approximate representation of reality and also because the data set might be corrupted with noise. Thus it is reasonable to minimize the quadratic error

$$J = (\psi\theta - b)^T(\psi\theta - b)$$

by choice of θ.

If we have more data than parameters, matrix ψ is tall and thin and least squares provides the unique optimal solution, provided that the columns of ψ are linearly independent. On the other hand, if matrix ψ is short and fat, there are many different possible solutions which minimize J.

STRANG [17] shows that for a linear system $\psi\theta = b$, the pseudo inverse solution

$$\hat{\theta} = V\Sigma^{\dagger}U^Tb$$

is the one corresponding to the least squares solution, where the matrices U and V are orthogonal and $\Sigma^{\dagger}$ is the pseudoinverse of the matrix of singular values of ψ. Furthermore, in the short and fat case, pseudo inverse solution has minimal norm two length, which is suitable for problem regularization.

This can be thought of as a one step learning algorithm, which arrives at the optimal solution efficiently and quickly, under the appropriate conditions (linear model with tall full rank ψ), when compared to other learning algorithms for nonlinear models, such as backpropagation in neural networks.

2.3.2 ARMAX training procedure

For an ARMAX model (equation 2.1), parameter estimation is not as straightforward because noise is not a measured variable. A popular method to overcome this is to use the so-called extended least squares (ELS) method, which is the following iterative method (see BILLINGS [18] for further details).

1) Solve as though it was an ARX model and calculate $e = y - \psi\hat{\theta}$.

2) Write the extended features matrix, whose row k is:

$$\psi_k^* = [y(k-1), ..., y(k-n_a), u(k-n_k), ...u(k-n_k-n_b+1), e(k-1), ..., e(k-n_c)]$$

and use the least squares algorithm to calculate θ^* using ψ^* instead of ψ.

3) Calculate $e = y - \psi^*\theta^*$ and go back to step 2 until a stopping criterion is satisfied.

This iterative process usually converges in few iterations. ARMAX sophisticates ARX models by applying a moving average filter to the error signal. It is a means by which coloured noise can be modeled.

The extended feature matrix attempts to make ψ_k^* and $e(k)$ uncorrelated, which is suitable for providing unbiased estimates of $A(q)$ and $B(q)$.

2.4 Model Structure Selection

The motivation is to find a simple structure in the context of black-box modeling, that is, at the same time, complex enough to fit the data and still able to make predictions. In any systems identification or learning approach, we want to find a structure that can, in fact, “learn” from data instead of “memorizing” it.

In the reservoir management problem, dynamics are evolving throughout the time and parameter estimation must necessarily be adaptive. We want to establish a framework capable of choosing a suitable model with adaptive parameters in order to make reliable predictions.

In time series literature, ACF (autocorelation function) and PACF (partial autocorrelation function) are used to select appropriate model structure under the assumption of stationarity. For instance, in moving average (MA) processes, ACF indicates appropriate n_c. In autoregressive (AR) models, PACF can provide auto

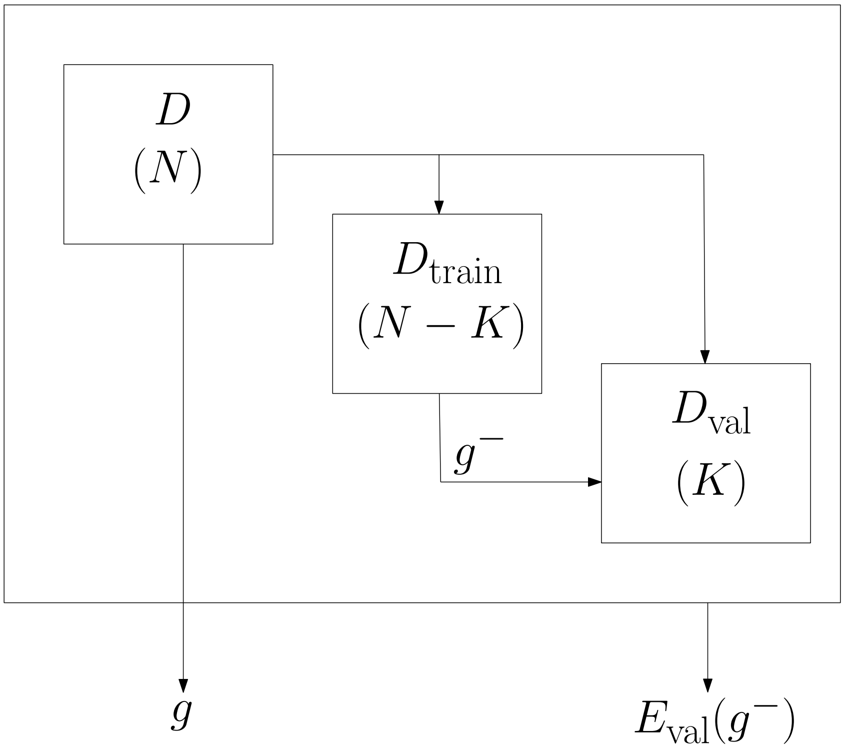

Figure 2.1: Adaptation from MOSTAFA *et al.* [19]. The diagram shows how the data set is split between training and validation data sets. The resulting model g^- obtained by model selection procedure is the one which presents the smallest quadratic error in the validation dataset.

regressive time series delay n_a. Indeed, ACF of error in ARX model can be used to pick an error delay structure n_c from ARX to ARMAX model.

More generally and in a more practical sense, our first approach is to use independent training and validation sets to find a model structure. MOSTAFA *et al.* [19] discusses the importance of using a validation set as a means of selecting the model structure. Generally, an independent validation set could be used to define a model structure and an appropriate regularization level to prevent over fitting. In this chapter, we handle polynomial models and the objective is to find their structures, which means n_a, n_b, n_k and n_c.

A time series with N time steps in monthly frequency is given, and the goal is to use all available data in order to choose an adequate structure. Figure 2.1 depicts training and validation sets. We choose K points to validate the model, in which the error is calculated between the observed data and the free simulation of the polynomial model, and $N - K$ points for training dataset, in which the hypothesis g^- is generated. Moreover, g^- is the result, for a given model structure, of the training procedure considering only the training dataset, which has $N - K$ points.

There is a trade off when it comes to define K. If K is too large, there are few points for training and g^- will not be good enough. On the other hand, if K is too small, the error assessed in validation test ($E_{val}(g^-)$) will not be reliable due to the lack of validation points. In fact, validation set gives an approximation of the error in forecasting and we can use it to pick an adequate set of time delays for ARX and ARMAX models. The procedure for selecting a model structure is as follows:

1) For all possible model structures ($\{n_a, n_b, n_k, n_c\}$), use the training set to calculate g_i^- and evaluate $E_i = E_{val}(g_i^-)$ in the validation set, where the index i corresponds to each possible model structure.

2) After testing all possible combinations, pick g^* corresponding to the smallest E_i.

3) Train all N points using g^* and report the final hypothesis g, which has the same model structure as g^*, but different coefficient values.

Concerning the second step presented above, testing all possible combinations is only feasible because we handle linear models. In the case of nonlinear models, testing all possibilities is unfeasible computationally.

To exemplify, given a time series data set containing 100 points (one corresponding to each month), the data corresponding to the first 60 months is used for training as training set, the data from the following 20 months for validation as validation set and the data from 20 last months for test as test set. Test set performance provides an unbiased estimation of how well the final hypothesis is performing, whereas the validation set provides an optimistically biased estimation of how well g can perform in terms of forecasting. It is worthwhile mentioning that, depending on the availability of data set, we cannot afford to provide an unbiased estimation in test set. In this case, we should use only training and validation sets in order to make the most of limited available data.

2.5 Estimation of Parameter Uncertainty

An important aspect of the reservoir problem is uncertainty estimation. In fact, data is observed only at a few points throughout the extension of the reservoir, which makes it difficult to ascertain how dynamics evolves. Therefore it is important to quantify uncertainties so that a forecast provides a range of predicted values instead of a single prediction.

The range of uncertainty amplitude depends on parameter covariance matrix, which is related to the error made by the hypothesis, as well as the volume of data used for analysis. For instance, if a model presents little error in a big data set for a long period, parameter estimation presents a small uncertainty ellipsoid around average parameters.

In this section, we will discuss the assumptions about how to characterize uncertainties in linear models, such as ARX. In addition to this, we will present a technique that sample models on the frontier of the uncertainty ellipsoid, which is faster than Monte Carlo simulation.

2.5.1 Covariance Matrix Estimation

AGUIRRE describes the meaning of bias in parameter estimation. Estimation unbiased when

$$b = \mathrm{E}[\hat{\theta}] - \theta = 0,$$

where θ is the true parameter and $\hat{\theta}$ is an estimate.

Least squares estimation provides $\hat{\theta} = Ay$, where $A = [\psi^T\psi]^{-1}\psi^T$. Therefore

$$\begin{aligned} b &= \mathrm{E}[\hat{\theta}] - \theta \\ &= \mathrm{E}[A(\psi\theta + e) - \theta] \\ &= \mathrm{E}[(A\psi - I)\theta] + \mathrm{E}[Ae] \\ &= \mathrm{E}[A\psi - I]\theta + A[Ae]. \end{aligned}$$

From the calculations above, the following conditions should hold in order to arrive at an unbiased estimate:

1) $A\psi = I$. This condition is satisfied by least squares.

2) $\mathrm{E}[Ae] = \mathrm{E}[A]\mathrm{E}[e]$, which means A and e are statistically independent.

In words, estimation is unbiased when regressor matrix ψ and error e are not correlated.

Least squares error can be calculated as:
$e_{LS} = y - \psi\hat{\theta} = \psi\theta + e - \psi\hat{\theta} = \psi(\theta - \hat{\theta}) + e$.

Least squares error (e_{LS}) is in left null space, which is perpendicular to column space of ψ. Therefore $\psi^T e_{LS} = 0 \implies \psi^T\psi(\theta - \hat{\theta}) + \psi^T e = 0 \implies$

$$\mathrm{E}[\psi^T e] = \mathrm{E}[\psi^T\psi(\hat{\theta} - \theta)] = \mathrm{E}[\psi^T\psi]\mathrm{E}[\hat{\theta} - \theta] = \mathrm{E}[\psi^T\psi]b.$$

Bias is zero when error and feature matrix are independent. This is why the extended feature matrix presented in section 2.3.2 includes $e(k-1), e(k-2), ...$ and so forth.

3) $\mathrm{E}[e] = 0$. This condition is easily accomplished by taking into account a bias column in feature matrix.

When estimation is unbiased, the covariance matrix can be estimated as follows:

$$\begin{aligned}
cov[\hat{\theta}] &= \mathrm{E}[(\hat{\theta} - \mathrm{E}[\hat{\theta}])(\hat{\theta} - \mathrm{E}[\hat{\theta}])^T] \\
&= \mathrm{E}[(Ay - \theta)(Ay - \theta)^T] \\
&= \mathrm{E}[Ayy^T A^T - Ay\theta^T - \theta y^T A^T + \theta\theta^T] \\
&= \mathrm{E}[A(\psi\theta + e)(\psi\theta + e)^T A^T - A(\psi\theta + e)\theta^T - \theta(\psi\theta + e)^T A^T + \theta\theta^T] \\
&= \mathrm{E}[(A\psi\theta + Ae)(A\psi\theta + Ae)^T - (A\psi\theta + Ae)\theta^T - \theta(A\psi\theta + Ae)^T + \theta\theta^T]
\end{aligned}$$

But $A\psi = [\psi^T\psi]^{-1}\psi^T\psi = I$, then:

$$\begin{aligned}
cov[\hat{\theta}] &= \mathrm{E}[(\theta + Ae)(\theta + Ae)^T - (\theta + Ae)\theta^T - \theta(\theta + Ae)^T + \theta\theta^T] \\
&= \mathrm{E}[Aee^T A^T]
\end{aligned}$$

By assuming error as a white noise with variance σ_e^2 :

$$cov[\hat{\theta}] = (\psi^T\psi)^{-1}{\sigma_e}^2. \tag{2.2}$$

2.5.2 Sampling on the boundary of the uncertainty ellipsoid

A proper ellipsoid in the n-dimensional space R^n centered at the origin may be defined by the quadratic form

$$E = \{x | x^T \Sigma^{-1} x = 1\}, \tag{2.3}$$

where Σ is the covariance matrix. Considering Σ to be the covariance matrix from a Gaussian distribution, E in equation 2.3 is the boundary of the uncertainty ellipsoid corresponding to $\sigma = 1$ standard deviation.

The covariance matrix is symmetric and can be written as

$$\Sigma = QDQ^T,$$

where D is the diagonal matrix of eigenvalues of Σ and Q is the orthogonal matrix of eigenvectors of Σ. Therefore

$$x^T(QDQ^T)^{-1}x = 1 \implies x^T QD^{-1}Q^T x = 1.$$

By choosing $y = Q^T x$,

$$y^T D^{-1} y = 1$$

implies

$$\frac{y_1^2}{\lambda_1} + \frac{y_2^2}{\lambda_2} + ... + \frac{y_n^2}{\lambda_n} = 1,$$

where λ_i is the i-th eigenvalue of Σ.

Choosing $t_i = y_i^2$,

$$\frac{t_1}{\lambda_1} + \frac{t_2}{\lambda_2} + ... + \frac{t_n}{\lambda_n} = 1 \tag{2.4}$$

In order to make left hand side of equation 2.4 equal to 1, $t_i = \lambda_i v_i$, where $v_i = \frac{k_i}{\sum\limits_{i=1}^{n} k_i}$. For randomization purposes, k_i is chosen from a uniform distribution $\mathcal{U}[0, 1]$.

Therefore $y_i = \pm\sqrt{t_i}$. The signal is chosen randomly between $+$ and $-$.

Finally, for σ uncertainty level estimation, a sample on the ellipsoid of uncertainties considering a Gaussian distribution centered at $\bar{x}$ with covariance matrix Σ is

$$x = \bar{x} + \sigma Q y.$$

2.6 Online Learning

In this section, we will present the recursive least squares algorithm, which provides an adaptive method in which parameter estimates are updated as new data become available. This algorithm is suitable for reservoir identification problem because the reservoir dynamics is time varying, especially when water or gas breakthrough occurs.

In this work, we will explore import aspects of RLS filter, such as forgetting factor and the choice of initial condition, as well as its mathematical formulation.

2.6.1 Recursive Least Squares

Given a model structure, the least squares pseudo inverse parameter estimate is

$$\hat{\theta} = (\psi^T \psi)^{-1} \psi^T y. \tag{2.5}$$

We define

$$P = (\psi^T \psi)^{-1}. \tag{2.6}$$

From pseudo inverse solution in equation 2.5, where ψ_{i-1} represents i-th row of the matrix ψ.

$$\hat{\theta_k} = P_k \sum_{i=1}^{k} \psi_{i-1}^T y_i = P_k \sum_{i=1}^{k-1} \psi_{i-1}^T y_i + P_k \psi_{k-1}^T y_k. \tag{2.7}$$

For instant $k-1$,

$$\hat{\theta}_{k-1} = P_{k-1} \sum_{i=1}^{k-1} \psi_{i-1}^T y_i \implies \sum_{i=1}^{k-1} \psi_{i-1}^T y_i = P_{k-1}^{-1} \hat{\theta}_{k-1}. \tag{2.8}$$

From the definition of matrix P in equation 2.6,

$$P_k = [\sum_{i=1}^{k} \psi_{i-1}^T \psi_{i-1}]^{-1} \tag{2.9}$$

$$P_k^{-1} = \sum_{i=1}^{k-1} \psi_{i-1}^T \psi_{i-1} + \psi_{k-1}^T \psi_{k-1} \tag{2.10}$$

$$= P_{k-1}^{-1} + \psi_{k-1}^T \psi_{k-1}. \tag{2.11}$$

Combining equations 2.7 and 2.8,

$$\hat{\theta_k} = P_k [P_{k-1}^{-1} \hat{\theta}_{k-1} + \psi_{k-1}^T y_k]. \tag{2.12}$$

From equations 2.12 and 2.11,

$$\hat{\theta_k} = P_k (P_k^{-1} - \psi_{k-1}^T \psi_{k-1}) \hat{\theta}_{k-1} + P_k \psi_{k-1}^T y_k \tag{2.13}$$

$$= \hat{\theta}_{k-1} + P_k \psi_{k-1}^T (y_k - \psi_{k-1} \hat{\theta}_{k-1}) \tag{2.14}$$

The gain matrix is defined as

$$K_k = P_k \psi_{k-1}^T \tag{2.15}$$

and the innovation at instant k is defined as

$$\eta_k = y_k - \psi_{k-1} \hat{\theta}_{k-1}. \tag{2.16}$$

The calculation of P_k still requires one matrix inversion at every algorithm iteration. This can be avoided by using the inversion lemma

$$(A+BCD)^{-1} = A^{-1} - A^{-1}B(C^{-1}+DA^{-1}B)^{-1}DA^{-1},$$

with

$$A = P_{k-1}^{-1}, B = \psi_{k-1}^{T}, C = I, D = \psi_{k-1},$$

and equation 2.11, P can be written as:

$$P_k = P_{k-1} - P_{k-1}\psi_{k-1}^{T}(\psi_{k-1}P_{k-1}\psi_{k-1}^{T}+1)^{-1}\psi_{k-1}P_{k-1}. \tag{2.17}$$

The gain matrix is calculated combining equations 2.15 and 2.17,

$$K_k = \frac{P_{k-1}\psi_{k-1}^{T}}{\psi_{k-1}P_{k-1}\psi_{k-1}^{T}+1}$$

To sum up, the RLS algorithm can be implemented recursively

$$\begin{aligned} K_k &= \frac{P_{k-1}\psi_{k-1}^{T}}{\psi_{k-1}P_{k-1}\psi_{k-1}^{T}+1} \\ \hat{\theta}_k &= \hat{\theta}_{k-1} + K_k(y_k - \psi_k\hat{\theta}_{k-1}) \\ P_k &= P_{k-1} - K_k\psi_{k-1}P_{k-1}. \end{aligned}$$

Furthermore, it is possible to give more importance to more recent data by using a forgetting factor λ. In this case, instead of regular least squares, a weighted least squares problem is solved.

The objective function to be minimized is $\sum_{i=0}^{k}[(y_i - \psi_{i-1}\theta]^2\lambda^{k-i}$, with $\lambda \leq 1$, typically between 0.95 and 1. If $\lambda = 1$, regular least squares is solved and all measured data have same importance. The lower λ is, the greater the weight given to more recent data, which means that past data is "forgotten" faster.

It is important to mention that depending on the choice of the forgetting factor, the filter may diverge. There is a trade-off between quick adaptation and filter convergence. A suitable forgetting factor level is chosen by comparing forecast match with real data.

The innovation η_k (equation 2.16) represents how fast parameters may change. If innovation is zero, filter does not adapt. On the other hand, if the dynamical system is no longer able to predict future correctly due to a change in dynamics, the RLS algorithm with forgetting factor provides an adaptive filter which aims to learn from new data.

RLS with forgetting factor (λ) algorithm can be implemented recursively:

$$
\begin{aligned}
K_k &= \frac{P_{k-1}\psi_{k-1}^T}{\psi_{k-1}P_{k-1}\psi_{k-1}^T + \lambda} \\
\hat{\theta}_k &= \hat{\theta}_{k-1} + K_k(y_k - \psi_k\hat{\theta}_{k-1}) \\
P_k &= \frac{1}{\lambda}(P_{k-1} - \frac{P_{k-1}\psi_{k-1}^T\psi_{k-1}P_{k-1}}{\psi_{k-1}P_{k-1}\psi_{k-1}^T + \lambda})
\end{aligned}
$$

If no prior information is known, a standard choice is $\theta = 0$ and $P = 10^k I$, where I is the identity matrix and $3 \leq k \leq 7$.

2.6.2 Practical Remarks about the Prior Knowledge in the RLS filters

If prior information, such as a reservoir simulation result is known, model selection procedure discussed previously can be used to define the model structure as well as the initial condition for the RLS filter. This is a means by which prior knowledge in an adaptive framework is able to provide physical aspects for the filter.

Prior knowledge can consider similarities between wells. In case of lack of information from reservoir simulator, a similar well history data may provide an initial condition guess for RLS filter.

2.7 Simulator-based identification and validation of reservoir models

We analyze the “Two Flow Model” reservoir simulator, developed by Emerick [20], in which there are 6 water injectors and 2 liquid producers.

Figure 2.4 depicts, in log scale, permeability distribution across reservoir cell grids. In fact, this is a realization of a prior model in which the prior covariance matrix follows a spherical covariance function.

There are two important connections, in which it is possible to notice a preferential path from injector 1 to producers 2 and 3, and another high permeability path from injector 2 to producers 4 and 5.

This case study aims to check whether system identification procedure can be successfully implemented in terms of capturing geological aspects, such as high permeability preferential paths. Moreover, we want to establish models that are suitable for forecasting and assess them in independent test sets.

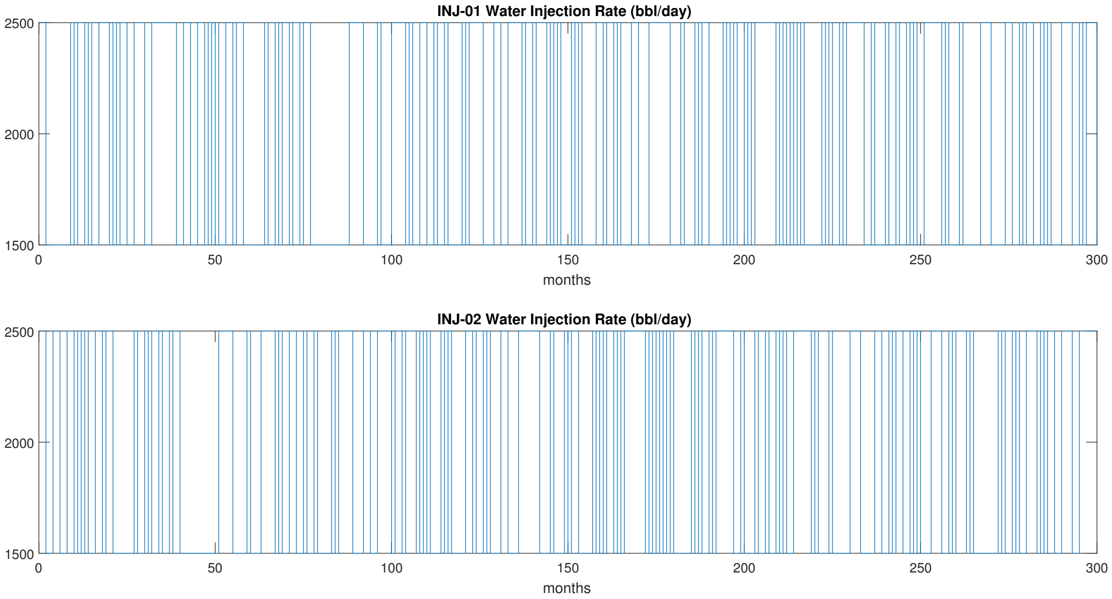

Figure 2.2: Water Injection Rates PRBS Excitation for System Identification. The Liquid Producer Rates are kept constant.

2.7.1 Choice of Features and Experimental Design

In this experimental identification design, Liquid Producer Rates are kept constant, while injection rates varies according to Table 2.1. PRBS (pseudo random binary sequence) excitation is applied in injectors, depicted in Figure 2.2.

Table 2.1: Range of perturbation

Well	MIN	MAX
INJ-01	1500	2500
INJ-02	1500	2500
PRO-01	400	400
PRO-02	700	700
PRO-03	700	700
PRO-04	700	700
PRO-05	700	700
PRO-06	400	400

We could consider, as possible choices of features, meaningful outputs such as BHP (bottom hole pressure), WC (water cut) and Oil Rate as functions of water injection rates, which were chosen as PRBS excitations. BHP could be analysed as output, given that liquid production rate in simulation is constant and BHP reflects block pressure variation, as depicted in Figure 1.1.

Another possibility is to assess water cut evolution and its correlations to water

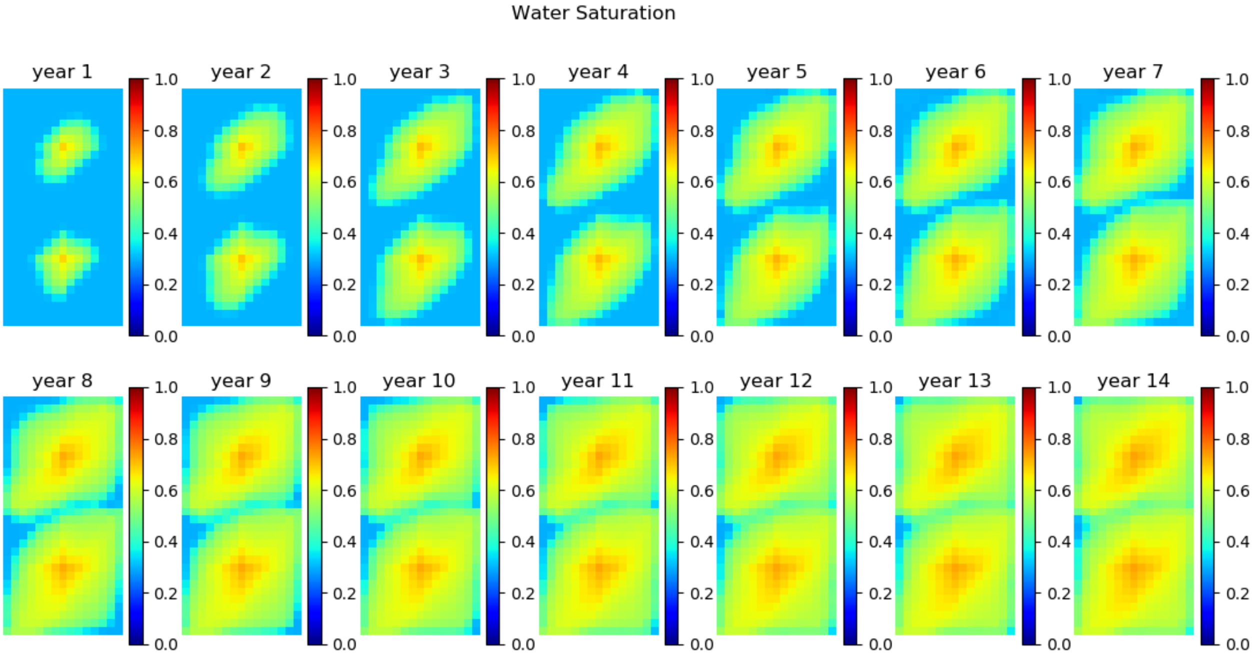

Figure 2.3: Water saturation map is represented during 14 years. It is worthwhile pointing out the time variant characteristics of the production system, which varies especially when water breakthrough occurs.

injection rates. Liquid Production Rate could be an interesting alternative choice if BHP of producer wells were held constant, as chosen in Figure 1.2, but in this case study the liquid production is chosen as a bias, since it is held constant in the simulations.

Figure 2.3 depicts water saturation evolution in the reservoir for a 14 years period. In 3 years, we can observe water breakthrough first in wells 2, 3 and 5 and subquently in wells 4, 6 and 1. Many of the examples in this chapter aim to show how effective polynomial ARX and ARMAX representations can be in terms of making predictions.

2.7.2 Results of ARX and ARMAX modeling

PRO-01 In this well, we will consider as output PRO-01 bottom hole pressure. The features includes water injection rates (for both injector wells) and liquid production rate (for producer PRO-01), which is bias in this case study.

The time window for analysis ranges from month 170 to 200, which means that, for a total period of 30 months, 60% of data is used for training, 20% for validation and 20% for test.

Figure 2.5 depicts training and validation results. 10 coefficients are used and

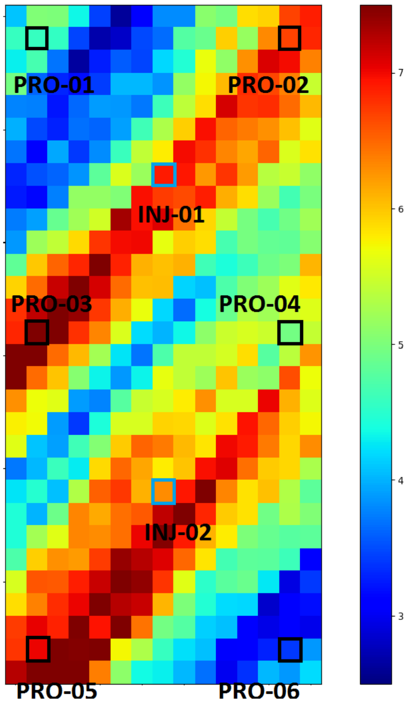

Figure 2.4: Reference log-permeability field, making evident the permeability path connectivity between INJ-01 and PRO-02/PRO-03 and INJ-02 and PRO-04/PRO-05.

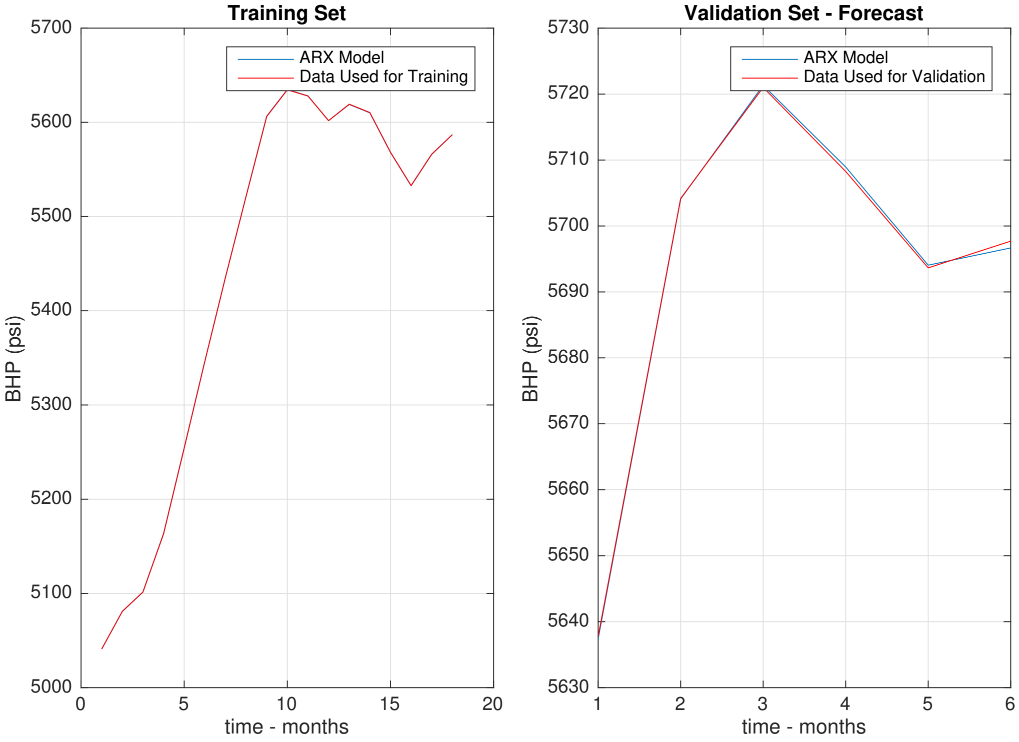

Figure 2.5: Model Structure choice based on the Validation Set for the producer well PRO-01.

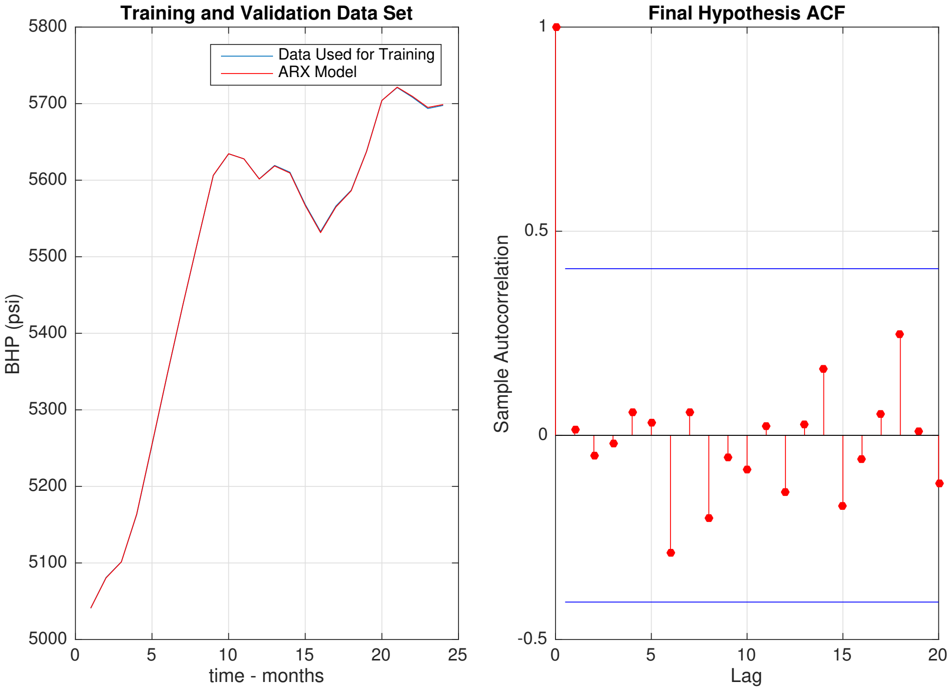

Figure 2.6: Chosen the model structure, training and validation data set are used for training. Residual analysis indicates that the error is a white noise for the producer well PRO-01.

ARX model structure is $n_a = 3, n_b = [1, 3, 3], n_k = [0, 0, 0]$. Once the ARX structure has been defined, training and validation data are used for parameter estimation and Figure 2.6 shows the results of a simulated model, which are quite similar to training data set. ACF suggests, from residual analysis, that one step ahead error is white noise. In other words, ACF means that ARX model captures all linear correlations in the data.

Uncertainty estimation analysis reveals little uncertainty with respect to the estimated parameters and, indeed, test dataset results lie within the range of uncertainty of model prediction. Uncertainty ellipsoid considering 3 standard deviation with 50 perturbations, shown in Figure 2.7, in month 200 ranges from 6125 to 6150 psi and real data corresponds to 6141 psi, an error smaller than 0.05% compared to simulated forecast. Monte Carlo analysis also presents little uncertainty with 3000 simulations and test dataset is bounded by P-40 and P-60 curves. In this work, P-x denotes a measure indicating the value above which a given percentage (x) of observations in a group of observations falls.

With regards to step response analysis, as depicted in Figure 2.9, positive slope for the average case in step response for both injectors suggests that they provoke bottom hole pressure increase. In fact, this should be expected, since block pressure increases and liquid rate is kept constant in simulation. Interestingly, step response

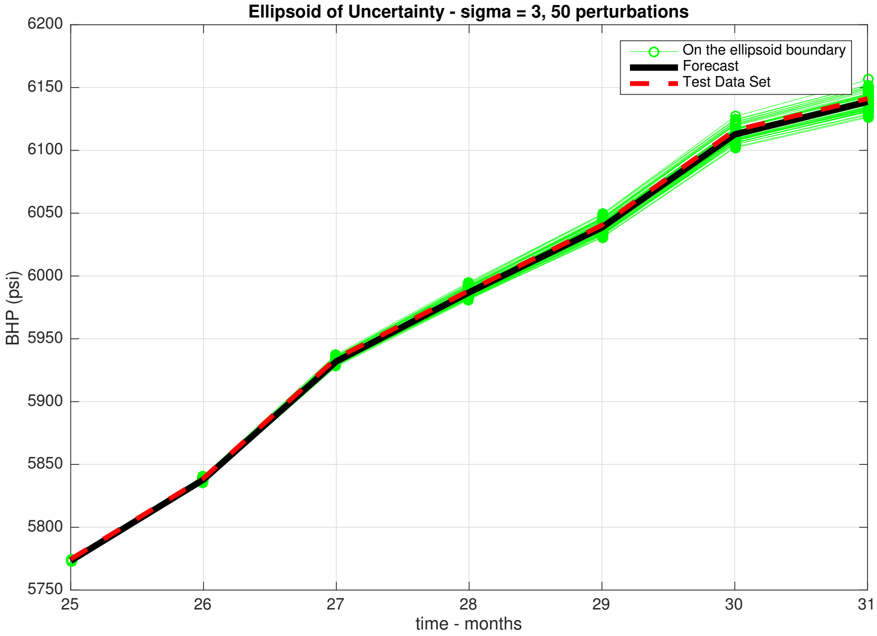

Figure 2.7: Test set assessment comparing the forecast on the average case (in black) and Test Data Set. In green, 50 different parameter vectors with the same model structure were sampled on the Ellipsoid of Uncertainty considering 3 standard deviation for the producer PRO-01.

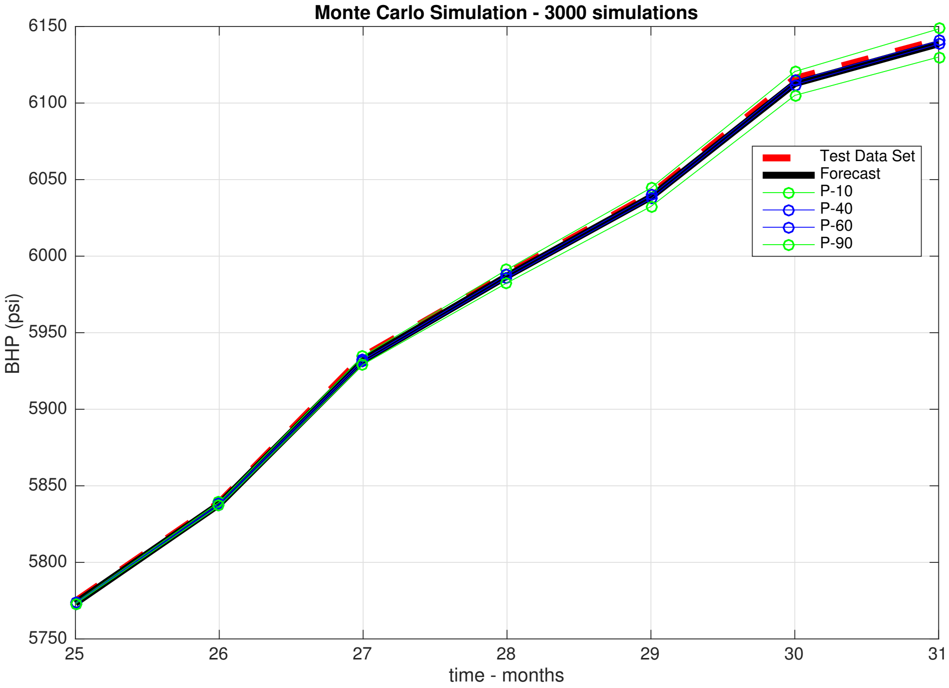

Figure 2.8: Instead of sampling on the ellipsoid of uncertainty, Monte Carlo simulation is performed. The curves P-10, P-40, P-60 and P-90 are defined with 3000 simulations for the producer well PRO-01.

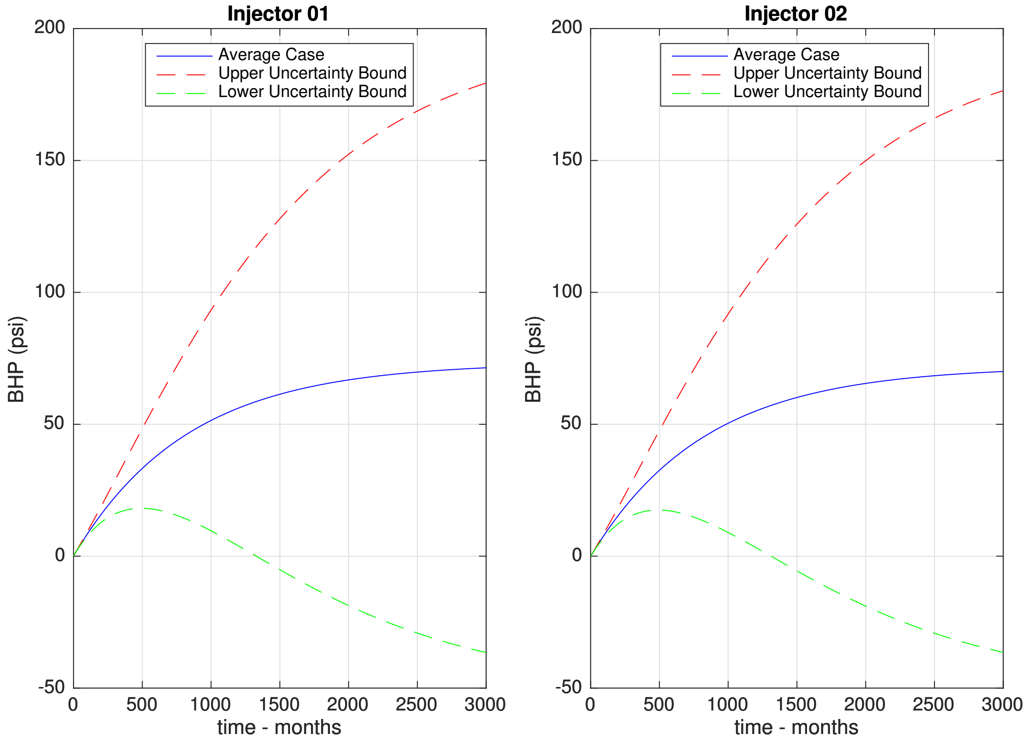

Figure 2.9: Step Response: BHP - PRO-01. Both INJ-01 and INJ-02 contribute for BHP increase for the producer well PRO-01 in the step response.

suggests little difference from injector 1 to 2, even though injector 1 is located closer to producer 1.

PRO-02 A period from month 40 to 140 is chosen for data analysis, where water cut ranges from 40% to 93%. Training, validation and test set are distributed according to 60%, 20% and 20%. Exogenous components are water injection rates, liquid rate (bias) and output is water cut.

Figures 2.4 and 2.3 show a strong correlation between water cut in well 2 and water injection rate from injector 1. The Water Saturation Map (Figure 2.3) shows how the water front propagates spatiotemporally from the injector wells towards the producer wells. The waterfront spatial propagation follows the permeability map (Figure 2.4) of the grid. Each block in the grid has a permeability value indicated by the corresponding color in the heat map in Figure 2.4. The preferential paths have higher permeability values (towards the red end of the spectrum). We would like to assess whether step response captures the correlation between water cut in well 2 and water injection rate from injector 1.

Figure 2.10 depicts training and validation results for ARX model. The structure which minimizes the error in the validation test is $n_a = 2$, $n_b = [1, 2, 3]$ and $n_k =$

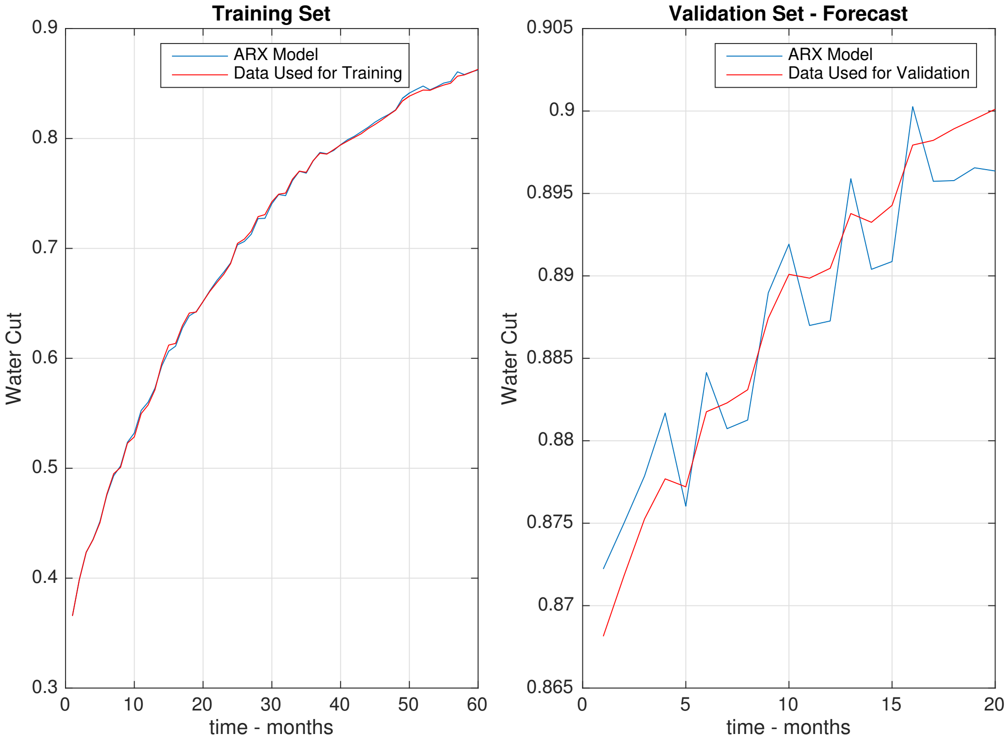

Figure 2.10: Model Structure choice based on the Validation Set for the producer well PRO-02.

$[0, 0, 1]$.

Once input and output delays are chosen for the ARX structure, the validation data set is also used for training. Figure 2.11 shows ACF error (auto-correlation function) considering one step ahead output prediction. ACF represents a white noise, which means there is no need to sophisticate the hypothesis by trying to model the noise.

Figures 2.12 and 2.13 show forecasts over the test set, considering the ellipsoid of uncertainty and Monte Carlo methods, respectively. Although results are quite similar, Monte Carlo methods require a great deal of simulations in order to characterize completely uncertainty distribution. Moreover, the sampling on the uncertainty ellipsoid with $\sigma = 3$ provides, as far as test set confirms, reliable bounds in terms of output prediction.

In terms of step response, the result follows what we would expect. It is possible to notice, in Figure 2.14 a strong influence from INJ-01 on PRO-02 in terms of water cut increase during the evaluation time. On the other hand, there is very little influence from INJ-02 on PRO-02, as a consequence of the distance between them and the reservoir permeability map.

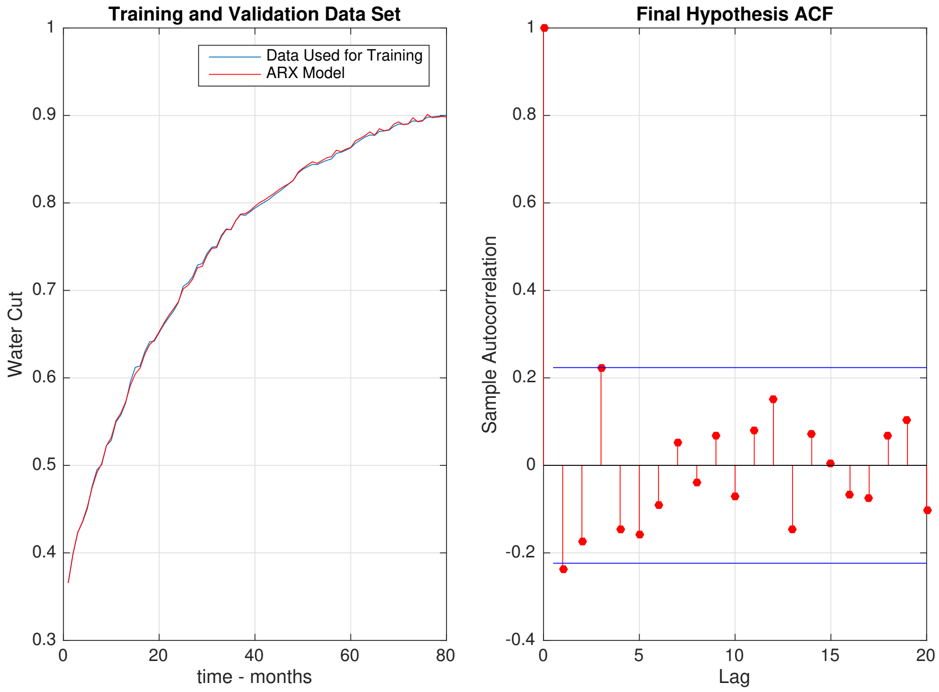

Figure 2.11: Chosen the model structure, training and validation data set are used for training. Residual analysis indicates that the error is not very different from a white noise for the producer well PRO-02.

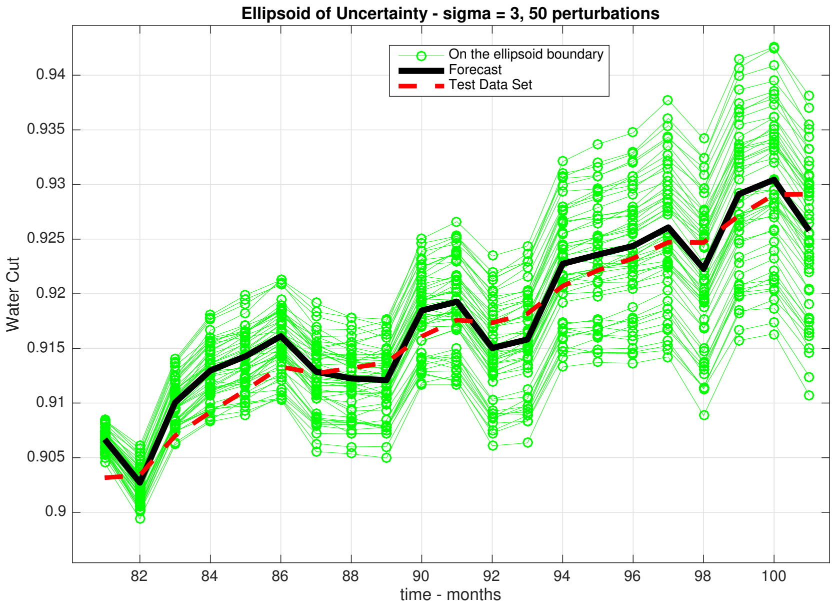

Figure 2.12: Test set assessment comparing the forecast on the average case (in black) and Test Data Set. In green, 50 different parameter vectors with the same model structure were sampled on the Ellipsoid of Uncertainty considering 3 standard deviation for the producer well PRO-02. It is worthwhile mentioning the scale, which ranges from 0.90 to 0.94, indicating a good agreement between the uncertainty analysis forecast and the test data set.

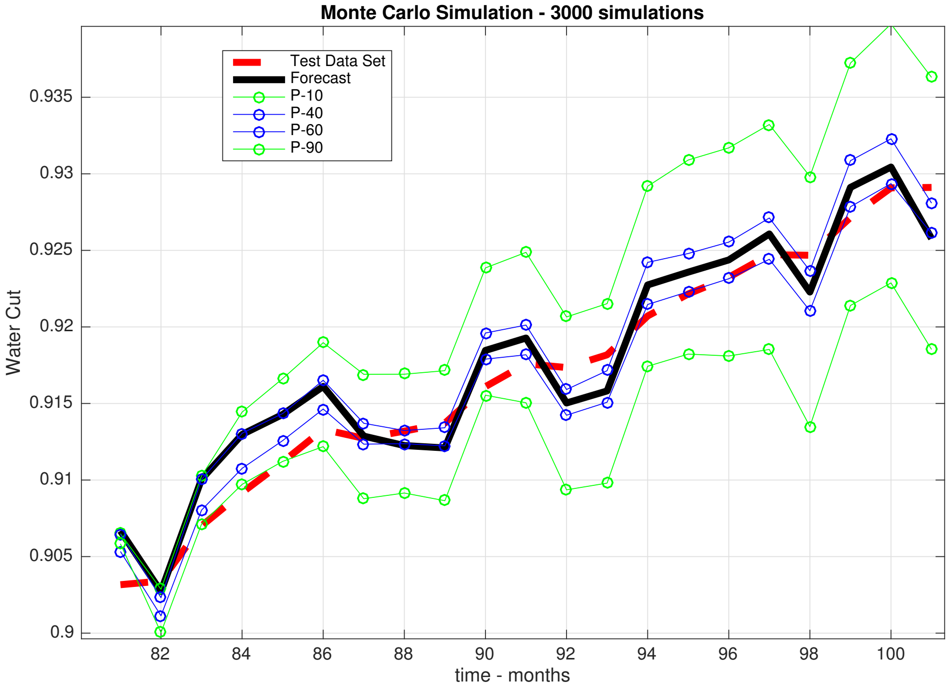

Figure 2.13: Instead of sampling on the ellipsoid of uncertainty, Monte Carlo simulation is performed. The curves P-10, P-40, P-60 and P-90 are defined with 3000 simulations for the producer PRO-02.

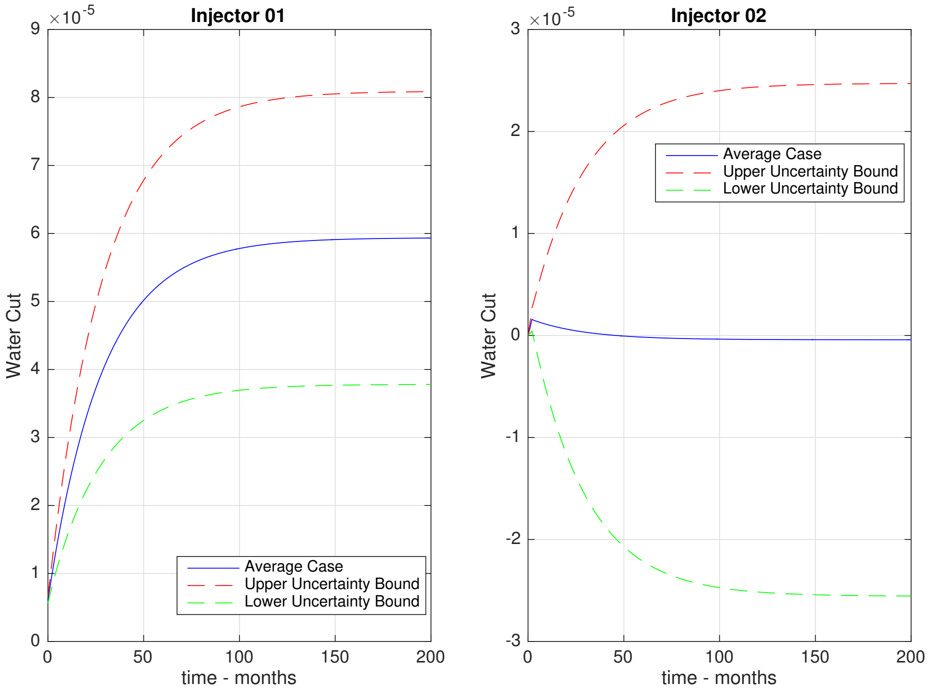

Figure 2.14: Step Response for the producer wel PRO-02. INJ-01 has a meaningful contribution for water cut increase in PRO-02, whereas INJ-02 seems to have very little influence.

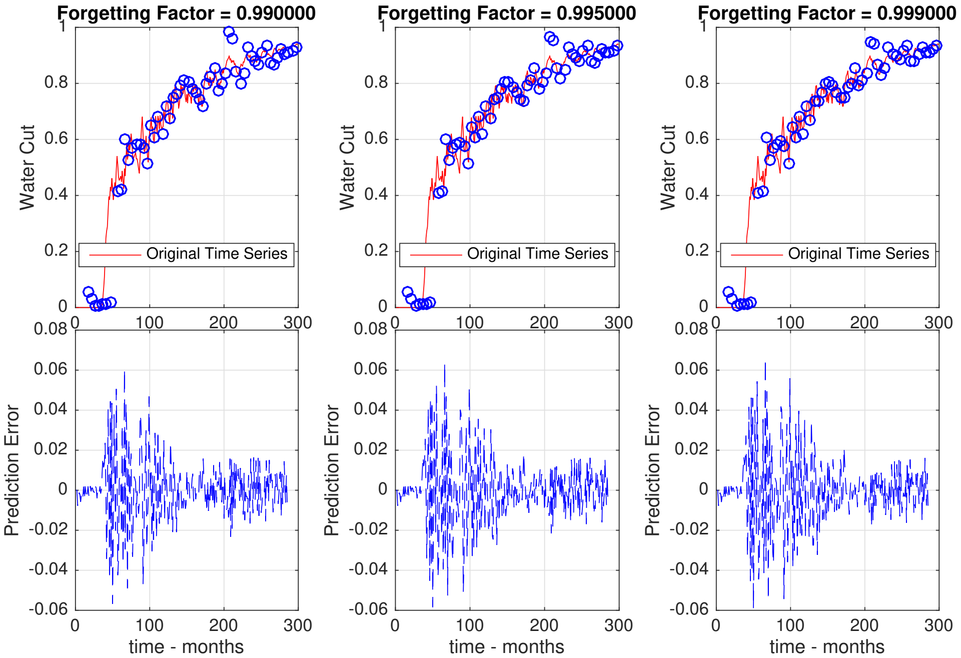

Figure 2.15: The graph shows 3 different choices for forgetting factors in the RLS filter. 12 months ahead prediction of water cut for the producer PRO-03 (in blue) is compared with the original time series, which comes from the model simulation.

PRO-03: Water Cut - Recursive Least Squares In this case study, we want to evaluate the performance of recursive least squares when it comes to adapting and forecasting. After considering different structures for this case study, we choose one that is simple enough to provide acceptable results in terms of both prediction accuracy and meaningful interpretation of the exogenous components. We choose $n_a = 2, n_b = [1, 2, 2], n_k = [0, 0, 0]$ considering liquid rate (bias) and water injection rates, respectively, as input features, and water cut as output.

Figure 2.15 shows 12 months ahead prediction in blue circle markers for three different levels of forgetting factors: $0.990, 0.995$ and 0.999. Initial condition for well 3 comes from the use of RLS during the whole period for well 4. This procedure avoids reuse of well 3 data to adjust parameters for well 3, which would be unfair. Additionally, this provides a realistic initial condition for well 3 based on the hypothesis that well 3 and 4 are close to each other.

Parameters adapt more when water breakthrough occurs (at time 35 months), which represents a significant change in well dynamics. This can be observed in prediction error, which provokes a bigger adaptation gain. No big differences are observed among forgetting factors, but it is possible to state that forgetting factor $= 0.990$ tends to make parameters more adaptive.

Considering forgetting factor 0.999, Figure 2.16 indicates how adaptation can

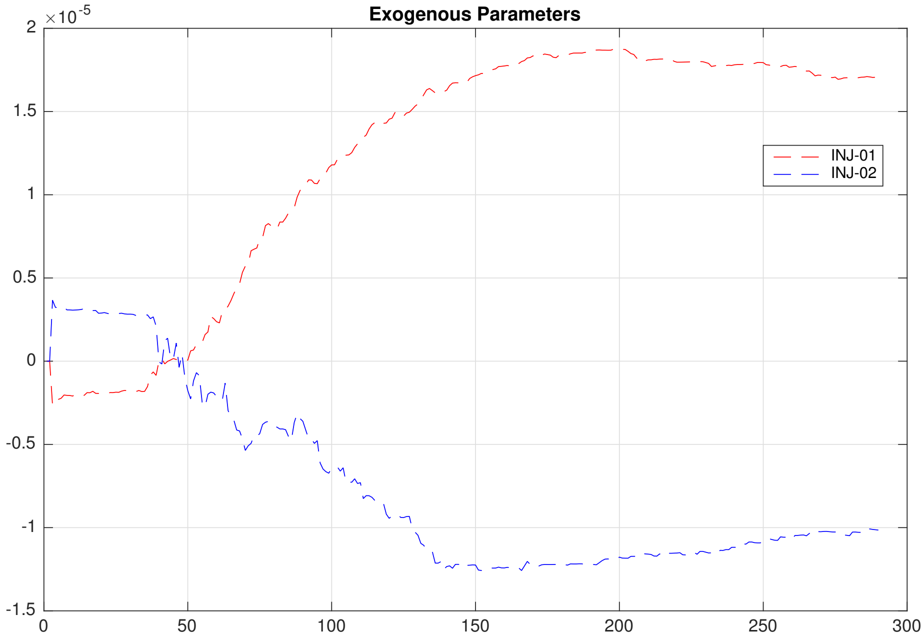

Figure 2.16: RLS - Exogenous Parameters Adaptation for the producer PRO-03. Interestingly, we see a pattern change after the water breakthrough, which means the filter adapts quickly and is able to capture the real connectivity among the wells.

provide a physical insight about dynamics in reservoir. Before water breakthrough, water cut was zero and no additional information is available, which means parameters were close to initial condition. Indeed, it is possible to observe a high permeability path between Injector-02 and Producer-04 in Figure 2.4.

At month 35, when water breakthrough occurs in well 3, water cut signal reveals information about reservoir and we see change in the parameter pattern, which is consistent from a physical viewpoint. In fact, from month 40 it is noticeable that injector-01 contributes to watercut increase in this well because the sum of injector-01 ARX parameters becomes and remains positive, whereas the sum of injector-02 ARX parameters becomes and remains negative, which suggests that injector-02 has the opposite effect, namely, injector-02 increases oil production of well 3, which is not an obvious conclusion a priori.

PRO-04: WC - Recursive Least Squares In this case study, we want to evaluate the evolution of parameters and their uncertainties while measured data is assimilated. RLS algorithm is performed and forgetting factor is 0.999. Initially, we considered a simple model structure (order two for both AR and exogenous components), which led to a biased forecast result. To avoid the difficulty in considering an adaptive model structure, we consider a more complex ARX model structure in

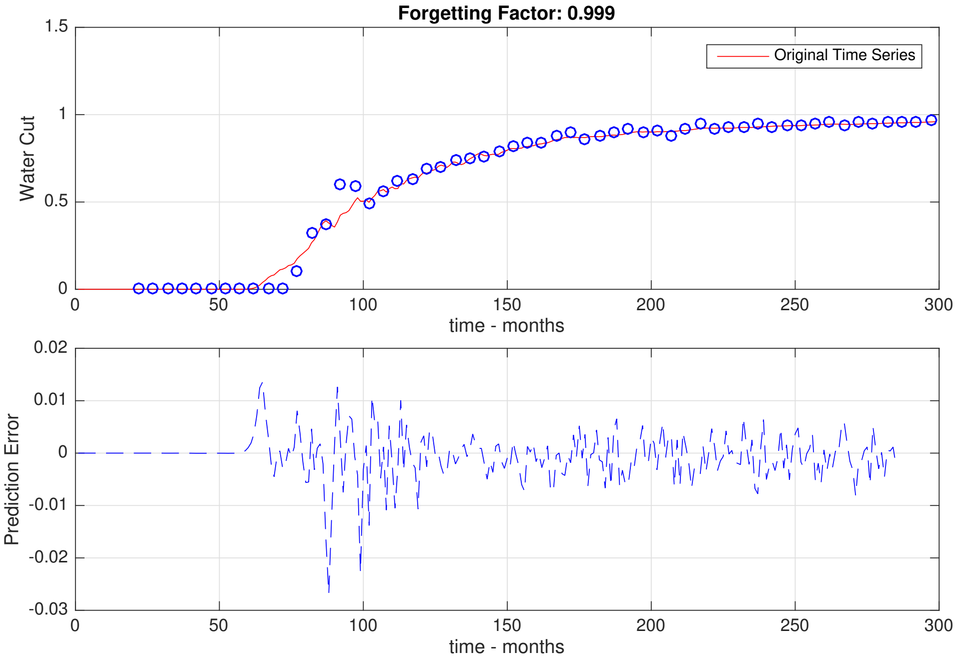

Figure 2.17: 12 months ahead prediction of water cut for the producer PRO-04 (in blue) is compared with the original time series, which comes from the model simulation. Bigger adaptation is observed when water breakthrough occurs.

which the parameters change in the RLS filter algorithm and the structure is fixed as $n_a = 8$, $n_b = [1, 8, 8]$ and $n_k = [0, 0, 0]$. This means that the first regressor is the liquid production rate (bias) and the others correspond to injection rates. We start from standard RLS initial condition, $\theta = 0$ and $P = 10^3 I$, where I is the identity matrix.

Figure 2.17 illustrates 12 months ahead prediction. Unlike the previous case, it is worthwhile mentioning that no prior information about the well is considered, only standard filter initial conditions. Prediction error graph indicates bigger adaptation when water breakthrough occurs.

Furthermore, Figure 2.18 presents the sum of exogenous parameters for each well. This graph suggests that INJ-02 has an important role regarding water cut increase in PRO-04.

PRO-05: BHP - ARX and ARMAX comparison So far, we have shown how effective ARX structure can be in terms of modeling and forecasting. We would like to investigate the benefits of considering ARMAX structure when residual analysis presents some indication that error could be modeled by a moving average filter.

From training and validation model selection previously presented in other cases, and considering BHP data from month 150 to 300, we find an ARX structure

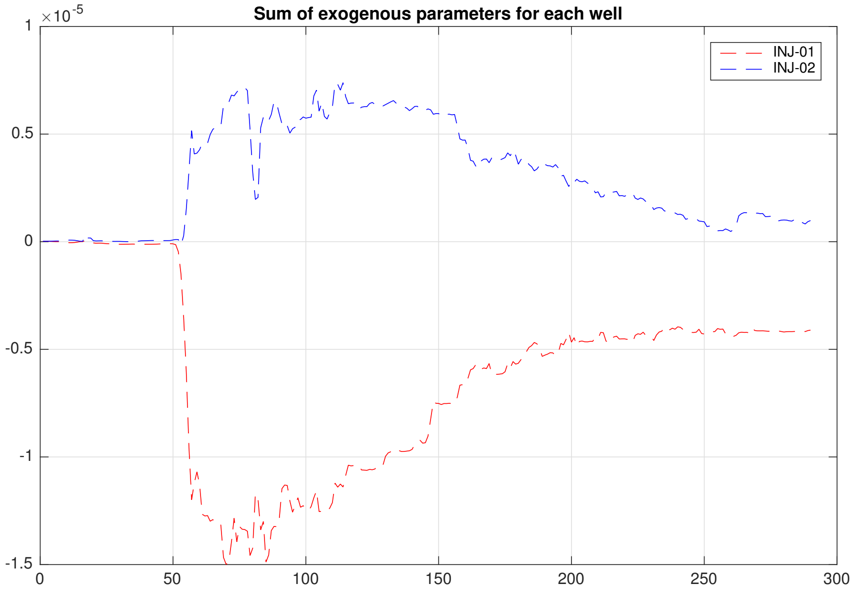

Figure 2.18: The evolution of the sum of exogenous parameters for each well for the producer PRO-04. Because the standard initial condition was chosen ($\theta = 0, P = 10^3 I$) the filter starts to learn from data when water breakthrough occurs. Until then, the output water cut is zero and there is no information to be learnt from. The parameters successfully indicate connectivity between INJ-02 and PRO-04.

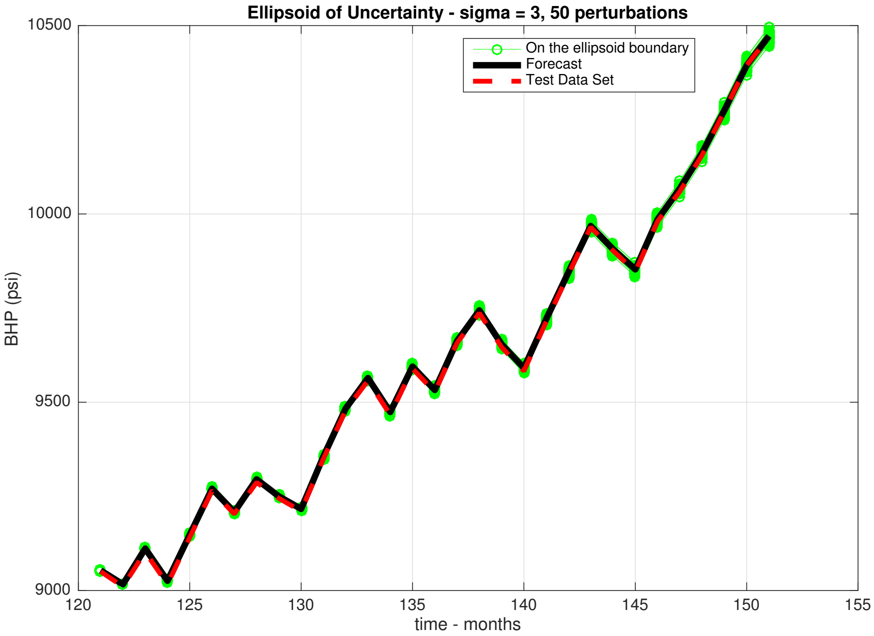

Figure 2.19: Test set assessment comparing the forecast on the average case (in black) and Test Data Set. In green, 50 different models were sampled on the Ellipsoid of Uncertainty considering 3 standard deviation for the producer PRO-05.

$n_a = 3, n_b = [1, 2, 2], n_k = [1, 0, 0]$ that presents remarkable results with respect to forecasting during 30 months, according to Figure 2.19.

However, residual analysis in Figure 2.20 presents 1-period lag out of ACF bounds, which means one step ahead error is not exactly white noise. For the sake of trying to model the noise, we choose an ARMAX structure keeping the same ARX delays and choosing $n_c = 1$, as far as residual analysis is concerned.

In fact, Figure 2.22 indicates white noise and ARMAX structure is still capable of making good predictions, as depicted in Figure 2.21.

Figure 2.23 illustrates little difference between both models in test set, in which ARMAX model performs slightly better. In practical terms, exogenous and auto regressive components play similar roles in both models and there is no great advantage to increasing complexity in the model. Therefore, even if the ARX model does not satisfy all requirements, it can be a good candidate, especially if it presents acceptable results in the test set.

Field Oil Rate Production - Prior Knowledge So far, we have considered data from the simulator. We will artificially create "observed" data from field by adding a perturbation. We would like to know how well a RLS filter can forecast taking into account prior knowledge from another source, such as the simulated model or another well with similar characteristics possessing a large history data

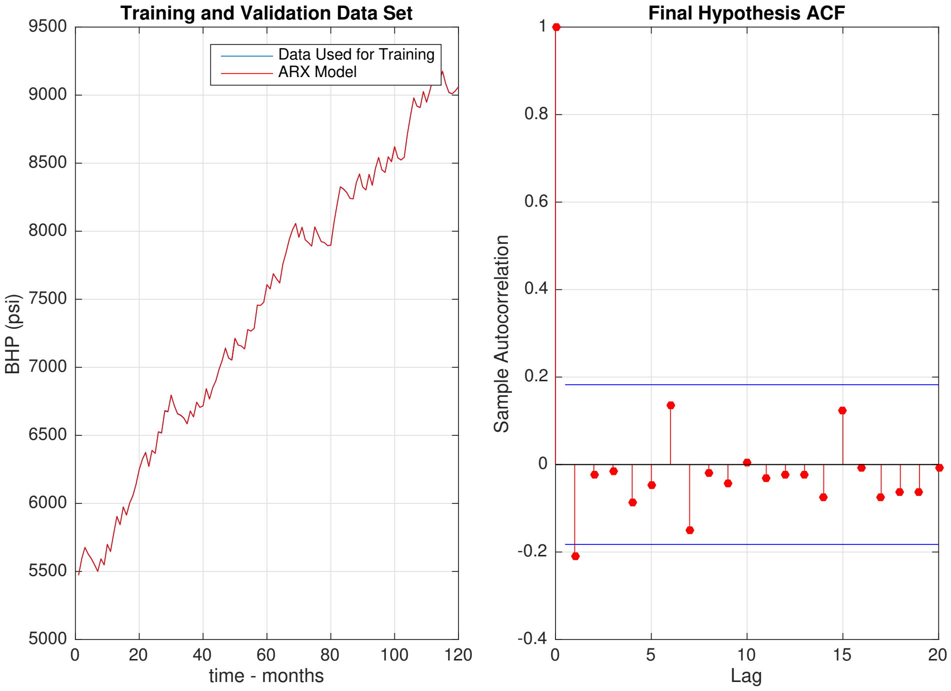

Figure 2.20: Residual analysis indicates 1-period lag out of ACF bounds, which suggests that error could be modeled with a moving average (MA) component. This is why we attempt to model it evolving from ARX structure to ARMAX.

set.

Figure 2.24 shows initially estimated oil rate, in red using prior information that could come from reservoir simulation, for instance. The blue curve represents observed data, which RLS filter will attempt to match.

Prior knowledge curve provides initial condition for RLS filter. We will compare the effect of choosing a prior initial condition with a standard RLS initial condition $(\theta = 0, P = 10^3 I)$.

Comparing Figures 2.26 and 2.25, both using forgetting factor $= 0.999, n_a = [2], n_b = [1, 2, 2], n_k = [0, 0, 0]$, the curve using an initial condition coming from data assimilation of a prior knowledge data set presents little difficulty in adapting when water cut is no longer zero, which happens when Field Oil Rate starts to decrease.

Even though RLS filter with standard initial condition filter $(\theta = 0, P = 10^3 I)$ does not adapt with ease, from month 100 ahead, 1 year ahead predictions present little mismatch with observed data. Therefore, this case study suggests that a prior knowledge information which provides an initial choice of the RLS parameters helps to boost the filter convergence, showing better forecast results, mostly when water breakthrough occurs.

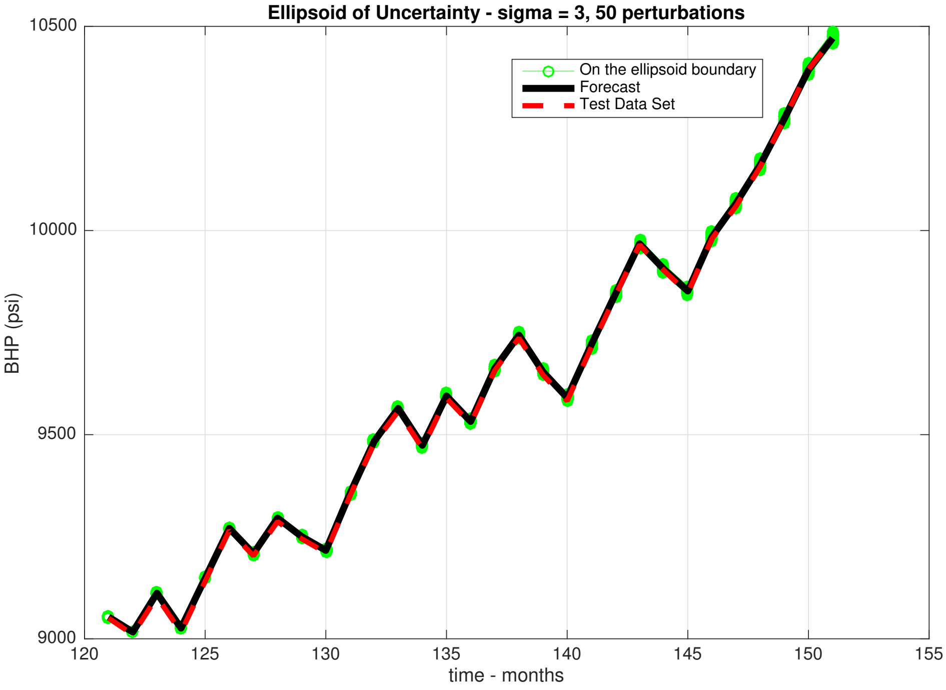

Figure 2.21: Test set assessment comparing the forecast on the average case (in black) and Test Data Set. In green, 50 different models were sampled on the Ellipsoid of Uncertainty considering 3 standard deviation for the producer PRO-05 using the ARMAX model.

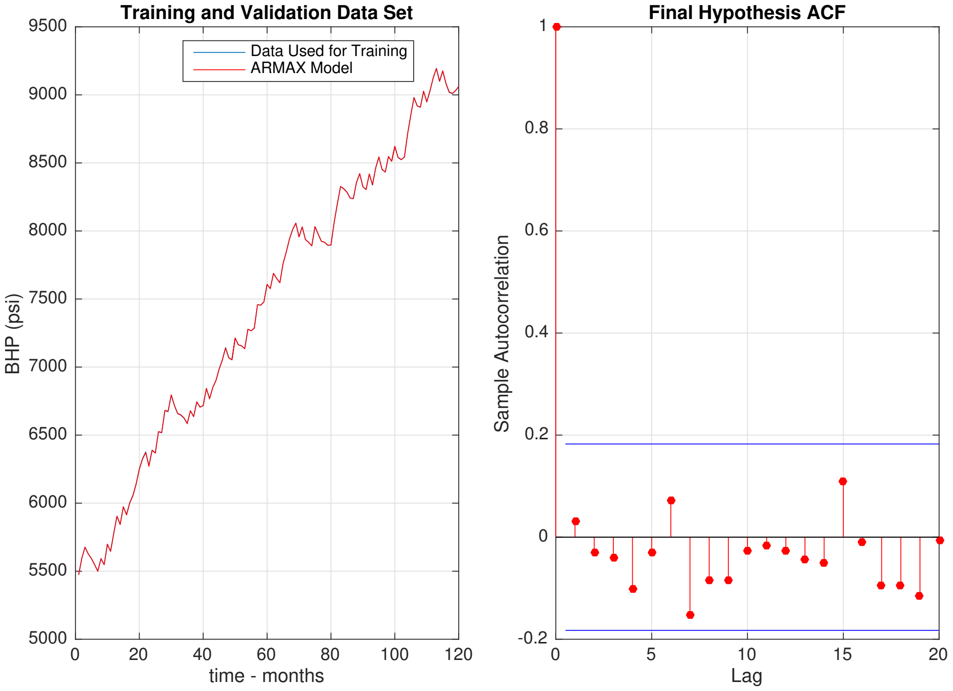

Figure 2.22: In fact, ARMAX model presents suitable results in residual analysis, which means choosing $n_c = 1$ successfully models the error.

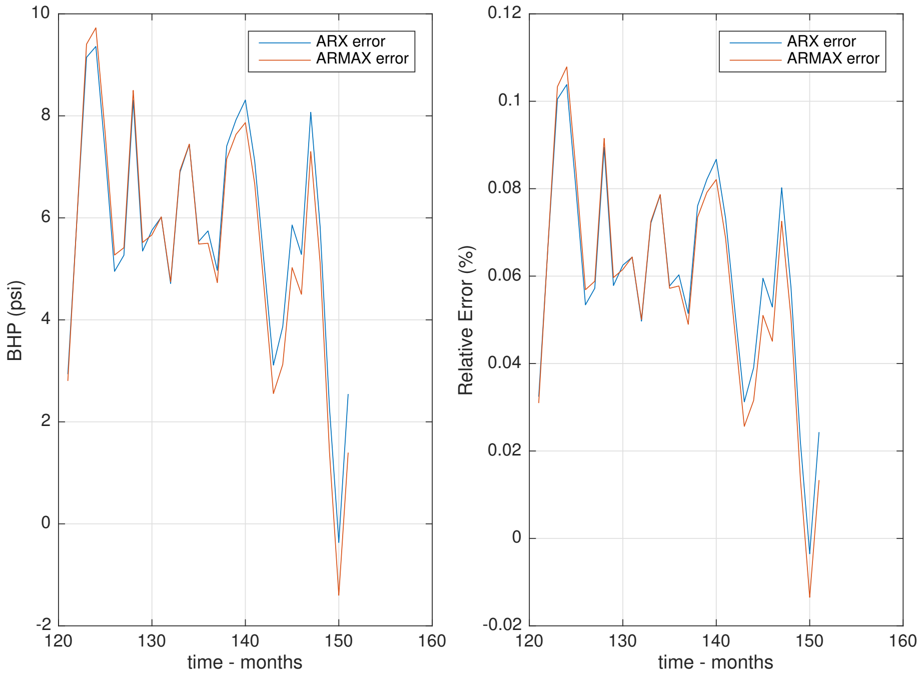

Figure 2.23: Comparison between ARX and ARMAX in the test data set. Despite its advantage of modeling the error, ARMAX presents results very similar to ARX in free simulation in the test data set.

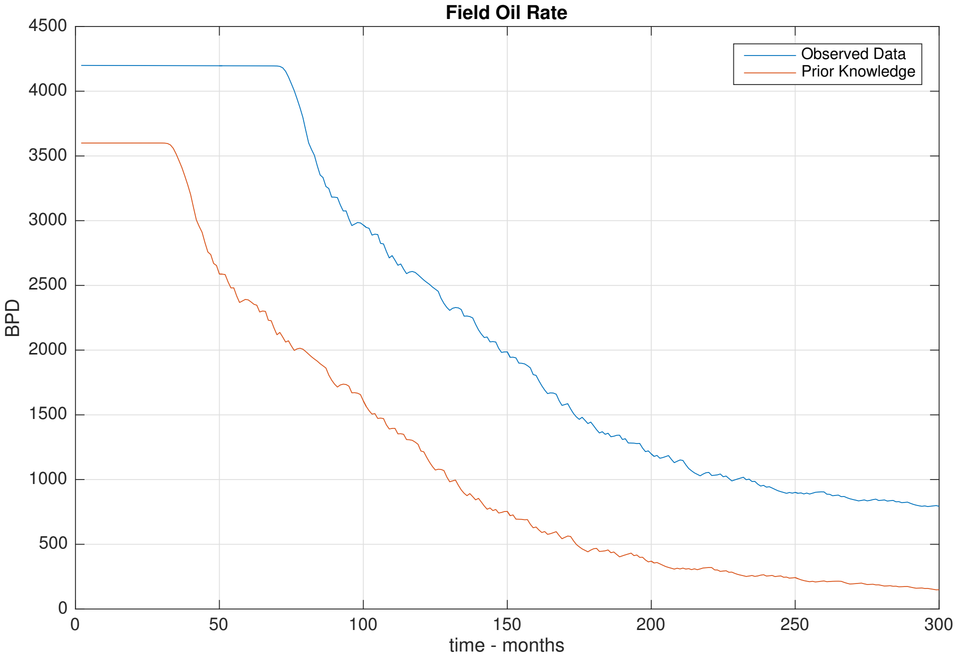

Figure 2.24: Observed Data and Prior Knowledge Data Set. Prior Knowledge may come from different sources, such as a numerical model reservoir simulation or another well with a large history data set with similar characteristics.

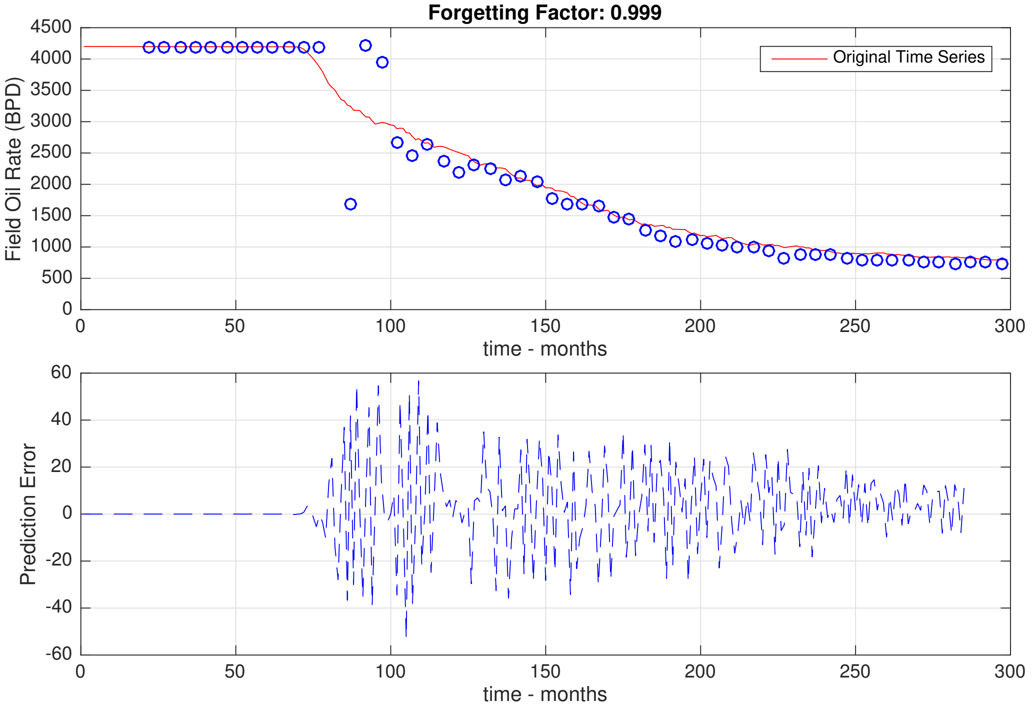

Figure 2.25: 12 months ahead prediction of the Field Oil Rate considering Standard RLS filter Initial Condition ($\theta = 0, P = 10^3 I$).

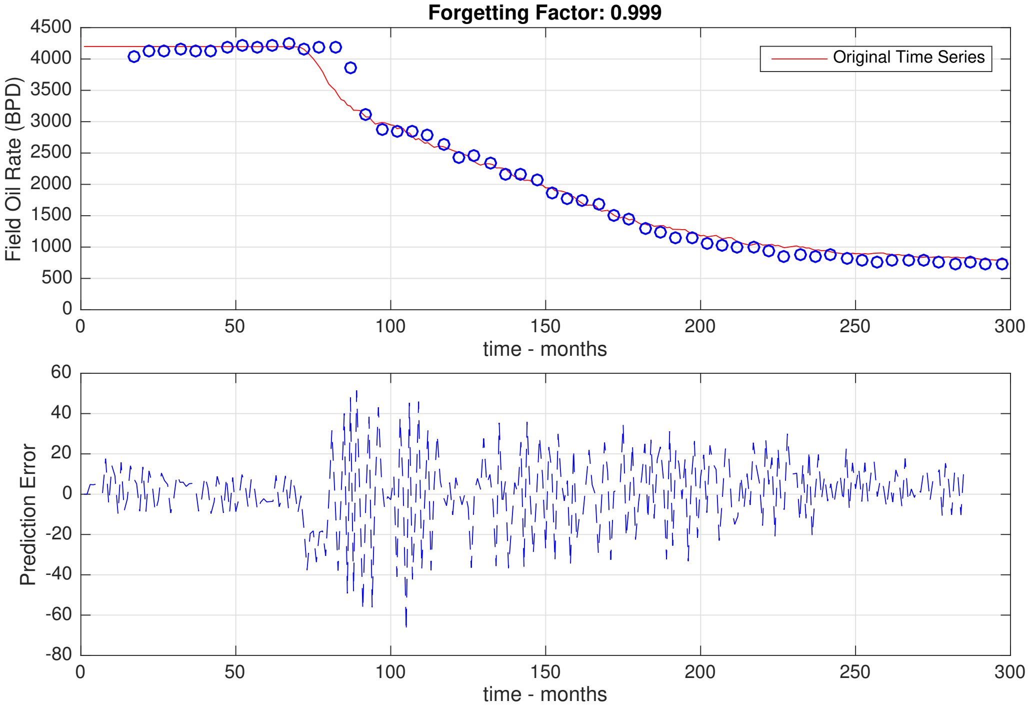

Figure 2.26: 12 months ahead prediction of the Field Oil Rate considering Initial Condition from data assimilation of a prior knowledge data set, as depicted in 2.24.

Chapter 3

Proxy States Based Model

While the previous chapter dealt with input-ouput models, this chapter will consider state space models. In the reservoir problem, states are not effectively measured and although they represent an abstraction from modeling perspective, in fact, they are grid block properties which evolve during the simulation.

This dissertation deals with a two phase simulation, whose states are water saturation and block pressure. The objective of this chapter is to explore the method of snapshots in order to create a proxy model by using principal component analysis, which aims to reduce dimensionality such that system identification can be performed with more ease.

To a certain extent, unlike linear transfer function representations, proxy states based models represent physical aspects, such as geometry and geology characteristics, by learning from a simulated model and its main features.

Furthermore, by making use of the linear state space proxy model obtained from system identification procedures, we aim to perform optimization such that the NPV (net present value) is maximized.

3.1 Method of snapshots

PINNAU [21] presents the method of snapshots, which represents trajectories of a dynamical system at discrete time events. For time steps $t_1, t_2, ..., t_m$, the snapshot matrix $X \in \mathbb{R}^{n \times m}$ is written as:

$$X = (x_1, x_2, ..., x_m),$$

where $x_i = x(t_i) \in \mathbb{R}^n$.

State dimension n is expected to be larger than the number of snapshots m. Reservoir dynamics present slow evolution, which does not require snapshots at high frequency, whereas n can be in the order of millions.

Proper orthogonal decomposition (POD) solves the problem of finding an orthogonal basis which contain as much information (or energy) as possible. In a simple case, where POD basis dimension is one ($k = 1$), a direction w which retains most information can be calculated by solving a constrained optimization problem:

$$\max_{w} \sum_{j=1}^{m} (x_j^T w)^2$$

subject to $w^T w = 1$.

Defining covariance matrix $S = \dfrac{\sum_j x_j x_j^T}{n-1}$, assuming x is centered at zero, the optimization problem can be stated as:

$$\max_{w} w^T S w$$

subject to $w^T w = 1$.

Lagrangian is $L(w, \lambda) = w^T S w + \lambda(1 - w^T w)$.

The gradient with respect to w is

$$\nabla_w(L) = 2Sw - 2\lambda w = 0.$$

Therefore $Sw = \lambda w$, which means w is an eigenvector of the covariance matrix S. The projected variance is $w^T S w = w^T \lambda w$ and its maximum value corresponds to the largest eigenvalue. As a consequence, w is the eigenvector corresponding to the largest eigenvalue of S.

Moreover, w is the eigenvector corresponding to the greatest right-singular value of X. By using SVD factorization,

$$X = U \Sigma V^T,$$

covariance matrix eigenvectors correspond to the columns of U.

For POD-basis with $k > 1$, we iterate this procedure and derive that $w_i (i = 2, ..., k)$ solves

$$\max_{w_i} \sum_{j=1}^{m} (x_j^T w_i)^2$$

subject to $w_i^T w_i = 1$ and $w_i^T w_s = 0$ for all $s \leq i$.

Therefore POD-basis with dimension k corresponds to the first k columns of matrix U, from the singular value decomposition of X. Denoting the reduced matrix $U_r = U[:, 1:k]$ in Matlab notation, a vector z in a reduced dimension is calculated projecting the array x onto the column space of U_r:

$$z = U_r^T x$$

and x can be reconstructed as a linear combination of the POD-basis as follows:

$$\widetilde{x} = U_r z.$$

3.2 Relation between Regularization and PCA

This section briefly recalls the connection between dimensionality reduction using principal component analysis and its regularization effect.

The least squares solution, with regularization parameter ($\alpha > 0$), for the linear system $Ax = b$ (with no assumptions on A) is:

$(\alpha I + A^T A)\hat{x} = A^T b$.

Using singular value decomposition, $A = U\Sigma V^T$.

$(\alpha I + V\Sigma^T U^T U\Sigma V^T)\hat{x} = V\Sigma^T U^T b$.

$V(\alpha I + \Sigma^T\Sigma)V^T\hat{x} = V\Sigma^T U^T b$.

$V^T V(\alpha I + \Sigma^T\Sigma)V^T\hat{x} = V^T V\Sigma^T U^T b$.

$(\alpha I + \Sigma^T\Sigma)V^T\hat{x} = \Sigma^T U^T b$.

Since $(\alpha I + \Sigma^T\Sigma)$ is a diagonal matrix, and α is positive,

$$\hat{x} = V(\alpha I + \Sigma^T\Sigma)^{-1}\Sigma^T U^T b = \sum_{j=1}^{N_r} v_j \frac{\sigma_j u_j^T b}{\sigma_j^2 + \alpha},$$

where N_r is the rank of A. Observing that

$$v_j \frac{\sigma_j u_j^T b}{\sigma_j^2 + \alpha} \approx \begin{cases} 0, & \text{if } 0 \approx \sigma_j \ll \alpha \\ \frac{v_j u_j^T b}{\sigma_j}, & \sigma_j \gg \alpha \end{cases} \tag{3.1}$$

It can be seen that the regularization parameter α has, in common with PCA rank reduction, a similar effect of removing the smaller singular values. On the other hand, larger singular values are left almost unchanged.

Intuitively speaking, this is why principal component analysis plays an important role in preventing overfitting. In other words, PCA dimensionality gets rid of less informative dimensions in order to help generalization.

In order to retain a percentage ϵ of the variance (or energy), the typical criterion is to keep k largest singular values, such that

$$\frac{\sum_{j=1}^{k} \sigma_j}{\sum_{j=1}^{N_r} \sigma_j} \leq \epsilon \leq 1. \tag{3.2}$$

3.3 State Matrix Estimation

AGUIRRE [10] presents matrix estimation procedure of a discrete linear model represented as:

$$x(k) = Ax(k-1) + Bu(k) + e(k) \tag{3.3}$$

where $x(k) = [x_1(k)x_2(k)...x_n(k)]^T$ is the vector that represents state at time step k, $u(k) = [u_1(k)u_2(k)...u_r(k)]^T$ is the vector that represents controls at time step k and $e(k) = [e_1(k)e_2(k)...e_n(k)]^T$ are the regression error for each state at time step k.

This linear system can also be written as:

$x(k) = \begin{bmatrix} A & B \end{bmatrix} \begin{pmatrix} x(k-1) \\ u(k-1) \end{pmatrix} = \gamma^T m^T(k-1)$

$x(k)^T = m(k-1)\gamma = \begin{bmatrix} x(k-1)^T & u(k-1)^T \end{bmatrix} \begin{pmatrix} A^T \\ B^T \end{pmatrix}$

For N time steps:

$x_N = \begin{pmatrix} x(1)^T \\ x(2)^T \\ ... \\ x(N)^T \end{pmatrix} = \begin{pmatrix} m(0) \\ m(1) \\ ... \\ m(N-1) \end{pmatrix} \gamma = M\gamma$

$\hat{\gamma} = \begin{pmatrix} \hat{A}^T \\ \hat{B}^T \end{pmatrix}$ is estimated by least squares:

$$\gamma \approx M^{\dagger} x_N,$$

where † denotes pseudo inverse solution.

Now we can combine dimensionality reduction with this linear state space identification procedure such that the matrices A and B are estimated using the vector $z(k) = U_r^T x(k)$ instead of $x(k)$. The procedure is as follows:

1) Write snapshot matrix and perform principal component analysis: $z = U_r^T x$.

2) Estimate matrices A and B, such that

$$z(k) = Az(k-1) + Bu(k-1) \tag{3.4}$$

3) Reconstruct the estimate of the original vector $\widetilde{x} = U_r z$.

Let N_x the dimension of state space, N_z the dimension of reduced space, N_u the dimension of control space.

For N time steps, $O = \begin{pmatrix} x(1)^T \\ x(2)^T \\ ... \\ x(N)^T \end{pmatrix}$ represents measured data, whose length is

N_xN. In original space, matrix A has N_x^2 parameters and B has N_xN_u. That is, number of parameters to be estimated is $N_x(N_x + N_u)$.

Least Squares estimation would be appropriate to obtain a good fit if $N_x(N_x+N_u)$ is significantly smaller than N_xN. However, the snapshot matrix X tends to be tall and thin, which means there are many more states associated with every grid block than meaningful snapshots.

Therefore, dimensionality reduction plays a key role in preventing overfitting in the matrix estimation problem.

Moreover, computational aspects play a critical role. It could be computationally very expensive to estimate matrices A and B in the original space. For instance, if $N_x = 10^6$, there are more than 10^{12} parameters to be estimated, apart from the fact that such a structure would be prone to overfitting.

3.3.1 Output Identification

So far, we have presented a procedure by which it is possible to estimate state variables, such as water saturation and block pressure. However, these variables are not measured and they represent an abstraction that comes from a simulation model. What we really measure can be defined as output variables, such as oil, water, gas rates and bottom hole pressure.

This section aims to describe a linear representation of outputs as a function of reduced states and controls. This kind of representation runs the risk of being an oversimplification because well flow models may be strongly non linear.

For instance, water cut in the “Two-phase-flow model” depends on relative permeability curve, which itself can be modeled by a non linear function of water saturation. Moreover, depending on how many perforations exist in the well, it might be necessary to consider pressure drop between perforations, which could make modeling more complicated.

Despite all this, as a first approximation, we will attempt to describe the output as a linear function of state and input, i.e.:

$$y(k) = Cz(k) + Du(k). \tag{3.5}$$

The advantage of modeling the dynamical system as a linear system is the fact that this will turn the optimization problem a LP (linear programming) problem, as described in section 3.4. If we were to consider rigorously all complex nonlinear modeling, it would not be possible to take advantage of efficient solvers for convex problems as we do in the LP problem.

In other words, this is a linear approximation in the neighborhood of a given control trajectory u. Similarly to what was described in Section 3.3, we aim to find matrices C and D such that quadratic error is minimized.

For N time steps:

$$\begin{pmatrix} y(1)^T \\ y(2)^T \\ ... \\ y(N)^T \end{pmatrix} = \begin{pmatrix} z(1)^T u(1)^T \\ z(2)^T u(2)^T \\ ... \\ z(N)^T u(N)^T \end{pmatrix} \begin{pmatrix} C^T \\ D^T \end{pmatrix}.$$

The least squares solution is $\begin{pmatrix} \hat{C}^T \\ \hat{D}^T \end{pmatrix} = \begin{pmatrix} z(1)^T u(1)^T \\ z(2)^T u(2)^T \\ ... \\ z(N)^T u(N)^T \end{pmatrix}^{\dagger} \begin{pmatrix} y(1)^T \\ y(2)^T \\ ... \\ y(N)^T \end{pmatrix}.$

3.4 Proxy based Optimization

In previous sections, we established a procedure which fits a linear system, that works as a proxy model around a specified trajectory control u.

This section reformulates the optimal control problem of maximizing the net present value (the performance index) at the end of a given time horizon, subject to the proxy model dynamics and additional constraints on the inputs, as a mathematical programming problem. This is done by treating the net present value as the objective function to be maximized, subject to the discrete time proxy model dynamics, rewritten as a sequence of equality constraints over the entire horizon. Since the net present value is a linear function of the outputs, the proxy dynamics are chosen as linear and the input constraints are all linear, the mathematical programming problem is actually a linear programming problem, which confers computational efficiency and scalability to the reformulation.

Given a reference trajectory control $\bar{u}$, around which a linear system was identified, the aim is to find an optimal trajectory control u around $\bar{u}$ such that NPV is maximized. In order to state the problem, parameters and variables in the optimization problem are first defined. The parameters in this problem are as follows:

j is the interest rate.

O_p is the oil price (dollar per barrel).

W_c is the water treatment cost (dollar per barrel).

I_c is the water injection cost (dollar per barrel).

$Q_{\max}$ represents the total liquid capacity constraint.

$u_{\max}$ and $u_{\min}$ are operational bounds for the trajectory controls.

Δ_u represents the box constraint in which the proxy model could be valid around $\bar{u}$.

s_r is the slew rate, which imposes softness in the trajectory control.

z_0 is the initial condition in the reduced space after dimensionality reduction procedure.

The variables u_p and u_i corresponds to the controls in producers and injectors, respectively. In this work, producers controls are liquid rate and injectors controls are water injection rates.

The other variables in this problem are:

$Q_o(t)$ is the oil rate vector for all producer wells at time step t.

$Q_{o,T}(t)$ is the total oil production at time step t, defined in equation (3.9).

$Q_{liq,T}(t)$ is the total liquid production at time step t, defined in equation (3.10).

$Q_{w,T}(t)$ is the total water production at time step t, defined in equation (3.12).

$Q_{inj,T}(t)$ is the total injection rate at time step t, defined in equation (3.13).

The corresponding linear programming problem is formulated as follows:

$$
\begin{aligned}
\max_{u_p,u_i} \quad & NPV = \sum_{t=1}^{T} \frac{Q_{o,T}(t)O_p - Q_{w,T}(t)W_c - Q_{inj,T}(t)I_c}{(1+j)^t} && (3.6)\\
\text{s.t.} \quad & z(t+1) = Az(t) + Bu(t+1) \quad t = 0,\dots,T-1 && (3.7)\\
& Q_o(t) = Cz(t) + Du(t) \quad t = 1,\dots,T && (3.8)\\
& Q_{o,T}(t) = 1^T Q_o(t) \quad t = 1,\dots,T && (3.9)\\
& Q_{liq,T}(t) = 1^T u_p(t) \quad t = 1,\dots,T && (3.10)\\
& u(t)^T = [u_p(t)^T u_i(t)^T] \quad t = 1,\dots,T && (3.11)\\
& Q_{w,T}(t) = Q_{liq,T}(t) - Q_{o,T}(t) \quad t = 1,\dots,T && (3.12)\\
& Q_{inj,T}(t) = 1^T u_i(t) \quad t = 1,\dots,T && (3.13)\\
& Q_{liq,T}(t) \le Q_{\max} \quad t = 1,\dots,T && (3.14)\\
& u(t) - \bar{u}(t) \le \Delta_u \quad t = 1,\dots,T && (3.15)\\
& u(t) - \bar{u}(t) \ge -\Delta_u \quad t = 1,\dots,T && (3.16)\\
& u_{\min} \le u(t) \le u_{\max} \quad t = 1,\dots,T && (3.17)\\
& u(t+1) - u(t) \ge -s_r \quad t = 0,\dots,T-1 && (3.18)\\
& u(t+1) - u(t) \le s_r \quad t = 0,\dots,T-1 && (3.19)\\
& z(0) = z_0, && (3.20)
\end{aligned}
$$

This linear program as well as the decision time horizon will both scale linearly with the number of wells being considered. Even so, this does not represent a big challenge in terms of computing time because modern LP software easily handles very large problems (millions of variables and constraints). In this dissertation, which deals with a toy proof of concept problem, we solve it by using cvxpy [15], a mathematical modeling language available in the Python environment for convex problems.

Trust Region Method Due to the nonlinear nature of this optimization problem, trust region methods (see CONN *et al.* [13]) represent a suitable technique in which the proxy model from system identification can be used in optimization framework.

Algorithm 1 describes the optimization procedure. Given an initial condition $(\bar{u}, \Delta_u)$, a linear system is identified in this region and used as a proxy model for a LP problem. This identification procedure, described in line 4 of algorithm 1, takes into account PRBS (pseudo random binary sequence) excitation in the region $(\bar{u}, \Delta_u)$.

If the optimization result is consistent with simulation in terms of objective function increase, the optimal control calculation is the new current point and, depending on how well the proxy model can describe the system, the trust region confidence region (Δ_u) may increase or decrease.

The proxy model may lack consistency, which means the simulation does not show an increase in the objective function, whereas the proxy model says so. In this case, $\rho \leq 0$ and the algorithm 1 performs system identification in order to find another suitable linear proxy model, more likely to represent the simulation around the current trajectory control.

Finally, the algorithm stops if the maximum iteration is reached or the trust region reduces to a region smaller than Δ_{min}, which means the algorithm converged towards an optimal solution.

3.5 Results

3.5.1 Assessment of Proxy Model

The objective of this section is to assess whether a surrogate (aka proxy) linear model is suitable for optimization purposes.

Linear Model Identification Given an initial trajectory control $\bar{u}$, we will perform system identification by using PRBS excitation in a region Δ_u, using the reservoir model described in 2.4. More specifically, we will derive a linear dynamical system, as described in Section 3.3.

Figure 3.1 depicts the PRBS excitation around the trajectory control

$$\bar{u}(k) = [600, 2500, 2500, 2500, 2500, 750, 5675, 5675]^T$$

with

$$\Delta_u(k) = [50, 50, 50, 50, 50, 50, 50, 50]^T$$

Algorithm 1 Adapted Trust Region algorithm

```
1: procedure TRUST REGION($\bar{u}$, $\Delta_u$)    ▷ initial condition
2:     $iter \leftarrow 0$    ▷ iteration count initialization
3:     Evaluate $f(\bar{u})$    ▷ by using the simulation model
4:     Identify the linear model in the region $(\bar{u}, \Delta_u)$    ▷ as shown in Section 3.3
5:     Evaluate linear proxy objective function $\hat{f}(\bar{u})$
6:     Calculate $u^*$ by solving the optimization problem in the region $(\bar{u}, \Delta_u)$    ▷ as shown in Section 3.4
7:     Evaluate $f(u^*)$    ▷ by using the simulation model
8:     Evaluate linear proxy objective function $\hat{f}(u^*)$
9:     Calculate $\rho = \frac{f(u^*)-f(\bar{u})}{\hat{f}(u^*)-\hat{f}(\bar{u})}$
10:    if $\rho \leq 0$ then    ▷ proxy is not consistent
11:        $\Delta_u \leftarrow 0.25\Delta_u$
12:        go to line 4
13:    end if
14:    if $0 \leq \rho \leq 0.25$ then    ▷ we no longer trust the proxy model as before
15:        $\Delta_u \leftarrow 0.25\Delta_u$
16:        $\bar{u} \leftarrow u^*$
17:        $f(\bar{u}) \leftarrow f(u^*)$
18:    end if
19:    if $0.25 \leq \rho \leq 0.75$ then
20:        $\Delta_u \leftarrow \Delta_u$
21:        $\bar{u} \leftarrow u^*$
22:        $f(\bar{u}) \leftarrow f(u^*)$
23:    end if
24:    if $\rho \geq 0.75$ then
25:        $\Delta_u \leftarrow 2\Delta_u$
26:        $\bar{u} \leftarrow u^*$
27:        $f(\bar{u}) \leftarrow f(u^*)$
28:    end if
29:    $iter \leftarrow iter + 1$
30:    if $\Delta_u \leq \Delta_{min}$ or $iter > N_{max}$ then
31:        break    ▷ stop criterion
32:    end if
33:    go to line 5
34: end procedure
```

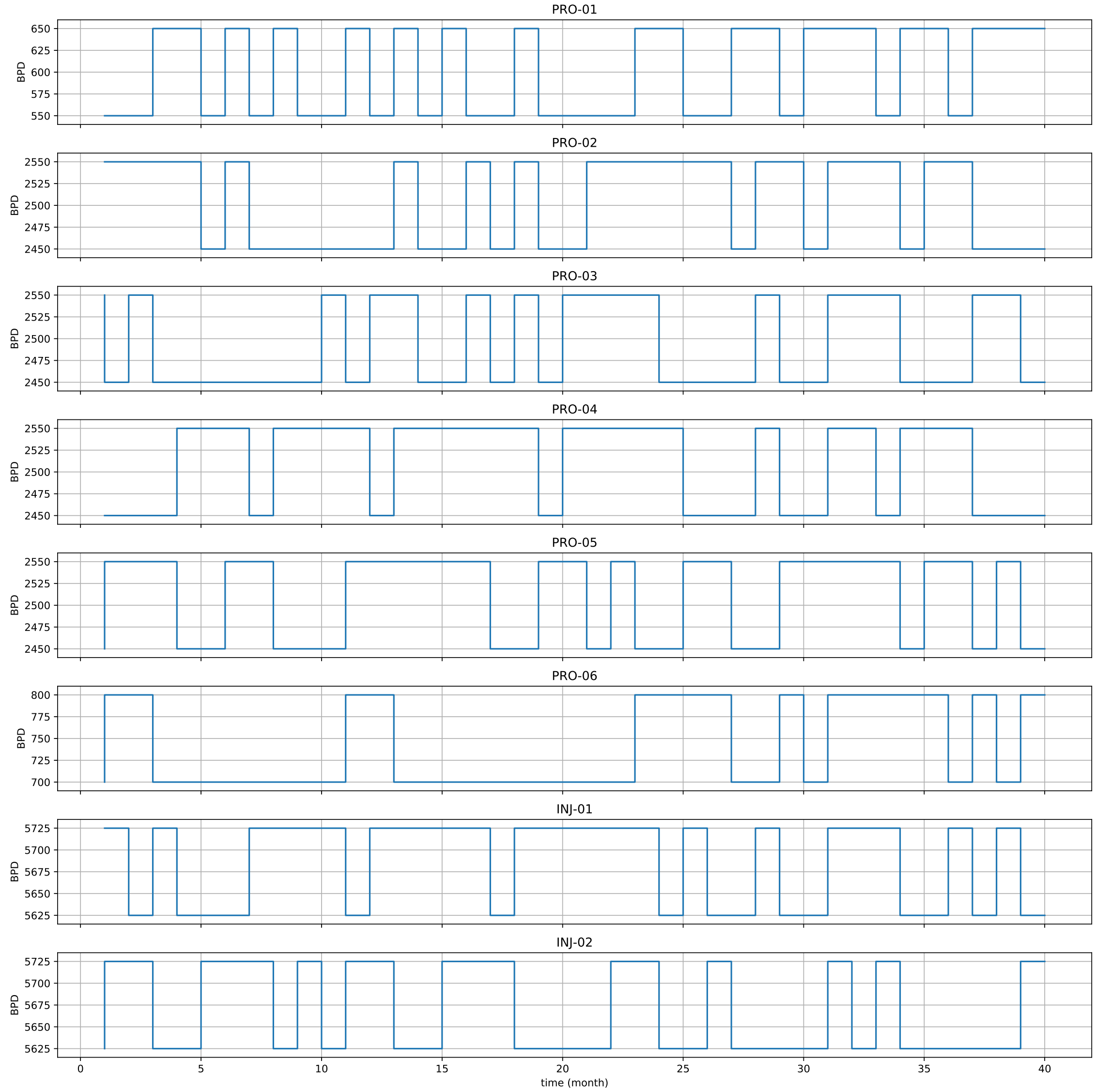

Figure 3.1: Pseudo Random Binary Sequence (PRBS) Excitation considering both producer and injector wells. This graph represents LPR (liquid production rate) for all producer wells (from PRO-01 to PRO-06) and WIR (water injection rate) for all injector wells (INJ-01 and INJ-02).

for all $k \in \{1, ..., 40\}$.

Figure 3.2 shows the singular values in log scale, from the greatest to the smallest, in both water cut and block pressure variables. Equation 3.2 provides a criterion for energy retention: $\epsilon = 99.999\%$ is used for water and $\epsilon = 99.9999\%$ for block pressure.

The snapshot matrix has 450 lines and 41 columns, which means that its rank is at most 40 because the columns are centered at the origin. In both cases, because the first 40 columns of the snapshot matrix are linearly independent, its rank is 40. Dimensionality was reduced by PCA from 450 to 19 for water saturation and to 29 for block pressure.

For the dynamical system described in equation 3.4, we used water saturation map after dimensionality reduction. For output equation 3.5, oil rate for each producer well was used as output variable.

Proxy Results We aim to evaluate how well the trajectory control derived from LP problem described in (3.6) - (3.20) matches simulation data for different levels of box constraints for the previous linear model. We will compare water saturation and block pressure maps and oil rates results.

Box constraint, defined in equations 3.15 and 3.16, delimit the feasible region for the LP problem, described in 3.4. In this case study, we will define 6 different box constraints:

$$\begin{aligned}
\Delta_u^0 &= [0, 0, 0, 0, 0, 0, 0, 0], \\
\Delta_u^{25} &= [25, 25, 25, 25, 25, 25, 25, 25], \\
\Delta_u^{50} &= [50, 50, 50, 50, 50, 50, 50, 50], \\
\Delta_u^{100} &= [100, 100, 100, 100, 100, 100, 100, 100], \\
\Delta_u^{200} &= [200, 200, 200, 200, 200, 200, 200, 200], \\
\Delta_u^{300} &= [300, 300, 300, 300, 300, 300, 300, 300].
\end{aligned}$$

For each case, a different trajectory control u is derived from the LP problem, and as Δ_u increases, we may expect that u becomes more distant from $\bar{u}$. As a consequence, we would expect that the proxy model deteriorates its capability of prediction, given that the proxy model was trained in a region around $\bar{u}$.

This effect can be seen in water saturation maps. From figure 3.3 to figure 3.6, it is possible to notice how the proxy prediction quality worsens as Δ_u increases and as the proxy dynamical system evolves throughout time. Despite this, it is evident that the proxy model can capture acceptably the geometry and spatial distribution of the variables in the reservoir.

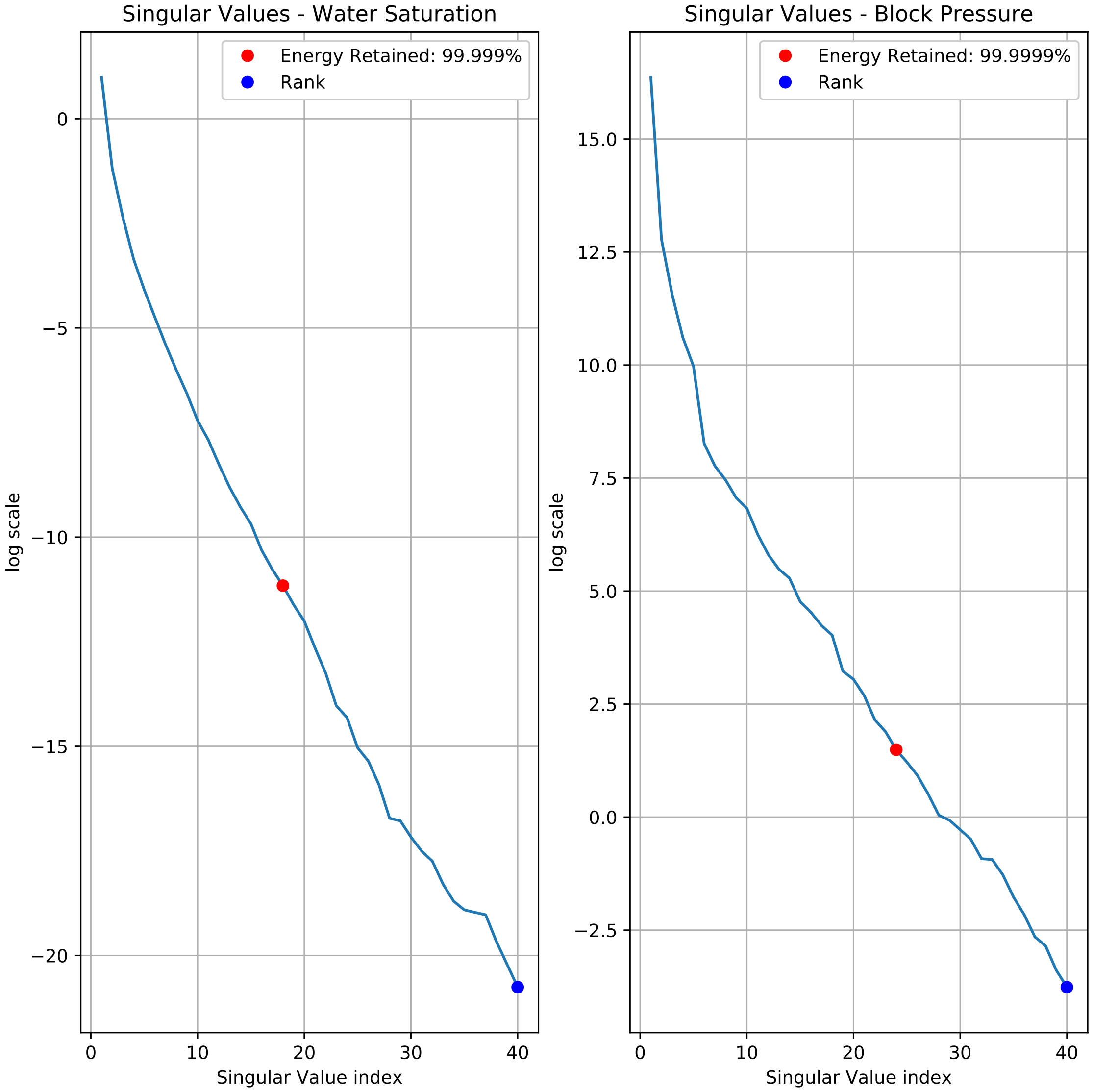

Figure 3.2: Singular Value Decomposition - from the greatest singular value to the smallest. The dimensionality was reduced from 450 to 19 for water saturation and to 29 for block pressure, which simplifies the identification problem and prevents from overfitting. The energy retained ϵ (which defines the dimension of the POD-basis) is chosen based on the maps reconstruction assessment for both water saturation and block pressure maps, as shown in Figures 3.3 - 3.9.

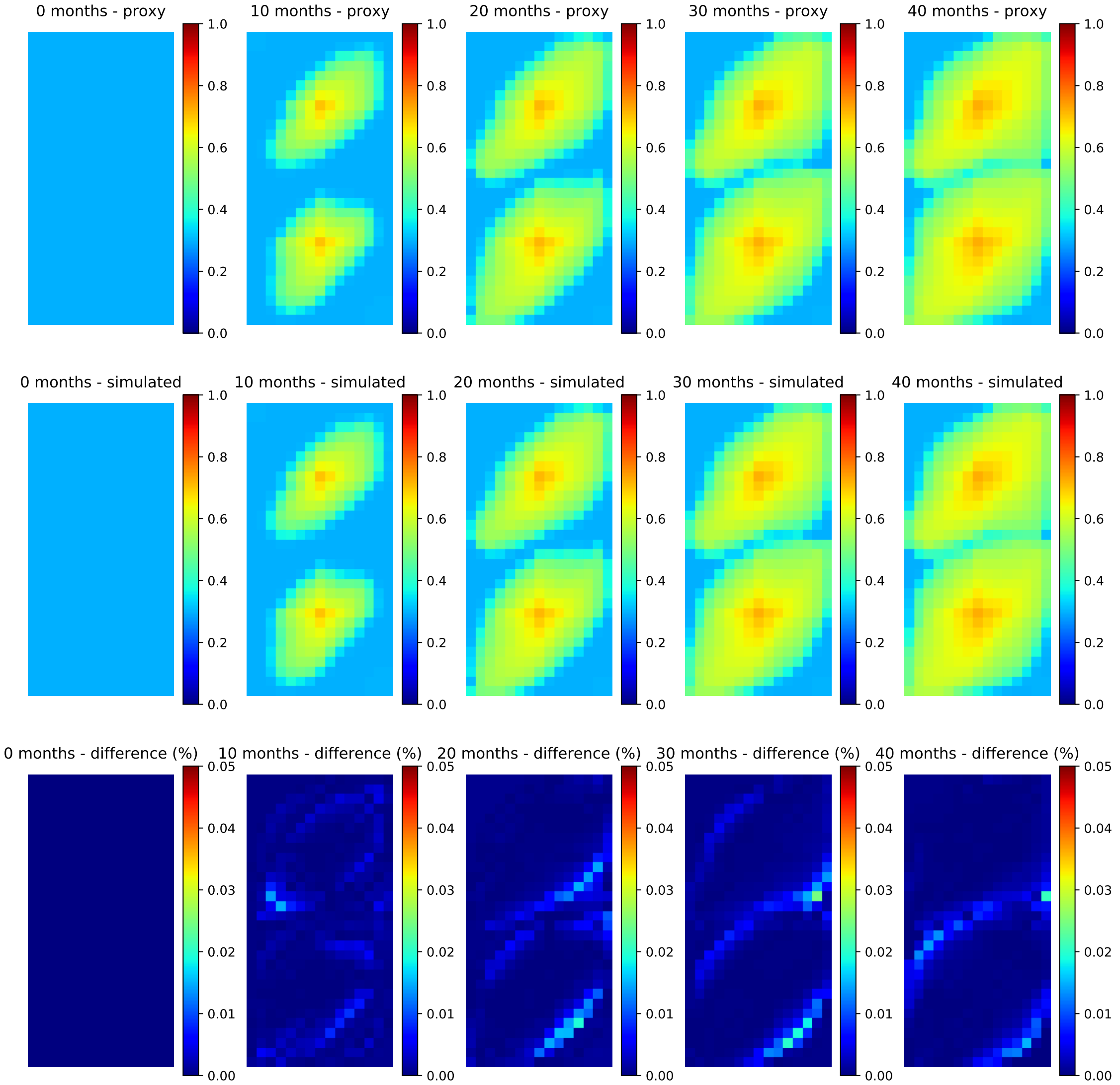

Figure 3.3: The first row shows water saturation evolution according to the proxy model, the second according to the simulated model and the last row shows the difference between the two, for the case in which the box constraint allows zero (Δ_u^0) deviation from reference trajectory.

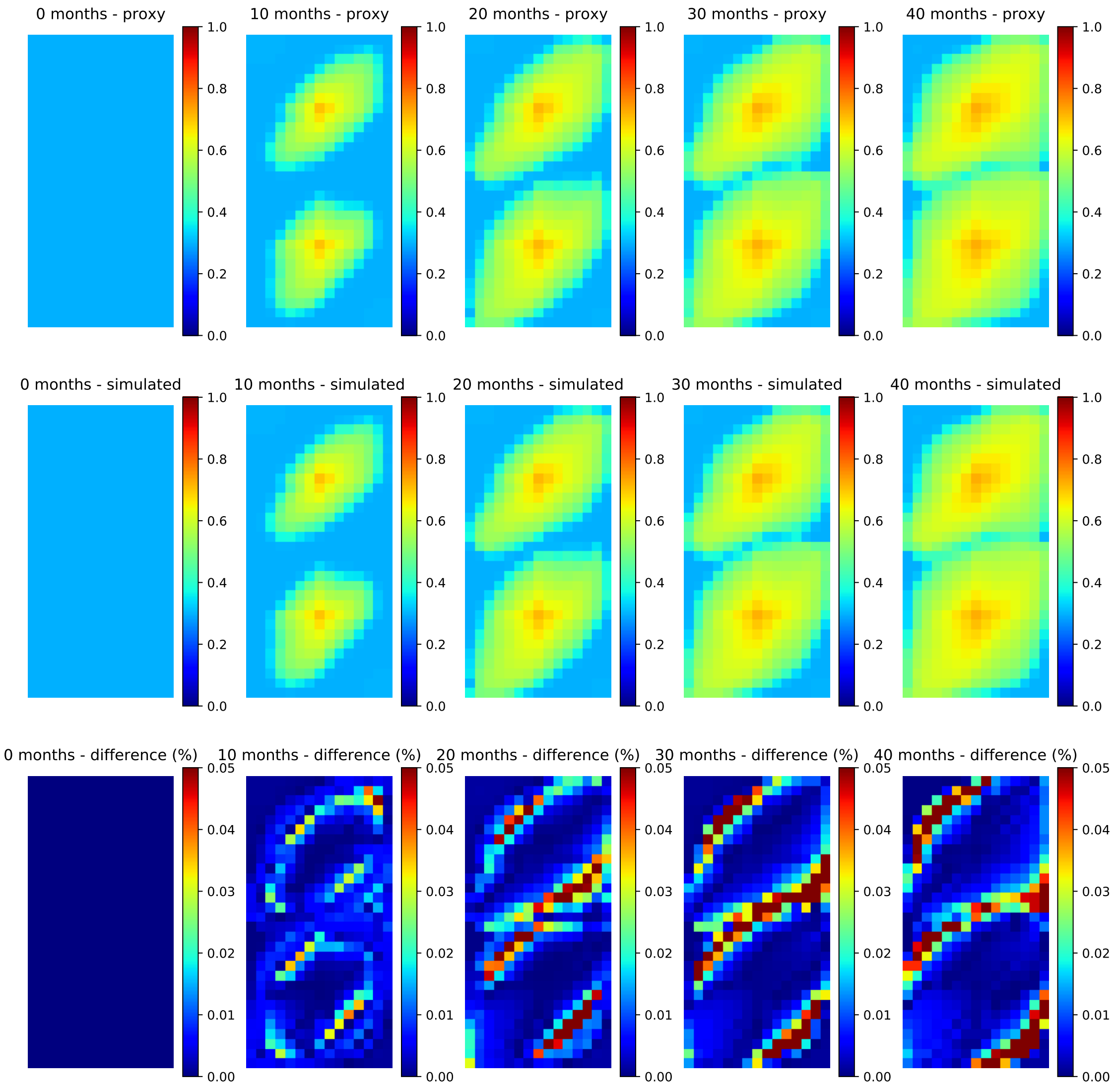

Figure 3.4: The first row shows water saturation evolution according to the proxy model, the second according to the simulated model and the last row shows the difference between the two, for the case in which the box constraint allows deviation ±100 barrels per day (Δ_u^{100}) from reference trajectory.

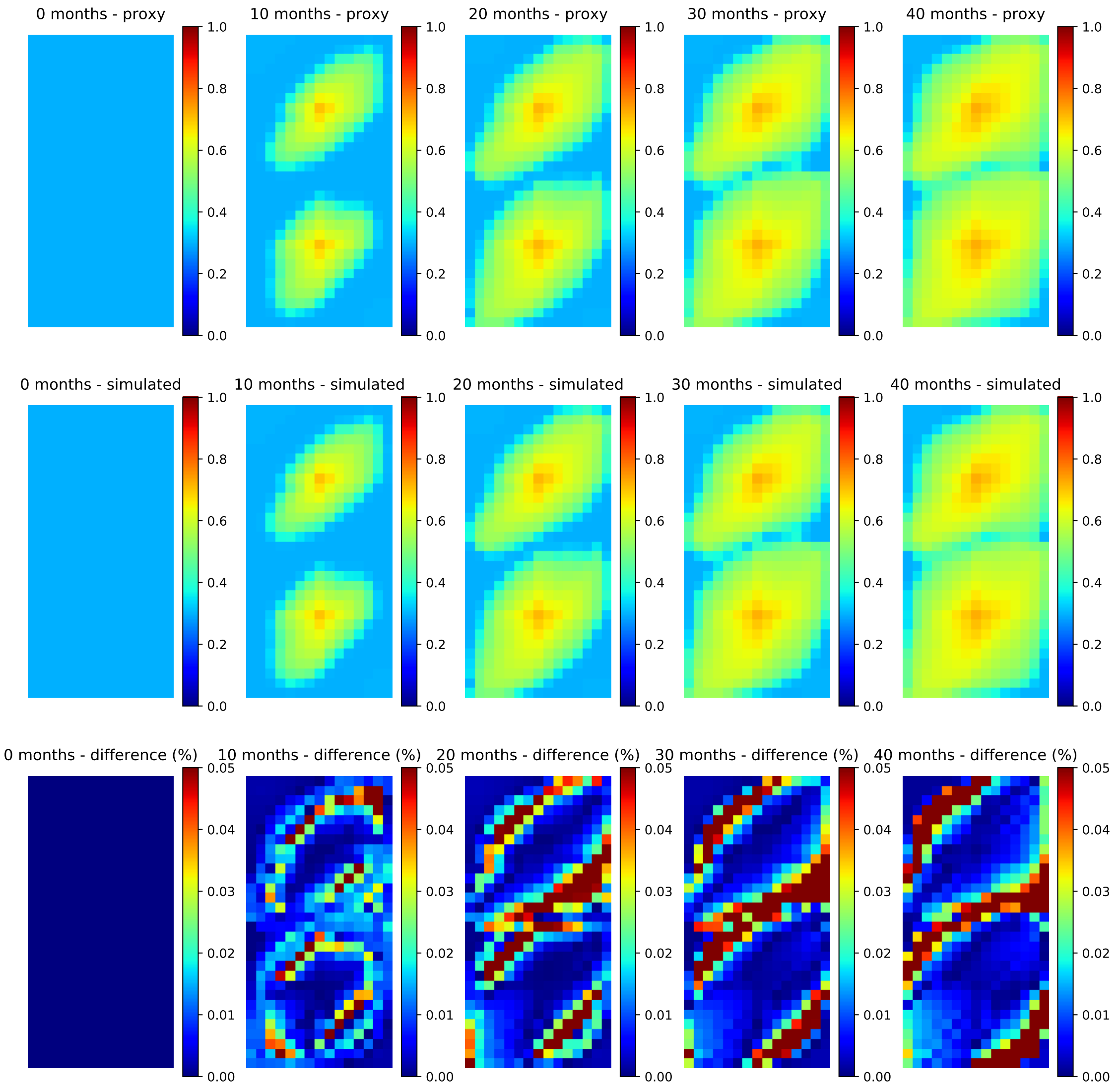

Figure 3.5: The first row shows water saturation evolution according to the proxy model, the second according to the simulated model and the last row shows the difference between the two, for the case in which the box constraint allows deviation ±200 barrels per day (Δ_u^{200}) from reference trajectory.

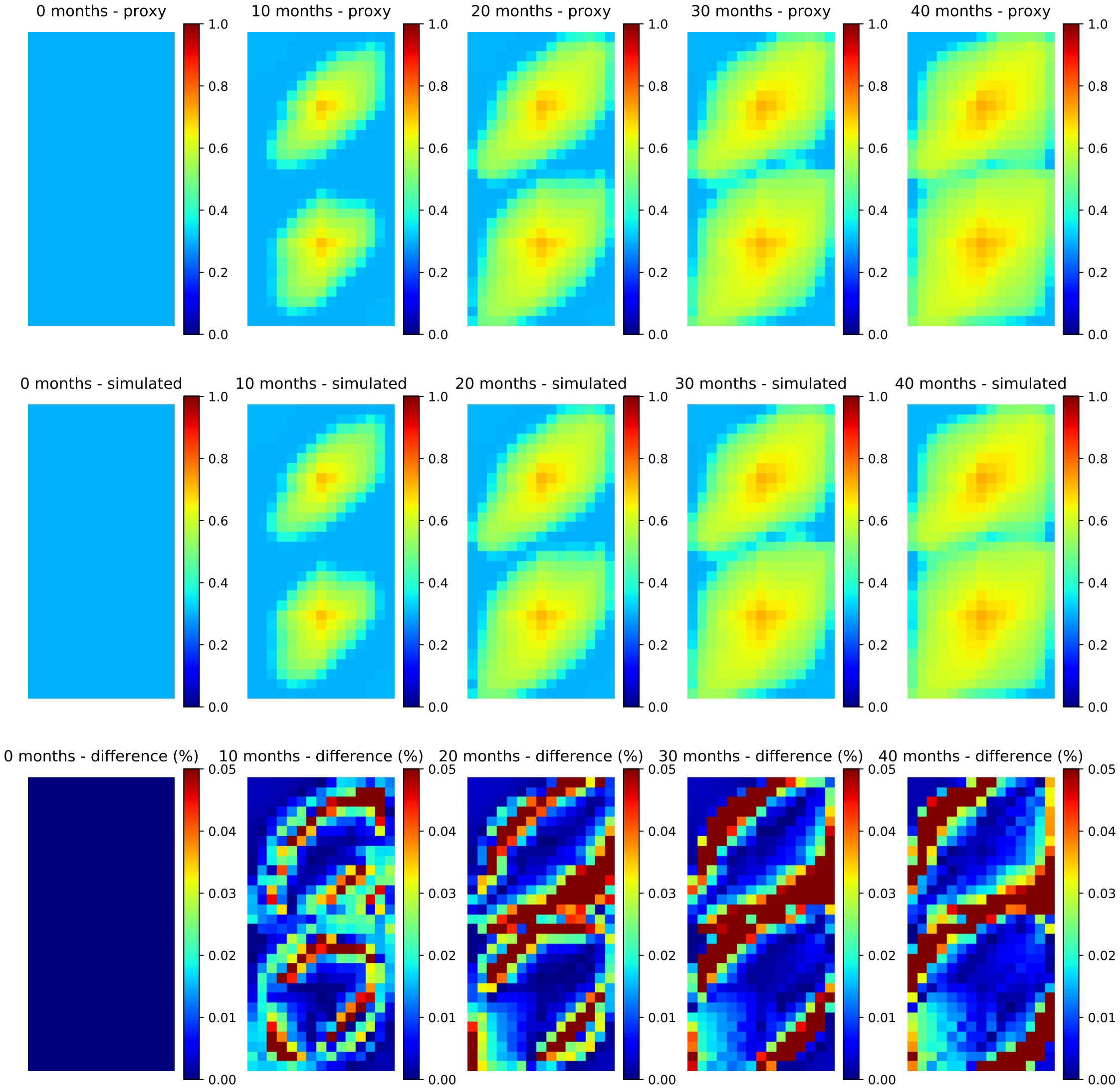

Figure 3.6: The first row shows water saturation evolution according to the proxy model, the second according to the simulated model and the last row shows the difference between the two, for the case in which the box constraint allows deviation ±300 barrels per day (Δ_u^{300}) from reference trajectory.

In figures 3.7 to figure 3.9, similar phenomena are observed for the block pressure variable. It is worth pointing out that the block pressure variable presents more difficulty to be explained by a linear model compared to water saturation variable, and requires more energy to be retained. In fact, the block pressure map presents bigger relative errors compared to water saturation ones. This may be due to the fact that the total compressibility of the reservoir system is quite small, which causes larger changes in dynamics in response to pressure changes.

Even though the proxy model was trained centered at $\bar{u}$ with amplitude $\Delta_u = [50, 50, 50, 50, 50, 50, 50, 50]$, as depicted in figure 3.1, for Δ_u^{100}, Δ_u^{200}, Δ_u^{300}, the results shown in figures 3.4, 3.5 and 3.6 for water saturation maps suggests good capability of generalization. Unlike water saturation maps, block pressure maps do not perform so well for deviations from the reference control trajectory larger than ±50 barrels per day.

Figure 3.10 represents the case where $u = \bar{u}$. In fact, this is the case where the proxy model presents the best match with simulation results in terms of output results, whereas from figure 3.11 to 3.13, we observe a slight deterioration in the proxy performance.

Even though a linear proxy model is not completely able to represent all complexity of the reservoir simulation, it is consistently able to indicate how the dynamical system evolves. The Trust Region Algorithm 1 makes use of this capability in order to handle the non linear problem as a sequence of linear problems.

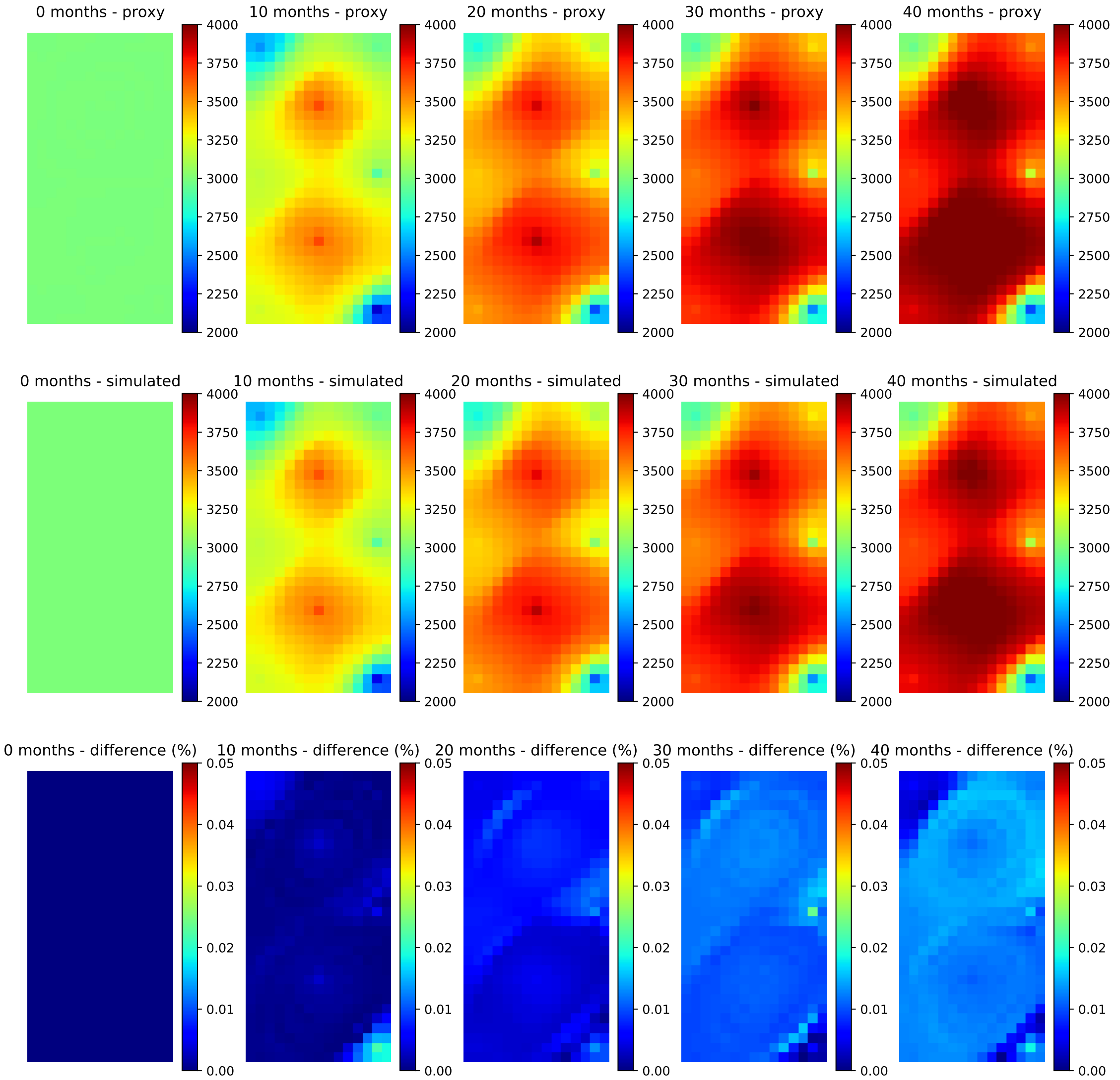

Figure 3.7: The first row shows block pressure (in psi) evolution according to the proxy model, the second according to the simulated model and the last row shows the difference between the two, for the case in which the box constraint allows zero deviation (Δ_u^0) from reference trajectory.

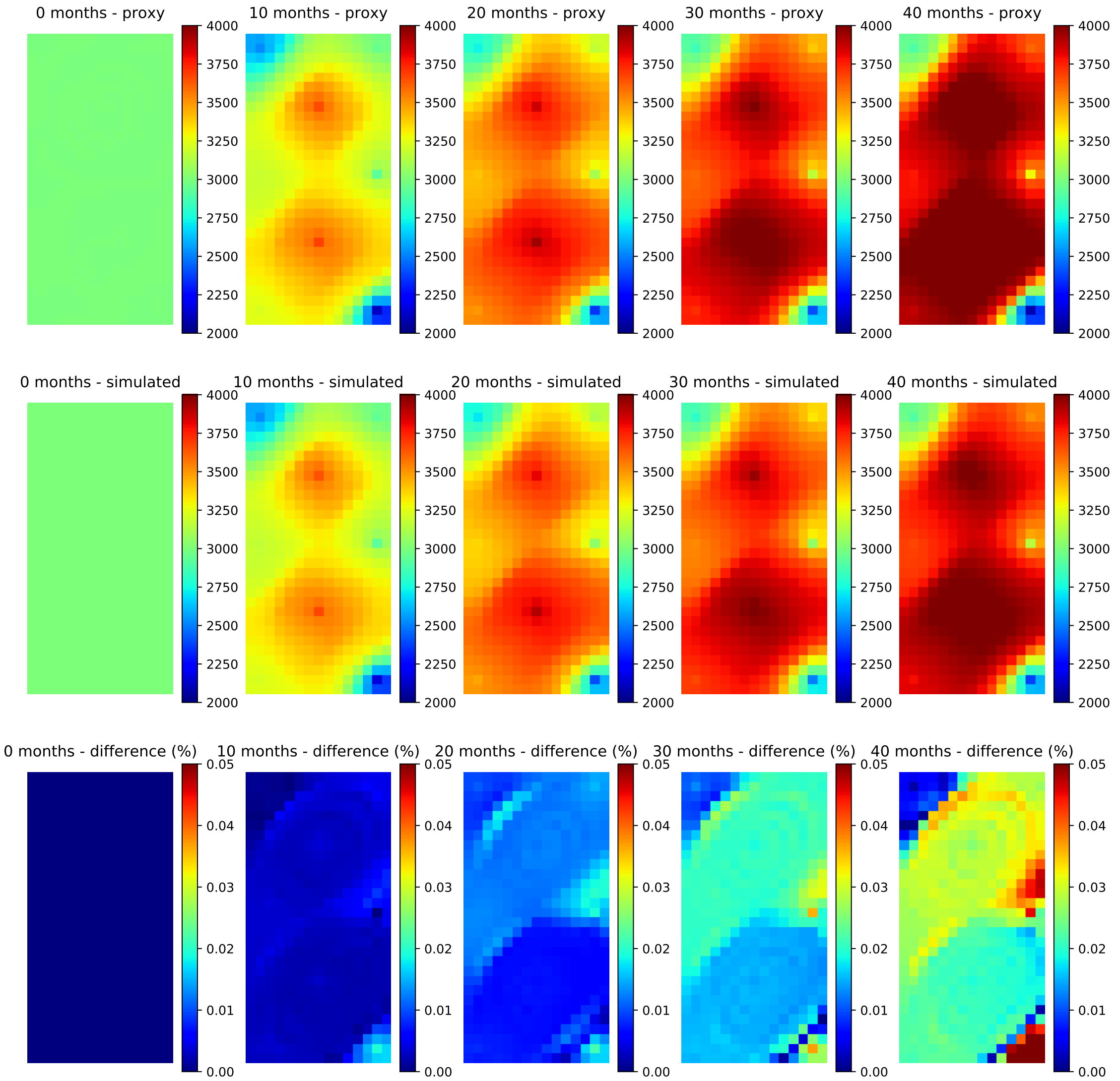

Figure 3.8: The first row shows block pressure (in psi) evolution according to the proxy model, the second according to the simulated model and the last row shows the difference between the two, for the case in which the box constraint allows a deviation of ±25 barrels per day (Δ_u^{25}) from reference trajectory.

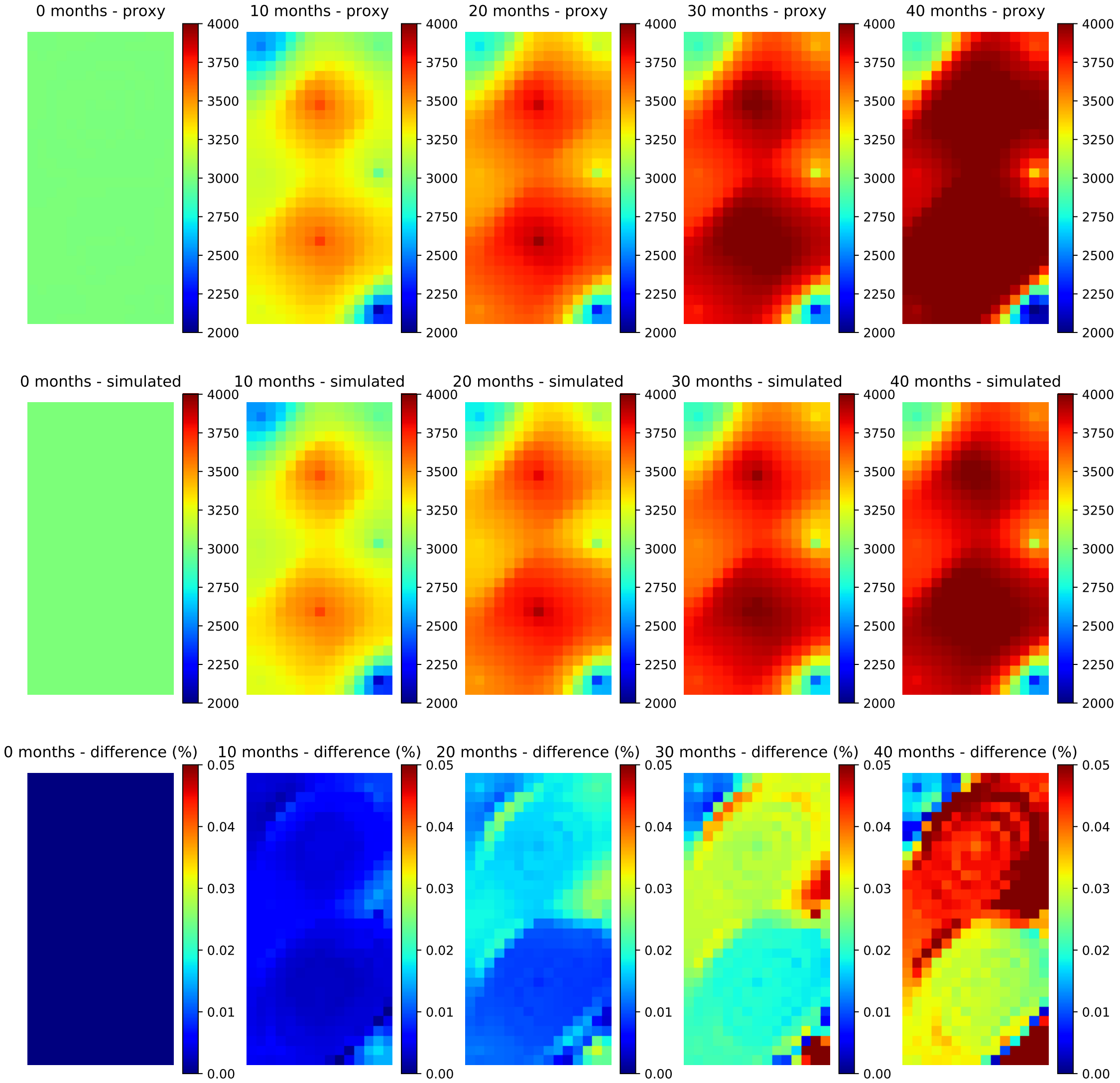

Figure 3.9: The first row shows block pressure (in psi) evolution according to the proxy model, the second according to the simulated model and the last row shows the difference between the two, for the case in which the box constraint allows a deviation of ±50 barrels per day (Δ_u^{50}) from reference trajectory.

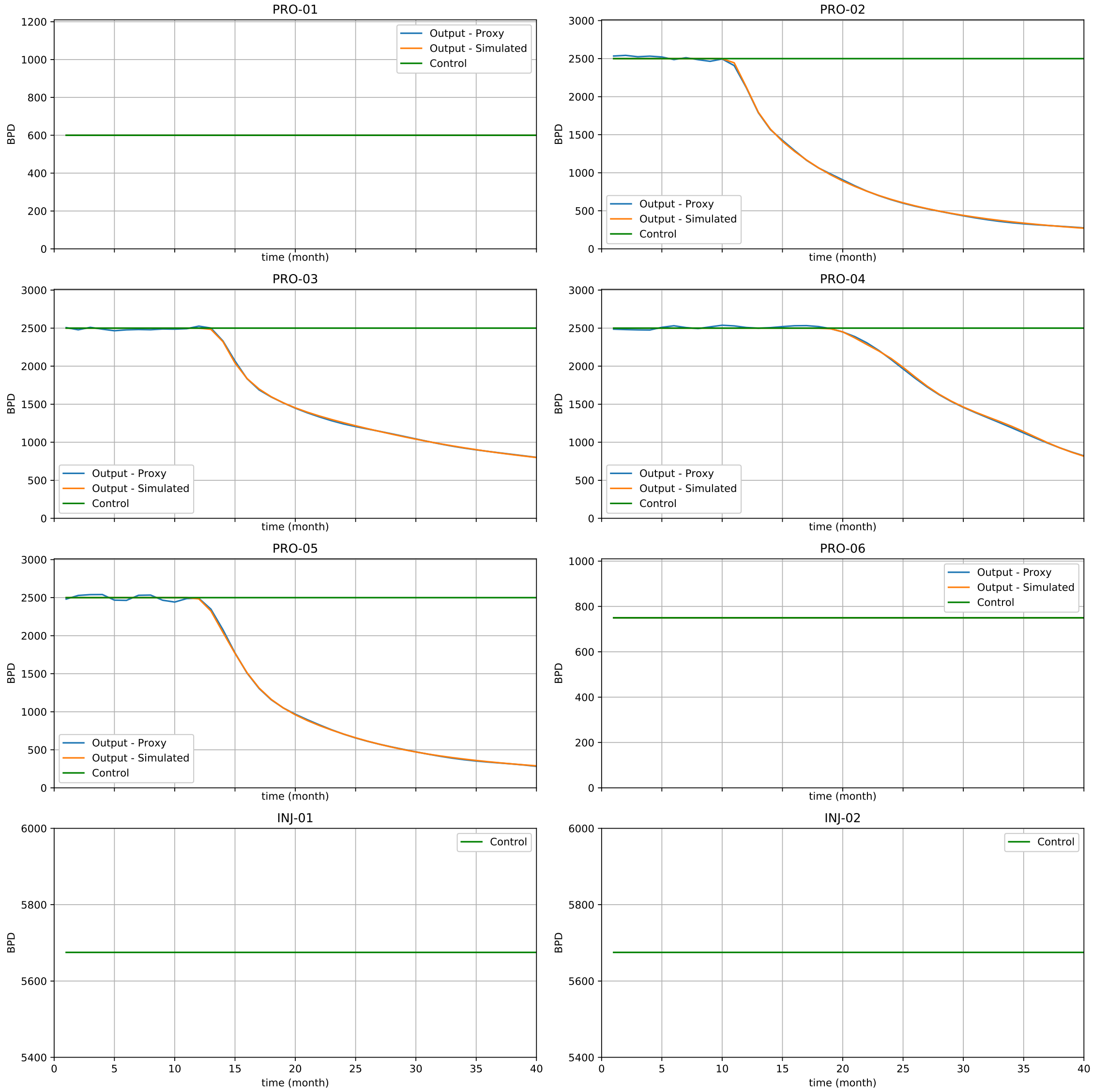

Figure 3.10: For all producer wells, the output variable is oil production rate. The figure compares the output of the proxy model with the simulated one. In green, the control (liquid production rate) resulted from the LP problem is shown in the case in which the box constraint allows zero deviation (Δ_u^0) from reference trajectory.

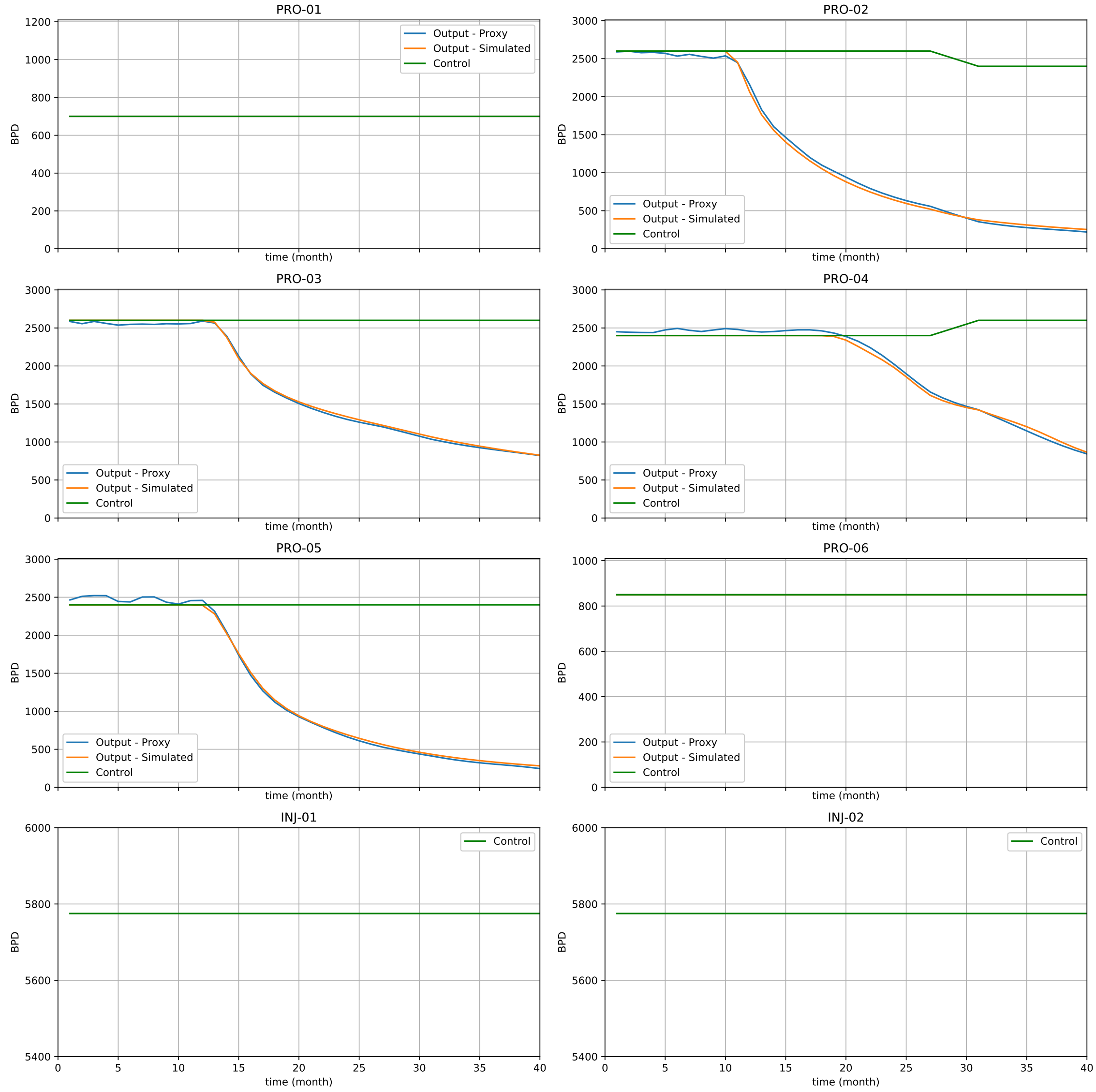

Figure 3.11: For all producer wells, the output variable is oil production rate. The figure compares the output of the proxy model with the simulated one. In green, the control (liquid production rate) resulted from the LP problem is shown in the case in which the box constraint allows a deviation of ±100 barrels per day (Δ_u^{100}) from reference trajectory.

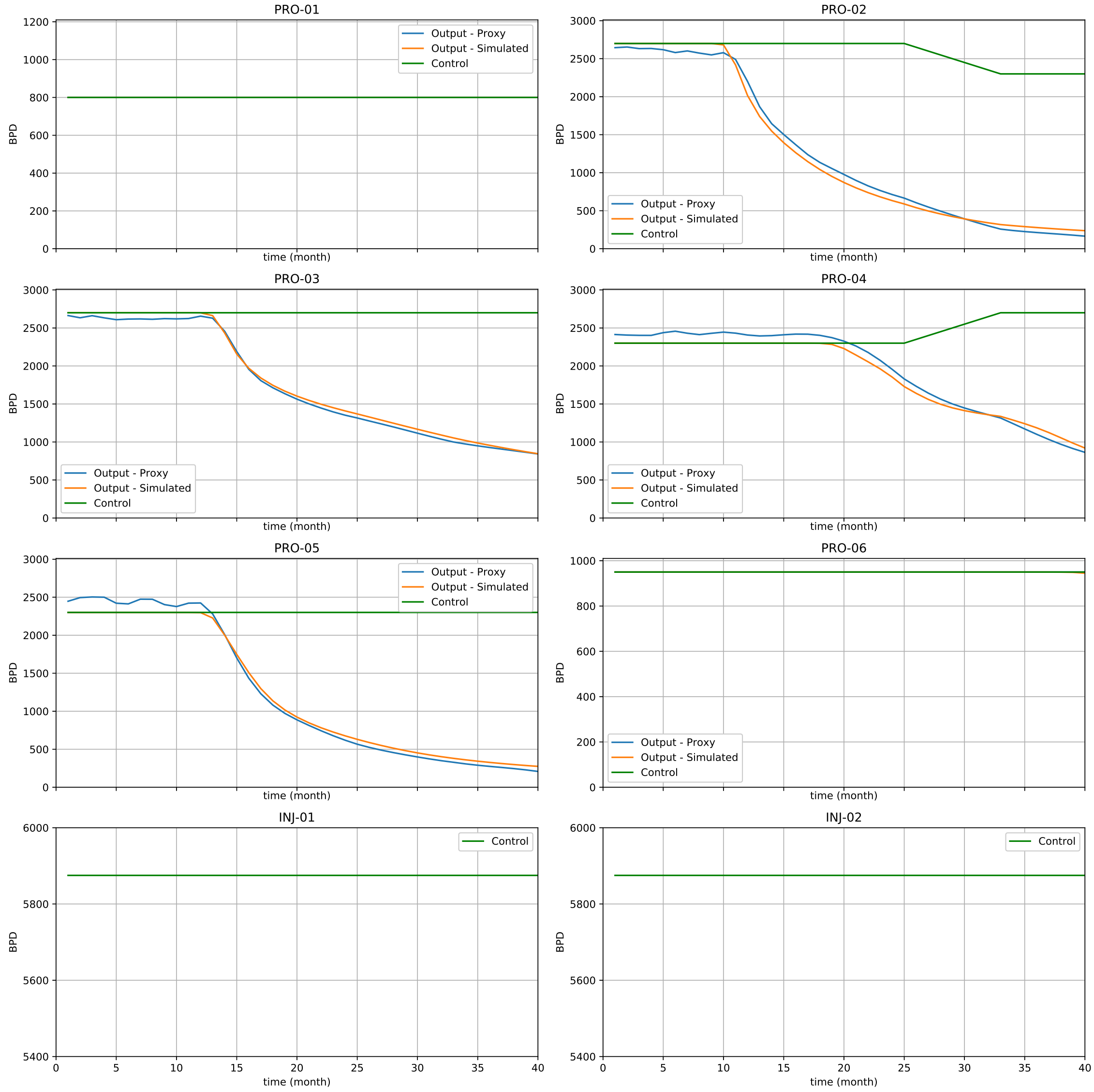

Figure 3.12: For all producer wells, the output variable is oil production rate. The figure compares the output of the proxy model with the simulated one. In green, the control (liquid production rate) resulted from the LP problem is shown in the case in which the box constraint allows a deviation of ±200 barrels per day (Δ_u^{200}) from reference trajectory.

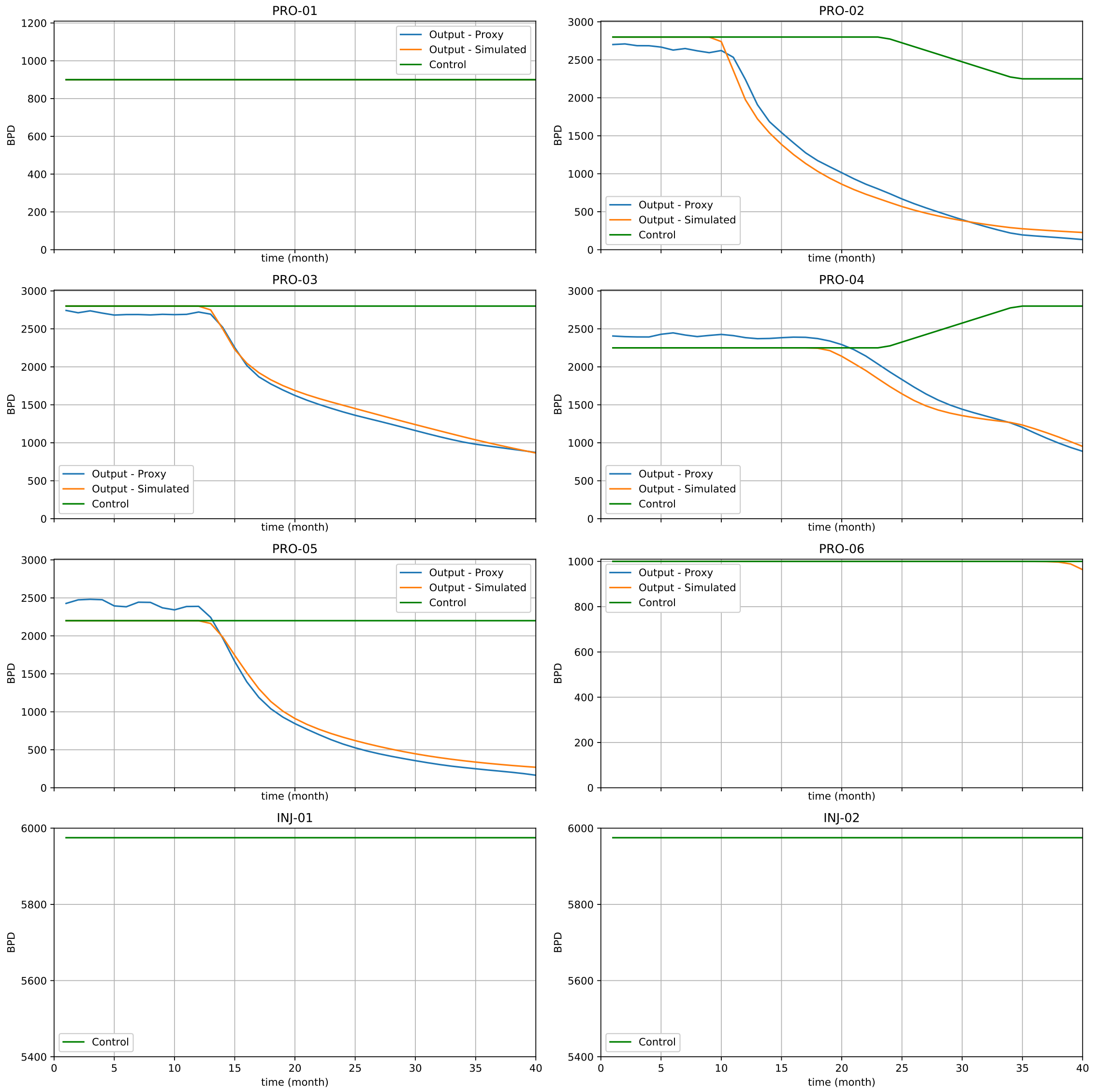

Figure 3.13: For all producer wells, the output variable is oil production rate. The figure compares the output of the proxy model with the simulated one. In green, the control (liquid production rate) resulted from the LP problem is shown in the case in which the box constraint allows a deviation of ±300 barrels per day (Δ_u^{300}) from reference trajectory. As the allowed deviation is larger, the proxy model tends to present larger deviations from the simulation model.

3.5.2 Optimization Results

As previously discussed, we aim to maximize the Net Present Value (NPV) by choice of the trajectory control. Table 3.1 presents the range chosen for each well: PRO-01 and PRO-06 ranges are different from the others due to their locally smaller permeability values.

The objective of this discussion is to present the results of the Trust Region algorithm (algorithm 1), showing how the controls can vary according to different capacity constraints (3.14).

The interest rate defined in this problem is 1% per month, O_p is 80 dollars per barrel, W_c is 10 dollar per barrel and I_c is zero. The slew rate (equation 3.18) is 50 BPD. Secondary controls (BHP bounds) are addressed internally in the simulation model using a standard specification.

Table 3.1: Liquid Rate Bounds (BPD)

Well	MIN	MAX
PRO-01	500	1200
PRO-02	2000	3000
PRO-03	2000	3000
PRO-04	2000	3000
PRO-05	2000	3000
PRO-06	500	1000
INJ-01	5000	8000
INJ-02	5000	8000

Let the initial condition be

$$\bar{u} = [600, 2500, 2500, 2500, 2500, 750, 5675, 5675],$$

with liquid capacity constraint equals 11350 BPD. We start from an initial condition respecting this constraint and we aim to find how liquid rates are balanced among producer wells, as well as the injector trajectory controls.

Figure 3.14 shows a fast evolution of the net present value in the first 5 iterations. The algorithm stopped due to the criterion $\Delta_{min} = [10, 10, 10, 10, 10, 10, 10, 10]$, which was reached in 15 iterations. The simulator was called 17 times: 15 times in line 8 of algorithm 1 and 2 times in line 4, where system identification procedure with PRBS excitation is performed.

NPV has increased by 9.10%, from 19.71 to 21.51 MM dollars. Cumulative oil production at month 40 evolved from 9.04 to 9.80 MM of barrels, which means an increase by 8.37% in the recovery factor.

Furthermore, the liquid capacity constraint is active during the whole period considered, which makes the problem more interesting. In fact, the optimization

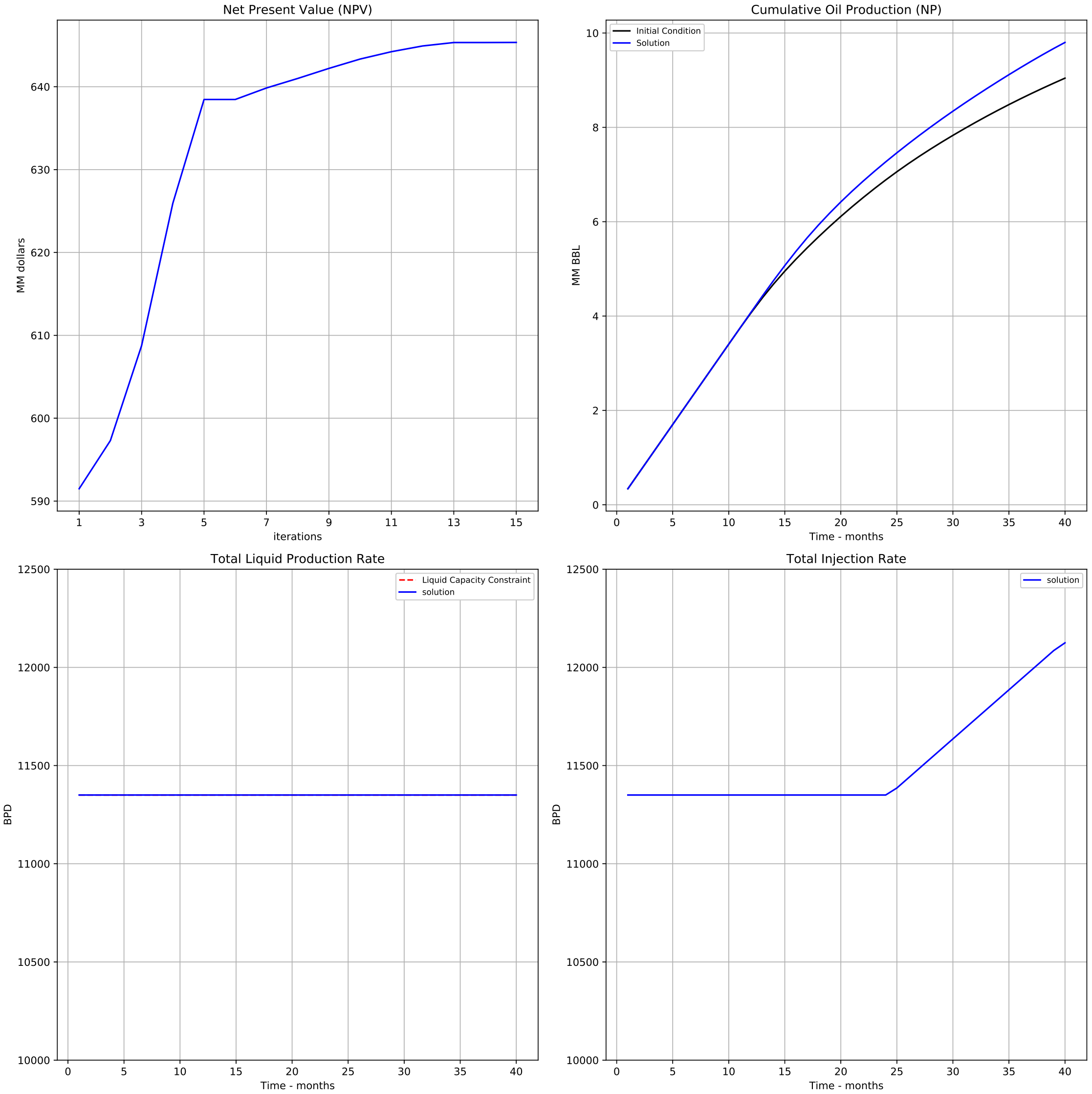

Figure 3.14: The first graph shows the rapid initial increase of the NPV due to the optimization algorithm achieving 9.1% overall gain. The second graph compares optimal cumulative oil production (blue curve) and initial solution (black curve). The third graph shows Total Liquid Production Rate attaining its upper bound (Liquid Capacity Constraint) and the fourth graph shows Total Injection Rate.

algorithm decides how to distribute the controls among the wells in order to optimize the reservoir water front efficiency as a consequence of maximizing the NPV. An important aspect of this methodology is to handle operational constraints in the optimization problem itself, not internally in the simulator, which may result in more significant gains.

Figures 3.15 and 3.16 show how the trajectory control evolves in the optimization procedure. The solution found for the wells PRO-01 and PRO-06 are stuck in the upper bound. This is quite reasonable due to the fact that their permeability are smaller, which makes them less efficient in terms of fluid production.

The wells PRO-02 and PRO-05 present the optimized trajectory control with smaller liquid rate control values than the initial one. This may be due to the existence of a preferred permeability path, as can be seen in figure 2.4. Because of this, figure 3.17 shows a breakthrough delay of 2 months for PRO-02 and PRO-05, which is desirable. Henceforth, the term water breakthrough delay will be used to refer to the delay in water breakthrough caused by the use of an optimal control. Thus, larger water breakthrough delays represent gains in oil production.

The wells PRO-03 and PRO-04 seem to balance each other such that liquid capacity constraint is reached. PRO-03 presents a smaller control value compared to the initial condition, which delays the water breakthough by 3 months. On the other hand, PRO-04 balances it and its water breakthrough occurs 4 months before in the optimal case. In spite of this, most wells presented a larger breakthrough delay in the improved case and their water cuts are kept smaller.

The injector wells respect an optional constraint, which was used in this work, namely that total injection rate is bigger than total liquid production. Interestingly, at time step 25, water injection in INJ-02 is increased, followed by a liquid production rate increase in PRO-04, which presents the smallest water cut among all producer wells of approximately 20%.

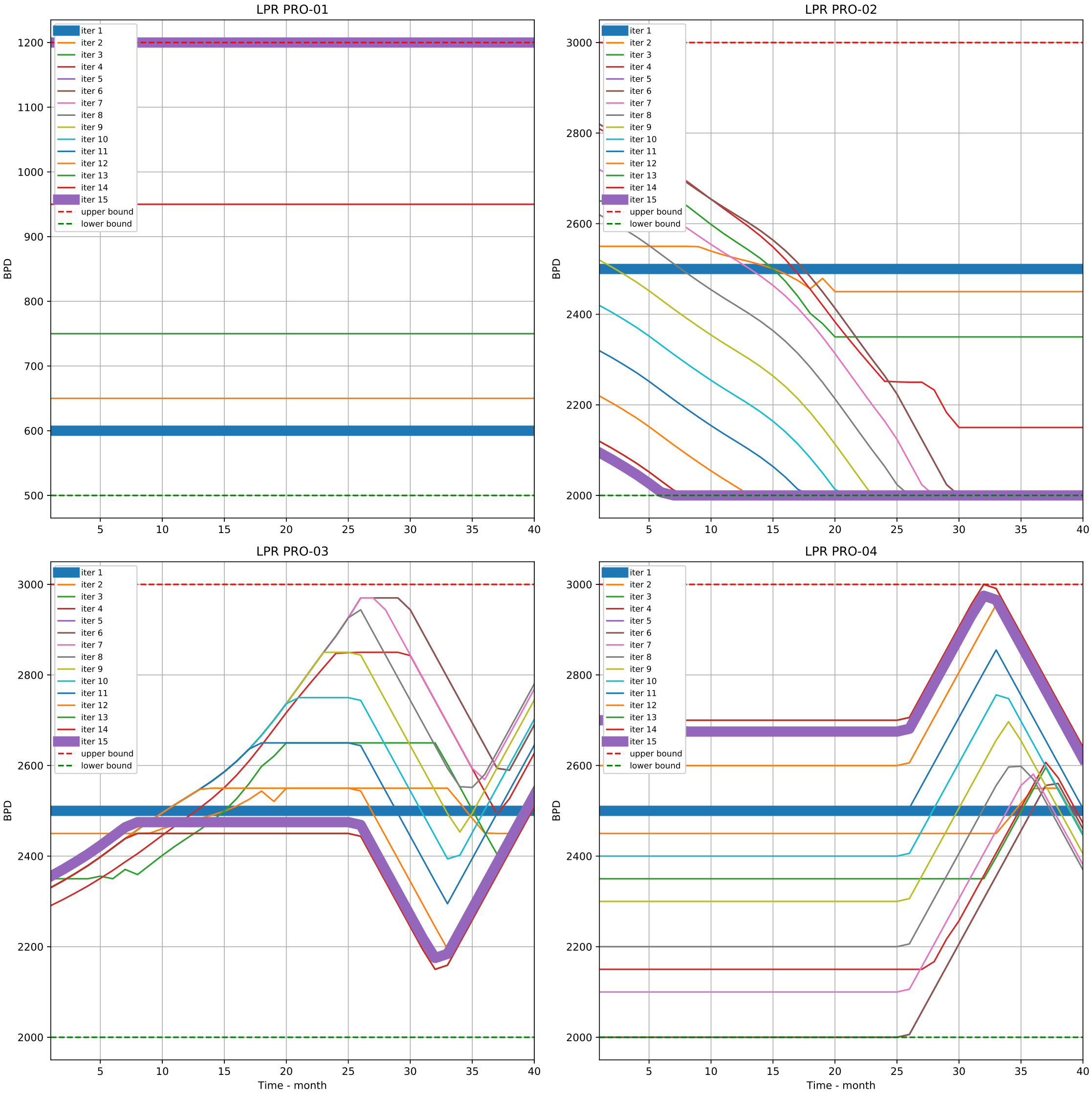

Figure 3.15: Figure shows the evolution of the trajectory controls (liquid production rate for producers PRO-01, PRO-02, PRO-03 and PRO-04) for all iterations in the trust region algorithm, from the initial condition (iter 1) to the final solution (iter 15).

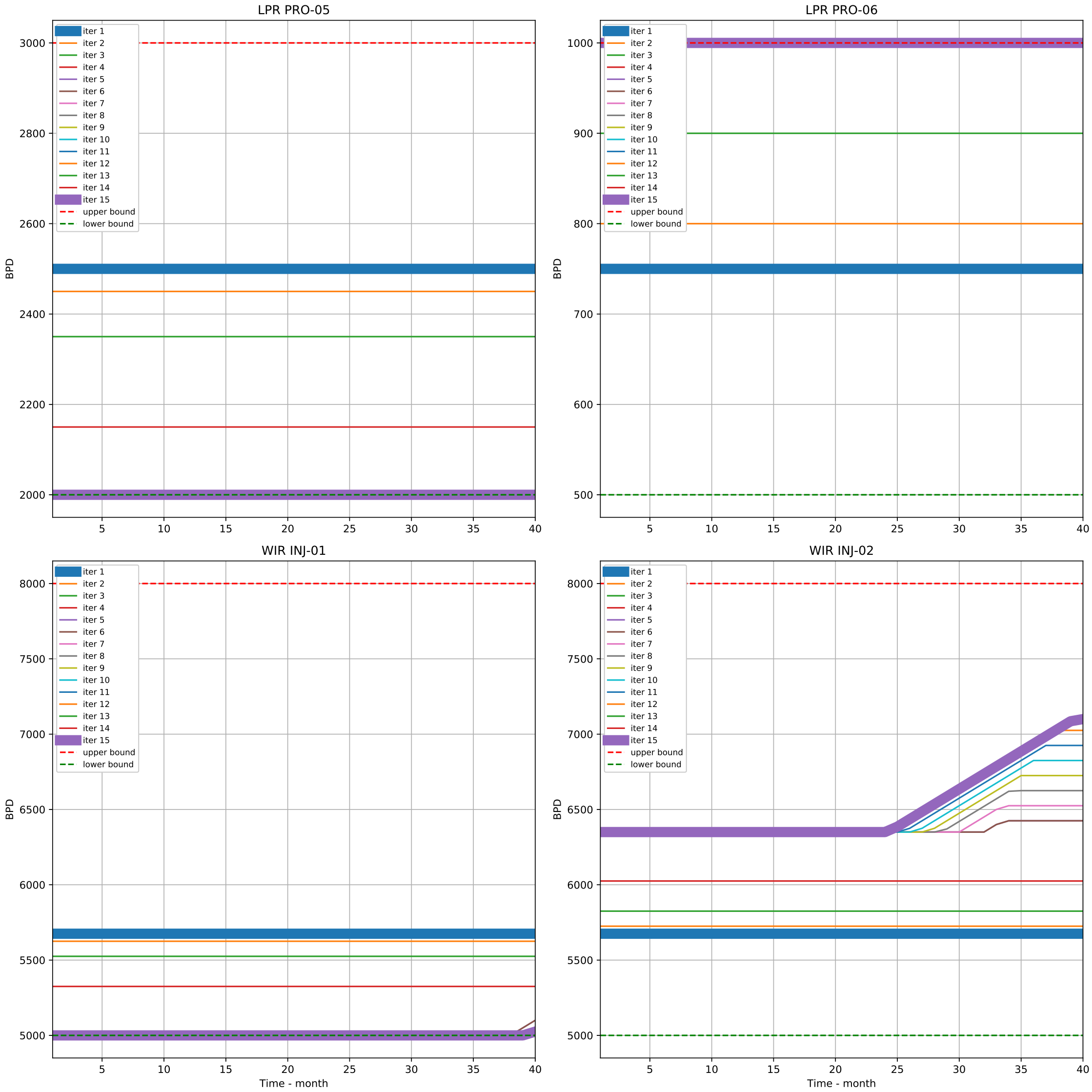

Figure 3.16: Figure shows the evolution of the trajectory control (liquid production rate for producers PRO-05 and PRO-06 and water injection rate for injector INJ-01 and INJ-02) for all iterations in the trust region algorithm, from the initial condition (iter 1) to the final solution (iter 15).

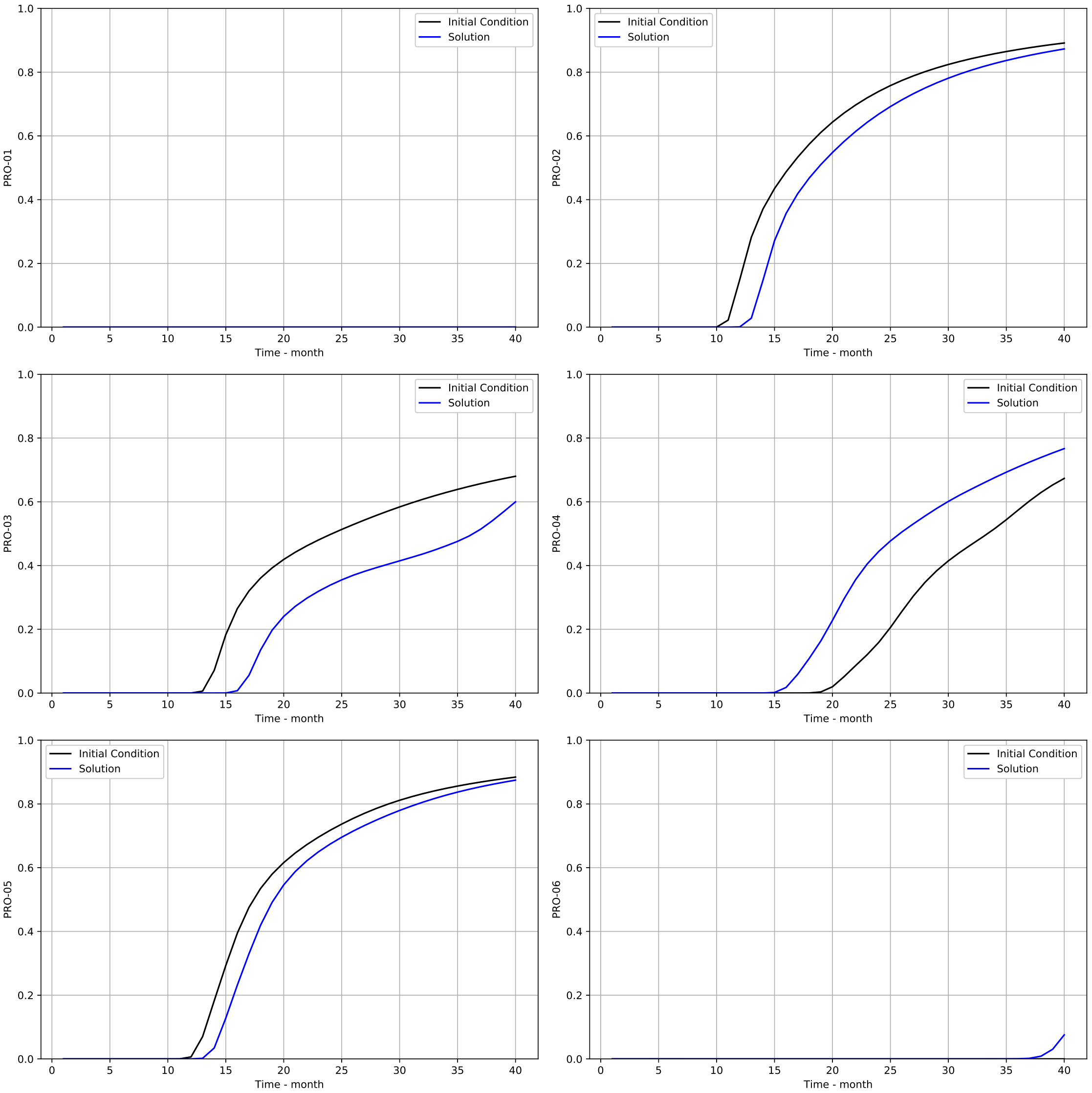

Figure 3.17: For a liquid capacity constraint of 11350 BPD, the graph indicates the water cut for all producer wells. For most wells, the solution derived from the trust region algorithm results in water breakthrough being delayed, which explains the gains in field cumulative oil production and the net present value (NPV).

Sensitivity Analysis Table 3.2 presents the results for sensitivity analysis in terms of total liquid capacity production. The last case (14200 BPD) represents an unconstrained case with respect to total liquid production because the sum of the upper bounds for all producer wells is 14200.

An advantage of this methodology is the fact that a significant NPV gain is reached in few iterations, with very little computational effort compared to derivative free algorithms or methods based on gradient estimation using ensembles (as related in [22], [23]).

In fact, for instance, in the unconstrained case, 18 evaluations of the objective function were required using the reservoir simulation model in 15 iterations. This means that the proxy model was re-assessed only 3 times, each of them requiring a new simulation, which implies that most of the consistent gains found in trust region method throughout the optimization process (see figure 3.14) were found with a single proxy model.

For the sake of comparison, ensemble optimization algorithm (EnOpt) requires a similar number of simulations used in column # of Simulations in table 3.2 in order to just estimate a descent direction.

In figure 3.19, for all cases, the control trajectories for PRO-01 and PRO-06 are stuck at the upper bound, which is reasonable because their water cuts are close to zero. Moreover, as the capacity constraint increases, the injector wells also inject more, and INJ-02 presents bigger injection rates than INJ-01 for all cases.

In terms of total liquid production rates, liquid capacity constraint is active in cases 11350, 12000 and 13000 BPD. Interestingly, in the unconstrained problem (14200 BPD), the solution found no longer reaches total production of 14200 BPD for the whole period considered in this study case.

Table 3.2: Sensitivity Analysis

Liquid Capacity (BPD)	NPV Gain (%)	# of Simulations	# of Iterations
11350	9.10	18	15
12000	10.55	28	24
13000	12.27	11	8
14200 (unconstrained)	13.31	13	10

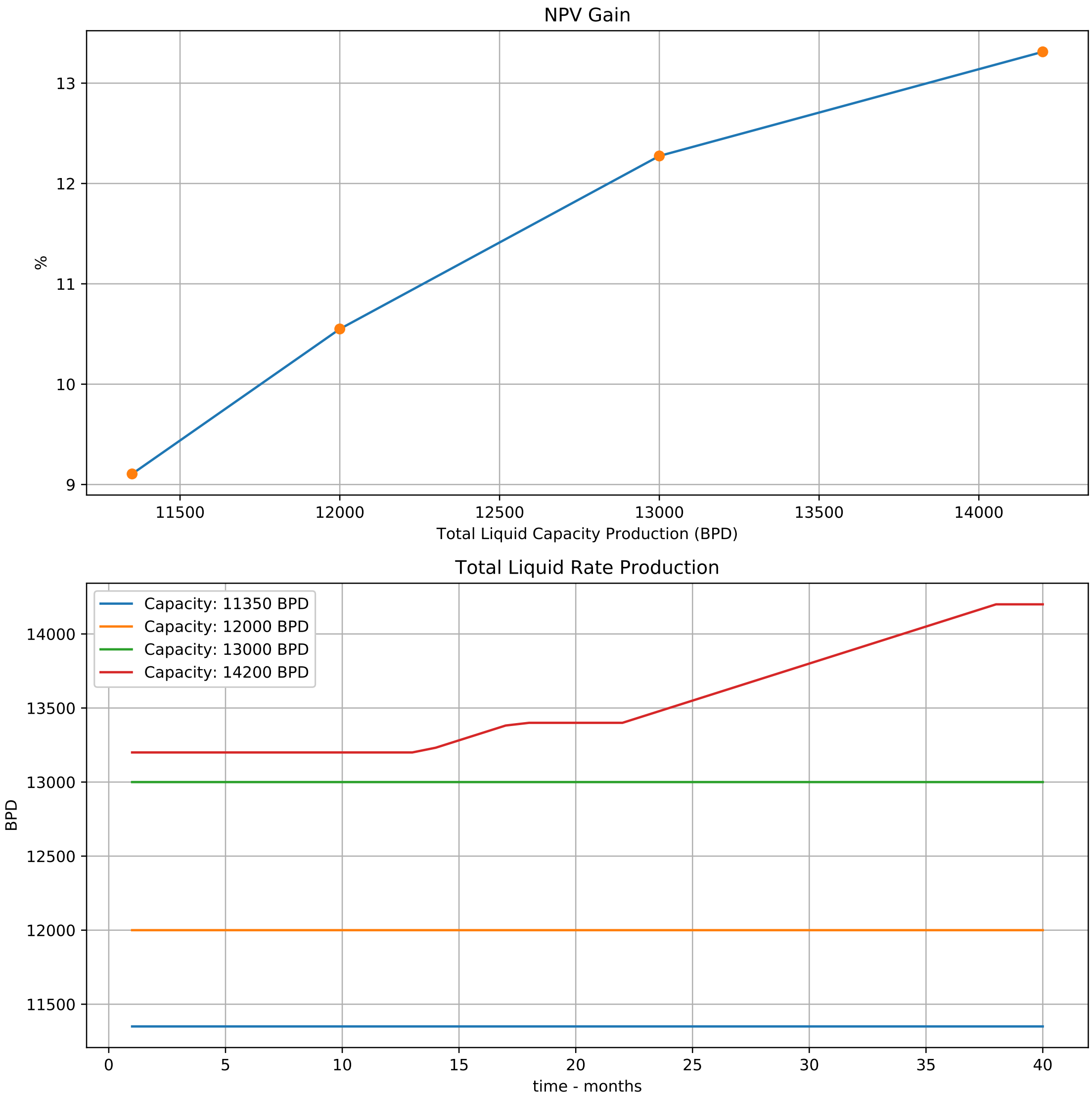

Figure 3.18: In the first graph, the sensitivity analysis shows the NPV Gain evolution when the Total Liquid Capacity constraint is increased. The second graph shows the total liquid production for different total liquid capacities. Interestingly, when the capacity is 14200 BPD, this constraint is not active during the entire period.

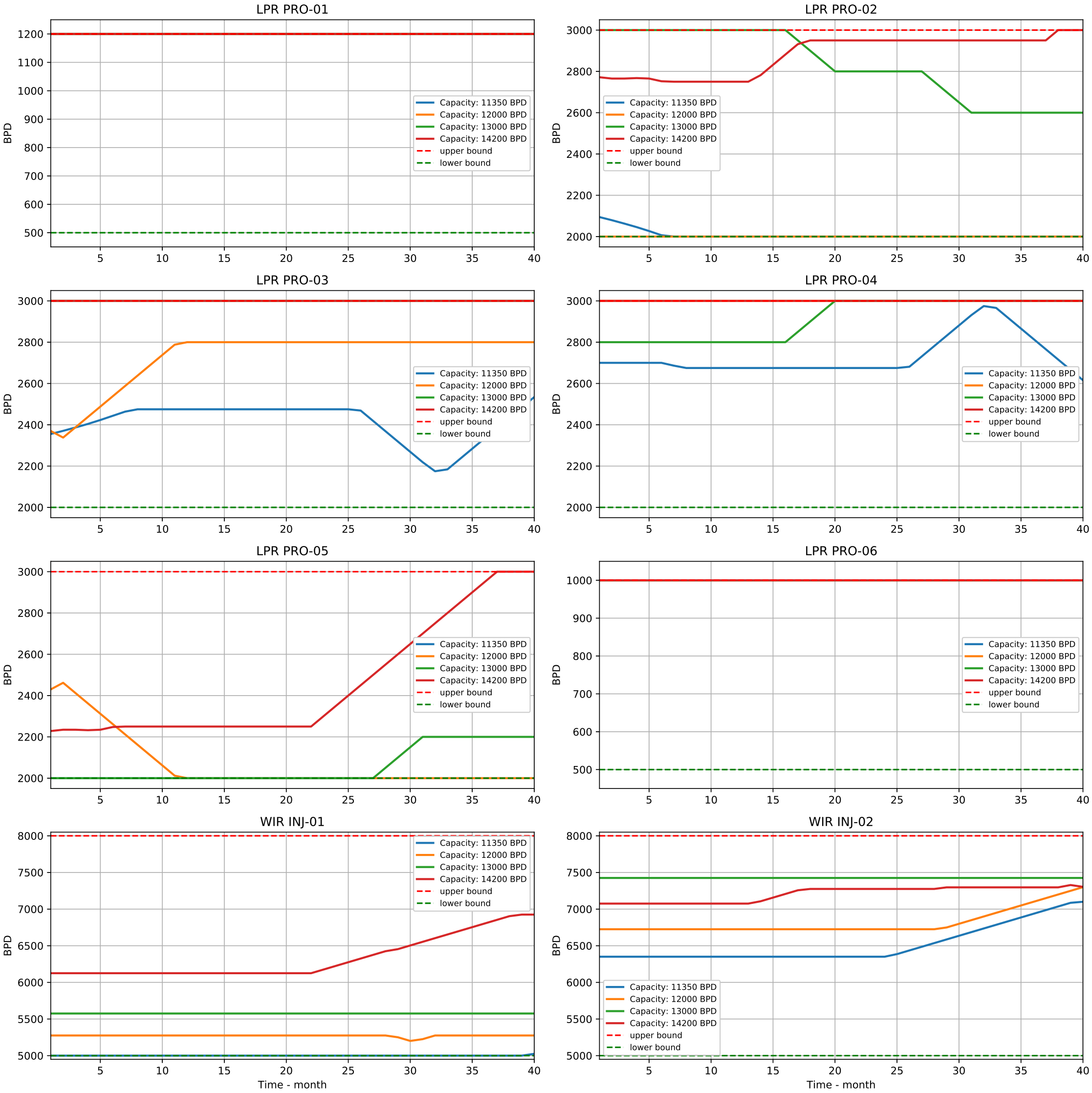

Figure 3.19: For all wells, both producers and injectors, the graph shows the trajectory controls (liquid production rate for producers and water injection rates for injectors) for different total liquid capacity constraints. Notice that the trajectory controls are not intuitively obvious, especially for different scenarios.

Chapter 4

Conclusions and Future Work

In chapter 2, polynomial structures ARX and ARMAX are presented. Based on the results presented, the main conclusions from this chapter are:

- From the test results, the ARX model structure and its uncertainty assessment provides acceptable results when it comes to forecasting.
- The gaussianity assumption about the model estimation is realistic, as far as the test set is concerned. Moreover, the results found by sampling in the ellipsoid of uncertainties with $\sigma = 3$ presents results very similar to Monte Carlo Simulation.
- No big improvements over the ARX model were found by using an ARMAX model in the presented case study, even though the residual analysis presents a better result when the noise is modelled.
- The step response provides a meaningful interpretation when it comes to establishing connections between the wells based only on the data driven model derived from system identification procedure.
- A drawback of modeling the water cut evolution using the RLS-ARX filter is the fact that water breakthrough is not predicted by the filter itself. In fact, the water cut signal does not reveal any value different from zero until the water breakthrough occurs. However, the filter presents quick adaptation and the results found in one year ahead prediction are quite similar to what the simulation model indicates, especially in the mature stage of field development, during which the filter needs to adapt less quickly.
- Forgetting factor choice in the RLS filter should be close enough to unity to preserve the filter stability.

- Prior knowledge used in the filter initial condition can improve its adaptation and it is a means by which physical information from a numerical simulation can contribute to the purely data driven approach.

The main topics for future work are as follows:

- Develop a closed-loop optimization framework, in which a filter (adaptive proxy model) can be used together with a MPC (model predictive controller) in order to maximize, for instance, the NPV.

- Use real observed field data and its measurement uncertainties, which were not available for this dissertation.

- Consider LASSO regression for feature selection, such that the regressors for each well are chosen automatically.

- Use more sophisticated structures, which may represent the entire range of simulation, such as more complex neural networks, Echo-State Networks, and so forth.

In chapter 3, we present a method that considers geometrical and physical aspects of a model simulation, by taking into account the state variables and attempting to describe them by a linear model. The main results of this chapter are:

- POD (or PCA) technique suitably reduces dimensionality, more easily for water saturation variables than block pressure in the case study used in this dissertation.

- The map reconstructions, as well as the output variables, tend to represent well the results found by the simulator using the trajectory controls derived from a LP problem in a region Δ_u around the current trajectory control, which means the proxy model is able to generate improved trajectory controls successfully.

- NPV gain from 9.10% was found using approximately 1% of the estimated number of simulations used in ensemble-based methods. Moreover, larger water breakthrough delays were observed in most wells in the optimal case. Notice that the optimal control methodology provides the balance among the producer rates such that the liquid capacity constraint is respected and this is a non intuitive result.

- The control trajectories can be easily implemented in daily field operations due to the slew rate constraint, which imposes smoothness in injection rates, which is highly desirable from an operational viewpoint.

As future work, there are many subjects worth studying:

- Identification of state space matrices promoting sparsity in the original dimension, with no dimensionality reduction.
- Use of prior knowledge about matrix structure, for instance, block pentadiagonal, or block diagonal matrices, as shown in CARDOSO [9].
- Use of the linear Kalman Filter for data assimilation and medium-term predictions.
- Use of non linear representations, in cases in which a linear model is unable to represent the dynamical system suitably.
- Use of the proposed method for a benchmark optimization case study of a larger problem.

Bibliography

[1] ZHOU, D. P., HU, Q., TOMLIN, C. J. "Quantitative Comparison of Data-Driven and Physics-Based Models for Commercial Building HVAC Systems", *American Control Conference*, v. 978-1-5090-5992-8, pp. 2900–2906, 2017.

[2] ARPS, J. "Analysis of decline curves", *Transactions of the AIME*, v. 160, n. 1, pp. 228–247, 1945.

[3] EVENSEN, G. "Sequential data assimilation with a nonlinear quasigeostrophic model using Monte Carlo Methods to forecast error statistics", *Journal of Geophysical Research*, v. 99, n. 65, pp. 10143–10162, 1994.

[4] VAN LEEUWEN, P. J., EVENSEN, G. "Data Assimilation and Inverse Methods in Terms of a Probabilistic Formulation", *Monthly Weather Review*, v. 124, pp. 2898–2913, 1996.

[5] EMERICK, A. A., REYNOLDS, A. C. "Investigation on the Sampling Performance of Ensemble-based Methods with a Simple Reservoir Model", *Computational Geosciences*, v. 17, pp. 325–350, 2013.

[6] NEGASH, B. M., TUFA, L. D., RAMASAMY, M., et al. "System Identification Based Proxy Model of a Reservoir under Water Injection", *Modelling and Simulation in Engineering*, 2017.

[7] HOURFAR, F. "Adaptive modeling of waterflooding process in oil reservoirs", *Journal of Petroleum Science and Engineering*, 2016.

[8] WEBER, D. *The Use of Capacitance-Resistance Models to Optimize Injection Allocation and Well Location in Water Floods.* PhD thesis, The University of Texas at Austin, Austin, Texas, USA, 2009.

[9] CARDOSO, M. A. *Development and application of reduced-order modeling procedures for reservoir simulation.* PhD thesis, Stanford University, 2009.

[10] AGUIRRE, L. A. *Introdução à Identificação de Sistemas.* 4 ed. Belo Horizonte, editora UFMG, 2015.

[11] LORENTZEN, R. J., BERG, A., NAEVDAL, G., et al. "A New Approach for Dynamic Optimization of Waterflooding Problems", *Proceedings of the SPE Intelligent Energy Conference and Exhibition*, , n. SPE 99690, 2006.

[12] CHEN, Y., OLIVER, D. S., ZHANG, D. "Efficient Ensemble-Based Closed-Loop Production Optimization", *SPE Journal*, v. 14, n. 4, pp. 634–645, 2009.

[13] CONN, A. R., GOULD, N. I. M., TOINT, PH. L. *Trust-Region Methods*. Philadelphia, PA, USA, SIAM, 2000.

[14] FRAGOSO, M. J. *Aplicações de um modelo substituto de ordem reduzida a estudos de gerenciamento de reservatórios de petróleo*. M.Sc. dissertation, UFPE, 2014.

[15] DIAMOND, S., BOYD, S. "CVXPY: A Python-Embedded Modeling Language for Convex Optimization", *Journal of Machine Learning Research*, v. 17, n. 83, pp. 1–5, 2016.

[16] LÓPEZ, C. I. *Implementation of optimization-based controllers for industrial processes*. M.Sc. dissertation, Escola Tècnica Superior d'Enginyeria Industrial de Barcelona, 2017.

[17] STRANG, G. *Linear Algebra and its applications*. 2 ed. New York, Cengage Learning, 2010.

[18] BILLINGS, S. A. *Nonlinear System Identification: NARMAX Methods in the Time, Frequency, and Spatio–Temporal Domains*. New York, John Wiley Sons, Ltd., 2013.

[19] MOSTAFA, Y. A., MAGDON-ISMAIL, M., LIN, H.-T. *Learning from Data*. AMLBook, 2012.

[20] EMERICK, A. A. "Theoretical Fundamentals of Assisted History Matching", Course notes, 2018.

[21] PINNAU, R. "Model Reduction via Proper Orthogonal Decomposition". In: Wilhelmus H.A. Schilders, Henk A. van der Vorst, J. R. (Ed.), *Model Order Reduction: Theory, Research Aspects and Applications*, Berlin, Germany, Springer-Verlag, 2008.

[22] SILVA, V. L. S. *Ajuste de Histórico e Otimização da Produção sob Incertezas - Aplicação do Gerenciamento de Reservatórios em Malha Fechada*. M.Sc. dissertation, UFRJ/COPPE, 2016.

[23] OLIVEIRA, D. F. B. *A New Hierarchical Multiscale Optimization Method: Gradient and non-gradient approaches for Waterflooding Optimization.* PhD thesis, The University of Tulsa, 2014.